Informatica®
10.1

Developer-Umwandlungen – Leitfaden

Informatica Developer-Umwandlungen – Leitfaden
10.1
Juni 2016

© Copyright Informatica LLC 2009, 2018

Diese Software und die zugehörige Dokumentation enthalten proprietäre Informationen der Informatica LLC, werden unter einem Lizenzvertrag mit Einschränkungen hinsichtlich Verwendung und Veröffentlichung zur Verfügung gestellt und sind urheberrechtlich geschützt. Das Zurückentwickeln (Reverse Engineering) der Software ist untersagt. Ohne ausdrückliche schriftliche Genehmigung der Informatica LLC darf kein Teil dieses Dokuments zu irgendeinem Zweck vervielfältigt oder übertragen werden, unabhängig davon, auf welche Art und Weise oder mit welchen Mitteln (elektronisch, mechanisch, durch Fotokopieren, Aufzeichnen usw.) dies geschieht. Diese Software ist möglicherweise durch US-amerikanische und/oder internationale Patente und weitere angemeldete Patente geschützt.

Die Verwendung, Vervielfältigung oder Veröffentlichung der Software durch die US-Regierung unterliegt den Bestimmungen des jeweiligen Softwarelizenzvertrags sowie ggf. den Bestimmungen in DFARS 227.7202-1(a) und 227.7702-3(a) (1995), DFARS 252.227-7013 © (1)(ii) (OCT. 1988), FAR 12.212(a) (1995), FAR 52.227-19 oder FAR 52.227-14 (ALT III).

Die in diesem Produkt und in dieser Dokumentation enthaltenen Informationen können jederzeit ohne vorherige Ankündigung geändert werden. Sollten Sie mit diesem Produkt oder dieser Dokumentation Probleme haben, teilen Sie uns dies bitte schriftlich mit.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging, Informatica Master Data Management und Live Data Map sind Marken oder eingetragene Marken der Informatica LLC in den USA und anderen Ländern. Alle weiteren Produkt- und Firmennamen sind möglicherweise Markennamen oder Warenzeichen der jeweiligen Eigentümer.

Teile dieser Software und/oder Dokumentation sind durch die Urheberrechte Dritter geschützt, einschließlich und ohne Einschränkung: Copyright DataDirect Technologies. Alle Rechte vorbehalten. Copyright © Sun Microsystems. Alle Rechte vorbehalten. Copyright © RSA Security Inc. Alle Rechte vorbehalten. Copyright © Ordinal Technology Corp. Alle Rechte vorbehalten. Copyright © Aandacht c.v. Alle Rechte vorbehalten. Copyright Genivia, Inc. Alle Rechte vorbehalten. Copyright Isomorphic Software. Alle Rechte vorbehalten. Copyright © Meta Integration Technology, Inc. Alle Rechte vorbehalten. Copyright © Intalio. Alle Rechte vorbehalten. Copyright © Oracle. Alle Rechte vorbehalten. Copyright © Adobe Systems Incorporated. Alle Rechte vorbehalten. Copyright © DataArt, Inc. Alle Rechte vorbehalten. Copyright © ComponentSource. Alle Rechte vorbehalten. Copyright © Microsoft Corporation. Alle Rechte vorbehalten. Copyright © Rouge Wave Software, Inc. Alle Rechte vorbehalten. Copyright © Teradata Corporation. Alle Rechte vorbehalten. Copyright © Yahoo! Inc. Alle Rechte vorbehalten. Copyright © Glyph & Cog, LLC. Alle Rechte vorbehalten. Copyright © Thinkmap, Inc. Alle Rechte vorbehalten. Copyright © Clearpace Software Limited. Alle Rechte vorbehalten. Copyright © Information Builders, Inc. Alle Rechte vorbehalten. Copyright © OSS Nokalva, Inc. Alle Rechte vorbehalten. Copyright Edifecs, Inc. Alle Rechte vorbehalten. Copyright Cleo Communications, Inc. Alle Rechte vorbehalten. Copyright © International Organization for Standardization 1986. Alle Rechte vorbehalten. Copyright © ej-technologies GmbH . Alle Rechte vorbehalten. Copyright © Jaspersoft Corporation. Alle Rechte vorbehalten. Copyright © International Business Machines Corporation. Alle Rechte vorbehalten. Copyright © yWorks GmbH. Alle Rechte vorbehalten. Copyright © Lucent Technologies. Alle Rechte vorbehalten. Copyright © Universität von Toronto. Alle Rechte vorbehalten. Copyright © Daniel Veillard. Alle Rechte vorbehalten. Copyright © Unicode, Inc. Copyright IBM Corp. Alle Rechte vorbehalten. Copyright © MicroQuill Software Publishing, Inc. Alle Rechte vorbehalten. Copyright © PassMark Software Pty Ltd. Alle Rechte vorbehalten. Copyright © LogiXML, Inc. Alle Rechte vorbehalten. Copyright © 2003-2010 Lorenzi Davide. Alle Rechte vorbehalten. Copyright © Red Hat, Inc. Alle Rechte vorbehalten. Copyright © The Board of Trustees of the Leland Stanford Junior University. Alle Rechte vorbehalten. Copyright © EMC Corporation. Alle Rechte vorbehalten. Copyright © Flexera Software. Alle Rechte vorbehalten. Copyright © Jinfonet Software. Alle Rechte vorbehalten. Copyright © Apple Inc. Alle Rechte vorbehalten. Copyright © Telerik Inc. Alle Rechte vorbehalten. Copyright © BEA Systems. Alle Rechte vorbehalten. Copyright © PDFlib GmbH. Alle Rechte vorbehalten. Copyright © Orientation in Objects GmbH. Alle Rechte vorbehalten. Copyright © Tanuki Software, Ltd. Alle Rechte vorbehalten. Copyright © Ricebridge. Alle Rechte vorbehalten. Copyright © Sencha, Inc. Alle Rechte vorbehalten. Copyright © Scalable Systems, Inc. Alle Rechte vorbehalten. Copyright © jQWidgets. Alle Rechte vorbehalten. Copyright © Tableau Software, Inc. Alle Rechte vorbehalten. Copyright© MaxMind, Inc. Alle Rechte vorbehalten. Copyright © TMate Software s.r.o. Alle Rechte vorbehalten. Copyright © MapR Technologies Inc. Alle Rechte vorbehalten. Copyright © Amazon Corporate LLC. Alle Rechte vorbehalten. Copyright © Highsoft. Alle Rechte vorbehalten. Copyright © Python Software Foundation. Alle Rechte vorbehalten. Copyright © BeOpen.com. Alle Rechte vorbehalten. Copyright © CNRI. Alle Rechte vorbehalten.

Dieses Produkt enthält Software, die von der Apache Software Foundation (http://www.apache.org/) entwickelt wurde, und andere Software, die unter den Bedingungen des Apache-Lizenzvertrags lizenziert ist („Lizenz“). Eine Kopie dieser Lizenzen finden Sie unter http://www.apache.org/licenses/. Sofern nicht gesetzlich vorgeschrieben oder schriftlich vereinbart, erfolgt der Vertrieb der Software unter der Lizenz auf der BASIS „WIE BESEHEN” OHNE GARANTIEN ODER KONDITIONEN IRGENDEINER ART, weder ausdrücklich noch impliziert. Berechtigungen und Einschränkungen für bestimmte Sprachen finden Sie in der Lizenz.

Dieses Produkt enthält Software, die von Mozilla (http://www.mozilla.org/) entwickelt wurde, Software Copyright The JBoss Group, LLC. Alle Rechte vorbehalten; Software Copyright © 1999-2006 by Bruno Lowagie und Paulo Soares, und andere Software, die gemäß den verschiedenen Versionen des GNU Lesser General Public License Agreement unter http://www.gnu.org/licenses/lgpl.html lizenziert ist. Die Materialien werden „wie besehen“ kostenlos von Informatica bereitgestellt, ohne ausdrückliche oder stillschweigende Gewährleistung, einschließlich, jedoch nicht beschränkt auf die stillschweigenden Gewährleistungen der Handelsüblichkeit und der Eignung für einen bestimmten Zweck.

Das Produkt enthält ACE(TM) und TAO(TM) Software, Copyright Douglas C. Schmidt und seine Forschungsgruppe an der Washington University, University of California, Irvine und Vanderbilt University, Copyright (©) 1993-2006. Alle Rechte vorbehalten.

Dieses Produkt enthält Software, die von OpenSSL Project zur Verwendung im OpenSSL Toolkit entwickelt wurde (Copyright The OpenSSL Project. Alle Rechte vorbehalten). Die erneute Verteilung dieser Software unterliegt den unter „http://www.openssl.org“ und „http://www.openssl.org/source/license.html“ verfügbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte Curl-Software (Copyright 1996-2013, Daniel Stenberg, <daniel@haxx.se>). Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://curl.haxx.se/docs/copyright.html“ verfügbaren Bedingungen. Die Erlaubnis, diese Software für jeden beliebigen Zweck gegen Gebühr oder kostenlos zu verwenden, zu kopieren, zu ändern und zu verteilen, wird hiermit erteilt, sofern die oben genannten urheberrechtlichen Hinweise und diese Erlaubnis in allen Exemplaren angegeben werden.

Das Produkt enthält urheberrechtlich geschützte Software, Copyright 2001-2005 (©) MetaStuff, Ltd. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://www.dom4j.org/ license.html“ verfügbaren Bedingungen.

Das Produkt enthält urheberrechtlich geschützte Software, Copyright © 2004-2007, The Dojo Foundation. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://dojotoolkit.org/license“ verfügbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte ICU-Software, Copyright International Business Machines Corporation und andere. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://source.icu-project.org/repos/icu/icu/trunk/license.html“ verfügbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte Software, Copyright © 1996-2006 Per Bothner. Alle Rechte vorbehalten. Das Ihnen erteilte Recht, diese Materialien zu verwenden, unterliegt den unter „http:// www.gnu.org/software/ kawa/Software-License.html“ verfügbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte OSSP UUID-Software (Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland). Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://www.opensource.org/licenses/mit-license.php“ verfügbaren Bedingungen.

Dieses Produkt enthält Software, die von Boost (http://www.boost.org/) oder unter der Softwarelizenz von Boost entwickelt wurde. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://www.boost.org/LICENSE_1_0.txt“ verfügbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte Software, Copyright © 1997-2007 University of Cambridge. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter http://www.pcre.org/license.txt einsehbaren Bedingungen.

Dieses Produkt enthält urheberrechtlich geschützte Software, Copyright © 2007 The Eclipse Foundation. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://www.eclipse.org/org/documents/epl-v10.php“ und „http://www.eclipse.org/org/documents/edl-v10.php“ verfügbaren Bedingungen.

Dieses Produkt enthält Software gemäß den Lizenzbedingungen unter http://www.tcl.tk/software/tcltk/license.html, http://www.bosrup.com/web/overlib/?License, http://www.stlport.org/doc/ license.html, http://asm.ow2.org/license.html, http://www.cryptix.org/LICENSE.TXT, http://hsqldb.org/web/hsqlLicense.html, http://httpunit.sourceforge.net/doc/ license.html, http://jung.sourceforge.net/license.txt, http://www.gzip.org/zlib/zlib_license.html, http://www.openldap.org/software/release/license.html, http://www.libssh2.org, http://slf4j.org/license.html, http://www.sente.ch/software/OpenSourceLicense.html, http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3- license-agreement; http://antlr.org/license.html; http://aopalliance.sourceforge.net/; http://www.bouncycastle.org/licence.html; http://www.jgraph.com/jgraphdownload.html; http://www.jcraft.com/jsch/LICENSE.txt; http://jotm.objectweb.org/bsd_license.html. http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231; http://www.slf4j.org/license.html; http://nanoxml.sourceforge.net/orig/copyright.html; http://www.json.org/license.html; http://forge.ow2.org/projects/javaservice/, http://www.postgresql.org/about/licence.html, http://www.sqlite.org/copyright.html, http://www.tcl.tk/software/tcltk/license.html, http://www.jaxen.org/faq.html, http://www.jdom.org/docs/faq.html, http://www.slf4j.org/license.html; http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License; http://www.keplerproject.org/md5/license.html; http://www.toedter.com/en/jcalendar/license.html; http://www.edankert.com/bounce/index.html; http://www.net-snmp.org/about/license.html; http://www.openmdx.org/#FAQ; http://www.php.net/license/3_01.txt; http://srp.stanford.edu/license.txt; http://www.schneier.com/blowfish.html; http://www.jmock.org/license.html; http://xsom.java.net; http://benalman.com/about/license/; https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js; http://www.h2database.com/html/license.html#summary; http://jsoncpp.sourceforge.net/LICENSE; http://jdbc.postgresql.org/license.html; http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto; https://github.com/rantav/hector/blob/master/LICENSE; http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html; http://jibx.sourceforge.net/jibx-license.html; https://github.com/lyokato/libgeohash/blob/master/LICENSE; https://github.com/hjiang/jsonxx/blob/master/LICENSE; https://code.google.com/p/lz4/; https://github.com/jedisct1/libsodium/blob/master/LICENSE; http://one-jar.sourceforge.net/index.php?page=documents&file=license; https://github.com/EsotericSoftware/kryo/blob/master/license.txt; http://www.scala-lang.org/license.html; https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt; http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html; https://aws.amazon.com/asl/; https://github.com/twbs/bootstrap/blob/master/LICENSE; https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt.

Dieses Produkt enthält Software, die unter der Academic Free License (http://www.opensource.org/licenses/afl-3.0.php), der Common Development Distribution License (http://www.opensource.org/licenses/cddl1.php), der Common Public License (http://www.opensource.org/licenses/cpl1.0.php), den Sun Binary Code License Agreement Supplemental License Terms, der BSD License (http:// www.opensource.org/licenses/bsd-license.php), der neuen BSD License (http://opensource.org/licenses/BSD-3-Clause), der MIT License (http://www.opensource.org/licenses/mit-license.php), der Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) und der Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/) lizenziert ist.

Dieses Produkt enthält urheberrechtlich geschützte Software, Copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den unter „http://xstream.codehaus.org/license.html“ verfügbaren Bedingungen. Dieses Produkt enthält Software, die von der Indiana University Extreme! Lab. entwickelt wurde. Weitere Informationen finden Sie unter http://www.extreme.indiana.edu/.

Dieses Produkt enthält Software, Copyright © 2013 Frank Balluffi und Markus Moeller. Alle Rechte vorbehalten. Die mit dieser Software verbundenen Berechtigungen und Einschränkungen unterliegen den Bedingungen der MIT-Lizenz.

Weitere Informationen über die Patente finden Sie unter https://www.informatica.com/legal/patents.html.

HAFTUNGSAUSSCHLUSS: Informatica LLC stellt diese Dokumentation „wie besehen“ bereit, ohne ausdrückliche oder stillschweigende Gewährleistung, einschließlich, jedoch nicht beschränkt auf die Gewährleistungen der Nichtverletzung der Rechte von Dritten, der Handelsüblichkeit oder Eignung für einen bestimmten Zweck. Informatica LLC garantiert nicht die Fehlerfreiheit dieser Software oder Dokumentation. Die in dieser Software oder Dokumentation bereitgestellten Informationen können technische Ungenauigkeiten oder Druckfehler enthalten. Die in dieser Software und in dieser Dokumentation enthaltenen Informationen können jederzeit ohne vorherige Ankündigung geändert werden.

HINWEISE

Dieses Informatica-Produkt (die „Software“) umfasst bestimmte Treiber (die „DataDirect-Treiber“) von DataDirect Technologies, einem Betreiber von Progress Software Corporation („DataDirect“), die folgenden Bedingungen und Bestimmungen unterliegen:

1. DIE DATADIRECT-TREIBER WERDEN „WIE GESEHEN“ OHNE JEGLICHE GEWÄHRLEISTUNG, WEDER AUSDRÜCKLICH NOCH STILLSCHWEIGEND, BEREITGESTELLT, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF DIE STILLSCHWEIGENDEN GEWÄHRLEISTUNGEN DER HANDELSÜBLICHKEIT, EIGNUNG FÜR EINEN BESTIMMTEN ZWECK UND DER NICHTVERLETZUNG VON RECHTEN DRITTER.
2. IN KEINEM FALL SIND DATADIRECT ODER DRITTANBIETER DEM ENDBENUTZER GEGENÜBER HAFTBAR FÜR UNMITTELBARE, MITTELBARE, KONKRETE, NEBEN-, FOLGE- ODER ANDERE SCHÄDEN, DIE SICH AUS DER VERWENDUNG DER ODBC-TREIBER ERGEBEN, UNABHÄNGIG DAVON, OB SIE IM VORAUS ÜBER DIE MÖGLICHKEIT SOLCHER SCHÄDEN INFORMIERT WORDEN SIND ODER NICHT. DIESE BESCHRÄNKUNGEN GELTEN FÜR ALLE KLAGEGEGENSTÄNDE, EINSCHLIESSLICH, JEDOCH NICHT BESCHRÄNKT AUF VERTRAGSBRUCH, GEWÄHRLEISTUNGSBRUCH, FAHRLÄSSIGKEIT, KAUSALHAFTUNG, TÄUSCHUNG UND ANDERE UNERLAUBTE HANDLUNGEN.

Publikationsdatum: 2018-07-02

Inhalt

Einleitung 28

Informatica-Ressourcen. 28

Informatica-Netzwerk. 28

Informatica-Wissensdatenbank. 28

Informatica-Dokumentation. 28

Informatica-Produktverfügbarkeitsmatrizen. 29

Informatica Velocity. 29

Informatica Marketplace. 29

Globaler Kundensupport von Informatica. 29

Kapitel 1: Einführung in Umwandlungen. 30

Einführung in Umwandlungen - Übersicht. 30

Aktive Umwandlungen. 30

Passive Umwandlungen. 31

Nicht verbundene Umwandlungen. 31

Mehrfachstrategieumwandlungen. 31

Umwandlungsbeschreibungen. 32

Verarbeiten von Umwandlungsdatentypen. 34

Datentyp „Dezimal“. 35

Zeitstempel mit Zeitzone. 36

Zeitstempel mit lokaler Zeitzone. 37

Entwickeln einer Umwandlung. 37

Umwandlungsports. 37

Erstellen von Ports. 38

Konfigurieren von Ports. 38

Multi-Gruppen-Umwandlungen. 38

Regeln und Richtlinien für Multi-Gruppen-Umwandlungen. 39

Ausdrücke in Umwandlungen. 39

Der Ausdrucks-Editor. 41

Portnamen in einem Ausdruck. 41

Hinzufügen von einem Ausdruck zu einem Port. 41

Kommentare in einem Ausdruck. 42

Ausdrucksvalidierung. 42

Testausdrücke. 42

Datentypumwandlung. 43

Lokale Variablen. 44

Temporäres Speichern von Daten und Vereinfachen von komplexen Ausdrücken. 44

Speichern von Werten in mehreren Zeilen. 45

Erfassen von Werten aus gespeicherten Prozeduren. 46

Richtlinien zum Konfigurieren von variablen Ports. 46

Variableninitialisierung. 47
Standardwerte für Ports. 47
Benutzerdefinierte Standardwerte. 48
Benutzerdefinierte Standardeingabewerte. 49
Standardwert der Validierung. 51
Benutzerdefinierte Standardausgabewerte 51
Allgemeine Regeln für Standardwerte. 53
Standardwert der Validierung. 54
Tracingebenen. 54
Wiederverwendbare Umwandlungen. 55
Wiederverwendbare Umwandlungsinstanzen und geerbte Änderungen. 55
Bearbeiten einer wiederverwendbaren Umwandlung. 56
Editoransichten für eine wiederverwendbare Umwandlung. 56
Nicht wiederverwendbare Umwandlungen. 56
Editor-Ansichten für eine nicht wiederverwendbare Umwandlung. 57
Erstellen einer Umwandlung. 57

Kapitel 2: Adressvalidator-Umwandlung. 58

Übersicht über Adressvalidator-Umwandlung. 58
Adressreferenzdaten. 59
Adressvalidierermodi und Adressreferenzdaten. 59
Portgruppen und Portauswahl. 61
Vorlagen. 62
Adressvalidator-Umwandlung - Eingabeportgruppen. 62
Adressvalidator-Umwandlung - Ausgabeportgruppen. 63
Ports mit mehreren Instanzen. 65
Adressvalidierungsprojekte. 67
Formatierte Adressen und Zustellungsstandards. 68
Komplettieren unvollständiger Adressen 69
Adressvalidator-Statusports. 69
Elementstatuscode-Definitionen. 70
Ausgabeportwerte für Adressauflösungscode. 72
Ausgabeportwerte für Elementeingabe-Status. 73
Elementrelevanz-Ausgabeportwerte. 74
Ausgabeportwerte für Elementergebnisstatus. 74
Ausgabeportwerte für den erweiterten Elementergebnisstatus. 76
Ausgabeportwerte für Zustellbarkeits-Score. 77
Ausgabeportwerte für Match-Code. 78
Ausgabeportwerte für Geokodierungsstatus. 80
Adressreferenzdaten-Dateistatus. 81
Allgemeine Einstellungen der Adressvalidator-Umwandlung. 82
Adressvalidator-Umwandlung - Erweiterte Eigenschaften. 83
Ortsalias. 83

Alternativer Straßenname. . . . 83
Schreibweise. . . . 84
Herkunftsland. . . . 85
Ländertyp. . . . 85
Standardland. . . . 86
Priorität bei doppelter Adresse. . . . 86
Elementabkürzung. . . . 86
Ausführungsinstanzen. . . . 87
Flexible Bereichserweiterung. . . . 87
Datentyp „Geocode“. . . . 88
Globale max. Feldlänge. . . . 89
Global bevorzugter Deskriptor. . . . 89
Eingabeformattyp. . . . 90
Eingabeformat mit Land . . . 90
Zeilentrennzeichen. . . . 90
Matching-Alternativen. . . . 91
Erweitertes Archiv abgleichen. . . . 91
Matching-Bereich. . . . 92
Maximale Ergebniszahl. . . . 92
Modus. . . . 93
Optimierungslevel. . . . 93
Ausgabeformattyp. . . . 94
Ausgabeformat mit Land. . . . 94
Bevorzugte Sprache. . . . 95
Bevorzugtes Skript. . . . 96
Zu erweiternde Bereiche. . . . 97
Ungültige Adressen standardisieren. . . . 98
Tracingebene. . . . 99
Zertifizierungsberichte. . . . 99
Felder im AMAS-Bericht. . . . 99
Felder im CASS-Bericht. . . . 100
SendRight-Bericht. . . . 101
Felder im SERP-Bericht. . . . 101
Konfigurieren einer Adressvalidator-Umwandlung. . . . 102
Hinzufügen von Ports zur Adressvalidator-Umwandlung. . . . 103
Erstellen von benutzerdefinierten Vorlagen. . . . 103
Definieren von Adressvalidator-Modellen. . . . 104
Definieren eines Zertifizierungsberichts. . . . 104

Kapitel 3: Aggregator-Umwandlung. . . . 105

Aggregator-Umwandlung - Übersicht. . . . 105
Aggregator-Umwandlungen in dynamischen Mappings. . . . 106
Entwickeln einer Aggregator-Umwandlung. . . . 106

Aggregator-Umwandlung - Ports. 106
Aggregat-Ausdrücke. 107
Aggregatfunktionen. 108
Eingebettete Aggregatfunktionen. 108
Bedingungsklauseln in Aggregat-Ausdrücken. 109
Nach Ports gruppieren. 109
Ports „Gruppieren nach” konfigurieren. 110
Parameter „Gruppieren nach”. 111
Standardwerte von "Nach Ports gruppieren" 111
Nicht-Aggregat-Ausdrücke. 111
Aggregator-Caches. 112
Sortierte Eingabe für eine Aggregator-Umwandlung. 112
Bedingungen für sortierte Eingabe. 113
Sortieren von Daten in einer Aggregator-Umwandlung. 113
Aggregatorumwandlung – Erweiterte Eigenschaften. 114
Erstellen einer wiederverwendbaren Aggregator-Umwandlung. 114
Erstellen einer Nicht-wiederverwendbaren Aggregator-Umwandlung. 115
Tipps für Aggregator-Umwandlungen. 115
Fehlerbehebung bei Aggregator-Umwandlungen. 116

Kapitel 4: Assoziations-Umwandlung. 117

Zuordnungsumwandlung - Übersicht. 117
Speicherzuordnung. 118
Zuordnungsumwandlung – Erweiterte Eigenschaften. 119

Kapitel 5: Ausnahmeumwandlung bei fehlerhaftem Datensatz. 120

Übersicht über die Ausnahmeumwandlung bei fehlerhaftem Datensatz. 120
Ausnahme bei fehlerhaftem Datensatz - Ausgabe Datensatztypen. 121
Prozessablauf für die Ausnahmeverwaltung bei fehlerhaften Datensätzen. 122
Ausnahme bei fehlerhaftem Datensatz - Mappings. 123
Ausnahme bei fehlerhaftem Datensatz - Qualitätsprobleme. 124
Human-Aufgaben. 124
Ausnahme bei fehlerhaftem Datensatz - Ports 125
Eingabeports der Ausnahmeumwandlung bei fehlerhaftem Datensatz. 125
Ausnahme-Umwandlungsausgabe bei fehlerhaftem Datensatz. 126
Ausnahme bei fehlerhaftem Datensatz - Konfigurationsansicht. 126
Erstellen der Tabelle mit fehlerhaften Datensätzen und der Problemtabelle. 128
Ausnahme bei fehlerhaftem Datensatz - Problemzuordnung 128
Ports den Qualitätsproblemen zuweisen. 129
Ausnahmeumwandlung – Erweiterte Eigenschaften. 129
Konfigurieren einer Ausnahmeumwandlung bei fehlerhaftem Datensatz. 130
Ausnahme bei fehlerhaftem Datensatz - Mapping-Beispiel. 131
Ausnahme bei fehlerhaftem Datensatz - Mapplet. 131

Ausnahme bei fehlerhaftem Datensatz - Beispiel für Eingabegruppen. 132
Beispielkonfiguration für Ausnahme bei fehlerhaftem Datensatz 133
Beispiel für eine Ausnahme bei fehlerhaftem Datensatz - Mapping-Ausgabe. 134

Kapitel 6: Fallumwandlerumwandlung. 136

Fallumwandlerumwandlung - Übersicht. 136
Fallstrategie-Eigenschaften. 136
Konfigurieren einer Fallumwandlersrategie. 137
Groß-/Kleinschreibungsumwandlung – Erweiterte Eigenschaften. 138

Kapitel 7: Klassifizierumwandlung. 139

Klassifizierumwandlung - Übersicht. 139
Klassifiziermodelle. 140
Klassifizieralgorithmen. 140
Klassifizierumwandlungsoptionen. 140
Klassifizierstrategien. 141
Klassifiziererumwandlung – Erweiterte Eigenschaften. 141
Konfigurieren einer Klassifizierstrategie. 142
Beispiel für Klassifizieranalyse. 142
Erstellen eines Klassifizier-Mappings. 143
Beispiel Eingabedaten. 144
Konfiguration der Datenquelle. 144
Konfiguration der Klassifizierumwandlung 144
Konfiguration der Routerumwandlung. 144
Konfiguration des Datenziels. 146
Ergebnis des Klassifizier-Mappings. 146

Kapitel 8: Vergleichsumwandlung. 147

Vergleichsumwandlung - Übersicht. 147
Feld-Matching-Strategien. 147
Bigramm. 148
Hamming-Abstand. 148
Abstand bearbeiten. 149
Jaro-Abstand. 149
Hamming-Abstand umkehren. 150
Identitäts-Matching-Strategien. 150
Konfigurieren einer Vergleichsstrategie. 151
Vergleichsumwandlung – Erweiterte Eigenschaften. 151

Kapitel 9: Konsolidierungsumwandlung. 152

Konsolidierungsumwandlung - Übersicht. 152
Konsolidierungs-Mappings. 153
Konsolidierungsumwandlung - Ports. 153

Konsolidierungsumwandlung - Ansichten. 154
Konsolidierungsumwandlung - Strategieansichten. 154
Konsolidierungsumwandlung – Erweiterte Eigenschaften. 154
Cache-Dateigröße. 155
Einfache Strategien. 156
Zeilenbasierte Strategien. 157
Erweiterte Strategien. 158
Einfache Konsolidierungsfunktionen. 158
CONSOL_AVG 159
CONSOL_LONGEST. 159
CONSOL_MAX. 160
CONSOL_MIN. 160
CONSOL_MOSTFREQ. 161
CONSOL_MOSTFREQ_NB. 162
CONSOL_SHORTEST. 162
Zeilenbasierte Konsolidierungsfunktionen. 163
CONSOL_GETROWFIELD. 163
CONSOL_MODALEXACT. 164
CONSOL_MOSTDATA. 165
CONSOL_MOSTFILLED. 165
Konsolidierungs-Mapping - Beispiel. 167
Eingabedaten. 167
Schlüsselgeneratorumwandlung. 167
Konsolidierungsumwandlung. 168
Konsolidierungs-Mapping-Ausgabe. 168
Konfigurieren einer Konsolidierungsumwandlung. 168

Kapitel 10: Datenmaskierungsumwandlung. 169

Datenmaskierungsumwandlung - Übersicht. 169
Maskierungstechniken. 170
Zufällige Maskierung. 170
Ausdrucksmaskierung. 172
Schlüsselmaskierung. 174
Substitutions-Maskierung. 175
Abhängige Maskierung. 179
Tokenisierungsmaskierung. 181
Maskierungsregeln. 181
Maskierungsformat. 182
Quellstringzeichen. 183
Ersatzzeichen für Ergebniszeichenfolge. 184
Bereich. 184
Unschärfe. 185
Spezielle Maskierungsformate. 186

Datenmaskierung der Kreditkartennummer. . . . 186
Datenmaskierung der E-Mail-Adresse. . . . 187
Erweiterte E-Mail-Maskierung. . . . 187
Maskierung der IP-Adresse . . . 188
Maskieren der Telefonnummer. . . . 188
Maskierung der Sozialversicherungsnummer . . . 188
Maskieren der URL-Adresse . . . 189
Maskierung der Sozialversicherungsnummer. . . . 189
Standardwertdatei. . . . 190
Konfiguration der Datenmaskierungsumwandlung. . . . 191
Konfigurieren des Datenintegrationsdiensts. . . . 191
Erstellen einer Datenmaskierungsumwandlung. . . . 191
Definieren der Ports. . . . 192
Konfigurieren der Datenmaskierung für jeden Port. . . . 192
Vorschau der maskierten Daten. . . . 192
Laufzeiteigenschaften der Datenmaskierungsumwandlung. . . . 193
Datenmaskierung - Beispiel. . . . 194
Read_Customer Data. . . . 194
Kundendaten-Maskierungsumwandlung. . . . 195
Ergebnisse der Kundentestdaten. . . . 195
Datenmaskierungsumwandlung – Erweiterte Eigenschaften. . . . 196

Kapitel 11: Datenprozessor-Umwandlung. . . . 197

Datenprozessorumwandlung - Überblick. . . . 197
Datenprozessorumwandlung - Ansichten. . . . 198
Datenprozessorumwandlung - Ports. . . . 199
Datenprozessorumwandlung - Eingabeports. . . . 199
Datenprozessorumwandlung - Ausgabeports. . . . 200
Pass-Through-Ports. . . . 201
Startkomponente. . . . 201
Verweise. . . . 202
Datenprozessorumwandlung – Einstellungen. . . . 202
Zeichencodierung. . . . 203
Regeln und Richtlinien für die Zeichencodierung. . . . 205
Ausgabeeinstellungen. . . . 205
Verarbeitungseinstellungen. . . . 206
XMap-Einstellungen. . . . 207
Konfigurieren der XML-Ausgabe. . . . 207
Ereignisse. . . . 209
Ereignistypen. . . . 209
Ansicht der Datenprozessor-Ereignisse. . . . 209
Protokolle. . . . 210
Entwicklungszeit-Ereignisprotokoll. . . . 210

Laufzeit-Ereignisprotokoll. 211
Anzeigen eines Ereignisprotokolls in der Ansicht für Datenprozessor-Ereignisse. 211
Benutzerprotokoll. 211
Entwicklung der Datenprozessorumwandlung. 212
Erstellen der Datenprozessorumwandlung. 212
Auswählen der Schemaobjekte 213
Erstellen von Objekten in einer leeren Datenprozessorumwandlung. 213
Erstellen von Ports. 216
Testen der Umwandlung. 216
Datenprozessorumwandlung - Importieren und Exportieren. 217
Exportieren der Datenprozessorumwandlung als Dienst. 217
Importieren eines Data Transformation-Diensts 217
Exportieren eines Mappings mit einer Datenprozessorumwandlung nach PowerCenter. 218

Kapitel 12: Entscheidungsumwandlung. 220

Entscheidungsumwandlung - Übersicht. 220
Entscheidungsumwandlung - Funktionen. 221
Entscheidungsumwandlung - Bedingte Anweisungen 223
Entscheidungsumwandlung - Operatoren. 224
Verarbeitung von NULL-Werten in Entscheidungsumwandlungen. 225
Konfigurieren einer Entscheidungsstrategie 225
Entscheidungsumwandlung – Erweiterte Eigenschaften. 226

Kapitel 13: Ausnahmeumwandlung bei dupliziertem Datensatz. 227

Übersicht über die Ausnahmeumwandlung bei dupliziertem Datensatz. 227
Prozessablauf der Ausnahmen bei duplizierten Datensätzen 228
Ausnahmen bei duplizierten Datensätzen. 228
Ansicht der Konfiguration der Ausnahme bei dupliziertem Datensatz 229
Erstellen einer duplizierten Datensatztabelle. 230
Ports. 231
Eingabeports der Ausnahmeumwandlung bei dupliziertem Datensatz. 231
Ausgabeports der Ausnahmeumwandlung bei dupliziertem Datensatz. 232
Erstellen von Ports 233
Ausnahmeumwandlung bei dupliziertem Datensatz - Erweiterte Eigenschaften. 233
Ausnahme bei mehrfachem Datensatz - Mapping-Beispiel. 234
Mapping der Ausnahmen bei duplizierten Datensätzen. 234
Match-Umwandlung. 235
Eingabegruppen der Ausnahme bei dupliziertem Datensatz. 235
Beispiel für eine Ausnahme bei dupliziertem Datensatz - Konfigurationsansicht. 236
Datensätze der Standardausgabentabelle 237
Cluster-Ausgabe. 238
Erstellen einer Ausnahmeumwandlung bei dupliziertem Datensatz. 239

Kapitel 14: Ausdrucksumwandlung. 241

Ausdrucksumwandlung - Übersicht. 241
Ausdrucksumwandlungsports. 242
Testausdrücke. 243
Datumsformatzeichenfolgen für Beispieldaten. 244
Testen eines Ausdrucks. 244
Portauswahlen. 244
Portauswahl-Konfiguration. 245
Auswahlregeln. 246
Erstellen einer Portauswahl. 246
Dynamische Ausdrücke. 248
Einstellungen für Ausgabeport. 249
Erstellen eines dynamischen Ausdrucks 250
Ausdrucksumwandlung - Erweiterte Eigenschaften. 252

Kapitel 15: Filterumwandlung. 254

Filterumwandlung - Übersicht. 254
Filterumwandlungen in dynamischen Mappings. 255
Filterbedingung. 255
Die Filterbedingung parametrisieren. 256
Filtern von Zeilen mit Nullwerten. 257
Filterumwandlung - Erweiterte Eigenschaften. 258
Filterumwandlung - Tipps für die Leistung. 258

Kapitel 16: Umwandlung von hierarchisch in relational. 259

Umwandlung von hierarchisch in relational – Übersicht. 259
Beispiel - Hierarchische-in-relational-Umwandlung. 259
Relationale Ausgabeports und die Ansicht „Übersicht“. 261
Umwandlungsports von hierarchisch in relational. 262
Schemaverweise. 263
Port-Konfiguration. 263
Entwicklung der Umwandlung von hierarchisch in relational. 264
Erstellen der Umwandlung von hierarchisch in relational 264
Konfigurieren der Ports und Mappings. 264
Testen der Umwandlung. 265

Kapitel 17: Java-Umwandlung. 266

Java-Umwandlung – Übersicht. 266
Wiederverwendbare und nicht wiederverwendbare Java-Umwandlungen. 267
Aktive und passive Java-Umwandlungen. 267
Datentypkonvertierung. 267
Entwickeln einer Java-Umwandlung. 268

Java-Umwandlungsports. 269
Erstellen von Ports. 269
Festlegen von Standardportwerten. 269
Java-Umwandlung – Erweiterte Eigenschaften. 270
Konfigurieren des Klassenpfads für den Developer Tool-Client. 272
Konfigurieren des Klassenpfads für den Data Integration Service. 273
Entwickeln von Java-Code. 273
Erstellen von Java-Codeausschnitten. 275
Importieren von Java-Paketen. 276
Definieren von Helfer-Code. 277
Java-Umwandlung – Java-Eigenschaften. 277
Registerkarte "Importe". 278
Registerkarte "Helfer". 278
Registerkarte "Bei Eingabe". 278
Registerkarte "Am Datenende". 279
Registerkarte "Funktionen". 280
Registerkarte "Full-Code". 280
Filteroptimierung mit der Java-Umwandlung. 280
Frühe Auswahloptimierung mit der Java-Umwandlung. 280
Push-Into-Optimierung mit der Java-Umwandlung. 282
Erstellen einer Java-Umwandlung. 283
Erstellen einer wiederverwendbaren Java-Umwandlung. 283
Erstellen einer nicht wiederverwendbaren Java-Umwandlung. 284
Kompilieren einer Java-Umwandlung. 284
Fehlerbehebung bei einer Java-Umwandlung. 285
Suchen der Quelle bei Kompilierungsfehlern. 285
Erkennen der Ursache von Kompilierungsfehlern. 286

Kapitel 18: Java-Umwandlungs-API-Referenz. 287

Java-Umwandlungs-API-Methoden - Übersicht. 287
defineJExpression. 288
failSession. 289
generateRow. 289
getInRowType. 290
getMetadata. 291
incrementErrorCount. 291
invokeJExpression. 292
isNull. 293
logError. 294
logInfo. 294
resetNotification. 295
setNull. 295
storeMetadata. 296

Kapitel 19: Java-Ausdrücke. 298

Java-Ausdrücke - Übersicht. 298

Ausdrucksfunktionstypen. 299

Verwenden des Dialogfelds Funktion definieren zum Definieren eines Ausdrucks. 299

Schritt 1. Konfigurieren der Funktion. 300

Schritt 2. Erstellen und Validieren des Ausdrucks. 300

Schritt 3. Generieren von Java-Code für den Ausdruck. 300

Erstellen eines Ausdrucks und Generieren von Java-Code mit dem Dialogfeld Funktion definieren. 300

Vorlagen für Java-Ausdrücke. 301

Arbeiten mit der einfachen Schnittstelle. 301

invokeJExpression. 302

Einfache Schnittstelle - Beispiel. 302

Arbeiten mit der erweiterten Schnittstelle. 303

Aufrufen eines Ausdrucks mit der erweiterten Schnittstelle. 303

Regeln und Richtlinien für das Arbeiten mit der erweiterten Schnittstelle. 304

EDataType-Klasse. 304

JExprParamMetadata-Klasse. 305

defineJExpression. 305

JExpression-Klasse. 306

Erweiterte Schnittstelle - Beispiel. 306

API-Referenz für JExpression-Klasse. 307

getBytes. 308

getDouble. 308

getInt. 308

getLong. 308

getResultDataType. 308

getResultMetadata. 309

getStringBuffer. 309

invoke. 309

isResultNull. 310

Kapitel 20: Joiner-Umwandlung. 311

Joiner-Umwandlung - Übersicht. 311

Joiner-Umwandlung – Erweiterte Eigenschaften. 312

Joiner-Caches. 313

Joiner-Umwandlungsports. 314

Joiner-Umwandlungen in dynamischen Mappings. 315

Portauswahlen in einer Joiner-Umwandlung. 315

Auswahlregeln 316

Erstellen einer Portauswahl. 317

Definieren einer Join-Bedingung. 318

Einfacher Bedingungstyp. . . . 319
Erweiterter Bedingungstyp. . . . 320
Portauswahlen in Join-Bedingungen. . . . 320
Dynamische Ports in Join-Bedingungen. . . . 321
Ausdrucksparameter. . . . 321
Join-Typen. . . . 322
Normales Join. . . . 322
Master Outer Join. . . . 323
Detail-Outer-Join. . . . 324
Full Outer Join. . . . 324
Sortierte Eingabe für eine Joiner-Umwandlung. . . . 325
Konfigurieren der Sortierreihenfolge. . . . 325
Hinzufügen von Umwandlungen zum Mapping. . . . 325
Regeln und Richtlinien für Join-Bedingungen. . . . 326
Beispiel für eine Join-Bedingung und Sortierreihenfolge. . . . 327
Zusammenfügen von Daten aus derselben Quelle. . . . 328
Zusammenfügen zweier Zweige derselben Pipeline. . . . 328
Zusammenfügen zweier Instanzen derselben Quelle. . . . 329
Richtlinien für das Zusammenfügen von Daten aus derselben Quelle. . . . 330
Blockieren der Quellen-Pipelines. . . . 330
Unsortierte Joiner-Umwandlung. . . . 331
Sortierte Joiner-Umwandlung. . . . 331
Leistungstipps für die Joiner-Umwandlung. . . . 331
Regeln und Richtlinien für eine Joiner-Umwandlung. . . . 332

Kapitel 21: Schlüsselgeneratorumwandlung. . . . 333

Schlüsselgeneratorumwandlung - Übersicht. . . . 333
Soundex-Strategie. . . . 334
Soundex-Strategieeigenschaften. . . . 334
String-Strategie. . . . 334
Zeichenfolgen-Strategieeigenschaften. . . . 335
NYSIIS-Strategie. . . . 335
Schlüsselgenerator-Ausgabeports. . . . 335
Konfigurieren einer Gruppierungsstrategie. . . . 336
Schlüsselerstellungseigenschaften. . . . 336
Erweiterte Eigenschaften der Schlüsselgeneratorumwandlung. . . . 337

Kapitel 22: Labeler-Umwandlung. . . . 338

Labeler-Umwandlung - Übersicht. . . . 338
Einsatz der Labeler-Umwandlung. . . . 339
Referenzdatenverwendung in der Labeler-Umwandlung. . . . 340
Zeichensätze. . . . 341
Probabilistische Modelle. . . . 341

Referenztabellen. . . . 341
Reguläre Ausdrücke. . . . 341
Tokensätze. . . . 342
Labeler-Umwandlung - Strategien. . . . 342
Zeichenkennzeichnungsoperationen. . . . 342
Tokenbeschriftungs-Operationen. . . . 343
Beschriftungsumwandlung - Ports. . . . 343
Zeichenbeschriftungs-Eigenschaften. . . . 344
Allgemeine Eigenschaften. . . . 344
Referenztabelleneigenschaften. . . . 344
Zeichensatzeigenschaften. . . . 345
Filtereigenschaften. . . . 345
Tokenbeschriftungseigenschaften. . . . 346
Allgemeine Eigenschaften. . . . 346
Eigenschaften von Tokensätzen. . . . 347
Benutzerdefinierte Beschriftungseigenschaften. . . . 348
Eigenschaften für das probabilistische Matching. . . . 348
Referenztabelleneigenschaften. . . . 348
Konfigurieren einer Zeichenbeschriftungs-Strategie. . . . 349
Konfigurieren einer Tokenbeschriftungsstrategie. . . . 350
Labeler-Umwandlung – Erweiterte Eigenschaften. . . . 350

Kapitel 23: Lookup-Umwandlung. . . . 351

Lookup-Umwandlung - Übersicht. . . . 351
Verbunden und nicht verbundene Lookups. . . . 352
Verbundene Lookups. . . . 354
Nicht verbundene Lookups. . . . 354
Entwickeln einer Lookup-Umwandlung. . . . 355
Lookup-Abfrage. . . . 355
Standard-Lookup-Abfrage. . . . 355
SQL-Überschreibung für eine Lookup-Abfrage. . . . 356
Parameter in einer SQL-Überschreibungsabfrage. . . . 356
Reservierte Wörter. . . . 357
Richtlinien für das Überschreiben der Lookup-Abfrage . . . 357
Überschreiben der Lookup-Abfrage. . . . 358
Lookup-Quellfilter. . . . 358
Filtern von Quellzeilen in einem Lookup. . . . 359
Lookup-Bedingung. . . . 359
Die Lookup-Bedingung konfigurieren. . . . 360
Regeln und Richtlinien für Lookup-Umwandlungsbedingungen. . . . 361
Lookup-Cache. . . . 362
Abfrage-Eigenschaften. . . . 362
Lookup-Umwandlungen in dynamischen Mappings. . . . 363

Dynamische Ports definieren. 364
Die Lookup-Quelle ändern. 364
Die Lookup-Quelle parametrisieren. 365
Lookup-Quellen mit Parametern 366
Parameter in einem doppelten Datenobjekt konfigurieren. 366
Portauswahlen. 368
Portauswahl-Konfiguration. 369
Auswahlregeln. 370
Die Lookup-Bedingung parametrisieren. 371
Erstellen einer Portauswahl. 372
Laufzeiteigenschaften. 374
Erweiterte Eigenschaften. 375
Erstellen einer wiederverwendbaren Lookup-Umwandlung. 376
Erstellen einer nicht wiederverwendbare Lookup-Umwandlung. 377
Erstellen einer nicht verbundenen Lookup-Umwandlung. 378
Nicht verbundenes Lookup - Beispiel. 379

Kapitel 24: Lookup-Caches. 382

Lookup-Caches – Übersicht. 382
Lookup-Cache-Typen. 383
Nicht zwischengespeichertes Lookup. 384
Statischer Lookup-Cache. 384
Persistenter Lookup-Cache. 385
Neuerstellen eines persistenten Lookup-Caches. 385
Dynamischer Lookup-Cache. 386
Gemeinsam genutzter Lookup-Cache. 386
Regeln und Richtlinien für die gemeinsame Nutzung eines Lookup-Caches. 387
Cache-Vergleich. 388
Cache-Partitionierung für Lookups. 388

Kapitel 25: Dynamischer Lookup-Cache. 389

Dynamischer Lookup-Cache – Übersicht. 389
Verwendung für einen dynamischen Lookup-Cache. 390
Dynamische Lookup-Cache-Eigenschaften. 390
Dynamischer Lookup-Cache und Ausgabewerte. 392
Lookup-Umwandlungswerte. 392
Lookup-Umwandlungswerte – Beispiel. 393
SQL-Überschreibung und dynamischer Lookup-Cache. 395
Mapping-Konfiguration für einen dynamischen Lookup-Cache. 396
Einfügen, andernfalls aktualisieren. 396
Aktualisieren, andernfalls einfügen. 397
Dynamischer Lookup-Cache und Ziel-Synchronisierung. 398
Konditionale dynamische Lookup-Cache-Updates. 399

Konditionale dynamische Lookup-Cache-Verarbeitung. 399
Konfigurieren eines konditionalen dynamischen Lookup-Cache. 399
Dynamisches Cache-Update mit Ausdrucksergebnissen. 400
Null-Ausdruckswerte. 400
Ausdrucksverarbeitung. 400
Konfigurieren eines Ausdrucks für dynamische Cache-Updates. 401
Beispiel für einen dynamischen Lookup-Cache. 401
Regeln und Richtlinien für dynamische Lookup-Caches. 402

Kapitel 26: Match-Umwandlung. 404

Match-Umwandlung - Übersicht. 404
Match-Analyse. 405
Spaltenanalyse. 405
Analyse mit Einzelquelle und Analyse mit Doppelquelle. 406
Feldvergleichsanalysen und Identitätsvergleichsanalysen. 406
Gruppen in der Match-Analyse. 407
Match-Paare und Cluster. 408
Übereinstimmungswert-Berechnungen. 409
Gewichtete Scores. 409
Null-Übereinstimmungswerte. 410
Cluster-Ausgabeoptionen. 410
Treiber-Scores und Verknüpfungs-Scores in der Cluster-Analyse. 411
Analyse von Master-Daten. 413
Mapping erneut verwenden. 414
Identitätsvergleichsanalyse und persistente Indexdaten. 414
Regeln und Richtlinien für persistente Indexdaten. 414
Match-Mapping-Leistung. 415
Anzeigen von Match-Cluster-Analysedaten. 416
Anzeigen von Match-Performance-Analysedaten. 417
Match-Leistung in der Identitätsanalyse. 418
Erstellen eines Datenspeichers für Identitätsindexdaten. 418
Verwenden des Indexdatenspeichers in der Analyse mit Einzelquelle. 419
Match-Umwandlungsansichten. 420
Match-Umwandlungsports. 422
Match-Umwandlungs-Eingabeports. 422
Vergleichsumwandlungs-Ausgabeports. 422
Persistenzstatuscodes und Persistenzstatusbeschreibungen. 423
Ausgabeports und Match-Ausgabe-Auswahl. 426
Match-Mapplets. 426
Erstellen eines Match-Mapplets. 427
Mit einem Match-Mapplet. 427
Konfigurieren eines Match-Analyse-Vorgangs. 428

Kapitel 27: Match-Umwandlungen in der Feldanalyse. 429

Feld-Match-Analyse. 429
Prozessablauf für Feld-Match-Analysen. 429
Optionen für den Feld-Match-Typ. 430
Feld-Match-Strategien. 430
Feld-Match-Algorithmen. 431
Eigenschaften für Feld-Match-Strategien. 433
Optionen der Feld-Match-Ausgabe. 434
Match-Ausgabetypen. 434
Eigenschaften für Match-Ausgabe. 434
Erweiterte Feld-Match-Eigenschaften. 436
Beispiel für die Feld-Match-Analyse. 436
Erstellen des Mappings. 436
Eingabedaten - Beispiel. 437
Konfiguration der Schlüsselgeneratorumwandlung. 438
Konfiguration der Match-Umwandlung. 438
Ausführen des Daten-Viewers. 440
Ergebnis. 441

Kapitel 28: Match-Umwandlungen in der Identitätsanalyse. 442

Identitäts-Match-Analyse. 442
Prozessablauf für Identitätsvergleichs-Analysen. 443
Eigenschaften für Identitätsvergleichstyp. 443
Eigenschaften für Indexverzeichnis und Cache-Verzeichnis. 446
Parameter der Persistenz-Methode. 447
Identitäts-Match-Strategien. 447
Identitäts-Match-Algorithmen. 448
Eigenschaften für Identitäts-Match-Strategien. 449
Optionen für Identitäts-Match-Ausgabe. 450
Match-Ausgabetypen. 450
Eigenschaften für Match-Ausgabe. 451
Erweiterte Identitäts-Match-Eigenschaften. 453
Persistenter Index - Fallstudie. 454
Identitätsvergleichsanalyse - Beispiel. 456
Erstellen des Mappings. 457
Eingabedaten - Beispiel. 457
Konfiguration der Ausdrucksumwandlung. 458
Konfiguration der Match-Umwandlung. 458
Ausführen des Daten-Viewers. 462
Ergebnis. 462

Kapitel 29: Merge-Umwandlung 464

Merge-Umwandlung - Übersicht 464
Konfigurieren einer Merge-Strategie 464
Merge-Umwandlung – Erweiterte Eigenschaften 465

Kapitel 30: Normalizer-Umwandlung 466

Normalizer-Umwandlung - Übersicht 466
Mehrfach vorkommende Felder 467
    Generierte Spalten-ID 467
Mehrfach vorkommende Datensätze 468
Definition der Eingabehierarchie 469
    Eingabeports für die Normalizer-Umwandlung 470
    Felder zusammenführen 471
Ausgabegruppen und Ports der Normalizer-Umwandlung 472
    Erstellen einer Ausgabegruppe 473
    Aktualisieren einer Ausgabegruppe 475
Schlüsselgenerierung für Ausgabegruppen 476
Normalizer-Umwandlung - Erweiterte Eigenschaften 476
    Generieren von Ausgabegruppen der ersten Ebene 477
Erstellen einer Normalizer-Umwandlung 477
Erstellen einer Normalizer-Umwandlung aus einer vorgelagerten Quelle 478
Normalizer-Mapping - Beispiel 478
    Beispiel für Normalizer-Mapping 479
    Beispiel für die Normalizer-Definition 480
    Beispiel für Normalizer-Eingabe- und Ausgabegruppen 480
    Beispiel für die Normalizer-Mapping-Ausgabe 481

Kapitel 31: Parser-Umwandlung 482

Parser-Umwandlung - Übersicht 482
Modi der Parserumwandlung 483
Einsatz der Parserumwandlung 483
Referenzdatenverwendung in der Parserumwandlung 484
    Mustersätze 485
    Probabilistische Modelle 485
    Referenztabellen 486
    Reguläre Ausdrücke 486
    Tokensätze 486
Token-Parsing-Operationen 486
Token-Parsing-Ports 487
Token-Parsing-Eigenschaften 488
    Allgemeine Eigenschaften 488
    Eigenschaften des probabilistischen Modells 489

Referenztabelleneigenschaften. 489
Eigenschaften von Tokensätzen. 489
Musterbasierter Parsing-Modus. 490
Musterbasierte-Parsing-Ports. 491
Konfigurieren einer Token-Parsing-Strategie. 491
Konfigurieren einer Muster-Parsing-Strategie. 492
Erweiterte Eigenschaften für Parserumwandlung. 493

Kapitel 32: Rang-Umwandlung. 494

Rang-Umwandlung - Übersicht. 494
Ranking von Stringwerten. 495
Rang-Umwandlungseigenschaften. 495
Rang-Umwandlungen in dynamischen Mappings. 495
Rang-Umwandlungsports. 496
Rang-Index. 496
Rang-Port. 497
Ports „Gruppieren nach" definieren. 497
Parameter „Gruppieren nach". 498
Rang-Caches. 499
Rangumwandlung - Erweiterte Eigenschaften. 499

Kapitel 33: Leseumwandlung. 501

Leseumwandlung – Übersicht. 501
Eigenschaften von Leseumwandlungen 501
Allgemeine Eigenschaften. 503
Datenobjekteigenschaften. 503
Abfrageeigenschaften. 503
Laufzeiteigenschaften 504
Quelleneigenschaften 504
Erweiterte Eigenschaften. 505
Relationale Datenobjekte synchronisieren. 505
Das Quelldatenobjekt ändern. 506
Das Lesedatenobjekt parametrisieren. 507
Parameter von Leseumwandlungen. 508
Einschränkungen 508
Eine Leseumwandlung erstellen. 509
Erstellen einer Leseumwandlung im Zuordnungseditor. 509

Kapitel 34: Relational-in-hierarchisch-Umwandlung. 511

Übersicht über Relational-in-Hierarchisch-Umwandlungen. 511
Beispiel - Relational-in-hierarchisch-Umwandlung. 512
Relationale Eingabeports und die Ansicht „Übersicht". 514
Ports für Relational-in-hierarchisch-Umwandlungen. 515

Schemaverweise. 515
Entwicklung der Relational-in-hierarchisch-Umwandlung. 515
Erstellen der Relational-in-hierarchisch-Umwandlung. 515
Erstellen der Ports. 516

Kapitel 35: REST-Webdienst-Verbraucher-Umwandlung. 517

REST-Webdienst-Verbraucher-Umwandlung - Übersicht. 517
REST-Webdienst-Verbraucher-Umwandlung - Prozess. 519
REST-Webdienst-Verbraucher-Umwandlung - Konfiguration. 519
Meldungskonfiguration. 519
Ressourcenkennung. 520
HTTP-Methoden. 521
HTTP-Get-Methode. 521
HTTP-Post-Methode. 522
HTTP-Put-Methode. 522
HTTP-Delete-Methode. 523
REST-Webdienst-Verbraucher-Umwandlung - Ports. 524
Eingabeports. 524
Ausgabeports. 524
Pass-Through-Ports. 525
Argumentports. 525
URL-Ports. 525
HTTP-Header-Ports. 525
Cookie-Ports. 526
Ausgabe-XML-Ports. 526
Antwortcode-Ports. 526
REST-Webdienst-Verbraucher-Umwandlung - Eingabe-Mapping. 527
Regeln und Richtlinien zum Zuordnen von Eingabeports zu Elementen. 527
Zuordnen von Eingabeports zur Methodeneingabe. 528
REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping. 529
Regeln und Richtlinien zum Zuordnen von Elementen zu Ausgabeports. 530
Anpassen von Anzeigeoptionen. 530
Zuordnen der Methodenausgabe zu Ausgabeports. 531
REST-Webdienst-Verbraucher-Umwandlung - Erweiterte Eigenschaften. 531
REST-Webdienst-Verbraucher-Umwandlung - Erstellung. 532
Erstellen einer REST-Webdienst-Verbraucher-Umwandlung. 533

Kapitel 36: Router-Umwandlung. 534

Router-Umwandlung - Übersicht. 534
Router-Umwandlungen in dynamischen Mappings. 535
Arbeiten mit Gruppen. 536
Eingabegruppe. 536
Ausgabegruppen. 536

Verwenden von Gruppenfilterbedingungen. 537
Dynamische Ports in Gruppenfilterbedingungen. 539
Den Gruppenfilter parametrisieren. 539
Hinzufügen von Gruppen. 539
Arbeiten mit Ports. 540
Verbinden von Router-Umwandlungen in einem Mapping. 540
Router-Umwandlung - Erweiterte Eigenschaften. 541

Kapitel 37: Sequenzgeneratorumwandlung. 542

Sequenzgeneratorumwandlung – Übersicht. 542
Erstellen von Schlüsseln. 542
Ersetzen fehlender Werte. 543
Durchlaufen eines Wertebereichs. 544
Sequenzgeneratorports. 544
Pass-Through-Ports. 544
NEXTVAL-Port. 545
Sequenzgeneratorumwandlung-Eigenschaften. 545
Zurücksetzen. 546
Sequenzdatenobjekt. 546
Sequenzdatenobjekt-Eigenschaften. 547
Häufig gestellte Fragen (FAQ). 547

Kapitel 38: Sortierer-Umwandlung. 549

Sortierer-Umwandlung – Übersicht. 549
Sortierer-Umwandlungen in dynamischen Mappings. 550
Entwicklung einer Sortier-Umwandlung. 550
Sortierer-Umwandlungsports. 551
Registerkarte „Sortieren“. 551
Sortierschlüssel konfigurieren. 551
Die Sortierschlüssel parametrisieren. 552
Sortierer-Umwandlung – Erweiterte Eigenschaften. 554
Sortierer-Cache. 554
Erstellen einer Sortier-Umwandlung. 555
Erstellen einer wiederverwendbaren Sortierer-Umwandlung. 555
Erstellen einer nicht wiederverwendbaren Sortierer-Umwandlung. 556
Sortierer-Umwandlung – Beispiel. 556

Kapitel 39: SQL-Umwandlung. 558

SQL-Umwandlung – Übersicht. 558
SQL-Umwandlungsports. 559
Eingabeports. 560
Ausgabeports. 560
Pass-Through-Ports 561

SQLError-Port. 562
Anzahl der betroffenen Zeilen. 563
SQL-Umwandlung – Erweiterte Eigenschaften. 563
SQL-Umwandlungsabfrage. 565
Definieren der SQL-Abfrage. 566
Kardinalität zwischen Eingabezeile und Ausgabezeile. 567
Verarbeitung von Abfrageanweisungen. 568
Port-Konfiguration. 569
Maximale Ausgabe des Zeilenzählers. 569
Fehlerzeilen. 569
Fortfahren bei SQL-Fehler. 571
Filteroptimierung mit der SQL-Umwandlung. 571
"Early Selection"-Optimierung mit der SQL-Umwandlung. 572
Push-Into-Optimierung mit der SQL-Umwandlung. 572
SQL-Umwandlung - Beispiel mit einer SQL-Abfrage. 573
Mapping logischer Datenobjekte. 573
Gehaltstabelle. 574
Mitarbeitertabelle 574
SQL-Umwandlung. 574
Ausgabe. 576
Gespeicherte Prozeduren. 577
SQL-Umwandlungsports für gespeicherte Prozeduren. 577
Ergebnissätze aus gespeicherten Prozeduren. 579
Gespeicherte Prozedur - Beispiel. 581
SQL-Umwandlungsverbindung. 582
Erstellen eines Parameters für einen Verbindungsnamen. 582
Manuelles Erstellen einer SQL-Umwandlung. 583
Erstellen einer SQL-Umwandlung aus einer gespeicherten Prozedur. 584

Kapitel 40: Standardisierungs-Umwandlung. 586

Standardisierer-Umwandlung - Übersicht. 586
Standardisierungsstrategien. 586
Standardisierungseigenschaften. 587
Konfigurieren einer Standardisierungsstrategie. 588
Standardisierer-Umwandlung – Erweiterte Eigenschaften. 588

Kapitel 41: Union-Umwandlung. 590

Union-Umwandlung - Übersicht. 590
Gruppen und Ports. 591
Union-Umwandlung - Erweiterte Eigenschaften. 591
Union-Umwandlung-Verarbeitung. 592
Erstellen einer Union-Umwandlung. 592
Erstellen einer wiederverwendbaren Union-Umwandlung. 592

Erstellen einer nicht wiederverwendbaren Union-Umwandlung. 592

Kapitel 42: Update-Strategie-Umwandlung. 594

Update-Strategie-Umwandlung - Übersicht. 594
Einstellen der Update-Strategie. 594
Aktualisierungsstrategie-Umwandlungen in dynamischen Mappings. 595
Kennzeichnen von Zeilen in einem Mapping. 595
Aktualisierungsstrategieausdrücke. 595
Update-Strategie-Umwandlung - Erweiterte Eigenschaften. 596
Aggregator- und Update-Strategie-Umwandlungen. 596
Festlegen von Aktualisierungsoptionen für einzelne Targets. 597

Kapitel 43: Web-Dienst-Verbraucher-Umwandlung. 598

Web-Dienst-Verbraucher-Umwandlung - Übersicht. 598
SOAP-Meldungen. 599
WSDL-Dateien. 599
Operationen. 600
Web-Dienst-Sicherheit. 600
WSDL-Auswahl. 601
Web-Dienst-Verbraucher-Umwandlung - Ports. 602
Eingabeports für HTTP-Kopfzeilen. 603
Andere Eingabeports. 603
Web-Dienst-Verbraucher-Umwandlung - Eingabemapping. 604
Regeln und Richtlinien zum Zuordnen von Eingabeports zu Knoten. 605
Anpassen von Anzeigeoptionen. 605
Zuordnen von Eingabeports zur Vorgangseingabe. 606
REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping. 607
Regeln und Richtlinien zum Zuordnen von Knoten zu Ausgabeports. 608
Mapping der SOAP-Meldung als XML 609
Anpassen von Anzeigeoptionen. 609
Zuordnen der Vorgangsausgabe zu Ausgabeports. 610
Erweiterte Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung. 611
Web-Dienst-Fehlerbehandlung. 613
Meldungskomprimierung 614
Parallelverarbeitung. 614
Filteroptimierungen. 615
Aktivieren der "Early Selection"-Optimierung mit der Webdienst-Verbraucher-Umwandlung. . . 615
Push-Into-Optimierung mit der Web-Dienst-Verbraucher-Umwandlung. 615
Erstellen einer Web-Dienst-Verbraucher-Umwandlung. 617
Web-Dienst-Verbraucher-Umwandlung - Beispiel. 619
Eingabedatei. 619
Logisches Datenobjektmodell. 619
Mapping logischer Datenobjekte. 620

Web-Dienst-Verbraucher-Umwandlung. 620

Kapitel 44: Parsen von Webdienst-SOAP-Meldungen. 623

Parsen von Webdienst-SOAP-Meldungen – Übersicht. 623
Benutzeroberfläche für Umwandlungen. 624
Konfigurieren von Mehrfachausgaben. 625
Normalisierte relationale Ausgabe. 625
Generierte Schlüssel. 625
Denormalisierte relationale Ausgabe. 626
Pivotierte relationale Ausgabe. 627
Parsen von anyType-Elementen. 627
Parsen von abgeleiteten Typen. 628
Parsen von QName-Elementen. 629
Parsen von Substitutionsgruppen. 630
Parsen von XML-Konstrukten in SOAP-Meldungen. 630
Auswahlelement. 630
Listenelement. 630
Union-Element. 631

Kapitel 45: Generieren von Webdienst-SOAP-Meldungen. 632

Generieren von Webdienst-SOAP-Meldungen – Übersicht. 632
Benutzeroberfläche für Umwandlungen. 633
Eingabeports, Bereich 633
Operationsbereich. 634
Beziehungen zwischen Port und Hierarchieebene 635
Schlüssel. 636
Zuordnen von Ports. 637
Zuordnen eines Ports 638
Zuordnen einer Gruppe. 638
Zuordnen mehrerer Ports. 639
Pivotieren mehrfach vorkommender Ports 639
Zuordnen denormalisierter Daten. 640
Abgeleitete Typen und Elementsubstitution. 642
Generieren abgeleiteter Typen. 642
Generieren von anyType-Elementen und -Attributen. 642
Generieren von Substitutionsgruppen. 643
Generieren von XML-Konstrukten in SOAP- Meldungen. 643
Auswahlelement. 643
Listenelement. 644
Union-Element. 645

Kapitel 46: Umwandlung für gewichteten Durchschnitt. 646

Umwandlung für gewichteten Durchschnitt - Übersicht. 646

Konfigurieren einer gewichteten Durchschnittsumwandlung. 647
Beispiel für gewichtete Übereinstimmungswerte. 647
Erweiterte Eigenschaften für gewichteten Durchschnitt. 648

Kapitel 47: Schreibumwandlung. 649

Schreibumwandlung – Übersicht. 649
Eigenschaften von Schreibumwandlungen 649
Allgemeine Eigenschaften. 650
Datenobjekteigenschaften. 651
Porteigenschaften. 651
Laufzeiteigenschaften. 652
Eigenschaften von Laufzeitverknüpfungen. 652
Erweiterte Eigenschaften. 653
Eine Schreibumwandlung erstellen. 655
Erstellen einer Schreibumwandlung aus einem Datenobjekt. 655
Erstellen einer Schreibumwandlung aus einem Zuordnungsfluss. 656
Erstellen einer Schreibumwandlung aus einem Parameter. 657
Erstellen einer Schreibumwandlung aus einer vorhandenen Umwandlung 658

Kapitel 48: Umwandlungs-Caches. 660

Umwandlungs-Caches – Übersicht. 660
Cache-Typen. 661
Cachedateien. 661
Cache-Dateiverzeichnis. 662
Cachegröße. 662
Automatische Cache-Größe. 662
Benutzerdefinierte Cachegröße. 664
Erhöhen der Cachegröße durch den Datenintegrationsdienst. 664
Cachegröße für partitionierte Caches. 665
Optimierung der Cachegröße. 665
Schritt 1. Festlegen der Tracingebene auf die ausführliche Initialisierung. 665
Schritt 2. Ausführen des Mappings im automatischen Cache-Modus. 666
Schritt 3. Analysieren der Caching-Leistung. 666
Schritt 4. Konfigurieren von benutzerdefinierten Cachegrößen. 666

Anhang A: Umwandlungs-Delimiter. 668

Umwandlungs-Delimiter - Übersicht. 668

Index. 669

Einleitung

Das *Informatica Developer-Umwandlungshandbuch* enthält Informationen zu gemeinsamen Funktionen im Developer-Tool. Es wurde für Entwickler von Datendiensten und Datenqualität verfasst. In diesem Handbuch wird davon ausgegangen, dass Ihnen die Datenqualitäskonzepte, Konzepte von Einfachdateien und relationalen Datenbanken sowie die Datenbank-Engines in Ihrer Umgebung bekannt sind.

Informatica-Ressourcen

Informatica-Netzwerk

Im Informatica-Netzwerk finden Sie den globalen Kundensupport von Informatica, die Informatica-Wissensdatenbank und andere Produktressourcen. Für den Zugriff auf das Informatica-Netzwerk besuchen Sie https://network.informatica.com.

Als Mitglied können Sie:

- zentral auf alle Ihre Informatica-Ressourcen zugreifen.
- Durchsuchen Sie die Wissensdatenbank nach Produktressourcen, einschließlich Dokumentation, häufig gestellter Fragen und bewährter Methoden.
- Zeigen Sie Informationen zur Produktverfügbarkeit an.
- Ihre Support-Fälle prüfen.
- Ihr lokales Informatica-Netzwerk für Benutzergruppen suchen und mit anderen Benutzern zusammenarbeiten.

Informatica-Wissensdatenbank

Verwenden Sie die Informatica-Wissensdatenbank, um das Informatica-Netzwerk nach Produktressourcen, wie z. B. Dokumentation, Ratgeberartikeln, bewährten Methoden und PAMs, zu durchsuchen.

Für den Zugriff auf die Wissensdatenbank besuchen Sie https://kb.informatica.com. Wenn Sie Fragen, Kommentare oder Ideen zur Wissensdatenbank haben, wenden Sie sich per E-Mail an das Team der Informatica-Wissensdatenbank unter KB_Feedback@informatica.com.

Informatica-Dokumentation

Navigieren Sie zur Informatica-Wissensdatenbank unter https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx, um die aktuelle Dokumentation für Ihr Produkt abzurufen.

Wenn Sie Fragen, Kommentare oder Ideen zu dieser Dokumentation haben, wenden Sie sich per E-Mail an das Informatica-Dokumentationsteam unter infa_documentation@informatica.com.

Informatica-Produktverfügbarkeitsmatrizen

Produktverfügbarkeitsmatrizen (PAMs) geben die Versionen der Betriebssysteme, Datenbanken und anderen Typen von Datenquellen und Zielen an, die in einer Produktversion unterstützt werden. Als Mitglied des Informatica-Netzwerks können Sie unter https://network.informatica.com/community/informatica-network/product-availability-matrices auf PAMs zugreifen.

Informatica Velocity

Bei Informatica Velocity handelt es sich um eine Sammlung von Tipps und bewährten Methoden, die von den professionellen Informatica-Diensten entwickelt wurden. Informatica Velocity basiert auf der Praxiserfahrung aus Hunderten von Datenmanagementprojekten und umfasst das kollektive Wissen unserer Berater, die mit Unternehmen aus der ganzen Welt an der Planung, Entwicklung, Bereitstellung und Wartung erfolgreicher Datenmanagementlösungen gearbeitet haben.

Als Mitglied des Informatica-Netzwerks können Sie unter http://velocity.informatica.com auf Informatica Velocity-Ressourcen zugreifen.

Wenn Sie Fragen, Anregungen oder Ideen zu Informatica Velocity haben, wenden Sie sich an die professionellen Informatica-Dienste unter ips@informatica.com.

Informatica Marketplace

Informatica Marketplace ist ein Forum, das Lösungen zur Erweiterung und Verbesserung Ihrer Informatica-Implementierungen bereitstellt. Indem Sie die zahlreichen Lösungen von Informatica-Entwicklern und -Partnern nutzen, können Sie Ihre Produktivität steigern und die Implementierungsdauer Ihrer Projekte verkürzen. Zugriff auf den Informatica Marketplace erhalten Sie unter https://marketplace.informatica.com.

Globaler Kundensupport von Informatica

Sie können sich telefonisch oder über den Online-Support mit einem globalen Support-Center im Informatica-Netzwerk in Verbindung setzen.

Die Telefonnummer des globalen Kundensupports von Informatica vor Ort finden Sie auf der Informatica-Website unter folgender Verknüpfung: http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

Als Mitglied des Informatica-Netzwerks können Sie den Online-Support unter http://network.informatica.com verwenden.

KAPITEL 1

Einführung in Umwandlungen

Dieses Kapitel umfasst die folgenden Themen:

- Einführung in Umwandlungen - Übersicht, 30
- Verarbeiten von Umwandlungsdatentypen, 34
- Entwickeln einer Umwandlung, 37
- Umwandlungsports, 37
- Multi-Gruppen-Umwandlungen, 38
- Ausdrücke in Umwandlungen, 39
- Lokale Variablen, 44
- Standardwerte für Ports, 47
- Tracingebenen, 54
- Wiederverwendbare Umwandlungen, 55
- Nicht wiederverwendbare Umwandlungen, 56
- Erstellen einer Umwandlung, 57

Einführung in Umwandlungen - Übersicht

Bei einer Umwandlung handelt es sich um ein Objekt, das Daten erzeugt, ändert oder übergibt.

Informatica Developer enthält eine Reihe von Umwandlungen, die bestimmte Funktionen durchführen. Eine Aggregator-Umwandlung führt beispielsweise Berechnungen für Datengruppen durch.

Umwandlungen in einem Mapping stellen die Operationen dar, die der Data Integration Service mit den Daten durchführt. Daten durchlaufen Umwandlungsports, die Sie in einem Mapping oder Mapplet verknüpft haben.

Es gibt aktive oder passive Umwandlungen. Umwandlungen können mit dem Datenfluss verbunden werden. Die Verbindungen der Umwandlungen mit dem Datenfluss können aber auch aufgehoben werden.

Aktive Umwandlungen

Eine aktive Umwandlung ändert die Anzahl der Zeilen, die eine Umwandlung durchlaufen. Sie kann ebenfalls den Zeilentyp ändern.

Die Filterumwandlung ist beispielsweise aktiv, weil sie Zeilen entfernt, die nicht die Filterbedingung erfüllen. Die Update-Strategie-Umwandlung ist aktiv, weil sie Zeilen zum Einfügen, Löschen, Aktualisieren oder Ablehnen kennzeichnet.

Sie kann nicht mehrere aktive Umwandlungen oder eine aktive und eine passive Umwandlung mit derselben nachgelagerten Umwandlung oder der Umwandlungs-Eingabegruppe verbinden, weil der Data Integration Service die von aktiven Umwandlungen übergebenen Zeilen möglicherweise nicht verketten kann.

Eine Verzweigung in einem Mapping enthält eine Update-Strategie-Umwandlung, die eine Zeile zum Löschen kennzeichnet. Ein anderer Verzweigung enthält eine Update-Strategie-Umwandlung, die eine Zeile zum Einfügen kennzeichnet. Wenn Sie diese Umwandlungen mit einer einzelnen Umwandlungs-Eingabegruppe verbinden, kann der Data Integration Service die Operationen zum Löschen und Einfügen für die Zeile nicht kombinieren.

Passive Umwandlungen

Eine passive Umwandlung ändert nicht die Anzahl der Zeilen, die die Umwandlung durchlaufen, und sie behält die Transaktionsgrenze und den Zeilentyp bei.

Sie können mehrere Umwandlungen mit derselben nachgelagerten Umwandlung oder Umwandlungseingabegruppe verbinden, wenn alle Umwandlungen in den vorangehenden Verzweigungen passiv sind. Die Umwandlung, die aus der Verzweigung stammt, kann aktiv oder passiv sein.

Nicht verbundene Umwandlungen

Umwandlungen können mit dem Datenfluss verbunden werden. Die Verbindungen der Umwandlungen mit dem Datenfluss können aber auch aufgehoben werden. Eine nicht verbundene Umwandlung ist nicht mit anderen Umwandlungen im Mapping verbunden. Eine nicht verbundene Umwandlung wird in einer anderen Umwandlung aufgerufen und gibt einen Wert an diese Umwandlung zurück.

Mehrfachstrategieumwandlungen

Eine Strategie ist ein Satz von mehreren Vorgängen, die eine Umwandlung für Daten durchführen kann. Sie können jeder Strategie in der Umwandlung einen anderen Satz an Eingabe- und Ausgabeports zuweisen. Die Umwandlung speichert die Strategien, die Sie in einem einzelnen Umwandlungsobjekt definiert haben.

Verwenden Sie die Ansicht **Abhängigkeiten anzeigen** zum Anzeigen der Ports, die jede Strategie verwendet.

Sie können mehrere Umwandlungsstrategien in den folgenden Umwandlungen definieren:

- Groß-/Kleinschreibungsumwandler
- Klassifizierer
- Entscheidung
- Schlüssel-Generator
- Labeler
- Match
- Zusammenführung
- Parser
  Um mehrere Strategien in einer Parserumwandlung zu verwenden, konfigurieren Sie die Umwandlung zum Parsen von Token.
- Standardisierer

Umwandlungsbeschreibungen

Das Developer tool enthält allgemeine sowie datenqualitätsbezogene Umwandlungen. Allgemeine Umwandlungen stehen in Informatica Data Quality und Informatica Data Services zur Verfügung. Datenqualitätsumwandlungen sind in Informatica Data Quality verfügbar.

In der folgenden Tabelle werden alle Umwandlungen beschrieben:

Umwandlung	Typ	Beschreibung
Adressvalidierer	Aktiv oder passiv/ Verbunden	Überprüft und verbessert die Genauigkeit von Postadressdatensätzen und fügt Informationen hinzu, die dem Benutzer dabei helfen, Postempfänger auszuwählen und Post zu versenden.
Zuordnung	Aktiv/ Verbunden	Erstellt Verknüpfungen zwischen duplizierten Datensätzen, die eine Match-Umwandlung unterschiedlichen Clustern zuweist.
Aggregator	Aktiv/ Verbunden	Führt Aggregatberechnungen durch.
Ausnahme bei fehlerhaftem Datensatz	Aktiv/ Verbunden	Ermittelt Datensätze, die möglicherweise Datenfehler enthalten, und lädt diese Datensätze in Tabellen, sodass ein Benutzer des Analyst Tools diese überprüfen und aktualisieren kann.
Groß-/ Kleinschreibungsumwandler	Passiv/ Verbunden	Standardisiert die Groß-/Kleinschreibung von Zeichenfolgen.
Klassifizierer	Passiv/ Verbunden	Schreibt Beschriftungen, die die Informationen in Eingabeportfelder zusammenfassen. Verwenden Sie Klassifizierer, wenn die Felder große Mengen an Text enthalten.
Vergleich	Passiv/ Verbunden	Generiert numerische Scores, die den Grad der Ähnlichkeit zwischen Paaren von Eingabezeichenfolgen angeben.
Konsolidierung	Aktiv/ Verbunden	Erstellt einen konsolidierten Datensatz aus Datensätzen, die von der Match-Umwandlung als Duplikate erkannt wurden.
Datenmaskierung	Passiv/ Verbunden oder nicht verbunden	Ersetzt vertrauliche Produktionsdaten durch realistische Testdaten für Nicht-Produktionsumgebungen.
Datenprozessor	Aktiv/ Verbunden	Verarbeitet unstrukturierte und halbstrukturierte Dateiformate in einem Mapping.
Entscheidung	Passiv/ Verbunden	Evaluiert Bedingungen in Eingabedaten und erstellt eine Ausgabe basierend auf den Ergebnissen dieser Bedingungen.
Ausnahme bei dupliziertem Datensatz	Aktiv/ Verbunden	Ermittelt Datensätze, die möglicherweise doppelte Informationen enthalten, und lädt diese Datensätze in Tabellen, sodass ein Benutzer des Analyst Tools diese überprüfen und aktualisieren kann.

Umwandlung	Typ	Beschreibung
Ausdruck	Passiv/ Verbunden	Berechnet einen Wert.
Filter	Aktiv/ Verbunden	Filtert Daten.
Hierarchisch in relational	Aktiv/ Verbunden	Verarbeitet hierarchische Eingaben und wandelt sie in relationale Ausgaben um.
Java	Aktiv oder passiv/ Verbunden	Führt in Java codierte Benutzerlogik aus. Das Repository speichert den Bytecode für die Benutzerlogik.
Joiner	Aktiv/ Verbunden	Führt Daten aus verschiedenen Datenbanken oder Einfachdateisystemen zusammen.
Schlüsselgenerator	Aktiv/ Verbunden	Weist Datensätze Gruppen auf Grundlage der Datenwerte in einer von Ihnen ausgewählten Spalte zu.
Beschriftung	Passiv/ Verbunden	Schreibt Beschriftungen, die Zeichen oder Zeichenfolgen in einem Eingabeportfeld beschreiben.
Lookup	Aktiv oder passiv/ Verbunden oder nicht verbunden	Nachschlagen und Rückgabe von Daten aus einer Einfachdatei, einem logischen Datenobjekt, einer Referenztabelle, einer relationalen Tabelle, einer Ansicht oder einem Synonym.
Übereinstimmung	Aktiv/ Verbunden	Generiert Scores, die den Grad der Ähnlichkeit zwischen Eingabedatensätzen angeben.
Zusammenführung	Passiv/ Verbunden	Liest die Datenwerte aus mehreren Eingabespalten und erstellt eine einzelne Ausgabespalte.
Normalisierer	Aktiv/ Verbunden	Verarbeitet Quellzeilen, die mehrfach vorkommende Daten enthalten, und gibt eine Zielezeile für jede Instanz der mehrfach vorkommenden Daten zurück.
Ausgabe	Passiv/ Verbunden	Definiert Mapplet-Ausgabezeilen.
Parser	Passiv/ Verbunden	Parst die Werte auf einem Eingabeport in separate Ausgabeports, basierend auf den in den Werten enthaltenen Informationstypen.
Rang	Aktiv/ Verbunden	Begrenzt Datensätze auf einen oberen und unteren Bereich.
Lesen	Passiv/ Verbunden	Liest Daten aus einer Quelle.
Relational-in-hierarchisch	Aktiv/ Verbunden	Verarbeitet relationale Eingaben und wandelt sie in hierarchische Ausgaben um.

Umwandlung	Typ	Beschreibung
REST-Webdienstbenutzer	Aktiv/ Verbunden	Verbindet sich als Webdienst-Client mit einem REST-Webdienst, um auf Daten zuzugreifen oder diese umzuwandeln.
Router	Aktiv/ Verbunden	Leitet Daten basierend auf den Gruppenbedingungen an zahlreiche Umwandlungen weiter.
Sequenzgenerator	Passiv/ Verbunden	Generiert eine numerische Sequenz aus Werten.
Sortierer	Aktiv/ Verbunden	Sortiert Daten auf Basis eines Sortierschlüssels.
SQL	Aktiv oder passiv/ Verbunden	Führt SQL-Abfragen anhand einer Datenbank durch.
Standardisierer	Passiv/ Verbunden	Generiert standardisierte Versionen von Eingabezeichenfolgen.
Union	Aktiv/ Verbunden	Führt Daten aus verschiedenen Datenbanken oder Einfachdateisystemen zusammen.
Aktualisierungsstrategie	Aktiv/ Verbunden	Legt fest, ob Zeilen eingefügt, gelöscht, aktualisiert oder abgelehnt werden sollen.
Webdienstbenutzer	Aktiv/ Verbunden	Verbindet sich als Webdienst-Client mit einem Webdienst, um auf Daten zuzugreifen oder diese umzuwandeln.
Gewichteter Durchschnitt	Passiv/ Verbunden	Liest die Übereinstimmungswerte, die eine Match-Umwandlung für die Datensätze in einem Dataset generiert, und berechnet für jedes Datensatzpaar einen Durchschnittsscore. Sie können die Scores, die die Umwandlung für jedes Datensatzpaar generiert, unterschiedlich gewichten.
Schreiben	Passiv/ Verbunden	Stellt ein Ziel dar, in welches das Mapping Daten schreibt.

Verarbeiten von Umwandlungsdatentypen

Umwandlungen können Funktionen für bestimmte Datentypen verarbeiten oder das Durchleiten von Daten zulassen, ohne die Daten zu verarbeiten. Der Datenintegrationsdienst verarbeitet einige Datentypen, wie z. B. „Dezimal“, „Zeitstempel mit Zeitzone“ und „Zeitstempel mit lokaler Zeitzone“ basierend auf Umwandlungen.

Datentyp „Dezimal“

Sie können den Datentyp „Dezimal“ verwenden, um Daten zu lesen und in Einfachdateien und unterstützte Datenbanken zu schreiben, wie z. B. Oracle, Microsoft SQL Server, IBM DB2 und ODBC.

Für Umwandlungen, die Unterstützung für eine Gesamtstellenanzahl von bis zu 38 Ziffern bieten, beträgt die Gesamtstellenanzahl 1 bis 38 Ziffern und die Dezimalstellen betragen 0 bis 38.

Umwandlungen mit Unterstützung des Datentyps „Dezimal“

Die folgenden Umwandlungen unterstützen den Datentyp „Dezimal“ mit einer Gesamtstellenanzahl von bis zu 38 Ziffern und können Datenberechnungen durchführen:

- Aggregator
- Datenmaskierung
- Ausdruck
- Filter
- Java
- Joiner
- Lookup
- Normalisierer
- Rang
- Router
- Sequenzgenerator
- Sortierer
- Union
- Aktualisierungsstrategie

Umwandlungen mit Pass-Through-Unterstützung für den Datentyp „Dezimal“

Einige Umwandlungen können nur Dezimaldaten mit einer Gesamtstellenanzahl von bis zu 38 Ziffern über die Umwandlung übergeben. Die Umwandlung kann keine Datenberechnungen durchführen. Wenn Sie den Datentyp „Dezimal“ mit einer Gesamtstellenanzahl von bis zu 38 Ziffern für die Durchführung von Berechnungen für die Umwandlungen verwenden, verarbeitet der Datenintegrationdienst den Datentyp als „Doppelt“.

Die folgenden Umwandlungen verfügen über Pass-Through-Unterstützung für den Datentyp „Dezimal“ mit einer Gesamtstellenanzahl von bis zu 38 Ziffern:

- Datenprozessor
- Hierarchisch in relational
- REST-Webdienstbenutzer-Umwandlung
- SQL
- Webdienstbenutzer

Umwandlungen ohne Unterstützung des Datentyps „Dezimal“

Einige Umwandlungen unterstützen den Datentyp „Dezimal“ nicht, wie z. B. Datenqualitätsumwandlungen.

Die Datenqualitätsumwandlungen auf der folgenden anwendbaren Liste unterstützen den Datentyp „Dezimal“ mit einer Gesamtstellenanzahl von bis zu 38 Ziffern nicht:

- Adressvalidierer

- Zuordnung
- Groß-/Kleinschreibungsumwandler
- Klassifizierer
- Vergleich
- Konsolidierung
- Entscheidung
- Schlüsselgenerator
- Labeler
- Match
- Zusammenführung
- Parser
- Standardisierer
- Gewichteter Durchschnitt

Wenn ein Datentyp „Dezimal“ eine Gesamtstellenanzahl von mehr als 28 Ziffern aufweist, aber eine Umwandlung den Datentyp „Dezimal“ mit 38 Ziffern nicht unterstützt, konvertiert der Datenintegrationsdienst Dezimalwerte im Modus für hohe Genauigkeit in Werte des Typs „Doppelt“.

Zeitstempel mit Zeitzone

„Zeitstempel mit Zeitzone“ ist eine Variante des Datentyps „Zeitstempel“, der eine Zeitzonenabweichung oder eine Zeitzonenregion beinhaltet.

Die folgenden Umwandlungen unterstützen den Datentyp „Zeitstempel mit Zeitzone“:

- Aggregator
- Ausdruck
- Filter
- Java
- Joiner
- Lookup
- Normalisierer
- Rang
- Router
- Sequenzgenerator
- Sortierer
- Union
- Aktualisierungsstrategie

Pass-Through-Unterstützung bedeutet, dass Sie die Daten über die Umwandlungen übergeben, jedoch keine Funktionen für den Datentyp „Zeitstempel“ ausführen können.

Die folgenden Umwandlungen verfügen über Pass-Through-Unterstützung für den Datentyp „Zeitstempel mit Zeitzone“:

- Datenmaskierung
- Datenprozessor

- Hierarchisch in relational
- SQL

Zeitstempel mit lokaler Zeitzone

„Zeitstempel mit lokaler Zeitzone“ wird implizit von Umwandlungen unterstützt, da dessen Funktionsweise der des Typs „Zeitstempel“ entspricht.

Entwickeln einer Umwandlung

Beim Erstellen eines Mappings fügen Sie Umwandlung hinzu und konfigurieren sie so, dass sie Daten entsprechend des jeweiligen Geschäftszwecks verarbeiten.

Führen Sie die folgenden Aufgaben durch, um eine Umwandlung zu entwickeln und in ein Mapping zu integrieren:

1. Fügen Sie eine nicht wiederverwendbare Umwandlung einem Mapping oder Mapplet hinzu. Oder erstellen Sie eine wiederverwendbare Umwandlung, die Sie zu mehreren Mappings oder Mapplets hinzufügen können.
2. Konfigurieren Sie die Umwandlung. Jede Umwandlungstyp weist einen eindeutigen Satz an Optionen auf, den Sie konfigurieren können.
3. Wenn die Umwandlung wiederverwendbar ist, fügen Sie sie zum Mapping oder Mapplet hinzu.
4. Verknüpfen Sie die Umwandlung mit anderen Objekten im Mapping oder Mapplet.

   Ziehen Sie Ports aus vorgelagerten Objekten in die Umwandlungseingabeports. Ziehen Sie Ausgabeports aus der Umwandlung für Ports für vorgelagerte Objekte. Einige Umwandlungen verwenden vordefinierte Ports, die Sie auswählen können.

Hinweis: Wenn Sie eine wiederverwendbare Umwandlung erstellen, fügen Sie die benötigten Eingabe- und Ausgabeports hinzu, bevor Sie die Umwandlung mit anderen Objekten verknüpfen. Sie können Ports auf dem Mapplet oder der Mapping-Arbeitsfläche zu der Umwandlungsinstanz hinzufügen. Um die Ports in eine wiederverwendbare Umwandlung zu aktualisieren, öffnen Sie das Umwandlungsobjekt aus dem Repositoryprojekt und fügen Sie die Ports hinzu.

Umwandlungsports

Wenn Sie eine Umwandlung erstellt haben, definieren Sie die Ports. Erstellen Sie die Ports und definieren Sie die Porteigenschaften.

Wenn Sie eine bestimmte Anzahl an Umwandlungen erstellen, müssen Sie nicht alle Ports manuell erstellen. Sie möchten beispielsweise eine Lookup-Umwandlung erstellen und eine Lookup-Tabelle referenzieren. Wenn Sie die Umwandlungsports anzeigen, können Sie sehen, dass die Umwandlung über einen Ausgabeport für jede Spalte in der Tabelle verfügt, die Sie referenziert haben. Sie müssen diese Ports nicht definieren.

Erstellen von Ports

Erstellen Sie einen Port wie folgt:

- **Ziehen Sie einen Port aus einer anderen Umwandlung.** Wenn Sie einen Port aus einer anderen Umwandlung ziehen, erstellt der Designer einen Port mit denselben Eigenschaften und verknüpft ihn mit zwei Ports.
- **Klicken Sie auf der Registerkarte „Ports“ auf die Schaltfläche „Hinzufügen“.** Der Designer erstellt einen leeren Port, den Sie konfigurieren können.

Konfigurieren von Ports

Beim Definieren der Umwandlungsports definieren Sie Porteigenschaften. Porteigenschaften enthalten den Portnamen, den Datentyp, den Porttyp und den Standardwert.

Konfigurieren Sie die folgenden Porteigenschaften:

- **Portname.** Der Name des Ports. Verwenden Sie die folgenden Konventionen beim Benennen von Ports:
  - Beginnen Sie mit einem Einzel- oder Doppelbyte-Buchstaben oder einem Einzel- oder Doppelbyte-Unterstrich (_).
  - Portnamen können alle der folgenden Einzel- oder Doppelbyte-Zeichen enthalten: ein Buchstabe, eine Zahl, ein Unterstrich (_), $, # oder @.
- **Datentyp, Gesamtstellenanzahl, Dezimalstellenanzahl** Wenn Sie einen Ausdruck oder eine Bedingung eingeben möchten, stellen Sie sicher, dass der Datentyp dem Rückgabewert des Ausdrucks entspricht.
- **Porttyp.** Umwandlungen können eine Kombination aus Eingabe-, Ausgabe-, Eingabe/Ausgabe- und variablen Porttypen enthalten.
- **Standardwert.** Weisen Sie einen Standardwert für einen Port zu, der Nullwerte oder einen Ausgabeumwandlungsfehler enthält. Sie können die Standardwerte in einigen Ports überschreiben.
- **Beschreibung.** Eine Beschreibung des Ports.
- **Weitere Eigenschaften.** Einige Umwandlungen verfügen über Eigenschaften, die dieser Umwandlung entsprechen, wie zum Beispiel Ausdrücke oder Group By-Eigenschaften.

Multi-Gruppen-Umwandlungen

Eine Umwandlung kann mehrere Eingabe- und Ausgabegruppen enthalten. Eine Gruppe ist eine Reihe von Ports, die eine Zeile von eingehenden und ausgehenden Daten definieren.

Eine Gruppe ist analog zu einer Tabelle in einer relationalen Quelle oder Zieldefinition. Die meisten Umwandlungen haben eine Eingabe- und eine Ausgabegruppe. Einige verfügen jedoch über mehrere Eingabegruppen, mehrere Ausgabegruppen oder beides. Eine Gruppe ist die Darstellung einer Datenzeile, die in eine Umwandlung eintritt oder diese verlässt.

Alle Multi-Gruppen-Umwandlungen sind aktive Umwandlungen. Sie können nicht mehrere aktive Umwandlungen oder eine aktive und eine passive Umwandlung in derselben nachgeschalteten Umwandlung oder Umwandlungseingabegruppe verbinden.

Einige Mehrfach-Eingabegruppenumwandlungen benötigen den Integrationsdienst, um Daten an einer Eingabegruppe zu blockieren, während der Integrationsdienst auf eine Zeile aus einer anderen Eingabegruppe

wartet. Eine blockierende Umwandlung ist eine Mehrfach-Eingabegruppen-Umwandlung, die eingehende Daten blockiert. Die folgenden Umwandlungen sind blockierende Umwandlungen:

- Benutzerdefinierte Umwandlung mit aktivierter Eigenschaft „Eingaben können blockieren“
- Für nicht sortierte Eingabe konfigurierte Joiner-Umwandlung

Wenn Sie ein Mapping speichern oder validieren, sind einige Mappings, die aktive oder blockierende Umwandlungen enthalten, möglicherweise nicht gültig.

Regeln und Richtlinien für Multi-Gruppen-Umwandlungen

Wenn Sie Umwandlungen in einem Mapping verbinden, müssen Sie einige Regeln und Richtlinien zum Verbinden von Multi-Gruppen-Umwandlungen beachten.

Beachten Sie die folgenden Regeln und Richtlinien für Multi-Gruppen-Umwandlungen:

- Sie können eine Gruppe mit einer Umwandlung oder einem Ziel verbinden.
- Sie können einen oder mehrere Ausgabeports in einer Gruppe mit mehreren Umwandlungen oder Zielen verbinden.
- Sie können keine Felder aus mehreren Ausgabegruppen in einer Umwandlung mit derselben Ausgabegruppe in einer anderen Umwandlung verbinden.
- Sie können keine Felder aus mehreren Ausgabegruppen in verschiedenen Umwandlungen mit derselben Eingabegruppe einer anderen Umwandlung verbinden, es sei denn, alle Umwandlungen zwischen der Quelle und der Umwandlung sind passive Umwandlungen.
- Sie können keine Felder aus mehreren Ausgabegruppen in einer Umwandlung mit derselben Ausgabegruppe in einer anderen Umwandlung verbinden, es sei denn, bei der anderen Umwandlung handelt es sich um eine blockierende Umwandlung.
- Sie können ein Ausgabefeld nicht mit mehreren Eingabefeldern in derselben Eingabegruppe verbinden, es sei denn, die Gruppe ist in einer Normalizer-Umwandlung enthalten.

Ausdrücke in Umwandlungen

Sie können Ausdrücke im **Ausdruckseditor** in einigen Umwandlungen eingeben. Ausdrücke ändern Daten oder überprüfen, ob Daten mit Bedingungen übereinstimmen.

Erstellen Sie Ausdrücke, die Umwandlungssprachfunktionen verwenden. Umwandlungssprachfunktionen sind SQL-ähnliche Funktionen, die Daten umwandeln.

Geben Sie einen Ausdruck in einen Port ein, der den Datenwert aus einem Eingabe- oder Eingabe-/Ausgabeport verwendet. Sie haben beispielsweise eine Umwandlung mit den Eingabeport IN_SALARY, der die Gehälter aller Mitarbeiter enthält. Verwenden Sie die Werte aus IN_SALARY-Spalte später im Mapping. Sie können die Umwandlung auch zum Berechnen des Gesamt- und Durchschnittsgehalts verwenden. Für das Developer Tool ist es erforderlich, dass Sie einen separaten Eingabeport für alle berechneten Daten erstellen.

In der folgenden Tabelle werden die Umwandlungen aufgelistet, in die Sie Ausdrücke eingeben können:

Umwandlung	Ausdruck	Rückgabewert
Aggregator	Führt eine Aggregat-Berechnung basierend auf allen durch die Umwandlung geleiteten Daten durch. Alternativ dazu können Sie einen Filter für Datensätze in der Aggregat-Berechnung festlegen, um bestimmte Arten von Datensätzen auszuschließen. Mit dieser Umwandlung können Sie beispielsweise die Gesamtanzahl und das Durchschnittsgehalt aller Mitarbeiter in einer Filiale anzeigen.	Ergebnis einer Aggregat-Berechnung für einen Port.
Ausdruck	Führt eine Berechnung basierend auf Werten innerhalb einer einzelnen Zeile durch. Basierend auf dem Preis und der Menge eines bestimmten Artikels können Sie zum Beispiel den Gesamteinkaufspreis für diese Position in einem Auftrag berechnen.	Ergebnis einer Berechnung auf Zeilenebene für einen Port.
Filter	Gibt eine Bedingung an, die zum Filtern von Zeilen verwendet wird, die über diese Umwandlung übergeben werden. Wenn Sie beispielsweise Kundendaten in die Tabelle BAD_DEBT für Kunden mit offenem Guthaben schreiben möchten, können Sie die Filterumwandlung zum Filtern von Kundendaten verwenden.	TRUE oder FALSE basierend darauf, ob eine Zeile die angegebene Bedingung erfüllt. Der Datenintegrationsdienst übergibt über diese Umwandlung Zeilen, die TRUE zurückgeben. Die Umwandlung wendet diesen Wert auf jede Zeile an, die über die Umwandlung übergeben wird.
Joiner	Gibt eine erweiterte Bedingung zum Zusammenfügen von unsortierten Quelldaten an. Sie können beispielsweise Vorname- und Nachname-Master-Ports verketten und diese anschließend mit dem Detail-Port für vollständige Namen abgleichen.	TRUE oder FALSE basierend darauf, ob die Zeile die angegebene Bedingung erfüllt. Je nach ausgewähltem Join-Typ fügt der Datenintegrationsdienst die Zeile entweder zum Ergebnissatz hinzu oder verwirft die Zeile.
Rang	Legt die Bedingungen für die in einem Rang enthaltenen Zeilen fest. Sie können beispielsweise die besten 10 Vertriebsmitarbeiter klassifizieren, die im Unternehmen beschäftigt sind.	Ergebnis einer Bedingung oder Berechnung für einen Port.

Umwandlung	Ausdruck	Rückgabewert
Router	Leitet Daten an mehrere Umwandlungen basierend auf einem Gruppenausdruck weiter. Verwenden Sie diese Umwandlung zum Beispiel, um die Gehälter der Mitarbeiter auf drei verschiedenen Gehaltsstufen zu vergleichen. Dafür können Sie drei Gruppen in der Router-Umwandlung erstellen. Erstellen Sie beispielsweise einen Gruppenausdruck für jeden Gehaltsbereich.	TRUE oder FALSE basierend darauf, ob eine Zeile den angegebenen Gruppenausdruck erfüllt. Der Datenintegrationsdienst übergibt Zeilen, die TRUE über jede benutzerdefinierte Gruppe in dieser Umwandlung zurückgeben. Zeilen, die FALSE zurückgeben, werden über die Standardgruppe übergeben.
Aktualisierungsstrategie	Kennzeichnet eine Zeile zum Aktualisieren, Einfügen, Löschen oder Ablehnen. Verwenden Sie diese Umwandlung, wenn Sie Aktualisierungen für ein Ziel basierend auf einigen von Ihnen angewendeten Bedingungen steuern möchten. Sie können beispielsweise die Aktualisierungsstrategie-Umwandlung verwenden, um alle Kundenzeilen für die Aktualisierung zu kennzeichnen, wenn sich die Postanschrift geändert hat. Alternativ dazu können Sie alle Mitarbeiterzeilen zum Ablehnen von Personen kennzeichnen, die nicht mehr für das Unternehmen arbeiten.	Numerischer Code zum Aktualisieren, Einfügen, Löschen oder Ablehnen. Die Umwandlung wendet diesen Wert auf jede Zeile an, die über die Umwandlung übergeben wurde.

Der Ausdrucks-Editor

Verwenden Sie den **Ausdrucks-Editor** zum Erstellen von SQL-ähnlichen Anweisungen.

Sie können einen Ausdruck manuell eingeben oder die Point-and-Click-Methode verwenden. Wählen Sie Funktionen, Ports, Variablen und Operatoren in der Point-and-Click-Schnittstelle aus, um Fehler bei der Erstellung von Ausdrücken zu vermeiden. Ein Ausdruck kann maximal 32.767 Zeichen beinhalten.

Portnamen in einem Ausdruck

Sie können Umwandlungs-Portnamen in einem Ausdruck eingeben.

Wenn Sie für verbunden Umwandlungen Portnamen in einem Ausdruck verwenden, aktualisiert das Developer Tool diesen Ausdruck, wenn Sie Portnamen in der Umwandlung ändern. Sie schreiben beispielsweise einen Ausdruck, der den Unterschied zwischen zwei Daten festlegt: Date_Promised und Date_Delivered. Wenn Sie den Date_Promised-Portnamen in Due_Date ändern, ändert das Developer Tool den Date_Promised-Portnamen im Ausdruck in Due_Date.

Hinweis: Sie können den Namen Due_Date an andere nicht wiederverwendbare Umwandlungen verteilen, die von diesem Port im Mapping abhängig sind.

Hinzufügen von einem Ausdruck zu einem Port

Sie können einen Ausdruck zu einem Ausgabeport hinzufügen.

1. Wählen Sie in der Umwandlung den Port aus und öffnen Sie den **Ausdrucksedit**or.
2. Geben Sie den Ausdruck ein. Verwenden Sie die Registerkarten „Funktionen“ und „Ports“ und die Operatorschlüssel.

   Hinweis: Sie können in einem Ausdruck kein Escape-Zeichen verwenden. Wenn Sie ein Escape-Zeichen in den Ausdruck einfügen, zeigt das Developer Tool möglicherweise einen Parsing-Fehler an.

3. Optional können Sie Kommentare zum Ausdruck hinzufügen.

   Verwenden Sie die Kommentarindikatoren -- oder //.
4. Klicken Sie auf das Validieren-Symbol, um den Ausdruck zu validieren.
5. Klicken Sie auf **OK**.
6. Wenn der Ausdruck nicht gültig ist, beheben Sie die Validierungsfehler und validieren Sie den Ausdruck erneut.
7. Wenn der Ausdruck gültig ist, klicken Sie auf **OK**, um den **Ausdruckseditor** zu schließen.

Kommentare in einem Ausdruck

Sie können einem Ausdruck Kommentare hinzufügen, um den Ausdruck zu beschreiben oder um eine gültige URL für den Zugriff auf Geschäftsdokumentation über den Ausdruck anzugeben.

Verwenden Sie zum Hinzufügen von Kommentaren im Ausdruck die Kommentarindikatoren -- oder //.

Ausdrucksvalidierung

Sie müssen einen Ausdruck validieren, um ein Mapping auszuführen oder die Mapplet-Ausgabe in der Vorschau anzuzeigen.

Verwenden Sie die Schaltfläche "Validieren" im **Ausdruckseditor**, um einen Ausdruck zu validieren. Wenn Sie einen Ausdruck nicht validieren, wird er beim Schließen des **Ausdruckseditors** vom Developer Tool validiert. Wenn der Ausdruck ungültig ist, zeigt das Developer Tool eine Warnung an. Sie können den ungültigen Ausdruck speichern oder ändern.

Testausdrücke

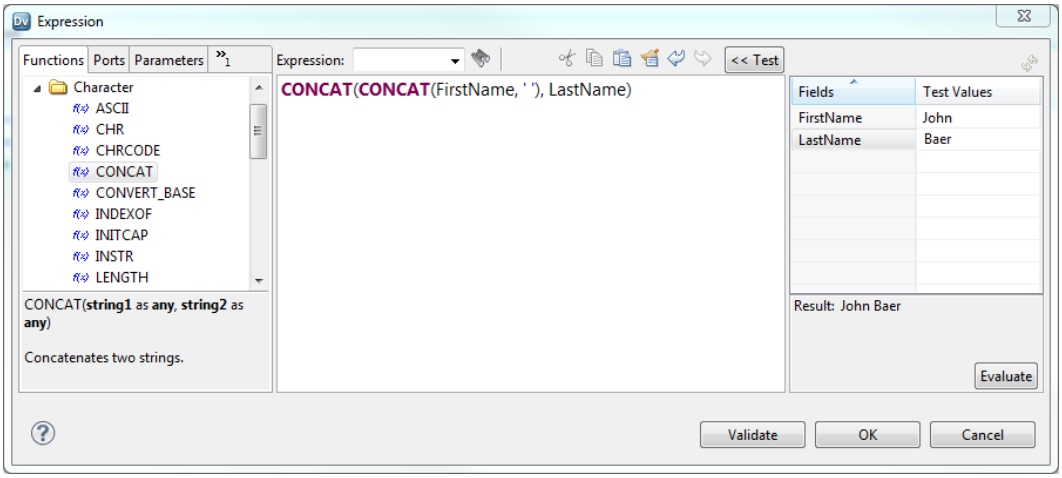
Sie können einige im Ausdruckseditor konfigurierte Ausdrücke testen. Wenn Sie einen Ausdruck testen, geben Sie Beispieldaten ein und werten den Ausdruck anschließend aus.

Sie können Ausdrücke testen, wenn Sie die Ausdrücke wie folgt konfigurieren:

- In einem Ausgabeport oder variablen Port in der Ausdrucksumwandlung
- In der Ansicht „Mapping-Ausgaben“ einer Ausdrucksumwandlung, nachdem die Umwandlung zu einem Mapping hinzugefügt wurde

Beispiel: Nach dem Konfigurieren eines Ausdrucks, der den Vornamen, ein Leerzeichen und den Nachnamen verkettet, können Sie Beispieldaten für die Ports eingeben und den Ausdruck anschließend auswerten, um das Ergebnis zu überprüfen.

Die folgende Abbildung zeigt die Ergebnisse eines Ausdrucks, bei dem ein Beispielvorname und -nachname verkettet werden:



Datumsformatzeichenfolgen für Beispieldaten

Wenn Sie einen Ausdruck testen, der einen Port des Datentyps „Datum/Uhrzeit“ oder „Zeitstempel mit Zeitzone“ verwendet, müssen Sie Beispieldaten für den Port mit der erforderlichen Zeichenfolge für das Datumsformat eingeben.

Für die Eingabe von Beispieldaten für einen Port des Datentyps „Datum/Uhrzeit“ verwenden Sie das Format MM/DD/YYYY HH24:MI:SS. Wenn Sie den Ausdruck auswerten, zeigt der Ausdruckseditor das Ergebnis im von Ihnen im Ausdruck angegebenen Format an. Wenn Sie die Formatzeichenfolge im Ausdruck weglassen, zeigt der Ausdruckseditor das Ergebnis in demselben Format MM/DD/YYYY HH24:MI:SS an.

Für die Eingabe von Beispieldaten für einen Port des Datentyps „Zeitstempel mit Zeitzone“ verwenden Sie das Format MM/DD/YYYY HH24:MI:SS TZR. Wenn Sie den Ausdruck auswerten, zeigt der Ausdruckseditor das Ergebnis im Format YYYY-MM-DD HH24:MI:SS.NS TZR an.

Datentypumwandlung

Durch mehrere Ausdrucks- und Aggregationsfunktionen können Daten eines Datentyps generiert werden, die von den Eingabedaten abweichen.

Wenn Sie zum Beispiel zwei Dezimalzahlen mit einer Gesamtstellenanzahl von 18 Ziffern multiplizieren, handelt es sich bei dem resultierenden Datentyp möglicherweise um eine Dezimalzahl mit einer Gesamtstellenanzahl von 28 Ziffern.

Beim Eingabedatentyp „Dezimal“ mit einer Gesamtstellenanzahl von 38 Ziffern können durch das Ergebnis bestimmter Vorgänge Daten erzeugt werden, die möglicherweise nicht zu dem resultierenden Datentyp passen. So kann es u. U. zu einer Überlaufausnahme kommen.

Die folgenden Funktionen erfordern möglicherweise eine Datentypumwandlung, um eine erhöhte Datengröße anpassen zu können, wenn die Daten mit den Eingabedatentypen verglichen werden:

- avg
- cume
- divide
- median
- movingavg

- movingsum
- multiply
- Percentile
- Sum

Wenn beispielsweise die Eingabedaten den Datentyp „Ganzzahl“ aufweisen und Sie den Multiplikationsvorgang verwenden, handelt es sich bei dem resultierenden Datentyp möglicherweise um den Datentyp „Große Ganzzahl“. Wenn gleichermaßen die Eingabedaten den Eingabedatentyp „Dezimal“ mit einer Gesamtstellenanzahl von 18 Ziffern aufweisen, ist das Ergebnis des Multiplikationsvorgangs möglicherweise groß und beläuft sich innerhalb des Datentyps „Dezimal“ auf eine Gesamtstellenanzahl von 28 Ziffern.

Lokale Variablen

Verwenden Sie lokale Variablen in Aggregator-, Ausdrucks- und Rang-Umwandlungen, um die Leistung zu verbessern. Sie können Variablen in einem Ausdruck referenzieren oder diese verwenden, um Daten temporär zu speichern.

Verwenden Sie Variablen, um die folgenden Aufgaben abzuschließen:

- Daten temporär speichern.
- Komplexe Ausdrücke vereinfachen.
- Werte aus vorherigen Zeilen speichern.
- Mehrere Rückgabewerte aus einer gespeicherten Prozedur erfassen.
- Werte vergleichen.
- Die Ergebnisse einer nicht verbundenen Lookup-Umwandlung speichern.

Temporäres Speichern von Daten und Vereinfachen von komplexen Ausdrücken

Variablen erhöhen die Leistung, wenn Sie mehrere zugehörige Ausdrücke in derselben Umwandlung eingeben. Sie können Komponenten als Variablen definieren, anstatt dieselben Ausdruckskomponenten mehrmals in der Umwandlung zu parsen und zu validieren.

Beispiel: Wenn eine Aggregatorumwandlung vor dem Berechnen der Summen und Durchschnittswerte dieselben Filterbedingungen verwendet, können Sie diese Bedingung als eine Variable definieren und dann die Bedingung in beiden Aggregat-Berechnungen wiederverwenden.

Sie können komplexe Ausdrücke vereinfachen. Wenn ein Aggregator dieselbe Berechnung in mehreren Ausdrücken enthält, können Sie die Leistung erhöhen, indem Sie eine Variable zum Speichern der Ergebnisse der Berechnung erstellen.

Beispiel: Sie können die folgenden Ausdrücke erstellen, um das durchschnittliche Gehalt und das Gesamtgehalt mit denselben Daten zu finden:

```
AVG( SALARY, ( ( JOB_STATUS = 'Full-time' ) AND (OFFICE_ID = 1000 ) ) )
SUM( SALARY, ( ( JOB_STATUS = 'Full-time' ) AND (OFFICE_ID = 1000 ) ) )
```

Anstatt dieselben Argumente für beide Berechnungen einzugeben, können Sie einen Variablenport für jede Bedingung in dieser Berechnung erstellen und dann den Ausdruck ändern, um die Variablen zu verwenden.

In der folgenden Tabelle wird aufgezeigt, wie Sie Variablen verwenden können, um komplexe Ausdrücke zu vereinfachen und Daten temporär zu speichern:

Port	Wert
V_CONDITION1	JOB_STATUS = 'Full-time'
V_CONDITION2	OFFICE_ID = 1000
AVG_SALARY	AVG(SALARY, (V_CONDITION1 AND V_CONDITION2))
SUM_SALARY	SUM(SALARY, (V_CONDITION1 AND V_CONDITION2))

Speichern von Werten in mehreren Zeilen

Sie können Variablen in Umwandlungen konfigurieren, um Daten aus Quellzeilen zu speichern. Sie können Variablen in Umwandlungsausdrücken verwenden.

Eine Quelldatei enthält beispielsweise folgende Zeilen:

```
California
California
California
Hawaii
Hawaii
New Mexico
New Mexico
New Mexico
```

Jede Zeile enthält einen Status. Sie müssen die Anzahl der Zeilen zählen und die Zeilenanzahl für jeden Status zurückgeben:

```
California,3
Hawaii    ,2
New Mexico,3
```

Sie können eine Aggregatorumwandlung konfigurieren, um die Quellzeilen nach Status zu gruppieren und die Anzahl der Zeilen in jeder Gruppe zu zählen. Konfigurieren Sie eine Variable in der Aggregatorumwandlung, um die Zeilenanzahl zu speichern. Definieren Sie eine andere Variable, um den Statusnamen aus der vorherigen Zeile zu speichern.

Die Aggregatorumwandlung verfügt über die folgenden Ports:

Port	Porttyp	Ausdruck	Beschreibung
Bundesland	Pass-through	`n/a`	Der Name eines Status. Die Quellzeilen werden nach Statusname gruppiert. Die Aggregatorumwandlung gibt eine Zeile für jeden Status zurück.
State_Count	Variable	`IIF (PREVIOUS_STATE = STATE, STATE_COUNT +1, 1)`	Die Zeilenanzahl für den aktuellen Status. Wenn der Wert der aktuellen Statusspalte der Previous_State-Spalte entspricht, erhöht der Integrationsdienst den State_Count-Status. Andernfalls wird State_Count auf 1 festgelegt.

Port	Porttyp	Ausdruck	Beschreibung
Previous_State	Variable	`Status`	Der Wert der Statusspalte in der vorherigen Zeile. Wenn der Integrationsdienst eine Zeile verarbeitet, verschiebt er den Statuswert in Previous_State.
State_Counter	Ausgabe	`State_Count`	Die Anzahl der Zeilen, die die Aggregatorumwandlung für einen Status verarbeitet hat. Der Integrationsdienst gibt für jeden Status einmal State_Counter zurück.

Erfassen von Werten aus gespeicherten Prozeduren

Variablen liefern eine Methode zum Erfassen von mehreren Spalten von Rückgabewerten aus gespeicherten Prozeduren.

Richtlinien zum Konfigurieren von variablen Ports

Beachten Sie die folgenden Faktoren beim Konfigurieren von Variablenports in einer Umwandlung:

- **Portreihenfolge.** Der Integrationsdienst bewertet Ports nach Abhängigkeit.. Die Reihenfolge der Ports in einer Umwandlung muss der Reihenfolge der Bewertung entsprechen: Eingabeports, Variablenports, Ausgabeports.
- **Datentyp.** Der von Ihnen ausgewählte Datentyp widerspiegelt den Rückgabewert des von Ihnen eingegebenen Ausdrucks.
- **Variableninitialisierung.** Der Integrationsdienst legt Anfangswerte in Variablenports fest, in denen Sie Zähler erstellen können.

Portreihenfolge

Der Integrationsdienst bewertet zuerst die Eingabeports. Anschließend bewertet der Integrationsdienst die Variablenports und zuletzt die Ausgabeports.

Der Integrationsdienst bewertet die Ports in folgender Reihenfolge:

1. **Eingabeports.** Der Integrationsdienst bewertet zuerst alle Eingabeports, da diese nicht von anderen Ports abhängig sind. Sie können Eingabeports daher in beliebiger Reihenfolge erstellen. Der Integrationsdienst sortiert Eingabeports nicht, da sich Eingabeports nicht auf andere Ports beziehen.
2. **Variablenports.** Variablenports können auf Eingabeports und Variablenports verweisen, aber nicht auf Ausgabeports. Da Variablenports auf Eingabeports verweisen können, bewertet der Integrationsdienst Variablenports nach Eingabeports. Variablen können auf andere Variablen verweisen, sodass die Anzeigereihenfolge für Variablenports der Reihenfolge entspricht, mit der der Integrationsdienst jede Variable bewertet.

   Wenn Sie beispielsweise den ursprünglichen Wert eines Gebäudes berechnen und dann eine Abschreibungskorrektur vornehmen, erstellen Sie unter Umständen die Berechnung des ursprünglichen Werts als Variablenport. Der Variablenport muss vor dem Port mit Abschreibungskorrektur angezeigt werden.
3. **Ausgabeports.** Der Integrationsdienst bewertet Ausgabeports zuletzt, da Ausgabeports auf Eingabeports und Variablenports verweisen können. Die Anzeigereihenfolge für Ausgabeports ist unerheblich, da Ausgabeports nicht auf andere Ausgabeports verweisen können. Ausgabeports werden unten in der Portliste angezeigt.

Datentyp

Wenn Sie einen Port als eine Variable konfigurieren, können Sie alle Audrücke oder Bedingungen eingeben. Der von Ihnen für diesen Port ausgewählte Datentyp gibt den Rückgabewert des eingegebenen Ausdrucks wieder. Wenn Sie eine Bedingung über den Variablenport eingeben, geben alle numerischen Datentypen die Werte für TRUE (nicht null) und FALSE (null) zurück.

Variableninitialisierung

Der Integrationsdienst legt den Anfangswert für Variablen nicht mit NULL fest.

Der Integrationsdienst verwendet die folgenden Richtlinien, um Anfangswerte für Variablen festzulegen:

- Null für numerische Ports
- Leere Zeichenfolgen für Zeichenfolgenports
- 01/01/0001 für Date/Time-Ports

Verwenden Sie daher Variablen als Zähler, die einen Anfangswert benötigen. Beispiel: Sie können eine numerische Variable mit dem folgenden Ausdruck erstellen:

```
VAR1 + 1
```

Dieser Ausdruck zählt die Anzahl der Zeilen im VAR1-Port. Wenn der Anfangswert der Variable auf NULL festgelegt ist, wird der Ausdruck immer auf NULL gesetzt. Aus diesem Grund wird der Anfangswert auf null gesetzt.

Standardwerte für Ports

Alle Umwandlungen verwenden Standardwerte, die festlegen, die der Integrationsdienst Eingabenullwerte und Ausgabeumwandlungsfehler verarbeitet.

Eingabe-, Ausgabe- und Eingabe-/Ausgabeports weisen einen Systemstandardwert auf, den Sie in einigen Fällen mit einem benutzerdefinierten Standardwert überschreiben können. Standardwerte haben verschiedene Funktionen in verschiedenen Porttypen:

- **Eingabeport.** Der Systemstandardwert für Null-Eingabeports ist NULL. Die Anzeige für den Standardwert in der Umwandlung ist leer. Wenn ein Eingabewert NULL ist, behält der Integrationsdienst den Wert NULL bei.
- **Ausgabeport.** Der Systemstandardwert für Ausgabeumwandlungsfehler ist ERROR. Der Standardwert wird in der Umwandlung als ERROR(„Umwandlungsfehler“) angezeigt. Wenn ein Umwandlungsfehler auftritt, überspringt der Integrationsdienst die Zeile. Der Integrationsdienst zeichnet alle Eingabezeilen, die die ERROR-Funktion überspringt, in der Protokolldatei auf.

  Bei den folgenden Fehlern handelt es sich um Umwandlungsfehler:

  - Datenkonvertierungsfehler, wie zum Beispiel die Übergabe einer Zahl an eine Datumsfunktion.
  - Ausdrucksbewertungsfehler, wie zum Beispiel das Teilen durch 0.
  - Aufruf an eine ERROR-Funktion.
- **Pass-Through-Port.** Der Systemstandardwert für Null-Eingabe ist mit NULL-Eingabeports identisch. Die Anzeige für den Systemstandardwert in der Umwandlung ist leer. Der Standardwert für Ausgabeumwandlungsfehler ist mit Ausgabeports identisch. Der Standardwert für Ausgabeumwandlungsfehler wird nicht in der Umwandlung angezeigt.

Hinweis: Die Java-Umwandlung wandelt PowerCenter-Datentypen basierend auf dem Porttyp der Java-Umwandlung in Java-Datentypen um. Standardwerte für Null-Eingabe unterscheiden sich basierend auf dem Java-Datentyp.

Die folgende Tabelle zeigt die Systemstandardwerte für Ports in verbundenen Umwandlungen an:

Porttyp	Standardwert	Integrationsdienst-Verhalten	Benutzerdefinierter Standardwert wird unterstützt
Eingabe, Pass-Through	NULL	Der Integrationsdienst übergibt alle Eingabe-Nullwerte als NULL.	Eingabe, Eingabe/Ausgabe
Ausgabe, Pass-Through	ERROR	Der Integrationsdienst ruft die ERROR-Funktion für Ausgabeport-Umwandlungsfehler auf. Der Integrationsdienst überspringt Zeilen mit Fehlern und schreibt die Eingabedaten und die Fehlermeldung in die Protokolldatei.	Ausgabe

Variablenports unterstützen keine Standardwerte. Der Integrationsdienst initialisiert Variablenports entsprechend dem Datentyp.

Sie können einige der Standardwerte überschreiben, um das Verhalten des Integrationsdiensts zu ändern, wenn dieser Null-Eingabewerte und Ausgabeumwandlungsfehler findet.

Benutzerdefinierte Standardwerte

Sie können die Systemstandardwerte mit benutzerdefinierten Standardwerten für unterstützte Eingabe-, Pass-Through- und Ausgabeports innerhalb einer verbundenen Umwandlung überschreiben.

Verwenden Sie die folgenden Regeln und Richtlinien, um die Systemstandardwerte für Ports zu überschreiben:

- **Eingabeports.** Sie können benutzerdefinierte Standardwerte für Eingabeports eingeben, wenn der Integrationsdienst Nullwerte nicht als NULL behandeln soll.
- **Ausgabeports.** Sie können benutzerdefinierte Standardwerte für Ausgabeports eingeben, wenn der Integrationsdienst die Zeile nicht überspringen soll oder der Integrationsdienst eine bestimmte Nachricht mit der übersprungenen Zeile ins Protokoll schreiben soll.
- **Pass-Through-Ports.** Sie können benutzerdefinierte Standardwerte für Pass-Through-Ports eingeben, wenn der Integrationsdienst Nullwerte nicht als NULL behandeln soll. Sie können keine benutzerdefinierten Standardwerte für Ausgabeumwandlungsfehler in einen Pass-Through-Port eingeben.

Hinweis: Der Integrationsdienst ignoriert benutzerdefinierte Standardwerte für nicht verbundene Umwandlungen. Wenn Sie zum Beispiel eine Lookup-Umwandlung oder eine Umwandlung einer gespeicherten Prozedur über einen Ausdruck aufrufen, ignoriert der Integrationsdienst alle benutzerdefinierten Standardwerte und wendet den Systemstandardwert an.

Verwenden Sie die folgenden Optionen, um benutzerdefinierte Standardwerte einzugeben:

- **Konstantenwert.** Verwenden Sie alle Konstanten (numerisch oder Text), einschließlich NULL.
- **Konstantenausdruck.** Sie können eine Umwandlungsfunktion in Konstantenparameter einbeziehen.
- **ERROR.** Generiert einen Umwandlungsfehler. Schreiben Sie die Zeile und eine Nachricht in das Mapping-Protokoll oder in das Zeilenfehlerprotokoll.
- **ABORT.** Brechen Sie das Mapping ab.

Konstantenwerte

Sie können jeden Konstantenwert als einen Standardwert eingeben. Der Konstantenwert muss dem Portdatentyp entsprechen.

Ein Standardwert für einen numerischen Port muss beispielsweise eine numerische Konstante sein. Einige Konstantenwerte umfassen:

```
0
9999
NULL
'Unknown Value'
'Null input data'
```

Konstantenausdrücke

Ein Konstantenausdruck ist ein beliebiger Ausdruck, der Umwandlungsfunktionen (außer Aggregatfunktionen) verwendet, um Konstantenausdrücke zu schreiben. Sie können keine Werte aus Eingabe-, Eingabe-/Ausgabe- oder Variablenports in einem Konstantenausdruck verwenden.

Einige gültige Konstantenausdrücke enthalten:

```
500 * 1.75
TO_DATE('January 1, 1998, 12:05 AM','MONTH DD, YYYY, HH:MI AM')
ERROR ('Null not allowed')
ABORT('Null not allowed')
SESSSTARTTIME
```

Sie können keine Werte aus Ports innerhalb des Ausdrucks verwenden, weil der Integrationsdienst Standardwerte für das gesamte Mapping zuweist, wenn es das Mapping initialisiert.

Die folgenden Beispiele sind nicht gültig, da sie Werte aus Ports verwenden:

```
AVG(IN_SALARY)
IN_PRICE * IN_QUANTITY
:LKP(LKP_DATES, DATE_SHIPPED)
```

Hinweis: Sie können keine gespeicherte Prozedur oder Lookup-Tabelle aus einem Standardwertausdruck aufrufen.

Die Funktionen ERROR und ABORT

Verwenden Sie die Funktionen ERROR und ABORT für Standardwerte für Eingabe- und Ausgabeports und Eingabewerte für Eingabe-/Ausgabeports. Der Integrationsdienst überspringt die Zeile, wenn er die ERROR-Funktion findet. Er bricht das Mapping ab, wenn er die ABORT-Funktion findet.

Benutzerdefinierte Standardeingabewerte

Sie können einen benutzerdefinierten Standardeingabewert eingeben, wenn der Integrationsdienst Nullwerte nicht als NULL behandeln soll.

Um Nullwerte zu überschreiben, führen Sie die folgenden Aufgaben durch:

- Ersetzen Sie den Nullwert mit einem Konstantenwert oder einem Konstantenausdruck.
- Überspringen Sie den Nullwert mit einer ERROR-Funktion.
- Brechen Sie das Mapping mit der ABORT-Funktion ab.

In der folgenden Tabelle wird zusammengefasst, wie der Integrationsdienst Null-Eingabe für Eingabe-/Ausgabeports verarbeitet:

Standardwert	Standardwerttyp	Beschreibung
NULL (leere Anzeige)	System	Integrationsdienst übergibt NULL.
Konstante oder Konstantenausdruck	Benutzerdefiniert	Der Integrationsdienst ersetzt den Nullwert mit Konstantenwert oder dem Konstantenausdruck.
ERROR	Benutzerdefiniert	Der Integrationsdienst behandelt dies als ein Umwandlungsfehler: - Erhöht die Umwandlungsfehleranzahl um 1. - Überspringt die Zeile und schreibt die Fehlermeldung in die Protokolldatei oder in das Zeilenfehlerprotokoll. Der Integrationsdienst schreibt die Zeilen nicht in die Ablehnungsdatei.
ABORT	Benutzerdefiniert	Das Mapping wird abgebrochen, wenn der Integrationsdienst einen Null-Eingabewert findet. Der Integrationsdienst erhöht nicht die Fehleranzahl bzw. schreibt keine Zeilen in die Ablehnungsdatei.

Ersetzen von Nullwerten

Verwenden Sie einen Konstantenwert oder einen Ausdruck, um einen angegebenen Wert für Nullwerte in einem Port anzugeben.

Wenn ein Eingabezeichenfolgenport beispielsweise DEPT_NAME lautet und Sie möchten Nullwerte mit der Zeichenfolge UNKNOWN DEPT ersetzen, können Sie den Standardwert auf UNKNOWN DEPT setzen. Je nach Umwandlung übergibt der Integrationsdienst UNKNOWN DEPT an einen Ausdruck oder an eine Variable innerhalb der Umwandlung oder an die nächste Umwandlung im Datenfluss.

Der Integrationsdienst ersetzt zum Beispiel alle Nullwerte in einem Port mit der Zeichenfolge UNKNOWN DEPT.

```
DEPT_NAME     REPLACED VALUE
Housewares      Housewares
NULL            UNKNOWN DEPT
Produce         Produce
```

Überspringen von Null-Datensätzen

Verwenden Sie die Funktion ERROR als den Standardwert, wenn keine Null-Werte an eine Umwandlung übergeben werden sollen. Sie möchten beispielsweise eine Zeile übersprungen, wenn NULL der Eingabewert von DEPT_NAME ist. Sie könnten den folgenden Ausdruck als Standardwert verwenden:

```
ERROR('Error. DEPT is NULL')
```

Wenn Sie die ERROR-Funktion als einen Standardwert verwenden, überspringt der Integrationsdienst die Zeile mit einem Null-Wert. Der Integrationsdienst schreibt alle von der ERROR-Funktion übersprungenen Zeilen in die Protokolldatei. Er schreibt diese Zielen nicht in die Ablehnungsdatei.

```
DEPT_NAME      RETURN VALUE
Housewares     Housewares
NULL           'Error. DEPT is NULL' (Row is skipped)
Produce        Produce
```

Das folgende Protokoll zeigt, an welcher Stelle der Integrationsdienst die Zeile mit dem Null-Wert überspringt:

```
TE_11019 Port [DEPT_NAME]: Default value is: ERROR(<<Transformation Error>> [error]:
Error. DEPT is NULL
... error('Error. DEPT is NULL')
).
CMN_1053 EXPTRANS: : ERROR: NULL input column DEPT_NAME: Current Input data:
CMN_1053   Input row from SRCTRANS: Rowdata: ( RowType=4 Src Rowid=2 Targ Rowid=2
  DEPT_ID (DEPT_ID:Int:): "2"
  DEPT_NAME (DEPT_NAME:Char.25:): "NULL"
  MANAGER_ID (MANAGER_ID:Int:): "1"
)
```

Abbrechen des Mappings

Verwenden Sie die Funktion ABORT, um ein Mapping abzubrechen, wenn der Integrationsdienst Null-Eingabewerte findet.

Standardwert der Validierung

Das Developer-Tool validiert Standardwerte, wenn Sie sie eingeben.

Das Developer-Tool validiert Standardwerte, wenn Sie ein Mapping speichern. Wenn Sie einen Standardwert eingeben, der nicht gültig ist, markiert das Developer-Tool das Mapping als ungültig.

Benutzerdefinierte Standardausgabewerte

Sie können benutzerdefinierte Standardwerte erstellen, um die Standardwerte für Ausgabeports zu überschreiben.

Sie können benutzerdefinierte Standardwerte für Ausgabeports eingeben, wenn der Integrationsdienst die Zeile mit Fehlern nicht überspringen soll oder der Integrationsdienst eine bestimmte Nachricht mit der übersprungenen Zeile ins Protokoll schreiben soll. Sie können Standardwerte eingeben, um die folgenden Funktionen auszuführen, wenn der Integrationsdienst Ausgabeumwandlungsfehler findet:

- Ersetzen Sie den Fehler mit einem Konstantenwert oder einem Konstantenausdruck. Der Integrationsdienst überspringt nicht die Zeile.
- Brechen Sie das Mapping mit der ABORT-Funktion ab.
- Schreiben Sie bestimmte Meldungen in das Protokoll für Umwandlungsfehler.

Sie können keine benutzerdefinierten Standardausgabewerte für Eingabe-/Ausgabeports eingeben.

Die folgende Tabelle enthält eine Zusammenfassung darüber, wie der Integrationsdienst Ausgabeport-Umwandlungsfehler und Standardwerte in Umwandlungen verarbeitet.

Standardwert	Standardwerttyp	Beschreibung
Umwandlungsfehler	System	Wenn ein Umwandlungsfehler auftritt und Sie den Standardwert nicht überschrieben haben, führt der Integrationsdienst die folgenden Aufgaben durch: - Erhöht die Umwandlungsfehleranzahl um 1. - Überspringt die Zeile und schreibt den Fehler und die Eingabezeile je nach Sitzungskonfiguration in die Sitzungsprotokolldatei oder in das Zeilenfehlerprotokoll. Der Integrationsdienst schreibt die Zeile nicht in die Ablehnungsdatei.
Konstante oder Konstantenausdruck	Benutzerdefiniert	Der Integrationsdienst ersetzt den Fehler mit dem Standardwert. Der Integrationsdienst erhöht nicht die Fehleranzahl bzw. schreibt keine Meldung ins Protokoll.
ABORT	Benutzerdefiniert	Das Mapping bricht ab und der Integrationsdienst schreibt eine Nachricht in das Protokoll. Der Integrationsdienst erhöht nicht die Fehleranzahl bzw. schreibt keine Zeilen in die Ablehnungsdatei.

Ersetzen von Fehlern

Wenn der Integrationsdienst beim Auftreten eines Umwandlungsfehlers keine Zeile überspringen soll, verwenden Sie eine Konstante oder einen Konstantenausdruck als Standardwert für einen Ausgabeport.

Wenn Sie beispielsweise über einen numerischen Ausgabeport namens NET_SALARY verfügen und den Konstantenwert 9999 bei Auftreten eines Umwandlungsfehlers verwenden möchten, weisen Sie dem NET_SALARY-Port den Standardwert 9999 zu. Wenn beim Berechnen des Werts von NET_SALARY ein Umwandlungsfehler (wie zum Beispiel das Teilen durch 0) auftritt, verwendet der Integrationsdienst den Standardwert 9999.

Abbrechen des Mappings

Verwenden Sie die ABORT-Funktion, um ein Mapping abzubrechen, wenn der Integrationsdienst Null-Eingabewerte findet.

Schreiben von Meldungen in das Mapping-Protokoll

Sie können einen benutzerdefinierten Standardwert in einem Ausgabeport konfigurieren, wenn der Integrationsdienst eine bestimmte Meldung im Mapping-Protokoll für eine übersprungene Zeile schreiben soll. Der Systemstandard ist FEHLER („Umwandlungsfehler“) und der Integrationsdienst schreibt die Meldung „Umwandlungsfehler“ und die übersprungene Zeile in das Protokoll. Sie können „Umwandlungsfehler“ ersetzen, wenn Sie eine andere Nachricht schreiben möchten.

ERROR-Funktionen in Ausgabeportausdrücken

Wenn Sie einen Ausdruck eingeben, der die ERROR-Funktion verwendet, kann der benutzerdefinierte Standardwert für den Ausgabeport die ERROR-Funktion im Ausdruck überschreiben.

Beispiel: Sie geben den folgenden Ausdruck ein, der den Integrationsdienst anweist, den Wert „Negative Sale“ zu verwenden, wenn er einen Fehler findet:

```
IIF( TOTAL_SALES>0, TOTAL_SALES, ERROR ('Negative Sale'))
```

Im folgenden Beispiel wird dargelegt, wie benutzerdefinierte Standardwerte die ERROR-Funktion im Ausdruck überschreiben können:

- **Konstantenwert oder Ausdruck.** Der Konstantenwert oder Ausdruck überschreibt die ERROR-Funktion im Ausgabeportausdruck.

  Wenn Sie zum Beispiel „0“ als Standardwert eingeben, überschreibt der Integrationsdienst die ERROR-Funktion im Ausgabeportausdruck. Er übergibt den Wert 0, wenn er einen Fehler findet. Er überspringt die Zeile nicht bzw. schreibt „Negative Sale“ nicht ins Protokoll.

- **ABORT.** Die ABORT-Funktion überschreibt die ERROR-Funktion im Ausgabeportausdruck.

  Wenn Sie die ABORT-Funktion als den Standardwert verwenden, bricht der Integrationsdienst ab, wenn ein Umwandlungsfehler auftritt. Die ABORT-Funktion überschreibt die ERROR-Funktion im Ausgabeportausdruck.

- **ERROR.** Wenn Sie die ERROR-Funktion als den Standardwert verwenden, enthält der Integrationsdienst die folgenden Informationen im Protokoll:

  - Fehlermeldung aus dem Standardwert

  - Fehlermeldung hat die ERROR-Funktion im Ausgabeportausdruck angegeben

  - Übersprungene Zeile

  Sie können beispielsweise den Standardwert mit der folgenden ERROR-Funktion überschreiben:

  ```
  ERROR('No default value')
  ```

  Der Integrationsdienst überspringt die Zeile und beinhaltet beide Fehlermeldungen im Protokoll.

  ```
   TE_7007 Transformation Evaluation Error; current row skipped...
   TE_7007 [<<Transformation Error>> [error]: Negative Sale
  ... error('Negative Sale')
  ]
  Sun Sep 20 13:57:28 1998
  TE_11019 Port [OUT_SALES]: Default value is: ERROR(<<Transformation Error>> [error]:
  No default value
  ... error('No default value')
  ```

Allgemeine Regeln für Standardwerte

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie Standardwerte erstellen:

- Der Standardwert muss entweder NULL, ein Konstantenwert, ein Konstantenausdruck, eine ERROR-Funktion oder eine ABORT-Funktion sein.
- Für Eingabe-/Ausgabeports verwendet der Integrationsdienst Standardwerte, um Null-Eingabewerte zu verarbeiten. Der Ausgabestandardwert von Eingabe-/Ausgabeports ist immer ERROR(„Umwandlungsfehler“).
- Variablenports verwenden keine Standardwerte.
- In der Aggregat-Umwandlung und in der Rang-Umwandlung können Sie Gruppen Standardwerte nach Ports zuweisen.
- Nicht alle Porttypen in allen Umwandlungen lassen benutzerdefinierte Standardwerte zu. Wenn ein Port keine benutzerdefinierten Standardwerte zulässt, ist das Feld für den Standardwert deaktiviert.
- Nicht alle Umwandlungen lassen benutzerdefinierte Standardwerte zu.
- Wenn eine Umwandlung nicht mit einem Mapping-Datenfluss verbunden ist, ignoriert der Integrationsdienst benutzerdefinierte Standardwerte.
- Wenn ein Eingabeport nicht verbunden ist, wird Null als sein Wert angenommen und der Integrationsdienst verwendet den Standardwert für diesen Eingabeport.

- Wenn ein Standardwert eines Eingabeports die ABORT-Funktion enthält und der Eingabewert ist NULL, beendet der Integrationsdienst das Mapping sofort. Verwenden Sie die ABORT-Funktion als Standardwert, um Null-Eingabewerte zu verhindern. Der erste Null-Wert in einem Eingabport beendet das Mapping.
- Wenn der Standardwert eines Ausgabeports die ABORT-Funktion enthält und ein Umwandlungsfehler für diesen Port auftritt, wird das Mapping sofort beendet. Verwenden Sie die ABORT-Funktion als Standardwert, um strikte Regeln für Umwandlungsfehler zu erzwingen. Der erste Umwandlungsfehler für diesen Port beendet das Mapping.
- Die ABORT-Funktion, Konstantenwerte und Konstantenausdrücke überschreiben in Ausgabeportausdrücken konfigurierte ERROR-Funktionen.

Standardwert der Validierung

Das Developer-Tool validiert Standardwerte, wenn Sie sie eingeben.

Das Developer-Tool validiert Standardwerte, wenn Sie ein Mapping speichern. Wenn Sie einen Standardwert eingeben, der nicht gültig ist, markiert das Developer-Tool das Mapping als ungültig.

Tracingebenen

Wenn Sie eine Umwandlung konfigurieren, können Sie festlegen, wie viele Details der Data Integration Service in das Protokoll schreibt.

Standardmäßig ist die Tracingebene für jede Umwandlung „Normal“. Ändern Sie die Tracingebene auf die Einstellung „Ausführlich“, wenn Sie die Fehlerbehebung einer Umwandlung vornehmen müssen, die nicht so arbeitet wie erwartet. Stellen Sie die Tracingebene auf „Kurz“ ein, wenn die Detailmenge im Protokoll erscheinen soll.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte „Erweitert“:

Tracingebene

Die Menge an Detail, die im Protokoll für eine Umwandlung erscheint.

In der folgenden Tabelle werden die Tracingebenen beschrieben:

Tracingebene	Beschreibung
Kurz	Protokolliert Initialisierungsinformationen, Fehlermeldungen und Benachrichtigungen zu abgelehnten Daten.
Normal	Protokolliert Initialisierungs- und Statusinformationen, entdeckte Fehler und aufgrund von Umwandlungszeilenfehlern übersprungene Zeilen. Fasst Mapping-Ergebnisse zusammen, aber nicht auf der Ebene von einzelnen Zeilen. Der Standardwert ist „Normal“.

Tracingebene	Beschreibung
Verbose-Initialisierung	Zusätzlich zum normalen Tracing werden auf dieser Ebene weitere Initialisierungsdetails, Namen von verwendeten Index- und Datendateien und detaillierte Umwandlungsstatistiken protokolliert.
Verbose-Daten	Zusätzlich zum Verbose-Initialisierungstracing werden auf dieser Ebene alle an das Mapping übergebenen Zeilen protokolliert. Stellt außerdem fest, an welcher Stelle Zeichenfolgedaten abgeschnitten wurden, um in eine Spalte zu passen, und bietet detaillierte Umwandlungsstatistiken. Beim Konfigurieren dieser Tracingebene werden Zeilendaten für alle Zeilen in einem Block in das Protokoll geschrieben, wenn eine Umwandlung verarbeitet wird.

Wiederverwendbare Umwandlungen

Wiederverwendbare Umwandlungen sind Umwandlungen, die Sie in mehreren Mappings oder Mapplets verwenden können.

Sie können beispielsweise eine Ausdrucksumwandlung erstellen, die die Mehrwertsteuer für Verkäufe in Kanada berechnet, um die Kosten für Geschäfte in diesem Land zu analysieren. Um zu vermeiden, dass Sie die dieselbe Arbeit jedes Mal von vorne machen, können Sie eine wiederverwendbare Umwandlung verwenden. Wenn Sie diese Umwandlung in ein Mapping integrieren müssen, fügen Sie eine Instanz der Umwandlung zum Mapping hinzu. Wenn Sie die Definition der Umwandlung ändern, werden die jeweiligen Änderungen auf alle Instanzen angewendet.

Das Developer Tool speichert jede wiederverwendbare Umwandlung als Metadaten, unabhängig von einem beliebigen Mapping oder Mapplet, das diese Umwandlung verwendet. Es speichert wiederverwendbare Umwandlungen in einem Projekt oder Ordner.

Beim Hinzufügen von Instanzen von wiederverwendbaren Umwandlungen zu Mappings kann das Mapping ungültig werden oder es können unerwartete Daten generiert werden, wenn Sie Änderungen an der Umwandlung vornehmen.

Wiederverwendbare Umwandlungsinstanzen und geerbte Änderungen

Beim Hinzufügen einer wiederverwendbare Umwandlung zu einem Mapping oder Mapplet fügen Sie eine Instanz der Umwandlung hinzu. Die Definition der Umwandlung ist außerhalb des Mappings oder Mapplets immer noch vorhanden, während eine Instanz der Umwandlung innerhalb des Mappings oder Mapplets erscheint.

Wenn Sie die Umwandlung ändern, werden diese Änderungen auf Instanzen der Umwandlung angewendet. Anstatt dieselbe Umwandlung in jedem Mapping, das diese Umwandlung verwendet, zu aktualisieren, können Sie die wiederverwendbare Umwandlung einmalig aktualisieren und alle Instanzen der Umwandlung übernehmen die Änderung. Instanzen vererben Änderungen an Ports, Ausdrücke, Eigenschaften und den Namen der Umwandlung.

Bearbeiten einer wiederverwendbaren Umwandlung

Beim Bearbeiten einer wiederverwendbaren Umwandlung erben alle Instanzen dieser Umwandlung die Änderungen. Einige Änderungen führen dazu, dass die Mappings, die diese wiederverwendbare Umwandlung verwenden, ungültig werden.

Sie können die Umwandlung im Editor öffnen, um eine wiederverwendbare Umwandlung zu bearbeiten. Sie können eine Instanz der Umwandlung in einem Mapping nicht bearbeiten. Sie können die Laufzeiteigenschaften der Umwandlung bearbeiten.

Wenn Sie an einer wiederverwendbaren Umwandlung jedoch die folgenden Änderungen vornehmen, sind die Mappings, die Instanzen dieser Umwandlung verwenden, möglicherweise nicht gültig.

- Wenn Sie einen oder mehrere Ports in einer Umwandlung löschen, trennen Sie die Instanz von Teilen oder vom gesamten Mapping-Datenfluss.
- Wenn Sie einen Portdatentyp ändern, können keine Daten mehr von diesem Port zu einem anderen Port zugeordnet werden, der einen inkompatiblen Datentyp verwendet.
- Wenn Sie einen Portnamen ändern, sind die Ausdrücke, die sich auf den Port beziehen, nicht mehr gültig.
- Wenn Sie einen ungültigen Ausdruck eingeben, der in der wiederverwendbaren Umwandlung nicht gültig ist, sind die Mappings, die diese Umwandlung verwenden, nicht mehr gültig. Der Data Integration Service kann Mappings nicht ausführen, die nicht gültig sind.

Editoransichten für eine wiederverwendbare Umwandlung

Sie definieren Eigenschaften und erstellen Java-Code für eine wiederverwendbare Java-Umwandlung in den Editoransichten.

Bei wiederverwendbaren Umwandlungen stehen die folgenden Ansichten zur Verfügung:

Übersicht

Geben Sie den Namen und die Beschreibung der Umwandlung ein, und erstellen und konfigurieren Sie Eingabe- und Ausgabeports.

Erweitert

Legen Sie erweiterte Eigenschaften für die Umwandlung fest.

Nicht wiederverwendbare Umwandlungen

Eine nicht wiederverwendbare Umwandlung ist eine Umwandlung, die Sie in einem bestimmten Mapping erstellen. Sie können die Umwandlung nicht in anderen Mappings verwenden.

Sie erstellen beispielsweise ein Mapping, das mehrere Umwandlungen enthält. Jede Umwandlung führt Berechnungen für die Quelldaten aus. Sie erstellen eine nicht wiederverwendbare Aggregatorumwandlung am Ende des Mappings, um die Ergebnisse zu verarbeiten. Wenn Sie eine nicht wiederverwendbare Umwandlung erstellen, können Sie die Ports aus einer Umwandlung in eine andere im Mapping ziehen und die Eingabeports erstellen.

Das Developer-Tool speichert die nicht wiederverwendbare Aggregatorumwandlung als Metadaten, die es mit dem Mapping beibehält.

Editor-Ansichten für eine nicht wiederverwendbare Umwandlung

Definieren Sie Eigenschaften für eine nicht wiederverwendbare Umwandlung in den Editor-Ansichten.

Die folgenden Ansichten werden für nicht wiederverwendbare Umwandlungen angezeigt:

Allgemein

Geben Sie den Namen und die Beschreibung der Umwandlung ein.

Ports

Erstellen und konfigurieren Sie Eingabe- und Ausgabeports.

Erweitert

Legen Sie erweiterte Eigenschaften für die Umwandlung fest.

Erstellen einer Umwandlung

Sie können eine wiederverwendbare Umwandlung zur Wiederverwendung in mehreren Mappings oder Mapplets erstellen. Sie können aber auch eine nicht wiederverwendbare Umwandlung erstellen, die nur ein Mal in einem Mapping oder Mapplet verwendet wird.

Klicken Sie zum Erstellen einer wiederverwendbaren Umwandlung, klicken Sie auf **Datei** > **Neu** > **Umwandlung** und schließen Sie den Assistenten ab.

Wählen Sie zum Erstellen einer nicht wiederverwendbaren Umwandlung in einem Mapping oder Mapplet eine Umwandlung aus der Umwandlungspalette aus und ziehen Sie die Umwandlung in den Editor.

Für bestimmte Umwandlungen müssen Sie einen Modus auswählen oder weitere Konfigurationsschritte ausführen, um die jeweilige Umwandlung zu erstellen. Wenn Sie beispielsweise eine Lookup-Umwandlung erstellen, müssen Sie ein Datenobjekt als Lookup-Quelle auswählen.

Nach Erstellung einer Umwandlung wird diese im Editor angezeigt. Einige Umwandlungen enthalten vordefinierte Ports und Gruppen. Andere Umwandlungen sind leer.

KAPITEL 2

Adressvalidator-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Übersicht über Adressvalidator-Umwandlung, 58
- Adressreferenzdaten, 59
- Portgruppen und Portauswahl, 61
- Vorlagen, 62
- Adressvalidator-Umwandlung - Eingabeportgruppen, 62
- Adressvalidator-Umwandlung - Ausgabeportgruppen, 63
- Ports mit mehreren Instanzen, 65
- Adressvalidierungsprojekte, 67
- Formatierte Adressen und Zustellungsstandards, 68
- Komplettieren unvollständiger Adressen , 69
- Adressvalidator-Statusports, 69
- Adressreferenzdaten-Dateistatus, 81
- Allgemeine Einstellungen der Adressvalidator-Umwandlung, 82
- Adressvalidator-Umwandlung - Erweiterte Eigenschaften, 83
- Zertifizierungsberichte, 99
- Konfigurieren einer Adressvalidator-Umwandlung, 102
- Hinzufügen von Ports zur Adressvalidator-Umwandlung, 103
- Erstellen von benutzerdefinierten Vorlagen, 103
- Definieren von Adressvalidator-Modellen, 104
- Definieren eines Zertifizierungsberichts, 104

Übersicht über Adressvalidator-Umwandlung

Die Adress-Validiererumwandlung ist eine Umwandlung mit mehreren Gruppen, die Eingabeadressdaten mit Adressreferenzdaten vergleicht. Sie bestimmt die Genauigkeit der Adressen und behebt Fehler in den

Adressen. Die Umwandlung erstellt Datensätze, die den Zustellungsstandards für den Inhalt und die Struktur der Daten entsprechen. Die Umwandlung fügt außerdem Statusinformationen für jede Adresse hinzu.

Die Adress-Validierungsumwandlung führt die folgenden Vorgänge auf Adressdaten aus:

- Die Umwandlung vergleicht die Adressdatensätze in den Quelldaten mit Adressdefinitionen in den Adressreferenzdaten.
- Sie generiert detaillierte Statusberichte anhand der Gültigkeit der einzelnen Eingabeadressen, deren Zustellbarkeitsstatus und der Natur aller enthaltenen Fehler oder Mehrdeutigkeiten.
- Sie behebt Fehler und vervollständigt Adressdatensätze. Zum Berichtigen einer Adresse muss bei der Umwandlung eine positive Übereinstimmung mit einer Adresse in den Referenzdaten gefunden werden. Die Umwandlung kopiert die erforderlichen Datenelemente aus den Adressreferenzdaten in die Adressdatensätze.
- Sie fügt Informationen hinzu, die nicht in der Standardadresse angezeigt werden, bei der Postzustellung jedoch hilfreich sind (beispielsweise Zustellpunkt-Informationen und Geokodierungsinformationen).
- Sie schreibt die Ausgabeadressen in dem Format, das das Datenprojekt und das Postbeförderungsunternehmen benötigen. Sie definieren das Format, wenn Sie bei der Umwandlung die Ausgabeports auswählen.

Adressreferenzdaten

Ein Adressreferenzdatensatz beschreibt die Adressen, die ein nationales Postbeförderungsunternehmen für jeden gültigen Briefkasten in einem Land erkennt. Bevor Sie die Adressvalidierung mit der Adress-Validiererumwandlung ausführen, installieren Sie die Adressreferenzdaten auf dem Informatica-Dienstecomputer in der Domäne. Sie können Adressenreferenzdaten von Informatica kaufen und herunterladen.

Sie installieren eine Adressreferenzdatendatei für jedes Land, auf das sich die Quelladressdaten beziehen. Ein Land mit einer hohen Bevölkerungszahl benötigt möglicherweise mehrere Dateien. Darüber hinaus können Sie Datendateien installieren, die die Quelladressdaten erweitern. Das Postbeförderungsunternehmen kann die Datenerweiterungen verwenden, um die Genauigkeit der Adressen zu bestätigen und die Zustellung zu beschleunigen.

Bei der Adressvalidierung vergleicht die Adress-Validiererumwandlung jeden Eingabedatendatensatz mit den Adressreferenzdaten. Wenn die Umwandlung die Eingabeadresse in den Adressreferenzdaten findet, aktualisiert die Umwandlung den Datensatz mit den richtigen und vollständigen Adressdaten. Wenn Sie zusätzliche Datensätze erworben haben, kann die Umwandlung auch die Adressdaten erweitern.

Verwenden Sie das Fenster **Einstellungen** im Developer-Tool, um Informationen über Adressreferenzdaten-Dateien auf dem Informatica-Dienstecomputer in der Domäne anzuzeigen.

Adressvalidierermodi und Adressreferenzdaten

Bei der Konfiguration der Adress-Validiererumwandlung wählen Sie den Typ der Adressvalidierung aus, der von der Umwandlung durchgeführt wird. Über den Validierungstyp wird festgelegt, ob die Umwandlung die Eingabeadresse mit den Adressreferenzdaten vergleicht. Mit dem Validierungstyp werden darüber hinaus die Adressreferenzdatentypen angegeben, die von der Umwandlung gelesen werden.

Die Adress-Validiererumwandlung kann die folgenden Adressreferenzdatentypen lesen:

Adresscode-Lookup

Installieren Sie Adresscode-Lookup-Daten, um eine unvollständige oder eine vollständige Adresse aus einem Codewert auf einem Eingabeport abzurufen. Die Vollständigkeit der Adresse richtet sich nach dem Grad der Adresscodeunterstützung in dem Land, zu dem die Adresse gehört. Wählen Sie zum Lesen des Adresscodes aus einer Eingabeadresse die länderspezifischen Ports aus der Portgruppe „Diskret“ aus.

Sie können Ports für die folgenden Länder auswählen:

- Deutschland. Gibt eine Adresse auf Orts-, Gemeinde- oder Straßenebene zurück.
- Japan. Gibt eine Adresse auf Ebene des eindeutigen Briefkastens zurück.
- Südafrika. Gibt eine Adresse auf Straßenebene zurück.
- Serbien. Gibt eine Adresse auf Straßenebene zurück.
- Vereinigtes Königreich. Gibt eine Adresse auf Ebene des eindeutigen Briefkastens zurück.

Die Adress-Validiererumwandlung liest Adresscode-Lookup-Daten, wenn Sie die Umwandlung zur Ausführung im Adresscode-Lookup-Modus konfigurieren.

Batch-Daten

Installieren Sie Batchdaten, um Adressvalidierung für eine Gruppe von Adressdatensätzen durchzuführen. Verwenden Sie Batch-Daten, um sicherzustellen, dass die Eingabeadressen auf der Grundlage der aktuellen Postdaten des nationalen Postbeförderungsunternehmen voll zustellfähig und vollständig sind.

Die Adress-Validierungsumwandlung liest Batch-Daten, wenn Sie die Umwandlung zur Ausführung im Batchmodus konfigurieren.

CAMEO-Daten

Installieren Sie CAMEO-Daten, um die Kundensegmentierungsdaten zu Datensätzen mit Wohnanschriften hinzuzufügen. Kundensegmentierungsdaten zeigen die wahrscheinlichen Einkommensebenen und Lifestyle-Präferenzen der Bewohner für alle Adressen an.

Die Adress-Validierungsumwandlung liest CAMEO-Daten, wenn Sie die Umwandlung zur Ausführung im Batchmodus bzw. zertifizierten Modus konfigurieren.

Zertifizierte Daten

Installieren Sie zertifizierte Daten, um sicherzustellen, dass die Adressdatensätze den Zertifizierungsstandards entsprechen, die ein Postbeförderungsunternehmen definiert. Eine Adresse entspricht einem Zertifizierungsstandard, wenn sie Datenelemente enthält, die einen eindeutigen Briefkasten wie beispielsweise Zustellpunktdatenelemente identifizieren können. Wenn eine Adresse einem Zertifizierungsstandard entspricht, berechnet das Postbeförderungsunternehmen eine reduzierte Versandrate.

Die folgenden Länder definieren Zertifizierungsstandards:

- Australien. Zertifiziert die Post gemäß dem Standard des Address Matching Approval System (AMAS).
- Kanada. Zertifiziert die Post gemäß dem Standard des Software Evaluation And Recognition Program (SERP).
- Frankreich. Zertifiziert die Post gemäß dem Standard des National Address Management Service (SNA).
- Neuseeland. Zertifiziert die Post gemäß dem SendRight-Standard.
- Vereinigte Staaten von Amerika. Zertifiziert die Post gemäß dem Standard des Coding Accuracy Support System (CASS).

Die Adress-Validierungsumwandlung liest die Batch-Daten, wenn Sie die Umwandlung zur Ausführung im zertifizierten Modus konfigurieren.

Geokodierungsdaten

Installieren Sie die Geokodierungsdaten, um die Geocodes zu Adressdatensätzen hinzuzufügen. Geocodes sind Breiten- und Längengradkoordinaten.

Die Adress-Validierungsumwandlung liest Geokodierungsdaten, wenn Sie die Umwandlung zur Ausführung im Batchmodus oder zertifizierten Modus konfigurieren.

Hinweis: Informatica stellt verschiedene Arten von Geokodierungsdaten bereit. Wenn Sie Ankunftspunkt- oder Parcel Centroid-Geocodes für Adressen benötigen, müssen Sie zusätzliche Geokodierungsdatensätze erwerben.

Interaktive Daten

Installieren Sie interaktive Daten zum Auffinden der vollständigen gültigen Adresse, wenn eine Eingabeadresse unvollständig ist oder wenn Sie hinsichtlich der Gültigkeit der Eingabeadresse unsicher sind.

Die Adress-Validiererumwandlung liest interaktive Daten, wenn Sie die Umwandlung zur Ausführung im interaktiven Modus konfigurieren.

Vorschlagslistendaten

Installieren Sie Vorschlagslistendaten, um nach alternativen gültigen Versionen eines unvollständigen Adressdatensatzes zu suchen. Verwenden Sie beim Konfigurieren eines Adressvalidierungs-Mappings Vorschlagslistendaten, um Adressdatensätze nacheinander in Echtzeit zu verarbeiten. Die Adress-Validiererumwandlung verwendet die Datenelemente in der unvollständigen Adresse, um eine doppelte Überprüfung der Vorschlagslistendaten durchzuführen. Die Umwandlung gibt jede gültige Adresse zurück, die die Informationen in der unvollständigen Adresse enthält.

Die Adress-Validierungsumwandlung liest Vorschlagslistendaten, wenn Sie die Umwandlung so konfigurieren, dass sie im Vorschlagslistenmodus ausgeführt wird.

Ergänzende Daten

Installieren Sie zusätzliche Daten, um Daten zu einem Adressdatensatz hinzuzufügen, die das Postbeförderungsunternehmen bei der Zustellung unterstützen können. Verwenden Sie die zusätzlichen Daten, um Details über den geografischen oder Postbereich hinzuzufügen, der die Adresse enthält. In bestimmten Ländern können zusätzliche Daten eine eindeutige Kennung für einen Briefkasten innerhalb des Postsystems bereitstellen.

Portgruppen und Portauswahl

Die Adress-Validierungsumwandlung enthält vordefinierte Portgruppen, die die zu verwendenden Eingabe- und Ausgabeports enthalten. Durchsuchen Sie beim Konfigurieren einer Adress-Validierungsumwandlung die Gruppen und wählen Sie die benötigten Ports aus.

Wählen Sie Eingabeports aus, die der Struktur der Adresseingabedaten entsprechen. Wählen Sie Ausgabeports mit Adressdaten aus, die vom Projekt benötigt werden.

Sie können Eingabe- und Ausgabeports direkt zur Umwandlung hinzufügen oder ein Standardmodell erstellen, das Eingabe- und Ausgabeports enthält. Wenn Sie Ports direkt zur Umwandlung hinzufügen, gelten die ausgewählten Ports nur für diese Umwandlung. Wenn Sie Ports zum Standardmodell hinzufügen, gelten die ausgewählten Ports für künftig zu erstellende Adress-Validierungsumwandlungen.

Hinweis: Das Standardmodell ist kein Repository-Objekt. Das Standardmodell befindet sich auf dem Computer, auf dem es erstellt wurde.

Sie können Pass-Through-Ports auch zur Umwandlung für Spalten hinzufügen, die Sie bei der Adress-Validierungsumwandlung nicht verarbeiten möchten.

Vorlagen

Sie können eine oder mehrere Portvorlagen in der Umwandlung erstellen. Eine Vorlage ist eine Teilmenge von Ports aus einer oder mehreren Portgruppen. Verwenden Sie Vorlagen zum Verwalten von Ports, die erwartungsgemäß häufig in einem Projekt verwendet werden.

Die von Ihnen erstellten Vorlagen werden in allen Adress-Validierungsumwandlungen im Modellrepository angezeigt.

Adressvalidator-Umwandlung - Eingabeportgruppen

Bevor Sie Adressdaten mit Eingabeports während der Umwandlung verbinden können, durchsuchen Sie die Eingabegruppen und wählen Sie die Ports aus, die der Struktur und dem Inhalt der Eingabedaten entsprechen. Durchsuchen Sie die Ausgabegruppen und wählen Sie die Ports aus, die Ihren Datenanforderungen entsprechen.

Die Adressvalidator-Umwandlung zeigt die Portgruppen in einem Basis- und einem erweiterten Modell an. Sie können die meisten Adressen mithilfe einer Portgruppe im Basismodell definieren. Verwenden Sie bei hoch komplexen Adressen die im erweiterten Modell verfügbaren zusätzlichen Ports.

Hinweis: Wählen Sie Ports nur aus einer Eingabeportgruppe aus.

Die Umwandlung weist die folgenden Eingabeportgruppen auf:

Speziell

Liest Datenspalten, die vollständige Informationen über ein einzelnes Datenelement enthalten, beispielsweise Gebäudenummer, Straßenname oder Postleitzahl. Die Gruppe „Speziell“ ist im Basismodell und im erweiterten Modell verfügbar.

Hybrid

Liest Datenspalten, die ein oder mehr Datenelemente enthalten. Die Gruppe „Hybrid“ kombiniert Ports aus den Gruppen „Speziell“ und „Mehrzeilig“. Verwenden Sie Hybrid-Ports, um Adressdatensätze zu erstellen, die Sie an ein Postbeförderungsunternehmen übermitteln können. Hybrid-Ports strukturieren eine Adresse nach den Standards von Postbeförderungsunternehmen und kennzeichnen die Art der Daten in jeder Zeile. Die Gruppe „Hybrid“ ist im Basismodell und im erweiterten Modell verfügbar.

Mehrzeilig

Liest Datenspalten, die mehrere Datenelemente enthalten. Jeder Eingabestring entspricht einer Zeile einer Adresse, die in dem Format strukturiert ist, das vom Postbeförderungsunternehmen verlangt wird. Verbinden Sie Adressspalten mit „Mehrzeilig“-Ports, um eine druckbare Gruppe von Adressdatensätzen zu erstellen.

Jeder „Mehrzeilig“-Port stellt eine Zeile in der gedruckten Adresse dar, beispielsweise die folgende Zeile einer Straßenadresse:

```
"123 Main Street Apartment 2"
```

„Mehrzeilig“-Ports geben nicht die Art der Daten an, die in jeder Adresszeile erscheinen. Die Gruppe „Mehrzeilig“ ist im Basismodell und im erweiterten Modell verfügbar.

Adressvalidator-Umwandlung - Ausgabeportgruppen

Vor dem Verbinden von Adressvalidator-Umwandlungen mit anderen Umwandlungen oder Datenobjekten bestimmen Sie die benötigten Informationstypen sowie die Struktur der Ausgabeadressen.

Durchsuchen Sie die Ausgabegruppen und wählen Sie die Ports aus, die Ihren Datenanforderungen entsprechen.

Hinweis: Sie können Ports mit gemeinsamen Funktionen und Ports aus mehreren Ausgabegruppen auswählen.

Die Umwandlung verfügt über die folgenden vordefinierten Ausgabegruppen:

Adresselemente

Schreibt Elemente der Hausadresse wie Hausnummer, Wohnungsnummer und Straßenname in verschiedene Ports. Suchen Sie die Gruppe „Adresselemente“ im Basismodell und im erweiterten Modell.

Ergänzend für AT

Schreibt Daten in österreichische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Postleitzahlendaten auf Gebäudeebene. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für AT“.

Spezifisch für Australien

Schreibt Daten in australische Adressen, damit die Adressen den Standards des AMAS (Address Matching Approval System) der australischen Post entsprechen. Suchen Sie die Gruppe „Spezifisch für Australien“ im Basismodell und im erweiterten Modell.

Ergänzend für BE

Schreibt Daten zu belgischen Adressen, die bei der Postzustellung hilfreich sein können. Die Daten enthalten Ortschafts- und Gebiets-ID-Codes von der Statistikabteilung in Belgien. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für BE“.

Ergänzend für BR

Schreibt Daten in brasilianische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Bezirkskenndaten des IBGE (Brasilianisches Institut für Geographie und Statistik). Suchen Sie im Basismodell nach der Gruppe „Ergänzend für BR“.

CAMEO

Generiert demografische und Einkommensübersichtsdaten, die in Kundensegmentierungsanalysen genutzt werden können. Suchen Sie die Gruppe CAMEO im Basismodell.

Spezifisch für Kanada

Schreibt Daten in kanadische Adressen, damit die Adressen den Standards des SER-Programms (Software Evaluation and Recognition Program) der kanadischen Post entsprechen. Suchen Sie die Gruppe „Spezifisch für Kanada“ im Basismodell.

Ergänzend für CH

Schreibt Daten in schweizer Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. erweiterte Postleitzahlendaten. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für CH“.

Kontaktelemente

Schreibt Personen- oder Kontaktdaten wie Namen, Anrede und Titel. Suchen Sie die Gruppe „Kontaktelemente“ im erweiterten Modell.

Land

Schreibt den von der International Organization for Standardization (ISO) definierten Ländernamen oder Ländercode. Suchen Sie die Gruppe „Land“ im Basismodell und im erweiterten Modell.

Ergänzend für DE

Schreibt Daten in deutsche Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Gemeinde- und Kreiscodedaten. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für DE“.

Formatierte Adresszeile

Schreibt Adressen, die druck- und postfertig formatiert sind. Suchen Sie die Gruppe „Formatierte Adresszeile“ im Basismodell und im erweiterten Modell.

Ergänzend für FR

Schreibt Daten in französische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Identifikationscodes des INSEE (Nationales Institut für Statistik und Wirtschaftsstudien). Suchen Sie im Basismodell nach der Gruppe „Ergänzend für FR“.

Spezifisch für Frankreich

Schreibt Daten in französische Adressen, damit die Adressen den Standards des SNA (Nationaler Adressverwaltungsdienst) der französischen Post entsprechen. Suchen Sie die Gruppe „Spezifisch für Frankreich“ im Basismodell.

Geokodierung

Generiert Geokodierungsdaten wie Breiten- und Längengradkoordinaten für eine Adresse. Suchen Sie die Gruppe „Geokodierung“ im Basismodell.

ID-Elemente

Schreibt Datensatz-IDs und Transaktionsschlüsseldaten. Suchen Sie die Gruppe „ID-Elemente“ im erweiterten Modell.

Ergänzend für JP

Schreibt Daten in japanische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Choumei Aza-Codes. Suchen Sie die Gruppe „Ergänzend für JP“ im Basismodell.

Ergänzend für KR

Schreibt Daten in südkoreanische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. eindeutige Bezeichner, die aktuelle und veraltete Versionen einer bestimmten Adresse angeben können. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für KR“.

Letzte Zeilenelemente

Schreibt Daten, die in der letzten Zeile einer inländischen Adresse enthalten sein können: Suchen Sie die Gruppe „Letzte Zeilenelemente“ im Basismodell und im erweiterten Modell.

Spezifisch für Neuseeland

Schreibt Daten in australische Adressen, damit die Adressen den SendRight-Standards der neuseeländischen Post entsprechen. Suchen Sie die Gruppe „Spezifisch für Neuseeland“ im Basismodell.

Ergänzend für PL

Schreibt Daten in polnische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. TERYT-Daten (territoriale Aufteilung). Suchen Sie im Basismodell nach der Gruppe „Ergänzend für PL“.

Rest

Schreibt Datenelemente, die von der Umwandlung nicht in andere Ports geparst werden können. Suchen Sie die Gruppe „Rest“ im Basismodell und im erweiterten Modell.

Ergänzend für RS

Schreibt Daten in serbische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Suffixdaten für Postleitzahlen. Suchen Sie die Gruppe „Ergänzend für RS“ im Basismodell.

Ergänzend für RU

Schreibt Daten zu russischen Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. den FIAS-Bezeichner (Federal Information Addressing System) für die Adresse. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für RU“.

Statusinformation

Generiert detaillierte Daten über die Qualität jeder Eingabe- und Ausgabeadresse. Suchen Sie die Gruppe „Statusinformation“ im Basismodell.

Ergänzend für UK

Schreibt Daten zu britischen Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. den Zustellpunkt und Daten der Vermessungsbehörde. Suchen Sie die Gruppe „Ergänzend für UK“ im Basismodell.

Spezifisch für USA

Schreibt Daten in australische Adressen, damit die Adressen den CASS-Standards (Coding Accuracy Support System) des US-amerikanischen Postdiensts entsprechen. Suchen Sie die Gruppe „Spezifisch für USA“ im Basismodell.

Ergänzend für USA

Schreibt geografische und demografische Daten, wie beispielsweise FIPS-Codes (Federal Information Processing Standard) für US-amerikanische Adressen. Suchen Sie die Gruppe „Ergänzend für USA“ im Basismodell.

XML

Schreibt die Daten des Adressdatensatzes in eine XML-Struktur, die von der Softwarebibliothek zur Adressüberprüfung definiert wird. Suchen Sie die XML-Gruppe im erweiterten Modell.

Ergänzend für ZA

Schreibt Daten in südafrikanische Adressen, die bei der Postzustellung hilfreich sein können, wie z. B. Daten der nationalen Adressdatenbank. Suchen Sie im Basismodell nach der Gruppe „Ergänzend für ZA“.

Ports mit mehreren Instanzen

Viele Arten von Adressdaten können mehrmals in einer Adresse vorkommen. Sie können mehrere Instanzen eines Ports auswählen, wenn die Adresse mehrere Vorkommen eines Datenelements enthält.

Ein Port mit mehreren Instanzen kann bis zu sechs Instanzen enthalten. Viele Adressen verwenden eine Portinstanz für jedes enthaltene Datenelement. Bestimmte Adressen verwenden eine zweite Portinstanz. Ein kleine Gruppe von Adressen verwendet mehr als eine Portinstanz.

Oft ist die erste Instanz eines Ports der primäre Name oder der größte Bereich, den der Port angibt. Sie müssen die Beziehungen zwischen den Portinstanzen für alle ausgewählten Ports überprüfen.

"Straße vollständig"-Port, Beispiel

Ein Adressdatensatz für Großbritannien kann zwei Straßennamen enthalten, wobei eine Straße Teil eines größeren Straßenplans ist.

Die folgende Tabelle enthält eine Adresse, die von zwei „Straße vollständig“-Ports verwendet wird:

Port	Daten
Hausnummer vollständig 1	1A
Straße vollständig 1	THE PHYGTLE
Straße vollständig 2	SOUTH STREET
Ortsname 1	NORFOLK
Postleitzahl 1	NR25 7QE

In diesem Beispiel richten sich die Straßendaten in „Straße vollständig 1“ nach den Straßendaten in „Straße vollständig 2“. Die Daten in "Straßennummer vollständig 1" verweisen auf die Daten in "Straße vollständig 1".

Hinweis: Obwohl „Straße vollständig 1“ den Standort des Briefkastens angibt, kann es sich bei „Straße vollständig 2“ um die größere Straße handeln.

Kontaktport, Beispiel

Ein Adressdatensatz kann mehrere Kontakte enthalten, wenn jeder Kontakt Mitglied desselben Haushalts ist.

Die folgende Tabelle enthält eine Adresse, die zwei „Kontakt Name“-Ports verwendet:

Port	Daten
Kontakt Name 1	MR. JOHN DOE
Kontakt Name 2	MS. JANE DOE
Formatierte Adresszeile 1	2 MCGRATH PLACE EAST
Formatierte Adresszeile 2	ST. JOHN'S NL A1B 3V4
Formatierte Adresszeile 3	KANADA

In diesem Beispiel kann das Unternehmen entscheiden, ob „Kontakt Name 1“ oder „Kontakt Name 2“ angewendet werden soll. Bei der Adressvalidator-Umwandlung wird den Kontaktdaten keine Priorität eingeräumt.

Wenn Sie Adressen zum Drucken formatieren, sollten Sie mehrere Instanzen der „Formatierte Adresszeile“-Ports verwenden. Sie können bis zu 12 Ports "Formatierte Adresszeile" auswählen.

Adressvalidierungsprojekte

Sie können die Adressvalidator-Umwandlung in zahlreichen Projekttypen verwenden. Sie erstellen eine Adressvorlage mit verschiedenen Ports für jeden Projekttyp.

Sie können ein Adressvalidierungs-Projekt mit mindestens einem der folgenden Ziele definieren:

Erstellen formatierter Adressen, die den Zustellungsstandards entsprechen

Sie können einen großen Adressdatensatz für eine Postkampagne vorbereiten. Wenn Sie die Adressen in dem vom Zusteller bevorzugten Format erstellen, reduzieren sich die Portokosten drastisch. Wenn Sie Adressen für den Versand vorbereiten, wählen Sie Ausgabeports aus, die jede Zeile der formatierten Adresse auf einen einzelnen Port schreiben. Sie können verschiedene Ports für den Kontaktnamen sowie für die Zeilen der Straßenadresse, des Orts und der Postleitzahl auswählen.

Adressen nach Einkommens- und Lifestyle-Indikatoren ordnen

Sie können Kundensegmentierungsdaten zum Datensatz mit Wohnanschriften hinzufügen. Kundensegmentierungsdaten zeigen die wahrscheinlichen Einkommensebenen und Lifestyle-Präferenzen der Bewohner für alle Adressen an. Wählen Sie die Ports aus der CAMEO-Ausgabegruppe aus, um Kundensegmentierungsdaten zu den Adressdatensätzen hinzuzufügen. Sie können Kundensegmentierungsdaten in Marketingkampagnen per E-Mail verwenden und so mehrere Absatzmärkte erreichen.

Erstellen von Adressen, die vom Zusteller zertifiziert sind

Wenn Sie einen Datensatz für die Australia POST, Kanada POST oder den USPS (United States Postal Service) vorbereiten, fügen Sie Daten hinzu, die die Zustellbarkeit jeder Adresse bestätigen.

Die Adressvalidator-Umwandlung kann Berichte erzeugen, die sicherstellen, dass die Adressdatensätze vollständig und genau mit den Datenstandards aller Zusteller übereinstimmen.

Erstellen von Adressen, die den gesetzlichen Anforderungen entsprechen

Sie können überprüfen, ob die Adressdatensätze Ihres Unternehmens mit den brancheninternen oder staatlichen Regelungen übereinstimmen. Wählen Sie Ausgabeports aus, die jedes Adressdatenelement in ein separates Feld schreiben. Wählen Sie auch die Adressvalidator-Statusports aus, die detaillierte Informationen zur Genauigkeit und Vollständigkeit der Ausgabedaten bereitstellen.

Unvollständige Adressen komplettieren

Sie können eine unvollständige Adresse eingeben und die gültigen vollständigen Adressen abrufen, die mit der unvollständigen Adresse in den Referenzdaten übereinstimmen. Konfigurieren Sie die Umwandlung zur Ausführung im Vorschlagslisten- oder interaktiven Modus, um unvollständige Adressen zu komplettieren. Sie können die Eingabeadresse als einzelne Zeile auf dem Port „Vollständige Adresse“ eingeben.

Verbessern der Datenqualität von Adressen

In Übereinstimmung mit anderen Datenprojekten können Sie die Struktur und allgemeine Datenqualität des Adressdatensatzes verbessern. Der Datensatz enthält beispielsweise mehr Spalten als benötigt oder denselben Datentyp in mehreren Spalten. Sie können die Anzahl der Spalten im Datensatz verringern und die Spalten vereinfachen, die für verschiedene Arten von Daten verwendet werden.

Formatierte Adressen und Zustellungsstandards

Beim Vorbereiten von Adressdatensätzen für eine Postkampagne erstellen Sie eine druckbare Adressstruktur, die mit den Formatierungsstandards des Zustellers übereinstimmen.

Der USPS beispielsweise verwaltet folgendes Adressformat für US-Adressen:

Line 1	**Person/Contact Data**	**JOHN DOE**
Line 2	Street Number, Street, Sub-Building	123 MAIN ST NW STE 12
Line 3	Locality, State, ZIP Code	ANYTOWN NY 12345

Sie können ein druckbares Adressformat festlegen, dass jede Adresszeile auf einen einzelnen Port schreibt. Sie können Ports verwenden, die die Datentypen in jeder Zeile erkennen, oder Ports, die die Datenstruktur ungeachtet der Daten in jeder Zeile ausfüllen.

In der folgenden Tabelle werden die verschiedenen Möglichkeiten zum Formatieren einer druckbaren US-Adresse angezeigt:

Adresse	Zu verwendende Ports	Alternativ zu verwendende Ports
JOHN DOE	Empfänger Zeile 1	Formatierte Adresszeile 1
123 MAIN ST NW STE 12	Lieferadresszeile 1	Formatierte Adresszeile 2
ANYTOWN NY 12345	Länderspezifische letzte Zeile 1	Formatierte Adresszeile 3

Verwenden Sie "Formatierte Adresszeile"-Ports, wenn der Datensatz verschiedene Adresstypen enthält, wie Geschäfts- oder Privatadressen. Eine Geschäftsadresse benötigt unter Umständen drei Adresszeilen für Kontakt- und Unternehmensdaten. Mit der Adressvalidator-Umwandlung wird sichergestellt, dass jede Geschäfts- oder Privatadresse korrekt formatiert wird, indem "Formatierte Adresszeile"-Ports nur dann verwendet werden, wenn Sie benötigt werden. In "Formatierte Adresszeile"-Ports werden jedoch nicht die Datentypen angegeben, die sie enthalten.

Verwenden Sie die Ports "Empfänger Zeile", "Lieferadresszeile" und "Länderspezifisch letzte Zeile", wenn alle Adressen im selben Format vorliegen. Die Ports "Empfänger Zeile", "Lieferadresszeile" und "Länderspezifisch letzte Zeile" trennen die Adressdatenelemente nach Informationstyp und machen den Datensatz leicht verständlich.

Hinweis: Sie können andere Ports auswählen, um diese Adresse zu verarbeiten. Dieses Beispiel befasst sich mit Ports, die die Adressen für den Druck und die Lieferung formatieren.

Demografische und geografische Daten

Beim Erstellen eines Datensatzes für eine Postkampagne können Sie zahlreiche Datentypen hinzufügen, die andernfalls nicht in der Adresse angezeigt werden. Verwenden Sie diese Daten zum Überprüfen der demografischen und geografischen Verteilung der Post.

Sie können beispielsweise den Kongressbezirk angeben, dem eine US-Adresse angehört. Sie können auch Koordinaten für Längen- und Breitengrade erzeugen, wenn das Zielland die Koordinaten in den Referenzdaten seinen Postsystems aufweist.

Komplettieren unvollständiger Adressen

Bei Auswahl des Vorschlagslisten- oder interaktiven Modus können Sie eine unvollständige Adresse eingeben und alle passenden vollständigen Adressen aus den Referenzdaten abrufen.

Wählen Sie den Vorschlagslistenmodus aus, wenn Sie sich bei der Adresse unsicher sind und eine Liste mit gültigen Kandidaten anzeigen möchten. Wählen Sie den interaktiven Modus aus, wenn Sie sich bei der Adresse sicher sind und die vollständige Form überprüfen möchten. In jedem Fall sucht die Adress-Validiererumwandlung nach Adressreferenzdaten und gibt alle Adressen zurück, die die Eingabedaten enthalten.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie die Umwandlung zur Ausführung im Vorschlagslisten- oder interaktiven Modus konfigurieren:

- Sie können eine Eingabeadresse auf mehreren Ports definieren oder alle Adresselemente auf dem Eingabeport „Vollständige Adresse“ eingeben.
- Wenn Sie die Umwandlung im Vorschlagslistenmodus konfigurieren, wählen Sie Ports aus der Eingabegruppe „Speziell“ aus. Alternativ können Sie den Port „Vollständige Adresse“ und optional den Port „Ländername“ aus der Gruppe „Mehrzeilig“ auswählen.
- Der Vorschlagslisten- und der interaktive Modus können mehrere Datensätze für jede Eingabeadresse zurückgeben. In der Eigenschaft „Maximale Ergebniszahl“ wird für die Anzahl der zurückgegebenen Adressen eine Obergrenze festgelegt. Übersteigt die Anzahl der übereinstimmenden Adressen den Wert „Maximale Ergebniszahl“ gibt der Port „Zählwertüberlauf“ die Anzahl der zusätzlichen Adressen zurück.
- Die Informatica-Adressüberprüfung verweist auf den Vorschlagslistenmodus als schnellen Fertigstellungsmodus.

Adressvalidator-Statusports

Die Adressvalidator-Umwandlung schreibt Statusinformationen für die Adresselemente, die von ihr gelesen werden, und schreibt auf Eingabe- und Ausgabeports. Verwenden Sie die „Statusinfo“-Ports zum Anzeigen der Statusinformationen.

Sie können folgende Statusports auswählen.

Adressauflösungscode

Beschreibt die nicht gültigen Adresselemente in der Adresse. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Adresstyp

Gibt den Adresstyp für die Fälle an, in denen das Postbeförderungsunternehmen mehr als eine Adressform erkennt. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Status der Elementeingabe

Beschreibt den Ähnlichkeitsgrad zwischen Eingabeadresselementen und Referenzdaten. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Elementrelevanz

Identifiziert die Adresselemente, die das Postbeförderungsunternehmen benötigt, um einen Briefkasten für die Adresse zu identifizieren. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Elementergebnisstatus

Beschreibt ein Update, das die Adressvalidierung an der Eingabeadresse vornimmt. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Erweiterter Elementergebnisstatus

Zeigt das Vorhandensein zusätzlicher Informationen für eine Adresse in den Referenzdaten an. Der Port kann Detailinformationen über die Updates enthalten, die die Adressvalidierung an einer Adresse vornimmt. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Geokodierungsstatus

Beschreibt den Typ der Geokodierungsdaten, die die Adressvalidierung für eine Adresse zurückgibt. Wählen Sie einen Port aus der Portgruppe „Geokodierung“ im Basismodell.

Zustellbarkeits-Score

Gibt die gesamte Zustellfähigkeit einer Eingabeadresse an. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Match-Code

Beschreibt die Ergebnisse der Adressvalidierung für eine Eingabeadresse. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Ergebnisprozentsatz

Stellt den Ähnlichkeitsgrad zwischen einer Eingabeadresse und der entsprechenden Ausgabeadresse als Prozentsatz dar. Wählen Sie den Port aus der Portgruppe „Statusinfo“ im Basismodell aus.

Elementstatuscode-Definitionen

Die Ports „Elementeingabestatus“, „Elementrelevanz“, „Elementergebnisstatus“ und „Erweiterter Elementergebnisstatus“ bieten Statusinformationen zur Gültigkeit von Eingabe- und Ausgabedatenelementen. Wählen Sie die entsprechenden Elementports, um die Ergebnisse einer Adressüberprüfung zu überprüfen.

Die Codes enthalten folgende Informationen:

- Die „Elementeingabestatus“-Codes geben die Qualität des Matches zwischen den Eingabeadressdaten und den Referenzdaten an.
- Die „Elementrelevanz“-Codes geben die Adresselemente an, die für die Zustellung an eine Adresse im Zielland benötigt werden.
- Die „Elementergebnisstatus“-Codes beschreiben Veränderungen, die an den Eingabedaten während der Verarbeitung vorgenommen wurden.
- Die „Erweiterter Elementergebnisstatus“-Codes geben an, dass die Adressreferenzdaten zusätzliche Informationen zum Adresselement enthalten.

Jeder Port gibt einen Code aus 20 Zeichen zurück, in dem jedes Zeichen auf ein anderes Adressdatenelement verweist. Wenn Sie die Ausgabecodes auf Elementports lesen, müssen Sie das Element kennen, auf das sich jedes Zeichen bezieht. Die 20 Zeichen bestehen aus zehn Paaren. Die beiden Codes in jedem Paar stellen einen Adressinformationstyp dar. Beispielsweise stellt die erste Position im Rückgabecode grundlegende Informationen zur Postleitzahl dar.

Hinweis: Der „Adressauflösungscode“-Port gibt eine Zeichenkette aus 20 Zeichen zurück basierend auf denselben Adresselementen wie die „Elementstatus“-Ports.

Die folgende Tabelle beschreibt die Adresselemente, auf die sich die Werte an jeder Position beziehen:

Position	Adresselement	Beschreibung	Beispiel für ein Adresselement
1	Ebenengenauigkeit für Postleitzahlen 0	Basis-Postleitzahlinformationen, beispielsweise eine fünfstellige Postleitzahl.	Die fünfstellige Postleitzahl 10118
2	Ebenengenauigkeit für Postleitzahlen 1	Zusätzliche Postleitzahlinformationen, beispielsweise die letzten vier Stellen eines ZIP+4-Codes.	0110, im ZIP+4-Code 10118-0110
3	Ort - Stufe 0	Hauptort, beispielsweise eine Stadt oder eine Gemeinde.	London, in England
4	Ort - Stufe 1	Abhängige Ortschaft, Vorstadt, Dorf.	Islington, in London
5	Provinz - Stufe 0	Hauptregion in einem Land, beispielsweise Name eines Bundesstaats in den USA, einer Provinz in Kanada, eines Kantons in der Schweiz.	Bundesstaat New York
6	Provinz - Stufe 1	County-Name in den USA.	Queens County, im Bundesstaat New York
7	Straße - Stufe 0	Informationen über Hauptstraßen.	South Great George's Street
8	Straße - Stufe 1	Informationen über abhängige Straßen.	George's Arcade, auf der South Great George's Street
9	Nummer - Stufe 0	Gebäude oder Hausnummer in Verbindung mit der Hauptstraße.	460, auf South Great George's Street
10	Nummer - Stufe 1	Gebäude oder Hausnummer in Verbindung mit der abhängigen Straße.	81, auf George's Arcade
11	Zustelldienst - Stufe 0	Postfach oder Postfachbeschreibung und Nummer.	Postfach 111
12	Zustelldienst - Stufe 1	Code des Postamts, das für die Zustellung zuständig ist.	MAIN STN
13	Gebäude - Stufe 0	Bezeichnung oder Nummer des Gebäudes. Identifiziert keine Hausnummer.	Alice Tully Hall
14	Gebäude - Stufe 1	Zusätzliche Bezeichnung oder Nummer des Gebäudes.	Starr Theater, in Alice Tully Hall

Position	Adresselement	Beschreibung	Beispiel für ein Adresselement
15	Gebäudeteil-Ebene 0	Bezeichnung oder Nummer für Wohnung, Suite oder Stockwerk	80, in 350 5th Avenue, Floor 80
16	Gebäudeteil-Ebene 1	Informationen über Wohnung, Suite oder Stockwerk, wenn sie den Gebäudeteil-Ebene-0-Informationen zugeordnet sind.	80-18, wobei 18 die Nummer der Suite und 80 die Stockwerknummer ist
17	Unternehmen - Stufe 0	Firmenname.	Address Doctor GmbH
18	Unternehmen - Stufe 1	Zusätzliche Unternehmensinformationen, beispielsweise eine Muttergesellschaft.	Informatica Corporation
19	Land - Stufe 0	Name des Landes.	Vereinigte Staaten von Amerika.
20	Land - Stufe 1	Territorium.	Jungferninseln der USA

Wenn ein Portname ein Nummernsuffix hat, bezieht sich Ebene 0 auf Daten im Port Nummer 1 und Ebene 1 auf Daten in den Ports 2 bis 6.

Informationen der Ebene 0 können in einer gedruckten Adresse vor oder nach den Informationen auf Ebene 1 kommen. Beispielsweise folgt Postleitzahl Ebene 1 auf Postleitzahl Ebene 0 und Ort - Stufe 1 kommt vor Ort - Stufe 0.

Ausgabeportwerte für Adressauflösungscode

Der Wert des Adressauflösungscodes ist eine Zeichenfolge aus 20 Zeichen, bei der jedes Zeichen ein anderes Eingabeadresselement darstellt. Der Wert für ein Zeichen beschreibt ein ungültiges Adresselement an der entsprechenden Stelle in der Adresse.

In der folgenden Tabelle werden die Adressauflösungscode-Portwerte beschrieben:

Code	Beschreibung
2	Das Adresselement ist für die Zustellung erforderlich, aber in der Eingabeadresse nicht vorhanden. Die Adressreferenzdaten enthalten das fehlende Adresselement. Eine Ausgabe von 2 zeigt an, dass die Adresse für die Zustellung ohne Adresselement nicht gültig ist.
3	Das Adresselement ist eine Haus- oder Straßennummer, die sich außerhalb des Bereichs für die Adresse befindet. Beispiel: Das Adresselement enthält eine Hausnummer, die in der angegebenen Straße nicht existiert. Der Modus der Vorschlagsliste gibt alternative Adressen zurück.
4	Die Adressvalidierung kann das Adresselement nicht überprüfen oder korrigieren, weil die Eingabeadresse mehr als eine Instanz des Elements enthält.

Code	Beschreibung
5	Das Adresselement ist in der aktuellen Adresse mehrdeutig und die Adressreferenzdaten enthalten Alternativen. Die Adressvalidierung kopiert das Eingabeelement in die Ausgabeadresse. Beispiel: Das Adresselement ist eine gültige Postleitzahl, die nicht mit einem gültigen Ort in der Adresse übereinstimmt.
6	Das Adresselement widerspricht einem anderen Element in der Adresse. Die Adressvalidierung kann das korrekte Element für die Adresse nicht ermitteln. Die Ausgabeadresse kopiert die Eingabeadresse.
7	Das Adresselement kann ohne mehrfache Änderungen an der Adresse nicht korrigiert werden. Die Adressvalidierung kann die Adresse korrigieren, aber die Anzahl von Änderungen zeigt, dass die Adresse nicht zuverlässig ist.
8	Die Daten entsprechen nicht den Validierungsregeln des Zustellers.

Ausgabeportwerte für Elementeingabe-Status

Der Status der Elementeingabe ist eine Zeichenfolge aus 20 Zeichen, bei der jedes Zeichen ein anderes Eingabeadresselement darstellt. Der Wert für jedes Zeichen stellt den Verarbeitungstyp dar, der für das Adresselement ausgeführt wird.

Den Port finden Sie in der Portgruppe für Statusinformationen.

In der folgenden Tabelle werden die Codes beschrieben, die der Elementeingabe-Status in jeder Position auf der Ausgabezeichenfolge im Batch-, Zertifizierungs- oder Vorschlagslistenmodus zurückgeben kann:

Code	Beschreibung
0	Die Eingabeadresse enthält keine Daten an der aktuellen Position.
1	Die Referenzdaten enthalten keine Daten an der aktuellen Position.
2	Die Daten können nicht überprüft werden, weil die Referenzdaten fehlen.
3	Die Daten an der aktuellen Position sind falsch. Die Referenzdatenbank schlägt vor, dass die Nummer oder der Zustelldienstwert außerhalb des Bereichs liegt, der von den Referenzdaten erwartet wird. Im Batch- und Zertifizierungsmodus übergibt die Umwandlung die Eingabedaten an dieser Position unkorrigiert als Ausgabe
4	Die Daten an der aktuellen Position stimmen mit den Referenzdaten überein, enthalten jedoch Fehler.
5	Die Daten der aktuellen Position stimmen mit den Referenzdaten überein, die Umwandlung hat die Daten jedoch korrigiert oder standardisiert.
6	Die Daten an der aktuellen Position stimmen mit den Referenzdaten fehlerfrei überein.

In der folgenden Tabelle werden die Codes beschrieben, die der Elementeingabe-Status in jeder Position auf der Ausgabezeichenfolge im Parsing-Modus zurückgeben kann:

Code	Beschreibung
0	Die Eingabeadresse enthält keine Daten an der aktuellen Position.
1	Die Umwandlung hat das Element an der aktuellen Position an eine andere Position in der Ausgabeadresse verschoben.
2	Das Element an der aktuellen Position stimmte mit dem Referenzdatenwert überein, die Umwandlung hat das Element in der Ausgabeadresse jedoch normalisiert.
3	Die Daten an der aktuellen Position sind korrekt.

Elementrelevanz-Ausgabeportwerte

Elementrelevanz-Werte geben an, ob ein Adressenelement für die Postzustellung erforderlich ist. Den Port finden Sie in der Portgruppe für Statusinformationen.

Der Elementrelevanzwert ist ein String aus 20 Zeichen, in dem jedes Zeichen einen anderen Typ von Adressdaten darstellen kann. Nachdem Sie das Adressvalidierung-Mapping ausgeführt haben, überprüfen Sie die Portausgabe, um die Adresselemente zu identifizieren, die für jede Adresse erforderlich sind. Verwenden Sie die Ergebnisse, um zu überprüfen, ob Sie die richtigen Ausgabeports für die Adressdaten ausgewählt haben. Wenn Sie keinen Ausgabeport für ein entsprechendes Adressdatenelement auswählen, ist die Ausgabe für diese Adresse nicht gültig.

In der folgenden Tabelle werden die Codes beschrieben, die „Elementrelevanz“ in jeder Position auf dem Ausgabestring zurückgeben kann:

Code	Beschreibung
0	Nicht relevant für die Zustellung an die Adresse.
1	Relevant für die Zustellung an die Adresse. Das nationale Postbeförderungsunternehmen kann ohne Daten an dieser Position im Ausgabestring keine Zustellung vornehmen.

Hinweis: Element-Relevanzwerte sind für Adressen mit dem Match-Code-Wert Cx oder Vx im Batch-Modus bzw. Cx, Vx, I3 oder I4 im interaktiven Modus verfügbar. Andere Bewertungscodes wie Elementeingabe Status, Elementergebnis Status, Erweiterter Elementergebnisstatus und Adressauflösungscode geben Werte unabhängig vom Match-Code-Wert zurück.

Ausgabeportwerte für Elementergebnisstatus

„Elementergebnisstatus“ ist eine Zeichenfolge aus 20 Zeichen, bei der jedes Zeichen ein anderes Eingabeadresselement darstellt. Der Wert für jedes Zeichen beschreibt ein Update, das der Validierungsprozess für das Adresselement vornimmt.

Den Port finden Sie in der Portgruppe für Statusinformationen.

In der folgenden Tabelle werden die Elementergebnisstatus-Portwerte beschrieben.

Code	Beschreibung
0	Die Ausgabeadresse enthält keine Daten an der aktuellen Position.
1	Die Umwandlung kann die Daten an der aktuellen Position in den Referenzdaten nicht finden. Die Umwandlung kopiert die Eingabedaten in die Ausgabedaten.
2	Die Daten sind an der aktuellen Position nicht aktiviert, aber standardisiert.
3	Die Daten an der aktuellen Position sind aktiviert, sie stimmen jedoch nicht mit den Referenzdaten überein. Die Referenzdaten lassen vermuten, dass sich die Zahlendaten nicht im gültigen Bereich befinden. Die Umwandlung kopiert die Eingabedaten in die Ausgabeports. Wird im Batchmodus angewendet.
4	Die Umwandlung kopiert die Eingabedaten in die Ausgabedaten, weil die Referenzdaten fehlen.
5	Die Daten an der aktuellen Position werden validiert, aber nicht geändert, weil in den Referenzdaten mehrere Matches vorhanden sind. Wird im Batchmodus angewendet.
6	Die Datenvalidierung hat den Eingabewert an der aktuellen Position gelöscht.
7	Die Daten an der aktuellen Position werden validiert, aber die Eingabedaten enthalten einen Rechtschreibfehler. Die Adressvalidierung hat den Fehler mit einem Wert aus den Referenzdaten korrigiert.
8	Die Daten an der aktuellen Position wurden mit einem Wert aus den Referenzdaten validiert und aktualisiert. Ein Wert von 8 kann auch bedeuten, dass die Referenzdatenbank zusätzliche Daten für das Eingabeelement enthält. Beispiel: Bei der Validierung kann eine Gebäudenummer oder eine Gebäudeteilnummer hinzugefügt werden, wenn eine genaue Übereinstimmung für den Straßennamen oder den Gebäudenamen gefunden wird.
9	Daten an der aktuellen Position werden validiert, aber nicht geändert. Außerdem ist der Zustellungsstatus unklar. Beispielsweise ist der DPV-Wert falsch.
C	Daten an der aktuellen Position werden validiert und überprüft, aber die Namensdaten sind veraltet. Die Namensdaten wurden von der Validierung geändert.
D	Daten an der aktuellen Position werden validiert und überprüft und von einem Exonym in eine offizielle Bezeichnung geändert.
E	Daten an der aktuellen Position werden validiert und überprüft. Die Groß-/Kleinschreibung der Zeichen oder die Sprache wurde jedoch von der Adressvalidierung standardisiert. Die Sprache kann von der Adressvalidierung geändert werden, wenn der Wert vollständig mit einer Sprachalternative übereinstimmt. Beispiel: Die Adressvalidierung kann „Brussels“ in einer belgischen Adresse in „Bruxelles“ ändern.
F	Aufgrund der genauen Übereinstimmung mit den Referenzdaten werden Daten an der aktuellen Position validiert, überprüft, aber nicht geändert.

Positionen 19 und 20 in der Ausgabezeichenfolge beziehen sich auf Landesdaten.

In der folgenden Tabelle werden die Werte, die bei der Validierung möglicherweise für die Positionen 19 und 20 zurückgegeben werden:

Code	Beschreibung
0	Die Ausgabeadresse enthält keine Daten an der aktuellen Position.
1	Die Adressvalidierung erkennt die Landesdaten nicht.
4	Die Adressvalidierung gibt das Land aus dem Wert „Standardland“ in der Adress-Validiererumwandlung an.
5	Die Adressvalidierung kann das Land nicht bestimmen, weil die Referenzdaten mehrere Übereinstimmungen enthalten.
6	Die Adressvalidierung gibt das Land aus einem Skript an.
7	Die Adressvalidierung gibt das Land aus dem Adressformat an.
8	Die Adressvalidierung gibt das Land aus Großstadt-Daten an.
9	Die Adressvalidierung gibt das Land aus Provinz-Daten an.
C	Die Adressvalidierung gibt das Land aus Gebiets-Daten an.
D	Die Adressvalidierung gibt das Land aus dem Namen des Landes an, der Name enthält jedoch Fehler.
E	Die Adressvalidierung gibt das Land aus den Adressdaten an, zum Beispiel aus einem ISO-Code oder einem Ländernamen.
F	Die Adressvalidierung gibt das Land aus dem Wertsatz „Land erzwingen“ in der Adress-Validiererumwandlung an.

Ausgabeportwerte für den erweiterten Elementergebnisstatus

„Erweiterter Elementergebnisstatus“ ist eine Zeichenfolge aus 20 Zeichen, bei der jedes Zeichen ein anderes Eingabeadresselement darstellt. Portausgabecodes ergänzen die Statusdaten der Ports „Elementeingabestatus“ und „Elementergebnisstatus“. Der Portausgabecode kann auch anzeigen, dass zusätzliche Informationen zu einem Adresselement in den Referenzdaten zur Verfügung stehen.

Den Port finden Sie in der Portgruppe für Statusinformationen.

In der folgenden Tabelle werden die Portwerte für den erweiterten Elementergebnisstatus beschrieben.

Code	Beschreibung
1	Die Adressreferenzdaten enthalten zusätzliche Informationen zu dem Adresselement. Die Adressvalidierung erfordert keine zusätzlichen Informationen.
2	Die Adressvalidierung hat das Adresselement aktualisiert, um einen Datenfehler oder Formatfehler zu beheben. Die Adressvalidierung hat das Adresselement nicht überprüft.

Code	Beschreibung
3	Die Adressvalidierung hat das Adresselement aktualisiert, um einen Datenfehler oder Formatfehler zu beheben. Die Adressvalidierung hat die Nummerndaten in dem Adresselement überprüft.
4	Die Adressvalidierung hat das Adresselement in ein anderes Feld verschoben, um einen Datenfehler zu beheben.
5	Adressreferenzdaten enthalten eine alternative Version des Adresselements, beispielsweise den Namen des bevorzugten Orts.
6	Die Adressvalidierung hat nicht alle Teile des Adresselements überprüft. Das Element enthält Daten, die die Adressvalidierung nicht validieren kann.
7	Die Adressvalidierung hat ein gültiges Adresselement an der falschen Stelle in einer Adresse gefunden. Die Adressvalidierung hat das Adresselement an die richtige Position verschoben.
8	Die Adressvalidierung hat ein gültiges Adresselement im falschen Datenfeld gefunden. Die Adressvalidierung hat das Adresselement in das richtige Feld verschoben.
9	Die Adressvalidierung hat das Ausgabeelement entsprechend den Validierungsregeln des Postbeförderungsunternehmens generiert.
A	Die Adressvalidierung hat Adresselemente verschiedener Adresstypen gefunden, die für die aktuelle Position in Frage kommen. Die Adressvalidierung hat das Ausgabe-Adresselement ausgewählt, das den Regeln des Postbeförderungsunternehmens im Zielland entspricht.
B	Die Adressvalidierung kann die Relevanz des Elements nicht ermitteln. Die Adressvalidierung gibt den Standardwert für das Land zurück, das in der Adresse angegeben ist.
C	Vorschlagslistenmodus. Die Adressvalidierung kann zusätzliche Adressenvorschläge für das Adresselement zurückgeben. Um die zusätzlichen Vorschläge zurückzugeben, aktualisieren Sie die Eigenschaft „Zähler Maximalergebnis“ für die Adress-Validiererumwandlung.
D	Die Adressvalidierung hat die Nummerndaten im Adresselement interpoliert.
E	Die Adressvalidierung kann das Adresselement in der bevorzugten Sprache nicht zurückgeben. Die Adressvalidierung gibt das Element in der Standardsprache zurück.
F	Adresscode-Lookup-Modus. Die Eingabeadresse ist veraltet.

Ausgabeportwerte für Zustellbarkeits-Score

In der folgenden Tabelle werden die Ausgabeportwerte für den Zustellbarkeits-Score beschrieben. Diesen Port finden Sie in der Portgruppe für Statusinformationen.

Die Adressvalidator-Umwandlung schätzt die Zustellbarkeit einer Adresse ein und schreibt diese Schätzung in Form einer Zahl auf den Zustellbarkeits-Score-Port. Ziehen Sie diese Zahl heran, wenn der Match-Code-Wert im Bereich I1 bis I4 liegt.

In der folgenden Tabelle werden die Ausgabecodes für den Zustellbarkeits-Score beschrieben:

Code	Beschreibung
5	Zustellung ganz sicher
4	Zustellen nahezu sicher
3	Zustellung sicher
2	Zustellung nahezu sicher
1	Zustellung zweifelhaft
0	Nicht zustellungsfähig

Ausgabeportwerte für Match-Code

In der folgenden Tabelle werden die Ausgabeportwerte für den Match-Code beschrieben. Diesen Port finden Sie in der Portgruppe für Statusinformationen.

Code	Beschreibung
A1	Beim Adresscode-Lookup wurde eine unvollständige oder eine vollständige Adresse für den Eingabecode gefunden.
A0	Beim Adresscode-Lookup wurde keine Adresse für den Eingabecode gefunden.
C4	Korrigiert. Alle postalisch relevanten Elemente wurden überprüft.
C3	Korrigiert. Bestimmte Elemente können nicht überprüft werden.
C2	Korrigiert, aber der Zustellstatus ist aufgrund von fehlenden Referenzdaten unklar.
C1	Korrigiert, aber der Zustellstatus ist aufgrund von Fehlern durch die Benutzerstandardisierung unklar.
I4	Daten können nicht vollständig berichtigt werden, aber es ist ein einzelnes Match mit einer Adresse in den Referenzdaten vorhanden.
I3	Daten können nicht vollständig berichtigt werden, aber es sind mehrere Matches mit Adressen in den Referenzdaten vorhanden.
I2	Daten können nicht berichtigt werden. Batchmodus gibt teilweise vorgeschlagene Adressen zurück.
I1	Daten können nicht berichtigt werden. Batchmodus kann keine Adresse vorschlagen.
N7	Validierungsfehler. Es wurde keine Validierung durchgeführt, da die Einzelzeilenvalidierung nicht entsperrt ist.
N6	Validierungsfehler. Es wurde keine Validierung durchgeführt, da Einzelzeilenvalidierung für das Zielland nicht unterstützt wird.

Code	Beschreibung
N5	Validierungsfehler. Es wurde keine Validierung durchgeführt, da die Referenzdatenbank veraltet ist.
N4	Validierungsfehler. Es wurde keine Validierung durchgeführt, da die Referenzdaten beschädigt oder schlecht formatiert sind.
N3	Validierungsfehler. Es wurde keine Validierung durchgeführt, da die Länderdaten nicht entsperrt werden können.
N2	Validierungsfehler. Es wurde keine Validierung durchgeführt, da die benötigte Referenzdatenbank nicht verfügbar ist.
N1	Validierungsfehler. Es wurde keine Validierung durchgeführt, da das Land nicht erkannt oder nicht unterstützt wird.
Q3	Vorschlagslistenmodus. Die Adressvalidierung kann nicht eine oder mehrere vollständige Adressen aus den Adressreferenzdaten abrufen, die der Eingabeadresse entsprechen.
Q2	Vorschlagslistenmodus. Die Adressvalidierung kann die Eingabeadresselemente und die Elemente aus den Adressreferenzdaten kombinieren, um eine vollständige Adresse zu erstellen.
Q1	Vorschlagslistenmodus. Die Adressvalidierung kann keine vollständige Adresse vorschlagen. Um einen Vorschlag für eine vollständige Adresse zu generieren, fügen Sie der Eingabeadresse Daten hinzu.
Q0	Vorschlagslistenmodus. Es sind nicht genügend Eingabedaten zum Generieren eines Vorschlags vorhanden.
RB	Land aus Abkürzung erkannt. Erkennt zwei- und dreistellige ISO-Ländercodes. Kann auch allgemeine Abkürzungen wie „GER“ für Deutschland erkennen.
RA	Land aus der Einstellung „Land erzwingen“ in der Umwandlung erkannt.
R9	Land aus der Einstellung „Standardland“ in der Umwandlung erkannt.
R8	Anhand des Ländernamens erkanntes Land.
R7	Anhand des Ländernamens erkanntes Land. Die Umwandlung hat jedoch Fehler in den Länderdaten ermittelt.
R6	Anhand der Gebietsdaten erkanntes Land.
R5	Anhand der Provinzdaten erkanntes Land.
R4	Anhand der Großstadtdaten erkanntes Land.
R3	Anhand des Adressformats erkanntes Land.
R2	Anhand eines Skripts erkanntes Land.
R1	Nicht erkanntes Land, da mehrere Entsprechungen zur Verfügung stehen.
R0	Land nicht erkannt.

Code	Beschreibung
S4	Parsing-Modus. Die Adresse wurde ordnungsgemäß geparst.
S3	Parsing-Modus. Die Adresse wurde mit mehreren Ergebnissen geparst.
S1	Parsing-Modus. Aufgrund einer Nichtübereinstimmung beim Eingabeformat ist ein Parsing-Fehler aufgetreten.
V4	Überprüft. Die Eingabedaten sind korrekt. Die Adressvalidierung prüfte alle postalisch relevanten Elemente und die Eingaben stimmten vollständig überein.
V3	Überprüft. Die Eingabedaten sind korrekt, aber einige oder alle Elemente wurden standardisiert, oder die Eingabe enthält veraltete Namen oder Synonyme.
V2	Überprüft. Die Eingabedaten sind korrekt, aber einige Elemente konnten aufgrund unvollständiger Referenzdaten nicht überprüft werden.
V1	Überprüft. Die Eingabedaten sind korrekt, aber die Benutzerstandardisierung hat negative Auswirkungen auf die Zustellbarkeit. Die Postleitzahl ist beispielsweise zu kurz.

Ausgabeportwerte für Geokodierungsstatus

In der folgenden Tabelle werden die Ausgabeportwerte für den Geokodierungsstatus beschrieben. Diesen Port finden Sie in der Portgruppe für Geokodierung.

Wählen Sie diesen Port aus, wenn Sie Geokodierungsreferenzdaten für ein Eingabeadressland installiert haben.

Wert	Beschreibung
EGC0	Geocodes können nicht an die Eingabeadresse angehängt werden, weil Geocodes für die Adresse nicht verfügbar sind.
EGC1-3	Für zukünftige Verwendung reserviert.
EGC4	Geocodes stimmen teilweise mit der Ebene für Postleitzahlen überein.
EGC5	Geocodes stimmen mit der Ebene für Postleitzahlen überein.
EGC6	Geocodes stimmen mit der Ebene für Orte überein.
EGC7	Geocodes stimmen mit der Ebene für Straßen überein.
EGC8	Geocodes stimmen mit der Ebene für Hausnummern überein. Geocodes schätzen den Standort für Hausnummern und beinhalten einen Versatz an der Seite der Straße, die den Briefkasten enthält.
EGC9	Geocodes stimmen mit dem Ankunftspunkt oder Rooftop überein.
EGCA	Geocodes stimmen mit dem Zentrum eines Grundstücks überein.
EGCC	Die Geocode-Datenbank ist beschädigt.

Wert	Beschreibung
EGCN	Die Geocode-Datenbank wurde nicht gefunden.
EGCU	Die Geocode-Datenbank ist nicht entsperrt.

Adressreferenzdaten-Dateistatus

Verwenden Sie das Developer-Tool zum Überprüfen des Status der Adressreferenzdaten-Dateien in der Domäne. Die Statusinformationen enthalten das Lizenzablaufdatum für jede Datei sowie den Verarbeitungstyp, den Sie für die Datei verwenden können.

Verwenden Sie das Fenster **Einstellungen** im Developer-Tool, um den Status der Adressreferenzdatendatei zu überprüfen. Wählen Sie die Option **Inhaltsstatus** im Fenster **Einstellungen** aus, um die Statusinformationen zu überprüfen.

In der folgenden Tabelle werden die Statuseigenschaften beschrieben, die bei Auswahl von **Inhaltsstatus** standardmäßig angezeigt werden:

Eigenschaft	Beschreibung
ISO-Länderkürzel	Das Land, für das die Adressreferenzdatendatei gilt. Diese Eigenschaft zeigt die dreistellige ISO-Abkürzung für das Land.
Ablaufdatum	Das Datum, an dem die Datei durch eine neuere Datei ersetzt werden muss. Sie können eine Adressreferenzdatendatei nach ihrem Ablaufdatum verwenden, die Daten in der Datei sind aber unter Umständen nicht mehr verlässlich.
Ländertyp	Die Art der Adressverarbeitung, die Sie mit den Daten durchführen können. Sie wählen auf der Registerkarte **Allgemeine Einstellungen** die Verarbeitungsart in der Option „Modus“ aus. Wenn der ausgewählte Modus nicht mit einer Adressdatendatei in der Domäne übereinstimmt, schlägt das Adressvalidierungs-Mapping fehl.
Ablaufdatum entsperren	Das Datum, an dem die Lizenz der Datei abläuft. Sie können die Datei nicht mehr verwenden, nachdem das Ablaufdatum entsperrt wurde.
Startdatum entsperren	Das Datum, ab dem die Lizenz für die Datei gültig ist. Sie können die Datei erst verwenden, wenn das Startdatum entsperrt wurde.

Klicken Sie mit der rechten Maustaste auf die Eigenschaftentabelle, um eine Liste mit zusätzlichen Eigenschaften anzuzeigen.

Allgemeine Einstellungen der Adressvalidator-Umwandlung

Konfigurieren Sie die allgemeinen Einstellungen zum Einrichten der erforderlichen Parameter für die Adressvalidierung.

Sie können die folgenden Eigenschaften in der Ansicht **Allgemeine Einstellungen** konfigurieren:

Standardland

Legt den von der Umwandlung verwendeten Adressreferenzdatensatz fest, wenn aus den Eingabeadressdaten keine Länderinformationen ermittelt werden können. Wählen Sie Keine aus, wenn die Daten Länderinformationen enthalten.

Land erzwingen

Optionale Eigenschaft. Ersetzt den Ländernamen oder die Abkürzung in der Eingabeadresse durch den Namen oder die Abkürzung des Standardlands. Wenn die Eingabeadresse kein Land identifiziert, fügt die Umwandlung die Daten des Standardlands zur Adresse hinzu.

Zeilentrennzeichen

Gibt das Trennzeichensymbol an, das Datenfelder innerhalb einer einzeiligen Adresse trennt.

Schreibweise

Legt den Buchstabenstil für Ausgabedaten fest. Wählen Sie die Option Gemischt aus, um den Standard für die Adressreferenzdaten für große Anfangsbuchstaben einzuhalten. Wählen Sie die Option Beibehalten aus, um die Adresse in der Schreibweise einzugeben, die für die Adressreferenzdaten verwendet wird.

Modus

Bestimmt den Typ der Validierung, die von der Umwandlung durchgeführt wird. Der Standardmodus lautet Batch.

Wählen Sie eine der folgenden Optionen aus:

Modustyp	Beschreibung
Adresscode-Lookup	Gibt eine Teiladresse oder eine vollständige Adresse aus den Referenzdaten zurück, wenn Sie einen Adresscode als eine Eingabe verwenden. Mehrere Länder unterstützen Adresscodes, die den Ort, die Straße, das Gebäude oder ein eindeutiges Postfach angeben.
Batch	Führt Adressvalidierung für die Datensätze in einem Dataset durch. Die Batchvalidierung zielt auf die Vollständigkeit der Adresse und die Zustellbarkeit ab. Der Batchmodus gibt keine Vorschläge für Adressen mit schlechter Qualität zurück.
Zertifiziert	Führt eine Adressvalidierung für die Datensätze in einem Dataset bezüglich der Zertifizierungsstandards des angegebenen Lands durch. Für die Zertifizierungsstandards ist es erforderlich, dass sich jede Adresse auf ein eindeutiges Postfach bezieht. Sie können die zertifizierte Adressvalidierung für Adressen in Australien, Frankreich, Neuseeland, im Vereinigten Königreich und in den Vereinigten Staaten durchführen.
Ländererkennung	Legt das Zielland für die Postadresse fest. Die Umwandlung führt im Ländererkennungsmodus keine Adressvalidierung durch.

Modustyp	Beschreibung
Interaktiv	Vervollständigt eine unvollständige gültige Adresse. Wenn eine unvollständige Eingabeadresse mehr als einer Adresse in den Referenzdaten entspricht, gibt die Umwandlung alle gültigen Adressen bis zu dem unter „Maximale Ergebniszahl“ festgelegten Limit zurück.
Parse	Parst Daten in Adressfelder. Die Umwandlung führt im Parsing-Modus keine Adressvalidierung durch.
Vorschlagsliste	Gibt eine Liste von gültigen Adressen aus den Referenzdaten zurück, wenn eine Eingabeadresse bruchstückhafte Informationen enthält. Wenn ein Adressfragment mit mehr als einer Adresse in den Referenzdaten übereinstimmt, gibt die Umwandlung alle gültigen Adressen bis zu dem unter „Maximale Ergebniszahl“ festgelegten Limit zurück.

Adressvalidator-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie erweiterte Eigenschaften, um anzugeben, wie der Data Integration Service Daten für die Adressvalidator-Umwandlung verarbeitet.

Ortsalias

Bestimmt, ob die Adressvalidierung einen gültigen Ortsalias durch den offiziellen Straßennamen ersetzt.

Ein Ortsalias ist eine andere Bezeichnung desselben Orts, die von der Post als Element in einer Zustelladresse erkannt wird. Sie können diese Eigenschaft verwenden, wenn Sie die Adressüberprüfungsumwandlung für die Überprüfung US-amerikanischer Adressdatensätze im zertifizierten Modus konfigurieren.

In der folgenden Tabelle werden Optionen für den Ortsalias beschrieben:

Option	Beschreibung
Aus	Deaktiviert die Eigenschaft „Aliasort“.
Offiziell	Ersetzt sämtliche alternativen Ortnamen oder Ortsaliase durch den offiziellen Ortsnamen. Standardoption.
Beibehalten	Behält einen gültigen alternativen Ortsnamen oder Ortsalias bei. Wenn der Eingabeortsname ungültig ist, ersetzt die Adressüberprüfung den Namen durch den offiziellen Namen.

Alternativer Straßenname

Bestimmt, ob die Adressvalidierung einen Straßenalias durch den offiziellen Straßennamen ersetzt.

Ein alternativer Straßenname ist eine andere Bezeichnung derselben Straße, die von der Post als Element in einer Zustelladresse erkannt wird. Sie können diese Eigenschaft verwenden, wenn Sie die

Adressüberprüfungsumwandlung für die Überprüfung US-amerikanischer Adressdatensätze im zertifizierten Modus konfigurieren.

Die folgende Tabelle beschreibt die Aliasstraße-Optionen:

Option	Beschreibung
Aus	Wendet die Eigenschaft nicht an.
Offiziell	Ersetzt sämtliche alternative Straßennamen oder Straßenalias durch den offiziellen Straßennamen. Standardoption.
Beibehalten	Behält einen gültigen alternativen Straßennamen oder Straßenalias bei. Wenn der Eingabestraßenname ungültig ist, ersetzt die Adressüberprüfung den Namen durch den offiziellen Straßennamen.

Schreibweise

Bestimmt die Groß- und Kleinschreibung, die die Umwandlung zum Schreiben von Ausgabedaten verwendet.

Die folgende Tabelle beschreibt die Schreibweiseoptionen:

Option	Beschreibung
Parameter zuweisen	Verwendet einen von Ihnen definierten Parameter, um die Schreibweise festzulegen.
Kleinbuchstaben	Schreibt die Ausgabeadresse in Kleinbuchstaben.
Gemischt	Verwendet die im Zielland verwendete Schreibweise, sofern dies möglich ist.
Datenbank	Wendet die in den Adressreferenzdaten verwendete Schreibweise an. Standardoption.
Beibehalten	Schreibt die Ausgabeadresse mit derselben Groß-/Kleinschreibung wie die Eingabeadresse.
Großbuchstaben	Schreibt die Ausgabeadresse in Großbuchstaben.

Sie können die Schreibweise auch auf der Registerkarte **Allgemeine Einstellungen** konfigurieren.

Parameterverwendung

Sie können einen der folgenden Parameter verwenden, um die Schreibweise anzugeben:

- LOWER. Schreibt die Ausgabeadresse in Kleinbuchstaben.
- MIXED. Verwendet die im Zielland verwendete Schreibweise, sofern dies möglich ist.
- NATIVE. Wendet die in den Adressreferenzdaten verwendete Schreibweise an. Standardoption. Gleicht die Datenbankoption auf der Registerkarte **Allgemeine Einstellungen** ab.
- NOCHANGE. Schreibt die Ausgabeadresse mit derselben Groß-/Kleinschreibung wie die Eingabeadresse. Gleicht die beibehaltene Option auf der Registerkarte **Allgemeine Einstellungen** ab.
- UPPER. Schreibt die Ausgabeadresse in Großbuchstaben.

Geben Sie den Parameterwert in Großbuchstaben ein.

Herkunftsland

Identifiziert das Land, in dem die Adressdatensätze versendet werden.

Wählen Sie ein Land aus der Liste aus. Die Eigenschaft ist standardmäßig leer.

Ländertyp

Bestimmt das Format des Ländernamens bzw. der Länderabkürzung im Port "Vollständige Adresszeile" oder "Formatierte Adresszeile" der Ausgabedaten. Die Umwandlung schreibt den Ländernamen oder die -abkürzung im Standardformat des ausgewählten Landes.

Die folgende Tabelle beschreibt die Optionen für den Ländertyp:

Option	Land
ISO 2	Zweistelliger ISO-Ländercode
ISO 3	Dreistelliger ISO-Ländercode
ISO #	Dreistelliger ISO-Ländercode
Abkürzung	(Für zukünftige Verwendung reserviert)
CN	Kanada
DA	(Für zukünftige Verwendung reserviert)
DE	Deutschland
EN	Großbritannien (Standard)
ES	Spanien
FI	Finnland
FR	Frankreich
GR	Griechenland
IT	Italien
JP	Japan
HU	Ungarn
KR	Korea, Republik
NL	Niederlande
PL	Polen
PT	Portugal
RU	Russland

Option	Land
SA	Saudi-Arabien
SE	Schweden

Standardland

Legt den Adressreferenzdatensatz fest, den die Umwandlung verwendet, wenn ein Adressdatensatz ein Zielland nicht identifiziert.

Wählen Sie ein Land aus der Liste aus. Verwenden Sie die Standardoption, wenn die Adressdatensätze Länderinformationen enthalten. Die Standardeinstellung ist „Keine“.

Sie können das Standardland auch auf der Registerkarte **Allgemeine Einstellungen** konfigurieren.

Parameterverwendung

Sie können einen Parameter verwenden, um das Standardland anzugeben. Wenn Sie den Parameter erstellen, geben Sie den ISO 3166-1 alpha-3-Code für das Land als Parameterwert ein. Wenn Sie einen Parameterwert eingeben, verwenden Sie Großbuchstaben. Beispiel: Wenn alle Adressdatensätze Länderinformationen enthalten, geben Sie NONE ein.

Priorität bei doppelter Adresse

Bestimmt den Typ der zu validierenden Adresse. Legen Sie die Eigenschaft fest, wenn die Eingabeadressdatensätze mehr als einen Typ gültiger Adressdaten enthält.

Verwenden Sie die Eigenschaft zum Beispiel, wenn ein Adressdatensatz sowohl Postfachelemente als auch Straßenelemente enthält. Die Adressvalidierung liest die Datenelemente mit dem Typ der Adressdaten, die Sie angeben. Die Adressvalidierung ignoriert alle inkompatiblen Daten in der Adresse.

Die folgende Tabelle beschreibt die Optionen für die Eigenschaft "Priorität bei doppelter Adresse":

Option	Beschreibung
Zustelldienst	Validiert die Datenelemente des Zustelldiensts in einer Adresse wie beispielsweise Postfachelemente.
Post-Admin	Validiert die Adresselemente, die der lokale Zusteller benötigt. Standardoption.
Straße	Validiert Straßendatenelemente in einer Adresse wie beispielsweise Gebäudenummerelemente und Straßennamenelemente.

Elementabkürzung

Bestimmt, ob die Umwandlung die Abkürzungsform eines Adresselements zurückgibt. Sie können die Umwandlung so einstellen, dass die abgekürzte Form zurückgegeben wird, wenn die Adressreferenzdaten Abkürzungen enthalten.

Beispiel: Der Standard des United States Postal Service (USPS) erfordert kurze und lange Formate für viele Straßen und Ortsnamen. Die Kurzform von `HUNTSVILLE BROWNSFERRY RD` lautet beispielsweise `HSV BROWNS FRY RD`. Sie können die Eigenschaft „Elementabkürzung“ auswählen, wenn die Straßen- oder Ortswerte die maximale Feldlänge überschreiten, die von USPS festgelegt wird.

Die Option ist standardmäßig deaktiviert. Setzen Sie die Eigenschaft auf „ON“, um die angekürzte Adresse wiederzugeben. Die Eigenschaft gibt den abgekürzten Ortsnamen und Ortscode zurück, wenn Sie die Umwandlung im Batchmodus verwenden. Die Eigenschaft gibt den abgekürzten Straßennamen, Ortsnamen und Ortscode zurück, wenn Sie die Umwandlung im zertifizierten Modus verwenden.

Ausführungsinstanzen

Gibt die Anzahl der Threads an, die der Datenintegrationsdienst für die aktuelle Umwandlung zur Laufzeit zu erstellen versucht. Der Datenintegrationsdienst berücksichtigt den Wert „Ausführungsinstanzen“, wenn Sie die Laufzeiteigenschaft „Maximaler Parallelismus“ auf demjenigen Mapping überschreiben, das die Umwandlung enthält. Der Standardwert für Ausführungsinstanzen ist 1.

Der Datenintegrationsdienst berücksichtigt mehrere Faktoren bei der Ermittlung der Anzahl von Threads, die der Umwandlung zugewiesen werden sollen. Die wichtigsten Faktoren sind der Wert „Ausführungsinstanzen“ und die Werte für das Mapping und für die zugeordneten Anwendungsdienste in der Domäne.

Der Datenintegrationsdienst liest die folgenden Werte, wenn er die Anzahl der für die Umwandlung zu verwendenden Threads berechnet:

- Der Wert *Maximaler Parallelismus* für den Datenintegrationsdienst. Standardwert ist 1.
- Alle *Maximaler Parallelismus*-Werte, die Sie auf der Mapping-Ebene festlegen. Standardwert ist „Auto“.
- Der Wert *Ausführungsinstanzen* für die Umwandlung. Standardwert ist 1.

Wenn Sie den Wert „Maximaler Parallelismus“ auf der Mapping-Ebene überschreiben, versucht der Datenintegrationsdienst, den niedrigsten Wert innerhalb der Eigenschaften für das Ermitteln der Anzahl der Threads zu verwenden.

Wenn Sie den Standardwert für „Maximaler Parallelismus“ auf der Mapping-Ebene verwenden, ignoriert der Datenintegrationsdienst den Wert „Ausführungsinstanzen“.

Der Datenintegrationsdienst berücksichtigt auch die Eigenschaft *Max. Adressobjektanzahl* für den Content-Management-Dienst, wenn er die Anzahl der zu erstellenden Threads berechnet. Mit der Eigenschaft *Max. Adressobjektanzahl* wird die maximale Anzahl der Adressvalidierungsinstanzen festgelegt, die in einem Mapping gleichzeitig ausgeführt werden können. Der Eigenschaftswert für *Max. Adressobjektanzahl* muss größer oder gleich dem Wert für *Maximaler Parallelismus* auf dem Datenintegrationsdienst sein.

Regeln und Richtlinien für die Eigenschaft „Ausführungsinstanzen“

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie die Anzahl der Ausführungsinstanzen festlegen:

- Mehrere Benutzer führen möglicherweise gleichzeitige Mappings auf einem Datenintegrationsdienst aus. Um die korrekte Anzahl der Threads zu berechnen, dividieren Sie die Anzahl der CPUs, auf die der Dienst zugreifen kann, durch die Anzahl gleichzeitiger Mappings.
- In PowerCenter wird in der Konfigurationsdatei *AD50.cfg* die maximale Anzahl der Adressvalidierungsinstanzen angegeben, die in einem Mapping gleichzeitig ausgeführt werden können.
- Wenn Sie die Standardwerte für „Ausführungsinstanzen“ und „Maximaler Parallelismus“ verwenden, sind die Umwandlungsoperationen nicht partitionierbar.
- Wenn Sie einen Wert „Ausführungsinstanzen“ auf größer als 1 festlegen, können Sie die Adressvalidierer-Umwandlung von einer passiven Umwandlung in eine aktive Umwandlung ändern.

Flexible Bereichserweiterung

Beschränkt den praktischen Grenzwert für die Anzahl von Adressen, die die Adress-Validierungsumwandlung zurückgibt, wenn Sie die Eigenschaft "Zu erweiternde Bereiche" festlegen. Sie können die Eigenschaft "Zu

erweiternde Bereiche" und "Flexible Bereichserweiterung" festlegen, wenn Sie die Umwandlung so konfigurieren, dass sie im Vorschlagslisten-Modus ausgeführt wird.

Die Eigenschaft "Zu erweiternde Bereiche" bestimmt, wie die Umwandlung Addressvorschläge zurückgibt, wenn eine Eingabeadresse keine Hausnummerndaten enthält. Wenn die Eingabeadresse keine Kontextinformationen enthält wie eine vollständige Postleitzahl, kann die Eigenschaft "Zu erweiternde Bereiche" eine große Anzahl sehr ähnlicher Adressen generieren. Die Eigenschaft "Flexible Bereichserweiterung" begrent die Anzahl von Adressen, die die Eigenschaft "Zu erweiternde Bereiche" für eine einzelne Adresse generiert. Legen Sie die Eigenschaft "Flexible Bereichserweiterung" auf "On" fest, wenn Sie die Eigenschaft "Zu erweiternde Bereiche" auf "Alle" festlegen.

Die folgende Tabelle beschreibt die Optionen für die Eigenschaft "Flexible Bereichserweiterung":

Option	Beschreibung
Ein	Die Adressvalidierung begrenzt die Anzahl von Adressen, die die Eigenschaft "Zu erweiternde Bereiche" zur Vorschlagsliste hinzufügt. Standardoption.
Aus	Die Adressvalidierung begrenzt nicht die Anzahl von Adressen, die die Eigenschaft "Zu erweiternde Bereiche" zur Vorschlagsliste hinzufügt.

Hinweis: Die Adress-Validierungsumwandlung wendet die Eigenschaft "Flexible Bereichserweiterung" auf eine andere Art und Weise auf jede Adresse an, die sie zur Vorschlagsliste zurückgibt. Die Umwandlung erfordert keinen festen Grenzwert für die Anzahl erweiterter Adressen in der Liste. Die Umwandlung berücksichtigt auch die Eigenschaftseinstellung "Maximale Ergebniszahl", wenn sie die Anzahl der erweiterten Adressen berechnet, die in die Liste aufgenommen werden sollen.

Datentyp „Geocode“

Bestimmt, wie die Adress-Validierungsumwandlung die Geokodierungsdaten für eine Adresse berechnet. Geocodes sind Breiten- und Längengradkoordinaten.

Sie können eine der folgenden Geokodierungsoptionen auswählen:

Ankunftspunkt

Gibt die Breiten- und Längengradkoordinaten des Eingangs zum Gebäude oder Grundstück zurück. Standardoption.

Sie können die Ankunftspunktoption für Adressen in den folgenden Länder auswählen:

Österreich, Kanada, Dänemark, Finnland, Deutschland, Ungarn, Lettland, Luxemburg, Mexiko, Niederlande, Norwegen, Slowenien, Schweden und die USA.

Wenn Sie Ankunftspunkt-Geocodes angeben und die Adress-Validiererumwandlung keine Geocodes für eine Adresse zurückgeben kann, stellt die Umwandlung interpolierte Geocodes bereit.

Parcel Centroid

Gibt die Breiten- und Längengradkoordinaten des geografischen Zentrums eines Grundstücks auf Bodenniveau zurück.

Sie können die Parcel Centroid-Option für Adressen in den folgenden Ländern auswählen:

Österreich, Kanada, Dänemark, Finnland, Deutschland, Ungarn, Lettland, Luxemburg, Niederlande, Norwegen, Slowenien, Schweden und USA.

Wenn Sie Parcel Centroid-Geocodes angeben und die Adress-Validiererumwandlung keine Geocodes für eine Adresse zurückgeben kann, stellt die Umwandlung keine Geokodierungsdaten bereit.

Rooftop

Gibt die Breiten- und Längengradkoordinaten zur Angabe des physischen Mittelpunkts des Gebäudes zurück, das den Briefkasten enthält. Die Adress-Validiererumwandlung gibt Rooftop-Koordinaten für britische Adressen zurück.

Wenn Sie Rooftop-Geocodes angeben und die Adress-Validiererumwandlung keine Geocodes für eine Adresse zurückgeben kann, stellt die Umwandlung keine Geokodierungsdaten bereit.

Standard

Gibt die geschätzten Breiten- und Längengradkoordinaten des Eingangs zum Gebäude oder Grundstück zurück. Geschätzte Geocodes werden als interpolierte Geocodes bezeichnet.

Die Adress-Validiererumwandlung verwendet die nächstliegenden verfügbaren Geocodes in den Adressreferenzdaten, um die Geocodes für die Adresse zu schätzen.

Parameterverwendung

Sie können einen Parameter verwenden, um den Geocode-Typ anzugeben. Geben Sie ARRIVAL_POINT, PARCEL_CENTROID, ROOFTOP oder NONE ein. Geben Sie NONE ein, um die Standard-Geocodes zurückzugeben.

Geben Sie den Parameterwert in Großbuchstaben ein.

Globale max. Feldlänge

Bestimmt die maximale Anzahl von Zeichen auf einer Zeile in der Adresse. Wenn die Adress-Validierungsumwandlung eine Ausgabe-Adresszeile schreibt, die mehr Zeichen enthält, als Sie festgelegt haben, kürzt die Umwandlung die Adresselemente auf der Linie ab.

Verwenden Sie die Eigenschaft zur Steuerung der Zeilenlänge in der Adresse. Beispiel: Die SNA-Standards verlangen, dass eine Adresse nicht mehr als 38 Zeichen auf einer Zeile enthält. Wenn Sie Adressen im SNA-Standard generieren, legen Sie die globale max. Feldlänge auf 38 fest.

Der Standardwert ist 1024.

Parameterverwendung

Sie können einen Parameter verwenden, um die maximale Anzahl von Adressen anzugeben. Geben Sie zum Festlegen der Parameterwerte eine Ganzzahl von 0 bis 1024 ein.

Global bevorzugter Deskriptor

Legt das Format der Deskriptoren für Gebäude, Gebäudeteile und Straßen fest, die von der Adress-Validiererumwandlung in die Ausgabedaten geschrieben werden. Wählen Sie einen Deskriptor aus, wenn die Adressreferenzdaten für das Zielland einen Deskriptorenbereich für mindestens ein Datenelement enthalten.

In der folgenden Tabelle werden die Optionen für die Eigenschaft beschrieben:

Option	Beschreibung
Datenbank	Gibt den Deskriptor zurück, den die Referenzdatenbank für das Element in der Adresse angibt. Wenn die Datenbank keinen Deskriptor für die Adresse angibt, kopiert die Umwandlung den Eingabewert in die Ausgabeadresse. Datenbank ist der Standardwert.
Lang	Gibt die vollständige Form des Deskriptors zurück, zum Beispiel *Straße*.

Option	Beschreibung
Eingabe beibehalten	Kopiert den Deskriptor aus der Eingabeadresse in die Ausgabeadresse. Wenn der Eingabedeskriptor keine gültige Version des Deskriptors ist, gibt die Umwandlung einen gleichwertigen gültigen Deskriptor aus der Referenzdatenbank zurück.
Kurz	Gibt eine abgekürzte Form des Deskriptors zurück, zum Beispiel *St*.

Eingabeformattyp

Beschreibt den häufigsten Informationstyp, der in Freitext enthalten ist. Verwenden Sie die Eigenschaft „Eingabeformattyp“, wenn Sie Eingabedaten mit folgenden Ports verwenden: „Vollständige Adresse“ oder „Formatierte Adresszeile“. Wählen Sie die Option aus, die die Informationen in den Mapping-Quelldaten am besten beschreibt.

Wählen Sie eine der folgenden Optionen aus:

- Alle
- Adresse
- Unternehmen
- Ansprechpartner
- Unternehmen/Ansprechpartner
  Die Adresse enthält Unternehmens- und Kontaktinformationen.
- Unternehmen/Abteilung
  Die Adresse enthält Unternehmens- und Abteilungsinformationen.

Voreingestellt ist „Alle“.

Eingabeformat mit Land

Legt fest, ob die Eingabe Länderdaten enthält. Wählen Sie diese Eigenschaft aus, wenn Sie Eingabedaten mit dem Port „Vollständige Adresse“ oder „Formatierte Adresszeile“ verbinden und die Daten Länderinformationen enthalten.

Die Option ist standardmäßig deaktiviert.

Zeilentrennzeichen

Legt das Trennzeichensymbol fest, das Zeilenumbrüche in einer formatierten Adresse anzeigt.

Wählen Sie eine der folgenden Optionen aus:

- Weisen Sie einen Parameter zu, der das Zeilentrennzeichen identifiziert.
- Wagenrücklauf
- Komma
- Zeilenvorschub/LF
- Senkrechter Strich
- Semikolon
- Leerzeichen

- Registerkarte
- Windows-Zeilenumbruch/CRLF

Standardwert ist Semikolon.

Sie können das Zeilentrennzeichen auch auf der Registerkarte **Allgemeine Einstellungen** konfigurieren.

Parameterverwendung

Sie können einen Parameter zur Angabe des Zeilentrennzeichens verwenden. Der Parameterwert unterscheidet zwischen Groß- und Kleinschreibung. Geben Sie den Parameterwert in Großbuchstaben ein.

Geben Sie einen der folgenden Werte ein:

- CR
- COMMA
- LF
- PIPE
- SEMICOLON
- SPACE
- TAB
- CRLF

Matching-Alternativen

Bestimmt, ob die Adressvalidierung alternative Ortsnamen in Eingabeadressen erkennt, zum Beispiel Synonyme oder frühere Namen. Die Eigenschaft gilt für Straßen-, Orts- und Provinzdaten.

Hinweis: Die Eigenschaft „Matching-Alternativen“ behält alternative Namen in überprüften Adressen nicht bei.

Die folgende Tabelle beschreibt die Optionen für die Matching-Alternativen:

Option	Beschreibung
Alle	Erkennt alle bekannten alternativen Straßen- und Platznamen. Standardoption.
Nur Archive	Erkennt nur historische Namen. Beispielsweise erkennt die Adressüberprüfung „Konstantinopel“ als frühere Bezeichnung von „Istanbul“.
Keine	Erkennt keine alternativen Straßen- oder Platznamen.
Nur Synonyme	Erhält nur Synonyme und Exonyme. Beispielsweise erkennt die Adressüberprüfung „Londres“ als Exonym von „London“.

Erweitertes Archiv abgleichen

Bestimmt, ob die Adressvalidierung einen eindeutigen Code des Zustellpunkts für eine veraltete japanische Adresse zurückgibt.

Die Adressreferenzdaten-Dateien für Japan enthalten neben den aktuellen Adressen für die entsprechenden Postfächer Daten für veraltete oder zurückgezogene Adressen. Bei Auswahl der Eigenschaft „Erweitertes Archiv abgleichen“ gibt die Adressvalidierung den Code des Zustellpunkts für die aktuelle Version jeder Adresse zurück. Die Adressvalidierung schreibt ebenfalls einen Wert in den Port „Erweiterter Elementergebnisstatus“, um anzugeben, dass die Eingabeadresse veraltet ist.

Um die aktuelle Adresse aus den Adressreferenzdaten abzurufen, geben Sie den Adresscode als ein Eingabeelement ein.

In der folgenden Tabelle werden die Optionen für „Erweitertes Archiv abgleichen“ beschrieben:

Option	Beschreibung
Aus	Wendet die Eigenschaft nicht an.
Ein	Gibt den Adresscode für die aktuelle Version einer veralteten japanischen Adresse zurück.

Die Eigenschaft „Erweitertes Archiv abgleichen“ verwendet zusätzliche Daten und Adresscode-Lookup-Daten für Japan. Um die Eigenschaft für die Adressvalidierung anzuwenden, konfigurieren Sie die Umwandlung zur Ausführung im Adresscode-Lookup-Modus.

Matching-Bereich

Bestimmt die Menge an Daten, die die Umwandlung während der Adressvalidierung mit den Adressreferenzdaten abgleicht.

Die folgende Tabelle beschreibt die Optionen für den Matching-Bereich:

Option	Beschreibung
Alle	Validiert alle ausgewählten Ports. Standardoption.
Zustellpunkt	Validiert Gebäude- und Gebäudeteiladressdaten zusätzlich zu den Daten, die von der Option „Straße“ validiert werden.
Ort	Validiert Provinz-, Orts- und Postleitzahldaten.
Straße	Validiert Straßenadressdaten zusätzlich zu den Daten, die von der Option „Ort“ validiert werden.

Maximale Ergebniszahl

Bestimmt die maximale Anzahl von Adressen, die die Adressvalidierung im Vorschlaglistenmodus zurückgeben kann.

Sie können ein Maximum im Bereich von 1 bis 100 festlegen. Standardwert ist 20.

Hinweis: Im Vorschlaglistenmodus wird eine Adresse mit Adressreferenzdaten verglichen und eine Liste mit Adressen ausgegeben, die mit der Eingabeadresse möglicherweise übereinstimmen. Wenn Sie eine Adresse im Vorschlaglistenmodus verifizieren, gibt die Adressüberprüfung die besten Treffer als Erstes aus.

Parameterverwendung

Sie können einen Parameter verwenden, um die maximale Anzahl von Adressen anzugeben. Geben Sie zum Festlegen der Parameterwerte eine Ganzzahl von 0 bis 100 ein.

Modus

Bestimmt den Typ der Adressanalyse, die die Umwandlung ausführt. Außerdem können Sie den Modus auf der Registerkarte **Allgemeine Einstellungen** der Umwandlung konfigurieren.

Parameterverwendung

Sie können einen Parameter verwenden, um den Analyse-Modus festzulegen. Wenn Sie den Parameter definieren, geben Sie den Modus als Standardwert ein. Der Wert verwendet Zeichen in Großbuchstaben und enthält keine Leerzeichen.

Die folgende Tabelle beschreibt die Modusmenüoption und die entsprechenden Parameterwerte, die Sie festlegen können:

Modus	Beschreibung	Parameterwert
Batch	Führt die Adressüberprüfung mithilfe von Referenzdaten durch. Standardoption.	BATCH
Zertifiziert	Führt die Adressvalidierung anhand von Referenzdaten zu Zertifizierungsstandards des Postdiensts durch.	CERTIFIED
Ländererkennung	Identifiziert Eingabeadressen, die keine Länderinformationen enthalten. Fügt Adressen Länderinformationen hinzu, wenn die Orts- oder Provinzdaten ein einzelnes Land identifizieren.	COUNTRYRECOGNITION
Parsen	Parst Daten in Adressfelder und führt keine Adressüberprüfung durch.	PARSE
Vorschlagsliste	Vergleicht eine Adresse mit Adressreferenzdaten und generiert eine Liste mit Adressen, die mit der Eingabeadresse möglicherweise übereinstimmen. Verwenden Sie den Modus „Vorschlagsliste“ mit Einstiegspunktanwendungen.	FASTCOMPLETION
Parameter zuweisen	Weisen Sie einen Parameter zu, der einen Adressvalidierungsmodus identifiziert.	Geben Sie einen Parameter-Wert aus der aktuellen Tabelle ein.

Optimierungslevel

Bestimmt, wie die Umwandlung mit den Eingabeadressdaten und Adressreferenzdaten übereinstimmt. Die Eigenschaft definiert den Typ der Übereinstimmung, der von der Umwandlung zwischen den

Eingabeadressdaten und den Referenzdaten gefunden werden muss, bevor der Adressdatensatz aktualisiert werden kann.

In der folgenden Tabelle werden die Optionen für die Optimierungsebene beschrieben:

Option	Beschreibung
Begrenzt	Die Umwandlung parst Gebäude- und Hausnummern anhand von Straßeninformationen vor dem Durchführen der Validierung. Andernfalls validiert die Umwandlung Eingabeadresselemente ausschließlich gemäß der Struktur des Eingabeports. Mit der Option „Begrenzt“ wird die schnellste Adressvalidierung durchgeführt. Es werden unter Umständen jedoch ungenauere Ergebnisse als bei anderen Optionen zurückgegeben.
Standard	Die Umwandlung analysiert mehrere Adressinformationstypen aus den Eingabedaten vor dem Durchführen der Validierung. Wenn Sie die Option „Standard“ auswählen, aktualisiert die Umwandlung eine Adresse, wenn den Referenzdaten mehrere Eingabewerte zugeordnet werden können. Voreingestellt ist „Standard“.
Weit	Die Umwandlung verwendet standardmäßige Parsing-Einstellungen und führt zusätzliche Parsing-Operationen für die Eingabedaten durch. Wenn Sie die Option „Weit“ auswählen, aktualisiert die Umwandlung eine Adresse, wenn den Referenzdaten mindestens ein Eingabewert zugeordnet werden kann. Die Option „Weit“ erhöht die Mapping-Laufzeiten.

Parameterverwendung

Sie können einen Parameter zur Angabe der Optimierungsebene verwenden. Geben Sie NARROW, STANDARD oder WIDE ein. Geben Sie den Parameterwert in Großbuchstaben ein.

Ausgabeformattyp

Beschreibt den häufigsten Informationstyp, der von der Umwandlung auf den Ausgabeport "Vollständige Adresse" oder "Formatierte Adresszeile" geschrieben wird. Wählen Sie die Option aus, die die Daten, die sie im Ausgabeport erwarten, am besten beschreibt.

Wählen Sie eine der folgenden Optionen aus:

- Alle
- Adresse
- Unternehmen
- Ansprechpartner
- Unternehmen/Ansprechpartner
  Die Adresse enthält Unternehmens- und Kontaktinformationen.
- Unternehmen/Abteilung
  Die Adresse enthält Unternehmens- und Abteilungsinformationen.

Voreingestellt ist „Alle“.

Ausgabeformat mit Land

Bestimmt, ob die Umwandlung Daten zur Länderidentifikation auf die Ausgabeports "Vollständige Adresse" oder "Formatierte Adresszeile" schreibt.

Die Option ist standardmäßig deaktiviert.

Bevorzugte Sprache

Legt die Sprachen fest, in denen eine Adresse von der Adress-Validiererumwandlung zurückgegeben wird, wenn die Referenzdatensätze Daten in mehreren Sprachen enthalten.

Ein Adressreferenzdatensatz gibt eine Standardsprache für jede gültige Adresse an. In bestimmten Ländern kann sich die Standardsprache von Region zu Region unterscheiden. Die Umwandlung gibt kanadische Adressen entsprechend der Provinz, die von der Adresse angegeben wird, beispielsweise in Englisch oder Französisch zurück. In den Adressreferenzdatensätzen für Kanada werden Adresselemente in beiden Sprachen gespeichert. Verwenden Sie die Eigenschaft „Bevorzugte Sprache“, um die Adressen in jedem Fall in der Nicht-Standardsprache zurückzugeben.

Hinweis: Ein Adressreferenzdatensatz enthält unter Umständen bestimmte, aber ansonsten keine Adresselemente in einer Nicht-Standardsprache. Wenn die Umwandlung kein Element in der von der Eigenschaft angegebenen Sprache findet, gibt die Umwandlung das Element in der Standardsprache zurück.

In der folgenden Tabelle werden die Optionen beschrieben, die Sie in der Eigenschaft „Bevorzugte Sprache“ auswählen können:

Option	Beschreibung
Alternative 1, Alternative 2, Alternative 3	Gibt jedes Adresselement unabhängig von der Standardsprache für das Land oder die Region in derselben Sprache zurück.
Datenbank	Gibt alle Adressen in der Sprache zurück, die von den Adressreferenzdaten angegeben wird. In den Adressreferenzdaten werden unter Umständen unterschiedliche Sprachen für Adressen in verschiedenen Regionen eines Landes angegeben. Datenbank ist die Standardoption.
Englisch	Gibt Orts- und Provinznamen in Englisch zurück, wenn die Referenzdaten die Daten in Englisch enthalten. Gibt die anderen Adresselemente in der Standardsprache der Region zurück, zu der die Adresse gehört.
Eingabe beibehalten	Gibt jede Adresse in der Sprache zurück, die von der Eingabeadresse verwendet wird. Die Umwandlung behält die Eingabesprache bei, wenn die Adressreferenzdaten die Adresse in der Eingabesprache enthalten.

Wenn die Eigenschaft eine Sprache angibt, die in den Adressreferenzdaten nicht verfügbar ist, gibt die Umwandlung die Adressen in der Standardsprache zurück.

Alternative Adressen in Belgien, Kanada und der Schweiz

Standardmäßig gibt die Adressvalidierung Adressen in der lokalen Sprache jeder Region in Belgien, Kanada und der Schweiz zurück. Sie können die Eigenschaft „Bevorzugte Sprache“ so konfigurieren, dass dieselbe Sprache bzw. dieselben Sprachen für alle Adressen innerhalb eines Landes verwendet werden. Verwenden Sie die Optionen „Alternative 1“, „Alternative 2“ und „Alternative 3“ zur Angabe der Sprache.

In der folgenden Tabelle werden die Optionen und Sprachen beschrieben, die für Adressen in Belgien festgelegt werden können:

Option	Sprache
Alternative 1	Flämisch
Alternative 2	Französisch
Alternative 3	Deutsch

In der folgenden Tabelle werden die Optionen und Sprachen beschrieben, die für Adressen in Kanada festgelegt werden können:

Option	Sprache
Alternative 1	Englisch
Alternative 2	Französisch
Alternative 3	Wird für kanadische Adressen nicht verwendet.

Wenn Sie die Option „Alternative 2“ für kanadische Adressen auswählen, werden die Straßenbeschreibungen und Provinznamen von der Adressvalidierung in Französisch zurückgegeben. Die Adressvalidierung gibt alle anderen Adresselemente in Englisch zurück.

In der folgenden Tabelle werden die Optionen und Sprachen beschrieben, die für Adressen in der Schweiz festgelegt werden können:

Option	Sprache
Alternative 1	Deutsch
Alternative 2	Französisch
Alternative 3	Italienisch

Wenn Sie die Option „Alternative 3“ für schweizer Adressen auswählen, werden die Ortsnamen von der Adressvalidierung in Italienisch zurückgegeben. Die Adressvalidierung gibt alle anderen Adresselemente in der Standardsprache für die Region zurück.

Hinweis: Wenn Sie für die Ausgabedaten eine bevorzugte Sprache und ein bevorzugtes Skript festgelegt haben, stellen Sie sicher, dass die ausgewählte Sprache und der ausgewählte Zeichencode kompatibel sind.

Bevorzugtes Skript

Bestimmt den Zeichensatz, der von der Adress-Validierungsumwandlung für Ausgabedaten verwendet wird.

Die Umwandlung kann eine Datenquelle verarbeiten, die Daten in mehreren Sprachen und Zeichensätzen enthält. Die Umwandlung wandelt alle Eingabedaten in den Unicode UCS-2-Zeichensatz um und verarbeitet die Daten im UCS-2-Format. Nach der Verarbeitung der Daten wandelt die Umwandlung die Daten in jedem Adressdatensatz in den Zeichensatz um, der in der Eigenschaft festgelegt wurde. Dieser Prozess wird als Transliteration bezeichnet. Standardmäßig verwendet die Eigenschaft den Zeichensatz, den die lokalen Adressreferenzdaten verwenden.

In der folgenden Tabelle werden die Optionen für die Eigenschaft beschrieben:

Option	Beschreibung
ASCII (Vereinfacht)	Gibt Adressdaten in ASCII-Zeichen zurück.
ASCII (Erweitert)	Gibt Adressdaten in ASCII-Zeichen zurück und erweitert Sonderzeichen. "Ö" wird zum Beispiel in "OE" transkribiert.
Datenbank	Gibt Adressdaten im Zeichensatz der Adressreferenzdaten zurück, die für die Adresse gültig sind. Standardwert ist „Datenbank“.
Lateinisch	Gibt Adressdaten in Latin I-Zeichen zurück.
Latin (Alt.)	Gibt Adressdaten in Latin I-Zeichen mit alternativer Transliteration zurück.
Post-Admin	Gibt Adressdaten je nach den Einstellungen des Postbeförderungsunternehmens im Zielland Latin I- oder ASCII-Zeichen zurück.
Post-Admin (Alt.)	Gibt Adressdaten in Latin I- oder ASCII-Zeichen zurück. Enthält alternative Zeichen, wenn das Postbeförderungsunternehmen im Zielland die Zeichen unterstützt.
Eingabe beibehalten	Gibt Adressdaten im Zeichensatz zurück, den die Eingabeadresse verwendet.

Die Transliteration kann die numerischen Darstellungen aller Zeichen in einem Zeichensatz verwenden, wenn Zeichen für die Verarbeitung umgewandelt werden. Die Transliteration kann Zeichen auch phonetisch umwandeln, wenn keine äquivalente numerische Darstellung eines Zeichens vorhanden ist. Wenn die Adressvalidator-Umwandlung ein Zeichen nicht zu UCS-2 zuordnen kann, wird das Zeichen in ein Leerzeichen umgewandelt.

Hinweis: Wenn Sie für die Ausgabedaten eine bevorzugte Sprache und ein bevorzugtes Skript festgelegt haben, stellen Sie sicher, dass die ausgewählte Sprache und die ausgewählte Zeichenkodierung kompatibel sind.

Zu erweiternde Bereiche

Bestimmt, wie die Adress-Validierungsumwandlung vorgeschlagene Adressen für eine Straßenadresse zurückgibt, die keine Hausnummer festlegt. Verwenden Sie die Eigenschaft, wenn die Umwandlung im Vorschlagslistenmodus ausgeführt wird.

Die Adress-Validierungsumwandlung liest eine teilweise oder unvollständige Straßenadresse im Vorschlagslistenmodus. Die Umwandlung vergleicht die Adresse mit den Adressreferenzdaten und gibt alle ähnlichen Adressen an den Endbenutzer zurück. Wenn die Eingabeadresse keine Hausnummer enthält, kann die Umwandlung eine oder mehrere Hausnummer-Vorschläge für die Straße zurückgeben. Die Eigenschaft "Zu erweiternde Bereiche" bestimmt, wie die Umwandlung die Adressen zurückgibt.

Die Umwandlung kann den Bereich gültiger Hausnummern in einer einzelnen Adresse zurückgeben oder eine separate Adresse für jede gültige Hausnummer zurückgeben. Die Umwandlung kann auch eine Adresse für jede Nummer im Bereich von der niedrigsten bis zur höchsten Hausnummer in der Straße zurückgeben.

In der folgenden Tabelle werden die Optionen für die Eigenschaft beschrieben:

Option	Beschreibung
Alle	Die Adressvalidierung gibt eine vorgeschlagene Adresse für jede Hausnummer im Bereich möglicher Hausnummern in der Straße zurück.
Keine	Die Adressvalidierung gibt eine einzelne Adresse zurück, die die niedrigsten und höchsten Hausnummern im gültigen Bereich für die Straße identifiziert.
Nur mit gültigen Einträgen	Die Adressvalidierung gibt eine vorgeschlagene Adresse für jede Hausnummer zurück, die die Adressreferenzdaten als zustellungsfähige Adresse erkennen.

Hinweis: Der Vorschlagslistenmodus kann andere Elemente in der Adresse verwenden, um den gültigen Bereich von Hausnummern festzulegen. Beispielsweise kann eine Postleitzahl den Häuserblock identifizieren, der den Briefkasten enthält. Die Adress-Validierungsumwandlung kann die Postleitzahl verwenden, um die niedrigsten und höchsten gültigen Hausnummern im Block zu identifizieren.

Wenn die Umwandlung keinen Hausnummernbereich in praktischen Grenzen bestimmen kann, kann die Anzahl der vorgeschlagenen Adressen auf eine unbrauchbare Größe ansteigen. Zum Einschränken der Anzahl von Adressen, die die Eigenschaft "Zu erweiternde Bereiche" generiert, legen Sie die Eigenschaft "Flexible Bereichserweiterung" auf "ON" fest.

Ungültige Adressen standardisieren

Gibt an, ob die Adressvalidierung die Datenwerte in einer nicht zustellbaren Adresse standardisiert. Die Eigenschaft gilt für Adressdatensätze, die einen Match-Code-Status im Bereich I1 bis I4 zurückgeben.

Wenn Sie die Daten standardisieren, erhöhen Sie die Wahrscheinlichkeit, dass ein nachgelagerter Datenprozess genaue Ergebnisse zurückgibt. Beispiel: Ein Mapping zur Duplikatsanalyse gibt unter Umständen einen höheren Übereinstimmungswert für zwei Datensätze zurück, die gemeinsame Adresselemente im selben Format aufweisen.

Die Adressvalidierung kann die folgenden Adresselemente standardisieren:

- Straßensuffixelemente, wie z. B. Straße und Boulevard.
- Vor-Richtungs- und Nach-Richtungselemente, wie z. B. Nord, Süd, Ost und West.
- Zustelldienstelemente, wie z. B. Postfach.
- Gebäudeteilelemente, wie z. B. Wohnung, Stockwerk und Appartement.
- Name des Bundesstaats bzw. der Provinz. Die Standardisierung gibt die abgekürzten Formen der Namen zurück.

In der folgenden Tabelle werden die Optionen für die Eigenschaft beschrieben:

Option	Beschreibung
Aus	Die Adressvalidierung korrigiert keine Datenfehler. Standardoption.
Ein	Die Adressvalidierung korrigiert Datenfehler.

Parameterverwendung

Sie können einen Parameter zuweisen, um die Standardisierungsrichtlinien für Datenfehler festzulegen. Geben Sie OFF oder ON als Parameterwert ein. Geben Sie den Wert in Großbuchstaben ein.

Tracingebene

Legt die im Protokoll einbezogene Menge der Details fest.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Zertifizierungsberichte

Sie können einen Bericht für einen Zusteller generieren, der die Adressvalidierungs-Operationen der Adressvalidator-Umwandlung zertifiziert.

Der Bericht überprüft, ob die Adressvalidierungs-Operationen den Standards des Zustellers entsprechen. Der USPS (United States Post Office) kann beispielsweise ein Softwaremodul zertifizieren, dass US-Adressen in ZIP+4-Codes validiert. Die Adress-Validierungsumwandlung validiert US-Adressen in ZIP+4-Codes und kann zusätzliche Daten bereitstellen, die beim Sortieren und Zustellen der Post nützlich sein können.

Die Adress-Validierungsumwandlung generiert Berichte für die folgenden Standards:

AMAS (Address Machine Approval System)

Die australische Post definiert den AMAS-Zertifizierungsstandard.

CASS (Coding Accuracy Support System)

Der USPS definiert den CASS-Zertifizierungsstandard.

SendRight

Die neuseeländische Post definiert den SendRight-Zertifizierungsstandard.

SERP (Software Evaluation and Recognition Program)

Die kanadische Post definiert den SERP-Zertifizierungsstandard.

Ein Zusteller kann Ihnen einen Preisnachlass gewähren, wenn Sie zertifizierte Software zum Validieren von Adressdaten verwenden.

Wenn Sie dem Zusteller Post bereitstellen, übermitteln Sie einen Bericht mit der Liste der Postadressen. Der Bericht enthält Daten zum Unternehmen. Sie geben die Daten beim Konfigurieren der Adressvalidator-Umwandlung ein. Die Umwandlung schreibt die Berichtsdatei in einen von Ihnen festgelegten Pfad.

Hinweis: Sie können die Adress-Validierungsumwandlung verwenden, um französische Adressdatensätze auf den Zertifizierungsstandard des französischen Nationalen Adressverwaltungsdiensts (SNA) zu zertifizieren. Generieren Sie keinen Bericht für die SNA-Adresszertifizierung.

Felder im AMAS-Bericht

Wenn Sie einen AMAS-Bericht konfigurieren, geben Sie Informationen über das Unternehmen an, das den zertifizierten Adressdatensatz an die australische Post übergibt. Speichern oder drucken Sie den Bericht und binden Sie den Bericht mit den Adressdatensätzen ein, die Sie an die australische Post übergeben.

Verwenden Sie die Ansicht **Berichte** zum Eingeben von Informationen.

Die folgende Tabelle beschreibt die von Ihnen eingegebenen Informationen:

Feld	Beschreibung
Name der Berichtsdatei	Name und Speicherort des Berichts, den die Adressvalidierung erstellt. Standardmäßig erstellt der Adressvalidierer den Bericht im `bin`-Verzeichnis des Datenintegrationsdienst-Computers. Geben Sie zum Schreiben der Berichtsdatei in ein anderes Verzeichnis auf dem Datenintegrationsdienst-Computer den Dateipfad und Dateinamen ein. Sie können einen vollständig qualifizierten Pfad oder einen relativen Pfad eingeben. Der relative Pfad verwendet das `bin`-Verzeichnis als Root-Verzeichnis. Das angegebene Verzeichnis muss vorhanden sein, bevor Sie das Adressvalidierungs-Mapping durchführen.
Name der Adressliste	Name des Adressdatensatzes, den Sie an die australische Post übergeben.
Listenprozessorname:	Name des Unternehmens, das den Adressdatensatz übergibt.
Name des Verwalters/ Eigentümers der Liste	Name des Verwalters oder Eigentümers der Adressdaten im Unternehmen.
Telefonnummer	Telefonnummer des Ansprechpartners in dem Unternehmen, das den Adressdatensatz übergibt.
Adresse	Adresse des Unternehmens, das den Adressdatensatz übergibt.

Verwandte Themen:

- "Definieren eines Zertifizierungsberichts" auf Seite 104

Felder im CASS-Bericht

Wenn Sie einen CASS-Bericht konfigurieren, geben Sie Informationen über das Unternehmen an, das den zertifizierten Adressdatensatz an den USPS übergibt. Speichern oder drucken Sie den Bericht und binden Sie den Bericht mit den Adressdatensätzen ein, die Sie an den USPS übergeben.

Verwenden Sie die Ansicht **Berichte** zum Eingeben von Informationen.

Die folgende Tabelle beschreibt die von Ihnen eingegebenen Informationen:

Feld	Beschreibung
Name der Berichtsdatei	Name und Speicherort des Berichts, den die Adressvalidierung erstellt. Standardmäßig erstellt der Adressvalidierer den Bericht im `bin`-Verzeichnis des Datenintegrationsdienst-Computers. Geben Sie zum Schreiben der Berichtsdatei in ein anderes Verzeichnis auf dem Datenintegrationsdienst-Computer den Dateipfad und Dateinamen ein. Sie können einen vollständig qualifizierten Pfad oder einen relativen Pfad eingeben. Der relative Pfad verwendet das `bin`-Verzeichnis als Root-Verzeichnis. Das angegebene Verzeichnis muss vorhanden sein, bevor Sie das Adressvalidierungs-Mapping durchführen.
Listenname/-ID	Name oder Identifikationsnummer der Adressliste, die von Ihnen an den Zusteller übermittelt wird.

Feld	Beschreibung
Listenprozessorname:	Name des Unternehmens, das die Adressvalidierung durchführt.
Name/Adresse:	Postname und -adresse des Unternehmens, dass die Adressvalidierung durchführt.

Verwandte Themen:

- "Definieren eines Zertifizierungsberichts" auf Seite 104

SendRight-Bericht

Geben Sie beim Konfigurieren eines SendRight-Berichts Informationen über das Unternehmen an, das den zertifizierten Adressdatensatz an die neuseeländische Post übergibt. Speichern oder drucken Sie den Bericht und binden Sie den Bericht mit den Adressdatensätzen ein, die Sie an die neuseeländische Post übergeben.

Verwenden Sie die Ansicht **Berichte** zum Eingeben von Informationen.

Die folgende Tabelle beschreibt die von Ihnen eingegebenen Informationen:

Feld	Beschreibung
Kundenname	Name des Unternehmens, das den Adressdatensatz übergibt.
NZP-Nummer des Kunden	Kontonummer der neuseeländische Post des Unternehmens, das den Adressdatensatz übergibt. Wenn ein Versanddienstleister die Datensätze im Auftrag des Unternehmens übergibt, geben Sie die Transportidentifikationsnummer (TPID) des Versanddienstleisters ein.
Kundendatenbank	Name der Datei, die den Adressdatensatz enthält. Die Adress-Validierungsumwandlung erstellt den Bericht in dem Speicherort, den Sie im Content-Managementdienst festgelegt haben. Verwenden Sie das Administrator-Tool, um den Speicherort festzulegen.
Kundenadresse	Adresse des Unternehmens, das den Adressdatensatz übergibt.

Verwandte Themen:

- "Definieren eines Zertifizierungsberichts" auf Seite 104

Felder im SERP-Bericht

Geben Sie beim Konfigurieren eines SERP-Berichts Informationen über das Unternehmen an, das den zertifizierten Adressdatensatz an die kanadische Post übergibt. Speichern oder drucken Sie den Bericht und binden Sie den Bericht mit den Adressdatensätzen ein, die Sie an den USPS übergeben.

Verwenden Sie die Ansicht **Berichte** zum Eingeben von Informationen.

Die folgende Tabelle beschreibt die von Ihnen eingegebenen Informationen:

Feld	Beschreibung
Name der Berichtsdatei	Name und Speicherort des Berichts, den die Adressvalidierung erstellt. Standardmäßig erstellt der Adressvalidierer den Bericht im `bin`-Verzeichnis des Datenintegrationsdienst-Computers. Geben Sie zum Schreiben der Berichtsdatei in ein anderes Verzeichnis auf dem Datenintegrationsdienst-Computer den Dateipfad und Dateinamen ein. Sie können einen vollständig qualifizierten Pfad oder einen relativen Pfad eingeben. Der relative Pfad verwendet das `bin`-Verzeichnis als Root-Verzeichnis. Das angegebene Verzeichnis muss vorhanden sein, bevor Sie das Adressvalidierungs-Mapping durchführen.
CPC-Kundennummer	Von der Canada Post Corporation ausgegebene Kundennummer für das Unternehmen, dass die Adressvalidierung durchführt.
Name/Adresse des Kunden:	Name und Adresse des Unternehmens, das die Adressvalidierung durchführt.

Verwandte Themen:

- "Definieren eines Zertifizierungsberichts" auf Seite 104

Konfigurieren einer Adressvalidator-Umwandlung

Verwenden Sie eine Adressvalidator-Umwandlung zum Validieren und Verbessern der Datenqualität Ihrer Postadressen.

Die Adressvalidator-Umwandlung liest Adressreferenzdaten. Stellen Sie sicher, dass das Developer Tool auf die benötigten Adressreferenzdaten-Dateien zugreifen kann.

1. Öffnen Sie die Umwandlung.
2. Klicken Sie auf die Ansicht **Allgemeine Einstellungen** und konfigurieren Sie die allgemeinen Eigenschaften.
3. Klicken Sie auf die Ansicht **Vorlagen**, um Eingabe- und Ausgabeports hinzuzufügen.
4. Klicken Sie auf die Ansicht **Berichte**, um Berichte für die Adresszertifizierung des Postdienstes zu erzeugen.
5. Klicken Sie auf die Ansicht **Erweitert**, um erweiterte Eigenschaften für die Adressvalidierung zu konfigurieren.
6. Verbinden Sie die Eingabe- und Ausgabeports.

   Hinweis: Verbinden Sie Eingabeports, die von der Adressumwandlung nicht auf die **Pass-Through**-Eingabeportgruppe validiert werden sollen.

Hinzufügen von Ports zur Adressvalidator-Umwandlung

Verwenden Sie die Ansicht **Vorlagen**, um der Adressvalidator-Umwandlung Ports hinzuzufügen.

1. Klicken Sie auf die Ansicht **Vorlagen**.
2. Erweitern Sie eine Vorlage.
   - Wählen Sie die Vorlage **Basismodell** aus, der allgemeine Adressfelder hinzugefügt werden sollen.
   - Wählen Sie die Vorlage **Erweitertes Modell** aus, der spezielle Adressfelder hinzugefügt werden sollen.
3. Erweitern Sie die Eingabeportgruppe, die dem Format der Eingabedaten entspricht. Die Eingabeportgruppen sind **Diskret**, **Mehrzeilig** und **Hybrid**.
4. Wählen Sie Eingabeports aus.

   Tipp: Halten Sie die Taste `CTRL` gedrückt, um mehrere Ports auszuwählen.
5. Klicken Sie mit der rechten Maustaste auf die Ports und wählen Sie **Port zur Umwandlung hinzufügen** aus.
6. Erweitern Sie die Ausgabeportgruppe, die die benötigten Felder enthält.
7. Klicken Sie mit der rechten Maustaste auf die Ports und wählen Sie **Port zur Umwandlung hinzufügen** aus.
8. Zum Hinzufügen von Pass-Through-Ports für Spalten, die nicht validiert werden sollen, klicken Sie auf die Ansicht **Ports**, wählen die Eingabeportgruppe **Pass-Through** aus und klicken auf **Neu**.

Erstellen von benutzerdefinierten Vorlagen

Erstellen Sie Vorlagen, um die erneut zu verwendenden Adressports zu gruppieren.

Erstellen Sie benutzerdefinierte Vorlagen durch Auswahl von Ports aus den Vorlagen "Basis" und "Erweitert". Sie können die benutzerdefinierten Vorlagen beim Erstellen nachfolgender Adressvalidator-Umwandlungen auswählen.

Hinweis: Vorlagen sind keine Repository-Objekte. Vorlagen sind auf dem Computer gespeichert, auf dem sie erstellt wurden.

1. Wählen Sie die Ansicht **Vorlagen** aus.
2. Klicken Sie auf **Neu**.
3. Geben Sie einen Namen für die Vorlage ein.
4. Erweitern Sie die Vorlage **Basismodell** oder **Erweitertes Modell** und wählen Sie die benötigten Ports aus.
5. Klicken Sie auf **OK**.

Definieren von Adressvalidator-Modellen

Adressvalidator-Modelle definieren die standardmäßigen Eingabe- und Ausgabeports für Adressvalidator-Umwandlungen.

Adressvalidator-Umwandlungen enthalten keine standardmäßigen Eingabe- und Ausgabeports. Sie können jedoch ein Modell definieren, um die Eingabe- und Ausgabeports anzugeben, die in Adressvalidator-Umwandlungen verwendet werden.

Hinweis: Modelle sind keine Repository-Objekte. Modelle sind auf dem Computer gespeichert, auf dem sie erstellt wurden.

Führen Sie zum Definieren eines Adressvalidator-Modells die folgenden Schritte aus:

1. Wählen Sie die Ansicht **Vorlagen** aus.
2. Erweitern Sie die Vorlage **Basismodell** oder **Erweitertes Modell** und wählen Sie die benötigten Ports aus.
3. Wählen Sie **Standard-AV-Modell mit ausgewählten Ports erstellen** aus.
4. Wählen Sie **Standard-AV-Modell löschen** zum Zurücksetzen des Modells und Entfernen aller Ports aus.

Definieren eines Zertifizierungsberichts

Wenn Sie in der Adressvalidator-Umwandlung einen Zertifizierungsbericht definieren, können Sie Optionen in den Ansichten **Allgemeine Einstellungen** und **Berichte** konfigurieren.

1. Setzen Sie in der Ansicht **Allgemeine Einstellungen** die Option **Modus** auf `Zertifiziert`.
2. Wählen Sie in der Ansicht **Berichte** den Typ der zu erzeugenden Berichte aus. Sie können die folgenden Berichtstypen auswählen:

Option	Beschreibung
AMAS-Bericht	Enthält Informationen, die die australische Post zum Verarbeiten des Datensatzes benötigt.
CASS-Bericht	Enthält Informationen, die der USPS zum Verarbeiten des Datensatzes benötigt.
SendRight-Bericht	Enthält Informationen, die die neuseeländische Post zum Verarbeiten des Datensatzes benötigt.
SERP-Bericht	Enthält Informationen, die die kanadische Post zum Verarbeiten des Datensatzes benötigt.

3. Geben Sie für jeden ausgewählten Berichtstyp die Berichtsdetails ein.

Übermitteln Sie die Berichtsdatei mit der Liste der Adressdatensätze, die mit der Adressvalidator-Umwandlung validiert wurden, an den Zusteller.

Verwandte Themen:

- "Felder im AMAS-Bericht" auf Seite 99
- "Felder im CASS-Bericht" auf Seite 100
- "SendRight-Bericht" auf Seite 101
- "Felder im SERP-Bericht" auf Seite 101

KAPITEL 3

Aggregator-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Aggregator-Umwandlung - Übersicht, 105
- Aggregator-Umwandlungen in dynamischen Mappings, 106
- Entwickeln einer Aggregator-Umwandlung, 106
- Aggregator-Umwandlung - Ports, 106
- Aggregat-Ausdrücke, 107
- Nach Ports gruppieren, 109
- Aggregator-Caches, 112
- Sortierte Eingabe für eine Aggregator-Umwandlung, 112
- Aggregatorumwandlung – Erweiterte Eigenschaften, 114
- Erstellen einer wiederverwendbaren Aggregator-Umwandlung, 114
- Erstellen einer Nicht-wiederverwendbaren Aggregator-Umwandlung, 115
- Tipps für Aggregator-Umwandlungen, 115
- Fehlerbehebung bei Aggregator-Umwandlungen, 116

Aggregator-Umwandlung - Übersicht

Konfigurieren Sie eine Aggregator-Umwandlung, um Aggregat-Berechnungen durchzuführen, wie z. B. Durchschnittswerte und Summen für Gruppen von Daten. Sie können eine Aggregator-Umwandlung zum Entfernen von doppelten Zeilen verwenden. Die Aggregator-Umwandlung ist eine aktive Umwandlung.

Die Aggregator-Umwandlung unterscheidet sich von der Ausdrucksumwandlung, da Sie die Aggregator-Umwandlung zur Durchführung von Berechnungen für Gruppen von Daten konfigurieren können. Eine Ausdrucksumwandlung gibt ein Ergebnis auf Zeilenbasis zurück.

Beispielsweise können Sie das durchschnittliche Gehalt für Mitarbeiter jeder Abteilung eines Unternehmens berechnen. Konfigurieren Sie eine Gruppe nach Abteilungsnummer. Konfigurieren Sie einen Ausdruck, um das durchschnittliche Gehalt zu berechnen und das Ergebnis für jede eindeutige Abteilungsnummer zurückzugeben.

Verwenden Sie die Umwandlungssprache zum Erstellen von Aggregat-Ausdrücken.

Der Datenintegrationsdienst führt Aggregat-Berechnungen beim Lesen und Speichern von Daten in einem Aggregat-Cache durch. Sie können die Eingabedaten sortieren, um die Leistung zu steigern. Der Datenintegrationsdienst erstellt keinen Cache, wenn Sie die Eingabedaten sortieren.

Aggregator-Umwandlungen in dynamischen Mappings

Sie können eine Aggregator-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports verweisen.

Sie können die folgenden Aufgaben durchführen, um eine Aggregator-Umwandlung in einem dynamischen Mapping zu konfigurieren:

Verweisen Sie auf einen dynamischen Port oder einen generierten Port als Spalte „Gruppieren nach".

Sie können dynamische Ports bzw. generierte Ports als Spalten „Gruppieren nach" einbeziehen. Wenn Sie einen dynamischen Port als eine Spalte „Gruppieren nach" festlegen, geben Sie an, dass alle generierten Ports im dynamischen Port Spalten „Gruppieren nach" sind. Aus diesem Grund können Sie keinen generierten Port als Spalte „Gruppieren nach" festlegen, wenn Sie den übergeordneten dynamischen Port als eine Spalte „Gruppieren nach" festgelegt haben. Wenn Sie auf einen generierten Port verweisen und der generierte Port zur Laufzeit nicht vorhanden ist, schlägt das Mapping fehl. Sie können die Spalten „Gruppieren nach" parametrisieren, um anzugeben, welche Spalten zur Laufzeit gruppiert werden sollen.

Verweisen Sie im Aggregat-Ausdruck auf einen generierten Port.

Sie können einen generierten Port in den Aggregat-Ausdruck einbeziehen. Wenn der Port zur Laufzeit nicht vorhanden ist, schlägt das Mapping fehl. Sie können im Aggregat-Ausdruck nicht auf einen dynamischen Port verweisen.

Erstellen Sie den Aggregat-Ausdruck in einem Ausgabeport.

Sie können keinen Aggregat-Ausdruck in einem generierten Port oder einem dynamischen Port erstellen. Der Aggregat-Ausdruck darf kein dynamischer Ausdruck sein.

Parametrisieren Sie Werte im Aggregat-Ausdruck.

Sie können Parameter in den Aggregat-Ausdruck einbeziehen. Sie können jedoch keinen Ausdrucksparameter oder Portparameter im Aggregat-Ausdruck verwenden.

Entwickeln einer Aggregator-Umwandlung

Wenn Sie eine Aggregator-Umwandlung erstellen, definieren Sie den Ausdruck zum Ausführen in jeder Zeile. Definieren Sie eine Liste zum Gruppieren nach Ports, von der Ergebnisse zurückgegeben werden sollen.

Führen Sie zum Erstellen einer Aggregator-Umwandlung die folgenden Schritte aus:

1. Definieren Sie die Umwandlung und erstellen Sie die Ports.
2. Konfigurieren Sie einen Aggregat-Ausdruck auf einem variablen Port oder Ausgabeport.
3. Definieren Sie eine Gruppe von Ports, von denen Aggregat-Ergebnisse zurückgegeben werden sollen.

Aggregator-Umwandlung - Ports

Eine Aggregator-Umwandlung verfügt über mehrere Porttypen, die Sie zum Konfigurieren von Gruppen und zum Zusammenführen von Ausdrücken verwenden können.

Eine Ansicht „Aggregator-Umwandlung – Ports" enthält die folgenden Felder:

Name

Name des Ports.

Typ

Der Portdatentyp.

Gesamtstellenanzahl

Länge des Felds.

Größenordnung

Die Anzahl der Positionen rechts von der Dezimalzahl für numerische Daten.

Eingabe

Gibt an, dass die Daten aus einer vorgelagerten Umwandlung stammen.

Ausgabe

Gibt an, dass der Port den Wert eines Ausdrucks zurückgibt. Der Ausdruck kann Nicht-Aggregat-Ausdrücke und Bedingungsklauseln enthalten. Sie können mehrere Aggregat-Ausgabeports erstellen.

Pass-Through

Gibt an, dass der Port ein Eingabe-/Ausgabeport ist, der die Daten unverändert zurückgibt.

Variable

Gibt an, dass der Port Werte oder Berechnungen für die Verwendung in einem Ausdruck speichern kann. Variable Ports können keine Eingabe- oder Ausgabeports sein. Sie übergeben Daten nur innerhalb der Umwandlung.

Ausdruck

Der Ausdruck zum Zusammenführen der Zeilen oder Zeilengruppen.

Standardwert

Der Standardwert für einen Port mit ungültigen Werten oder Nullwerten.

Eingaberegeln

Ein Satz von Regeln, welche die in die Umwandlung einzubeziehenden bzw. auszuschließenden Ports basierend auf Portnamen oder Datentypen filtern. Konfigurieren Sie Eingaberegeln, wenn Sie dynamische Ports definieren.

Aggregat-Ausdrücke

Konfigurieren Sie Aggregat-Ausdrücke in den variablen Ports oder in den Ausgabeports einer Aggregator-Umwandlung. Sie können keinen Ausdruck eingeben, wenn es sich bei dem Port um einen Eingabeport, einen Pass-Through-Port oder einen dynamischen Port handelt.

Ein Aggregat-Ausdruck kann Bedingungsklauseln sowie Funktionen enthalten, die Nicht-Aggregatfunktionen sind. Die Aggregatfunktion kann außerdem eine in einer anderen Aggregatfunktion geschachtelte Aggregatfunktion enthalten, wie z. B.:

```
MAX( COUNT( ITEM ))
```

Das Ergebnis eines Aggregat-Ausdrucks variiert basierend auf den Gruppierungsports in der Umwandlung. Mit dem folgenden Aggregat-Ausdruck kann man beispielsweise die Gesamtmenge der verkauften Artikel ermitteln:

```
SUM( QUANTITY )
```

Wenn Sie jedoch denselben Ausdruck verwenden und nach dem ITEM-Port gruppieren, gibt der Datenintegrationsdienst die Gesamtmenge nach Artikel zurück.

Sie können einen Aggregat-Ausdruck in allen Ausgabeports erstellen und mehrere Aggregat-Ports in einer Umwandlung verwenden.

Aggregatfunktionen

Konfigurieren Sie Aggregatfunktionen innerhalb einer Aggregator-Umwandlung. Sie können eine Aggregatfunktion in eine andere Aggregatfunktion einbetten.

Die Umwandlungssprache enthält die folgenden Aggregatfunktionen:

- AVG
- COUNT
- FIRST
- LAST
- MAX
- MEDIAN
- MIN
- PERCENTILE
- STDDEV
- SUM
- VARIANCE

Müssen Sie diese Funktionen in einem Ausdruck innerhalb einer Aggregator-Umwandlung verwenden.

Eingebettete Aggregatfunktionen

Sie können mehrere einstufige oder eingebettete Funktionen in verschiedenen Ausgabeports in einer Aggregator-Umwandlung verwenden.

Sie können nicht sowohl einstufige als auch eingebettete Funktionen in einer Aggregator-Umwandlung verwenden. Wenn eine Aggregator-Umwandlung daher eine einstufige Funktion in einem beliebigen Ausgabeport enthält, können Sie keine eingebettete Funktion in anderen Ports in dieser Umwandlung verwenden. Wenn Sie einstufige oder eingebettete Funktionen in dieselbe Aggregator-Umwandlung einbeziehen, markiert das Developer Tool das Mapping oder Mapplet als ungültig. Wenn Sie sowohl einstufige als auch eingebettete Funktionen erstellen müssen, erstellen Sie separate Aggregator-Umwandlungen.

Bedingungsklauseln in Aggregat-Ausdrücken

Verwenden Sie Bedingungsklauseln im Aggregat-Ausdruck, um die Anzahl der Zeilen in der Aggregation zu verringern. Die Bedingungsklausel kann eine Klausel sein, die sich als TRUE oder FALSE herausstellt.

Verwenden Sie zum Beispiel den folgenden Ausdruck, um die Gesamtprovision der Mitarbeiter zu berechnen, die ihre Quartalsquote überschritten haben:

```
SUM( COMMISSION, COMMISSION > QUOTA )
```

Nach Ports gruppieren

Sie können Gruppen von Zeilen für das Aggregieren definieren, anstatt eine Aggregation für alle Eingabedaten durchzuführen. Beispielsweise können Sie den Gesamtumsatz des Unternehmens berechnen oder Sie können den nach Region gruppierten Gesamtumsatz ermitteln.

Definieren Sie eine Gruppe für den Aggregat-Ausdruck und wählen Sie die entsprechenden Eingabe-, Eingabe-/Ausgabe-, Ausgabe- und Variablenports in der Aggregator-Umwandlung aus. Sie können mehrere Portgruppierungen zum Erstellen einer neuen Gruppe für jede eindeutige Kombination auswählen. Der Datenintegrationsdienst führt dann die definierte Aggregation für jede Gruppe durch.

Wenn Sie die Werte gruppieren, produziert der Datenintegrationsdienst eine Zeile für jede Gruppe. Wenn Sie keine Werte gruppieren, gibt der Datenintegrationsdienst eine Zeile für alle Eingabezeilen zurück. Der Datenintegrationsdienst gibt die letzte Zeile jeder Gruppe mit dem Ergebnis der Aggregation zurück. Sie können eine bestimmte Zeile angeben, die zurückgegeben werden soll. Wenn Sie beispielsweise die Aggregatorfunktion FIRST verwenden, gibt der Datenintegrationsdienst die erste Zeile zurück.

Wenn Sie mehrere Gruppierungsports in der Aggregator-Umwandlung auswählen, verwendet der Datenintegrationsdienst die Portreihenfolge, um die Reihenfolge der Gruppierung festzulegen. Die Gruppenreihenfolge kann sich auf die Ergebnisse auswirken. Sortieren Sie die Gruppe nach Ports, um eine ordnungsgemäße Gruppierung zu gewährleisten. Sie können die Portreihenfolge ändern, nachdem Sie die Ports in der Gruppe ausgewählt haben.

Beispielsweise können Sie einen Ausgabeport namens Price_Out erstellen. Der Ausdruck für Price_Out ist SUM (Qty * Price). Sie definieren Store_ID und Item als die Gruppierungsports. Die Umwandlung gibt den Gesamtpreis für jedes Element nach Lager zurück.

Die Eingabezeilen können die folgenden Daten enthalten:

Store_ID	Item	Qty	Price
101	battery	3	2.99
101	battery	1	3.19
101	battery	2	2.59
101	AAA	2	2.45
201	battery	1	1.99
201	battery	4	1.59
301	battery	1	2.45

Der Datenintegrationsdienst führt die Aggregat-Berechnung für die folgenden eindeutigen Gruppen durch:

Store_Id	Item
101	battery
101	AAA

Store_Id	Item
201	battery
301	battery

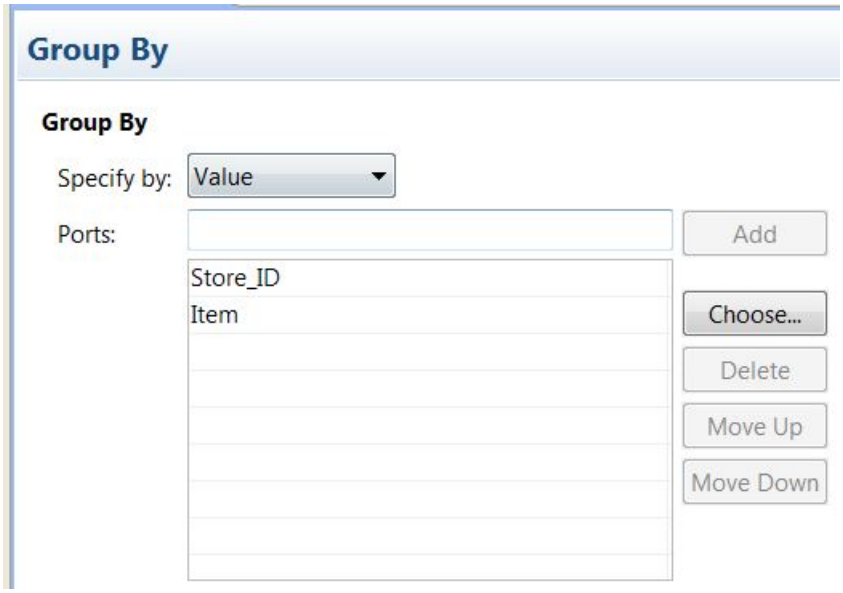
Der Datenintegrationsdienst gibt Store_ID, Item, Qty und Price aus der letzten Zeile mit der Summe von (Price * Qty) für jedes Element nach Lager zurück:

Store_ID	Item	Qty	Price	Price_Out
101	battery	2	2.59	17.34
101	AAA	2	2.45	4.90
201	battery	4	1.59	8.35
301	battery	1	2.45	2.45

Ports „Gruppieren nach" konfigurieren

Definieren Sie auf der Registerkarte **Gruppieren nach** in der Ansicht **Eigenschaften** der Umwandlung die Gruppierungsports.

Die folgende Abbildung zeigt die Registerkarte „Gruppieren nach":



Die Registerkarte „Gruppieren nach" enthält die folgenden Optionen:

Angeben nach

Wählen Sie **Wert** oder **Parameter** aus. Wählen Sie **Wert** für die Verwendung von Portnamen aus. Wählen Sie **Parameter** für die Verwendung eines Parameters für die Portliste aus.

Hinzufügen

Akzeptiert einen Portnamen, den Sie manuell eingeben. Sie müssen einen gültigen Namen eingeben, bevor Sie auf **Hinzufügen** klicken.

Auswählen

Klicken Sie auf **Auswählen**, um Ports auszuwählen, die zur Gruppe hinzugefügt werden sollen. Das Developer Tool enthält eine Liste mit Ports aus der Umwandlung, die zur Auswahl stehen.

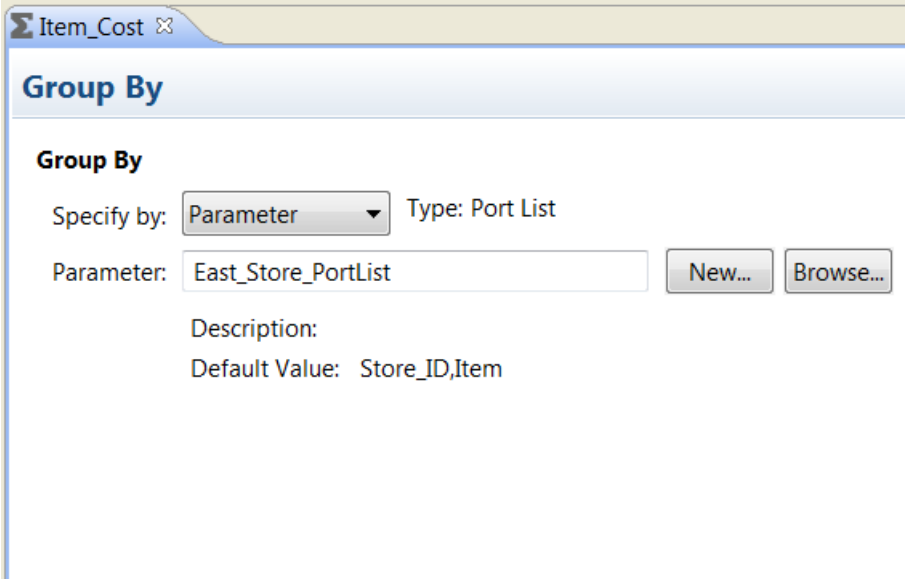
Nach oben verschieben und nach unten verschieben

Sie können die Reihenfolge der Ports in der Gruppe ändern. Wählen Sie den Portnamen aus und klicken Sie anschließend auf eine der Schaltflächen zum Verschieben, um den Portnamen in der Sortierreihenfolge nach oben oder nach unten zu verschieben.

Parameter „Gruppieren nach”

Sie können einen Portlistenparameter konfigurieren, der mindestens einen in die Gruppe einzuschließenden Port enthält. Erstellen Sie einen Portlistenparameter, indem Sie Ports aus einer Liste mit den Ports in der Umwandlung auswählen.

Die folgende Abbildung zeigt die Registerkarte **Gruppieren nach**, wenn Sie einen Parameter zur Angabe der Ports in der Gruppe verwenden:



Sie können nach einem Portlistenparameter suchen oder auf **Neu** klicken, um einen Portlistenparameter zu erstellen. Wenn Sie einen Portlistenparameter erstellen möchten, können Sie die Ports aus einer Liste mit den Ports in der Umwandlung auswählen.

Standardwerte von "Nach Ports gruppieren"

Der Datenintegrationsdienst erstellt keine Gruppe, wenn ein Port „Gruppieren nach” Nullwerte enthält. Sie können einen Standardwert für jeden Port in der Gruppe definieren, um alle Null-Eingabewerte zu ersetzen. Anschließend kann der Datenintegrationsdienst die Zeilen in die Gesamtwerte der Aggregation einfügen.

Nicht-Aggregat-Ausdrücke

Verwenden Sie Nicht-Aggregat-Ausdrücke in Portgruppierungen, um Gruppen zu ändern oder zu ersetzen.

Wenn Sie beispielsweise "AAA battery" vor der Gruppierung ersetzen möchten, können Sei einen nach Ausgabe gruppierten Port namens CORRECTED_ITEM mit dem folgenden Ausdruck erstellen:

```
IIF( ITEM = 'AAA battery', battery, ITEM )
```

Aggregator-Caches

Wenn Sie ein Mapping ausführen, das eine Aggregatorumwandlung verwendet, erstellt der Datenintegrationsdienst einen Index- und Daten-Cache im Speicher, um die Umwandlung auszuführen. Wenn der Datenintegrationsdienst mehr Speicherplatz benötigt als im Speicher-Cache vorhanden ist, werden Überlaufdaten in Cachedateien gespeichert.

Der Datenintegrationsdienst erstellt die folgenden Caches für die Aggregatorumwandlung:

- Index-Cache. Speichert die Gruppenwerte so, wie sie in „Nach Ports gruppieren” konfiguriert wurden.
- Daten-Cache. Speichert Berechnungen basierend auf „Nach Ports gruppieren”.

Der Datenintegrationsdienst verwendet keinen Cache-Arbeitsspeicher, um eine Aggregatorumwandlung mit sortierten Ports auszuführen. Sie brauchen keinen Cache-Arbeitsspeicher für Aggregator-Umwandlungen zu konfigurieren, die sortierte Ports verwenden.

Sortierte Eingabe für eine Aggregator-Umwandlung

Sie können die Leistung der Aggregator-Umwandlung mithilfe der Option "Sortierte Eingabe" steigern.

Wenn Sie die sortierte Eingabe verwenden, geht der Datenintegrationsdienst davon aus, dass alle Daten nach Gruppe sortiert werden. Der Dienst führt Aggregatberechnungen durch, während er die Zeilen einer Gruppe liest. Falls erforderlich speichert der Datenintegrationsdienst Gruppeninformationen im Speicher. Zum Verwenden der Option "Sortierte Eingabe" müssen Sie sortierte Daten an die Aggregator-Umwandlung übergeben.

Wenn Sie die sortierte Eingabe nicht verwenden, führt der Datenintegrationsdienst Aggregatberechnungen beim Lesen durch. Da die Daten nicht sortiert sind, speichert der Datenintegrationsdienst Daten für jede Gruppe bis zum Lesen der gesamten Quelle, um sicherzustellen, dass alle Aggregatberechnungen genau sind.

Eine Aggregator-Umwandlung verfügt beispielsweise bei ausgewählter Option "Sortierte Eingabe" über die STORE_ID- und ITEM-Gruppe nach Ports. Wenn Sie die folgenden Daten über den Aggregator übergeben, führt der Datenintegrationsdienst eine Aggregation für die drei Zeilen in der 101/Batterie-Gruppe durch, wenn die Gruppe 201/Batterie gefunden wird:

STORE_ID	ITEM	QTY	PRICE
101	'battery'	3	2.99
101	'battery'	1	3.19
101	'battery'	2	2.59
201	'battery'	4	1.59
201	'battery'	1	1.99

Wenn Sie sortierte Eingabe verwenden und die Daten nicht vorsortieren, kann der Datenintegrationsdienst das Mapping nicht ausführen.

Bedingungen für sortierte Eingabe

Bestimmte Bedingungen können Sie daran hindern, sortierte Eingabe zu verwenden.

Sie können die sortierte Eingabe nicht verwenden, wenn eine der folgenden Bedingungen zutrifft:

- Der Aggregat-Ausdruck enthält verschachtelte Aggregatfunktionen.
- Quelldaten sind datengesteuert.

Wenn eine dieser Bedingungen zutrifft, verarbeitet der Datenintegrationsdienst die Umwandlung so, als wäre die sortierte Eingabe nicht ausgewählt worden.

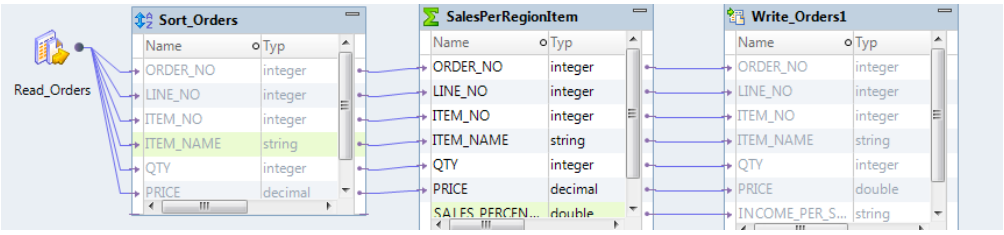
Sortieren von Daten in einer Aggregator-Umwandlung

Zum Verwenden der sortierten Eingabe übergeben Sie sortierte Daten über eine Aggregator-Umwandlung.

Sie müssen Daten der Aggregator-Gruppe in der Reihenfolge nach Ports sortieren, in der sie in der Aggregator-Umwandlung angezeigt werden.

Verwenden Sie für die relationale und Einfachdateieingabe die Sortiererumwandlung, um Daten im Mapping vor der Übergabe an die Aggregatorumwandlung zu sortieren. Sie können die Sortiererumwandlung an einer beliebigen Stelle vor dem Aggregator im Mapping platzieren, wenn keine Umwandlung die Reihenfolge der sortierten Daten ändert. Die Gruppierung nach Spalten in der Aggregatorumwandlung muss dieselbe Reihenfolge aufweisen, in der sie in der Sortiererumwandlung angezeigt wird.

Das folgende Mapping zeigt eine Sortierer-Umwandlung an, die zum Sortieren der Daten in aufsteigender Reihenfolge nach ITEM_NO konfiguriert ist.



Die Sortiererumwandlung sortiert die Daten wie folgt:

ITEM_NO	ITEM_NAME	QTY	PRICE
345	Soup	4	2.95
345	Soup	1	2.95
345	Soup	2	3.25
546	Cereal	1	4.49
546	Cereal	2	5.25

Mit der sortierten Eingabe gibt die Aggregatorumwandlung die folgenden Ergebnisse zurück:

ITEM_NAME	QTY	PRICE	INCOME_PER_ITEM
Cereal	2	5.25	14.99
Soup	2	3.25	21.25

Aggregatorumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Datenintegrationsdienst Daten für die Aggregator-Umwandlung verarbeitet.

Konfigurieren Sie die folgenden erweiterten Eigenschaften für eine Aggregatorumwandlung:

Cache-Verzeichnis

Verzeichnis, in dem der Datenintegrationsdienst die Index-Cache- und Daten-Cachedateien erstellt. Stellen Sie sicher, dass das Verzeichnis vorhanden ist und genügend Speicherplatz für die Cachedateien enthält.

Geben Sie mehrere Verzeichnisse durch Semikolon getrennt ein, um die Leistung während der Cache-Partitionierung zu erhöhen. Die Cache-Partitionierung erstellt für jede Partition, die die Umwandlung verarbeitet, einen separaten Cache.

Standardwert ist der CacheDir-Systemparameter. Sie können für diese Eigenschaft einen anderen Systemparameter oder einen benutzerdefinierten Parameter konfigurieren.

Daten-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Daten-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto” auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto”.

Index-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Index-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto” auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto”.

Sortierte Eingabe

Gibt an, dass die Eingabedaten nach Gruppen vorsortiert sind. Wählen Sie diese Option nur aus, wenn das Mapping sortierte Daten an die Aggregator-Umwandlung übergibt.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Verwandte Themen:

- “Cachegröße” auf Seite 662

Erstellen einer wiederverwendbaren Aggregator-Umwandlung

Erstellen Sie eine wiederverwendbare Aggregator-Umwandlung zur Verwendung in mehreren Mappings oder Mapplets.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.

2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung.**

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die Aggregator-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Geben Sie einen Namen für die Umwandlung ein.
6. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.
7. Klicken Sie auf die Schaltfläche **Neu**, um der Umwandlung einen Port hinzuzufügen.
8. Bearbeiten Sie den Port, um den Namen, den Datentyp und die Präzision zu bearbeiten.
9. Bestimmen den Typ der jeden Port: Eingabe, Ausgabe, Pass-Through oder Variable.
10. Klicken Sie auf das Feld „Ausdruck“, um Aggregatausdrücke für Ausgabeports zu konfigurieren. Sie können Ports und Parameter zum Definieren des Aggregatausdrucks auswählen.
11. Klicken Sie auf die Ansicht **Erweitert** und bearbeiten Sie die Umwandlungseigenschaften.

Erstellen einer Nicht-wiederverwendbaren Aggregator-Umwandlung

Erstellen Sie eine nicht wiederverwendbare Aggregator-Umwandlung in einem Mapping oder Mapplet.

1. Ziehen Sie in einem Mapping oder Mapplet eine Aggregator-Umwandlung aus der Umwandlungspalette auf den Editor.

   Die Umwandlung wird im Editor angezeigt.
2. Bearbeiten Sie in der Ansicht **Eigenschaften** den Namen und die Beschreibung der Umwandlung.
3. Klicken Sie auf der Registerkarte **Ports** auf die Schaltfläche **Neu** und fügen Sie der Umwandlung Ports hinzu.
4. Bearbeiten Sie die Ports, um den Namen, den Datentyp und die Genauigkeit festzulegen.
5. Bestimmen Sie den Typ jedes Ports: Eingabe, Ausgabe, Pass-Through oder Variable.
6. Konfigurieren Sie Aggregat-Ausdrücke für Ausgabeports.
7. Bearbeiten Sie in der Ansicht **Erweitert** die Umwandlungseigenschaften.

Tipps für Aggregator-Umwandlungen

Sie können Tipps zur Verwendung der Aggregator-Umwandlung effizienter nutzen.

Verwenden Sie die sortierte Eingabe, um die Verwendung von Aggregat-Caches zu reduzieren.

Mit der sortierten Eingabe reduzieren Sie die Menge der während der Mapping-Ausführung zwischengespeicherten Daten und verbessern die Leistung. Verwenden Sie diese Option mit der Sortier-Umwandlung, um sortierte Daten an die Aggregator-Umwandlung zu übergeben.

Beschränken Sie verbundene Eingabe-/Ausgabe- oder Ausgabeports.

Beschränken Sie die Anzahl der verbundenen Eingabe-/Ausgabe- oder Ausgabeports, um die Menge der Daten zu reduzieren, die die Aggregator-Umwandlung im Daten-Cache speichert.

Filtern Sie die Daten, bevor Sie sie aggregieren.

Wenn Sie eine Filterumwandlung im Mapping verwenden, platzieren Sie die Umwandlung vor die Aggregator-Umwandlung, um unnötige Aggregation zu vermeiden.

Die Aggregatorumwandlung stellt möglicherweise keine sortierte Ausgabe bereit.

Um die Ausgabe einer Aggregatorumwandlung zu sortieren, verwenden Sie eine Sortiererumwandlung.

Fehlerbehebung bei Aggregator-Umwandlungen

Fehler in Aggregator-Umwandlungen können behoben werden.

Ich habe die sortierte Eingabe ausgewählt, das Mapping nimmt jedoch genauso viel Zeit in Anspruch wie zuvor.

Sie können die sortierte Eingabe nicht verwenden, wenn eine der folgenden Bedingungen zutrifft:

- Der Aggregat-Ausdruck enthält verschachtelte Aggregatfunktionen.
- Quelldaten sind datengesteuert.

Wenn eine dieser Bedingungen zutrifft, verarbeitet der Datenintegrationsdienst die Umwandlung so, als wäre die sortierte Eingabe nicht ausgewählt worden.

Ein Mapping mit einer Aggregator-Umwandlung führt zu schlechter Leistung.

Der Datenintegrationsdienst lagert unter Umständen Daten auf die Festplatte aus. Sie können die Leistung erhöhen, indem Sie den Index und die Daten-Cachegröße in den Umwandlungseigenschaften vergrößern.

KAPITEL 4

Assoziations-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Zuordnungsumwandlung - Übersicht, 117
- Speicherzuordnung, 118
- Zuordnungsumwandlung – Erweiterte Eigenschaften, 119

Zuordnungsumwandlung - Übersicht

Die Zuordnungsumwandlung verarbeitet Ausgabedaten aus einer Match-Umwandlung. Sie erstellt Links zwischen duplizierten Datensätzen, die unterschiedlichen Match-Clustern zugeordnet sind, sodass diese Datensätze zusammen in Datenkonsolidierungs- und Masterdatenverwaltungs-Operationen zugeordnet werden können.

Die Zuordnungsumwandlung generiert einen **AssociationID**-Wert für jede Zeile in einer Gruppe von zugeordneten Datensätzen und schreibt die ID-Werte an einen Ausgabeport.

Die Konsolidierungsumwandlung liest die Ausgabe aus der Zuordnungsumwandlung. Mithilfe einer Konsolidierungsumwandlung können Sie einen Master-Datensatz basierend auf Datensätzen mit ID-Werten für allgemeine Zuordnungen erstellen.

Die Zuordnungsumwandlung akzeptiert Zeichenfolgen- und numerische Datenwerte an Eingabeports. Wenn Sie einen Eingabeport eines anderen Datentyps hinzufügen, wandelt die Umwandlung die Portdatenwerte in Zeichenfolgen um.

Der AssociationID-Ausgabeport schreibt Integer-Daten. Die Umwandlung kann Zeichenfolgendaten an einen AssociationID-Port schreiben, wenn die Umwandlung in früheren Versionen von Informatica Data Quality konfiguriert wurde.

Beispiel: Zuordnen von Match-Umwandlungsausgaben

Die folgende Tabelle enthält drei Datensätze, die dieselbe Person kennzeichnen könnten:

ID	Name	Adresse	Ort	Bundesland	PLZ	SSN
1	David Jones	100 Admiral Ave.	New York	NY	10547	987-65-4321
2	Dennis Jones	1000 Alberta Ave.	New Jersey	NY	-	987-65-4321
3	D. Jones	Admiral Ave.	New York	NY	10547-1521	-

Eine in einer Match-Umwandlung definierte Duplikatsanalyse-Operation kennzeichnet nicht alle drei Datensätze als gegenseitige Duplikate. Dies ist auf die folgenden Ursachen zurückzuführen:

- Wenn Sie eine Duplikat-Suche für Namen und Adressen durchführen, werden die Datensätze 1 und 3 als Duplikate gekennzeichnet, Datensatz 2 wird jedoch ausgelassen.
- Wenn Sie eine Duplikat-Suche für Namen und Sozialversicherungsnummern durchführen, werden die Datensätze 1 und 2 als Duplikate gekennzeichnet, Datensatz 3 wird jedoch ausgelassen.
- Wenn Sie eine Duplikat-Suche für alle drei Attribute durchführen (Name, Adresse, Sozialversicherungsnummer), kennzeichnet die Match-Umwandlung möglicherweise keinen der Datensätze als Matches.

Die Zuordnungsumwandlung verknüpft Daten aus unterschiedlichen Match-Clustern, damit die Datensätze, die eine Cluster-ID gemeinsam verwenden, einen gemeinsamen AssociationID-Wert erhalten. Wie in der folgenden Tabelle dargestellt, erhalten alle drei Datensätze in diesem Beispiel dieselbe AssociationID:

ID	Name	Adresse	Ort	Bundesland	PLZ	SSN	Cluster-ID für Name und Adresse	Cluster-ID für Name und SSN	Zuordnungs-ID
1	David Jones	100 Admiral Ave.	New York	NY	10547	987-65-4320	1	1	1
2	Dennis Jones	1000 Alberta Ave.	New Jersey	NY	-	987-65-4320	2	1	1
3	D. Jones	Alberta Ave.	New York	NY	10547-1521	-	1	2	1

Sie können die Daten aus duplizierten Datensätzen in der Konsolidierungsumwandlung konsolidieren.

Speicherzuordnung

Sie können die in der Zuordnungsumwandlung verwendete Mindestspeichergröße für den Cache festlegen. Die Standardeinstellung ist 400.000 Byte.

Legen Sie den Wert in der Eigenschaft **Cache-Dateigröße** auf der Registerkarte **Erweitert** fest.

Der Standardwert stellt die von der Umwandlung verwendete Mindestspeichergröße dar. Je nach zugewiesenen Ports versucht die Zuordnungsumwandlung, ein Mehrfaches des Standardwerts abzurufen. Die Umwandlung verwendet diese Formel, um Cache-Arbeitsspeicher abzurufen:

(Anzahl der Zuordnungs-Ports + 1) x Standard-Cache-Arbeitsspeicher

Wenn Sie beispielsweise sieben Zuordnungs-Ports konfigurieren, versucht die Umwandlung, dem Cache-Arbeitsspeicher 3,2 Millionen Byte oder 3,05 MB zuzuordnen.

Wenn Sie die Standardeinstellung ändern, versucht die Umwandlung nicht, zusätzlichen Speicher abzurufen.

Hinweis: Wenn Sie einen Cache-Arbeitsspeicherwert unter 65536 eingeben, liest die Zuordnungsumwandlung den Wert in Megabyte.

Zuordnungsumwandlung – Erweiterte Eigenschaften

Die Zuordnungsumwandlung enthält erweiterte Eigenschaften, die das Verhalten des Cache-Arbeitsspeichers und die Tracingebene bestimmen.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Cache-Dateiverzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst temporäre Daten für die aktuelle Umwandlung schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis, wenn das Volumen der Eingabedaten größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cache-Dateigröße

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst zum Sortieren der Eingabedaten bei der Umwandlung verwendet. Der Standardwert ist 400.000 Byte. Sie können einen Parameter verwenden, um die Cache-Dateigröße anzugeben.

Bevor die Daten sortiert werden, weist der Datenintegrationsdienst die von Ihnen angegebene Speichermenge zu. Wenn der Sortiervorgang eine größere Menge an Daten generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Dateiverzeichnis. Wenn der Speicherbedarf für den Sortiervorgang die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 5

Ausnahmeumwandlung bei fehlerhaftem Datensatz

Dieses Kapitel umfasst die folgenden Themen:

- Übersicht über die Ausnahmeumwandlung bei fehlerhaftem Datensatz, 120
- Ausnahme bei fehlerhaftem Datensatz - Ausgabe Datensatztypen, 121
- Prozessablauf für die Ausnahmeverwaltung bei fehlerhaften Datensätzen, 122
- Ausnahme bei fehlerhaftem Datensatz - Mappings, 123
- Ausnahme bei fehlerhaftem Datensatz - Ports , 125
- Ausnahme bei fehlerhaftem Datensatz - Konfigurationsansicht, 126
- Ausnahme bei fehlerhaftem Datensatz - Problemzuordnung , 128
- Ausnahmeumwandlung – Erweiterte Eigenschaften, 129
- Konfigurieren einer Ausnahmeumwandlung bei fehlerhaftem Datensatz, 130
- Ausnahme bei fehlerhaftem Datensatz - Mapping-Beispiel, 131

Übersicht über die Ausnahmeumwandlung bei fehlerhaftem Datensatz

Es handelt sich bei der Ausnahmeumwandlung bei fehlerhaftem Datensatz um eine aktive Umwandlung, die die Ausgabe eines Datenqualitätsprozesses liest und Datensätze erkennt, die manuell überprüft werden müssen. Die Ausnahmeumwandlung bei fehlerhaftem Datensatz ist eine aus mehreren Gruppen bestehende Umwandlung.

Konfigurieren Sie eine Ausnahmeumwandlung bei fehlerhaftem Datensatz zur Analyse der Ausgabe eines Prozesses, das Datenqualitätsprobleme in Datensätzen ermittelt. Ein Datensatz mit einem Datenqualitätsproblem, das weiter geprüft werden muss, wird als Ausnahme bezeichnet.

Die Ausnahmeumwandlung bei fehlerhaftem Datensatz empfängt die Eingabe von einer anderen Umwandlung oder aus einem Datenobjekt in einem anderen Mapping. Die Eingabe für die Umwandlung bei fehlerhaftem Datensatz muss einen oder mehrere Qualitätsproblemports beinhalten, die Textbeschreibungen der Datenqualitätsprobleme empfangen. Die Eingabe für die Umwandlung bei fehlerhaftem Datensatz kann auch einen numerischen Datensatz-Score enthalten, den die Umwandlung zur Bestimmung der Datenqualität aller Datensätze verwenden kann. Legen Sie in der Ausnahmeumwandlung einen oberen und unteren Score-Schwellenwert fest, um Datensätze guter und fehlerhafter Qualität, ausgehend von dem Datensatz-Score, zu

klassifizieren. Die Ausnahmeumwandlung bei fehlerhaftem Datensatz schreibt Ausnahmen und den Text der zugehörigen Qualitätsprobleme in eine Tabelle mit fehlerhaften Datensätzen.

Beispiel: Ein Unternehmen muss die Kundenadressen validieren, bevor Post an Kunden versendet werden kann. Ein Entwickler erstellt ein Mapping, das die Angaben zu Stadt, Bundesland und Postleitzahl des Kunden anhand einer Referenztabelle mit einer Beschriftungsumwandlung validiert. Die Beschriftungsumwandlung validiert die Felder und fügt jeder Zeile basierend auf den Ergebnissen einen Datensatz-Score zu. Die Beschriftungsumwandlung fügt außerdem Text hinzu, der die Qualitätsprobleme für jeden fehlerhaften Datensatz beschreibt. Die Beschriftungsumwandlung fügt den Ausnahmen Text bezüglich der Qualitätsprobleme hinzu, wie zum Beispiel `Ort ungültig` oder `keine Postleitzahl`. Die Ausnahmeumwandlung bei fehlerhaftem Datensatz schreibt Kundendatensätze, die manuell überprüft werden müssen, in die Tabelle mit fehlerhaften Datensätzen. Datenanalysten überprüfen und korrigieren fehlerhafte Datensätze im Analyst-Tool.

Ausnahme bei fehlerhaftem Datensatz - Ausgabe Datensatztypen

Die Ausnahme bei fehlerhaften Datensätzen untersucht Datensatz-Scores der Eingabe, um die Datensatzqualität zu bestimmen. Sie gibt die Datensätze an verschiedene Ausgabegruppen zurück.

Die Ausnahmeumwandlung identifiziert die folgenden Datensatztypen basierend auf jedem Datensatz-Score:

Intakte Datensätze

Datensätze mit Scores, die größer als oder gleich dem oberen Schwellenwert sind. Intakte Datensätze sind gültig und müssen nicht überprüft werden. Wenn Sie beispielsweise den oberen Schwellenwert mit 90 konfigurieren, ist für alle Datensätze größer als 90 keine Überprüfung erforderlich.

Fehlerhafte Datensätze

Datensätze mit Scores, die kleiner als der obere Schwellenwert und Scores, die größer als oder gleich dem unteren Schwellenwert sind. Fehlerhafte Datensätze sind die Ausnahmen, die Sie im Analyst-Tool überprüfen müssen. Wenn beispielsweise der untere Schwellenwert 40 ist, muss jeder Datensatz mit einem Score zwischen 40 und 90 manuell überprüft werden.

Abgelehnte Datensätze

Datensätze mit Scores, die kleiner als der untere Schwellenwert sind. Abgelehnte Datensätze sind ungültig. Standardmäßig entfernt die Ausnahmeumwandlung abgelehnte Datensätze aus dem Datenfluss. Bei diesem Beispiel ist jeder Datensatz mit einem Score von 40 oder weniger ein abgelehnter Datensatz.

Hinweis: Wenn die Qualitätsproblemfelder NULL sind, ist der Datensatz keine Ausnahme. Wenn ein Qualitätsproblem Text oder eine leere Zeichenfolge enthält, ist der Datensatz eine Ausnahme. Stellen Sie sicher, dass ein Qualitätsproblemport Nullwerte enthält, wenn ein Feld keine Fehler hat. Wenn die Qualitätsproblemports Leerzeichen anstatt Nullwerte enthält, markiert die Ausnahmeumwandlung jeden Datensatz als eine Ausnahme. Wenn ein Benutzer die Probleme im Analyst-Tool beheben muss, kann der Benutzer die Ausnahmen nicht nach dem Datenqualitätsproblem filtern.

Wenn ein Datensatz einen Score von kleiner als null oder größer als 100 hat, ist die Zeile nicht gültig. Der Datenintegrationsdienst protokolliert in einer Fehlermeldung, dass die Zeile nicht gültig ist und überspringt die Verarbeitung des Datensatzes.

Wenn Sie einen Datensatz-Score nicht als Eingabe mit einer Ausnahmeumwandlung verbinden, schreibt die Umwandlung alle Datensätze mit Qualitätsproblemen in die Tabelle mit fehlerhaften Datensätzen.

Wenn Sie eine Ausnahmeumwandlung bei fehlerhaftem Datensatz in eine Mapping-Aufgabe einschließen, können Sie eine Human-Aufgabe im gleichen Arbeitsablauf konfigurieren, um eine manuelle Überprüfung der Ausnahmen einzuschließen. Die Human-Aufgabe beginnt, wenn eine Mapping-Aufgabe im Arbeitsablauf endet. Die Human-Aufgabe erfordert, dass Benutzer auf das Analyst-Tool zugreifen, um die Qualitätsprobleme zu beheben. Ein Benutzer kann die Daten aktualisieren und den Qualitätsstatus jedes Datensatzes in der Tabelle mit fehlerhaften Datensätzen ändern.

Prozessablauf für die Ausnahmeverwaltung bei fehlerhaften Datensätzen

Die Ausnahmeumwandlung erhält die Datensatz-Scores von Datenqualitätsumwandlungen und erzeugt Tabellen, die Datensätze mit unterschiedlichen Datenqualitätsstufen enthalten. Sie müssen die Datenqualitätsumwandlungen konfigurieren, um Qualitätsprobleme zu finden und einen Datensatz-Score für jede Zeile anzugeben.

Sie können die Datenqualitätsumwandlungen in einem einzigen Mapping konfigurieren, oder Sie können Mappings für verschiedene Phasen des Datenqualitätsprozesses erzeugen.

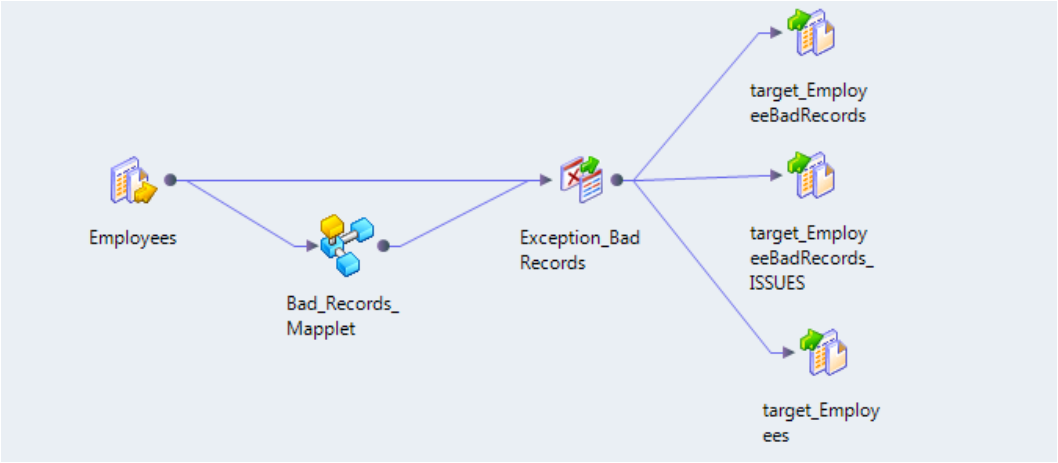
Führen Sie die folgenden Verwaltungsaufgaben der Ausnahmen bei fehlerhaften Datensätzen aus:

1. Definieren Sie Umwandlungen im Developer-Tool, die Score-Werte für Quellendaten in Abhängigkeit der Datenqualitätsprobleme generieren, die Sie festlegen. Definieren Sie Umwandlungen, die Text zurückgeben, um die Qualität der Quelldaten zu beschreiben. Beispielsweise können Sie eine Beschriftungsumwandlung konfigurieren, um Quelldaten mit Referenztabellen abzugleichen. Für jeden Vergleich können Sie dann eine beschreibende Beschriftung verfassen. Sie können eine WENN/DANN-Regel in einer Entscheidungsumwandlung definieren, um ein Datenfeld zu untersuchen. Sie können mehrere Umwandlungen und Mapplets definieren, die verschiedene Vorgänge zur Datenqualität durchführen.
2. Konfigurieren Sie eine Ausnahmeumwandlung, um die Datensatz-Scores zu analysieren, die sie von den Vorgängen zur Datenqualität erhält. Konfigurieren Sie die Umwandlung so, dass sie in Abhängigkeit von den Score-Werten in den Datensätzen die Datensätze in Datenbanktabellen schreibt. Sie können separate Tabellen für intakte Datensätze, fehlerhafte Datensätze, Qualitätsprobleme und abgelehnte Datensätze erstellen.
3. Weisen Sie jedem Eingabeport, der fehlerhafte Daten enthalten könnte, einen Qualitätsproblemport zu.
4. Konfigurieren Sie optional Zieldatenobjekte für intakte und fehlerhafte Datensätze. Verbinden Sie die Umwandlungsausgabeports der Ausnahme mit den Zieldatenobjekten im Mapping.
5. Erstellen Sie die Zieldatenobjekte für fehlerhafte Datensätze. Wählen Sie aus, eine Tabelle mit fehlerhaften Datensätzen zu erstellen, und fügen Sie diese dem Mapping hinzu. Wenn Sie eine Tabelle mit fehlerhaften Datensätzen erstellen, erstellt das Developer-Tool auch eine Tabelle mit Qualitätsproblemen. Fügen Sie die Tabelle mit Qualitätsproblemen dem Mapping hinzu.
6. Fügen Sie das Mapping dem Arbeitsablauf hinzu.
7. Konfigurieren Sie eine Human-Aufgabe, um den Benutzern die manuelle Überprüfung fehlerhafter Datensätze zuzuweisen. Benutzer können fehlerhafte Datensätze im Analyst-Tool überprüfen und aktualisieren.

Ausnahme bei fehlerhaftem Datensatz - Mappings

Wenn Sie ein Mapping erstellen, das Ausnahmen aufgrund fehlerhafter Datensätze erkennt, konfigurieren Sie das Mapping so, dass basierend auf der Qualität der Daten in den Datensätzen Datensätze in ein oder mehrere Datenbankziele geschrieben werden.

Die folgende Abbildung zeigt ein Beispiel für ein Mapping der Ausnahmen bei fehlerhaftem Datensatz:



Das Mapping enthält die folgenden Objekte:

Datenquelle

Eine Mitarbeiter-Datenquelle, die die Datensätze für die Analyse der Datenqualität enthält.

Mapplet

Das Bad_Records_Mapplet enthält Umwandlungen, die Qualitätsprobleme prüfen und diese mit den Datensatz-Scores den Quelldatensätzen hinzufügen. Bei den Regeln handelt es sich um Umwandlungen, die Daten analysieren und Qualitätsprobleme finden. Sie können zum Beispiel eine Beschriftungsumwandlung einfügen, um die Eingabedaten mit einer Referenztabelle zu vergleichen. Je nach Ergebnis können Sie die Beschriftungsumwandlung konfigurieren, damit die Qualitätsprobleme in einer weiteren Spalte in den Zeilen erscheinen. Sie können eine Entscheidungsumwandlung konfigurieren, die die Angaben IF, THEN, ELSE verwendet, um die Daten zu prüfen. Außerdem können Sie den Eingabedaten Qualitätsprobleme und Datensatz-Scores zuweisen.

Ausnahmeumwandlung

Die Ausnahmeumwandlung bestimmt, welche Datensätze in die Datenziele geschrieben werden sollen, einschließlich der Tabelle mit fehlerhaften Datensätzen und der Problemtabelle.

Tabelle mit intakten Datensätzen

Die Ausnahmeumwandlung schreibt alle Datensätze mit guter Qualität in die target_Employees-Tabelle.

Tabelle mit fehlerhaften Datensätzen

Die Ausnahmeumwandlung schreibt alle Datensätze mit schlechter Qualität in die target_EmployeeBadRecords-Tabelle. Fehlerhafte Datensätze müssen manuell geprüft werden.

Problemtabelle

Die Ausnahmeumwandlung schreibt Qualitätsprobleme in die target_EmployeeBadRecords_ISSUES-Tabelle. Wenn Sie die Datensätze mit schlechter Qualität im Analyst-Tool anzeigen, verknüpft die Benutzeroberfläche die Qualitätsprobleme mit den fehlerhaften Datensätzen.

Wahlweise kann die Ausnahmeumwandlung abgelehnte Datensätze in eine Tabelle mit abgelehnten Datensätzen schreiben. Sie müssen eine separate Ausgabegruppe für abgelehnte Datensätze in der Ansicht**Konfiguration** der Umwandlung erstellen.

Ausnahme bei fehlerhaftem Datensatz - Qualitätsprobleme

Qualitätsprobleme sind Textzeichenfolgen, die die Art des Datenqualitätsproblems beschreiben, das zu einem niedrigen Datensatz-Score geführt hat. Die Ausnahmeumwandlung bei fehlerhaftem Datensatz empfängt Qualitätsprobleme, die mit jeder Zeile des Quelldatensatzes, die einen niedrigen Datensatz-Score enthält, verbunden ist. Sie können verschiedene Arten von Umwandlungen konfigurieren, die die Qualitätsprobleme und Datensatz-Scores bestimmen.

Sie können zum Beispiel eine Entscheidungsumwandlung erstellen, die die Telefonnummern untersucht. Die Entscheidungsumwandlung erzeugt den Datensatz-Score und die Qualitätsprobleme für Telefonnummern.

Die folgende Entscheidungsstrategie bestimmt in einer Entscheidungsumwandlung Telefonnummern, deren Länge falsch ist:

```
IF LENGTH(Phone_Number) > 10 THEN
  Score:=50
  Phone_Quality_Issue:='Phone num too long'
ELSEIF  LENGTH(Phone_Number) < 10 THEN
  Score:=50
  Phone_Quality_Issue:=' Phone num too short’
ELSE
  Score:=90
ENDIF
```

Beim Konfigurieren der Ausnahmeumwandlung müssen Sie Phone_Quality_Issue dem Port Phone_Number zuordnen. Die Ports stammen aus unterschiedlichen Eingabegruppen.

Die Ausnahmeumwandlung liest die Scores, die von der Entscheidungsumwandlung erzeugt wurden, und weist Datensätze mit einem Score von "50" den fehlerhaften Datensätzen der Ausgabeports zu. Das Phone_Quality_Issue wird der Problemgruppe der Ausgabeports zugeschrieben.

Human-Aufgaben

Wenn Sie einen Arbeitsablauf konfigurieren, der eine Ausnahmeumwandlung enthält, schließen Sie das Mapping in eine Mapping-Aufgabe mit ein. Sie können demselben Arbeitsablauf eine Human-Aufgabe hinzufügen. Die Human-Aufgabe erfordert einen oder mehrere Benutzer, um die Ausnahmedatensätze im Analyst-Tool zu korrigieren.

Die Mapping-Aufgabe gibt Datensätze in den Quelldaten an, die nicht behobene Datenqualitätsprobleme enthalten. Datenanalysten nutzen das Analyst-Tool, um die Probleme zu beheben und den Datenqualitätsstatus jedes Datensatzes zu aktualisieren.

Beim Konfigurieren einer Human-Aufgabe erstellen Sie eine oder mehrere Aufgabeninstanzen und einen oder mehrere Aufgabenschritte. Eine Aufgabeninstanz stellt die Daten dar, die der Benutzer bearbeiten muss. Ein Aufgabeschritt stellt die Art von Arbeit dar, die ein Benutzer mit den Datensätzen in seiner Aufgabeninstanz durchführt. Sie können mehrere Aufgabeninstanzen erstellen, sodass verschiedene Benutzer an verschiedene Teilen der Daten im Analyst-Tool arbeiten.

Ein Benutzer kann den Status eines fehlerhaften Datensatzes im Analyst-Tool anhand einer der folgenden Methoden aktualisieren:

- Wenn ein Datensatz gültig ist, aktualisiert der Benutzer die Tabellenmetadaten, um den Datensatz für den beständigen Speicher in der Datenbank zu bestätigen.
- Wenn ein Datensatz nicht gültig ist, aktualisiert der Benutzer die Tabellenmetadaten, um den Datensatz aus der Datenbank auf einer späteren Stufe aus dem Arbeitsablauf zu entfernen.

- Wenn der Status eines Datensatzes nicht bestätigt ist, aktualisiert der Benutzer die Tabellenmetadaten, sodass der Datensatz in den Arbeitsablauf für die weitere Verarbeitung in der Mapping-Aufgabe zurückgegeben wird.

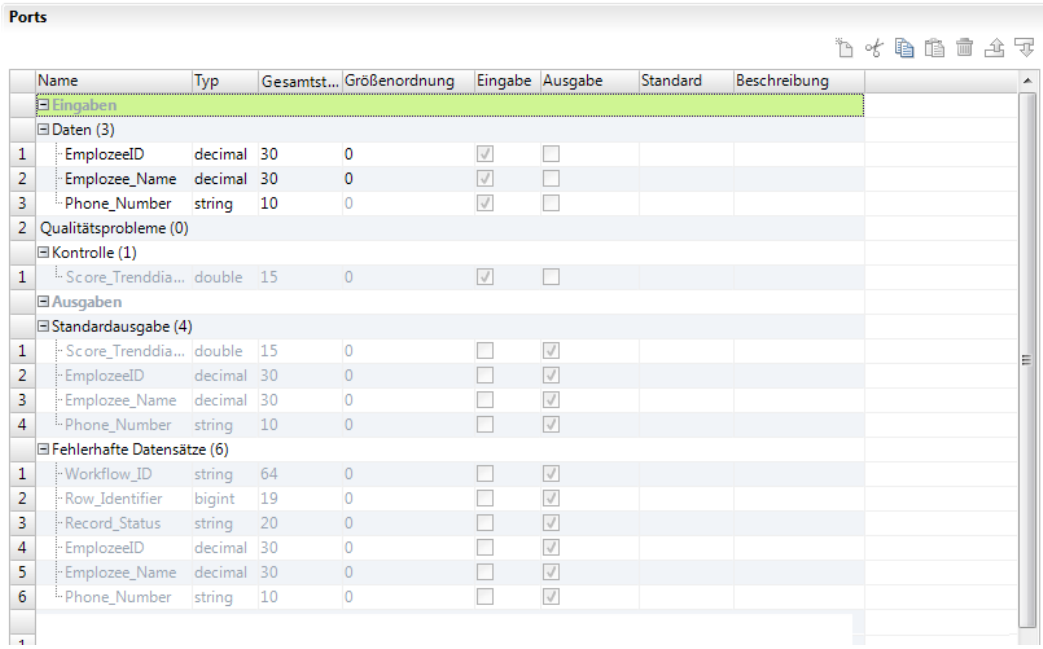
Weitere Informationen zu Human-Aufgaben finden Sie im *Informatica Developer-Arbeitsablaufhandbuch*.

Ausnahme bei fehlerhaftem Datensatz - Ports

Konfigurieren Sie die Eingabe- und Ausgabeports auf der Registerkarte **Ports** der Ausnahmeumwandlung bei fehlerhaftem Datensatz.

Die Ausnahmeumwandlung bei fehlerhaftem Datensatz enthält Eingabe- und Ausgabeportgruppen.

Die folgende Abbildung zeigt die Registerkarte **Ports**:



Ports

	Name	Typ	Gesamtst...	Größenordnung	Eingabe	Ausgabe	Standard	Beschreibung
	Eingaben							
	Daten (3)							
1	EmplozeeID	decimal	30	0	☑	☐		
2	Emplozee_Name	decimal	30	0	☑	☐		
3	Phone_Number	string	10	0	☑	☐		
2	Qualitätsprobleme (0)							
	Kontrolle (1)							
1	Score_Trenddia...	double	15	0	☑	☐		
	Ausgaben							
	Standardausgabe (4)							
1	Score_Trenddia...	double	15	0	☐	☑		
2	EmplozeeID	decimal	30	0	☐	☑		
3	Emplozee_Name	decimal	30	0	☐	☑		
4	Phone_Number	string	10	0	☐	☑		
	Fehlerhafte Datensätze (6)							
1	Workflow_ID	string	64	0	☐	☑		
2	Row_Identifier	bigint	19	0	☐	☑		
3	Record_Status	string	20	0	☐	☑		
4	EmplozeeID	decimal	30	0	☐	☑		
5	Emplozee_Name	decimal	30	0	☐	☑		
6	Phone_Number	string	10	0	☐	☑		

Eingabeports der Ausnahmeumwandlung bei fehlerhaftem Datensatz

Eine Ausnahmeumwandlung bei fehlerhaftem Datensatz verfügt über getrennte Eingabegruppen für die Daten, die Qualitätsprobleme und den Datensatz-Score.

Die Ausnahmeumwandlung bei fehlerhaftem Datensatz enthält die folgenden Eingabegruppen:

Daten

Die Felder der Quelldaten.

Qualitätsprobleme

Enthält Ports, mit denen die Probleme der Datensatzqualität beschrieben werden. Qualitätsproblemports können Zeichenfolgen enthalten, wie z. B. "Excess_Characters" oder "Bad_Data_Format". In jedem Datensatz kann es mehrere Qualitätsprobleme geben. Die Umwandlung verbindet Ports der Gruppe der

Qualitätsprobleme mit den Feldern der Quelldaten erst dann, wenn Sie die Probleme den Datenports in der Ansicht **Problemzuordnung** zugewiesen haben.

Kontrolle

Der Datensatz-Score. Die Ausnahmeumwandlung analysiert den Datensatz-Score, um zu bestimmen, ob es sich bei Eingabezeilen um Ausnahmen handelt. Wenn Sie den Score-Port nicht verbinden, erkennt die Ausnahmeumwandlung eine Zeile als Ausnahme, wenn ein Qualitätsproblemport Daten enthält.

Ausnahme-Umwandlungsausgabe bei fehlerhaftem Datensatz

Eine Ausnahmeumwandlung bei fehlerhaftem Datensatz hat mehrere Ausgabegruppen.

Die Ausnahmeumwandlung bei fehlerhaftem Datensatz enthält die folgenden Ausgabegruppen:

Standardausgabe

Die Datensätze mit guter Qualität, die Sie nicht auf Datenqualitätsprobleme überprüfen müssen.

Jeder Datensatz in der Standardausgabengruppe enthält einen Score-Port, der die Datenqualität des Datensatzes darstellt.

Fehlerhafte Datensätze

Die Ausnahmen, die Sie auf Datenqualitätsprobleme überprüfen müssen.

Jeder Datensatz in der Gruppe mit fehlerhaften Datensätzen enthält eine Arbeitsablauf-ID, eine Zeilenkennung und einen Datensatzstatus-Port.

Probleme

Die Qualitätsprobleme für Datensätze in der fehlerhaften Datensatzgruppe. Die Qualitätsprobleme sind Metadatenelemente, die das Analyst-Tool anzeigt, wenn Sie fehlerhafte Datensätze überprüfen.

Jeder Datensatz in der Problemgruppe enthält eine Arbeitsablauf-ID und einen Zeilenkennungsport, der die Reihe des fehlerhaften Datensatzes kennzeichnet, in der die Probleme auftreten.

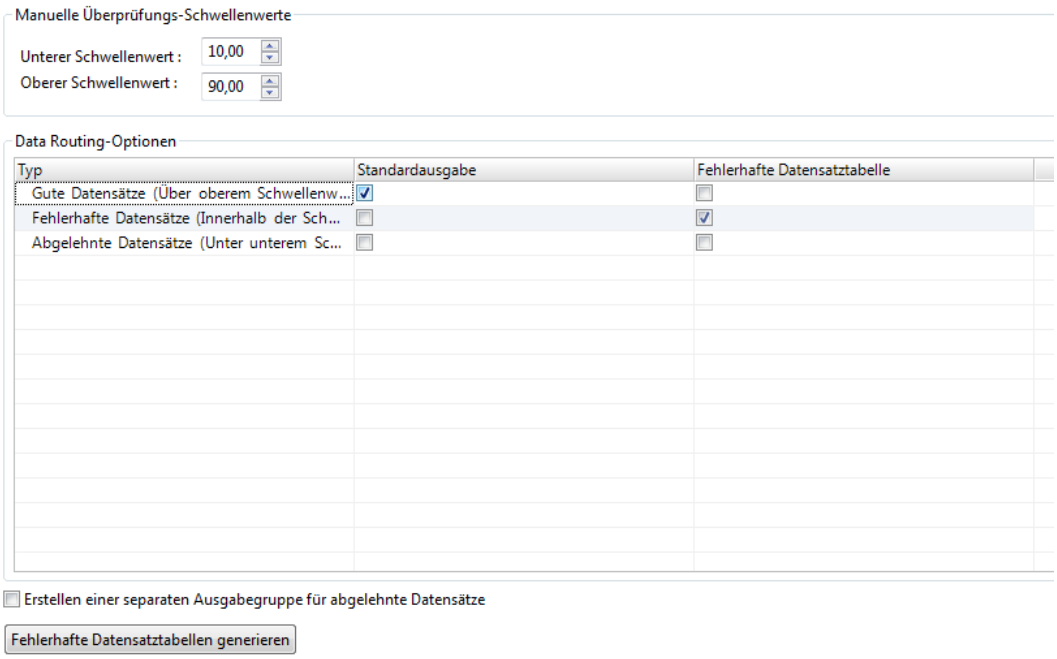
Abgelehnte Datensätze

Optionale Gruppe mit Datensätzen, die Sie möglicherweise aus der Datenbank entfernen möchten. Jeder Datensatz in der Gruppe der abgelehnten Datensätze enthält einen niedrigen Datensatz-Score im Score-Port.

Ausnahme bei fehlerhaftem Datensatz - Konfigurationsansicht

In der Ansicht **Konfiguration** sind die oberen und die unteren Schwellenwerte angegeben, die die Umwandlung zum Ermitteln von intakten und fehlerhaften Datensätzen verwendet. In der Ansicht **Konfiguration** sind auch die Zieltabellen für die Datensätze mit Scores über und unter den Schwellenwerten aufgelistet.

Die folgende Abbildung zeigt die **Konfigurationsansicht** der Ausnahmeumwandlung:



Sie können die folgenden Eigenschaften in der Ansicht **Konfiguration** konfigurieren:

Unterer Schwellenwert

Die untere Grenze für den Score-Bereich fehlerhafter Datensätze. Die Umwandlung ermittelt Datensätze mit Scores, die unter dem unteren Schwellenwert liegen und als abgelehnte Datensätze gekennzeichnet sind.

Oberer Schwellenwert

Die obere Grenze für den Score-Bereich fehlerhafter Datensätze. Die Umwandlung kennzeichnet Datensätze mit Scores, die über dem oberen Schwellenwert liegen oder mit diesem identisch sind und daher als intakte Datensätze gekennzeichnet sind.

Data Routing-Optionen

Die Typen von Ausgabe-Datensätzen. In der Standardkonfiguration schreibt die Umwandlung intakte Datensätze in die Standardausgabe und fehlerhafte Datensätze in die Tabelle mit fehlerhaften Datensätzen. Standardmäßig schreibt die Umwandlung die abgelehnten Datensätze nicht in eine Datenbanktabelle.

Standardausgabe

Die Datensatztypen, die von der Umwandlung auf die Standardausgabeports geschrieben werden. Standardwert ist "Intakte Datensätze".

Tabelle mit fehlerhaften Datensätzen

Die Datensatztypen, die von der Umwandlung auf die Ausgabeports der fehlerhaften Datensätze geschrieben werden. Standardwert ist "Fehlerhafte Datensätze".

Separate Ausgabegruppe für abgelehnte Datensätze erstellen

Erstellt eine separate Ausgabegruppe für abgelehnte Datensätze. Die Option ist standardmäßig deaktiviert.

Erstellen einer Tabelle mit fehlerhaften Datensätzen

Erstellt eine Datenbanktabelle für die fehlerhaften Datensätze. Wenn Sie diese Option auswählen, erstellt die Ausnahmeumwandlung die Datenbanktabelle und fügt das Datenobjekt zum Modellrepository sowie

eine Instanz des Objekts zur Mapping-Arbeitsfläche hinzu. Sie können in einem Mapping die Tabelle mit fehlerhaften Datensätzen für eine Instanz der Ausnahmeumwandlung erzeugen. Wenn Sie eine Tabelle für fehlerhafte Datensätze erstellen, erstellt das Developer-Tool auch eine Problemtabelle zum Speichern von beschreibenden Metadaten über die Datensätze.

Hinweis: Das Developer-Tool fügt ein Suffix mit 12 Zeichen zu jedem Spaltennamen in den Tabellen für fehlerhafte Datensätze hinzu. Wenn Sie eine Oracle-Datenbank verwenden, darf der Name der Quellspalte nicht mehr als 18 Zeichen enthalten.

Erstellen der Tabelle mit fehlerhaften Datensätzen und der Problemtabelle

Wenn Sie die Umwandlung einem Mapping hinzufügen, können Sie die Tabelle mit fehlerhaften Datensätzen und die Problemtabelle generieren. Das Developer-Tool fügt die Tabellen zum Modellrepository hinzu.

1. Klicken Sie zum Erstellen der Tabelle auf **Erstellen einer Tabelle mit fehlerhaften Datensätzen**.

   Das Dialogfeld **Relationales Datenobjekt erstellen** erscheint.
2. Durchsuchen Sie die Datenbankverbindungen. Wählen Sie eine Verbindung zu der Datenbank aus, die die Tabelle enthalten soll.
3. Geben Sie einen Namen für die Tabelle mit fehlerhaften Datensätzen ein. Das Developer-Tool wendet den von Ihnen in die Tabelle mit fehlerhaften Datensätzen und in die Problemtabelle eingegebenen Namen an.

   Das Developer-Tool hängt die folgende Zeichenfolge an den Namen der Problemtabelle an:

   ```
   _ISSUE
   ```

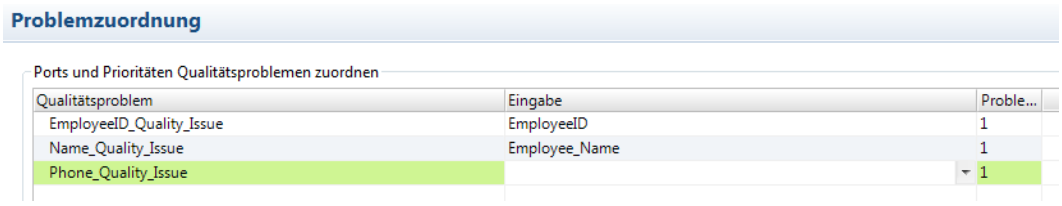
   Wenn Sie eine Verbindung zu einer Oracle-Datenbank herstellen, darf die Tabelle mit fehlerhaften Datensätzen nicht mehr als 24 Zeichen enthalten.
4. Geben Sie einen Namen für das Objekt der Tabelle mit fehlerhaften Datensätzen im Modellrepository ein.
5. Klicken Sie auf **Fertig stellen**.

   Das Developer-Tool fügt die Tabellen der Mapping-Arbeitsfläche und dem Modellrepository hinzu.

Ausnahme bei fehlerhaftem Datensatz - Problemzuordnung

Sie müssen den Datenqualitätsproblemen Ports und Prioritäten zuordnen.

Die folgende Abbildung zeigt die Ansicht **Problemzuordnung**:



Die Ansicht **Problemzuordnung** enthält die folgenden Felder:

Qualitätsproblem

Alle Qualitätsproblemports, die Sie in der Eingabegruppe der Qualitätsprobleme bestimmt haben, erscheinen in der Spalte **Qualitätsproblem**.

Eingabe

Die Spalte **Eingabe** enthält die Datenports, die Sie den Qualitätsproblemen in der Ansicht **Problemzuordnung** zugewiesen haben. Verbinden Sie einen Eingabeport mit jedem Qualitätsproblemport. Jeder Eingabeport, der Daten schlechter Qualität enthält, muss mindestens einen dazugehörigen Qualitätsproblemport enthalten, der den Problemtyp angibt. Sie können als Beispiel den Phone_Number für Phone_Quality_Issue auswählen. Sie können einen Port mehreren Qualitätsproblemen zuordnen.

Problempriorität

Die Problemprioritäten bestimmen, welche Qualitätsprobleme die wichtigsten sind, wenn Sie denselben Eingabeport mehreren Qualitätsproblemen zuordnen. Wenn mehr als ein Qualitätsproblem für einen Eingabeport eintritt, wendet der Datenintegrationsdienst das Qualitätsproblem mit der höchsten Priorität an. Wenn mehr als ein Qualitätsproblem für einen Eingabeport vorhanden ist und die Probleme dieselbe Priorität haben, wendet der Datenintegrationsdienst das oberste Qualitätsproblem in der Liste an. Geben Sie eine Priorität von 1 bis 99 ein, wobei 1 die höchste Priorität darstellt.

Definieren Sie die Problemprioritäten, um Datensätze im Analyst-Tool zu filtern.

Ports den Qualitätsproblemen zuweisen

Bestimmen Sie einen Port, den Sie mit jedem Qualitätsproblem verbinden. Das Developer-Tool erstellt Ports in der Problem-Ausgabegruppe für jede Verbindung, die Sie in der Ansicht **Problemzuordnung** hinzufügen.

1. Klicken Sie für jedes Qualitätsproblem auf das Feld **Eingabe**, um eine Liste der Eingabeports anzuzeigen.
2. Wählen Sie einen Eingabeport, den Sie mit dem Qualitätsproblem verbinden.

   Sie können den gleichen Port für mehrere Probleme auswählen.
3. Klicken Sie auf die Spalte **Problem** und wählen Sie eine Priorität für das Qualitätsproblem.

Ausnahmeumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Datenintegrationsdienst die Daten für die Ausnahmeumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Konfigurieren einer Ausnahmeumwandlung bei fehlerhaftem Datensatz

Konfigurieren Sie die Eingabeports und die Qualitätsprobleme, die in jedem Port entstehen können, wenn Sie die Ausnahmeumwandlung bei fehlerhaftem Datensatz konfigurieren. Bestimmen Sie den oberen und den unteren Schwellenwerte, um die Datenqualität zu ermitteln. Geben Sie an, wo die Ausnahmen und abgelehnten Datensätze aufgezeichnet werden sollen.

1. Erstellen Sie eine wiederverwendbare oder eine nicht wiederverwendbare Ausnahmeumwandlung bei fehlerhaftem Datensatz.
   - Wählen Sie zum Erstellen einer wiederverwendbaren Umwandlung **Datei** > **Neu** > **Umwandlung** aus und wählen Sie eine Ausnahmeumwandlung bei fehlerhaftem Datensatz aus.
   - Öffnen Sie zum Erstellen einer nicht wiederverwendbaren Umwandlung ein Mapping und fügen Sie eine Ausnahmeumwandlung zur Mapping-Arbeitsfläche hinzu. Wählen Sie vom Assistenten eine Ausnahmeumwandlung bei fehlerhaftem Datensatz aus.
2. Klicken Sie auf **Weiter** oder auf **Beenden**.

   Wenn Sie auf **Weiter** klicken, können Sie die Standardschwellenwerte und die Data Routing-Optionen aktualisieren, bevor Sie die Umwandlung erstellen.
3. Konfigurieren Sie die Eingabeports.
   - Wenn Sie eine wiederverwendbare Umwandlung erstellen, wählen Sie die Registerkarte **Ports** aus und fügen Sie Ports für die Daten hinzu, die mit der Umwandlung verbunden werden sollen.
   - Wenn Sie eine nicht wiederverwendbare Umwandlung erstellen, fügen Sie der Mapping-Arbeitsfläche andere Objekte hinzu und ziehen Sie die Eingabeports in die Umwandlung.
4. Wählen Sie die Ansicht **Konfiguration** aus.
5. Konfigurieren Sie die oberen und unteren Schwellenwerte der Scores.
6. Konfigurieren Sie im Abschnitt **Data Routing-Optionen** die Eigenschaften der Standardausgabe und der Ausnahmetabelle, um den Ort festzulegen, in den die Umwandlung den jeweiligen Datensatztyp schreibt.

   Geben Sie an, wo die intakten Datensätze, die fehlerhaften Datensätze und die abgelehnten Datensätze aufgezeichnet werden sollen. Sie können diese in der Standardausgabe oder in der Tabelle mit fehlerhaften Datensätzen aufzeichnen.
7. Öffnen Sie die Ansicht **Problemzuordnung**. Weisen Sie den Datenports die Datenqualitätsprobleme zu.

   Weisen Sie jedem Problem eine Priorität zu. Wenn ein Port Werte mit mehreren Problemen enthält, zeigt die Umwandlung das Problem mit der höchsten Priorität an.
8. Wählen Sie die Option, um eine Tabelle mit fehlerhaften Datensätzen zu erstellen. Geben Sie die Datenbankverbindung sowie Informationen zum Tabellennamen ein. Die Tabelle muss aus dem Standardschema stammen.
   - Wenn Sie eine fehlerhafte Datensatztabelle erzeugen, erstellen Sie eine Tabelle für die Datensätze und eine weitere Tabelle für die Datenqualitätsprobleme, die sich auf die Datensätze beziehen. Die Umwandlung erstellt ein Datenbankobjekt im Modellrepository.
9. Verbinden Sie die Umwandlungsausgabeports mit einem oder mehreren Datenzielen. Verbinden Sie die Ausgabeports mit den Datenobjekten, die den Ausgabeoptionen entsprechen, die Sie in der Ansicht **Konfiguration** festgelegt haben.
   - Wenn Sie eine wiederverwendbare Umwandlung erstellen, fügen Sie die Umwandlung zu einem Mapping hinzu und verbinden Sie die Ausgabeports.

- Wenn Sie eine wiederverwendbare Umwandlung erstellen, verbindet die Umwandlung die Ports mit der Tabelle mit fehlerhaftem Datensatz. Sie verbinden Ausgabeports mit einem beliebigen anderen Daten-Target.

Ausnahme bei fehlerhaftem Datensatz - Mapping-Beispiel

Ein Unternehmen führt ein Datenprojekt durch, um neue Kundendaten zu überprüfen. Das Unternehmen muss sicherstellen, dass die Kundenkontaktdaten gültig sind. Das folgende Beispiel zeigt, wie man eine Ausnahmeumwandlung bei fehlerhaften Datensätzen definiert, die Datensätze von einem Mapplet erhält, welches eine Datenqualitätsanalyse der Kundendatensätze durchführt.

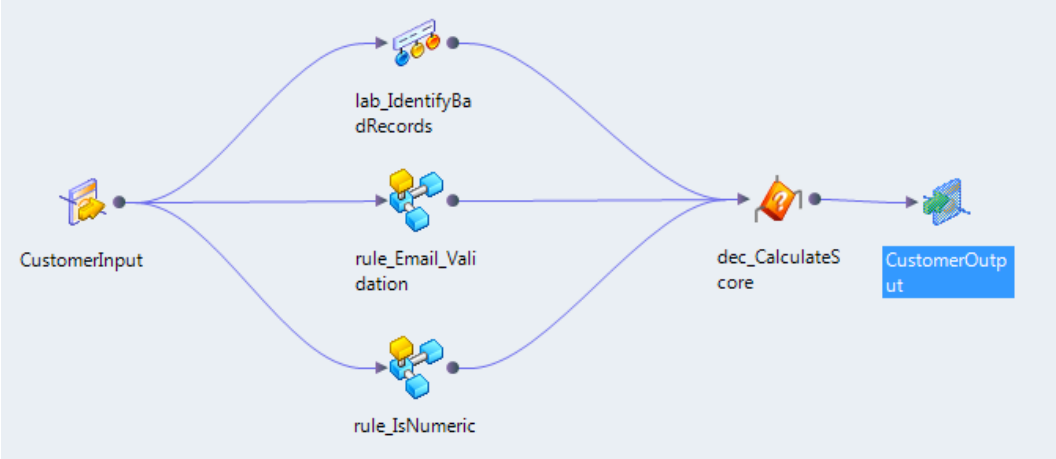
Erstellen Sie ein Mapplet mit Datenqualitätsumwandlungen, welche das Format und die Genauigkeit der Kundendaten auswerten. Das Mapplet beinhaltet Umwandlungen, die anhand der Ergebnisse der Datenqualitätsanalyse einen Datensatz-Score generieren. Die Umwandlungen definieren auch die Qualitätsprobleme für die Daten in Abhängigkeit von den Ergebnissen der Analyse.

Ausnahme bei fehlerhaftem Datensatz - Mapplet

Erstellen Sie ein Mapplet, das Datenqualitätsumwandlungen enthält, um die Werte bestimmter Felder zu prüfen. Die Umwandlungen überprüfen Referenztabellen und Inhaltssätze, um zu bestimmen, ob die Felder in den Datensätzen gültig sind. Die Umwandlungen wenden einen Datensatz-Score auf jeden Datensatz an, der auf den Ergebnissen basiert. Die Ausnahmeumwandlung erhält die Datensätze von dem Mapplet und leitet jeden Datensatz an die entsprechende Ausgabe weiter, die auf dem Datensatz-Score basiert.

Das Mapplet besteht aus Beschriftungsumwandlungen, Entscheidungsumwandlungen und Ausdrucksumwandlungen.

Die folgende Abbildung zeigt die Objekte im Mapplet:



Das Mapplet führt folgende Aufgaben aus:

- Eine Beschriftungsumwandlung überprüft den Ort, das Bundesland/den Bundesstaat, den Ländercode, die Postleitzahl und die Postcodedaten, die sie in den Eingabeports erhält. Die Umwandlung enthält eine Strategie für jeden Port. Die Strategien vergleichen die Quelldaten mit den Referenztabellen und identifizieren Werte, die nicht gültig sind.
- Ein Mapplet der Ausdrucksumwandlung überprüft, ob die Telefonnummer numerisch ist und aus 10 Ziffern besteht.
- Ein Mapplet der Beschriftungsumwandlung und der Ausdrucksumwandlung überprüft, ob die E-Mail-Adresse gültig ist. Die Ausdrucksumwandlung überprüft die Struktur der Zeichenfolge der E-Mail. Die Beschriftungsumwandlung gleicht die IP-Adresse mit einer Referenztabelle mit internationalen Endungen für IP-Adressen ab.
- Eine Entscheidungsumwandlung erhält die Ausgabe von der Umwandlung und den Mapplets. Sie berechnet einen Gesamt-Score des Datensatzes für den Datensatz des Kundenkontakts.

Erstellen Sie ein Mapping der Ausnahme bei fehlerhaften Datensätzen, welche das Mapplet enthält. Das Mapping der Ausnahme bei fehlerhaften Datensätzen beinhaltet eine Ausnahmeumwandlung, welche die Ausnahmen in eine Datenbanktabelle für fehlerhafte Datensätze schreibt. Ein Datenanalyst untersucht und aktualisiert die Ausnahmedatensätze in der Tabelle fehlerhafter Datensätze mit dem Analyst-Tool.

Ausnahme bei fehlerhaftem Datensatz - Beispiel für Eingabegruppen

Die Ausnahmeumwandlung umfasst drei Eingabegruppen. Die Umwandlung umfasst eine Datengruppe, die die Quelldaten erhält. Sie enthält die Qualitätsproblemgruppe mit den Datenqualitätsproblemen, die von den Datenqualitätsumwandlungen gefunden werden. Sie enthält außerdem die Kontrollgruppe, die den Datensatz-Score für die Zeile enthält.

Die folgende Abbildung zeigt die Eingabegruppen in der Ausnahmeumwandlung:

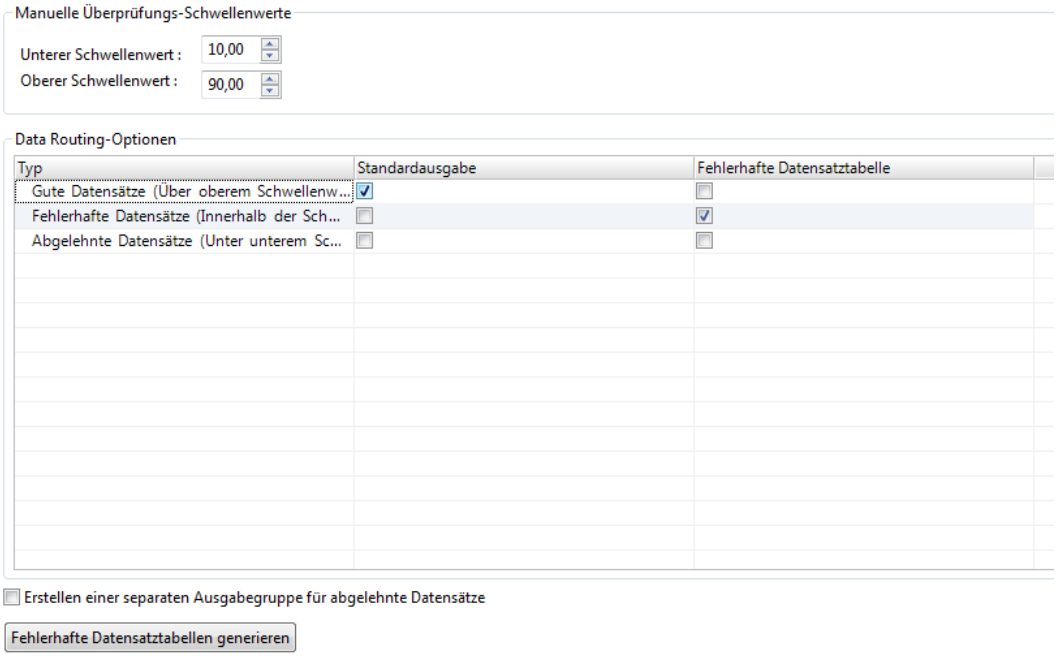
	Name	Typ	Gesamtstellenanzahl	Größenordnung	Eingabe	Ausgabe
	⊟ Eingaben					
	⊟ Daten (11)					
1	CUST_ID	decimal	8	0	☑	☐
2	COMPANY	string	49	0	☑	☐
3	CONTACT	string	19	0	☑	☐
4	TITLE	string	35	0	☑	☐
5	ADDR1	string	47	0	☑	☐
6	ADDR2	string	47	0	☑	☐
7	ADDR3	string	42	0	☑	☐
8	ADDR4	string	36	0	☑	☐
9	COUNTRY	string	47	0	☑	☐
10	PHONE	string	34	0	☑	☐
11	EMAIL	string	25	0	☑	☐
	⊟ Qualitätsprobleme (6)					
1	CompanySt...	string	19	0	☑	☐
2	LocalityStatus	string	22	0	☑	☐
3	ProvinceSta...	string	19	0	☑	☐
4	CountryStat...	string	19	0	☑	☐
5	ZipStatus	string	8	0	☑	☐
6	EmailStatus	string	13	0	☑	☐
	⊟ Kontrolle (1)					
1	Score	double	15	0	☑	☐

Beispielkonfiguration für Ausnahme bei fehlerhaftem Datensatz

Definieren Sie die oberen und unteren Schwellenwerte in der Ansicht **Konfiguration**. Geben Sie an, wohin die Umwandlung intakte Datensätze, fehlerhafte Datensätze und abgelehnte Datensätze schreiben soll.

Akzeptieren Sie den Standardwert der Konfiguration für die Weiterleitung der intakten Datensätze, der fehlerhaften Datensätze und der Probleme.

Die folgende Abbildung zeigt die **Konfigurationsansicht** für die Ausnahmeumwandlung:



In der folgenden Tabelle werden die Konfigurationseinstellungen beschrieben:

Option	Einstellung
Unterer Schwellenwert	10
Oberer Schwellenwert	90
Intakte Datensätze	Standardausgabe
Fehlerhafte Datensätze	Tabelle mit fehlerhaften Datensätzen
Abgelehnte Datensätze	-

Klicken Sie auf **Fehlerhafte Datensätze erstellen**, um die Tabellen mit fehlerhaften Datensätzen und Problemen zu erstellen.

Beispiel für eine Ausnahme bei fehlerhaftem Datensatz - Mapping-Ausgabe

Fügen Sie dem Mapping ein Schreibdatenobjekt hinzu und verbinden Sie die Standardausgabeports mit dem Datenobjekt. Das Mapping enthält auch das Datenbankobjekt für fehlerhafte Datensätze und das Datenbankobjekt für Probleme, die in der Ansicht **Konfiguration** erstellt wurden.

Tabelle mit fehlerhaften Datensätzen

Die Tabelle mit fehlerhaften Datensätzen enthält die Ausnahmen mit Datensatz-Scores zwischen den unteren und oberen Schwellenwerten.

Die folgende Abbildung zeigt die fehlerhaften Datensätze, die die Ausnahmeumwandlung zurückgibt:

Name: exc_BadRecord.Bad_Output_DI

	Workflow_ID	Row_Identifier	Record_Status	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2
1	DummyWorkf...	0	INVALID	1001590	E-AGENCY	OAKLAND	Federal Expre...	291 3RD STRE...	2500 WESTER...
2	DummyWorkf...	1	INVALID	1001599	BANK ONE	ELGIN	US Postal Inte...	2500 WESTER...	530 CHESNUT...
3	DummyWorkf...	2	INVALID	1001604	KPMG PEAT...	WOODCLIFF L...	US Postal 2-day	530 CHESNUT...	"50 BROAD S...
4	DummyWorkf...	3	INVALID	1001622	INVESTEX	NEW YORK	United Parcel...	"50 BROAD S...	20TH FL"
5	DummyWorkf...	4	INVALID	7121564	"ARTHUR VA...	INC."	WHITEHOUSE...	Federal Expre...	310 BRIER RO...
6	DummyWorkf...	5	INVALID	7121565	OSTERREICHI...	NEW YORK	United Parcel...	767 FIFTH AV...	570 LEXINGT...
7	DummyWorkf...	6	INVALID	7121566	K2 ADVISORS	NEW YORK	Courier	570 LEXINGT...	51 JFK PARK...
8	DummyWorkf...	7	INVALID	7121567	MFP INVESTO...	SHORT HILLS	Federal Expre...	51 JFK PARK...	ONE PARKER...
9	DummyWorkf...	8	INVALID	7121568	EASTON	FORT LEE	US Postal Inte...	ONE PARKER...	3840 HIGHW...
10	DummyWorkf...	9	INVALID	7121569	RICE VOELKER	MANDEVILLE	Federal Expre...	3840 HIGHW...	"2ND & D ST...
11	DummyWorkf...	10	INVALID	1001660	OPPENHEIME...	ENGLEWOOD	US Postal Inte...	6803 S. TUCS...	6803 S. TUCS...

Die Tabelle mit fehlerhaften Datensätzen enthält alle Felder des Quelldatensatzes. Ein fehlerhafter Datensatz beinhaltet auch die folgenden Felder:

Workflow_ID

Der Name des Arbeitsablaufs, der die Ausnahmeumwandlung beinhaltet. Der Arbeitsablauf enthält die Mapping-Aufgabe der Ausnahmeumwandlung und die Human-Aufgabe, um die Probleme zu überprüfen. Workflow_ID enthält `DummyWorkflowID`, wenn die Ausnahmeumwandlung nicht in einem Arbeitsablauf integriert ist.

Row_Identifier

Eine einmalige Nummer, die jede Zeile kennzeichnet.

Record_Status

Ein Datensatzstatus für das Analyst-Tool. Jeder Datensatz in der Tabelle fehlerhafter Datensätze erhält den Status `ungültig`. Sie können den Datensatzstatus beibehalten, wenn Sie Datensätze im Analyst-Tool aktualisieren.

Problemtabelle

Die Problemtabelle enthält eine Zeile für jede Zeile in der Tabelle mit fehlerhaften Datensätzen. Jede Zeile enthält die Probleme, die in der Datenqualitätsanalyse für den Quelldatensatz gefunden wurden.

Die folgende Abbildung zeigt die Spalten in der Problemtabelle:

Ausgabe

Name: exc_BadRecord.Issues_DI

	Workflow_ID	Row_Identifier	ADDR2	DQAPRIORITY_ADDR2
1	DummyWorkflowID	0	6803 S. TUCSON WAY	1
2	DummyWorkflowID	1	6803 S. TUCSON WAY	1
3	DummyWorkflowID	2	6803 S. TUCSON WAY	1
4	DummyWorkflowID	3	6803 S. TUCSON WAY	1
5	DummyWorkflowID	4	6803 S. TUCSON WAY	1
6	DummyWorkflowID	5	6803 S. TUCSON WAY	1
7	DummyWorkflowID	6	38TH FLOOR"	1
8	DummyWorkflowID	7	6803 S. TUCSON WAY	1
9	DummyWorkflowID	8	SUITE 200"	1

Die Problemtabelle enthält die folgenden Spalten:

Workflow_ID

Kennzeichnet den Arbeitsablauf, der den Datensatz erstellt hat. Der Arbeitsablauf enthält die Mapping-Aufgabe der Ausnahmeumwandlung und die Human-Aufgabe, um das Problem zu überprüfen.

Row_Identifier

Kennzeichnet die Datensatzzeile in der Datenbanktabelle. Die Zeilenkennung kennzeichnet die Spalte in der Tabelle mit fehlerhaften Datensätzen, die der Spalte in der Problemtabelle entspricht.

Problemfeldname

Der Feldname ist der Name des Felds, in dem es zu Qualitätsproblemen kommen könnte. Wenn das Feld einen Fehler enthält, ist der Text bezüglich des Qualitätsproblems der Wert der Spalte. In der obigen Abbildung enthält der Feldname ADDR2 ein `invalid_locality`-Qualitätsproblem.

DQAPriority

Das Problempriorität. Wenn mehrere Probleme im selben Feld auftreten, erscheint das Problem mit der höchsten Priorität im Problemfeldnamen.

Tabelle mit intakten Datensätzen

Jeder Datensatz in der Tabelle mit intakten Datensätzen hat einen Datensatz-Score, der den oberen Schwellenwert übersteigt. In diesem Beispiel ist der obere Schwellenwert 90.

Die folgende Abbildung zeigt die intakten Datensätze, die die Ausnahmeumwandlung zurückgibt:

Name: exc_BadRecord.Output_DI

	Score	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2	ADDR3
1	<Null>	1001599	BANK ONE	ELGIN	US Postal Inte...	2500 WESTER...	530 CHESNUT...	6803 S. TUCS...
2	<Null>	7121565	OSTERREICHI...	NEW YORK	United Parcel...	767 FIFTH AV...	570 LEXINGT...	6803 S. TUCS...
3	<Null>	7121566	K2 ADVISORS	NEW YORK	Courier	570 LEXINGT...	51 JFK PARK...	6803 S. TUCS...
4	<Null>	7121567	MFP INVESTO...	SHORT HILLS	Federal Expre...	51 JFK PARK...	ONE PARKER...	6803 S. TUCS...
5	<Null>	7121568	EASTON	FORT LEE	US Postal Inte...	ONE PARKER...	3840 HIGHW...	6803 S. TUCS...
6	<Null>	1001660	OPPENHEIME...	ENGLEWOOD	US Postal Inte...	6803 S. TUCS...	6803 S. TUCS...	6803 S. TUCS...
7	<Null>	1001659	THOMSON LE...	FT. WORTH	Courier	301 COMMER...	301 COMMER...	6803 S. TUCS...
8	<Null>	1001658	LYNCH JONE...	NEW YORK	US Postal Ove...	875 3RD AVE.	875 3RD AVE.	6803 S. TUCS...
9	<Null>	1001691	EPICENTRIC	SAN FRANCIS...	US Postal 2-day	333 BRYANT...	333 BRYANT...	6803 S. TUCS...
10	<Null>	1001664	CHARLES SCH...	SAN ANTONIO	United Parcel...	1100 N.E. LO...	1100 N.E. LO...	6803 S. TUCS...

Die Tabelle mit intakten Datensätzen enthält den Datensatz-Score und die Felder der Quelldaten.

KAPITEL 6

Fallumwandlerumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Fallumwandlerumwandlung - Übersicht, 136
- Fallstrategie-Eigenschaften, 136
- Konfigurieren einer Fallumwandlersrategie, 137
- Groß-/Kleinschreibungsumwandlung – Erweiterte Eigenschaften, 138

Fallumwandlerumwandlung - Übersicht

Die Fallumwandlerumwandlung ist eine passive Umwandlung, die Datenuniformität durch Standardisierung der Groß-/Kleinschreibung von Strings in Eingabedaten herstellt.

Wenn Sie die Datenqualität verbessern möchten, verwenden Sie die Fallumwandlerumwandlung, um Daten in ähnliche Zeichenformate zu formatieren. Sie können vordefinierte Fallumwandlertypen auswählen, z. B. Großbuchstaben, Kleinbuchstaben, Groß-/Kleinschreibung umkehren, ersten Buchstaben im Wort groß schreiben und ersten Buchstaben im Satz groß schreiben.

Sie können auch Referenztabellen verwenden, um die Fallumwandlung zu steuern. Verwenden Sie Referenztabellen, um die Groß-/Kleinschreibung von Eingabestrings in die Werte aus der Spalte "Gültig" aus der Referenztabelle zu ändern. Sie können Referenztabellen verwenden, wenn es sich bei dem Fallumwandlungstyp um **Ersten Buchstaben im Wort groß schreiben** oder **Ersten Buchstaben im Satz groß schreiben** handelt.

Innerhalb einer Fallumwandlerumwandlung können Sie mehrere Fallumwandlungsstrategien erstellen. Jeder Strategie verwendet einen einzigen Konvertierungstyp. Die Fallumwandlerumwandlung bietet einen Assistenten, den Sie zum Erstellen von Strategien verwenden.

Fallstrategie-Eigenschaften

Sie können die Eigenschaften für Fallumwandlungsstrategien konfigurieren.

In der Ansicht **Strategien** können Sie die folgende Fallumwandlungseigenschaften konfigurieren.

Konvertierungstyp

Definiert die von einer Strategie verwendete Fallumwandlungsmethode. Sie können die folgenden Fallumwandlungstypen anwenden:

- **Großbuchstaben**. Wandelt alle Buchstaben in Großbuchstaben um.
- **Ersten Buchstaben im Satz groß schreiben** Schreibt den ersten Buchstaben des Felddatenstrings groß.
- **Groß-/Kleinschreibung umkehren**. Wandelt Kleinbuchstaben in Großbuchstaben und Großbuchstaben in Buchstaben um.
- **Ersten Buchstaben im Wort groß schreiben** Schreibt den ersten Buchstaben in jedem Substring groß.
- **Kleinbuchstaben**. Wandelt alle Buchstaben in Kleinbuchstaben um.

Die Standard-Fallumwandlungsmethode ist Großbuchstaben.

Lassen Sie groß geschriebene Wörter unverändert.

Überschreibt die ausgewählte Großschreibung für Großbuchstabenstrings.

Delimiter

Definiert, wie Großschreibungsfunktionen für die Umwandlung für die in einem Wort groß geschriebenen ersten Buchstaben funktioniert. Wählen Sie beispielsweise einen Gedankenstrich als Delimiter zum Umwandeln von "smith-jones" in "Smith-Jones" aus. Der Standard-Delimiter ist das Leerzeichen.

Referenztabelle

Wendet das in einer Referenztabelle angegebene Großschreibungsformat an. Gilt nur, wenn es sich bei der Fallumwandlungsoption um **Groß-/Kleinschreib. der Überschrift** oder **Groß-/Kleinschreib. des Satzes** handelt. Klicken Sie auf **Neu**, um eine Referenztabelle zur Strategie hinzuzufügen.

Hinweis: Wenn ein Referenztabellen-Match am Anfang eines Tokens auftritt, wird das nächste Zeichen in diesem Token in einen Großbuchstaben geändert. Wenn es sich beispielsweise bei der Eingabezeichenfolge um mcdonald handelt und die Referenztabelle einen Eintrag für Mc aufweist, lautet die Ausgabezeichenfolge McDonald.

Konfigurieren einer Fallumwandlersrategie

Konfigurieren Sie zum Ändern der Groß-/Kleinschreibung von Eingabestrings die Einstellungen in der Ansicht **Strategien** einer Fallumwandlerumwandlung.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Klicken Sie auf **Neu**.

   Der Assistent **Neue Strategie** wird geöffnet.
3. Bearbeiten Sie optional den Namen und die Beschreibung der Strategie.
4. Klicken Sie auf die Felder **Eingaben** und **Ausgaben**, um Ports für die Strategie auszuwählen.
5. Konfigurieren Sie die Strategieeigenschaften. Die Standard-Umwandlungsstrategie lautet **Großbuchstaben**.
6. Klicken Sie auf **Weiter**.
7. Optional können Sie Referenztabellen hinzufügen, um die Groß-/Kleinschreibungsoptionen für Eingabedaten anzupassen, die mit Referenztabelleneinträgen übereinstimmen. Die Anpassung der

Groß-/Kleinschreibungsoptionen für Referenztabellen gilt nur für Strategien für die in einem Wort und einem Satz groß geschriebenen ersten Buchstaben.

8. Klicken Sie auf **Fertig stellen**.

Groß-/Kleinschreibungsumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für den Groß-/Kleinschreibungsumwandler verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 7

Klassifizierumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Klassifizierumwandlung - Übersicht, 139
- Klassifiziermodelle, 140
- Klassifizieralgorithmen, 140
- Klassifizierumwandlungsoptionen, 140
- Klassifizierstrategien, 141
- Klassifiziererumwandlung – Erweiterte Eigenschaften, 141
- Konfigurieren einer Klassifizierstrategie, 142
- Beispiel für Klassifizieranalyse, 142

Klassifizierumwandlung - Übersicht

Die Klassifizierumwandlung ist eine passive Umwandlung, die Eingabefelder analysiert und den Informationstyp in jedem Feld identifiziert. Verwenden Sie eine Klassifizierumwandlung, wenn Eingabefelder mehrere Textwerte enthalten.

Beim Konfigurieren der Klassifizierumwandlung wählen Sie ein Klassifiziermodell und einen Klassifizieralgorithmus. Ein Klassifiziermodell ist eine Art Referenzdatenobjekt. Ein Klassifizieralgorithmus ist ein Satz von Regeln, die die Anzahl der ähnlichen Wörter in einer Zeichenfolge und der relativen Positionen der Wörter berechnet. Die Umwandlung vergleicht die Algorithmusanalyse mit dem Inhalt des Klassifiziermodells. Die Umwandlung gibt die Modellklassifizierung zurück, um den dominanten Informationstyp in dem String zu identifizieren.

Die Klassifizierumwandlung kann Zeichenfolgen von großer Länge analysieren. Beispielsweise können Sie die Umwandlung verwenden, um den Inhalt von E-Mail-Nachrichten, Meldungen aus sozialen Netzwerken und Dokumenttext zu klassifizieren. Übergeben Sie den Inhalt jedes einzelnen Dokumentes oder jeder einzelnen Meldung in ein Feld in einer Datenquellspalte und verbinden Sie die Spalte mit einer Klassifizierumwandlung. Bereiten Sie in jedem Fall die Datenquelle vor, sodass jedes Feld den gesamten Inhalt eines Dokuments oder einer Zeichenfolge enthält, die Sie analysieren möchten.

Klassifiziermodelle

Die Klassifizierumwandlung verwendet ein Referenzdatenobjekt, das auch als Klassifiziermodell bezeichnet wird, um Eingabedaten zu analysieren. Wählen Sie das Klassifiziermodell, wenn Sie die Klassifizierumwandlung konfigurieren. Die Umwandlung vergleicht eine Eingabedatenspalte mit den Daten des Klassifizierermodells und gibt eine Beschriftung zurück, die den Informationstyp in jedem Eingabefeld beschreibt.

Ein Klassifizierermodell enthält Referenzdatenzeilen und Beschriftungswerte. Die Zeilen stellen die Eingabedaten auf dem Port dar, den Sie unter Umständen mit der Klassifiziererumwandlung verbinden. Die Beschriftungswerte beschreiben die Informationstypen, die in den Datenzeilen enthalten sind. Wenn Sie ein Klassifizierermodell konfigurieren, weisen Sie jeder Referenzdatenzeile im Modell eine Beschriftung zu.

Sie kompilieren das Modell, um die Referenzdatenzeilen mit den Beschriftungen in einem Klassifizierermodell zu verknüpfen. Der Kompilierungsprozess erzeugt eine Reihe von logischen Verbindungen zwischen den Datenzeilen und den Beschriftungswerten. Wenn Sie eine Zuordnung ausführen, die das Modell liest, wendet der Datenintegrationsdienst die Logik des Modells auf die Eingabedaten der Klassifiziererumwandlung an. Der Datenintegrationsdienst gibt die Beschriftungen zurück, die die Informationen in den Eingabedatenfeldern am genauesten beschreiben.

Sie erstellen ein Klassifizierermodell im Developer-Tool. Das Modellrepository speichert das Klassifizierermodellobjekt. Das Developer-Tool schreibt die Datenzeilen, die Beschriftungen und die Kompilierungsdaten in eine Datei in der Informatica-Verzeichnisstruktur.

Klassifizieralgorithmen

Beim Hinzufügen eines Klassifiziermodells zu einer Umwandlungsstrategie wählen Sie gleichzeitig auch einen Klassifizieralgorithmus. Der Algorithmus bestimmt, wie die Umwandlung die Klassifizierungsmodelldaten mit den Eingabedaten vergleicht.

Sie können den **Naiver Bayes**-Algorithmus oder **maximale Entropie**-Algorithmus auswählen.

Berücksichtigen Sie die folgenden Faktoren, wenn Sie einen Algorithmus auswählen:

- Der maximale Entropie-Algorithmus führt eine gründlichere Analyse durch als der Naive Bayes-Algorithmus.
- Ein Mapping, das den Naive Bayes-Algorithmus verwendet, wird schneller ausgeführt als ein Mapping, das den maximalen Entropie-Algorithmus für die gleichen Daten verwendet.
- Wählen Sie den maximalen Entropie-Algorithmus mit dem Klassifiziermodell aus, das im Kernbeschleuniger von Informatica enthalten ist.

Klassifizierumwandlungsoptionen

Die Klassifizierumwandlung zeigt konfigurierbare Optionen auf einer Reihe von Registerkarten oder Ansichten im Developer-Tool an.

Beim Öffnen einer wiederverwendbaren Umwandlung werden die Optionen auf einer Reihe von Registerkarten im Umwandlungseditor angezeigt. Beim Öffnen einer nicht wiederverwendbaren Umwandlung in einem Mapping werden die Optionen in einer Reihe von Ansichten im Mapping-Editor angezeigt. Wählen Sie die

Registerkarte "Eigenschaften" in Mappings, um Ansichten für nicht wiederverwendbare Umwandlungen einzusehen.

Sie können die folgenden Ansichten wählen:

Allgemein

Anzeigen und aktualisieren des Umwandlungsnamens und -beschreibung.

Ports

Anzeigen des Eingabe- und Ausgabeports für die Umwandlung.

Hinweis: In einer wiederverwendbaren Klassifizierumwandlung sind die Ansichten "Allgemein" und "Ports" in der Registerkarte **Übersicht** zusammengefasst.

Strategien

Hinzufügen, Entfernen oder Bearbeiten einer Strategie.

Abhängigkeiten

Anzeigen der Eingabe- und Ausgabeports für jede Strategie.

Erweitert

Legen Sie die Detailebene fest, die die Umwandlung in die Protokolldateien schreibt.

Klassifizierstrategien

Eine Strategie ist ein Satz von Datenanalysevorgängen, den eine Umwandlung auf die Eingabedaten ausführt. Erstellen Sie mindestens eine Strategie in der Klassifizierumwandlung. Eine Klassifizierstrategie liest einen einzelnen Eingabeport aus.

Definieren Sie einen oder mehrere Vorgänge in einer Strategie. Ein Klassifiziervorgang identifiziert ein Klassifiziermodell und einen Klassifizieralgorithmus, der auf die Eingabeportdaten angewendet wird. Jeder Vorgang schreibt in einen anderen Ausgabeport. Erstellen Sie mehrere Vorgänge in einer Strategie, wenn Sie einen Eingabeport auf verschiedene Weise analysieren möchten.

Hinweis: Wenn Sie die Sprachen in den Quelldaten identifizieren möchten, wählen Sie den maximalen Entropie-Algorithmus in dem Klassifiziervorgang.

Klassifiziererumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Klassifiziererumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Konfigurieren einer Klassifizierstrategie

Konfigurieren einer Strategie, um die Informationstypen in den Daten zu identifizieren. Jede Strategie analysiert einen Eingabeport.

Verbinden Sie die Eingabeports in einer nicht wiederverwendbaren Umwandlung mit der Umwandlung, bevor Sie die Strategie konfigurieren.

1. Öffnen Sie die Umwandlung und wählen Sie die Ansicht **Strategien**.
2. Klicken Sie auf **Neue Strategie**.

   Der Assistent für die Strategieerstellung wird geöffnet.
3. Geben Sie einen Namen und optional eine Beschreibung für die Strategie ein.
4. Wählen Sie aus dem Feld "Eingaben" einen Eingabeport aus.
5. Stellen Sie sicher, dass der Genauigkeitswert des Eingabeports hoch genug ist, um alle Felder im Eingabeport zu lesen. Der Port schneidet die Eingabedaten ab, wenn das Limit des Genauigkeitswerts erreicht wurde.
6. Wählen oder deaktivieren Sie die Option, um Bewertungsdaten zur Strategieausgabe hinzuzufügen.
7. Klicken Sie auf **Weiter**.
8. Bestätigen Sie den Typ des Klassifiziervorgang und klicken Sie auf **Weiter**.
9. Wählen Sie einen Klassifizieralgorithmus. Sie können aus den folgenden Algorithmen auswählen:
   - Naiver Bayes
   - Maximale Entropie

   Hinweis: Zur Identifizierung der verwendeten Sprache in den Quelldaten wählen Sie die maximale Entropie.
10. Überprüfen Sie den Ausgabeport.

    Die Umwandlung erstellt einen einzigen Ausgabeport für jeden Vorgang in einer Strategie. Sie können Sie den Portnamen und den Genauigkeitswert bearbeiten.
11. Wählen Sie ein Klassifiziermodell.

    Der Assistent listet die Klassifiziermodellobjekte im Modellrepository auf.
12. Klicken Sie auf **Weiter**, um einen anderen Vorgang der Strategie hinzuzufügen. Klicken Sie andernfalls auf **Fertigstellen**.

Beispiel für Klassifizieranalyse

Sie sind ein Datenverwalter in einem Softwareunternehmen, das eine neue Smartphone-Anwendung veröffentlicht hat. Das Unternehmen möchte die öffentliche Reaktion auf die Anwendung und die Beachtung in den Medien analysieren. Das Unternehmen beauftragt Sie und Ihr Team damit, die Kommentare über die Anwendung in sozialen Medien zu analysieren.

Sie entscheiden sich für eine Analyse der Daten aus Twitter-Feeds, in denen Smartphones diskutiert werden. Sie verwenden die Programmierschnittstelle für die Twitter-Anwendung, um den Twitter-Datenstrom zu filtern. Sie erstellen eine Datenquelle, die die Twitter-Daten enthält, die Sie analysieren möchten.

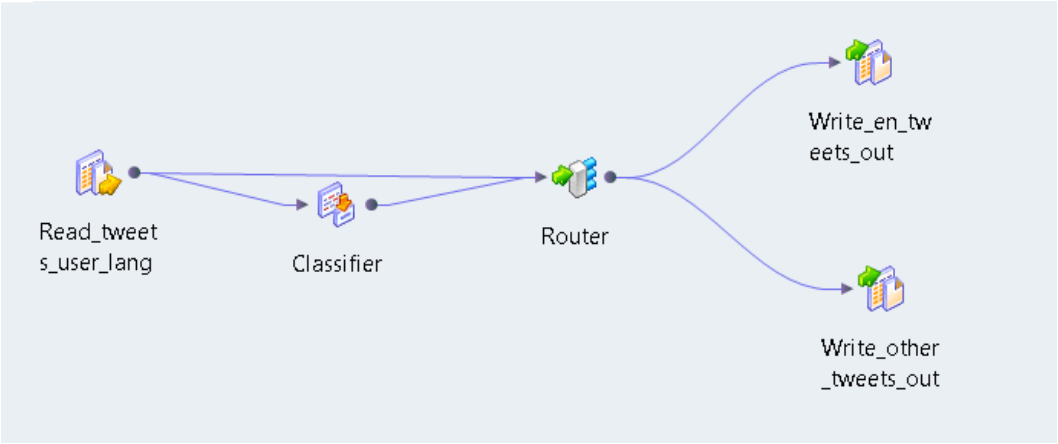
Weil der Twitter-Feed Meldungen in mehreren Sprachen enthält, müssen Sie die verwendete Sprache in jeder Meldung identifizieren. Sie entscheiden sich dafür, eine Klassifizierumwandlung zum Analysieren der

Sprachen zu verwenden. Sie erstellen ein Mapping, um die Sprachen in den Quelldaten zu identifizieren und die Twitter-Meldungen in englische und nicht englische Datenziele zu schreiben.

Erstellen eines Klassifizier-Mappings

Sie können ein Mapping erstellen, das eine Datenquelle liest, die Sprachen in den Daten klassifiziert und die Daten anhand der enthaltenen Sprachen in die Ziele schreibt.

Die folgende Abbildung zeigt das Mapping im Developer-Tool:



Das Mapping, das Sie erstellt haben, enthält die folgenden Objekte:

Objektname	Beschreibung
Read_tweet_user_lang	Datenquelle. Enthält die Twitter-Meldungen
Klassifizierer	Klassifizierumwandlung. Identifiziert die Sprachen, die in den Twitter-Meldungen verwendet werden.
Router	Routerumwandlung Leitet die Twitter-Meldungen für Datenzielobjekte entsprechend den Sprachen weiter, die sie enthalten.
Write_en_tweets_out	Datenziel. Enthält Twitter-Meldungen in englischer Sprache.
Write_other_tweets_out	Datenziel. Enthält Twitter-Meldungen in nicht englischer Sprache.

Beispiel Eingabedaten

Die folgenden Datenausschnitte zeigen ein Beispiel der Twitter-Daten, die Sie im Mapping analysieren:

```
Twitter Message

RT @GanaphoneS3: Faltan 10 minutos para la gran rifa de un iPhone 5...
RT @Clarified: How to Downgrade Your iPhone 4 From iOS 6.x to iOS 5.x (Mac)...
RT @jerseyjazz: The razor was the iPhone of the early 2000s
RT @KrissiDevine: Apple Pie that I made for Thanksgiving. http://t.com/s9ImzFxO
RT @sophieHz: Dan yang punya 2 kupon undian. Masuk dalam kotak undian yang berhadiah Samsung
RT @IsabelFreitas: o galaxy tem isso isso isso e a bateria Ã melhor que do iPhone
RT @PremiusIpad: Faltan 15 minutos para la gran rifa de un iPhone 5...
RT @payyton3: I want apple cider
RT @wiesteronder: Retweet als je iets van Apple, Nike, Adidas of microsoft hebt!
```

Konfiguration der Datenquelle

Die Datenquelle enthält einen einzigen Port. Jede Zeile für den Port enthält eine einzige Twitter-Meldung.

In der folgenden Tabelle wird die Konfiguration der Datenquelle beschrieben:

Portname	Porttyp	Genauigkeit
Text	n/v	200

Konfiguration der Klassifizierumwandlung

Die Klassifizierumwandlung verwendet einen einzigen Eingabe- und Ausgabeport. Der Umwandlungseingabeport liest das Textfeld aus der Datenquelle aus. Der Ausgabeport enthält die Sprache, die für jede Twitter-Meldung in dem Textfeld erkannt wurde. Die Klassifizierumwandlung verwendet ISO-Ländercodes, um die Sprache zu identifizieren.

In der folgenden Tabelle wird die Konfiguration der Klassifizierumwandlung beschrieben:

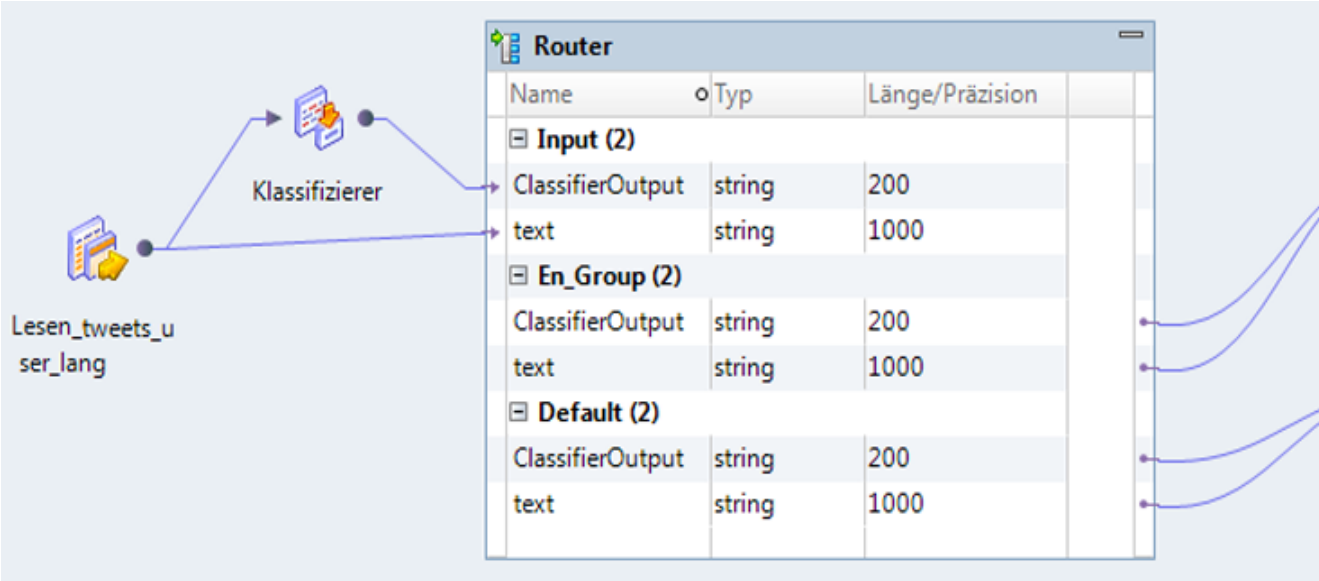
Portname	Porttyp	Genauigkeit	Strategie
text_input	Eingabe	200	Klassifizierer 1
Classifier_Output	Ausgabe	2	Klassifizierer 1

Konfiguration der Routerumwandlung

Die Routerumwandlung verwendet zwei Eingabeports. Es werden die Twitter-Meldungen aus der Datenquelle und die ISO-Ländercodes aus der Klassifizierumwandlung ausgelesen. Die Routerumwandlung leitet die

Daten an die Eingabeports zu unterschiedlichen Ausgabeports weiter, abhängig von den Bedingungen, die Sie angegeben haben.

Die folgende Abbildung zeigt die Portgruppen und Portverbindungen der Routerumwandlung:



In der folgenden Tabelle wird die Konfiguration der Routerumwandlung beschrieben:

Portname	Porttyp	Portgruppe	Genauigkeit
Classifier_Output	Eingabe	Eingabe	2
Text	Eingabe	Eingabe	200
Classifier_Output	Eingabe	Voreingestellter Wert	2
Text	Eingabe	Voreingestellter Wert	200
Classifier_Output	Eingabe	En_Group	2
Text	Eingabe	En_Group	200

Konfigurieren Sie die Umwandlung, um einen Datenstrom für englischsprachige Meldungen und Meldungen in anderen Sprachen zu erstellen. Fügen Sie zum Erstellen eines Datenstroms eine Ausgabeportgruppe zu der Umwandlung hinzu. Verwenden Sie bei der Umwandlung die Optionen **Gruppen** , um die Portgruppe hinzuzufügen.

Um zu bestimmen, wie die Umwandlung Daten an jeden Datenstrom weiterleitet, wird eine Bedingung für eine Portgruppe definiert. Die Bedingung identifiziert einen Port und gibt einen möglichen Wert für den Port an. Wenn die Umwandlung einen Eingabeportwert findet, der mit der Bedingung übereinstimmt, werden die Eingabedaten an die Portgruppe weitergeleitet, für die die Bedingung gilt.

Definieren Sie die folgende Bedingung auf En_Group:

```
ClassifierOutput='en'
```

Hinweis: Die Routerumwandlung liest Daten aus zwei Objekten im Mapping. Die Umwandlung kann die Daten in jeder Ausgabegruppe kombinieren, weil die Zeilensequenz, die in den Datenobjekten definiert ist, nicht geändert wird.

Konfiguration des Datenziels

Das Mapping enthält ein Datenziel für englischsprachige Twitter-Meldungen und ein Ziel für Meldungen in anderen Sprachen. Verbinden Sie die Ports aus einer Ausgabegruppe der Routerumwandlung mit einem Datenziel.

In der folgenden Tabelle wird die Konfiguration des Datenziels beschrieben:

Portname	Porttyp	Genauigkeit
Text	n/v	200
Classifier_Output	n/v	2

Ergebnis des Klassifizier-Mappings

Beim Ausführen des Mappings identifiziert die Klassifizierumwandlung die Sprache für jede Twitter-Meldung. Die Routerumwandlung schreibt den Meldungstext in die Datenziele, abhängig von den Sprachklassifikationen.

Der folgende Datenausschnitt zeigt ein Beispiel der englischsprachigen Zieldaten:

```
ISO Country CodeTwitter Message

en              RT @Clarified: How to Downgrade Your iPhone 4 From iOS 6.x to iOS 5.x
                (Mac)...

en              RT @jerseyjazz: The razor was the iPhone of the early 2000s

en              RT @KrissiDevine: Apple Pie that I made for Thanksgiving. http://t.com/
                s9ImzFxO

en              RT @payyton3: I want apple cider
```

Der folgende Datenausschnitt zeigt ein Beispiel der Zieldaten, die für andere Sprachen identifiziert wurden:

```
ISO Country     Twitter Message
Code

es              RT @GanaphoneS3: Faltan 10 minutos para la gran rifa de un iPhone 5...

id              RT @sophieHz: Dan yang punya 2 kupon undian. Masuk dalam kotak undian yang
                berhadiah Samsung Champ.

pt              RT @IsabelFreitas: o galaxy tem isso isso isso e a bateria Ã melhor que do
                iPhone

es              RT @PremiusIpad: Faltan 15 minutos para la gran rifa de un iPhone 5...

nl              RT @wiesteronder: Retweet als je iets van Apple, Nike, Adidas of microsoft
                hebt! http://t.co/Je6Ts00H
```

KAPITEL 8

Vergleichsumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Vergleichsumwandlung - Übersicht, 147
- Feld-Matching-Strategien, 147
- Identitäts-Matching-Strategien, 150
- Konfigurieren einer Vergleichsstrategie, 151
- Vergleichsumwandlung – Erweiterte Eigenschaften, 151

Vergleichsumwandlung - Übersicht

Die Vergleichsumwandlung ist eine passive Umwandlung, die die Ähnlichkeit zwischen Paaren von Eingabezeichenfolgen evaluiert und den Grad der Ähnlichkeit für jedes Paar als Punktzahl berechnet.

Beim Konfigurieren der Umwandlung wählen Sie ein Eingabespaltenpaar aus und weisen ihm eine Matching-Strategie zu.

Die Vergleichsumwandlung gibt Übereinstimmungswerte in einem Bereich von `0` bis `1` aus, wobei `1` eine genaue Übereinstimmung angibt.

Hinweis: Die in der Vergleichsumwandlung verfügbaren Strategien stehen auch in der Match-Umwandlung bereit. Verwenden Sie die Vergleichsumwandlung zum Definieren von Match-Vergleichsoperationen, die einem passenden Mapplet hinzugefügt werden sollen. Sie können dem Mapplet mehrere Vergleichsumwandlungen hinzufügen. Verwenden Sie die Match-Umwandlung zum Definieren von Match-Vergleichen in einer einzelnen Umwandlung. Sie können ein passendes Mapplet in eine Match-Umwandlung einbetten.

Feld-Matching-Strategien

Die Vergleichsumwandlung enthält vordefinierte Feld-Matching-Strategien, die Paare von Eingabedatenfeldern vergleichen.

Bigramm

Verwenden Sie den Bigramm-Algorithmus, um lange Textzeichenfolgen zu vergleichen, wie z. B. Anschriften, die in ein einzelnes Feld eingegeben wurden.

Der Bigramm-Algorithmus berechnet einen Übereinstimmungswert für zwei Datenzeichenfolgen basierend auf dem Vorkommen aufeinander folgender Zeichen in beiden Zeichenfolgen. Der Algorithmus sucht nach Paaren aufeinander folgender Zeichen, die in beiden Zeichenfolgen identisch sind. Er dividiert die Anzahl der Paare, die in beiden Zeichenfolgen übereinstimmen, durch die Gesamtanzahl der Zeichenpaare.

Bigramm - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `larder`
- `lerder`

Diese Zeichenfolgen ergeben die folgenden Bigramm-Gruppen:

```
l a, a r, r d, d e, e r
l e, e r, r d, d e, e r
```

Beachten Sie, dass für das zweite Vorkommen der Zeichenfolge "`e r`" innerhalb der Zeichenfolge "`lerder`" keine Übereinstimmung gefunden wird, da kein zweites Vorkommen von "`e r`" in der Zeichenfolge "`larder`" vorhanden ist.

Zur Berechnung des Bigramm-Übereinstimmungswerts dividiert die Umwandlung die Anzahl der übereinstimmenden Paare (6) durch die Gesamtzahl der Paare in beiden Zeichenfolgen (10). In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 60% auf und der Übereinstimmungswert lautet `0,60`.

Hamming-Abstand

Verwenden Sie den Algorithmus für den Hamming-Abstand, wenn die Position der Datenzeichen einen wichtigen Faktor darstellt, beispielsweise in numerischen oder Codefeldern, wie Telefonnummern, Postleitzahlen oder Produktcodes.

Der Algorithmus für den Hamming-Abstand berechnet die Übereinstimmung für zwei Zeichenfolgen, indem die Anzahl der Positionen ermittelt wird, an denen sich die Zeichen zwischen den Zeichenfolgen unterscheiden. Bei Zeichenfolgen mit unterschiedlicher Länge wird jedes zusätzliche Zeichen in der längsten Zeichenfolge als Unterschied zwischen den Zeichenfolgen gewertet.

Hamming-Abstand - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `Morlow`
- `M`**`a`**`rlow`**`es`**

Die markierten Zeichen geben die Positionen an, die vom Hamming-Algorithmus als verschieden angegeben werden.

Zur Berechnung des Hamming-Übereinstimmungswerts wird bei der Umwandlung die Anzahl der übereinstimmenden Zeichen (5) durch die Länge der längsten Zeichenfolge (8) dividiert. In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 62,5% auf und der Übereinstimmungswert lautet `0.625`.

Abstand bearbeiten

Verwenden Sie den Algorithmus "Abstand bearbeiten", um Wörter oder kurze Textzeichenfolgen, wie z. B. Namen, zu vergleichen.

Der Algorithmus "Abstand bearbeiten" berechnet die minimalen "Kosten" der Umwandlung einer Zeichenfolge in eine andere Zeichenfolge durch Einfügen, Löschen, oder Ersetzen von Zeichen.

Abstand bearbeiten - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `Levenston`
- `Levens**h**t**ei**n`

Die markierten Zeichen geben die Operationen an, die zum Umwandeln einer Zeichenfolge in eine andere Zeichenfolge benötigt werden.

Der Algorithmus "Abstand bearbeiten" dividiert die Anzahl der unveränderten Zeichen (8) durch die Länge der längsten Zeichenfolge (11). In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 72,7% auf und der Übereinstimmungswert lautet `0.727`.

Jaro-Abstand

Verwenden Sie den Algorithmus für den Jaro-Abstand, um zwei Zeichenfolgen miteinander zu vergleichen, wenn die Ähnlichkeit der ersten Zeichen in den Zeichenfolgen eine Priorität darstellt.

Der Übereinstimmungswert „Jaro-Abstand“ gibt den Grad der Übereinstimmung zwischen den ersten vier Zeichen beider Zeichenfolgen und der Anzahl erkannter Zeichentranspositionen an. Die Umwandlung wichtet die Bedeutung der Übereinstimmung zwischen den ersten vier Zeichen mithilfe des Werts, den Sie in der Eigenschaft Strafe eingegeben haben.

Jaro-Abstand - Eigenschaften

Beim Konfigurieren des Algorithmus für den Jaro-Abstand können folgende Eigenschaften konfiguriert werden:

Strafe

Bestimmt die Strafe für den Übereinstimmungswert, wenn die ersten vier Zeichen in zwei verglichenen Zeichenfolgen nicht identisch sind. Wenn das erste Zeichen nicht übereinstimmt, wird bei der Umwandlung der komplette Strafwert abgezogen. Bei der Umwandlung werden Teile des Strafwerts basierend auf der Position der anderen nicht übereinstimmenden Zeichen abgezogen. Der Standardstrafwert lautet `0,20`.

Groß-/Kleinschreibung

Bestimmt, ob der Algorithmus für den Jaro-Abstand beim Vergleichen der Zeichen die Groß-/Kleinschreibung beachtet.

Jaro-Abstand - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `391859`
- `813995`

Wenn Sie den standardmäßigen Straf-wert von `0,20` zum Analysieren dieser Zeichenfolgen verwenden, gibt der Algorithmus für den Jaro-Abstand einen Übereinstimmungswert von `0,513` zurück. Dieser Übereinstimmungswert gibt an, dass die Zeichenfolgen eine Ähnlichkeit von 51,3% aufweisen.

Hamming-Abstand umkehren

Verwenden Sie den Algorithmus "Hamming-Abstand umkehren", um den Prozentsatz der Zeichenpositionen zu berechnen, die zwischen zwei Zeichenfolgen von rechts nach links unterschiedlich sind.

Der Algorithmus für den Hamming-Abstand berechnet die Übereinstimmung für zwei Zeichenfolgen, indem die Anzahl der Positionen ermittelt wird, an denen sich die Zeichen zwischen den Zeichenfolgen unterscheiden. Bei Zeichenfolgen unterschiedlicher Länge wertet der Algorithmus alle zusätzlichen Zeichen in der längsten Zeichenfolge als Unterschied zwischen den Zeichenfolgen.

Hamming-Abstand umkehren - Beispiel

Beachten Sie die folgenden Zeichenfolgen, die von rechts nach links angeordnet sind, um den Algorithmus "Hamming-Abstand umkehren" zu simulieren:

- 1-999-9999
- **011-0**1-999-999**1**

Die markierten Zeichen zeigen die Positionen an, die vom Algorithmus "Hamming-Abstand umkehren" als unterschiedlich erkannt wurden.

Zur Berechnung des Übereinstimmungswerts für "Hamming-Abstand umkehren" wird bei der Umwandlung die Anzahl der übereinstimmenden Zeichen (9) durch die Länge der längsten Zeichenfolge (15) dividiert. In diesem Beispiel liegt der Übereinstimmungswert bei `0,6`, was angibt, dass die Zeichenfolgen eine Ähnlichkeit von 60% aufweisen.

Identitäts-Matching-Strategien

Die Vergleichsumwandlung enthält vordefinierte Identitäts-Matching-Strategien, die zum Auffinden von Übereinstimmungen für Personen, Adressen oder Unternehmenseinheiten verwendet werden können.

In der folgenden Tabelle wird die Übereinstimmungsoperation beschrieben, die von jeder Identitätsvergleichsstrategie durchgeführt wird:

Identitäts-Matching-Strategie	Übereinstimmungsoperation
Adresse	Gibt eine Adressübereinstimmung an.
Ansprechpartner	Gibt eine Kontaktperson innerhalb eines Unternehmens an einem Standort an.
Unternehmenseinheit	Gibt ein Unternehmen gemäß seines rechtlichen Unternehmensnamens an.
Division	Gibt ein Unternehmen unter einer Adresse an.
Familie	Gibt eine Familie nach Familienname und Adresse oder Telefonnummer an.
Felder	Gibt benutzerdefinierte von Ihnen ausgewählte Felder an.
Haushalt	Gibt Mitglieder einer Familie am selben Wohnort an.
Person	Gibt eine Person nach Name und ID oder Geburtsdatum an.
Unternehmen	Gibt ein Unternehmen nach Name an.

Identitäts-Matching-Strategie	Übereinstimmungsoperation
Personenname	Gibt eine Person nach Name an.
Bewohner	Gibt eine Person unter einer Adresse an.
Wide Contact	Gibt eine Kontaktperson in einem Unternehmen unabhängig vom Standort an.
Wide Household	Gibt Mitglieder derselben Familie unabhängig vom Standort an.

Hinweis: Identitäts-Matching-Strategien lesen Referenzdaten-Dateien, die als **Populationen** bezeichnet werden. Wenden Sie sich an den Informatica Administrator, um Informationen zu den auf Ihrem System installierten Populationsdatendateien zu erhalten.

Konfigurieren einer Vergleichsstrategie

Um eine Vergleichsstrategie zu konfigurieren, ändern Sie die Einstellungen in der Ansicht **Strategien** der Vergleichsumwandlung.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Wählen Sie eine Vergleichsstrategie aus dem Abschnitt **Strategien**.
3. Doppelklicken Sie in dem Abschnitt **Felder** auf die Spalte **Verfügbare Felder**, um eine Eingabe auszuwählen.

   Hinweis: Sie müssen eine Eingabe für jede Zeile auswählen, die einen fett formatierten Eingabenamen in der Spalte **Eingabefelder** anzeigt.

Vergleichsumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Vergleichsumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 9

Konsolidierungsumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Konsolidierungsumwandlung - Übersicht, 152
- Konsolidierungs-Mappings, 153
- Konsolidierungsumwandlung - Ports, 153
- Konsolidierungsumwandlung - Ansichten, 154
- Einfache Strategien, 156
- Zeilenbasierte Strategien, 157
- Erweiterte Strategien, 158
- Einfache Konsolidierungsfunktionen, 158
- Zeilenbasierte Konsolidierungsfunktionen, 163
- Konsolidierungs-Mapping - Beispiel, 167
- Konfigurieren einer Konsolidierungsumwandlung, 168

Konsolidierungsumwandlung - Übersicht

Die Konsolidierungsumwandlung ist eine aktive Umwandlung, die Gruppen von verwandten Datensätzen analysiert und einen konsolidierten Datensatz für jede Gruppe erstellt. Verwenden Sie die Konsolidierungsumwandlung, um von Umwandlungen erstellte Datensatzgruppen zu konsolidieren, wie zum Beispiel Schlüsselgenerator-, Match- und Assoziations-Umwandlungen.

Die Konsolidierungsumwandlung generiert konsolidierte Datensätze, indem Sie Strategien auf Gruppen und verwandte Datensätze anwendet. Die Umwandlung enthält einen Ausgabeport, der angibt, welcher Datensatz der konsolidierte Datensatz ist. Sie können die Umwandlungsausgabe auf konsolidierte Datensätze beschränken.

Sie können zum Beispiel Gruppen mit doppelten Mitarbeiterdatensätzen konsolidieren, die die Match-Umwandlung generiert. Die Konsolidierungsumwandlung kann einen konsolidierten Datensatz erstellen, der zusammengeführte Daten aus allen Datensätzen in der Gruppe enthält.

Sie können die Konsolidierungsumwandlung konfigurieren, um verschiedene Strategien basierend auf Ihren Konsolidierungsanforderungen zu verwenden. Verwenden Sie einfache Strategien zum Erstellen eines konsolidierten Datensatzes aus mehreren Datensätzen. Wenn Sie einfache Strategien verwenden, geben Sie eine Strategie für jeden Port an. Verwenden Sie eine zeilenbasierte Strategie, um Zeilen in der Datensatzgruppe zu analysieren und um einen konsolidierten Datensatz mit Werten aus einer der Zeilen zu erstellen. Verwenden Sie eine erweiterte Strategie, um einen konsolidierten Datensatz mit einem von Ihnen erstellten Ausdruck zu erstellen.

Konsolidierungs-Mappings

Um konsolidierte Datensätze zu erstellen, erstellen Sie ein Mapping, das Gruppen mit verwandten Datensätzen erstellt. Fügen Sie eine Konsolidierungsumwandlung zu einem Mapping hinzu, und konfigurieren Sie die Umwandlung zum Konsolidieren jeder Datensatzgruppe in einem einzigen Master-Datensatz.

Verbinden Sie eine Konsolidierungsumwandlung entsprechend den Geschäftszielen und Datenanforderungen mit anderen Umwandlungen. Um abgeglichene Datensätze zu konsolidieren, können Sie die Konsolidierungsumwandlung mit einer Match-Umwandlung verbinden. Um Datensätze als Teil der Verwaltung von Ausnahmedatensätzen zu konsolidieren, verbinden Sie die Konsolidierungsumwandlung mit einer Ausnahmeumwandlung. Wenn Sie beispielsweise eine Schlüsselgeneratorumwandlung für Gruppendatensätze verwenden, können Sie eine Konsolidierungsumwandlung direkt mit der Schlüsselgeneratorumwandlung verbinden. Die Konsolidierungsumwandlung erstellt einen konsolidierten Datensatz für jede Gruppe, die durch die Schlüsselgeneratorumwandlung erstellt wird.

Mapping-Ausgabe in nativen und in Hadoop-Umgebungen

Beim Ausführen eines Konsolidierungs-Mappings in einer nativen und in einer Hadoop-Umgebung kann die Konsolidierungsumwandlung zu unterschiedlichen Ergebnissen führen. Da das Mapping auf mehreren Knoten in Hadoop ausgeführt wird, können die Eingabedatensätze auf andere Weise in die Konsolidierungsumwandlung eingehen als in die native Umgebung. Für den gleichen Eingabedatensatz können daher bei der Umwandlung letztendlich unterschiedliche Regelsätze in jeder Umgebung generiert werden. Die Umwandlungsberechnungen und die konsolidierten Ergebnisse sind für die Reihenfolge der Eingabezeilen in jedem Fall genau.

Damit dieselben Datensätze sowohl für native als auch für Hadoop-Umgebungen generiert werden, konfigurieren Sie die Konsolidierungsumwandlung so, dass die Datensätze in folgender Reihenfolge sortiert werden:

- Sortieren Sie die Datensätze zuerst nach dem Port „Gruppieren nach“.
- Sortieren Sie die Datensätze danach in der Reihenfolge, in der die Eingabeports in der Umwandlung angezeigt werden.

Konsolidierungsumwandlung - Ports

Das Developer Tool erstellt einen Ausgabeport für jeden von Ihnen hinzugefügten Eingabeport. Sie können der Umwandlung die Ausgabeports nicht manuell hinzufügen. Die Konsolidierungsumwandlung enthält auch einen **IsSurvivor**-Ausgabeport, der die konsolidierten Datensätze angibt.

Einer der von Ihnen zur Konsolidierungsumwandlung hinzugefügten Eingabeports muss Gruppenschlüssel enthalten. Für die Konsolidierungsumwandlung sind Gruppenschlüsselinformationen erforderlich, weil Konsolidierungsstrategien Datensatzgruppen anstatt vollständige Datensätze verarbeiten.

Wenn Sie einen Eingabeport hinzufügen, erstellt das Developer Tool einen Ausgabeportnamen, indem er dem Eingabeportnamen den Suffix "1" hinzufügt. Die Umwandlung enthält außerdem den **IsSurvivor**-Ausgabeport, der angibt, ob es sich bei einem Datensatz um den konsolidierten Datensatz handelt. Für die konsolidierten Datensätze schreibt die Konsolidierungsumwandlung den Sting "Y" in den **IsSurvivor**-Port. Für Eingabedatensätze schreibt die Konsolidierungsumwandlung den String "n" in den **IsSurvivor**-Port.

Konsolidierungsumwandlung - Ansichten

Die Konsolidierungsumwandlung enthält Ansichten für Ports, Strategien und erweiterte Eigenschaften.

Konsolidierungsumwandlung - Strategieansichten

Die Ansicht **Strategien** enthält Eigenschaften für einfache, zeilenbasierte und erweiterte Strategien.

In der folgenden Liste werden die Konsolidierungsstrategietypen beschrieben:

Einfache Strategie

Eine einfache Strategie analysiert alle Werte eines Ports in einem Datensatz und wählt einen Wert aus. Sie geben eine einfache Strategie für jeden Port an. Die Konsolidierungsumwandlung verwendet die von allen einfachen Strategien ausgewählten Portwerte, um einen konsolidierten Datensatz zu erstellen. Beispiele für einfache Strategien beinhalten den häufigsten Wert in einem Port, den längsten Wert in einem Port oder den häufigsten Nicht-Leerzeichen-Wert in einem Port.

Zeilenbasierte Strategie

Eine zeilenbasierte Strategie analysiert Zeilen in der Datensatzgruppe und wählt eine Zeile aus. Die Konsolidierungsumwandlung verwendet die Portwerte aus dieser Zeile zum Erstellen eines konsolidierten Datensatzes. Beispiele für zeilenbasierte Strategien enthalten die höchste Anzahl an Zeichen, die niedrigste Anzahl an leeren Feldern und die höchste Anzahl der am meisten frequentierten Felder.

Erweiterte Strategie

Eine erweiterte Strategie analysiert eine Datensatzgruppe mithilfe der von Ihnen definierten Strategien. Sie erstellen erweiterte Strategien mithilfe von Konsolidierungsfunktionen in einem Ausdruck. Die Konsolidierungsumwandlung erstellt einen konsolidierten Datensatz basierend auf der Ausgabe des Ausdrucks. Der von Ihnen erstellte Ausdruck kann ebenfalls alle in der Entscheidungsumwandlung verfügbaren Funktionen verwenden.

Konsolidierungsumwandlung – Erweiterte Eigenschaften

Die Konsolidierungsumwandlung enthält erweiterte Eigenschaften, die das Sortierungsverhalten, den Ausgabemodus, das Verhalten des Cache-Arbeitsspeichers und die Tracingebene bestimmen.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Sortieren

Legt fest, ob die Umwandlung die Eingabezeilen in den **Gruppieren nach**-Portdaten sortiert. Die Eigenschaft ist standardmäßig aktiviert.

Wählen Sie die Eigenschaft aus, wenn die Eingabezeilen nicht vorsortiert sind.

Sortierung mit Unterscheidung von Groß-/Kleinschreibung

Legt fest, ob bei der Sortierung die Groß-/Kleinschreibung beachtet werden muss. Die Eigenschaft ist standardmäßig aktiviert.

Ausgabemodus

Legt fest, ob die Umwandlung alle Datensätze als Ausgabe oder die konsolidierten Datensätze als Ausgabe schreibt. Der Standardwert ist „Alle“.

Cache-Dateiverzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst temporäre Daten für die aktuelle Umwandlung schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis, wenn das Volumen der Eingabedaten größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cache-Dateigröße

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst zum Sortieren der Eingabedaten bei der Umwandlung verwendet. Der Standardwert ist 400.000 Byte. Sie können einen Parameter verwenden, um die Cache-Dateigröße anzugeben.

Bevor die Daten sortiert werden, weist der Datenintegrationsdienst die von Ihnen angegebene Speichermenge zu. Wenn der Sortiervorgang eine größere Menge an Daten generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Dateiverzeichnis. Wenn der Speicherbedarf für den Sortiervorgang die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Cache-Dateigröße

Die Eigenschaft für die Cache-Dateigröße bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst der Konsolidierungsumwandlung für Sortiervorgänge zuweist. Konfigurieren Sie die Eigenschaft mit einem Wert kleiner oder gleich der Menge an Arbeitsspeicher auf dem Computer, der den Datenintegrationsdienst hostet.

Um eine optimale Leistung zu erzielen, geben Sie eine Cache-Dateigröße von mindestens 16 MB an.

Bevor ein Sortiervorgang gestartet wird, weist der Datenintegrationsdienst den Arbeitsspeicher zu, den die Eigenschaft für die Cache-Dateigröße angibt. Vor dem Durchführen eines Sortiervorgangs übergibt der Datenintegrationsdienst alle Eingabedaten an die Konsolidierungsumwandlung.

Wenn das Volumen der Eingabedaten größer als die Cache-Dateigröße ist, schreibt der Datenintegrationsdienst Daten in das Cache-Dateiverzeichnis. Beim Schreiben von Daten in das Cache-Dateiverzeichnis belegt der Datenintegrationsdienst Speicherplatz, dessen Volumen das Volumen der Eingabedaten mindestens um das Doppelte übersteigt.

Verwenden Sie die folgende Formel, um die Größe der eingehenden Daten festzulegen:

```
[number_of_input_rows *( Sum (column_size) + 16]
```

In der folgenden Tabelle werden die möglichen Datentypen und die Spaltengrößenwerte aufgelistet, die in Cache-Datendateiberechnungen angewendet werden:

Datentyp	Spaltengröße
Binär	Genauigkeit + 8. Gerundet auf das nächstgelegene Mehrfache von 8
Datum/Uhrzeit	29
Dezimal, hohe Genauigkeit aus (Gesamtgenauigkeit)	16
Dezimal, hohe Genauigkeit ein (Genauigkeit <=18)	24
Dezimal, hohe Genauigkeit ein (Genauigkeit >18, <=28)	32
Dezimal, hohe Genauigkeit ein (Genauigkeit >28)	16
Dezimal, hohe Genauigkeit ein (negative Dezimalstellenzahl)	16
Doppel	16
Gleitkommazahl	16
Ganzzahl	16
Zeichenfolge, Text	Unicode-Modus: 2 * (Genauigkeit + 5) ASCII-Modus: Genauigkeit + 9

Einfache Strategien

Eine einfache Strategie analysiert einen Port in einer Datensatzgruppe und gibt einen Wert zurück. Sie geben eine einfache Strategie für jeden Port an. Die Konsolidierungsumwandlung verwendet die von allen einfachen Strategien ausgewählten Portwerte, um einen konsolidierten Datensatz zu erstellen.

Beim Konfigurieren einer Strategie in der Ansicht **Strategien** bei der Option Umwandlung wird der folgende Text als Konsolidierungsmethode von der Strategie angezeigt:

Standard verwenden

Der Standardwert der Strategie ist "Höchste Reihen-ID".

Sie können aus den folgenden einfachen Strategien auswählen:

Average

Analysiert einen Port in der Datensatzgruppe und gibt den Durchschnitt aller Werte zurück.

Für Zeichenfolge- und Datums-/Zeitdatentypen gibt die Strategie den am häufigsten auftretenden Wert zurück.

Longest

Analysiert einen Port in der Datensatzgruppe und gibt den Wert mit der höchsten Anzahl an Zeichen zurück. Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Zeichen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Maximal

Analysiert einen Port in der Datensatzgruppe und gibt den höchsten Wert zurück.

Für den String-Datentyp gibt die Strategie die längste Zeichenfolge zurück. Für den Date/time-Datentyp gibt die Strategie das aktuellste Datum zurück.

Minimum

Analysiert einen Port in der Datensatzgruppe und gibt den niedrigsten Wert zurück.

Für den String-Datentyp gibt die Strategie die kürzeste Zeichenfolge zurück. Für den Date/time-Datentyp gibt die Strategie das früheste Datum zurück.

Most frequent

Analysiert einen Port in der Datensatzgruppe und gibt den am häufigsten auftretenden Wert zurück. Dazu zählen auch Leerzeichen und Nullwerte. Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Vorkommen aufweisen, wird von der Strategie der erste qualifizierende Wert zurückgegeben.

Most frequent non-blank

Analysiert einen Port in der Datensatzgruppe und gibt den am häufigsten auftretenden Wert zurück. Leerzeichen und Nullwerte sind davon ausgeschlossen. Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Nicht-Leerzeichen-Vorkommen aufweisen, wird von der Strategie der erste qualifizierende Wert zurückgegeben.

Shortest

Analysiert einen Port in der Datensatzgruppe und gibt den Wert mit der niedrigsten Anzahl an Zeichen zurück. Wenn zwei oder mehr Werte dieselbe niedrigste Anzahl an Zeichen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Höchste Zeilen-ID

Analysiert einen Port in der Datensatzgruppe und gibt den Wert mit der höchsten Zeilen-ID zurück.

Zeilenbasierte Strategien

Eine zeilenbasierte Strategie analysiert Zeilen in der Datensatzgruppe und wählt eine Zeile aus. Die Konsolidierungsumwandlung verwendet die Portwerte aus dieser Zeile zum Erstellen eines konsolidierten Datensatzes. Die Standardstrategie lautet "most data".

Wählen Sie eine der folgenden zeilenbasierten Strategien aus:

Most data

Wählt die Zeile mit der höchsten Anzahl an Zeichen aus. Wenn zwei oder mehr Zeilen dieselbe Höchstanzahl an Zeichen aufweisen, gibt die Strategie den letzten Wert mit dieser Anzahl zurück.

Most filled

Wählt die Zeilen mit der höchsten Anzahl an Nicht-Leerzeichen-Spalten aus. Wenn zwei oder mehr Zeilen dieselbe Höchstanzahl an Nicht-Leerzeichen-Spalten aufweisen, gibt die Strategie den letzten Wert mit dieser Anzahl zurück.

Modal exact

Wählt die Zeilen mit der höchsten Anzahl an "most frequent non-blank"-Werten aus. Angenommen ein Zeile hat drei Ports, die die häufigsten Werte in der Datensatzgruppe enthalten. Die Anzahl der häufigsten Werte für diese Zeile ist "3."

Wenn zwei oder mehr Zeilen dieselbe Höchstanzahl an "most frequent non-blank"-Spalten aufweisen, gibt die Strategie den letzten Wert mit dieser Anzahl zurück.

Zeilenbasierte Strategie - Beispiel

In der folgenden Tabelle wird eine Beispiel-Datensatzgruppe angezeigt. Die letzte Spalte beschreibt die Ursachen dafür, warum bestimmte zeilenbasierte Strategien verschiedene Zeilen in dieser Datensatzgruppe auswählen.

Produkt-ID	Vorname	Last Name	Postleitzahl	Strategieauswahl
2106	Bartholomew		28516	Die "Most Data"-Strategie wählt diese Zeile aus, weil die Zeile mehr Zeichen als die anderen Zeilen enthält.
2236	Bart	Smith	28579	Die "Most Filled"-Strategie wählt diese Zeile aus, weil die Zeile mehr Nicht-Leerzeichen-Spalten als andere Zeilen aufweist.
2236	<Blank>	Smith	28516	Die "Modal Exact"-Strategie wählt diese Zeile aus, weil die Zeile die höchste Anzahl der häufigsten Werte enthält.

Erweiterte Strategien

Sie können erweiterte Strategien zum Erstellen von Konsolidierungsstrategien aus vordefinierten Funktionen verwenden. Sie können Konsolidierungsfunktionen und andere Informatica-Funktionen verwenden.

Sie können Ausdrücke erstellen, die einfache Konsolidierungsfunktionen oder zeilenbasierte Konsolidierungsfunktionen enthalten. Verwenden Sie die einfachen Konsolidierungsfunktionen zum Erstellen eines auf Portwerten in der Datensatzgruppe basierten konsolidierten Datensatzes. Verwenden Sie die zeilenbasierten Konsolidierungsfunktionen zum Auswählen einer Zeile aus der Datensatzgruppe.

Konsolidierungsausdrücke müssen alle Ausgabeports in der Konsolidierungsumwandlung füllen. Wenn die Konsolidierungsausdrücke nicht alle Ausgabeports verwenden, verursacht die Umwandlung das Fehlschlagen von Mappings.

Sie können eine einfache oder zeilenbasierte Strategie als Vorlage für eine erweiterte Strategie verwenden. Konfigurieren Sie eine einfache oder zeilenbasierte Strategie und wählen Sie "Erweitert" aus. Die Konsolidierungsumwandlung generiert einen Ausdruck mit Funktionen, die die Strategie ausführen. Sie können mehrere Funktionen zum Implementieren weiterer Anforderungen hinzufügen.

Einfache Konsolidierungsfunktionen

Einfache Konsolidierungsfunktionen wählen einen Wert aus allen Portwerten in einer Datensatzgruppe aus. Wenn Sie eine einfache Konsolidierungsfunktion verwenden, geben Sie die Funktion mit einem Port und dem Gruppierungsport an.

CONSOL_AVG

Analysiert einen Port in der Datensatzgruppe und gibt den Durchschnitt aller Werte zurück.

Syntax

```
CONSOL_AVG(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der Durchschnittswert aller Werte in einem Port.

Für String- und Date/time-Datentypen gibt die Funktion den am häufigsten auftretenden Wert zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_AVG`, um den Durchschnittswert des SalesTotal-Eingabeports zu ermitteln:

```
SalesTotal1:= CONSOL_AVG(SalesTotal, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_AVG` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den SalesTotal-Port und gibt den Durchschnittswert zurück. Der Ausdruck schreibt den Durchschnittswert in den SalesTotal1-Ausgabeport.

CONSOL_LONGEST

Analysiert einen Port in der Datensatzgruppe und gibt den Wert mit der höchsten Anzahl an Zeichen zurück.

Syntax

```
CONSOL_LONGEST(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der Portwert mit der höchsten Anzahl an Zeichen.

Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Zeichen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_LONGEST`, um den FirstName-Eingabeport zu analysieren und den Wert mit der höchsten Anzahl an Zeichen zu ermitteln:

```
FirstName1:= CONSOL_LONGEST(FirstName, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_LONGEST` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den FirstName-Port und gibt den längsten Wert zurück zurück. Der Ausdruck schreibt diesen Wert in den FirstName1-Ausgabeport.

CONSOL_MAX

Analysiert einen Port in der Datensatzgruppe und gibt den höchsten Wert zurück.

Syntax

```
CONSOL_MAX(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der höchste Portwert.

Für den String-Datentyp gibt die Funktion die längste Zeichenfolge zurück. Für den Date/time-Datentyp gibt die Funktion das aktuellste Datum zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MAX`, um den SalesTotal-Eingabeport zu analysieren und den höchsten Wert zu ermitteln:

```
SalesTotal1:= CONSOL_MAX(SalesTotal, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_MAX` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den SalesTotal-Port und gibt den höchsten Wert zurück. Der Ausdruck schreibt diesen Wert in den SalesTotal1-Ausgabeport.

CONSOL_MIN

Analysiert einen Port in der Datensatzgruppe und gibt den niedrigsten Wert zurück.

Syntax

```
CONSOL_MIN(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der niedrigste Portwert.

Für den String-Datentyp gibt die Funktion die kürzeste Zeichenfolge zurück. Für den Date/time-Datentyp gibt die Funktion das früheste Datum zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MIN`, um den SalesTotal-Eingabeport zu analysieren und den niedrigsten Wert zu ermitteln:

```
SalesTotal1:= CONSOL_MIN(SalesTotal, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_MIN` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den SalesTotal-Port und gibt den niedrigsten Wert zurück. Der Ausdruck schreibt diesen Wert in den SalesTotal1-Ausgabeport.

CONSOL_MOSTFREQ

Analysiert einen Port in der Datensatzgruppe und gibt den am häufigsten auftretenden Wert zurück. Dazu zählen auch Leerzeichen und Nullwerte.

Syntax

```
CONSOL_MOSTFREQ(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der am häufigsten auftretende Wert. Dazu zählen Leerzeichen oder Nullwerte.

Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Vorkommen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MOSTFREQ`, um den Company-Eingabeport zu analysieren und den am häufigsten auftretenden Wert zu ermitteln:

```
Company1:= CONSOL_MOSTFREQ(Company, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_MOSTFREQ` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den Company-Port und gibt den am häufigsten auftretenden Wert zurück. Der Ausdruck schreibt diesen Wert in den Company1-Ausgabeport.

CONSOL_MOSTFREQ_NB

Analysiert einen Port in der Datensatzgruppe und gibt den am häufigsten auftretenden Wert zurück. Leerzeichen und Nullwerte sind davon ausgeschlossen.

Syntax

```
CONSOL_MOSTFREQ_NB(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der am häufigsten auftretende Wert. Leerzeichen oder Nullwerte sind davon ausgeschlossen.

Wenn zwei oder mehr Werte dieselbe Höchstanzahl an Vorkommen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MOSTFREQ_NB`, um den Company-Eingabeport zu analysieren und den am häufigsten auftretenden Wert zu ermitteln:

```
Company1:= CONSOL_MOSTFREQ_NB(Company, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_MOSTFREQ_NB` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den Company-Port und gibt den am häufigsten auftretenden Wert zurück. Der Ausdruck schreibt diesen Wert in den Company1-Ausgabeport.

CONSOL_SHORTEST

Analysiert einen Port in der Datensatzgruppe und gibt den Wert mit der niedrigsten Anzahl an Zeichen zurück.

Syntax

```
CONSOL_SHORTEST(string, group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

Der Portwert mit der niedrigsten Anzahl an Zeichen.

Wenn zwei oder mehr Werte dieselbe niedrigste Anzahl an Zeichen aufweisen, gibt die Strategie den ersten Wert mit dieser Anzahl zurück.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_SHORTEST`, um den FirstName-Eingabeport zu analysieren und den Wert mit der niedrigsten Anzahl an Zeichen zu ermitteln:

```
FirstName1:= CONSOL_SHORTEST(FirstName, GroupKey)
```

In diesem Ausdruck verwendet die Funktion `CONSOL_SHORTEST` den GroupKey-Port zur Kennzeichnung einer Datensatzgruppe. In dieser Datensatzgruppe analysiert die Funktion den FirstName-Port und gibt den kürzesten Wert zurück zurück. Der Ausdruck schreibt diesen Wert in den FirstName1-Ausgabeport.

Zeilenbasierte Konsolidierungsfunktionen

Verwenden Sie die zeilenbasierten Konsolidierungsfunktionen, um einen Datensatz in einer Datensatzgruppe auszuwählen. Sie müssen zeilenbasierte Konsolidierungsfunktionen innerhalb von `IF-THEN-ELSE`-Anweisungen verwenden.

CONSOL_GETROWFIELD

Liest die durch eine zeilenbasierte Konsolidierungsfunktion gekennzeichnete Zeile und gibt den Wert für den von Ihnen angegebenen Port zurück. Verwenden Sie ein numerisches Argument, um einen Port anzugeben.

Sie müssen die Funktion `CONSOL_GETROWFIELD` in Verbindung mit einer der folgenden zeilenbasierten Konsolidierungsfunktionen verwenden:

- `CONSOL_MODALEXACT`
- `CONSOL_MOSTDATA`
- `CONSOL_MOSTFILLED`

Für jeden Eingabeport in einer zeilenbasierten Konsolidierungsfunktion müssen Sie eine Instanz der Funktion `CONSOL_GETROWFIELD` verwenden.

Syntax

```
CONSOL_GETROWFIELD(value)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/ Optional	Beschreibung
wert	Erforderlich	Zahl, die einen Eingabeport in der zeilenbasierten Konsolidierungsfunktion angibt. Verwenden Sie "0", um den Port ganz links in der Funktion anzugeben. Verwenden Sie nachfolgende Zahlen, um andere Ports anzugeben.

Rückgabewert

Der Wert des von Ihnen angegebenen Ports. Die Funktion liest diesen Wert aus einer von einer zeilenbasierten Konsolidierungsfunktion gekennzeichneten Zeile.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_GETROWFIELD` in Verbindung mit der Funktion `CONSOL_MOSTDATA`:

```
IF (CONSOL_MOSTDATA(First_Name,Last_Name,GroupKey,GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

In diesem Ausdruck analysiert die Funktion `CONSOL_MOSTDATA` Zeilen in einer Datensatzgruppe und kennzeichnet eine einzelne Zeile. Die Funktionen `CONSOL_GETROWFIELD` verwenden aufeinander folgende Zahlen zum Lesen der Portwerte für diese Zeile und schreibt die Werte in die Ausgabeports.

CONSOL_MODALEXACT

Kennzeichnet die Zeilen mit der höchsten Anzahl der am häufigsten vorkommenden Werte.

Angenommen ein Zeile hat drei Ports, die die häufigsten Werte in der Datensatzgruppe enthalten. Die Anzahl der häufigsten Werte für diese Zeile ist "3."

Sie müssen diese Funktion in Verbindung mit der Funktion `CONSOL_GETROWFIELD` verwenden. Die Funktion `CONSOL_GETROWFIELD` gibt die Werte aus der Zeile zurück, die die Funktion `CONSOL_MODALEXACT` kennzeichnet.

Syntax

```
CONSOL_MODALEXACT(string1, [string2, ..., stringN,]
 group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

TRUE für die Zeile, die die höchste Anzahl der am häufigsten auftretenden Felder erzielt, FALSE für alle anderen Zeilen.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MODALEXACT`, um die Zeile zu suchen, die die höchste Anzahl der am häufigsten auftretenden Felder enthält:

```
IF (CONSOL_MODALEXACT(First_Name,Last_Name,GroupKey,GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

In diesem Ausdruck analysiert die Funktion `CONSOL_MODALEXACT` Zeilen in einer Datensatzgruppe und kennzeichnet eine einzelne Zeile. Die Funktionen `CONSOL_GETROWFIELD` verwenden aufeinander folgende Zahlen zum Lesen der Portwerte für diese Zeile und schreibt die Werte in die Ausgabeports.

CONSOL_MOSTDATA

Kennzeichnet die Zeile, die die meisten Zeichen in allen Ports enthält.

Sie müssen diese Funktion in Verbindung mit der Funktion `CONSOL_GETROWFIELD` verwenden. Die Funktion `CONSOL_GETROWFIELD` gibt die Werte aus der Zeile zurück, die die Funktion `CONSOL_MOSTDATA` kennzeichnet.

Syntax

```
CONSOL_MOSTDATA(string1, [string2, ..., stringN,]
 group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

TRUE für die Zeile, die die meisten Zeichen in allen Ports enthält, FALSE für alle anderen Zeilen.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MOSTDATA`, um die Zeile zu suchen, die die meisten Zeichen enthält:

```
IF (CONSOL_MOSTDATA(First_Name,Last_Name,GroupKey,GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

In diesem Ausdruck analysiert die Funktion `CONSOL_MOSTDATA` Zeilen in einer Datensatzgruppe und kennzeichnet eine einzelne Zeile. Die Funktionen `CONSOL_GETROWFIELD` verwenden aufeinander folgende Zahlen zum Lesen der Portwerte für diese Zeile und schreibt die Werte in die Ausgabeports.

CONSOL_MOSTFILLED

Kennzeichnet die Zeile mit der höchsten Anzahl an Nicht-Leerzeichen-Feldern.

Sie müssen diese Funktion in Verbindung mit der Funktion `CONSOL_GETROWFIELD` verwenden. Die Funktion `CONSOL_GETROWFIELD` gibt die Werte aus der Zeile zurück, die die Funktion `CONSOL_MOSTFILLED` kennzeichnet.

Syntax

```
CONSOL_MOSTFILLED(string1, [string2, ..., stringN,]
 group by)
```

In der folgenden Tabelle werden die Argumente für diesen Befehl beschrieben:

Argument	Erforderlich/Optional	Beschreibung
string	Erforderlich	Name des Eingabeports.
group by	Erforderlich	Name des Eingabeports, der die Gruppen-ID enthält.

Rückgabewert

TRUE für die Zeile, die die höchste Anzahl an Nicht-Leerzeichen-Feldern enthält, FALSE für alle anderen Zeilen.

Beispiel

Der folgende Ausdruck verwendet die Funktion `CONSOL_MOSTFILLED`, um die Zeile zu suchen, die die meisten Zeichen enthält:

```
IF (CONSOL_MOSTFILLED(First_Name,Last_Name,GroupKey,GroupKey))
THEN
First_Name1 := CONSOL_GETROWFIELD(0)
Last_Name1 := CONSOL_GETROWFIELD(1)
GroupKey1 := CONSOL_GETROWFIELD(2)
ELSE
First_Name1 := First_Name
Last_Name1 := Last_Name
GroupKey1 := GroupKey
ENDIF
```

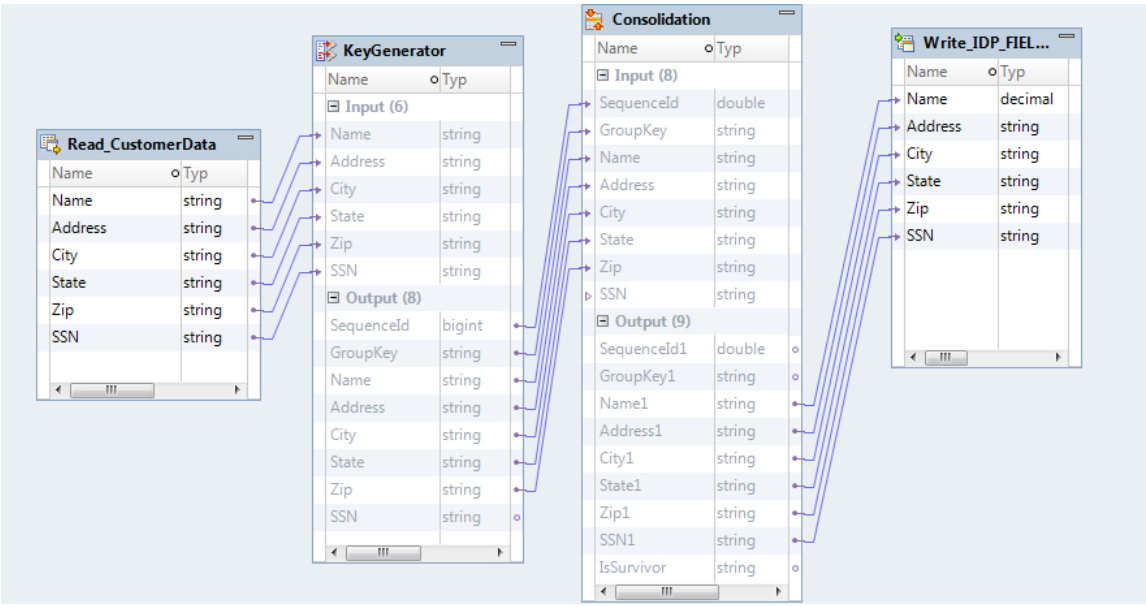
In diesem Ausdruck analysiert die Funktion `CONSOL_MOSTFILLED` Zeilen in einer Datensatzgruppe und kennzeichnet eine einzelne Zeile. Die Funktionen `CONSOL_GETROWFIELD` verwenden aufeinander folgende Zahlen zum Lesen der Portwerte für diese Zeile und schreibt die Werte in die Ausgabeports.

Konsolidierungs-Mapping - Beispiel

Ihr Unternehmen muss doppelte Kundendatensätze konsolidieren. Zum Konsolidieren der Kundendatensätze gruppieren Sie die Daten mit einer Schlüsselgeneratorumwandlung und verwenden die Konsolidierungsumwandlung, um die Datensätze zu konsolidieren.

Sie erstellen ein Mapping mit einer Datenquelle mit Kundendatensätzen, einer Schlüsselgeneratorumwandlung, einer Konsolidierungsumwandlung und einem Datenziel. Das Mapping gruppiert Kundendatensätze, konsolidiert die Gruppen und schreibt einen einzelnen konsolidierten Datensatz.

Die folgende Abbildung zeigt das Mapping:



Eingabedaten

Die Eingabedaten, die Sie analysieren möchten, enthalten Kundeninformationen.

Die folgende Tabelle enthält die Eingabedaten für dieses Beispiel:

Name	Adresse	Ort	Bundesland	PLZ	SSN
Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320
Dennis Jones	1000 Alberta Rd	New York	NY	10547	987-65-4320
D Jones	100 All Saints Ave	New York	NY	10547-1521	

Schlüsselgeneratorumwandlung

Verwenden Sie die Schlüsselgeneratorumwandlung, um die Eingabedaten basierend auf dem PLZ-Codeport zu gruppieren.

Die Umwandlung gibt die folgenden Daten zurück:

SequenceId	GroupKey	Name	Adresse	Ort	Bundesland	Postleitzahl	SSN
1	10547	Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320
2	10547	Dennis Jones	1000 Alberta Rd	New York	NY	10547	
3	10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320

Konsolidierungsumwandlung

Verwenden Sie die Konsolidierungsumwandlung, um den konsolidierten Datensatz zu generieren.

Konfigurieren Sie die Konsolidierungsumwandlungen, um den zeilenbasierten Strategietyp zu verwenden. Wählen Sie Modal Exact-Strategie aus, um die Zeile mit der höchsten Anzahl der am häufigsten auftretenden Werte auszuwählen. Die Modal Exact-Strategie verwendet die Werte aus dieser Zeile zum Generieren eines konsolidierten Datensatzes. Der konsolidierte Datensatz ist der Datensatz mit dem "Y"-Wert im IsSurvivor-Port.

Die Umwandlung gibt die folgenden Daten zurück:

GroupKey	Name	Adresse	Ort	Bundesland	Postleitzahl	SSN	IsSurvivor
10547	Dennis Jones	100 All Saints Ave	New York	NY	10547	987-65-4320	N
10547	Dennis Jones	1000 Alberta Rd	New York	NY	10547		N
10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320	N
10547	D Jones	100 All Saints Ave	New York	NY	10547-1521	987-65-4320	Y

Konsolidierungs-Mapping-Ausgabe

Konfigurieren Sie die Konsolidierungsumwandlung so, dass die Mapping-Ausgabe nur konsolidierte Datensätze enthält.

In diesem Beispiel sind Sie sich nahezu sicher, dass es sich bei den meisten am häufigsten vorkommenden Werte, die von der Modal Exact-Strategie ausgewählt wurden, um korrekte Portwerte handelt. Wenn Sie ausschließlich konsolidierte Datensätze in ein Mapping-Target schreiben möchten, wählen Sie die Ansicht **Erweitert** aus und stellen Sie den Ausgabemodus "Survivor Only" ein.

Wenn Sie das Mapping ausführen, enthält die Mapping-Ausgabe nur konsolidierte Datensätze.

Konfigurieren einer Konsolidierungsumwandlung

Beim Konfigurieren der Konsolidierungsumwandlung wählen Sie Strategietypen aus, wählen Strategien aus oder schreiben Ausdrücke, wählen einen Gruppierungsport aus und konfigurieren erweiterte Optionen.

1. Wählen Sie die Ansicht **Konsolidierung** aus.
2. Wählen Sie einen Strategietyp aus.
3. Konfigurieren Sie die Strategie.
   - Wählen Sie für den einfachen Strategietyps eine Strategie für jeden Port aus.
   - Wählen Sie für den zeilenbasierten Strategietyp eine Strategie aus.
   - Erstellen Sie für den erweiterten Strategietyp einen Ausdruck, der Konsolidierungsfunktionen verwendet.
4. Wählen Sie im Feld "Gruppieren nach" den Port aus, der die Gruppen-ID enthält.
5. Aktivieren Sie die Sortierung in der Ansicht **Erweitert**, wenn die Eingabedaten nicht sortiert sind.
6. Konfigurieren Sie die Ausgabe so, dass diese konsolidierte Datensätzen oder alle Datensätze enthalten kann.

KAPITEL 10

Datenmaskierungsumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Datenmaskierungsumwandlung - Übersicht, 169
- Maskierungstechniken, 170
- Maskierungsregeln, 181
- Spezielle Maskierungsformate, 186
- Standardwertdatei, 190
- Konfiguration der Datenmaskierungsumwandlung, 191
- Laufzeiteigenschaften der Datenmaskierungsumwandlung, 193
- Datenmaskierung - Beispiel, 194
- Datenmaskierungsumwandlung – Erweiterte Eigenschaften, 196

Datenmaskierungsumwandlung - Übersicht

Die Datenmaskierungsumwandlung verändert vertrauliche Produktionsdaten in realistische Testdaten für Nicht-Produktionsumgebungen. Die Datenmaskierungsumwandlung ändert Quelldaten auf der Grundlage von Maskierungstechniken, die Sie für jede Spalte konfigurieren.

Erstellen Sie maskierte Daten für Softwareentwicklung, Tests, Training und Data Mining. Sie können Datenbeziehungen in den maskierten Daten sowie die referenzielle Integrität zwischen Datenbanktabellen aufrechterhalten.

Die Datenmaskierungsumwandlung stellt Maskierungsregeln auf der Grundlage des Quelldatentyps und der Maskierungstechnik bereit, die für eine Spalte konfiguriert werden. Bei Strings können Sie einschränken, welche Zeichen in einem String ersetzt werden sollen. Sie können einschränken, welche Zeichen in einer Maske angewendet werden sollen. Hinsichtlich Zahlen und Daten können Sie einen Zahlenbereich für die maskierten Daten angeben. Sie können einen Bereich konfigurieren, der eine feste oder prozentuale Abweichung von der ursprünglichen Zahl darstellt. Der Datenintegrationsdienst ersetzt Zeichen auf der Grundlage des für die Umwandlung konfigurierten Gebietsschemas.

Maskierungstechniken

Die Maskierungstechnik ist der Typ der Datenmaskierung, der auf die ausgewählte Spalte angewendet werden soll.

Sie können eine der folgenden Maskierungstechniken für eine Eingabespalte auswählen:

Zufällig

Erzielt zufällige, nicht wiederholbare Ergebnisse für dieselben Quelldaten und Maskierungsregeln. Sie können Datums-, numerische und Stringdatentypen maskieren. Die zufällige Maskierung erfordert keinen Seed-Wert. Die Ergebnisse der zufälligen Maskierung sind nicht deterministisch.

Ausdruck

Wendet einen Ausdruck auf eine Quellspalte an, um Daten zu erstellen oder zu maskieren. Sie können alle Datentypen maskieren.

Schlüssel

Ersetzt Quelldaten durch wiederholbare Werte. Die Datenmaskierungsumwandlung erzielt deterministische Ergebnisse für dieselben Quelldaten, Maskierungsregeln und Seed-Werte. Sie können Datums-, numerische und Stringdatentypen maskieren.

Substitution

Ersetzt eine Datenspalte durch ähnliche, aber nicht verbundene Daten aus einem Wörterbuch. Sie können den Stringdatentyp maskieren.

Abhängig

Ersetzt die Werte einer Quellspalte basierend auf den Werten einer anderen Quellspalte. Sie können den Stringdatentyp maskieren.

Tokenisierung

Ersetzt Quelldaten durch Daten, die basierend auf benutzerdefinierten Datenmaskierungskriterien generiert wurden. Die Datenmaskierungsumwandlung wendet Regeln in einem benutzerdefinierten Algorithmus an. Sie können den Stringdatentyp maskieren.

Spezielle Maskierungsformate

Kreditkartennummer, E-Mail-Adresse, IP-Adresse, Telefonnummer, SSN, SIN oder URL. Die Datenmaskierungsumwandlung wendet integrierte Regeln an, um diese häufigen Typen vertraulicher Daten intelligent zu maskieren.

Keine Maskierung

Die Datenmaskierungsumwandlung ändert die Quelldaten nicht.

Standard ist keine Maskierung.

Zufällige Maskierung

Die zufällige Maskierung generiert zufällige nicht deterministische maskierte Daten. Die Datenmaskierungsumwandlung gibt unterschiedliche Werte zurück, wenn der gleiche Quellwert in unterschiedlichen Zeilen vorkommt. Sie können Maskierungsregeln definieren, die sich auf das Format der Daten auswirken, die von der Datenmaskierungsumwandlung zurückgegeben werden. Maskieren Sie numerische Werte, Stringwerte und Datumswerte mit der zufälligen Maskierung.

Maskieren von Stringwerten

Konfigurieren Sie die zufällige Maskierung, um eine zufällige Ausgabe für Stringspalten zu generieren. Konfigurieren Sie ein Maskierungsformat, um Einschränkungen für jedes Zeichen im Ausgabestring zu konfigurieren. Konfigurieren Sie Filterzeichen, um die zu maskierenden Quellzeichen ebenso wie die Zeichen zu definieren, mit denen sie maskiert werden.

Sie können die folgenden Maskierungsregeln für einen Stringport verwenden:

Bereich

Konfigurieren Sie die minimale und maximale Stringlänge. Die Datenmaskierungsumwandlung gibt einen String zufälliger Zeichen zwischen der minimalen und maximalen Stringlänge zurück.

Maskierungsformat

Definieren Sie den Zeichentyp, der für jedes Zeichen in den Eingabedaten ersetzt werden soll. Sie können jedes Zeichen auf einen alphabetischen, numerischen oder alphanumerischen Zeichentyp beschränken.

Quellstringzeichen

Definieren Sie die Zeichen in der Quellzeichenfolge, die Sie maskieren möchten. Maskieren Sie z. B. das Doppelkreuz-Zeichen (#) immer, wenn es in den Eingabedaten vorkommt. Die Datenmaskierungsumwandlung maskiert alle Eingabezeichen, wenn "Quellstringzeichen" leer ist.

Ersatzzeichen für Ergebniszeichenfolge

Ersetzen Sie die Zeichen im Target-String mit den in "Ergebnisstringzeichen" definierten Zeichen. Geben Sie z. B. die folgenden Zeichen ein, um jede Maskierung so zu konfigurieren, dass die alphabetischen Zeichen A-Z in Großbuchstaben eingebunden werden:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Maskieren von numerischen Werten

Wenn Sie numerische Daten maskieren, können Sie einen Ausgabewertebereich für eine Spalte konfigurieren. Die Datenmaskierungsumwandlung gibt einen Wert zwischen dem minimalen und dem maximalen Wert des Bereichs je nach Portpräzision zurück. Um den Bereich zu definieren, konfigurieren Sie das Bereichsminimum und -maximum oder einen Unschärfebereich auf der Grundlage einer Abweichung vom ursprünglichen Quellwert.

Sie können die folgenden Maskierungsparameter für numerische Daten konfigurieren:

Bereich

Definieren Sie einen Ausgabewertebereich. Die Datenmaskierungsumwandlung gibt numerische Daten zwischen dem minimalen und maximalen Wert zurück.

Unschärfebereich

Definieren Sie einen Bereich für Ausgabewerte, die innerhalb einer festen oder einer prozentualen Abweichung von den Quelldaten liegen. Die Datenmaskierungsumwandlung gibt numerische Daten zurück, die sich dem Wert der Quelldaten annähern. Sie können einen Bereich und einen Unschärfebereich konfigurieren.

Maskieren von Datumswerten

Um Datumswerte mit der zufälligen Maskierung zu maskieren, konfigurieren Sie entweder einen Ausgabedatenbereich oder wählen eine Varianz aus. Wenn Sie eine Varianz konfigurieren, wählen Sie einen Teil des Datums für die Unschärfe aus. Wählen Sie das Jahr, den Monat, den Tag, die Stunde, die Minute oder die Sekunde aus. Die Datenmaskierungsumwandlung gibt ein Datum zurück, das innerhalb des konfigurierten Bereichs liegt.

Sie können die folgenden Maskierungsregeln konfigurieren, wenn Sie einen Datetime-Wert maskieren:

Bereich

Legt den minimalen und den maximalen Wert fest, die für den ausgewählten Datetime-Wert zurückgegeben werden sollen.

Unschärfe

Maskiert ein Datum auf der Grundlage einer Varianz, die Sie auf eine Einheit des Datums anwenden. Die Datenmaskierungsumwandlung gibt ein Datum zurück, das innerhalb dieser Varianz liegt. Sie können Unschärfe auf das Jahr, den Monat, den Tag, die Stunde, die Minute oder die Sekunde anwenden. Wählen Sie einen niedrigen und einen hohen anzuwendenden Varianzwert aus.

Ausdrucksmaskierung

Die Ausdrucksmaskierung wendet einen Ausdruck auf einen Port an, um die Daten zu ändern oder neue Daten zu erstellen. Erstellen Sie beim Konfigurieren der Ausdrucksmaskierung einen Ausdruck im Ausdruckseditor. Wählen Sie die Eingabe- und Ausgabeports, Funktionen, Variablen und Operatoren aus, um Ausdrücke zu erstellen.

Sie können Daten aus mehreren Ports verketten, um einen Wert für einen anderen Port zu erstellen. Sie müssen beispielsweise einen Anmeldenamen erstellen. Die Quelle verfügt über Spalten für Vor- und Nachname. Maskieren Sie den Vor- und Nachnamen aus Lookup-Dateien. Erstellen Sie in der Datenmaskierungsumwandlung einen anderen Port mit der Bezeichnung "Anmeldung". Konfigurieren Sie für den Port "Anmeldung" einen Ausdruck, um den ersten Buchstaben des Vornamens mit dem Nachnamen zu verketten:

```
SUBSTR(FIRSTNM,1,1)||LASTNM
```

Wählen Sie Funktionen, Ports, Variablen und Operatoren in der Point-and-Click-Schnittstelle aus, um Fehler bei der Erstellung von Ausdrücken zu vermeiden.

Der Ausdruckseditor zeigt die Ausgabeports an, die nicht für die Ausdrucksmaskierung konfiguriert sind. Sie können die Ausgabe aus einem Ausdruck nicht als Eingabe in einem anderen Ausdruck verwenden. Wenn Sie den Namen des Ausgabeports manuell zum Ausdruck hinzufügen, erhalten Sie unter Umständen unerwartete Ergebnisse.

Stellen Sie beim Erstellen eines Ausdrucks sicher, dass der Ausdruck einen Wert zurückgibt, der mit dem Portdatentyp übereinstimmt. Die Datenmaskierungsumwandlung gibt Null zurück, wenn der Datentyp des Ausdrucksports numerisch und der Datentyp des Ausdrucks nicht identisch ist. Die Datenmaskierungsumwandlung gibt Null zurück, wenn der Datentyp des Ausdrucksports ein String und der Datentyp des Ausdrucks nicht identisch ist.

Wiederholbare Ausdrucksmaskierung

Konfigurieren Sie die wiederholbare Ausdrucksmaskierung, wenn eine Quellspalte in mehr als einer Tabelle vorkommt und die Spalte aus jeder Tabelle mit demselben Wert maskiert werden muss.

Wenn Sie die wiederholbare Ausdrucksmaskierung konfigurieren, speichert die Datenmaskierungsumwandlung die Ergebnisse eines Ausdrucks in einer Speichertabelle. Wenn die Spalte in einer anderen Quelltabelle vorkommt, gibt die Datenmaskierungsumwandlung den maskierten Wert aus der Speichertabelle statt aus dem Ausdruck zurück.

Wörterbuchname

Beim Konfigurieren der wiederholbaren Ausdrucksmaskierung müssen Sie einen Wörterbuchnamen eingeben. Der Wörterbuchname ist ein Schlüssel, mit dem mehrere Datenmaskierungsumwandlungen dieselben maskierten Werte aus denselben Quellwerten erzeugen können. Definieren Sie denselben Wörterbuchnamen in jeder Datenmaskierungsumwandlung. Bei dem Wörterbuchnamen kann es sich um beliebigen Text handeln.

Speichertabelle

Der Speichertabelle enthält die Ergebnisse der wiederholbaren Ausdrucksmaskierung zwischen Sitzungen. Eine Speichertabellenzeile enthält die Quellspalte und ein maskiertes Wertpaar. Die Ausdrucksmaskierung und die Substitutions-Maskierung befinden sich in verschiedenen Speichertabellen.

Jedes Mal, wenn die Datenmaskierungsumwandlung einen Wert mit einem wiederholbaren Ausdruck maskiert, wird die Speichertabelle nach Wörterbuchname, Gebietsschema, Spaltenname und Eingabewert durchsucht. Bei Auffinden einer Zeile in der Speichertabelle wird der maskierte Wert aus der Speichertabelle zurückgegeben. Wenn die Datenmaskierungsumwandlung keine Zeile findet, wird ein maskierter Wert aus dem Ausdruck für die Spalte erstellt.

Sie müssen Speichertabellen für die Ausdrucksmaskierung verschlüsseln, wenn sich unverschlüsselte Daten im Speicher befinden und denselben Wörterbuchnamen als Schlüssel verwenden.

Verschlüsseln von Speichertabellen für Ausdrucksmaskierung

Sie können Kodierfunktionen der Umwandlungssprache zum Verschlüsseln von Speichertabellen verwenden. Sie müssen die Speichertabellen verschlüsseln, wenn die Speicherverschlüsselung aktiviert wurde.

1. Erstellen Sie ein Mapping mit der Speichertabelle IDM_EXPRESSION_STORAGE als Quelle.
2. Eine Datenmaskierungsumwandlung erstellen.
3. Die Technik der Ausdrucksmaskierung auf maskierte Wertporte anwenden.
4. Wenden Sie den folgenden Ausdruck auf den Port MASKEDVALUE an:

   ```
   Enc_Base64(AES_Encrypt(MASKEDVALUE, Key))
   ```

5. Verknüpfen Sie die Ports mit dem Ziel.

Beispiel

Die Tabelle "Mitarbeiter" (Employees) enthält beispielsweise die folgenden Spalten:

```
FirstName
LastName
LoginID
```

Maskieren Sie in der Datenmaskierungsumwandlung die Anmelde-ID (LoginID) mit einem Ausdruck, der Vornamen (FirstName) und Nachnamen (LastName) kombiniert. Konfigurieren Sie die Ausdrucksmaskierung als wiederholbar. Geben Sie einen Wörterbuchnamen als Schlüssel für die wiederholbare Maskierung ein.

Die Tabelle "Computerbenutzer" (Computer_Users) enthält eine Anmelde-ID (LoginID), aber keine Spalten für Vorname (FirstName) oder Nachname (LastName):

```
Dept
LoginID
Password
```

Zum Maskieren der Anmelde-ID (LoginID) in der Tabelle "Computerbenutzer" (Computer_Users) mit derselben Anmelde-ID (LoginID) wie in der Tabelle "Mitarbeiter" (Employees)konfigurieren Sie die Ausdrucksmaskierung für die Spalte "Anmelde-ID" (LoginID). Aktivieren Sie die wiederholbare Maskierung und geben Sie denselben Wörterbuchnamen ein, den Sie für die Anmelde-ID (LoginID) aus der Tabelle "Mitarbeiter" (Employees) definiert haben. Der Integration Service ruft die Anmelde-ID-Werte (LoginID) aus der Speichertabelle ab.

Erstellen Sie einen Standardausdruck, der verwendet werden soll, wenn der Integration Service in der Speichertabelle keine Zeile für die Anmelde-ID (LoginID) finden kann. Die Tabelle "Computerbenutzer" (Computer_Users) enthält weder die Spalte "Vorname" (FirstName) noch die Spalte "Nachname" (LastName), so dass vom Ausdruck eine weniger sinnvolle Anmelde-ID (LoginID) erstellt wird.

Speichertabellenskripts

Informatica stellt Skripts bereit, die zum Erstellen der Speichertabelle ausgeführt werden können. Die Skripts befinden sich an folgendem Speicherort:

```
<PowerCenter installation directory>\client\bin\Extensions\DataMasking
```

Das Verzeichnis enthält ein Skript für Sybase-, Microsoft SQL Server-, IBM DB2- und Oracle-Datenbanken. Jede Skript ist mit der Bezeichnung <Ausdruck_<Datenbanktyp> versehen.

Regeln und Richtlinien für die Ausdrucksmaskierung

Verwenden Sie die folgenden Regeln und Richtlinien für die Ausdrucksmaskierung:

- Sie können die Ausgabe aus einem Ausdruck nicht als Eingabe in einem anderen Ausdruck verwenden. Wenn Sie den Namen des Ausgabeports manuell zum Ausdruck hinzufügen, erhalten Sie unter Umständen unerwartete Ergebnisse.
- Verwenden Sie die Point-and-Click-Methode zum Erstellen von Ausdrücken. Wählen Sie Funktionen, Ports, Variablen und Operatoren in der Point-and-Click-Schnittstelle aus, um Fehler bei der Erstellung von Ausdrücken zu vermeiden.
- Wenn die Datenmaskierungsumwandlung für die wiederholbare Maskierung konfiguriert wurde und die Speichertabelle nicht vorhanden ist, ersetzt der Integration Service die Quelldaten durch Standardwerte.

Schlüsselmaskierung

Eine für die Schlüsselmaskierung konfigurierte Spalte gibt deterministische, maskierte Daten jedes Mal zurück, wenn Quellwert und Seed-Wert übereinstimmen. Die Datenmaskierungsumwandlung gibt eindeutige Werte für die Spalte zurück.

Beim Konfigurieren einer Spalte für die Schlüsselmaskierung erstellt die Datenmaskierungsumwandlung einen Seed-Wert für die Spalte. Sie können den Seed-Wert ändern, um wiederholbare Daten unter verschiedenen Datenmaskierungsumwandlungen zu erzielen. Konfigurieren Sie beispielsweise die Schlüsselmaskierung, um referenzielle Integrität zu erzwingen. Verwenden Sie denselben Seed-Wert, um einen Primärschlüssel in einer Tabelle und den Fremdschlüsselwert in einer anderen Tabelle zu maskieren.

Sie können Maskierungsregeln definieren, die sich auf das Format der Daten auswirken, die von der Datenmaskierungsumwandlung zurückgegeben werden. Maskieren Sie String- und numerische Werte mit der Schlüsselmaskierung.

Maskieren von Stringwerten

Sie können die Schlüsselmaskierung konfigurieren, um eine wiederholbare Ausgabe für Strings zu generieren. Konfigurieren Sie ein Maskierungsformat, um Einschränkungen für jedes Zeichen im Ausgabestring zu definieren. Konfigurieren Sie Quellstringzeichen, mit denen definiert wird, welche Quellzeichen maskiert werden sollen. Konfigurieren Sie Ersatzzeichen für Ergebniszeichenfolgen, um die maskierten Daten auf bestimmte Zeichen zu beschränken.

Sie können die folgenden Maskierungsregeln für Schlüsselmaskierungsstrings konfigurieren:

Seed

Wenden Sie einen Seed-Wert an, um deterministische, maskierte Daten für eine Spalte zu generieren. Sie können eine Zahl zwischen 1 und 1.000 eingeben.

Maskierungsformat

Definieren Sie den Zeichentyp, der für jedes Zeichen in den Eingabedaten ersetzt werden soll. Sie können jedes Zeichen auf einen alphabetischen, numerischen oder alphanumerischen Zeichentyp beschränken.

Quellstringzeichen

Definieren Sie die Zeichen in der Quellzeichenfolge, die Sie maskieren möchten. Maskieren Sie z. B. das Doppelkreuz-Zeichen (#) immer, wenn es in den Eingabedaten vorkommt. Die Datenmaskierungsumwandlung maskiert alle Eingabezeichen, wenn "Quellstringzeichen" leer ist. Die Datenmaskierungsumwandlung gibt nicht immer eindeutige Daten zurück, wenn die Anzahl der Quellstringzeichen kleiner ist als die Anzahl der Ergebnisstringzeichen.

Zeichen der Ergebniszeichenfolge

Ersetzen Sie die Zeichen im Target-String mit den in "Ergebnisstringzeichen" definierten Zeichen. Geben Sie z. B. die folgenden Zeichen ein, um jede Maskierung so zu konfigurieren, dass alle alphabetischen Zeichen in Großbuchstaben eingebunden werden:

```
ABCDEFGHIJKLMNOPQRSTUVWXYZ
```

Maskieren von numerischen Werten

Konfigurieren Sie die Schlüsselmaskierung für numerische Quelldaten, um eine deterministische Ausgabe zu generieren. Beim Konfigurieren einer Spalte für numerische Schlüsselmaskierung weisen Sie der Spalte einen zufälligen Seed-Wert zu. Wenn die Datenmaskierungsumwandlung die Quelldaten maskiert, wendet sie einen Maskierungsalgorithmus an, der den Seed erfordert.

Sie können den Seed-Wert für eine Spalte ändern, um wiederholbare Ergebnisse zu erzielen, wenn derselbe Quellwert in einer anderen Spalte vorkommt. Sie möchten beispielsweise eine Primärschlüssel-Fremdschlüssel-Beziehung zwischen zwei Tabellen beibehalten. Geben Sie in jeder Datenmaskierungsumwandlung den gleichen Seed-Wert für die Primärschlüsselspalte wie für die Fremdschlüsselspalte ein. Die Datenmaskierungsumwandlung erzielt deterministische Ergebnisse für dieselben numerischen Werte. Die referenzielle Integrität wird zwischen den Tabellen beibehalten.

Maskieren von Datetime-Werten

Wenn Sie Schlüsselmaskierung für Datetime-Werte konfigurieren können, ist für die Datenmaskierungsumwandlung eine statistische Zahl als Seed erforderlich. Sie können den Seed so ändern, dass er mit dem Seed-Wert für eine andere Spalte übereinstimmt, damit wiederholbare Datetime-Werte in den Spalten zurückgegeben werden.

Die Datenmaskierungsumwandlung kann Daten zwischen 1753 und 2400 mit der Schlüsselmaskierung maskieren. Wenn das Quelljahr ein Schaltjahr ist, gibt die Datenmaskierungsumwandlung ein Jahr zurück, das auch ein Schaltjahr ist. Wenn der Quellmonat 31 Tage umfasst, gibt die Datenmaskierungsumwandlung einen Monat mit 31 Tagen zurück. Wenn der Quellmonat Februar ist, gibt die Datenmaskierungsumwandlung Februar zurück.

Die Datenmaskierungsumwandlung generiert immer gültige Daten.

Substitutions-Maskierung

Bei der Substitutions-Maskierung wird eine Spalte mit Daten durch ähnliche, aber nicht zugehörige Daten ersetzt. Verwenden Sie die Substitutions-Maskierung, um Produktionsdaten durch realistische Daten zu

ersetzen. Definieren Sie beim Konfigurieren der Substitutions-Maskierung das Wörterbuch, das die Ersatzwerte enthält.

Die Datenmaskierungsumwandlung führt ein Lookup im konfigurierten Wörterbuch durch. Die Datenmaskierungsumwandlung ersetzt Quelldaten durch Daten aus dem Wörterbuch. Wörterbuchdateien können Stringdaten, Datums-/Uhrzeitwerte, Ganzzahlen und Gleitkommazahlen enthalten. Geben Sie Datums-/Uhrzeitwerte in folgendem Format ein:

```
mm/dd/yyyy
```

Sie können Daten durch wiederholbare oder nicht wiederholbare Werte ersetzen. Wenn Sie wiederholbare Werte auswählen, erzeugt die Datenmaskierungsumwandlung deterministische Ergebnisse für dieselben Quelldaten und denselben Seed-Wert. Sie müssen einen Seed-Wert konfigurieren, um Daten durch deterministische Ergebnisse zu ersetzen. Der Integration Service verwaltet eine Speichertabelle mit Quell- und maskierten Daten für die wiederholbare Maskierung.

Sie können mehr als eine Spalte mit Daten durch maskierte Werte aus derselben Wörterbuchzeile ersetzen. Konfigurieren Sie die Substitutions-Maskierung für eine Eingabespalte. Konfigurieren Sie die abhängige Datenmaskierung für die anderen Spalten, die maskierte Daten aus derselben Wörterbuchzeile empfangen.

Wörterbücher

Ein Wörterbuch ist eine Referenztabelle, die die Ersatzdaten und eine Seriennummer für jede Zeile in der Tabelle enthält. Erstellen Sie eine Referenztabelle für die Substitutions-Maskierung aus einer Einfachdatei oder einer relationalen Tabelle, die Sie in das Model Repository importieren.

Die Datenmaskierungsumwandlung generiert eine Zahl zum Abrufen einer Wörterbuchzeile nach Seriennummer. Die Datenmaskierungsumwandlung generiert einen Hashschlüssel für die wiederholbare Substitutions-Maskierung oder eine Zufallszahl für die nicht wiederholbare Maskierung. Sie können eine zusätzliche Lookup-Bedingung festlegen, wenn Sie die wiederholbare Substitutions-Maskierung konfigurieren.

Sie können ein Wörterbuch konfigurieren, um mehrere Ports in der Datenmaskierungsumwandlung zu maskieren.

Wenn die Datenmaskierungsumwandlung Ersatzdaten aus einem Wörterbuch abruft, wird bei der Umwandlung nicht überprüft, ob der Ersatzdatenwert mit dem ursprünglichen Wert übereinstimmt. Bei der Datenmaskierungsumwandlung wird der Name "John" unter Umständen durch denselben Namen (John) aus einer Wörterbuchdatei ersetzt.

Das folgende Beispiel zeigt eine Wörterbuchtabelle, die den Vornamen und das Geschlecht enthält:

SNO	GESCHLECHT	VORNAME
1	M	Adam
2	M	Adeel
3	M	Adil
4	W	Alice
5	W	Alison

In diesem Wörterbuch handelt es sich bei dem ersten Feld in der Zeile um die Seriennummer und bei dem zweiten Feld um das Geschlecht. Der Integration Service sucht einen Wörterbuchdatensatz immer nach der Seriennummer. Sie können das Geschlecht als Lookup-Bedingung hinzufügen, wenn Sie die wiederholbare Maskierung konfigurieren. Der Integration Service ruft eine Zeile aus dem Wörterbuch mithilfe eines

Hashschlüssels ab und findet eine Zeile mit einem Geschlecht, das dem Geschlecht in den Quelldaten entspricht.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie eine Referenztabelle erstellen:

- Jeder Datensatz in der Tabelle muss eine Seriennummer aufweisen.
- Die Seriennummern sind sequentielle Ganzzahlen, die mit 1 beginnen. Die Seriennummern dürfen keine fehlende Zahl in der Sequenz aufweisen.
- Die Spalte mit den Seriennummern kann sich an einer beliebigen Stelle in der Tabelle befinden. Sie kann eine beliebige Beschriftung aufweisen.

Wenn Sie eine Einfachdatei-Tabelle mithilfe einer Referenztabelle erstellen, verwenden Sie die folgenden Regeln und Richtlinien:

- Die erste Zeile der Einfachdatei-Tabelle muss Spaltenbeschriftungen aufweisen, um die Felder in jedem Datensatz anzugeben. Die Felder sind durch Kommas getrennt. Wenn die erste Zeile keine Spaltenbeschriftungen enthält, übernimmt der Integration Service die Werte der Felder in der ersten Zeile als Spaltennamen.
- Wenn Sie eine Einfachdatei-Tabelle unter Windows erstellen und auf einen UNIX-Computer kopieren, stellen Sie sicher, dass das Dateiformat für UNIX richtig ist. Windows und UNIX verwenden beispielsweise unterschiedliche Zeichen zum Markieren des Zeilenendes.

Speichertabellen

Die Datenmaskierungsumwandlung verwaltet Speichertabellen für die wiederholbare Substitution zwischen Sitzungen. Eine Speichertabellenzeile enthält die Quellspalte und ein maskiertes Wertpaar. Jedes Mal, wenn die Datenmaskierungsumwandlung einen Wert mit einem wiederholbaren Ersatzwert maskiert, wird die Speichertabelle nach Wörterbuchname, Gebietsschema, Spaltenname, Eingabewert und Seed durchsucht. Bei Auffinden einer Zeile wird der maskierte Wert aus der Speichertabelle zurückgegeben. Wenn die Datenmaskierungsumwandlung keine Zeile findet, wird eine Zeile aus dem Wörterbuch mit einem Hashschlüssel abgerufen.

Das Format des Wörterbuchnamens in der Speichertabelle ist für ein Einfachdatei-Wörterbuch und ein relationales Wörterbuch unterschiedlich. Der Name eines Einfachdatei-Wörterbuchs wird durch den Dateinamen angegeben. Der Name des relationalen Wörterbuchs hat folgende Syntax:

```
<Connection object>_<dictionary table name>
```

Informatica stellt Skripts bereit, die zum Erstellen einer relationalen Speichertabelle ausgeführt werden können. Die Skripts befinden sich an folgendem Speicherort:

```
<PowerCenter Client installation directory>\client\bin\Extensions\DataMasking
```

Das Verzeichnis enthält ein Skript für Sybase-, Microsoft SQL Server-, IBM DB2- und Oracle-Datenbanken. Jedes Skript wird folgendermaßen benannt: Substitution_<Datenbanktyp>. Sie können eine Tabelle in einer anderen Datenbank erstellen, wenn Sie die SQL-Anweisungen und die Primärschlüsseleinschränkungen konfigurieren.

Sie müssen Speichertabellen für die Substitutions-Maskierung verschlüsseln, wenn sich unverschlüsselte Daten im Speicher befinden und den gleichen Seed-Wert und das gleiche Wörterbuch zum Verschlüsseln der gleichen Spalten verwenden.

Verschlüsseln der Speichertabellen für die Substitutions-Maskierung

Sie können Kodierfunktionen der Umwandlungssprache zum Verschlüsseln von Speichertabellen verwenden. Sie müssen die Speichertabellen verschlüsseln, wenn die Speicherverschlüsselung aktiviert wurde.

1. Erstellen Sie ein Mapping mit der Speichertabelle IDM_SUBSTITUTION_STORAGE als Quelle.

2. Eine Datenmaskierungsumwandlung erstellen.
3. Anwenden der Technik der Substitutions-Maskierung auf den Eingabewert und die maskierten Wertports.
4. Wenden Sie den folgenden Ausdruck auf den Port INPUTVALUE an:

   ```
   Enc_Base64(AES_Encrypt(INPUTVALUE, Key))
   ```

5. Wenden Sie den folgenden Ausdruck auf den Port MASKEDVALUE an:

   ```
   Enc_Base64(AES_Encrypt(MASKEDVALUE, Key))
   ```

6. Verknüpfen Sie die Ports mit dem Ziel.

Eigenschaften der Substitutions-Maskierung

Sie können die folgenden Maskierungsregeln für die Substitutions-Maskierung konfigurieren:

- **Wiederholbare Ausgabe**. Gibt deterministische Ergebnisse zwischen Sitzungen zurück. Die Datenmaskierungsumwandlung speichert maskiert Werte in der Speichertabelle.
- **Seed-Wert.** Wenden Sie einen Seed-Wert an, um deterministische, maskierte Daten für eine Spalte zu generieren. Geben Sie eine Zahl zwischen 1 und 1.000 ein.
- **Eindeutige Ausgabe**. Erzwingen Sie bei der Datenmaskierungsumwandlung die Erstellung eindeutiger Ausgabewerte für eindeutige Eingabewerte. Die Maskierung zweier Eingabewerte führt nie zum selben Ausgabewert. Das Wörterbuch muss über genügend eindeutige Zeilen verfügen, um eine eindeutige Ausgabe zu ermöglichen.
  Wenn Sie die eindeutige Ausgabe deaktivieren, kann die Datenmaskierungsumwandlung Eingabewerte unter Umständen nicht auf eindeutige Ausgabewerte abbilden. Das Wörterbuch enthält möglicherweise weniger Zeilen.
- **Eindeutiger Port**. Der Port, der zum Angeben eindeutiger Datensätze für die Substitutions-Maskierung verwendet wird. Sie möchten beispielsweise in einer Tabelle mit der Bezeichnung "Kunde" Vornamen maskieren. Wenn Sie die Tabellenspalte auswählen, die die Vornamen als eindeutigen Port enthält, ersetzt die Datenmaskierungsumwandlung doppelte Vornamen durch den gleichen maskierten Wert. Wenn Sie die Spalte "Customer_ID" (Kunden-ID) als eindeutigen Port auswählen, ersetzt die Datenmaskierungsumwandlung jeden Vornamen durch einen eindeutigen Wert.
- **Wörterbuchinformationen**. Konfigurieren Sie die Referenztabelle, die die Ersatzdatenwerte enthält. Klicken Sie auf **Quelle auswählen**, um eine Referenztabelle auszuwählen.
  - **Wörterbuchname**. Zeigt den Namen der ausgewählten Referenztabelle an.
  - **Ausgabespalte**. Wählen Sie die Spalte aus, die an die Datenmaskierungsumwandlung zurückgegeben werden soll.
- **Lookup-Bedingung.** Konfigurieren Sie eine Lookup-Bedingung, um die für die Substitutions-Maskierung zu verwendende Wörterbuchzeile weiter zu qualifizieren. Die Lookup-Bedingung ähnelt der WHERE-Klausel in einer SQL-Abfrage. Wenn Sie eine Lookup-Bedingung konfigurieren, vergleichen Sie den Wert einer Spalte in der Quelle mit einer Spalte im Wörterbuch.

  Sie möchten beispielsweise den Vornamen maskieren. Die Quelldaten und das Wörterbuch verfügen über eine Spalte für Vornamen und eine Spalte für das Geschlecht. Sie können eine Bedingung hinzufügen, gemäß derer jeder weibliche Vorname durch einen weiblichen Vornamen aus dem Wörterbuch ersetzt wird. Die Lookup-Bedingung vergleicht das Geschlecht in der Quelle mit dem Geschlecht im Wörterbuch.
  - **Eingabeport**. Im Lookup zu verwendende Quelldatenspalte.
  - **Wörterbuchspalte**. Wörterbuchspalte, mit der der Port verglichen wird.

Regeln und Richtlinien für die Substitutions-Maskierung

Verwenden Sie die folgenden Regeln und Richtlinien für die Substitutions-Maskierung:

- Wenn für eine eindeutige, wiederholbare Substitutions-Maskierung keine Speichertabelle vorhanden ist, schlägt die Sitzung fehl.
- Wenn das Wörterbuch keine Zeilen enthält, gibt die Datenmaskierungsumwandlung eine Fehlermeldung zurück.
- Wenn die Datenmaskierungsumwandlung einen Eingabewert mit dem Gebietsschema, Wörterbuch und Seed in der Speichertabelle findet, wird der maskierte Wert abgerufen, selbst wenn die Zeile im Wörterbuch nicht mehr vorhanden ist.
- Wenn Sie ein Verbindungsobjekt löschen oder das Wörterbuch ändern, schneiden Sie die Speichertabelle ab. Andernfalls erhalten Sie unter Umständen unerwünschte Ergebnisse.
- Unterschreitet die Anzahl der Werte im Wörterbuch die Anzahl der eindeutigen Werte in der Datenquelle, kann die Datenmaskierungsumwandlung die Daten nicht mit eindeutigen, wiederholbaren Werten maskieren. Die Datenmaskierungsumwandlung gibt eine Fehlermeldung zurück.

Abhängige Maskierung

Bei der abhängigen Maskierung werden mehrere Spalten mit Quelldaten durch Daten aus derselben Wörterbuchzeile ersetzt.

Wenn die Datenmaskierungsumwandlung eine Substitutions-Maskierung für mehrere Spalten durchführt, enthalten die maskierten Daten unter Umständen unrealistische Felderkombinationen. Sie können die Datenmaskierung konfigurieren, um Daten für mehrere Eingabespalten aus derselben Wörterbuchzeile zu ersetzen. Die maskierten Daten empfangen gültige Kombinationen, wie z. B. "New York, New York" oder "Chicago, Illinois".

Beim Konfigurieren der abhängigen Maskierung konfigurieren Sie zuerst eine Eingabespalte für die Substitutions-Maskierung. Konfigurieren Sie andere Eingabespalten, die von dieser Substitutionsspalte abhängig sind. Sie können beispielsweise die Spalte mit Postleitzahlen für die Substitutions-Maskierung auswählen und die Spalten mit Stadt und Bundesstaat so einrichten, dass sie von der Spalte mit den Postleitzahlen abhängig sind. Bei der abhängigen Maskierung wird sichergestellt, dass die ersetzten Werte für Stadt und Bundesstaat für den ersetzten Postleitzahlenwert gültig sind.

Hinweis: Sie können eine Spalte für die abhängige Maskierung erst konfigurieren, wenn Sie eine Spalte für die Substitutions-Maskierung konfiguriert haben.

Konfigurieren Sie die folgenden Maskierungsregeln beim Konfigurieren einer Spalte für die abhängige Maskierung:

Abhängige Spalte

Der Name der Eingabespalte, den Sie für die Substitutions-Maskierung konfiguriert haben. Die Datenmaskierungsumwandlung ruft Ersatzdaten mithilfe der Maskierungsregel für diese Spalte aus einem Wörterbuch ab. Die für die Substitutions-Maskierung konfigurierte Spalte wird zur Schlüsselspalte für das Abrufen von maskierten Daten aus dem Wörterbuch.

Ausgabespalte

Der Name der Wörterbuchspalte, die den Wert für die Spalte enthält, die Sie mit der abhängigen Maskierung konfigurieren.

Beispiel für eine abhängige Maskierung

Ein Datenmaskierungswörterbuch enthält unter Umständen Zeilen mit den folgenden Werten:

SNO	STRASSE	STADT	BUNDESSTAAT	PLZ	LAND
1	32 Apple Lane	Chicago	IL	61523	US
2	776 Ash Street	Dallas	TX	75240	US
3	2229 Big Square	Atleeville	TN	38057	US
4	6698 Cowboy Street	Houston	TX	77001	US

Sie müssen Quelldaten mit gültigen Kombinationen aus Stadt, Bundesland und Postleitzahl aus dem Adresswörterbuch maskieren.

Konfigurieren Sie den Postleitzahl-Port für die Substitutions-Maskierung. Geben Sie folgende Maskierungsregeln für den Postleitzahl-Port ein:

Regel	Wert
Wörterbuchname	Adresse
Seriennummernspalte	SNO
Ausgabespalte	PLZ

Konfigurieren Sie den Stadt-Port für die abhängige Maskierung. Geben Sie folgende Maskierungsregeln für den Stadt-Port ein:

Regel	Wert
Abhängige Spalte	PLZ
Ausgabespalte	Stadt

Konfigurieren Sie den Bundesstaat-Port für die abhängige Maskierung. Geben Sie folgende Maskierungsregeln für den Bundesstaat-Port ein:

Regel	Wert
Abhängige Spalte	PLZ
Ausgabespalte	Bundesstaat

Wenn die Datenmaskierungsumwandlung die Postleitzahl maskiert, werden die richtige Stadt und der richtige Bundesstaat für die Postleitzahl aus der Wörterbuchzeile zurückgegeben.

Tokenisierungsmaskierung

Verwenden Sie die Tokenisierungs-Maskierungstechnik, um Quellzeichenfolgendaten basierend auf den von Ihnen in einen Algorithmus angegebenen Kriterien zu maskieren. Beispielsweise können Sie einen Algorithmus erstellen, der eine falsche E-Mail-Adresse enthält, um Feldeinträge in den Quelldaten zu ersetzen.

Konfigurieren Sie das Format der maskierten Daten unter Verwendung der Tokenisierungs-Maskierung. Sie müssen eine Tokenisierung zum Maskierungsalgorithmus zuweisen, bevor Sie sie verwenden können. Der Name der Tokenisierung verweist auf den verwendeten Maskierungsalgorithmus (JAR). Geben den Namen der Tokenisierung an, wenn Sie die Tokenisierungs-Maskierungstechnik anwenden möchten.

Konfigurieren der Tokenisierungsmaskierung

Führen Sie die folgenden Aufgaben durch, bevor Sie die Tokenisierungs-Maskierungstechnik verwenden:

1. Navigieren Sie zum Verzeichnis `tokenprovider` in dem folgenden Pfad: `<Informatica_home>\services\shared`.
2. Öffnen Sie die folgende XML-Datei: `com.informatica.products.ilm.tx-tokenizerprovider.xml`.
3. Fügen Sie den Tokenisierungsnamen und den vollständig qualifizierten Namen der Klassendatei für jede Tokenisierung hinzu, die Sie verwenden möchten. Implementieren Sie die Tokenisierungsklasse innerhalb der Klasse `com.informatica.products.ilm.tx-tokenprovider-<Build-Number>.jar` im Verzeichnis `tokenprovider`. Geben Sie für jede Tokenisierung die Informationen in der XML-Datei wie im folgenden Beispiel ein:

   ```
   <TokenizerProvider>
   <Tokenizer Name="CCTokenizer"
   ClassName="com.informatica.tokenprovider.CCTokenizer"/>
   </TokenizerProvider>
   ```

   wobei:

   - Der Tokenisierungsnamen ist der benutzerdefinierte Name in Anführungszeichen.
   - ClassName ist der benutzerdefinierte Name für das CLASSNAME-Attribut. Führen Sie die Implementierung dieses Inhalts über `com.informatica.products.ilm.tx-tokenprovider-<Build-Number>.jar` durch.

Nach der Konfiguration können Sie die Tokenisierungs-Maskierungstechnik verwenden. Geben Sie den Tokenisierungsnamen ein, um den beim Erstellen eines Mappings zu verwendenden Algorithmus anzugeben.

Maskierungsregeln

Maskierungsregeln sind die Optionen, die Sie nach Auswahl der Maskierungstechnik konfigurieren.

Wenn Sie die Technik der zufälligen Maskierung oder der Schlüsselmaskierung auswählen, können Sie das Maskierungsformat, die Quellstringzeichen und die Ergebnisstringzeichen konfigurieren. Sie können beim zufälligen Maskieren den Bereich oder die Unschärfe konfigurieren.

In der folgenden Tabelle werden die Maskierungsregeln beschrieben, die Sie für die jeweilige Maskierungstechnik konfigurieren können:

Maskierungsregel	Beschreibung	Maskierungstechnik	Quelldatentyp
Maskierungsformat	Maskierung, mit der jedes Zeichen in einem Ausgabestring auf ein alphabetisches, numerisches oder alphanumerisches Zeichen beschränkt wird.	Zufällig und Schlüssel	String
Quellstringzeichen	Satz von Quellzeichen, die maskiert oder von der Maskierung ausgeschlossen werden sollen.	Zufällig und Schlüssel	String
Ersatzzeichen für Ergebniszeichenfolge	Ein Satz von Zeichen, die in eine Maskierung einbezogen oder von einer Maskierung ausgeschlossen werden sollen.	Zufällig und Schlüssel	String
Bereich	Ein Ausgabewertebereich. - Numerisch. Die Datenmaskierungsumwandlung gibt numerische Daten zwischen dem minimalen und maximalen Wert zurück. - String. Gibt einen String von zufälligen Zeichen zwischen der minimalen und der maximalen Stringlänge zurück. - Datum/Uhrzeit. Gibt ein Datum und eine Uhrzeit innerhalb des minimalen und des maximalen Datetime-Werts zurück.	Zufällig	Numerisch String Date/Time
Unschärfe	Ausgabewertebereich mit fester oder prozentualer Abweichung von den Quelldaten. Die Datenmaskierungsumwandlung gibt Daten zurück, die sich dem Wert der Quelldaten annähern. Datetime-Spalten erfordern eine feste Varianz. Spalten erfordern eine feste Varianz.	Zufällig	Numerisch Date/Time

Maskierungsformat

Konfigurieren Sie ein Maskierungsformat, um jedes Zeichen in der Ausgabespalte auf ein alphabetisches, numerisches oder alphanumerisches Zeichen zu beschränken. Verwenden Sie die folgenden Zeichen, um ein Maskenformat zu definieren:

```
A, D, N, X, +, R
```

Hinweis: Das Maskenformat umfasst Zeichen in Großbuchstaben. Wenn Sie ein Maskierungszeichen in Kleinbuchstaben eingeben, wandelt die Datenmaskierungsumwandlung das Zeichen in einen Großbuchstaben um.

In der folgenden Tabelle werden Maskierungsformatzeichen beschrieben:

Zeichen	Beschreibung
A	Alphabetische Zeichen. Zum Beispiel ASCII-Zeichen a bis z und A bis Z.
D	Ziffern 0 bis 9. Die Datenmaskierungsumwandlung gibt ein "X" für alle anderen Zeichen als Ziffern 0 bis 9 aus.

Zeichen	Beschreibung
N	Alphanumerische Zeichen. Zum Beispiel ASCII-Zeichen a bis z, A bis Z und 0-9.
X	Beliebiges Zeichen. Zum Beispiel alphanumerisches Zeichen oder Symbol.
+	Keine Maskierung.
R	Verbleibende Zeichen. R gibt an, dass die verbleibenden Zeichen im String einen beliebigen Zeichentyp aufweisen können. R muss als letztes Zeichen der Maskierung auftreten.

Ein Abteilungsname hat beispielsweise das folgende Format:

```
nnn-<department_name>
```

Sie können eine Maske konfigurieren, um zu erzwingen, dass die ersten drei Zeichen numerisch sind, der Abteilungsname alphabetisch ist und der Gedankenstrich in der Ausgabe erhalten bleibt. Konfigurieren Sie das folgende Maskierungsformat:

```
DDD+AAAAAAAAAAAAAAAA
```

Die Datenmaskierungsumwandlung ersetzt die ersten drei Zeichen mit numerischen Zeichen. Das vierte Zeichen wird nicht ersetzt. Die Datenmaskierungsumwandlung ersetzt die verbleibenden Zeichen mit alphabetischen Zeichen.

Wenn Sie kein Maskierungsformat definieren, ersetzt die Datenmaskierungsumwandlung jedes Quellzeichen mit einem beliebigen Zeichen. Wenn das Maskierungsformat länger als der Eingabestring ist, ignoriert die Datenmaskierungsumwandlung die zusätzlichen Zeichen im Maskierungsformat. Wenn das Maskierungsformat kürzer als die Eingabezeichenfolge ist, maskiert die Datenmaskierungsumwandlung die verbleibenden Zeichen im R-Format.

Hinweis: Sie können ein Maskierungsformat nicht mit der Bereichsoption konfigurieren.

Quellstringzeichen

Quellstringzeichen sind Quellzeichen, für die Sie eine Maskierung bzw. Nicht-Maskierung festlegen. Die Position der Zeichen im Quellstring ist dabei unwichtig. Bei den Quellzeichen wird die Groß- und Kleinschreibung berücksichtigt.

Sie können eine beliebige Anzahl an Zeichen konfigurieren. Wenn unter "Zeichen" keine Eingabe erfolgt, ersetzt die Datenmaskierungs-Umwandlung alle Quellzeichen in der Spalte.

Wählen Sie eine der folgenden Optionen für Quellstringzeichen aus:

Nur maskieren

Die Datenmaskierungs-Umwandlung maskiert Zeichen in der Quelle, die Sie als Quellstringzeichen konfigurieren. Wenn Sie beispielsweise die Zeichen A, B und c eingeben, ersetzt die Datenmaskierungs-Umwandlung A, B und c durch ein anderes Zeichen, wenn das Zeichen in den Quelldaten vorkommt. Andere Quellzeichen als A, B und c werden nicht geändert. Bei der Maskierung wird zwischen Groß- und Kleinschreibung unterschieden.

Alle maskieren außer

Maskiert alle Zeichen mit Ausnahme der Quellstringzeichen, die im Quellstring vorkommen. Wenn Sie beispielsweise das Filterquellzeichen "-" eingeben und "Alle maskieren außer" auswählen, ersetzt die Datenmaskierungs-Umwandlung das Zeichen "-" nicht, wenn es in den Quelldaten vorkommt. Die übrigen Quellzeichen werden geändert.

Quellstring - Beispiel

Eine Quelldatei beinhaltet eine Spalte namens "Abhängige". Die Spalte "Abhängige" enthält mehr als einen durch Kommas getrennten Namen. Sie müssen die Spalte "Abhängige" maskieren und das Komma als Delimiter der Namen in den Testdaten beibehalten.

Wählen Sie "Quellstringzeichen" für die Spalte "Abhängige" aus. Wählen Sie "Nicht maskieren" aus und geben Sie "," als das zu überspringende Quellzeichen ein. Geben Sie keine Anführungszeichen ein.

Die Datenmaskierungsumwandlung ersetzt bis auf das Komma alle Zeichen im Quellstring.

Ersatzzeichen für Ergebniszeichenfolge

Ersatzzeichen für Ergebniszeichenfolgen sind Zeichen, die Sie als Substitute in maskierten Daten bestimmen können. Wenn Sie Ersatzzeichen für Ergebniszeichenfolgen konfigurieren, werden Zeichen in der Quellzeichenfolge von der Datenmaskierungsumwandlung mit diesen Ersatzzeichen ersetzt. Konfigurieren Sie einen weiten Bereich von Ersatzzeichen, um die Generierung derselben Ausgabe für unterschiedliche Eingabewerte zu verhindern, oder maskieren Sie nur wenige Quellzeichen. Die Position der jeweiligen Zeichen im String ist nicht wichtig.

Wählen Sie eine der folgenden Optionen für Ersatzzeichen für Ergebniszeichenfolgen aus:

Nur Folgende verwenden

Maskieren Sie die Quelle nur mit den von Ihnen als Ersatzzeichen für Ergebniszeichenfolgen definierten Zeichen. Wenn Sie beispielsweise die Zeichen A, B und c eingeben, ersetzt die Datenmaskierungsumwandlung jedes Zeichen in der Quellspalte mit einem A, B oder c. Das Wort "Pferd" könnte möglicherweise mit "BAcBA" ersetzt werden.

Alle verwenden außer

Maskieren Sie die Quelle mit allen Zeichen ausgenommen den von Ihnen als Ersatzzeichen für Ergebniszeichenfolgen definierten Zeichen. Wenn Sie beispielsweise die Ersatzzeichen A, B und c für Ergebniszeichenfolgen eingeben, umfassen die maskierten Daten niemals die Zeichen A, B und c.

Ersatzzeichen für Ergebniszeichenfolge - Beispiel

Führen Sie die folgenden Aufgaben durch, um alle Kommas in der Abhängigen-Spalte mit Semikolons zu ersetzen:

1. Konfigurieren Sie das Komma als Zeichen der Quellzeichenfolge und wählen Sie "Nur maskieren" aus.

   Die Datenmaskierungsumwandlung maskiert nur das Komma, wenn es in der Abhängigen-Spalte vorhanden ist.

2. Konfigurieren Sie das Semikolon als Ersatzzeichen für Ergebniszeichenfolgen und wählen Sie "Nur Folgende verwenden" aus.

   Die Datenmaskierungsumwandlung ersetzt jedes Komma in der Abhängigen-Spalte mit einem Semikolon.

Bereich

Definieren Sie einen Bereich für numerische Daten, Datumsdaten oder Stringdaten. Wenn Sie einen Bereich für numerische Werte oder Datumswerte definieren, maskiert die Datenmaskierungsumwandlung die Quelldaten mit einem Wert zwischen dem minimalen und dem maximalen Wert. Wenn Sie einen Bereich für einen String konfigurieren, konfigurieren Sie einen Bereich von Stringlängen.

Stringbereich

Wenn Sie eine Datenmaskierung nach dem Zufallsprinzip konfigurieren, werden von der Datenmaskierungs-Umwandlung Strings erzeugt, deren Länge nicht mit der Länge des Quellstrings übereinstimmt. Optional können Sie eine Mindest- und Höchstbreite für Strings konfigurieren. Bei den Werten, die von Ihnen als Mindest- oder Höchstbreite eingegeben werden, muss es sich um positive Ganzzahlen handeln. Jede Breite muss kleiner oder gleich der Portpräzision sein.

Numerischer Bereich

Legen Sie den minimalen und den maximalen Wert für eine numerische Spalte fest. Der maximale Wert muss kleiner oder gleich der Portpräzision sein. Der Standardbereich reicht von eins bis zur Portpräzisionslänge.

Datumsbereich

Legen Sie den minimalen und den maximalen Wert für einen Datetime-Wert fest. Die Felder für Minimum und Maximum enthalten das standardmäßige minimale bzw. maximale Datum. Das Standardformat für Datetime ist MM/DD/YYYY HH24:MI:SS. Der maximale Datetime-Wert muss nach dem minimalen Datetime-Wert liegen.

Unschärfe

Die Unschärfe erstellt einen Ausgabewert, der innerhalb einer festen oder prozentualen Abweichung vom Quelldatenwert liegt. Konfigurieren Sie die Unschärfe so, dass ein zufälliger Wert zurückgegeben wird, der sich dem Originalwert annähert. Sie können Unschärfe auf numerische Werte und Datumswerte anwenden.

Unschärfe für numerische Werte

Wählen Sie eine feste oder prozentuale Abweichung aus, um die Unschärfe auf einen numerischen Quellwert anzuwenden. Der niedrige Unschärfewert ist eine Varianz unterhalb des Quellwerts. Der hohe Unschärfewert ist eine Varianz oberhalb des Quellwerts. Der niedrige und der hohe Wert müssen größer oder gleich Null sein. Wenn die Datenmaskierungsumwandlung maskierte Daten zurückgibt, liegen die numerischen Daten innerhalb des definierten Bereichs.

In der folgenden Tabelle werden die Maskierungsergebnisse für Unschärfebereichswerte bei einem Eingabequellwert von 66 beschrieben:

Unschärfetyp	Low	High	Ergebnis
Fest	0	10	Zwischen 66 und 76
Fest	10	0	Zwischen 56 und 66
Fest	10	10	Zwischen 56 und 76
Prozent	0	50	Zwischen 66 und 99
Prozent	50	0	Zwischen 33 und 66
Prozent	50	50	Zwischen 33 und 99

Unschärfe für Datumswerte

Maskieren Sie ein Datum als Varianz des Quelldatums, indem Sie Unschärfe konfigurieren. Wählen Sie eine Einheit des Datums aus, auf die die Varianz angewendet werden soll. Sie können das Jahr, den Monat, den

Tag oder die Stunde auswählen. Geben Sie die untere und die obere Grenze ein, um eine Varianz oberhalb und unterhalb der Einheit des Quelldatums zu definieren. Die Datenmaskierungsumwandlung wendet die Varianz an und gibt ein Datum zurück, das innerhalb der Varianz liegt.

Um z. B. das maskierte Datum auf ein Datum zu beschränken, das in einem Zweijahresbereich ausgehend vom Quelldatum liegt, wählen Sie Jahr als Einheit aus. Geben Sie zwei als untere und obere Grenze ein. Wenn das Quelldatum 02/02/2006 lautet, gibt die Datenmaskierungsumwandlung ein Datum zwischen 02/02/2004 und 02/02/2008 zurück.

Standardmäßig ist Jahr die Unschärfeeinheit.

Spezielle Maskierungsformate

Spezielle Maskierungsformate sind Maskierungen, die auf allgemeine Datentypen angewendet werden können. Mit einem speziellen Maskierungsformat gibt die Datenmaskierungs-Umwandlung einen maskierten Wert zurück, der zwar ein realistisches Format aufweist, aber kein gültiger Wert ist.

Wenn Sie beispielsweise eine Sozialversicherungsnummer maskieren, gibt die Datenmaskierungs-Umwandlung eine Sozialversicherungsnummer zurück, die im korrekten Format vorliegt, aber ungültig ist. Sie können wiederholbare Maskierung für Sozialversicherungsnummern konfigurieren.

Konfigurieren Sie spezielle Maskierungen für die folgenden Datentypen:

- Sozialversicherungsnummern
- Kreditkartennummern
- Telefonnummern
- URL-Adressen
- E-Mail-Adressen
- IP-Adressen
- Sozialversicherungsnummern (SIN)

Wenn das Quelldatenformat oder der Datentyp für eine Maskierung ungültig ist, wendet der Data Integration Service eine Standardmaskierung auf die Daten an. Der Integration Service wendet maskierte Werte aus der Standardwertdatei an. Sie können die Standardwertdatei bearbeiten, um die Standardwerte zu ändern.

Datenmaskierung der Kreditkartennummer

Die Datenmaskierungsumwandlung generierte eine logisch gültige Kreditkartennummer, wenn sie eine gültige Kreditkartennummer maskiert. Die Länge der Quellkreditkartennummer muss zwischen 13 und 19 Ziffern liegen. Die Eingabekreditkartenummer muss eine gültige Prüfsumme basierend auf den gängigen Regeln für Kreditkartennummern sein.

Die Quellkreditkartennummer kann Zahlen, Leerzeichen und Bindestriche enthalten. Wenn die Kreditkarte falsche Zeichen oder eine falsche Länge aufweist, schreibt der Integration Service einen Fehler in das Sitzungsprotokoll. Der Integration Service wendet eine Standardmaske für Kreditkartennummern an, wenn die Quelldaten ungültig sind.

Die Datenmaskierungsumwandlung maskiert nicht die sechsstellige Bankleitzahl. Die Datenmaskierungsumwandlung kann zum Beispiel die Kreditkartennummer 4539 1596 8210 2773 als 4539 1516 0556 7067 maskieren. Die Datenmaskierungsumwandlung erstellt eine maskierte Nummer mit einer gültigen Prüfsumme.

Datenmaskierung der E-Mail-Adresse

Verwenden Sie die Datenmaskierungsumwandlung zum Maskieren der E-Mail-Adresse, die einen Zeichenfolgewert enthält. Die Datenmaskierungsumwandlung kann eine E-Mail-Adresse mit zufälligen ASCII-Zeichen maskieren oder die E-Mail-Adresse mit einer realistischen E-Mail-Adresse ersetzen.

Sie können die folgenden Maskierungstypen auf die E-Mail-Adresse anwenden:

Standard-E-Mail-Maskierung

Die Datenmaskierungsumwandlung gibt zufällige ASCII-Zeichen zurück, wenn damit eine E-Mail-Adresse maskiert wird. Die Datenmaskierungsumwandlung kann beispielsweise Georgesmith@yahoo.com als KtrIupQAPyk@vdSKh.BIC maskieren. Voreingestellt ist Standard.

Erweiterte E-Mail-Maskierung

Die Datenmaskierungsumwandlung maskiert die E-Mail-Adresse mit einer anderen realistischen E-Mail-Adresse, die von dem Umwandlungsausgabeport oder den Wörterbuchspalten abgeleitet wurde.

Erweiterte E-Mail-Maskierung

Mit dem erweiterten E-Mail-Maskierungstyp können Sie die E-Mail-Adresse mit einer anderen realistischen E-Mail-Adresse maskieren. Die Datenmaskierungsumwandlung erstellt die E-Mail-Adresse aus den Wörterbuchspalten oder aus der Umwandlung von Ausgabeports.

Sie können den lokalen Teil in der E-Mail-Adresse aus Mapping-Ausgabeports erstellen. Oder Sie erstellen den lokalen Teil in der E-Mail-Adresse aus einer relationalen Tabelle oder Einfachdateispalten.

Mit der Datenmaskierungsumwandlung kann der Domänenname für die E-Mail-Adresse aus einem konstanten Wert oder aus einem zufälligen Wert aus dem Domänenwörterbuch erstellt werden.

Sie können eine erweiterte E-Mail-Maskierung anhand der folgenden Optionen erstellen:

E-Mail-Adresse basierend auf abhängigen Ports

Sie können eine E-Mail-Adresse anhand der Datenmaskierungsumwandlung von Ausgabeports erstellen. Wählen Sie die Umwandlungs-Ausgabeports für die Spalten mit Vorname und Nachname. Die Datenmaskierungsumwandlung maskiert den Vornamen, den Nachnamen oder beide Namen, abhängig von den Werten, die Sie für die Länge der Vornamen und Nachnamen festgelegt haben.

E-Mail-Adresse basierend auf Wörterbuch

Sie können eine E-Mail-Adresse anhand der Spalten eines Wörterbuchs erstellen. Wählen Sie eine Referenztabelle als Quelle für das Wörterbuch.

Wählen Sie die Wörterbuchspalten für den Vornamen und Nachnamen aus. Die Datenmaskierungsumwandlung maskiert den Vornamen, den Nachnamen oder beide Namen, abhängig von den Werten, die Sie für die Länge der Vornamen und Nachnamen festgelegt haben.

Konfiguration der Parameter für einen erweiterten E-Mail-Adressen-Maskierungstyp

Geben Sie Konfigurationsparameter beim Konfigurieren des erweiterten E-Mail-Adresse-Maskierungstyps an.

Sie können die folgenden Konfigurationsparameter angeben:

Delimiter

Sie können einen Delimiter, z. B. einen Punkt, einen Bindestrich oder Unterstrich, auswählen, um den Vor- und Nachnamen in der E-Mail-Adresse voneinander zu trennen. Wenn Sie Vor- und Nachnamen in der E-Mail-Adresse nicht voneinander trennen möchten, lassen Sie den Delimiter leer.

Spalte Vorname (FirstName)

Wählen Sie einen Umwandlungsausgabeport der Datenmaskierung oder eine Wörterbuchspalte zum Maskieren des Vornamens in der E-Mail-Adresse.

Spalte Nachname (LastName)

Wählen Sie einen Umwandlungsausgabeport der Datenmaskierung oder eine Wörterbuchspalte zum Maskieren des Nachnamens in der E-Mail-Adresse.

Länge der Spalten FirstName und LastName

Begrenzt die Zeichenlänge zum Maskieren in den Spalten Vornamen und Nachnamen. Zum Beispiel lauten die Eingabedaten Timothy für den Vornamen und Smith für den Nachnamen. Wählen Sie 5 als Länge der Spalte mit den Vornamen. Wählen Sie 1 als Länge der Spalte mit den Nachnamen mit einem Punkt als Delimiter. Die Datenmaskierungsumwandlung generiert die folgende E-Mail-Adresse:

```
timot.s@<domain_name>
```

DomainName

Sie können einen konstanten Wert, wie z. B. gmail.com, für den Domänennamen verwenden. Oder Sie können eine andere Wörterbuchdatei verwenden, die eine Liste mit Domänennamen enthält. Das Domänenwörterbuch kann eine Einfachdatei oder eine relationale Tabelle sein.

Maskierung der IP-Adresse

Die Datenmaskierungsumwandlung maskiert eine IP-Adresse als eine andere IP-Adresse, indem Sie sie in vier durch einen Punkt getrennte Zahlen aufteilt. Die erste Zahl ist das Netzwerk. Die Datenmaskierungsumwandlung maskiert die Netzwerknummer innerhalb eines Netzwerkbereichs.

Die Datenmaskierungsumwandlung maskiert eine IP-Adresse der Klasse A als eine IP-Adresse der Klasse A und eine 10.x.x.x-Adresse als 10.x.x.x-Adresse. Die Datenmaskierungsumwandlung maskiert nicht die Klasse und die private Netzwerkadresse. Die Datenmaskierungsumwandlung kann beispielsweise 11.12.23.34 als 75.32.42.52 und 10.23.24.32 als 10.61.74.84 maskieren.

Hinweis: Wenn Sie IP-Adressen maskieren, kann die Datenmaskierungsumwandlung nicht eindeutige Werte zurückgeben, weil sie die Klasse oder das private Netzwerk der IP-Adressen nicht maskiert.

Maskieren der Telefonnummer

Die Datenmaskierungsumwandlung maskiert eine Telefonnummer, ohne das Format der ursprünglichen Telefonnummer zu ändern. Die Datenmaskierungsumwandlung kann beispielsweise die Telefonnummer (408) 382 0658 als (607) 256 3106 maskieren.

Die Quelldaten können Zahlen, Leerzeichen, Bindestriche und Klammern enthalten. Der Integration Service maskiert keine Buchstaben oder Sonderzeichen.

Die Datenmaskierungsumwandlung kann String-, Integer- und Bigint-Daten maskieren.

Maskierung der Sozialversicherungsnummer

Die Datenmaskierungsumwandlung erzeugt eine Sozialversicherungsnummer, die basierend auf der aktuellen Ausgabeliste (High Group List) der Sozialversicherung ungültig ist. Die Ausgabeliste (High Group List) enthält gültige Nummern, die von der Sozialversicherung ausgegeben wurden.

Die Standardausgabeliste (High Group List) ist eine Textdatei, die sich an folgendem Speicherort befindet:

```
<Installation Directory>\infa_shared\SrcFiles\highgroup.txt
```

Zum Verwenden dieser Textdatei in Arbeitsabläufen kopieren Sie sie in das Quellverzeichnis, das für den Data Integration Service konfiguriert wurde.

Die Datenmaskierungsumwandlung erzeugt Sozialversicherungsnummern, die sich nicht in der Ausgabeliste (High Group List) befinden. Die Sozialversicherungsverwaltung aktualisiert jeden Monat die Ausgabeliste (High Group List). Laden Sie die aktuelle Version der Liste von folgendem Speicherort herunter:

`http://www.socialsecurity.gov/employer/ssns/highgroup.txt`

Format der Sozialversicherungsnummern

Die Datenmaskierungs-Umwandlung akzeptiert jedes SVN-Format (Sozialversicherungsnummer), das aus neun Ziffern besteht. Die Ziffern können durch einen beliebigen Zeichensatz getrennt werden. Die Datenmaskierungs-Umwandlung akzeptiert beispielsweise das folgende Format: `+=54-*9944$#789-,*()”`.

Anforderung an Ortsvorwahlen

Die Datenmaskierungsumwandlung gibt eine Sozialversicherungsnummer (SSN) zurück, die mit demselben Format wie die Quelle nicht gültig ist. Die ersten drei Ziffern der SSN definieren die Vorwahl. Die Datenmaskierungsumwandlung maskiert nicht die Vorwahl. Sie maskiert die Gruppen- und Seriennummer. Die Quell-SSN muss eine gültige Vorwahl enthalten. Die Datenmaskierungsumwandlung sucht die Vorwahl in der High Group-Liste und legt einen Bereich von nicht verwendeten Nummern fest, die als maskierte Daten verwendet werden können. Wenn die SSN nicht gültig ist, maskiert die Datenmaskierungsumwandlung nicht die Quelldaten.

Wiederholbare Maskierung von Sozialversicherungsnummern

Die Datenmaskierungsumwandlung gibt deterministische Sozialversicherungsnummern mit wiederholbarer Maskierung zurück. Die Datenmaskierungsumwandlung kann nicht alle eindeutigen Sozialversicherungsnummern zurückgeben, weil sie keine gültigen Sozialversicherungsnummern zurückgeben kann, die von der Sozialversicherungsbehörde ausgegeben wurden.

Maskieren der URL-Adresse

Die Datenmaskierungs-Umwandlung parst eine URL, indem sie nach der Zeichenfolge "://" sucht und die Subzeichenfolge rechts davon parst. Die Quell-URL muss die Zeichenfolge "://" enthalten. Die Quell-URL kann Zahlen und alphabetische Zeichen enthalten.

Die Datenmaskierungs-Umwandlung maskiert nicht das Protokoll der URL. Wenn beispielsweise die URL `http://www.yahoo.com` lautet, kann die Datenmaskierungs-Umwandlung `http://MgL.aHjCa.VsD/` zurückgeben. Die Datenmaskierungs-Umwandlung kann eine ungültige URL generieren.

Hinweis: Die Datenmaskierungs-Umwandlung gibt immer ASCII-Zeichen für eine URL zurück.

Maskierung der Sozialversicherungsnummer

Die Datenmaskierungs-Umwandlung maskiert eine neunstellige Sozialversicherungsnummer. Die Ziffern können durch einen beliebigen Zeichensatz getrennt werden.

Wenn die Nummer keine Trennzeichen enthält, weist die maskierte Nummer ebenfalls keine Trennzeichen auf. Andernfalls wird folgendes Format für die maskierte Nummer verwendet:

```
xxx-xxx-xxx
```

Wiederholbaren SIN-Nummern

Sie können Datenmaskierungsumwandlung konfigurieren, um wiederholbare SIN-Werte zurückzugeben. Wenn Sie einen Port für wiederholbare SIN-Maskierung konfigurieren, gibt die Datenmaskierungsumwandlung jedes Mal deterministisch maskierte Daten zurück, wenn der SIN-Quellwert dem Seed-Wert entspricht.

Um wiederholbare SIN-Nummern zurückzugeben, aktivieren Sie die Option **Wiederholbare Werte** und geben Sie eine Seed-Nummer ein. Die Datenmaskierungsumwandlung gibt eindeutige Werte für jede SIN zurück.

SIN-Anfangsziffer

Sie können die erste Ziffer der maskierten SIN definieren.

Aktivieren Sie **Ziffer starten** und geben Sie die Ziffer ein. Die Datenmaskierungs-Umwandlung erstellt maskierte SIN-Nummern, die mit der von Ihnen eingegebenen Nummer beginnen.

Standardwertdatei

Wenn Quelldatenformat oder Datentyp für eine Maskierung ungültig sind, wendet der Data Integration Service eine Standardmaskierung auf die Daten an. Der Integration Service wendet maskierte Werte aus der Standardwertdatei an. Sie können die Standardwertdatei bearbeiten, um die Standardwerte zu ändern.

Die Standardwert ist eine XML-Datei an folgendem Speicherort:

```
<Installationsverzeichnis>\infa_shared\SrcFiles\defaultValue.xml
```

Kopieren Sie zum Verwenden der Standardwertdatei in Arbeitsabläufen diese in das Quellverzeichnis, das für den Data Integration Service konfiguriert wurde.

Die Datei "defaultValue.xml" enthält die folgenden Name-Wert-Paare:

```
<?xml version="1.0" standalone="yes" ?>
<defaultValue
default_char = "X"
default_digit = "9"
default_date = "11/11/1111 00:00:00"
default_email = "abc@xyz.com"
default_ip = "99.99.9.999"
default_url = "http://www.xyz.com"
default_phone = "999 999 999 9999"
default_ssn = "999-99-9999"
default_cc = "9999 9999 9999 9999"
default_sin = "999-999-999"
default_seed = "500"/>
```

Verwandte Themen:

- "Konfigurieren des Datenintegrationsdiensts" auf Seite 191

Konfiguration der Datenmaskierungsumwandlung

Führen Sie die folgenden Schritte aus, um eine Datenmaskierungsumwandlung zu konfigurieren.

1. Konfigurieren Sie Ausnahmeoptionen für den Data Integration Service.
2. Erstellen Sie die Umwandlung.
3. Definieren Sie die Eingabeports.
4. Konfigurieren Sie Maskierungsregeln für jeden Port, den Sie ändern möchten.
5. Zeigen Sie die Daten in der Vorschau an, um die Ergebnisse zu überprüfen.

Konfigurieren des Datenintegrationsdiensts

Sie können Ausführungsoptionen für den Datenintegrationsdienst in Informatica Administrator (das Administrator Tool) konfigurieren.

Konfigurieren Sie Ausführungsoptionen zum Einrichten der folgenden Standardverzeichnisse:

- Basisverzeichnis. Enthält das Quell- und das Cache-Verzeichnis.
- Quellverzeichnis. Enthält die Quelldateien für Arbeitsabläufe. Das Quellverzeichnis kann beispielsweise die Dateien "highgrp.txt" und "defaultvalue.xml" enthalten.
- Cache-Verzeichnis. Enthält Cache-Dateien für die Substitutions-Maskierung.

Öffnen Sie zum Einrichten von Werten für die Ausführungsoptionen das Administrator Tool und wählen Sie den Datenintegrationsdienst im **Domänennavigator** aus. Klicken Sie auf die Ansicht **Eigenschaften** und dann im Abschnitt **Ausführungsoptionen** auf **Bearbeiten**.

Verwandte Themen:

- "Standardwertdatei" auf Seite 190

Erstellen einer Datenmaskierungsumwandlung

Erstellen Sie eine Datenmaskierungsumwandlung im Developer Tool.

Erstellen Sie die Quelle, bevor Sie die Datenmaskierungsumwandlung erstellen. Importieren Sie eine Einfachdatei oder relationale Datenbanktabelle als physisches Datenobjekt.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die Datenmaskierungsumwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Geben Sie einen Namen für die Umwandlung ein.
6. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.

Definieren der Ports

Fügen Sie Datenmaskierungseingabeports in der Ansicht **Übersicht** hinzu. Beim Erstellen eines Eingabeports erstellt das Developer Tool standardmäßig einen entsprechenden Ausgabeport. Der Ausgabeport hat den gleichen Namen wie der Eingabeport.

1. Klicken Sie in der Ansicht **Übersicht** auf **Neu**, um einen Port hinzuzufügen.
2. Konfigurieren Sie den Datentyp, die Präzision und die Größenordnung für die Spalte.

   Sie müssen den Datentyp der Spalte konfigurieren, bevor Sie Maskierungsregeln für die Spalte definieren.
3. Klicken Sie zum Konfigurieren der Datenmaskierung für den Port auf den Pfeil in der Maskierungstypspalte der Ansicht **Übersicht**.

Konfigurieren der Datenmaskierung für jeden Port

Wählen Sie eine Maskierungstechnik und die entsprechenden Maskierungsregeln für einen Port im Dialogfeld "Datenmaskierung" aus. Das Dialogfeld "Datenmaskierung" wird angezeigt, wenn Sie auf die Datenmaskierungsspalte auf der Registerkarte **Ports** klicken.

1. Aktivieren Sie **Maskierung anwenden**, um die Maskierung für den ausgewählten Port zu konfigurieren.

   Das Developer-Tool zeigt eine Liste der Maskierungstechniken an, die Sie ausgehend vom Datentyp des Ports, den Sie maskieren, verwenden können.
2. Wählen Sie eine Maskierungstechnik aus der Liste aus.

   Das Developer-Tool zeigt basierend auf Ihrer Auswahl der Maskierungstechnik verschiedene Maskierungsregeln an. Einige spezielle Maskierungsformate haben keine konfigurierbaren Maskierungsregeln.
3. Konfigurieren Sie die Maskierungsregeln.
4. Klicken Sie auf **OK**, um die Datenmaskierungskonfiguration für den Port anzuwenden.

   Wenn Sie die Datenmaskierung für einen Port festlegen, erstellt das Developer-Tool einen Ausgabeport namens **out-<Portname>**. Der **<Portname>** entspricht dem Namen des Eingabeports. Die Datenmaskierungsumwandlung gibt die maskierten Daten im Port **out-<Portname>** zurück.

Vorschau der maskierten Daten

Sie können die maskierten Daten mit den Originaldaten beim Anzeigen der Ergebnisse der Datenmaskierungsumwandlung im **Daten-Viewer** vergleichen.

1. Nach dem Konfigurieren der Datenmaskierungsumwandlungsports und Maskierungsregeln erstellen Sie ein Mapping, das die physische Datenobjektquelle und die Datenmaskierungsumwandlung enthält.
2. Verbinden Sie die Quelle mit der Datenmaskierungsumwandlung.
3. Stellen Sie sicher, dass die Quelle über Daten in einem freigegebenen Speicherort verfügt, auf den der Datenintegrationsdienst zugreifen kann.
4. Klicken Sie auf die Datenmaskierungsumwandlung, um sie im Mapping auszuwählen.
5. Klicken Sie auf **Daten-Viewer** und anschließend auf **Ausführen**.

   Das Developer-Tool zeigt die Daten für alle Ausgabeports der Datenmaskierungsumwandlung an. Die Ports mit dem **Ausgabe**-Präfix enthalten die maskierten Daten. Sie können die maskierten Daten mit den Originaldaten in der Ansicht **Daten-Viewer** vergleichen.

Laufzeiteigenschaften der Datenmaskierungsumwandlung

Sie können Laufzeiteigenschaften der Datenmaskierungsumwandlung konfigurieren, um die Leistung zu erhöhen.

Konfigurieren Sie die folgenden Laufzeiteigenschaften:

Cachegröße

Die Größe des Wörterbuch-Caches im primären Speicher. Erhöhen Sie die Speichergröße, um die Leistung zu verbessern. Die Mindestanforderung ist 32 MB für 100.000 Datensätze. Voreingestellt ist 8 MB.

Cache-Verzeichnis

Der Speicherort des Wörterbuch-Caches. Sie müssen über eine Schreibberechtigung für das Verzeichnis verfügen. Der Standardwert ist CacheDir.

Gemeinsam genutzte Speichertabelle

Ermöglicht die Freigabe der Speichertabelle für die Umwandlungsinstanzen der Datenmaskierung. Aktivieren Sie "Gemeinsam genutzte Speichertabelle", wenn zwei Umwandlungsinstanzen der Datenmaskierung die gleiche Wörterbuchspalte für die Datenbankverbindung, den Seed-Wert und das Gebietsschema verwenden. Sie können auch die gemeinsam genutzte Speichertabelle aktivieren, wenn zwei Ports in der gleichen Datenmaskierungsumwandlung die gleiche Wörterbuchspalte für die Verbindung, Seed und Gebietsschema nutzen. Deaktivieren Sie die gemeinsam genutzte Speichertabelle, wenn die Datenmaskierungsumwandlung oder die Ports nicht die gleiche Wörterbuchspalte verwenden. Diese Einstellung ist standardmäßig deaktiviert.

Speicher-Commit-Intervall

Die Anzahl der Zeilen, die gleichzeitig auf die Speichertabelle festgelegt werden. Erhöhen Sie den Wert, um die Leistung zu erhöhen. Konfigurieren Sie das Commit-Intervall, wenn Sie die gemeinsam genutzte Speichertabelle nicht konfigurieren. Voreingestellt ist 100.000.

Speicher verschlüsseln

Verschlüsselt Speichertabellen, z. B. IDM_SUBSTITUTION_STORAGE und IDM_EXPRESSION_STORAGE. Stellen Sie sicher, dass Sie die Daten in Speichertabellen verschlüsselt haben, bevor Sie die Eigenschaft "Speicher verschlüsseln" aktivieren. Löschen Sie diese Option, wenn Sie die Speichertabellen nicht verschlüsseln möchten. Diese Einstellung ist standardmäßig deaktiviert.

Speicher-Verschlüsselungsschlüssel

Die Datenmaskierungsumwandlung verschlüsselt den Speicher abhängig von dem Schlüssel für die Speicherverschlüsselung. Verwenden Sie den gleichen Verschlüsselungsschlüssel für jede Sitzung derselben Umwandlungsinstanz der Datenmaskierung.

Eigentümername des Substitutionswörterbuchs

Name des Tabelleneigentümers des Substitutionswörterbuchs, wenn Sie den Substitutions-Maskierungstyp auswählen. Wenn der in der Datenbankverbindung angegebene Datenbankbenutzer nicht der Eigentümer der Substitutionswörterbuchtabelle in einer Sitzung ist, müssen Sie den Tabelleneigentümer angeben.

Speicher-Eigentümername

Name des Tabelleneigentümers für IDM_SUBSTITUTION_STORAGE oder IDM_EXPRESSION_STORAGE, wenn Sie wiederholbare Ausdrücke oder eindeutige, wiederholbare Substitutions-Maskierungstypen auswählen.

Datenmaskierung - Beispiel

Ein Entwickler muss Testdaten für Kundenanwendungen erstellen. Die Daten müssen realistische Kundeninformationen enthalten, auf die andere Entwickler in der Entwicklungsumgebung des Unternehmens zugreifen können.

Der Entwickler erstellt einen Datendienst, der maskierte Kundendaten wie Kunden-ID, Kreditkartennummer und Einkünfte zurückgibt. Das Mapping enthält eine Datenmaskierungsumwandlung, mit der die Kundendaten umgewandelt werden.

Die folgende Abbildung zeigt das Mapping:

Das Mapping hat die folgenden Umwandlungen:

- Read_Customer_Data. Enthält Informationen zu Kreditkarte und Einkünften des Kunden.
- Customer_Data_Masking Umwandlung. Maskiert alle Spalten außer FirstName und LastName. Die Datenmaskierungsumwandlung übergibt die maskierten Spalten an das Target.
- Customer_TestData. Ausgangsumwandlung, die die maskierten Kundendaten erhält.

Read_Customer Data

Die Kundendaten umfassen die folgenden Spalten:

Spalte	Datentyp
CustomerID	Integer
LastName	String
FirstName	String
CreditCard	String
Income	Integer
Join_Date	Datetime (MM/DD/YYYY)

Die folgende Tabelle enthält die Beispielkundendaten:

CustomerID	LastName	FirstName	CreditCard	Income	JoinDate
0095	Bergeron	Barbara	4539-1686-3069-3957	12000	12/31/1999
0102	Brosseau	Derrick	5545-4091-5232-8948	4000	03/03/2011
0105	Anderson	Lauren	1234-5678-9012-3456	5000	04/03/2009
0106	Boonstra	Pauline	4217-9981-5613-6588	2000	07/07/2007
0107	Chan	Brian	4533-3156-8865-3156	4500	06/18/1995

Kundendaten-Maskierungsumwandlung

Die Datenmaskierungsumwandlung maskiert alle Spalten in der Kundenzeile mit Ausnahme von Vorname und Nachname.

Die Datenmaskierungsumwandlung führt die folgenden Maskierungstypen durch:

- Schlüsselmaskierung
- Zufällige Maskierung
- Kreditkarten-Maskierung

In der folgenden Tabelle sind die Maskierungsregeln für jeden Port in der Datenmaskierungsumwandlung dargestellt:

Eingabeport	Maskierungstyp	Maskierungsregeln	Beschreibung
CustomerID	Schlüssel	Der Seed ist 934. Die Kunden-ID hat kein Maskierungsformat. Ersatzzeichen für Ergebniszeichenfolge sind 1234567890.	Die CustomerID-Maskierung ist deterministisch. Die maskierte Kunden-ID enthält Zahlen.
LastName	Keine Maskierung	-	-
FirstName	Keine Maskierung	-	-
CreditCard	CreditCard	-	Die Datenmaskierungsumwandlung maskiert die Kreditkartennummer mit einer anderen Nummer, die eine gültige Prüfsumme hat.
Income	Zufällig	Unschärfe Prozent Untere Grenze = 1 Obere Grenze = 10	Die maskierten Einkünfte liegen im Bereich von zehn Prozent der Quelleinkünfte.
JoinDate	Zufällig	Unschärfe Einheit = Jahr Untere Grenze = 5 HighBound=5	Das maskierte Datum liegt im Fünfjahresbereich ausgehend vom ursprünglichen Datum.

Ergebnisse der Kundentestdaten

Die Customer_TestData-Umwandlung erhält realistische Kundendaten aus der Datenmaskierungsumwandlung.

Das Customer_TestData-Target erhält die folgenden Daten:

out-CustomerID	out-LastName	outFirstName	out-CreditCard	out-Income	out-JoinDate
3954	Bergeron	Barbara	4539-1625-5074-4106	11500	03/22/2001
3962	Brosseau	Derrick	5545-4042-8767-5974	4300	04/17/2007
3964	Anderson	Lauren	1234-5687-2487-9053	5433	09/13/2006
3965	Boonstra	Pauline	4217-9935-7437-4879	1820	02/03/2010
3966	Chan	Brian	4533-3143-4061-8001	4811	10/30/2000

Die Einkünfte liegen im Bereich von zehn Prozent der Originaleinkünfte. Das Beitrittsdatum liegt im Fünfjahresbereich ausgehend vom ursprünglichen Datum.

Datenmaskierungsumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Datenmaskierungsumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 11

Datenprozessor-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Datenprozessorumwandlung - Überblick, 197
- Datenprozessorumwandlung - Ansichten, 198
- Datenprozessorumwandlung - Ports, 199
- Startkomponente, 201
- Verweise, 202
- Datenprozessorumwandlung – Einstellungen, 202
- Ereignisse, 209
- Protokolle, 210
- Entwicklung der Datenprozessorumwandlung, 212
- Datenprozessorumwandlung - Importieren und Exportieren, 217

Datenprozessorumwandlung - Überblick

Die Datenprozessorumwandlungsprozesse verarbeiten unstrukturierte und halb strukturierte Dateiformate in einem Mapping. Konfigurieren Sie die Umwandlung für die Verarbeitung von Messaging-Formaten, HTML-Seiten, XML- JSON- und PDF-Dokumenten. Sie können auch strukturierte Formate wie ACORD, HIPAA, HL7, EDI-X12, EDIFACT, AFP und SWIFT konvertieren.

Ein Mapping verwendet eine Datenprozessorumwandlung, um Dokumente aus einem Format in ein anderes zu konvertieren. Die Datenprozessorumwandlung verarbeitet Dateien eines beliebigen Formats in einem Mapping. Wenn Sie eine Datenprozessorumwandlung erstellen, definieren Sie die Komponenten, die die Daten konvertieren.

Eine Datenprozessorumwandlung kann mehrere Komponenten zum Verarbeiten von Daten enthalten. Jede Komponente kann wiederum andere Komponenten enthalten.

Beispiel: Sie können Kundenrechnungen in Microsoft Word-Dateien erhalten. Sie konfigurieren eine Datenprozessorumwandlung, um die Daten von allen Word-Dateien zu parsen. Extrahieren Sie die Kundendaten in eine Kundentabelle. Extrahieren Sie Bestellinformationen in eine Tabelle der Bestellungen.

Wenn Sie eine Datenprozessorumwandlung erstellen, definieren Sie eine XMap, ein Skript oder eine Bibliothek. Eine XMap wandelt eine hierarchische Eingabedatei in eine hierarchische Ausgabedatei mit einer anderen Struktur um. Eine Bibliothek konvertiert einen branchenüblichen Nachrichtentyp in ein XML-Dokument mit einer Hierarchiestruktur oder aus XML in ein branchenübliches Standardformat. Ein Skript kann Quelldokumente in ein hierarchisches Format parsen, das hierarchische Format in andere Dateiformate konvertieren oder ein hierarchisches Dokument einem anderen hierarchischen Format zuordnen.

Definieren Sie Skripts im IntelliScript-Editor der Datenprozessorumwandlung. Sie können die folgenden Skript-Typen definieren:

- Parser. Wandelt Quelldokumente in XML um. Parser geben immer XML aus. Die Eingabe kann ein beliebiges Format aufweisen, zum Beispiel Text, HTML, Word, PDF oder HL7.
- Serializer. Wandelt eine XML-Datei in ein Ausgabedokument beliebigen Formats um. Die Ausgabe eines Serializer kann ein beliebiges Format aufweisen, z. B. ein Text-, HTML- oder PDF-Dokument.
- Mapper. Wandelt ein XML-Quelldokument in eine andere XML-Struktur oder ein anderes XML-Schema um. Sie können dieselben XML-Dokumente wie in einer XMap konvertieren.
- Transformer. Ändert die Daten in jedes beliebige Format. Fügt Text hinzu, wandelt Text, entfernt und ändert Text. Verwenden Sie Transformer mit einem Parser, Mapper oder Serializer. Sie können einen Transformer auch als eigenständige Komponente ausführen.
- Streamer. Splittet umfangreiche Eingabedokumente, z. B. mehrere Gigabyte große Datenströme, in Segmente. Der Streamer verarbeitet Dokumente, die mehrere Meldungen oder Datensätze enthalten, wie z. B. HIPAA- oder EDI-Dateien.

Weitere Informationen finden Sie im *Data Transformation-Benutzerhandbuch*.

Datenprozessorumwandlung - Ansichten

Die Datenprozessorumwandlung verfügt über zahlreiche Ansichten, auf die Sie beim Konfigurieren der Umwandlung zugreifen und die Sie im Developer Tool ausführen.

Einige der Ansichten der Datenprozessorumwandlung werden nicht standardmäßig im Developer Tool angezeigt. Um die Ansichten für die Umwandlung zu wechseln, klicken Sie auf **Fenster** > **Ansicht einblenden** > **Andere** > **Informatica**. Wählen Sie die Ansichten aus, die Sie verwenden möchten.

Die Datenprozessorumwandlung hat folgende feste Ansichten:

Übersicht-Ansicht

Zum Konfigurieren von Ports und Definieren der Startkomponente.

Referenzen-Ansicht

Zum Hinzufugen oder Entfernen von Schemata aus der Umwandlung.

Einstellungen-Ansicht

Zum Konfigurieren von Umwandlungseinstellungen für die Codierung, Ausgabesteuerung und XML-Generierung.

Objekte-Ansicht

Hinzufügen, Ändern oder Löschen von Skript-, XMap- und Bibliotheks-Objekten aus der Umwandlung.

Sie haben auch Zugriff auf folgende Ansichten der Datenprozessorumwandlung:

Datenprozessor-Hex-Quelle-Ansicht

Zeigt ein Eingabedokument im Hexadezimalformat.

Ansicht der Datenprozessor-Ereignisse

Zeigt Informationen über Ereignisse, die bei der Umwandlung im Developer Tool auftreten. Zeigt Initialisierungs-, Ausführungs- und Übersichtereignisse.

Skript-Hilfe-Ansicht

Zeigt kontextbezogene Hilfe für den Skript-Editor an.

Daten-Viewer-Ansicht

Zum Anzeigen von Beispielseingabedaten, Ausführen der Umwandlung und Anzeigen der Ausgabeergebnisse.

Datenprozessorumwandlung - Ports

Definieren Sie die Datenprozessorumwandlungsports in der Ansicht **Übersicht**.

Eine Datenprozessorumwandlung kann Eingaben aus einer Datei, einem Puffer oder einem Streamline-Puffer mithilfe eines Readers für komplexe Dateien lesen. Sie können einen Einfachdatei-Reader gleichzeitig als Puffer zum Lesen einer ganzen Datei verwenden. Sie können eine Eingabedatei auch aus einer Datenbank lesen.

Die Ausgabeports, die Sie erstellen, hängen davon ab, ob Sie einen String, komplexe Dateien oder Zeilen relationaler Daten aus der Umwandlung zurückgeben möchten.

Datenprozessorumwandlung - Eingabeports

Wenn Sie eine Datenprozessorumwandlung erstellen, erstellt das Developer Tool einen Standard-Eingabeport. Wenn Sie einen weiteren Eingabeport in einer Skript-Startkomponente definieren, erstellt das Developer-Tool einen weiteren Eingabeport in der Umwandlung.

Der Eingabetyp bestimmt den Datentyp, der vom Datenintegrationsdienst an die Datenprozessorumwandlung übergeben wird. Der Eingabetyp bestimmt, ob es sich bei der Eingabe um Daten oder einen Quelldateipfad handelt.

Konfigurieren Sie einen der folgenden Eingabetypen:

Puffer

Die Datenprozessorumwandlung erhält Zeilen mit Quelldaten im Eingabeport. Wählen Sie den Puffereingabe-Typ, wenn Sie die Umwandlung für das Empfangen von Daten von einer flachen Datei oder von einer Informatica-Umwandlung konfigurieren.

Datei

Die Datenprozessorumwandlung empfängt den Quelldateipfad im Eingabeport. Die Startkomponente des Datenprozessors öffnet die Quelldatei. Verwenden Sie den Eingabetyp "Datei", um Binärdateien (z. B. Microsoft Excel- oder Microsoft Word-Dateien) zu analysieren. Sie können zur Verarbeitung mit einem Puffereingabeport auch den Datei-Eingabetyp für große Dateien verwenden, die möglicherweise viel Systemspeicher benötigen.

Dienstparameter

Die Datenprozessorumwandlung empfängt Werte zur Anwendung auf Variable in den Dienstparameter-Ports. Wenn Sie auswählen, dass die Variablen Eingabedaten empfangen, erstellt das Developer Tool einen Dienstparameterport für jede Variable.

Output_Filename

Wenn Sie den Standard-Ausgabeport so konfigurieren, dass er einen Dateinamen statt Rohdaten ausgibt, so erstellt das Developer Tool eine Output_Filename-Port. Sie können aus einem Mapping einen Dateinamen an den Output_Filename-Port übergeben.

Bei der Definition eines Eingabeports können Sie den Ort der Beispiels-Eingabedatei für den Port festlegen. Eine Beispiels-Eingabedatei ist ein kleines Beispiel der Eingabedatei. Referenzieren Sie eine Beispiels-Eingabedatei, wenn Sie Skripte erstellen. Sie verwenden die Beispiels-Eingabedatei auch, wenn Sie die Umwandlung in der **Daten-Viewer**-Ansicht testen. Definieren Sie die Beispiels-Eingabedatei im Feld **Eingabespeicherort**.

Ports für Dienstparameter

Sie können Eingabeports erstellen, die Werte für Variable empfangen. Die Variablen können einen beliebigen Datentyp enthalten, beispielsweise einen String, ein Datum oder eine Zahl. Darüber hinaus kann eine Variable eine Position für ein Quelldokument enthalten. Diese Variablen können Sie in einer Datenprozessor-Komponente referenzieren.

Wenn Sie für eine Variable einen Eingabeport erstellen, zeigt das Developer Tool eine Liste mit Variablen an, aus der Sie die Variable auswählen können.

Erstellen von Serviceparameterports

Sie können Eingabeports erstellen, die Werte für Variablen entgegennehmen. Sie können auch die Ports entfernen, die Sie aus Variablen erstellen.

1. Öffnen Sie die Ansicht **Übersicht** für die Datenprozessorumwandlung.
2. Klicken Sie auf **Auswählen.**

   Das Developer Tool zeigt eine Liste der Variablen an und legt fest, welche Variablen bereits Ports haben.
3. Wählen Sie eine oder mehrere Variablen aus.

   Das Developer Tool erstellt einen Puffereingabeport für jede von Ihnen ausgewählte Variable. Sie können den Port nicht ändern.
4. Um einen Port zu entfernen, den Sie aus einer Variablen erstellen, deaktivieren Sie die Auswahl aus der Variablenliste. Wenn Sie die Auswahl deaktivieren, entfernt das Developer Tool den Eingabeport.

Datenprozessorumwandlung - Ausgabeports

Die Datenprozessorumwandlung hat standardmäßig einen Ausgabeport. Wenn Sie zusätzliche Ausgabeports in einem Skript definieren, fügt das Developer Tool die Ports der Datenprozessorumwandlung hinzu. Sie können Gruppen und Ports erstellen, wenn Sie die Umwandlung zur Rückgabe relationaler Daten konfigurieren. Sie können auch Dienstparameterports und Pass-Through-Ports erstellen.

Standard-Ausgabeport

Die Datenprozessorumwandlung hat standardmäßig einen Ausgabeport. Wenn Sie relationale Ausgaben erstellen, können Sie Gruppen verbundener Ausgabeports anstatt des Standard-Ausgabeports definieren. Wenn Sie einen weiteren Ausgabeport in einer Skriptkomponente definieren, fügt das Developer-Tool einen weiteren Ausgabeport zur Umwandlung hinzu.

Konfigurieren Sie einen der folgenden Ausgabetypen für einen Standard-Ausgabeport:

Puffer

Die Datenprozessorumwandlung gibt XML über den Ausgabeport aus. Wählen Sie den Pufferdateityp, wenn Sie Dokumente parsen oder in der Datenprozessorumwandlung XML anderen XML-Dokumenten zuordnen.

Datei

Der Datenintegrationsdienst gibt einen Namen der Ausgabedatei im Ausgabeport für jede Quellinstanz oder -zeile aus. Die Datenprozessorumwandlungs-Komponente schreibt die Ausgabedatei, statt Daten über die Ausgabeports der Datenprozessorumwandlung auszugeben.

Wenn Sie einen Dateiausgabeport auswählen, erstellt das Developer Tool einen Output_Filename-Eingabeport. Sie können einen Dateinamen an den Output_Filename-Port übergeben. Die Datenprozessorumwandlung erstellt die Ausgabedatei mit einem Namen, den es in diesem Port empfängt.

Wenn der Name der Ausgabedatei leer ist, gibt der Datenintegrationsdienst einen Fehler aus. Wenn ein Fehler auftritt, schreibt der Datenintegrationsdienst einen Nullwert auf den Ausgabeport und gibt einen Zeilenfehler aus.

Wählen Sie den Ausgabetyp "Datei" aus, wenn Sie XML in eine Binärdatendatei, etwa eine PDF- oder Microsoft Excel-Datei, umwandeln möchten.

Pass-Through-Ports

Sie können Pass-Through-Ports für beliebige Datenprozessorumwandlungen konfigurieren. Pass-Through-Ports sind Eingabe- und Ausgabe-Ports, die Eingabedaten empfangen und dieselben Daten in ein zurückgeben, ohne sie zu ändern.

Sie können Pass-Through-Ports in einer Datenprozessorumwandlungsinstanz definieren, die sich in einem Mapping befindet.

Um einen Pass-Through-Port hinzuzufügen, ziehen Sie einen Port von einer anderen Umwandlung in das Mapping. Sie können Ports auch über die Registerkarte **Ports** in der Ansicht **Eigenschaften** hinzufügen. Klicken Sie auf **Neu**, um einen Pass-Through-Port hinzuzufügen.

Hinweis: Wenn Sie Pass-Through-Ports zu einer Datenprozessorumwandlung mit relationaler Eingabe und hierarchischer Ausgabe hinzufügen, fügen Sie die Ports zur Root-Gruppe der relationalen Struktur hinzu.

Datenprozessorumwandlungen können Pass-Through-Ports mit benutzerdefinierten Datentypen enthalten.

Startkomponente

Eine Startkomponente definiert die Komponente, die die Verarbeitung in der Datenprozessorumwandlung startet. Konfigurieren Sie die Startkomponente in der Ansicht **Übersicht**.

Eine Datenprozessorumwandlung kann mehrere Komponenten zum Verarbeiten von Daten enthalten. Jede Komponente kann wiederum andere Komponenten enthalten. Sie müssen die Komponente ermitteln, die den Eintrittspunkt für die Umwandlung darstellt.

Wenn Sie die Startkomponente in einer Datenprozessorumwandlung konfigurieren, können Sie als Startkomponente eine XMap, eine Bibliothek oder eine Skriptkomponente auswählen. In Bezug auf Skripte können Sie einen der folgenden Komponententypen auswählen:

- Parser. Wandelt Quelldokumente in XML um. Die Eingabe kann ein beliebiges Format aufweisen, beispielsweise Text, HTML, Word, PDF oder HL7.
- Mapper. Wandelt ein XML-Quelldokument in eine andere XML-Struktur oder ein anderes XML-Schema um.
- Serializer. Wandelt eine XML-Datei in ein Ausgabedokument beliebigen Formats um.

- Streamer. Splittet umfangreiche Eingabedokumente, z. B. mehrere Gigabyte große Datenströme, in Segmente.
- Transformer. Ändert die Daten in jedes beliebige Format. Dient dem Hinzufügen, Entfernen, Umwandeln oder Ändern von Text. Verwenden Sie Transformer mit einem Parser, Mapper oder Serializer. Sie können einen Transformer auch als eigenständige Komponente ausführen.

Hinweis: Wenn die Startkomponente keine XMap oder Bibliothek ist, können Sie die Startkomponente auch stattdessen in einem Skript in der Ansicht **Übersicht** konfigurieren.

Verweise

Sie können Umwandlungsverweise (z. B. Schema- oder Mapplet-Verweise) definieren, indem Sie ein Schema oder Mapplet auswählen, das als Verweis dienen soll. Bestimmte Datenprozessorumwandlungen erfordern ein hierarchisches Schema, um die Eingabe- oder Ausgabehierarchie für relevante Komponenten in der Umwandlung zu definieren. Zum Verwenden des Schemas in der Umwandlung definieren Sie einen Schemaverweis für die Umwandlung. Sie können auch eine spezielle Aktion (die RunMapplet-Aktion) verwenden, um ein Mapplet aus einer Datenprozessorumwandlung abzurufen. Zum Abrufen eines Mapplets müssen Sie zuerst einen Mapplet-Verweis für die Umwandlung definieren.

Sie können Umwandlungsverweise, wie z. B. Schema- oder Mapplet-Verweise, in der Ansicht **Verweise** der Umwandlung definieren.

Schemaverweise

Die Datenprozessorumwandlung verweist auf Schemaobjekte im Modellrepository. Die Schemaobjekte können im Repository vorhanden sein, bevor Sie die Umwandlung erstellen. Sie können Schemas auch aus der Ansicht **Verweise** der Umwandlung importieren.

Die Schemacodierung muss mit der Eingabecodierung für Serializer- oder Mapper-Objekte übereinstimmen. Die Schemacodierung muss mit der Ausgabecodierung für Parser- oder Mapper-Objekte übereinstimmen. Konfigurieren Sie die Arbeitscodierung in der Ansicht **Einstellungen** der Umwandlung.

Ein Schema kann auf zusätzliche Schemas verweisen. Das Developer-Tool zeigt den Namensraum und das Präfix für jedes Schema, auf das die Datenprozessorumwandlung verweist. Wenn Sie mit leeren Namensräumen auf mehrere Schemas verweisen, ist die Umwandlung ungültig.

Mapplet-Verweise

Sie können ein Mapplet mit der RunMapplet-Aktion aus einer Datenprozessorumwandlung abrufen. Bevor Sie die RunMapplet-Aktion zu einer Datenprozessorumwandlungs-Komponente hinzufügen, müssen Sie zuerst einen Verweis auf das aufzurufende Mapplet definieren.

Datenprozessorumwandlung – Einstellungen

Konfigurieren Sie Codepages, XML-Verarbeitungsoptionen und Protokollierungseinstellungen in der Ansicht **Einstellungen** der Datenprozessorumwandlung.

Zeichencodierung

Eine Zeichenkodierung ist eine Zuordnung der Zeichen von einer Sprache oder Sprachgruppe zu Hexadezimalcode.

Wenn Sie ein Skript zuweisen, definieren Sie die Codierung der Eingabe- und Ausgabedokumente. Definieren Sie die Arbeitscodierung, um festzulegen, wie der IntelliScript-Editor Zeichen anzeigt und wie die Datenprozessorumwandlung die Zeichen verarbeitet.

Arbeitscodierung

Die Arbeitscodierung ist die Codepage für die Daten im Arbeitsspeicher und die Codepage für die Daten, die in der Benutzerschnittstelle und in Arbeitsdateien erscheinen. Sie müssen eine Arbeitscodierung auswählen, die mit der Codierung der Schemata übereinstimmt, die Sie bei der Datenprozessorumwandlung aufrufen.

Aus der folgenden Tabelle sind die Einstellungen für die Arbeitscodierung ersichtlich:

Einstellung	Beschreibung
Standardcodepage der Datenverarbeitung verwenden	Verwendet die Standardcodierung aus der Datenprozessorumwandlung.
Andere	Wählen Sie die Codierung aus der Liste aus.
XML-Codierung von Sonderzeichen	Legt die Darstellung von XML-Sonderzeichen fest. Sie können **Keine** oder **XML** auswählen. - **Keine.** Als & < > " ' lassen Entitätsreferenzen für XML-Sonderzeichen werden als Text interpretiert. Das Zeichen > erscheint zum Beispiel als `>` Standardwert ist "keine". - **XML**. In & < > " ' umwandeln Entitätsreferenzen für XML-Sonderzeichen werden als normale Zeichen interpretiert. Beispielsweise erscheint > als folgendes Zeichen: `>`

Eingabecodierung

Die Eingabecodierung bestimmt, wie Zeichendaten in Eingabedokumenten codiert werden. Sie können die Codierung für zusätzliche Eingabeports in einem Skript konfigurieren.

Die nachstehende Tabelle beschreibt die Codierungs-Einstellungen für den **Eingabe**-Bereich:

Einstellung	Beschreibung
Codierung aus Eingabedokument verwenden	Verwendet die Codepage, die im Quelldokument festgelegt wurde, wie z. B. das Codierungsattribut eines XML-Dokuments. Wenn das Quelldokument keine Codierungsspezifikation enthält, verwendet die Datenprozessorumwandlung die Codierungseinstellungen aus der Ansicht **Einstellungen**.
Arbeitscodierung verwenden	Es wird dieselbe Codierung verwendet wie für die Arbeitscodierung.
Andere	Wählen Sie die Eingabecodierung aus der Dropdown-Liste aus.

Einstellung	Beschreibung
XML-Codierung von Sonderzeichen	Legt die Darstellung von XML-Sonderzeichen fest. Sie können **Keine** oder **XML** auswählen. - **Keine.** Als & < > " ' lassen Entitätsreferenzen für XML-Sonderzeichen werden als Text interpretiert, beispielsweise erscheint das Zeichen > als `>` Standardeinstellung ist "Keine". - **XML**. In & < > " ' umwandeln Entitätsreferenzen für XML-Sonderzeichen werden als normale Zeichen interpretiert. Beispielsweise erscheint > als folgendes Zeichen: >
Bytereihenfolge	Beschreibt, wie Mehrbytezeichen im Eingabedokument angezeigt werden. Sie können die folgenden Optionen auswählen: - Little-Endian. Das Byte mit den am wenigsten signifikanten Bit erscheint zuerst. Standard. - Big-Endian. Das Byte mit den signifikantesten Bits erscheint zuerst. - Keine binäre Umwandlung.

Ausgabecodierung

Die Ausgabecodierung bestimmt, wie Zeichendaten im Haupt-Ausgabedokument codiert werden.

Die nachstehende Tabelle beschreibt die Codierungs-Einstellungen für den **Ausgabe**-Bereich:

Einstellung	Beschreibung
Arbeitscodierung verwenden	Die Ausgabecodierung ist mit der Arbeitscodierung identisch.
Andere	Der Benutzer wählt die Ausgabecodierung aus der Liste aus.
XML-Codierung von Sonderzeichen	Legt die Darstellung von XML-Sonderzeichen fest. Sie können **Keine** oder **XML** auswählen. - **Keine.** Als & < > " ' lassen Entitätsreferenzen für XML-Sonderzeichen werden als Text interpretiert, beispielsweise erscheint das Zeichen > als `>` Standard. - **XML**. In & < > " ' umwandeln Entitätsreferenzen für XML-Sonderzeichen werden als normale Zeichen interpretiert. Beispielsweise erscheint > als folgendes Zeichen: >
Identisch mit Eingabecodierung	Die Ausgabecodierung ist mit der Eingabecodierung identisch.
Bytereihenfolge	Beschreibt, wie Mehrbytezeichen im Eingabedokument angezeigt werden. Sie können die folgenden Optionen auswählen: - Little-Endian. Das Byte mit den am wenigsten signifikanten Bit erscheint zuerst. Standard. - Big-Endian. Das Byte mit den signifikantesten Bits erscheint zuerst. - Keine binäre Umwandlung.

Regeln und Richtlinien für die Zeichencodierung

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie Codierungen konfigurieren:

- Um die Leistung zu steigern, legen Sie die Arbeitscodierung mit der Codierung fest, die auch das Ausgabedokument aufweist.
- Legen Sie die Eingabecodierung mit der Codierung des Eingabedokuments fest.
- Legen Sie die Ausgabecodierung mit der Codierung des Ausgabedokuments fest.
- Für Sprachen mit Mehrbyte-Zeichen legen Sie die Arbeitscodierung mit UTF-8 fest. Sie können für die Eingabe- und Ausgabecodierung eine Unicode-Codierung wie UTF-8 oder eine Doppelbyte-Codepage wie Big-5 oder Shift_JIS verwenden.

Ausgabeeinstellungen

Konfigurieren Sie Ausgabesteuerungseinstellungen, um zu steuern, ob die Datenprozessorumwandlung Ereignisprotokolle erstellt und Ausgabedokumente speichert.

Sie können den Meldungstyp steuern, den die Datenprozessorumwandlung in das Entwicklungszeit-Ereignisprotokoll schreibt: Wenn Sie die geparsten Eingabedokumente mit den Ereignisprotokollen speichern, können Sie den Kontext, in dem der Fehler auftrat, in der **Ereignis**-Ansicht anzeigen.

Die folgende Tabelle beschreibt die Einstellungen im**Entwurfszeit-Ereignisse** Bereich:

Einstellung	Beschreibung
Entwurfszeitereignisse protokollieren	Legt fest, ob ein Entwicklungszeit-Ereignisprotokoll erstellt wird. Standardmäßig protokolliert die Datenprozessorumwandlung Benachrichtigungen, Warnungen und Fehlschläge im Entwicklungszeit-Ereignisprotokoll. Sie können die folgenden Typen von Ereignissen ausschließen: - Benachrichtigungen - Warnungen - Fehler
Geparste Dokumente speichern	Legt fest, wann die Datenprozessorumwandlung ein geparstes Eingabedokument speichert. Sie können die folgenden Optionen auswählen: - Immer. - Nie - Bei Ausfall Voreingestellt ist der Wert "Immer".

Die folgende Tabelle beschreibt die Einstellungen im**Laufzeitereignisse**-Bereich:

Einstellung	Beschreibung
Laufzeitereignisse protokollieren	Legt fest, ob bei der Ausführung der Umwandlung aus einer Zuordnung ein Ereignisprotokoll erstellt wird. - Nie. - Bei Ausfall Standardwert ist „Nie“.

Die folgende Tabelle beschreibt die Einstellungen im**Ausgabe**-Bereich:

Einstellung	Beschreibung
Automatische Ausgabe deaktivieren	Legt fest, ob die Datenprozessorumwandlung die Ausgabe in die Standardausgabedatei schreibt. Deaktivieren Sie die Standardausgabe in folgenden Situationen: - Sie geben die Ausgabe eines Parsers an die Eingabe einer anderen Komponente weiter, bevor die Umwandlung eine Ausgabedatei erstellt. - Sie verwenden eine **WriteValue**-Aktion, um Daten direkt vom Skript in die Ausgabe zu schreiben, anstatt die Daten durch die Ausgabeports zu schicken.
Wertkomprimierung deaktivieren	Legt fest, ob die Datenprozessorumwandlung zur Optimierung der Speichernutzung Wertkomprimierung verwendet. Wichtig: Deaktivieren Sie Wertkomprimierung nur auf Anraten des globalen Kundensupports von Informatica.

Verarbeitungseinstellungen

Die Verarbeitungseinstellungen definieren, wie die Datenprozessorumwandlung ein Element ohne einen definierten Datentyp verarbeitet. Die Einstellungen betreffen Skripte. Die Einstellungen betreffen keine Elemente, die von einem XMap-Objekt verarbeitet werden.

Die folgende Tabelle beschreibt die Verarbeitungseinstellungen, die die XML-Verarbeitung in Skripten beeinflussen:

Einstellung	Beschreibung
Als xs:string behandeln	Die Datenprozessorumwandlung behandelt ein Element ohne Typ als Zeichenfolge. Im Dialogfeld **XPath auswählen** wird das Element oder Attribut als einzelner Knoten angezeigt.
Als xs:anyType behandeln	Die Datenprozessorumwandlung behandelt ein Element ohne Typ als anyType. Im Dialogfeld **XPath auswählen** erscheint das Element oder Attribut als Knotenbaumstruktur. Ein Knoten ist der Typ xs:string und alle benannten komplexen Datentypen erscheinen als Baumknoten.

In der folgenden Tabelle wird eine Verarbeitungseinstellung beschrieben, die sich auf die Streamer-Verarbeitung auswirkt:

Einstellung	Beschreibung
Datenblockgröße	Diese Einstellung definiert die Menge der Daten, die der Streamer jedes Mal aus einem Eingabedateistrom liest. Die Datenprozessorumwandlung wendet diese Einstellung auf einen Streamer mit einer Dateieingabe an.

In der folgenden Tabelle wird eine Verarbeitungseinstellung beschrieben, die sich auf die hierarchische und die relationale Umwandlungsverarbeitung auswirkt:

Einstellung	Beschreibung
XML-Eingabe normalisieren	Mit dieser Einstellung wird festgelegt, ob die Datenprozessorumwandlung die XML-Eingabe normalisiert. Standardmäßig führt die Umwandlung Normalisierung für XML-Eingabe durch. In bestimmten Fällen möchten Sie die automatische Normalisierung vielleicht überspringen, um die Leistung zu steigern.

XMap-Einstellungen

Die XMap-Einstellung definiert, wie die Datenprozessorumwandlung XMap-Eingabeelemente verarbeitet, die nicht in Ausgabeelemente umgewandelt werden. Die ungelesenen Elemente werden an einen dedizierten Port namens **XMap_Unread_Input_Values** übergeben. Die Einstellung hat nur Auswirkungen, wenn die XMap als Startkomponente ausgewählt ist. Die Einstellung hat keine Auswirkungen auf Elemente, die die XMap verarbeitet.

Um ungelesene XMap-Elemente an einen dedizierten Port zu übergeben, aktivieren Sie die Einstellung **Ungelesene Elemente in einen zusätzlichen Ausgabeport schreiben**.

Konfigurieren der XML-Ausgabe

Die Einstellungen der XML-Erzeugung definieren die Merkmale von XML-Ausgabedokumenten.

In der folgenden Tabelle werden die Einstellungen der XML-Erzeugung im Bereich **Schematitel** beschrieben:

Einstellung	Beschreibung
Speicherort des Schemas	Definiert die schemaLocation für das Stammelement des Hauptausgabedokuments.
Speicherort des Schemas ohne Namensraum	Definiert das Attribut „xsi:noNamespaceSchemaLocation“ für das Stammelement des Hauptausgabedokuments.

Konfigurieren Sie die Einstellungen des XML-Ausgabemodus, um festzulegen, wie sich die Datenprozessorumwandlung bei fehlenden Elementen oder Attributen im XML-Eingabedokument verhalten soll. In der folgenden Tabelle werden die Einstellungen der XML-Erzeugung im Bereich **XML-Ausgabemodus** beschrieben:

Einstellung	Beschreibung
Unverändert	Fügen Sie keine leeren Elemente hinzu bzw. entfernen Sie sie nicht. Standardwert ist „Aktiviert“.
Komplett	Alle erforderlichen und optionalen Elemente, die im Ausgabeschema definiert sind, werden in die Ausgabe geschrieben. Elemente ohne Inhalt werden als leere Elemente geschrieben.
Kompakt	Entfernt leere Elemente aus der Ausgabe. Wenn **Für Elemente hinzufügen** aktiviert ist, entfernt die Datenprozessorumwandlung nur die optionalen Elemente. Wenn **Für Elemente hinzufügen** deaktiviert ist, entfernt die Datenprozessorumwandlung alle leeren Elemente. Möglicherweise ist die XML-Ausgabe nicht gültig.

In der folgenden Tabelle werden die Einstellungen der XML-Erzeugung im Bereich **Standardwerte für benötigte Knoten** beschrieben:

Einstellung	Beschreibung
Für Elemente hinzufügen	Wenn das Ausgabeschema einen Standardwert für ein erforderliches Element definiert, wird das Element mit einem Standardwert in die Ausgabe aufgenommen. Standardwert ist „Aktiviert“.
Für Attribute hinzufügen	Wenn das Ausgabeschema einen Standardwert für ein erforderliches Attribut definiert, wird das Attribut mit dem zugehörigen Standardwert in die Ausgabe aufgenommen. Standardwert ist „Aktiviert“.
Hinzugefügte Werte validieren	Legt fest, ob die Datenprozessorumwandlung leere Elemente validiert, die von der Ausgabe im kompletten Modus hinzugefügt werden. Standardwert ist „Deaktiviert“. Wenn **Hinzugefügte Werte validieren** aktiviert ist und das Schema keine leeren Elemente zulässt, ist die XML-Ausgabe möglicherweise ungültig.

In der folgenden Tabelle werden die Einstellungen der XML-Erzeugung im Bereich **Verarbeitungsanweisungen** beschrieben:

Einstellung	Beschreibung
XML-Verarbeitungsanweisungen hinzufügen	Definiert die Zeichencodierung und die XML-Version des Ausgabedokuments. Standardwert ist "selected".
XML-Version	Definiert die XML-Version. Die Einstellung der XML-Version weist die folgenden Optionen auf: - 1.0 - 1.1 Standard ist 1.0.
Codierung	Definiert die Zeichencodierung, die in der Verarbeitungsanweisung angegeben ist. Die Einstellung „Codierung“ weist die folgenden Optionen auf: - Wie Ausgabecodierung. Die Ausgabecodierung in der Verarbeitungsanweisung ist mit der Ausgabecodierung identisch, die in den Einstellungen der Datenprozessorumwandlung definiert ist. - Benutzerdefiniert. Definiert die Ausgabecodierung in der Verarbeitungsanweisung. Der Benutzer gibt den Wert in dem Feld ein.
Benutzerdefinierte Verarbeitungsanweisungen hinzufügen	Fügt dem Ausgabedokument weitere Verarbeitungsanweisungen hinzu. Geben Sie die Verarbeitungsanweisung genau so ein, wie sie im Ausgabedokument angezeigt wird. Standardwert ist „Deaktiviert“.

In der folgenden Tabelle werden die Einstellungen der XML-Erzeugung im Bereich **XML-Root** beschrieben:

Einstellung	Beschreibung
XML-Stammelement hinzufügen	Fügt dem Ausgabedokument ein Stammelement hinzu. Verwenden Sie diese Option, wenn das Ausgabedokument mehrere Instanzen des Stammelements enthält, das im Ausgabeschema definiert ist. Standardwert ist „Deaktiviert“.
Name des Stammelements	Definiert einen Namen für das Stammelement, das dem Ausgabedokument hinzugefügt wird.

Ereignisse

Ein Ereignis ist ein Datensatz eines Verarbeitungsschritts einer Komponente in einer Datenprozessorumwandlung. In einem Skript oder einer Bibliothek generiert jeder Anker, jede Aktion oder jeder Transformer ein Ereignis. In einem XMap-Objekt generiert jede Mapping-Anweisung ein Ereignis.

Sie können Ereignisse in der Ansicht **Datenprozessor-Ereignisse** anzeigen.

Ereignistypen

Die Datenprozessorumwandlung schreibt Ereignisse in Protokolldateien. Jedes Ereignis hat einen Ereignistyp, der anzeigt, ob das Ereignis erfolgreich war, fehlgeschlagen ist oder mit Fehlern ausgeführt wurde.

Eine Komponente kann ein oder mehrere Ereignisse generieren. Die Komponente kann übergeben werden oder fehlschlagen, je nachdem, ob die Ereignisse erfolgreich sind oder fehlschlagen. Schlägt ein Ereignis fehl, so schlägt eine Komponente fehl.

Die folgende Tabelle beschreibt die Ereignistypen, die die Datenprozessorumwandlung generiert:

Ereignistyp	Beschreibung
Benachrichtigung	Normaler Vorgang.
Warnung	Die Datenprozessorumwandlung wurde ausgeführt, aber es trat eine unerwartete Bedingung auf. Beispiel: Die Datenprozessorumwandlung schrieb Daten mehrmals in dasselbe Element. Die Datenprozessorumwandlung generiert jedes Mal, wenn das Element überschrieben wird, eine Warnung.
Fehler	Die Datenprozessorumwandlung wurde ausgeführt, aber eine Komponente schlug fehl. Beispiel: Ein erforderliches Eingabeelement war leer.
Optionaler Fehler	Die Datenprozessorumwandlung wurde ausgeführt, aber eine optionale Komponente schlug fehl. Beispiel: Ein optionaler Anker fehlte im Quelldokument.
Schwerwiegender Fehler	Die Datenprozessorumwandlung schlug aufgrund eines schwerwiegenden Fehlers fehl. Beispiel: Das Eingabedokument war nicht vorhanden.

Ansicht der Datenprozessor-Ereignisse

Die **Datenprozessor-Ereignisse**-Ansicht zeigt Ereignisse an, wenn Sie eine Datenprozessorumwandlung über das Developer Tool ausführen.

Die **Datenprozessor-Ereignisse**-Ansicht hat einen **Navigation**sbereich und einen**Detail**bereich. Der Navigationsbereich enthält einen Navigationsbaum. Im Navigationsbaum sind die Komponenten aufgelistet, die die Umwandlung in chronologischer Reihenfolge ausführte. Jeder Knoten hat ein Symbol unten im Baum, das das schwerwiegendste Ereignis darstellt. Wenn Sie einen Knoten im **Navigation**sbereich auswählen, werden im **Detail**bereich Ereignisse angezeigt.

Der Navigationsbaum enthält die folgenden Knoten auf der höchsten Ebene:

- Dienstinitialisierung. Beschreibt die Dateien und Variablen, die die Datenprozessorumwandlung initialisiert.
- Ausführung. Listet die Komponenten auf, die durch das Skript, die Bibliothek oder die XMap ausgeführt wurden.
- Zusammenfassung. Zeigt Statistiken zu Verarbeitung an.

Wenn Sie XMap ausführen, hat jeder Knotenname im Navigationsbereich eine Zahl in eckigen Klammern, z. B. [5]. Um die Anweisung zu identifizieren, die die Ereignisse für den Knoten generierte, klicken Sie mit der rechten Maustaste in das Anweisungen-Gitter und wählen Sie "Zu Zeilennummer wechseln". Geben Sie die Knotennummer ein.

Wenn Sie ein Skript ausführen und auf ein Ereignis im **Navigation**s- oder **Detail**bereich doppelklicken, hebt der Skript-Editor die Skriptkomponente hervor, die das Ereignis generiert hat. Der **Eingabe**-Bereich der **Daten-Viewer**-Ansicht hebt den Teil des Beispiel-Quelldokuments hervor, der das Ereignis generierte.

Protokolle

Ein Protokoll enthält einen Datensatz der Datenprozessorumwandlung. Die Datenprozessorumwandlung schreibt Ereignisse in Protokolle.

Die Datenprozessorumwandlung erstellt die folgenden Protokolltypen

Entwicklungszeit-Ereignisprotokoll

Das Entwicklungszeit-Ereignisprotokoll enthält Ereignisse, die eintreten, wenn Sie die Datenprozessorumwandlung in der Ansicht **Daten-Viewer** ausführen. Zeigen Sie das Entwicklungszeit-Protokoll in der Ansicht **Ereignisse** an.

Laufzeit-Ereignisprotokoll

Das Laufzeit-Ereignisprotokoll enthält Ereignisse, die eintreten, wenn Sie die Datenprozessorumwandlung in einem Mapping ausführen. Sie können das Laufzeit-Ereignisprotokoll in einem Texteditor anzeigen oder ein Laufzeit-Ereignisprotokoll in die Ansicht **Ereignisse** der Datenprozessorumwandlung ziehen.

Benutzerprotokoll

Das Benutzerprotokoll enthält Ereignisse, die Sie für Komponenten in einem Skript konfigurieren. Die Datenprozessorumwandlung schreibt in das Benutzerprotokoll, wenn Sie sie aus der Ansicht **Daten-Viewer** in einem Mapping ausführen. Sie können das Benutzerprotokoll in einem Texteditor ansehen.

Entwicklungszeit-Ereignisprotokoll

Das Entwicklungszeit-Ereignisprotokoll enthält die Ereignisse, die bei der Ausführung der Datenprozessorumwandlung über den **Daten-Viewer** im Developer Tool auftreten.

Bei der Ausführung einer Datenprozessorumwandlung über die **Daten-Viewer**-Ansicht wird das Entwicklungszeit-Ereignisprotokoll in der **Datenprozessor-Ereignisse**-Ansicht angezeigt. Standardmäßig enthält das Entwicklungszeit-Ereignisprotokoll Benachrichtigungen, Warnungen und Fehlschläge. In den Umwandlungseinstellungen können Sie die Datenprozessorumwandlung so konfigurieren, dass ein oder mehrere Ereignistypen aus dem Protokoll ausgeschlossen sind.

Wenn Sie die Eingabedokumente mit den Protokollen speichern, können Sie ein Ereignis in der **Datenprozessor**-Ansicht anklicken, um den Ort im Eingabedokument zu finden, der das Ereignis generierte. Wenn Sie die Einstellungen der Datenprozessorumwandlung konfigurieren, können Sie die Eingabedateien entweder bei jeder Ausführung oder nur bei einem Fehlschlag speichern lassen.

Das Entwicklungszeit-Ereignisprotokoll heißt `events.cme`. Sie finden das Entwicklungszeit-Ereignisprotokoll für die letzte Ausführung der Datenprozessorumwandlung im folgenden Verzeichnis:

```
C:\<Installation_directory>\clients\DT\CMReports\Init\events.cme
```

Die Datenprozessorumwandlung überschreibt das Entwicklungszeit-Ereignisprotokoll jedes Mal, wenn Sie die Umwandlung im**Daten-Viewer** ausführen. Wenn Sie das Entwicklungszeit-Ereignisprotokoll nach einem späteren Ausführen der Umwandlung anzeigen oder die Protokolle verschiedener Ausführungen vergleichen möchten, müssen Sie es umbenennen. Beim Schließen des Developer Tools speichert der Developer keine Dateien im

Laufzeit-Ereignisprotokoll

Das Laufzeit-Ereignisprotokoll zeichnet Ereignisse auf, die eintreten, wenn Sie die Datenprozessorumwandlung in einem Mapping ausführen.

Wenn die Datenprozessorumwandlung den Durchgang ohne Fehler abschließt, wird kein Ereignisprotokoll geschrieben. Wenn Fehler vorkommen, wird die Datenprozessorumwandlung ein zweites Mal durchgeführt. Während des zweiten Durchgangs wird ein Ereignisprotokoll geschrieben. Das Laufzeit-Ereignisprotokoll hat die Bezeichnung `events.cme`.

Auf einem Windows-Rechner befindet sich das Laufzeit-Ereignisprotokoll im folgenden Verzeichnis:

```
C:<Installation_Directory>\clients\DT\CMReports\Tmp\
```

Auf einem Linux- oder UNIX-Rechner befindet sich das Laufzeit-Ereignisprotokoll für einen Root-Benutzer im folgenden Verzeichnis:

```
/root/<Installation_Dirctory>/clients/DT/CMReports/Tmp
```

Auf einem Linux- oder UNIX-Rechner befindet sich das Laufzeit-Ereignisprotokoll für einen Nicht-Root-Benutzer im folgenden Verzeichnis:

```
/home/[UserName]/<Installation_Directory>/DT/CMReports/Tmp
```

Sie können den Standort des Laufzeit-Ereignisprotokolls mit dem Konfigurationseditor ändern.

Anzeigen eines Ereignisprotokolls in der Ansicht für Datenprozessor-Ereignisse

Zeigen Sie mithilfe der Ansicht **Datenprozessor-Ereignisse** Ereignisprotokolle zur Entwicklungszeit oder zur Laufzeit an.

Öffnen Sie Windows Explorer und suchen Sie nach der Ereignisprotokolldatei, die Sie anzeigen möchten. Ziehen Sie das Protokoll aus dem Windows Explorer-Fenster in die Ansicht **Datenprozessor-Ereignisse**. Klicken Sie mit der rechten Maustaste auf die Ansicht **Datenprozessor-Ereignisse** und wählen Sie anschließend **Suchen** aus, um das Protokoll zu suchen.

Hinweis: Um das aktuellste Ereignisprotokoll zur Entwicklungszeit neu zu laden, klicken Sie mit der rechten Maustaste auf die Ansicht **Datenprozessor-Ereignisse** und wählen Sie anschließend **Projektereignisse neu laden** aus.

Benutzerprotokoll

Das Benutzerprotokoll enthält benutzerdefinierte Meldungen, die Sie zu Fehlern von Komponenten in einem Skript konfigurieren.

Die Datenprozessorumwandlung schreibt Meldungen in das Benutzerprotokoll, wenn Sie ein Skript von der Ansicht **Daten-Viewer** aus ausführen und wenn Sie es in einem Mapping ausführen.

Wenn eine Skriptkomponente die Eigenschaft **on_fail** aufweist, können Sie es so konfigurieren, dass es eine Meldung in das Benutzerprotokoll schreibt, wenn es fehlschlägt. Legen Sie im Skript die Eigenschaft **on_fail** auf einen der folgenden Werte fest:

- LogInfo
- LogWarning
- LogError

Bei jeder Ausführung des Skripts wird ein neues Benutzerprotokoll erstellt. Der Dateiname des Benutzerprotokolls enthält den Umwandlungsnamen mit einer eindeutigen GUID.

```
<Transformation_Name>_<GUID>.log
```

Beispiel: `CalculateValue_Aa93a9d14-a01f-442a-b9cb-c9ba5541b538.log`

Auf einem Windows-Computer befindet sich das Benutzerprotokoll im folgenden Verzeichnis:

```
c:\Users\[UserName]\AppData\Roaming\Informatica\DataTransformation\UserLogs
```

Auf einem Linux- oder UNIX-Computer befindet sich das Benutzerprotokoll für den Root-Benutzer im folgenden Verzeichnis:

```
/<Installation_Directory>/DataTransformation/UserLogs
```

Auf einem Linux- oder UNIX-Computer befindet sich das Benutzerprotokoll für einen Benutzer, der kein Root-Benutzer ist, im folgenden Verzeichnis:

```
home/<Installation_Dirctory>/DataTransformation/UserLogs
```

Entwicklung der Datenprozessorumwandlung

Verwenden Sie den Assistenten für neue Umwandlungen, um eine Datenprozessorumwandlung automatisch zu generieren, oder erstellen Sie eine leere Datenprozessorumwandlung und konfigurieren Sie sie später. Wenn Sie eine leere Datenprozessorumwandlung erstellen, müssen Sie ein Skript-, XMap-, Bibliotheks- oder Validierungsregelobjekt in der Umwandlung erstellen. Ein Skript kann Quelldokumente in ein hierarchisches Format parsen, das hierarchische Format in andere Dateiformate konvertieren oder ein hierarchisches Dokument einem anderen hierarchischen Format zuordnen. Eine XMap wandelt eine hierarchische Eingabedatei in eine hierarchische Ausgabedatei mit einer anderen Struktur um. Eine Bibliothek konvertiert einen branchenüblichen Nachrichtentyp in ein XML-Dokument mit einer Hierarchiestruktur oder aus XML in ein branchenübliches Standardformat. Wählen Sie die Schemata aus, die die Eingabe- oder Ausgabehierarchien definieren.

1. Erstellen Sie die Umwandlung im Developer Tool.
2. Führen Sie für eine leere Datenprozessorumwandlung die folgenden zusätzlichen Schritte aus:
   a. Fügen Sie Schemareferenzen hinzu, die die Eingabe- oder Ausgabe-XML-Hierarchien definieren.
   b. Erstellen Sie ein Skript-, XMap-, Bibliotheks- oder Validierungsregelobjekt.
3. Konfigurieren Sie die Eingabe- und Ausgabeports.
4. Testen Sie die Umwandlung.

Erstellen der Datenprozessorumwandlung

Erstellen Sie eine Datenprozessorumwandlung im Developer Tool. Wenn Sie eine leere Datenprozessorumwandlung erstellen, müssen Sie anschließend ein Skript-, XMap-, Bibliotheks- oder

Validierungsregelobjekt in der Umwandlung erstellen. Alternativ können Sie den Assistenten für neue Umwandlungen verwenden, um eine Datenprozessorumwandlung automatisch zu erzeugen.

1. Klicken Sie im Developer-Tool auf **Datei** > **Neu** > **Umwandlung**.
2. Wählen Sie die Datenprozessorumwandlung aus und klicken Sie auf **Weiter**.
3. Geben Sie einen Namen für die Umwandlung ein und suchen Sie nach einem Modellrepository-Speicherort, um die Umwandlung abzulegen.
4. Geben Sie an, ob die Datenprozessorumwandlung mit einem Assistenten oder ob eine leere Datenprozessorumwandlung erstellt werden soll.
5. Wenn Sie das Erstellen einer leeren Datenprozessorumwandlung ausgewählt haben, klicken Sie auf **Fertig stellen**.

   Das Developer-Tool erstellt die leere Umwandlung im Repository. Im Developer-Tool wird die Ansicht **Übersicht** angezeigt.
6. Wenn Sie das Erstellen einer Datenprozessorumwandlung mit einem Assistenten ausgewählt haben, führen Sie die folgenden Schritte aus:
   a. Klicken Sie auf **Weiter**.
   b. Wählen Sie ein Eingabeformat aus.
   c. Suchen und wählen Sie ein Schema, ein Copybook, eine Beispiel- oder Spezifikationsdatei aus, falls dies für bestimmte Eingabeformate wie COBOL, JSON oder ASN.1 erforderlich ist.
   d. Wählen Sie ein Ausgabeformat aus.
   e. Suchen und wählen Sie ggf. ein Schema, ein Copybook, eine Beispiel- oder Spezifikationsdatei für das Ausgabeformat aus.
   f. Klicken Sie auf **Fertig stellen**. Der Assistent erstellt die Umwandlung im Repository.

   Die Umwandlung enthält möglicherweise einen Parser, Serializer, Mapper oder ein Objekt mit gemeinsamen Komponenten. Wenn Sie ein Schema, ein Copybook, eine Beispiel- oder Spezifikationsdatei ausgewählt haben, erstellt der Assistent ebenfalls ein Schema im Repository, das der Hierarchie in der Datei entspricht.

Auswählen der Schemaobjekte

Wählen Sie die Schemaobjekte aus, die die Eingabe- oder Ausgabehierarchien für die jeweilige XMap oder Skriptkomponente definieren, deren Erstellung Sie beabsichtigen.

In der Ansicht „Referenzen“ können Sie Schemareferenzen hinzufügen. Alternativ können Sie die Schemareferenzen hinzufügen, wenn Sie Skript- oder XMap-Objekte erstellen. Bevor Sie ein Schemaobjekt in einem Skript oder einer XMap referenzieren können, muss es im Modellrepository vorhanden sein

1. Klicken Sie in der Ansicht **Referenzen** der Datenprozessorumwandlung auf **Hinzufügen.**
2. Wenn das Schemaobjekt im Modellrepository vorhanden ist, suchen Sie das Schema und wählen Sie es aus.
3. Wenn das Schema im Modellrepository nicht vorhanden ist, klicken Sie auf **Neues Schemaobjekt erstellen** und importieren Sie ein Schemaobjekt aus einer hierarchischen Schemadatei.
4. Klicken Sie auf "Fertigstellen", um der Datenprozessorumwandlung die Schemareferenz hinzuzufügen.

Erstellen von Objekten in einer leeren Datenprozessorumwandlung

Erstellen Sie ein Skript, Bibliotheks-, XMap- oder Validierungsregelobjekt in der Ansicht **Objekte** der Umwandlung. Nach der Erstellung des Objekts können Sie das Objekt von der Ansicht **Objekte** aus öffnen, um es zu konfigurieren.

Erstellen eines Skripts

Erstellen Sie ein Skriptobjekt und definieren Sie den zu erstellenden Skript-Komponententyp. Optional können Sie eine Schemareferenz und eine Beispielquelldatei definieren.

1. Klicken Sie in der Ansicht **Objekte** der Datenprozessorumwandlung auf **Neu**.
2. Geben Sie einen Namen für das Skript ein und klicken Sie auf **Weiter.**
3. Wählen Sie die Erstellung eines Parsers oder Serializers aus. Wählen Sie „Andere“, um eine Mapper-, Transformer- oder Streamer-Komponente zu erstellen.
4. Geben Sie einen Namen für die Komponente ein.
5. Handelt es sich bei der Komponente um die erste Komponente, die Daten in der Umwandlung verarbeitet, aktivieren Sie **Als Startkomponente festlegen**.
6. Klicken Sie auf **Weiter**, wenn Sie eine Schemareferenz für dieses Skript eingeben möchten. Klicken Sie auf **Beenden**, wenn Sie die Schemareferenz nicht eingeben möchten.
7. Wenn Sie eine Schemareferenz erstellen, wählen Sie **Referenz einem Schemaobjekt hinzufügen** aus und suchen Sie nach dem Schemaobjekt im Modellrepository. Klicken Sie auf **Neues Schemaobjekt erstellen**, um ein Schemaobjekt im Modellrepository zu erstellen.
8. Klicken Sie auf **Weiter**, um eine Beispielquellenreferenz oder einen Beispieltext einzugeben. Klicken Sie auf **Beenden**, wenn Sie keine Beispielquelle definieren möchten.

   Verwenden Sie eine Beispielquelle, um Beispieldaten zu definieren und das Skript zu testen.
9. Wenn Sie eine Beispielquelle auswählen, wählen Sie **Datei** aus und suchen Sie nach der Beispieldatei.

   Sie können auch Beispieltext im **Text**-Bereich eingeben. Das Developer-Tool verwendet den Text zum Testen eines Skripts.
10. Klicken Sie auf **Fertig stellen.**

    Die **Skript**-Ansicht wird im Developer Tool-Editor angezeigt.

Erstellen einer XMap

Erstellen Sie eine XMap in der Data Transformation-Ansicht **Objekt**. Beim Erstellen einer XMap benötigen Sie ein Schema, das die Dokumente für das Eingabe- und Ausgabehierarchieschema beschreibt. Sie wählen das Element in dem Schema aus, bei dem es sich um das Root-Element für die Eingabehierarchie handelt.

1. Klicken Sie in der Ansicht **Objekte** der Datenprozessorumwandlung auf **Neu.**
2. Wählen Sie "XMap" und klicken Sie auf **Weiter.**
3. Geben Sie den Namen für die XMap ein.
4. Handelt es sich bei der XMap-Komponente um die erste Komponente, die Daten in der Umwandlung verarbeitet, aktivieren Sie **Als Startkomponente festlegen**.

   Klicken Sie auf **Weiter.**
5. Wenn Sie eine Schemareferenz erstellen, wählen Sie **Referenz einem Schemaobjekt hinzufügen** aus und suchen Sie nach dem Schemaobjekt im Modellrepository.

   Um ein neues Schemaobjekt zu importieren, klicken Sie auf **Neues Schemaobjekt erstellen**.
6. Wenn Sie über eine Beispielhierarchiedatei verfügen, mit der Sie die XMap testen können, navigieren Sie zu dieser Datei und wählen Sie sie im Dateisystem aus.

   Sie können die Beispielhierarchiedatei ändern.
7. Wählen Sie die Root als Eingabe-Hierarchie.

Wählen Sie im Dialogfeld **Auswahl des Root-Elements** ein Element im Schema aus, bei dem es sich um das Root-Element für die Eingabehierarchiedatei handelt. Sie können im Schema nach einem Element suchen. Sie können Muster suchen. Für die Übereinstimmung einer beliebigen Zeichenanzahl in der Zeichenfolge geben Sie `*<string>` ein. Für die Übereinstimmung eines einzelnen Zeichens geben Sie `?<character>` ein.

8. Klicken Sie auf **Fertig stellen**.

   Das Developer-Tool erstellt eine Ansicht für jede XMap, die Sie erstellen. Klicken Sie auf die Ansicht, um das Mapping zu konfigurieren.

Erstellen einer Bibliothek

Erstellen Sie ein Bibliotheksobjekt in der Data Transformation-Ansicht **Objekte**. Wählen Sie den Meldungstyp, die Komponente und den Namen aus. Optional können Sie eine Beispielmeldungstyp-Quelldatei definieren, die Sie zum Testen des Bibliotheksobjekts verwenden können.

Bevor Sie eine Bibliothek in der Datenprozessorumwandlung erstellen, installieren Sie das Bibliothekssoftwarepaket auf Ihrem Computer.

1. Klicken Sie in der Ansicht **Objekte** der Datenprozessorumwandlung auf **Neu.**
2. Wählen Sie "Bibliothek" aus und klicken Sie auf **Weiter.**
3. Suchen Sie den Meldungstyp und wählen Sie ihn aus.
4. Wählen Sie die Erstellung eines Parsers oder Serializers aus.

   Erstellen Sie einen Parser, wenn die Eingabe des Bibliotheksobjekts ein Meldungstyp und die Ausgabe XML ist. Erstellen Sie einen Serializer, wenn die Eingabe des Bibliotheksobjekts XML und die Ausgabe ein Meldungstyp ist.
5. Handelt es sich bei der Bibliothek um die erste Komponente, die Daten in der Datenprozessorumwandlung verarbeitet, aktivieren Sie **Als Startkomponente festlegen**.

   Klicken Sie auf **Weiter.**
6. Wenn Sie über eine Beispielmeldungstyp-Quelldatei verfügen, die Sie zum Testen der Bibliothek verwenden können, suchen Sie die Datei im System und wählen Sie sie aus.

   Die können die Beispieldatei ändern.
7. Klicken Sie auf **Fertig stellen**.

   Das Developer-Tool erstellt eine Ansicht für jeden von Ihnen erstellten Meldungstyp. Klicken Sie auf die Ansicht, um auf das Mapping zuzugreifen.

Erstellen von Validierungsregeln

Erstellen Sie ein Validierungsregelobjekt in der Ansicht **Objekte** der Datenprozessorumwandlung.

1. Klicken Sie in der Ansicht **Objekte** der Datenprozessorumwandlung auf **Neu.**
2. Wählen Sie „Validierungsregeln“ aus und klicken Sie auf **Weiter.**
3. Geben Sie einen Namen für die Validierungsregeln ein.
4. Wenn Sie über eine XML-Beispieldatei verfügen, mit der Sie die Validierungsregeln testen können, navigieren Sie zu dieser Datei und wählen Sie sie im Dateisystem aus.

   Die können die Beispiel-XML-Datei wechseln.
5. Klicken Sie auf **Fertig stellen**.

   Das Developer-Tool erstellt ein Validierungsregelobjekt und öffnet es im Validierungsregeleditor.

Hinzufügen einer Beispielquelle

Wählen Sie die Beispielquelle aus, um das Skript, die XMap, die Bibliothek oder die Validierungsregeln zu testen, die Sie erstellen möchten.

Sie können eine Beispielquelle hinzufügen, wenn Sie ein Skript, eine XMap, eine Bibliothek oder Validierungsregeln erstellen. Nach der Auswahl wird die Beispielquelle zum Modellrepository hinzugefügt. Aufgrund von Modellrepository-Beschränkungen ist die Größe der Beispielquelldatei auf 5 MB begrenzt.

Sie können die Beispielquelle ändern.

Erstellen von Ports

Konfigurieren Sie in der Ansicht **Übersicht** die Eingabe- und die Ausgabeports.

Wenn Sie weitere Eingabe- oder Ausgabeports in einem Skript konfigurieren, fügt das Developer-Tool der Umwandlung standardmäßig weitere Eingabe- und Ausgabeports hinzu. Es werden keine Eingabeports in der Ansicht **Übersicht** hinzugefügt.

1. Wenn Sie die Ausgabedaten anstelle einer XML in Zeilen zurückgeben möchten, aktivieren Sie die Option **Relationale Ausgabe**.

   Bei Aktivierung der relationalen Ausgabe entfernt das Developer-Tool den standardmäßigen Ausgabeport.
2. Wählen Sie den Datentyp des Eingabeports, den Porttyp, die Gesamtstellenanzahl und die Größenordnung aus.
3. Wenn Sie keine relationalen Ausgabeports definieren, definieren Sie den Datentyp des Ausgabeports, den Porttyp, die Gesamtstellenanzahl und die Größenordnung.
4. Wenn ein Skript weitere Eingabeports aufweist, können Sie den Speicherort der Beispieleingabedatei für die Ports definieren. Klicken Sie im Feld **Eingabespeicherort** auf die Schaltfläche **Öffnen**, um nach der Datei zu suchen.
5. Wenn die relationale Ausgabe aktiviert ist, klicken Sie auf **Mapping-Ausgabe**, um die Ausgabeports zu erstellen.
6. Ordnen Sie in der Ansicht „Ports“ den Feldern im Bereich **Relationale Ports** Knoten aus dem Bereich **Hierarchische Ausgabe** zu.

Testen der Umwandlung

Testen Sie in der Ansicht **Daten-Viewer** die Datenprozessorumwandlung.

Überprüfen Sie vor dem Testen der Umwandlung, ob die Startkomponente definiert wurde. Sie können die Startkomponente in einem Skript definieren oder Sie können die Startkomponente auf der Registerkarte **Übersicht** auswählen. Es muss außerdem eine Beispieleingabedatei zum Testen ausgewählt sein.

1. Öffnen Sie die Ansicht **Daten-Viewer**.
2. Klicken Sie auf **Ausführen**

   Das Developer Tool validiert die Umwandlung. Wenn kein Fehler vorliegt, zeigt das Developer Tool die Beispieldatei im Bereich **Eingabe** an. Die Ausgabeergebnisse werden in der Maske „Ausgabe“ angezeigt.
3. Klicken Sie auf **Ereignisse anzeigen**, um die Ansicht **Datenprozessor-Ereignisse** anzuzeigen.
4. Doppelklicken Sie in der Ansicht **Datenprozessor-Ereignisse** auf ein Ereignis, um das Ereignis im Skripteditor zu debuggen.

5. Klicken Sie auf **Mit Editor synchronisieren**, um die Eingabedatei zu ändern, falls Sie mehrere Komponenten mit jeweils einer anderen Beispieleingabedatei testen.

   Wenn der Inhalt der Beispieldatei im Dateisystem geändert wird, werden die Änderungen im Bereich **Eingabe** angezeigt.

Datenprozessorumwandlung - Importieren und Exportieren

Sie können eine Datenprozessorumwandlung als Dienst exportieren und aus einem Data Transformation-Repository ausführen. Sie können auch einen Data Transformation-Dienst in das Developer-Tool importieren. Wenn Sie einen Data Transformation-Dienst importieren, erstellt das Developer-Tool eine Datenprozessorumwandlung aus dem Dienst.

Hinweis: Wenn Sie einen Data Transformation-Dienst in das Modellrepository importieren, importiert das Developer-Tool die zugehörigen Schemata in das Repository. Wenn Sie das Schema im Repository ändern, erscheinen die Änderungen nicht sofort in den Umwandlung-Schemareferenzen. Sie können die Verbindung zum Modellrepository schließen und öffnen oder das Developer Tool schließen und öffnen, damit die Schemaänderungen in der Umwandlung erscheinen.

Exportieren der Datenprozessorumwandlung als Dienst

Sie können die Datenprozessorumwandlung als Data Transformation-Dienst exportieren. Exportieren Sie den Dienst in das Dateisystem-Repository der Maschine, in der sie den Dienst ausführen möchten. Sie können den Dienst mit PowerCenter, benutzerdefinierten Anwendungen oder dem Befehl Data Transformation CM_console ausführen.

1. Klicken Sie in der Ansicht **Objekt-Explorer** mit der rechten Maustaste auf die zu exportierende Datenprozessorumwandlung, und wählen Sie **Exportieren** aus.

   Das Dialogfeld **Export** wird eingeblendet
2. Wählen Sie **Informatica** > **Datenprozessorumwandlung exportieren**aus und klicken Sie auf **Weiter**.

   Die Seite **Auswählen** erscheint.
3. Klicken Sie auf **Weiter**.

   Die Seite **Dienstnamen und Zielordner auswählen** erscheint.
4. Wählen Sie einen Zielordner aus:
   - Um den Dienst auf der Maschine zu exportieren, auf dem das Developer-Tool gehostet wird, klicken Sie auf **Dienstordner**.
   - Um den Dienst auf einer anderen Maschine bereitzustellen, klicken Sie auf **Ordner**. Navigieren Sie zum Verzeichnis `\ServiceDB` auf der Maschine, auf der Sie den Dienst bereitstellen möchten.
5. Klicken Sie auf **Fertig stellen**.

Importieren eines Data Transformation-Diensts

Sie können einen Data Transformation-Dienst aus dem Dateisystem-Repository des Computers importieren, auf dem Sie den Dienst gespeichert haben. Importieren Sie eine CMW-Datei des Data Transformation-Diensts

in das Modellrepository, um eine Datenprozessorumwandlung zu erstellen. Das Developer-Tool importiert Umwandlungs-, Schema- und Beispieldaten mit der .cmw-Datei.

1. Klicken Sie auf **Datei** > **Importieren**.

   Das Dialogfeld **Importieren** wird eingeblendet.

2. Wählen Sie **Informatica Data-Transformation-Dienst importieren** und klicken Sie auf **Weiter**.

   Die Seite **Data Transformation-Dienst importieren** wird angezeigt.

3. Navigieren Sie zur Dienstdatei mit der Erweiterung .cmw, die Sie importieren möchten.

   Das Developer-Tool benennt die Umwandlung entsprechend des Dienstdateinamens. Sie können den Namen ändern.

4. Navigieren Sie zu einem Speicherort im Repository, wo Sie die Umwandlung speichern möchten. Dann klicken Sie auf **Fertig stellen**.

   Das Developer-Tool importiert die Umwandlungs-, Schema- und Beispieldaten mit der `.cmw`-Datei.

5. Zum Bearbeiten der Umwandlung doppelklicken Sie in der Ansicht **Objekt-Explorer** auf die Umwandlung.

Exportieren eines Mappings mit einer Datenprozessorumwandlung nach PowerCenter

Wenn Sie ein Mapping mit einer Datenprozessorumwandlung in PowerCenter exportieren, können Sie die Objekte in eine lokale Datei exportieren und anschließend das Mapping in PowerCenter importieren. Alternativ können Sie das Mapping direkt in das PowerCenter-Repository exportieren.

1. Wählen Sie in der Ansicht **Objekt-Explorer** das zu exportierende Mapping aus. Klicken Sie mit der rechten Maustaste und wählen Sie **Exportieren** aus.

   Das Dialogfeld **Export** wird eingeblendet

2. Wählen Sie **Informatica** > **PowerCenter**.
3. Klicken Sie auf **Weiter**.

   Das Dialogfeld **Export an PowerCenter** wird eingeblendet.

4. Wählen Sie das Projekt.
5. Wählen Sie die PowerCenter-Version.
6. Wählen Sie den Exportspeicherort, eine PowerCenter-Import-XML-Datei oder ein PowerCenter-Repository.
7. Geben Sie die Exportoptionen an.
8. Klicken Sie auf **Weiter**.

   Sie werden im Developer-Tool aufgefordert, die Objekte für den Export auszuwählen.

9. Wählen Sie die zu exportierenden Objekte und klicken Sie auf **Fertig stellen**.

   Das Developer-Tool exportiert die Objekte an den von Ihnen gewählten Speicherort. Wenn Sie das Mapping an einen Speicherort exportiert haben, exportiert das Developer-Tool auch die Datenprozessorumwandlungen im Mapping, wie z. B. Dienste, in einen Ordner an dem Speicherort, den Sie angegeben haben.

10. Wenn Sie das Mapping in ein PowerCenter-Repository exportiert haben, werden die Dienste in den folgenden Verzeichnispfad exportiert: `%temp%\DTServiceExport2PC\`

    Die Exportfunktion erstellt einen separaten Ordner für jeden Dienst mit dem folgenden Namen: `<date><serviceFullName>`

    Wenn die Umwandlung relationale Mappings enthält, wird ein Ordner für relationale zu hierarchischen Mappings und ein separater Ordner für hierarchische zu relationalen Mappings erstellt.

11. Kopieren Sie den Ordner oder die Ordner mit Datenprozessorumwandlungsdiensten von dem lokalen Speicherort, wo Sie die Dateien in den PowerCenter ServiceDB-Ordner exportiert haben.
12. Wenn Sie das Mapping in eine PowerCenter-XML-Datei importiert haben, importieren Sie das Mapping in PowerCenter. Weitere Informationen über das Importieren eines Objekts in PowerCenter finden Sie im *PowerCenter 9.6.0 Repository-Handbuch*.

KAPITEL 12

Entscheidungsumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Entscheidungsumwandlung - Übersicht, 220
- Entscheidungsumwandlung - Funktionen, 221
- **Entscheidungsumwandlung - Bedingte Anweisungen** , 223
- Entscheidungsumwandlung - Operatoren, 224
- Verarbeitung von NULL-Werten in Entscheidungsumwandlungen, 225
- **Konfigurieren einer Entscheidungsstrategie** , 225
- Entscheidungsumwandlung – Erweiterte Eigenschaften, 226

Entscheidungsumwandlung - Übersicht

Die Entscheidungsumwandlung ist eine passive Umwandlung, die Bedingungen in Eingabedaten bewertet und Ausgabe basierend auf den Ergebnissen in diesen Bedingungen erstellt.

Konfigurieren Sie eine Entscheidungsumwandlung, um unterschiedliche Werte basierend auf den in Eingabefeldern gefundenen Werte zu generieren. Wenn zum Beispiel das Betriebseinkommen eines Kunden höher ist als ein bestimmter Betrag, können Sie dem Kundennamen den String "Priorität" hinzufügen.

Sie können der Entscheidungsumwandlung mehrere Entscheidungsstrategien hinzufügen. Jeder Strategie wertet eine bedingte `IF-THEN-ELSE`-Anweisung aus. In dieser Anweisung können Sie `ELSEIF`-Bedingungen verwenden oder zusätzliche `IF-THEN-ELSE`-Anweisungen verschachteln.

Die Entscheidungsumwandlung ähnelt der Ausdrucksumwandlung darin, dass Sie mit ihr bedingte Anweisungen und Funktionen zum Testen von Quelldaten verwenden können. Die Entscheidungsumwandlung weicht von der Ausdrucksumwandlung jedoch in folgenden Punkten ab:

- Die Entscheidungsumwandlung verwendet `IF-THEN-ELSE`-Anweisungen zur Auswertung von Bedingungen. Die Ausdrucksumwandlung verwendet `IIF`-Anweisungen.
- Die Entscheidungsumwandlung enthält Funktionen, die in der Ausdrucksumwandlung nicht zur Verfügung stehen.
- Jede Entscheidungsstrategie kann mehrere Ausgaben generieren.

Entscheidungsumwandlung - Funktionen

Die Entscheidungsumwandlung bietet Zugriff auf vordefinierte Funktionen, die Sie zum Definieren von Entscheidungsstrategien verwenden.

Der Entscheidungsumwandlungs-Ausdruckseditor enthält einen Ordner namens Entscheidung. Dieser Ordner enthält für die Entscheidungsumwandlung spezifische Funktionen. Der Editor enthält auch andere Ordner, die Zugriff auf die Entscheidungsumwandlungsfunktionen bieten.

Wenn Sie im Ausdruckseditor auf eine Funktion klicken, zeigt die Umwandlung neben einer Beschreibung der Funktionseigenschaften die Verwendung und die Datentypen für die Funktion an.

Hinweis: Nicht alle Ausdrucksumwandlungsfunktionen sind mit der Entscheidungsumwandlung kompatibel. Die Ausdrucksumwandlung bietet nur Zugriff auf kompatible Ausdrucksumwandlungsfunktionen.

Liste mit Entscheidungsumwandlungsfunktionen

- ABS
- ADD_TO_DATE
- ASCII
- CEIL
- CHOOSE
- CHR
- CHRCODE
- CONCAT
- CONTAINS
- CONVERT_BASE
- COS
- COSH
- CRC32
- CUME
- CURDATE
- CURTIME
- DATE_COMPARE
- DATE_DIFF
- DATECONVERT
- EXP
- FLOOR
- FV
- GET_DATE_PART
- GREATEST
- IN
- INDEXOF
- INITCAP
- INSTR

- IS_DATE
- IS_NUMBER
- ISNULL
- LAST_DAY
- LEAST
- LEFTSTR
- LENGTH
- LN
- LOG
- LOWER
- LPAD
- LTRIM
- MAKE_DATE_TIME
- MAX
- MD5
- METAPHONE
- MIN
- MOD
- MONTHCOMPARE
- MOVINGAVG
- MOVINGSUM
- NPER
- PMT
- POWER
- PV
- RAND
- RATE
- REG_EXTRACT
- REG_MATCH
- REG_REPLACE
- REPLACECHR
- REPLACESTR
- REVERSE
- RIGHTSTR
- ROUND
- RPAD
- RTRIM
- SET_DATE_PART
- SIGN

- SIN
- SINH
- SOUNDEX
- SQRT
- SUBSTR
- TAN
- TANH
- TIMECOMPARE
- TO_CHAR
- TO_DATE
- TO_FLOAT
- TO_INTEGER
- TRUNC
- UPPER
- XOR

Hinweis: Verwenden Sie einen konstanten Wert zum Definieren des Datumsformats in den CURDATE-, DATE_COMPARE-, DATECONVERT- und MONTHCOMPARE-Funktionen in der Entscheidungsumwandlung.

Entscheidungsumwandlung - Bedingte Anweisungen

Die Entscheidungsumwandlung verwendet bedingte IF-THEN-ELSE-Anweisungen zum Auswerten von Eingabedaten.

In dieser bedingten Anweisung können Sie ELSEIF-Bedingungen verwenden oder zusätzliche IF-THEN-ELSE-Anweisungen verschachteln. Für die bedingten Anweisungen in der Entscheidungsumwandlung wird folgendes Format verwendet:

```
// Primary condition
IF  <Boolean expression>
THEN <Rule Block>
// Optional - Multiple ELSEIF conditions
ELSEIF  <Boolean expression>
THEN  <Rule Block>
// Optional ELSE condition
ELSE  <Rule Block>
ENDIF
```

Sie können zusätzliche bedingte Anweisungen in einem Regelblock verschachteln.

Entscheidungsumwandlung - Operatoren

Verwenden Sie Entscheidungsumwandlungs-Operatoren zum Definieren von Entscheidungsstrategien.

In der folgenden Tabelle werden die Entscheidungsumwandlungs-Operatoren beschrieben:

Operatortyp	Operator	Beschreibung
Zuweisung	:=	Weist einem Port einen Wert hinzu.
Boolescher Wert	AND	Fügt eine erforderliche logische Bedingung hinzu. Damit der übergeordnete boolesche Ausdruck wahr ist, müssen alle von diesem Operator verknüpften logischen Bedingungen wahr sein.
Boolescher Wert	OR	Fügt eine logische Bedingung hinzu. Damit der übergeordnete boolesche Ausdruck wahr ist, muss mindestens eine von diesem Operator verknüpfte logische Bedingung wahr sein.
Boolescher Wert	NOT	Gibt eine negative logische Bedingung an. Damit der übergeordnete boolesche Ausdruck wahr ist, muss die mit diesem Operator verknüpfte negative Bedingung wahr sein.
Entscheidung	=	Testet, ob verglichene Einträge gleich sind. Mit String- oder numerischen Datentypen verwenden.
Entscheidung	<>	Testet, ob verglichene Einträge ungleich sind. Mit String- oder numerischen Datentypen verwenden.
Entscheidung	<	Testet, ob ein Wert kleiner als ein anderer Wert ist. Mit numerischen Datentypen verwenden.
Entscheidung	<=	Testet, ob ein Wert kleiner oder gleich einem anderen Wert ist. Mit numerischen Datentypen verwenden.
Entscheidung	>	Testet, ob ein Wert größer als ein anderer Wert ist. Mit numerischen Datentypen verwenden.
Entscheidung	>=	Testet, ob ein Wert größer oder gleich einem anderen Wert ist. Mit numerischen Datentypen verwenden.
Numerisch	-	Subtraktion
Numerisch	NEG	Negation
Numerisch	+	Hinzufügung
Numerisch	*	Multiplikation
Numerisch	/	Division
Numerisch	%	Modulo. Gibt den Rest nach dem Dividieren einer Zahl nach der anderen zurück.
String	||	Verkettet Strings.

Verarbeitung von NULL-Werten in Entscheidungsumwandlungen

Die Verarbeitungsweise von NULL-Werten legt fest, wie der Datenintegrationsdienst Daten mit NULL-Werten in Entscheidungsumwandlungen verarbeitet.

Wenn Sie die Verarbeitung von NULL-Werten aktivieren, behält die Entscheidungsumwandlung die ursprüngliche Form von NULL-Eingabedaten bei. Die Umwandlung wertet Funktionen mithilfe des NULL-Eingabewerts aus.

Wenn Sie die Verarbeitung von NULL-Werten deaktivieren, weist die Entscheidungsumwandlung den NULL-Eingabedaten einen Standardwert zu. Die Umwandlung wertet Funktionen mithilfe des Standardwerts aus. Wenn beispielsweise ein Eingabefeld vom Typ Integer einen NULL-Wert enthält, weist die Entscheidungsumwandlung der Eingabe den Wert 0 zu und wertet Funktionen mithilfe des Eingabewerts 0 aus.

Standardmäßig ist die Verarbeitung von NULL-Werten in der Entscheidungsumwandlung deaktiviert. Sie können die Verarbeitung von NULL-Werten auf der Registerkarte **Strategien** aktivieren. Sie können die Verarbeitung von NULL-Werten nach der Konfiguration einer Strategie für die Umwandlung aktivieren.

Konfigurieren einer Entscheidungsstrategie

Um eine Entscheidungsstrategie zu konfigurieren, verbinden Sie die Quelldaten mit der Entscheidungsumwandlung und bearbeiten Sie die Eigenschaften in den Umwandlungsansichten.

1. Öffnen Sie eine Entscheidungsumwandlung.
2. Überprüfen Sie, ob die Umwandlung Eingabe- und Ausgabeports enthält.
3. Wählen Sie die Ansicht **Entscheidung** aus.
4. Klicken Sie auf **Hinzufügen**.
5. Geben Sie einen Namen für die Strategie ein.
6. Geben Sie im Bereich **Ausdruck** eine bedingte `IF-THEN-ELSE`-Anweisung ein.
7. Um eine Funktion hinzuzufügen, navigieren Sie zu den Funktionen auf der Registerkarte **Funktionen** und doppelklicken Sie auf einen Funktionsnamen.

   Tipp: Um schnell eine Funktion einzugeben, geben Sie die ersten Buchstaben des Funktionsnamens ein und wählen Sie `Strg-Leertaste` aus.
8. Um einen Port im Ausdruck einzugeben, navigieren Sie zu den Ports auf der Registerkarte **Ports**. Doppelklicken Sie auf einen Portnamen, um ihn zum Ausdruck hinzuzufügen. Optional können Sie auf **Ausgabeports bearbeiten** klicken, um Ausgabeporteinstellungen zu bearbeiten oder Ausgabeports hinzuzufügen.
9. Sie können auch Kommentarzeilen hinzufügen, indem Sie "//" gefolgt von Ihren Kommentaren eingeben.
10. Klicken Sie auf **Validieren**, um zu bestimmen, ob der Entscheidungsausdruck gültig ist.
11. Klicken Sie auf **OK**, um die Strategie zu speichern.
12. Optional können Sie zusätzliche Strategien hinzufügen. Jede Strategie muss eindeutige Ausgabeports verwenden. Strategien können Ausgabeports nicht gemeinsam nutzen.

Entscheidungsumwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Datenintegrationsdienst Daten für die Entscheidungsumwandlung verarbeitet.

Konfigurieren Sie die folgenden erweiterten Eigenschaften für eine Entscheidungsumwandlung:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Partitionierbar

Die Umwandlung kann mit mehreren Threads verarbeitet werden. Löschen Sie diese Option, wenn der Datenintegrationsdienst einen Thread verwenden soll, um die Umwandlung zu verarbeiten. Der Datenintegrationsdienst kann mehrere Threads verwenden, um die verbleibenden Mapping-Pipeline-Stages zu verarbeiten.

Unter Umständen möchten Sie Partitionierung für eine Entscheidungsumwandlung deaktivieren, wenn die Umwandlung eine der folgenden numerischen Funktionen verwendet: CUME, MOVINGSUM oder MOVINGAVG. Diese Funktionen berechnen laufende Gesamt- und Durchschnittswerte Zeile für Zeile. Eine partitionierte Umwandlung, die die Funktionen CUME, MOVINGSUM oder MOVINGAVG verwendet, gibt möglicherweise nicht dasselbe berechnete Ergebnis mit jeder Mapping-Ausführung zurück.

Wenn die Umwandlung die Funktionen CUME, MOVINGSUM und MOVINGAVG nicht verwendet, aktivieren Sie zur Leistungsoptimierung Partitionierung für die Umwandlung.

KAPITEL 13

Ausnahmeumwandlung bei dupliziertem Datensatz

Dieses Kapitel umfasst die folgenden Themen:

- Übersicht über die Ausnahmeumwandlung bei dupliziertem Datensatz, 227
- Prozessablauf der Ausnahmen bei duplizierten Datensätzen , 228
- Ausnahmen bei duplizierten Datensätzen, 228
- Ansicht der Konfiguration der Ausnahme bei dupliziertem Datensatz , 229
- Ports, 231
- Ausnahmeumwandlung bei dupliziertem Datensatz - Erweiterte Eigenschaften, 233
- Ausnahme bei mehrfachem Datensatz - Mapping-Beispiel, 234
- Erstellen einer Ausnahmeumwandlung bei dupliziertem Datensatz, 239

Übersicht über die Ausnahmeumwandlung bei dupliziertem Datensatz

Bei der Ausnahmeumwandlung bei dupliziertem Datensatz handelt es sich um eine aktive Umwandlung, die die Ausgabe eines Datenqualitätsprozesses liest und duplizierte Datensätze erkennt, die manuell überprüft werden müssen. Die Ausnahmeumwandlung bei dupliziertem Datensatz ist eine aus mehreren Gruppen bestehende Umwandlung.

Die Ausnahmeumwandlung bei dupliziertem Datensatz erhält Eingaben von einer anderen Umwandlung oder von einem Datenobjekt in einem anderen Mapping. Die Eingabe in die Ausnahmeumwandlung muss einen numerischen Übereinstimmungswert enthalten, welchen die Umwandlung verwendet, um festzustellen, ob es sich bei dem Datensatz um ein Duplikat handelt. Legen Sie einen oberen und unteren Übereinstimmungswert für die Ausnahmeumwandlung bei dupliziertem Datensatz fest.

Die Ausnahmeumwandlung bei dupliziertem Datensatz führt einen der folgenden Schritte aus:

- Wenn der Übereinstimmungswert größer als oder gleich der oberen Schwelle ist, behandelt die Umwandlung den Datensatz als ein Duplikat und schreibt ihn ein Datenbankziel.
- Wenn der Übereinstimmungswert kleiner als die obere Schwelle und größer als die untere Schwelle ist, behandelt die Umwandlung den Datensatz als mögliches Duplikat und schreibt ihn in ein anderes Ziel zur manuellen Überprüfung. Wenn der Datensatz zu einem Cluster gehört, schreibt die Umwandlung alle Datensätze im Cluster in das Ziel.

- Wenn ein Cluster einen Übereinstimmungswert hat, der unter der unteren Schwelle liegt, wechseln alle Datensätze in dem Cluster zur Ausgabegruppe eindeutiger Datensätze. Cluster der Größe 1 werden unabhängig vom Übereinstimmungswert an die eindeutige Gruppe weitergeleitet. Standardmäßig schreibt die Ausnahmeumwandlung eindeutige Datensätze nicht in ein Ziel. Sie können die Umwandlung für die Rückgabe der eindeutigen Datensätze konfigurieren.
- Wenn ein Übereinstimmungswert in einem Cluster nicht zwischen 0 und 100 liegt, ignoriert die Ausnahmeumwandlung alle Zeilen in diesem Cluster. Der Datenintegrationsdienst protokolliert eine Nachricht mit der clusterID.

Prozessablauf der Ausnahmen bei duplizierten Datensätzen

Die Ausnahmeumwandlung bei dupliziertem Datensatz analysiert die Ausgaben anderer Datenqualitätsumwandlungen und erzeugt Tabellen, die Datensätze mit unterschiedlichen Datenqualiltäts-Stufen enthalten.

Sie können Datenqualitätsumwandlungen in einem einzigen Mapping konfigurieren, oder Sie können Mappings für verschiedene Prozessphasen erzeugen.

Sie können das Analyst-Tool verwenden, um duplizierte Datensätze zu überprüfen und zu aktualisieren, die manuell überprüft werden müssen.

Nutzen Sie das Developer-Tool, um die folgenden Aufgaben durchzuführen:

1. Erzeugen Sie ein Mapping, das Score-Werte für Datenqualitätsprobleme erstellt.
2. Verwenden Sie eine Match-Umwandlung im Cluster-Modus, um Score-Werte für duplizierte Datensatzausnahmen zu erstellen.
3. Konfigurieren Sie die Ausnahmeumwandlung bei dupliziertem Datensatz, um die Match-Umwandlungsausgabe zu lesen. Konfigurieren Sie die Umwandlung so, dass Datensätze basierend auf den Übereinstimmungswerten in die Datenbanktabellen geschrieben werden.
4. Konfigurieren Sie Zieldatenobjekte für die automatische Konsolidierung von Datensätzen.
5. Klicken Sie auf die Option **Erstellen einer duplizierten Datensatztabelle**, um die duplizierte Datensatztabelle zu erstellen, und fügen Sie diese der Mapping-Arbeitsfläche hinzu.
6. Fügen Sie das Mapping dem Arbeitsablauf hinzu.
7. Konfigurieren Sie eine Human-Aufgabe, um den Benutzern die manuelle Überprüfung möglicher duplizierter Datensätze zuzuweisen. Benutzer können die Datensätze im Analyst-Tool überprüfen und aktualisieren.

Ausnahmen bei duplizierten Datensätzen

Sie können eine Ausnahmeumwandlung bei dupliziertem Datensatz verwenden, um Cluster duplizierter Daten zu identifizieren, die manuell überprüft werden müssen. Den Übereinstimmungswert der Datensätze in Clustern bestimmt die potenziellen Duplikate. Sie können die oberen und unteren Schwellenwerte für die

Übereinstimmungswerte bei der Umwandlung konfigurieren. Die oberen und unteren Schwellenwerte bestimmen den Grad der Ähnlichkeit.

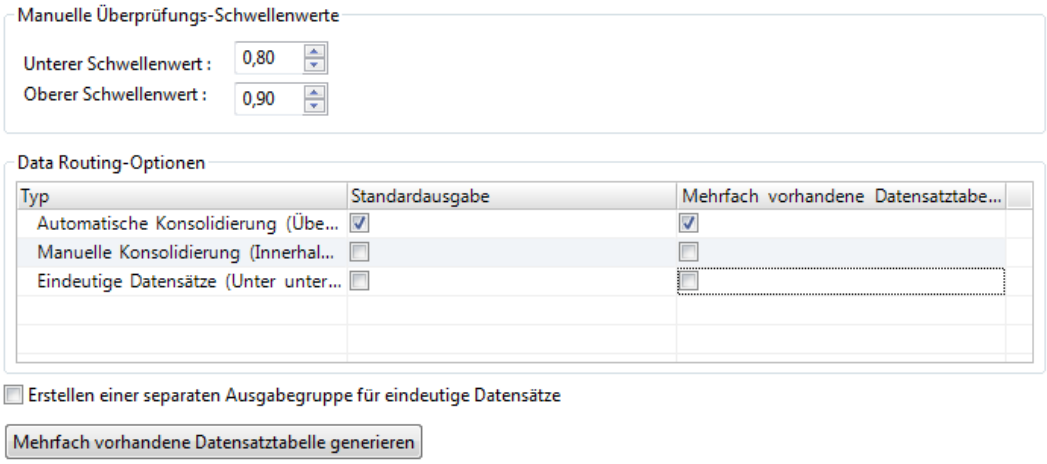
Ein Cluster enthält verwandte Datensätze, die einen Matching-Vorgang zusammenfasst. Die Match-Umwandlung erstellt Cluster, indem sie den Duplikatsanalysevorgang und den Identitätsauflösungsvorgang verwendet. Jeder Datensatz in einem Cluster hat dieselbe Cluster-ID. Wenn der niedrigste Übereinstimmungswert in einem Cluster zwischen dem oberen und dem unteren Schwellenwert liegt, identifiziert die Ausnahmeumwandlung bei dupliziertem Datensatz den Cluster als Ausnahmecluster bei dupliziertem Datensatz. Die Match-Umwandlung fügt allen Datensätzen eine Cluster-ID-Wertspalte hinzu. Duplizierte Datensätze erhalten dieselbe Cluster-ID.

Der niedrigste Datensatz-Score in einem Cluster bestimmt den Clustertyp. Ein Cluster kann 11 Datensätze mit einem Übereinstimmungswert von 0,95 und einen Datensatz mit einem Übereinstimmungswert von 0,79 aufweisen. Wenn der obere Schwellenwert 0,9 und der untere Schwellenwert 0,8 ist, schreibt die Ausnahmeumwandlung die Datensätze in die Tabelle für eindeutige Datensätze.

Ansicht der Konfiguration der Ausnahme bei dupliziertem Datensatz

Definieren Sie die Match-Score-Schwellenwerte und konfigurieren Sie das Verzeichnis, in das die Ausnahmeumwandlung bei dupliziertem Datensatz die Ausgabedaten des verschiedenen Typs schreibt.

Die folgende Abbildung zeigt die Eigenschaften, die Sie konfigurieren können:



Sie können die folgenden Eigenschaften konfigurieren:

Unterer Schwellenwert

Die untere Grenze für den Score-Bereich duplizierter Datensätze. Die Umwandlung verarbeitet Datensätze mit Match-Scores unter diesem Wert als eindeutige Datensätze. Der untere Schwellenwert ist eine Zahl zwischen 0 und 1.

Oberer Schwellenwert

Die obere Grenze für den Score-Bereich duplizierter Datensätze. Die Umwandlungsprozess-Datensätze mit Match-Scores größer oder gleich dem oberen Schwellenwert als duplizierte Datensätze. Der obere Schwellenwert ist eine Zahl größer als die Zahl für den unteren Schwellwert.

Automatische Konsolidierung

Cluster, in denen alle Datensätze Match-Scores aufweisen, die über dem oberen Schwellenwert liegen. Automatische Konsolidierungscluster müssen nicht überprüft werden. Die Datensätze sind dupliziert. Mit der Konsolidierungsumwandlung können Sie Datensätze kombinieren. Standardmäßig schreibt die Ausnahmeumwandlung bei dupliziertem Datensatz automatische Konsolidierungscluster in Standardausgabeports.

Manuelle Konsolidierung

Cluster, in denen alle Datensätze Match-Scores aufweisen, die größer oder gleich dem unteren Schwellenwert sind, und der Match-Score mindestens eines Datensatzes unter dem oberen Schwellenwert liegt. Sie müssen eine manuelle Überprüfung der Cluster durchführen, um zu ermitteln, ob sie duplizierte Datensätze enthalten. Standardmäßig schreibt die Ausnahmeumwandlung bei dupliziertem Datensatz manuelle Konsolidierungsdatensätze in die duplizierte Datensatztabelle.

Eindeutige Konsolidierung

Cluster mit einer Clustergröße gleich 1 oder Cluster, in denen alle Datensätze einen Match-Score aufweisen, der den unteren Schwellenwert unterschreitet. Eindeutige Datensatzcluster sind keine Duplikate. Standardmäßig schreibt die Ausnahmeumwandlung bei dupliziertem Datensatz keine eindeutigen Datensätze in eine Ausgabetabelle.

Standardausgabe

Die Datensatztypen, die von der Umwandlung auf die Standardausgabeports geschrieben werden.

Als Standardeinstellung ist die automatische Konsolidierung von Datensätzen festgelegt.

Duplizierte Datensatztabelle

Gibt die Datensatztypen an, die von der Umwandlung auf die Ausgabeports für duplizierte Datensätze geschrieben werden. Als Standardeinstellung ist die manuelle Konsolidierung von Datensätzen festgelegt.

Erstellen einer separaten Ausgabegruppe für eindeutige Datensätze

Erstellt eine separate Ausgabegruppe für eindeutige Datensätze. Wenn Sie keine separate Tabelle für eindeutige Datensätze erstellen, können Sie die Umwandlung so konfigurieren, dass die eindeutigen Datensätze in eine der anderen Gruppen geschrieben werden. Sie können das Schreiben von eindeutigen Datensätzen in eine Ausgabetabelle überspringen. Standardwert ist "Deaktiviert".

Generieren einer Tabelle für duplizierte Datensätze

Erstellt ein Datenbankobjekt, das die Clusterdaten von duplizierten Datensätzen enthält. Wenn Sie diese Option auswählen, erstellt das Developer-Tool das Datenbankobjekt. Das Developer-Tool fügt das Objekt zum Modellrepository hinzu, fügt eine Instanz des Objekts zur Mapping-Arbeitsfläche hinzu und verknüpft die Ports mit dem Objekt.

Erstellen einer duplizierten Datensatztabelle

Sie können eine duplizierte Datensatztabelle aus einer Ausnahmeumwandlung bei dupliziertem Datensatz in einem Mapping erstellen.

1. Klicken Sie in der Ansicht **Konfiguration** zum Erstellen der Tabelle auf **Erstellen einer doppelten Datensatztabelle**.

   Das Dialogfeld **Relationales Datenobjekt erstellen** wird geöffnet.
2. Suchen Sie und wählen Sie eine Verbindung zur Datenbank aus, die die Tabelle enthalten soll.
3. Geben Sie in der Datenbank einen Namen für die duplizierte Datensatztabelle ein.
4. Geben Sie einen Namen für das doppelte Datensatztabellenobjekt im Modellrepository ein.

5. Klicken Sie auf **Fertig stellen**.

   Das Developer-Tool fügt die neue Tabelle zur Mapping-Arbeitsfläche hinzu.

Ports

Die Ausnahmeumwandlung bei dupliziertem Datensatz verfügt über mehrere Gruppen von Eingabe- und Ausgabeports.

Die folgende Abbildung zeigt ein Beispiel für Eingabe- und Ausgabeports:

Ports

	Name	Typ	Gesamtstellenanzahl	Größenordnung	Eingabe	Ausgabe	Standard	Beschreibung
	Eingaben							
	Daten (3)							
1	Employee	decimal	10	0	☑	☐		
2	Name	string	10	0	☑	☐		
3	Addr1	string	10	0	☑	☐		
	Kontrolle (3)							
1	Score	double	15	0	☑	☐		
2	Row_Ide...	string	25	0	☑	☐		
3	Cluster_ID	integer	10	0	☑	☐		
	Ausgaben							
	Standardausgabe (6)							
1	Score	double	15	0	☐	☑		
2	Row_Ide...	string	25	0	☐	☑		
3	Cluster_ID	integer	10	0	☐	☑		
4	Employee	decimal	10	0	☐	☑		
5	Name	string	10	0	☐	☑		
6	Addr1	string	10	0	☐	☑		
	Cluster-Daten (9)							
1	Row_Ide...	bigint	19	0	☐	☑		
2	Sequent...	bigint	19	0	☐	☑		
3	Cluster_ID	integer	10	0	☐	☑		

Eingabeports der Ausnahmeumwandlung bei dupliziertem Datensatz

Eine Ausnahmeumwandlung bei dupliziertem Datensatz verfügt über eine Datengruppe und eine Kontrollgruppe an Eingabeports.

Die Gruppe der**Daten** enthält die benutzerdefinierten Ports, die die Quelldaten empfangen.

Die Ports für die **Kontrolle** empfangen Metadaten, die die Match-Umwandlung den Quelldaten hinzufügt. In der folgenden Tabelle werden die Ports für die **Kontrolle** beschrieben:

Port	Beschreibung
Score	Dezimalwert zwischen 0 und 1. Kennzeichnet den Grad der Ähnlichkeit mit dem Datensatz, der den Datensatz mit dem Cluster verknüpft hat.
Row_Identifier	Eindeutiger Bezeichner für den Datensatz
Cluster_ID	Die ID des Match-Clusters, zu dem der Datensatz gehört

Ausgabeports der Ausnahmeumwandlung bei dupliziertem Datensatz

Eine Ausnahmeumwandlung bei duplizierten Datensätzen hat mehrere Ausgabegruppen. Standardmäßig schreibt die Umwandlung duplizierte Datensätze in die Gruppe **Standardausgabe**. Die Umwandlung schreibt potenzielle Matches in die Gruppe **Cluster-Daten**. Sie können eine Ausgabegruppe für eindeutige Datensätze hinzufügen.

Sie können die Datensatztypen, welche die Umwandlung in die Ausgabeports schreibt, durch eine Änderung der Standardeinstellungen in der Ansicht **Konfiguration** ändern.

In der folgenden Tabelle werden die Ausgabeports für die Gruppe **Standardausgabe** beschrieben:

Port	Beschreibung
Score	Dezimalwert zwischen 0 und 1. Identifiziert den Grad der Ähnlichkeit zwischen dem Datensatz und einem anderen Datensatz in einem Cluster.
Row_Identifier	Eindeutiger Bezeichner für den Datensatz
Cluster_ID	ID eines Clusters, dem die Match-Umwandlung den Datensatz zugeordnet hat.
Benutzerdefinierte Ports	Die Felder der Quelldaten.

In der folgenden Tabelle werden die Ausgabeports für die Gruppe **Clusterdaten** beschrieben:

Port	Beschreibung
Row_Identifier	Der eindeutige Bezeichner für den Datensatz.
Sequential_Cluster_ID	Identifiziert den Cluster in einer Human-Aufgabe. Ein Arbeitsablauf verwendet die sequenzielle Cluster-ID, um einer Instanz einer Human-Aufgabe einen Cluster zuzuweisen.
Cluster_ID	Identifiziert den Cluster, zu dem der Datensatz gehört. Die Match-Umwandlung weist allen Datensätzen eine Cluster-ID zu.
Score	Dezimalwert zwischen 0 und 1. Kennzeichnet den Grad der Ähnlichkeit mit dem Datensatz, der den Datensatz mit dem Cluster verknüpft hat.
Is_Master	Eine Zeichenfolge, die angibt, ob der Datensatz der bevorzugte Datensatz im Cluster ist. Standardmäßig ist die erste Zeile eines Clusters der bevorzugte Datensatz. Wert ist J oder N.

Port	Beschreibung
Workflow_ID	ID, die den Arbeitsablauf für den Datensatz in einer Aufgabe identifiziert. Wenn das Mapping außerhalb eines Arbeitsablaufs läuft, lautet die Arbeitsablauf-ID DummyWorkflowID.
Benutzerdefinierte Ports	Die Ports der Quelldaten.

Erstellen von Ports

Fügen Sie jeden Eingabeport der Datengruppe hinzu. Wenn Sie den Eingabeport hinzufügen, fügt das Developer-Tool einen Ausgabeport mit dem gleichen Namen der Standardausgabegruppe, der Cluster-Datengruppe und der eindeutigen Datensatzgruppe hinzu.

1. Wählen Sie die Dateneingabegruppe aus.

   Die Gruppe ist hervorgehoben.
2. Klicken Sie auf **Neu (Einfügen)**.

   Das Developer-Tool fügt der Datengruppe, der Standardausgabegruppe, der Cluster-Datengruppe und der eindeutigen Datensatzgruppe ein Feld hinzu.
3. Ändern Sie je nach Bedarf den Namen des Felds.

   Das Developer-Tool ändert den Namen des Felds in den anderen Gruppen.
4. Geben Sie die verbleibenden Ports ein, die Sie für die Quelldaten hinzufügen müssen.

Ausnahmeumwandlung bei dupliziertem Datensatz - Erweiterte Eigenschaften

Die Ausnahmeumwandlung für duplizierte Datensätze enthält erweiterte Eigenschaften, die das Sortierungsverhalten, das Verhalten des Cache-Arbeitsspeichers und die Tracingebene bestimmen.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Sortieren

Legt fest, ob die Umwandlung die Eingabezeilen in den **Cluster-ID**-Portdaten sortiert. Die Eigenschaft ist standardmäßig aktiviert.

Wählen Sie die Eigenschaft aus, wenn die Eingabezeilen nicht vorsortiert sind.

Cache-Dateiverzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst temporäre Daten für die aktuelle Umwandlung schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis, wenn das Volumen der Eingabedaten größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cache-Dateigröße

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst zum Sortieren der Eingabedaten bei der Umwandlung verwendet. Der Standardwert ist 400.000 Byte.

Bevor die Daten sortiert werden, weist der Datenintegrationsdienst die von Ihnen angegebene Speichermenge zu. Wenn der Sortiervorgang eine größere Menge an Daten generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Dateiverzeichnis. Wenn der Speicherbedarf für den Sortiervorgang die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Ausnahme bei mehrfachem Datensatz - Mapping-Beispiel

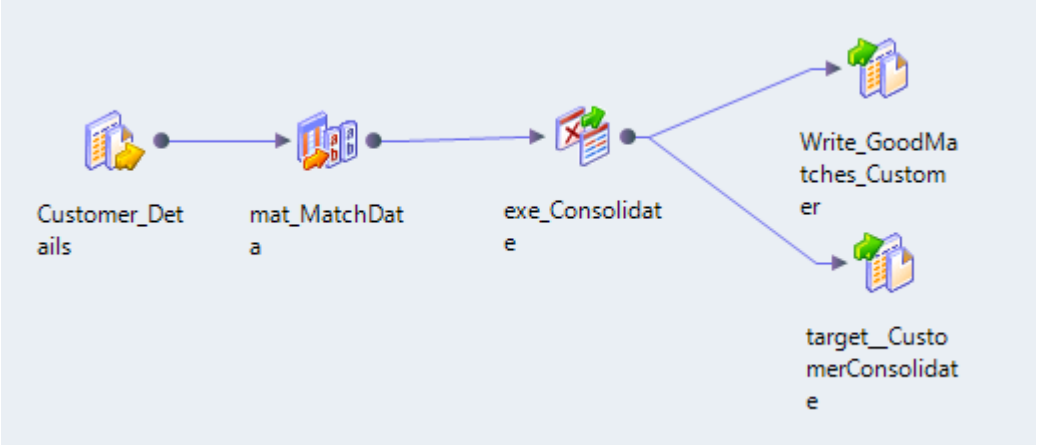
Ein Unternehmen führt ein Datenprojekt zur Überprüfung von Kundendaten durch. Vom Unternehmen wird festgestellt, dass die Kundendaten mehrere Mehrfachdatensätze aufweisen. Das Unternehmen muss manuell einige der Datensätze überprüfen, bei denen es sich möglicherweise um Duplikate handelt.

Erstellen Sie ein Datenqualitäts-Mapping, um duplizierte Kundendatensätze zu ermitteln. Das Mapping enthält die Match-Umwandlung. Die Ausnahmeumwandlung bei dupliziertem Datensatz empfängt die Ergebnisse der Match-Umwandlung. Die Ausnahmeumwandlung bei dupliziertem Datensatz schreibt jedem Datensatz-Cluster, mit einem unklaren Status, eine Datenbanktabelle zu. Ein Datenanalyst prüft die Daten im Analyst-Tool und bestimmt, bei welchen Datensätzen es sich um duplizierte Datensätze handelt.

Mapping der Ausnahmen bei duplizierten Datensätzen

Konfigurieren Sie ein Mapping der Ausnahmen bei duplizierten Datensätzen, das die Kundendatensätze prüft und duplizierte Datensätze findet.

Die folgende Abbildung zeigt das Mapping der Ausnahmen bei duplizierten Datensätzen:



Das Mapping enthält die folgenden Objekte:

Customer_Details

Die Datenquelle, die möglicherweise duplizierte Datensätze enthält.

mat_MatchData

Eine Match-Umwandlung, die die Kundendaten prüft, um zu bestimmen, ob Datensätze übereinstimmen. Die Match-Umwandlung erstellt einen numerischen Score, durch den der Grad der Ähnlichkeit zwischen zwei Spaltenwerten dargestellt wird. Ein Algorithmus berechnet einen Übereinstimmungswert als Dezimalwert im Bereich 0 bis 1. Ein Algorithmus weist einen Score von eins zu, wenn zwei Spaltenwerte identisch sind.

exc_Consolidate

Eine Ausnahmeumwandlung bei dupliziertem Datensatz, die bestimmt, welche Datensätze mögliche duplizierte Kunden, bekannte duplizierte Kunden oder einmalige Kundendatensätze sind.

Tabelle Write_GoodMatches_Customer

Tabelle mit allen Datensätzen, die keine manuelle Überprüfung erfordern. Die Ausnahmeumwandlung bei dupliziertem Datensatz schreibt duplizierte Datensätze und eindeutige Datensätze in diese Tabelle.

Target_CustomerConsolidate table

Die Ausnahmeumwandlung schreibt die möglichen duplizierten Datensätze in die Tabelle Target_CustomerConsolidate. Datensätze in dieser Tabelle müssen im Analyst-Tool manuell überprüft werden.

Match-Umwandlung

Die Match-Umwandlung empfängt die Kundendaten und führt einen Identitätsvergleich aus.

Konfigurieren Sie die Match-Umwandlung für den Ausgabetyp Clusters-Match All. Die Match-Umwandlung gibt übereinstimmende Datensätze in Clustern zurück. Jeder Datensatz in einem Cluster muss mit mindestens einem anderen Datensatz im Cluster übereinstimmen, der einen größeren Score hat als der Schwellenwert oder diesem entspricht. Der Schwellenwert für die Übereinstimmung ist .75.

Wählen Sie die Geschäftsbereich Matching-Strategie in der Registerkarte Match-Umwandlung **Strategien**. Die Geschäftsbereichsstrategie ist eine vordefinierte Matching-Strategie, die ein Unternehmen basierend auf den Adressfeldern, kennzeichnet. Wählen Sie auf der Registerkarte Match-Umwandlung **Strategien** die Eingabeports, die in einer Übereinstimmung untersucht werden sollen. Konfigurieren Sie die Gewichtung der Strategie als .5.

Die folgende Abbildung zeigt die Konfiguration der Geschäftsbereichsstrategie für die Match-Umwandlung:

Die Match-Umwandlung fügt Cluster-Informationen zu jedem Datensatz der Ausgabe hinzu. Die Umwandlung fügt außerdem jedem Datensatz eine einmalige RowID hinzu.

Eingabegruppen der Ausnahme bei dupliziertem Datensatz

Die Ausnahmeumwandlung bei dupliziertem Datensatz hat zwei Eingabegruppen. Die Umwandlung hat eine Datengruppe, die Kundendaten empfängt. Die Umwandlung verfügt über die Kontrollgruppe, die den Übereinstimmungswert für die Zeile, die Zeilenkennung und die Cluster-ID enthält.

Die folgende Abbildung zeigt die Eingabegruppen in der Ausnahmeumwandlung:

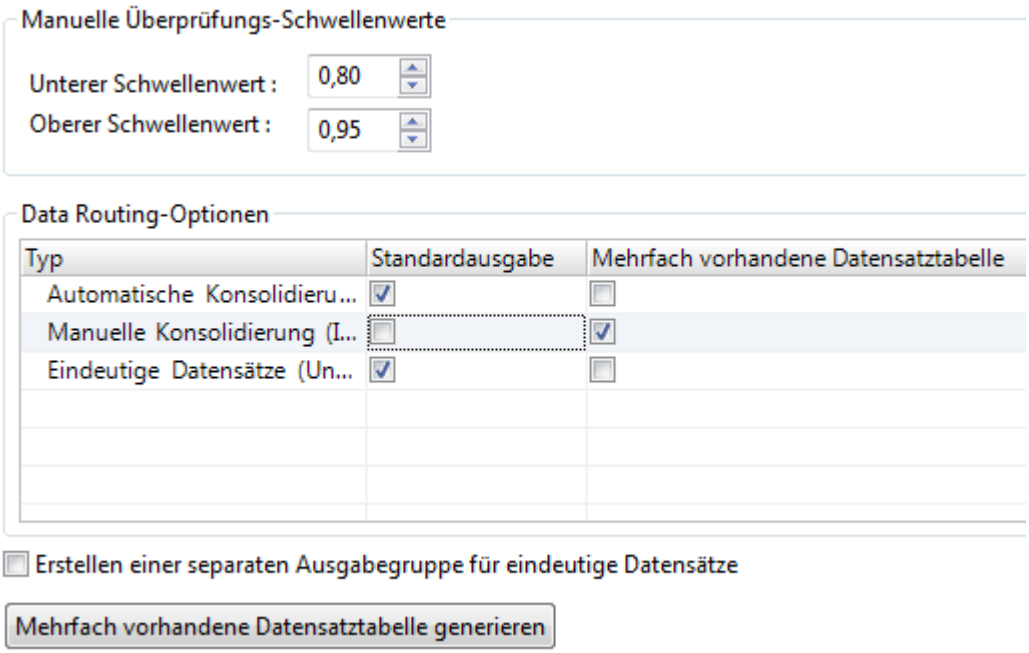
	Name	Typ	Gesamtstellenanzahl	Größenordnung	Eingabe	Ausgabe	Standard	Beschreibung
	Eingaben							
	Daten (11)							
1	CUST_ID	decimal	20	0	☑	☐		
2	COMPANY	string	200	0	☑	☐		
3	CONTACT	string	200	0	☑	☐		
4	TITLE	string	200	0	☑	☐		
5	ADDR1	string	200	0	☑	☐		
6	ADDR2	string	100	0	☑	☐		
7	ADDR3	string	100	0	☑	☐		
8	ADDR4	string	50	0	☑	☐		
9	COUNTRY	string	50	0	☑	☐		
10	PHONE	string	100	0	☑	☐		
11	EMAIL	string	100	0	☑	☐		
	Kontrolle (3)							
1	Score	double	15	0	☑	☐		
2	Row_Identif...	string	25	0	☑	☐		
3	Cluster_ID	integer	10	0	☑	☐		

Die Datengruppe enthält die Kundendaten. Die Kundendaten bestehen aus Feldern mit den Informationen: Kunden-ID, Kontakt, Anrede und Adressen. Bei der Kontrollgruppe handelt es sich um die zusätzlichen Metadaten, die die Match-Umwandlung jedem Kundendatensatz hinzugefügt hat. Der Gruppe enthält den Übereinstimmungswert, die rowID und die Cluster-ID.

Beispiel für eine Ausnahme bei dupliziertem Datensatz - Konfigurationsansicht

Definieren Sie die oberen und unteren Schwellenwerte in der Ansicht **Konfiguration**. Geben Sie an, wohin die Umwandlung die duplizierten Benutzerdatensätze, die möglichen duplizierten Datensätze und die eindeutigen Benutzerdatensätze schreiben soll.

Die folgende Abbildung zeigt die Ansicht **Konfiguration** der Ausnahmeumwandlung bei dupliziertem Datensatz:



In der folgenden Tabelle werden die Konfigurationseinstellungen beschrieben:

Option	Einstellung
Unterer Schwellenwert	0,80
Oberer Schwellenwert	0,95
Automatische Konsolidierung	Tabelle der Standardausgabe
Manuelle Konsolidierung	Duplizierte Datensatztabelle
Eindeutige Datensätze	Tabelle der Standardausgabe

Klicken Sie zum Erstellen der duplizierten Datensatztabelle auf **Erstellen einer duplizierten Datensatztabelle**. Erstellen Sie keine separate Tabelle für die eindeutigen Datensätze. Die Umwandlung schreibt die eindeutigen Datensätze in die Tabelle der Standardausgabe.

Datensätze der Standardausgabentabelle

Die Zieltabelle Write_GoodMatches_Customer erhält Zeilen aus der Standardausgabegruppe. Die Tabelle erhält eindeutige Datensätze und duplizierte Datensätze. Diese Datensätze erfordern keine manuelle Überprüfung.

Die folgende Abbildung zeigt die Datensätze der Standardausgabe, die von der Ausnahmeumwandlung zurückgeschickt werden:

Ausgabe

Name: exc_Consoli.Good_Records

	Score	Row_Identifier	Cluster_ID	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2	ADDR3
1	1	1 - 4	3	1001622	INVESTEX	NEW YORK	United Parcel Se...	"50 BROAD S...	20TH FL"	310 BRIER ROAD
2	1	1 - 5	4	7121564	"ARTHUR VAN D...	INC."	WHITEHOUSE ST...	Federal Expre...	310 BRIER RO...	767 FIFTH AVE...
3	0,90476190476...	1 - 6	4	7121565	OSTERREICHISCH...	NEW YORK	United Parcel Se...	767 FIFTH AV...	570 LEXINGT...	6803 S. TUCSO...
4	0,90476190476...	1 - 7	4	7121566	K2 ADVISORS	NEW YORK	Courier	570 LEXINGT...	51 JFK PARK...	6803 S. TUCSO...
5	0,90476190476...	1 - 8	4	7121567	MFP INVESTORS	SHORT HILLS	Federal Express I...	51 JFK PARK...	ONE PARKER...	6803 S. TUCSO...
6	0,90476190476...	1 - 9	4	7121568	EASTON	FORT LEE	US Postal Intern...	ONE PARKER...	3840 HIGHW...	6803 S. TUCSO...
7	0,90476190476...	1 - 10	4	7121569	RICE VOELKER	MANDEVILLE	Federal Express...	3840 HIGHW...	"2ND & D ST...	SW HOUSE A...
8	1	1 - 1	16	1001590	E-AGENCY	OAKLAND	Federal Express...	291 3RD STRE...	2500 WESTER...	6803 S. TUCSO...
9	0,90476190476...	1 - 2	16	1001599	BANK ONE	ELGIN	US Postal Intern...	2500 WESTER...	530 CHESNU...	6803 S. TUCSO...
10	1	1 - 3	16	1001604	KPMG PEAT MAR...	WOODCLIFF...	US Postal 2-day	530 CHESNU...	"50 BROAD S...	20TH FL"
11	1	1 - 11	16	1001658	HOUSE INFORMA...	WASHINGTON	Federal Express...	"2ND & D ST...	SW HOUSE...	<Null>
12	0,90476190476...	1 - 12	16	1001660	OPPENHEIMER M...	ENGLEWOOD	US Postal Intern...	6803 S. TUCS...	6803 S. TUCS...	6803 S. TUCSO...
13	0,90476190476...	1 - 13	16	1001659	THOMSON LEAR...	FT. WORTH	Courier	301 COMMER...	301 COMMER...	6803 S. TUCSO...
14	1	1 - 14	16	1001658	LYNCH JONES &...	NEW YORK	US Postal Overni...	875 3RD AVE.	875 3RD AVE.	6803 S. TUCSO...
15	0,90476190476...	1 - 15	16	1001691	EPICENTRIC	SAN FRANCIS...	US Postal 2-day	333 BRYANT...	333 BRYANT...	6803 S. TUCSO...
16	0,90476190476...	1 - 16	16	1001664	CHARLES SCHWAB	SAN ANTONIO	United Parcel Se...	1100 N.E. LO...	1100 N.E. LO...	6803 S. TUCSO...
17	0,90476190476...	1 - 17	16	1001694	BANK OF MONTR...	BOSTON	US Postal 2-day	"125 BROAD...	38TH FLOOR"	"125 BROAD ST...
18	1	1 - 18	16	1001729	JOSEPH H. GIBBO...	HIGHLAND P...	United Parcel Se...	1212 LINCOL...	1212 LINCOL...	6803 S. TUCSO...
19	0,90476190476...	1 - 19	16	1001724	D E SHAW & CO...	NEW YORK	US Postal Overni...	120 WEST 45...	120 WEST 45...	6803 S. TUCSO...
20	0,90476190476...	1 - 20	16	1001732	BANG NETWORKS	SAN FRANCIS...	Courier	808 BRANNA...	808 BRANNA...	6803 S. TUCSO...
21	0,90476190476...	1 - 21	16	1001736	FOREX CAPITAL...	NEW YORK	Federal Express...	11 BROADW...	11 BROADW...	6803 S. TUCSO...
22	0,90476190476...	1 - 22	16	1001738	"INTERFACE SYST...	INC."	ANN ARBOR	United Parcel...	5855 INTERF...	5855 INTERFAC...
23	0,90476190476...	1 - 23	16	1001758	BANK OF NEW Y...	LAKE MARY	United Parcel Se...	100 COLONIA...	100 COLONIA...	6803 S. TUCSO...

Der Datensatz enthält die folgenden Felder:

Score

Ein Übereinstimmungswert, der den Grad der Ähnlichkeit angibt, der zwischen einem Datensatz und einem anderen Datensatz im Cluster besteht. Bei Datensätzen mit einem Übereinstimmungswert, der 1 entspricht, handelt es sich um duplizierte Datensätze, die nicht geprüft werden müssen. Ein Cluster, in dem der Übereinstimmungswert des Datensatzes unter dem unteren Schwellenwert liegt, ist kein duplizierter Cluster.

Row_Identifier

Eine Zeilennummer, die jede Zeile einmalig in der Tabelle kennzeichnet. In diesem Beispiel ist die Kunden-ID in der Zeilenkennung enthalten.

Cluster-ID

Eine einmalige Bezeichnung für ein Cluster. Jeder Datensatz in einem Cluster erhält dieselbe Cluster-ID. Die ersten vier Datensätze in den Daten der Probeausgabe sind einmalig. Jeder Datensatz hat eine einmalige Cluster-ID. Die Zeilen fünf bis neun gehören zu Cluster fünf. Bei den Datensätzen in diesen Cluster handelt es sich um einen duplizierten Datensatz, da Ähnlichkeiten in den Adressfeldern bestehen.

Felder der Quelldaten

Der Tabellengruppe der Standardausgabe empfängt alle Quelldatenfelder.

Cluster-Ausgabe

Die Tabelle Target_CustomerConsolidate empfängt Datensätze von der Ausgabegruppe des Clusters. Die Ausgabegruppe des Clusters gibt Datensätze, die möglicherweise duplizierte Datensätze sind, zurück. Die Datensätze in der Target_CustomerConsolidate-Tabelle müssen im Analyst-Tool manuell überprüft werden.

Die folgende Abbildung zeigt einige der Datensätze und der Felder in der Tabelle Target_CustomerConsolidate:

Name: exc_Consoli.Cluster_Data

	Row_Identifier	Sequential_Cluster_ID	Cluster_ID	Score	Is_Master	Workflow_ID	CUST_ID	COMPANY	CONTACT	TITLE	ADDR1	ADDR2
1	0	0	3	1	Y	DummyWorkfl...	1001622	INVESTEX	NEW YORK	United Parc...	"50 BROAD S...	20TH FL"
2	1	0	3	1	N	DummyWorkfl...	1001622	INVESTEX	NEW YORK	United Parc...	"50 BROAD S...	20TH FL"
3	2	1	4	1	Y	DummyWorkfl...	7121564	"ARTHUR VAN...	INC."	WHITEHOU...	Federal Expre...	310 BRIER R
4	3	1	4	1	N	DummyWorkfl...	7121564	"ARTHUR VAN...	INC."	WHITEHOU...	Federal Expre...	310 BRIER R
5	4	1	4	0,90476190...	N	DummyWorkfl...	7121565	OSTERREICHIS...	NEW YORK	United Parc...	767 FIFTH AV...	570 LEXINGT
6	5	1	4	0,90476190...	N	DummyWorkfl...	7121566	K2 ADVISORS	NEW YORK	Courier	570 LEXINGT...	51 JFK PARK
7	6	1	4	0,90476190...	N	DummyWorkfl...	7121567	MFP INVESTORS	SHORT HIL...	Federal Exp...	51 JFK PARK...	ONE PARKE
8	7	1	4	0,90476190...	N	DummyWorkfl...	7121568	EASTON	FORT LEE	US Postal I...	ONE PARKER...	3840 HIGHW
9	8	1	4	0,90476190...	N	DummyWorkfl...	7121569	RICE VOELKER	MANDEVIL...	Federal Exp...	3840 HIGHW...	"2ND & D S
10	9	2	16	1	Y	DummyWorkfl...	1001590	E-AGENCY	OAKLAND	Federal Exp...	291 3RD STRE...	2500 WESTE
11	10	2	16	1	N	DummyWorkfl...	1001590	E-AGENCY	OAKLAND	Federal Exp...	291 3RD STRE...	2500 WESTE
12	11	2	16	0,90476190...	N	DummyWorkfl...	1001599	BANK ONE	ELGIN	US Postal I...	2500 WESTER...	530 CHESNU
13	12	2	16	1	N	DummyWorkfl...	1001604	KPMG PEAT M...	WOODCLI...	US Postal 2...	530 CHESNUT...	"50 BROAD
14	13	2	16	1	N	DummyWorkfl...	1001658	HOUSE INFOR...	WASHING...	Federal Exp...	"2ND & D ST...	SW HOUSE
15	14	2	16	0,90476190...	N	DummyWorkfl...	1001660	OPPENHEIMER...	ENGLEWO...	US Postal I...	6803 S. TUCS...	6803 S. TUC
16	15	2	16	0,90476190...	N	DummyWorkfl...	1001659	THOMSON LE...	FT. WORTH	Courier	301 COMMER...	301 COMME
17	16	2	16	1	N	DummyWorkfl...	1001658	LYNCH JONES...	NEW YORK	US Postal...	875 3RD AVE.	875 3RD AVE
18	17	2	16	0,90476190...	N	DummyWorkfl...	1001691	EPICENTRIC	SAN FRAN...	US Postal 2...	333 BRYANT...	333 BRYANT
19	18	2	16	0,90476190...	N	DummyWorkfl...	1001664	CHARLES SCH...	SAN ANTO...	United Parc...	1100 N.E. LO...	1100 N.E. LC
20	19	2	16	0,90476190...	N	DummyWorkfl...	1001694	BANK OF MON...	BOSTON	US Postal 2...	"125 BROAD...	38TH FLOO

Der Datensatz enthält die folgenden Felder:

Row_Identifier

Eine Zahl kennzeichnet einmalig jede Zeile in der Tabelle.

Sequenzielle Cluster-ID

Eine sequenzielle Kennung für jedes Cluster, das in einer Human-Task überprüft werden soll. Die Ausnahmeumwandlung bei dupliziertem Datensatz fügt die sequenzielle Cluster-ID den Datensätzen in der Ausgabegruppe der Cluster-Daten hinzu.

Cluster-ID

Eine einmalige Bezeichnung für ein Cluster. Die Match-Umwandlung weist allen Datensätzen der Ausgabe eine Cluster-ID zu. Duplizierte Datensätze und möglicherweise duplizierte Datensätze haben eine gemeinsame Cluster-ID. Ein eindeutiger Datensatz erhält eine Cluster-ID. Die ID-Nummer wird mit keinem anderen Datensatz geteilt.

Score

Ein Übereinstimmungswert, der den Grad der Ähnlichkeit angibt, der zwischen einem Datensatz und einem anderen Datensatz im Cluster besteht. Die Datensätze, die manuelle überprüft werden müssen, haben einen Übereinstimmungswert der unter .95 und über .80 liegt.

Is Master

Gibt an, ob es sich bei dem Datensatz, um den bevorzugten Datensatz im Cluster handelt.

WorkflowID

Bei der WorkflowID handelt es sich um eine DummyWorkflowID, da die Umwandlung sich nicht in einem Arbeitsablauf befindet.

Datensatzfelder

Die anderen Felder im Datensatz enthalten die Quelldaten der Kunden.

Erstellen einer Ausnahmeumwandlung bei dupliziertem Datensatz

Konfigurieren Sie die Eingabeports, wenn Sie eine Ausnahmeumwandlung bei dupliziertem Datensatz konfigurieren. Definieren Sie die oberen und die unteren Schwellenwerte, um die Übereinstimmungen zu

ermitteln. Geben Sie an, wohin die duplizierten Datensätze und die eindeutigen Datensätze geschrieben werden sollen.

1. Erstellen Sie eine wiederverwendbare oder nicht wiederverwendbare Ausnahmeumwandlung bei dupliziertem Datensatz.
   - Wählen Sie zum Erstellen einer wiederverwendbaren Umwandlung **Datei** > **Neu** > **Umwandlung** und wählen Sie eine Ausnahmeumwandlung bei dupliziertem Datensatz aus.
   - Öffnen Sie zum Erstellen einer nicht wiederverwendbaren Umwandlung ein Mapping und fügen Sie die Umwandlung zur Mapping-Arbeitsfläche hinzu. Wählen Sie vom Assistenten eine Ausnahmeumwandlung bei dupliziertem Datensatz aus.
2. Klicken Sie auf **Weiter** oder auf **Beenden**.

   Wenn Sie auf **Fertig stellen** klicken, können Sie die Standardschwellenwerte und die Data Routing-Optionen aktualisieren, bevor Sie die Umwandlung erstellen.
3. Geben Sie in der Konfigurationsansicht den oberen und unteren Schwellenwert für die Übereinstimmung an.
4. Konfigurieren Sie im Abschnitt **Data Routing-Optionen** die Eigenschaften der Standardausgabe und der Ausnahmetabelle, um den Ort festzulegen, in den die Umwandlung den jeweiligen Datensatztyp schreibt.

   Sie können wahlweise ändern, wohin die duplizierten Datensätze, die zu prüfenden duplizierten Datensätze und die eindeutigen Datensätze geschrieben werden sollen.
5. Wahlweise können Sie eine eindeutige Datensatztabelle erzeugen. Geben Sie die Datenbankverbindung und die Angaben zum Tabellennamen für die neue Tabelle ein. Wenn Sie eine eindeutige Datensatztabelle erzeugen, wird bei der Umwandlung ein Datenbankobjekt im Modellrepository erstellt.
6. Konfigurieren Sie die Eingabeports. Wenn Sie einen Eingabeport hinzufügen, gibt das Developer Tool der Ausgabegruppe denselben Portnamen.
   - Wenn Sie eine wiederverwendbare Umwandlung erstellen, wählen Sie die Registerkarte **Ports** aus und fügen Sie Ports für die Daten hinzu, die mit der Umwandlung verbunden werden sollen.
   - Wenn Sie eine nicht wiederverwendbare Umwandlung erstellen, fügen Sie der Mapping-Arbeitsfläche andere Objekte hinzu und ziehen Sie die Eingabeports in die Umwandlung.
7. Verbinden Sie die Umwandlungsausgabeports mit einem oder mehreren Datenzielen. Verbinden Sie die Ausgabeports mit den Datenobjekten, die den Ausgabeoptionen entsprechen, die Sie in der Ansicht **Konfiguration** festgelegt haben.
   - Wenn Sie eine wiederverwendbare Umwandlung erstellen, fügen Sie die Umwandlung zu einem Mapping hinzu und verbinden Sie die Ausgabeports.
   - Wenn Sie eine nicht wiederverwendbare Umwandlung erstellen, verbindet die Umwandlung die Ports mit der Cluster-Datentabelle. Verbinden Sie Ausgabeports mit den anderen Datenzielen.

KAPITEL 14

Ausdrucksumwandlung

Dieses Kapitel umfasst die folgenden Themen:

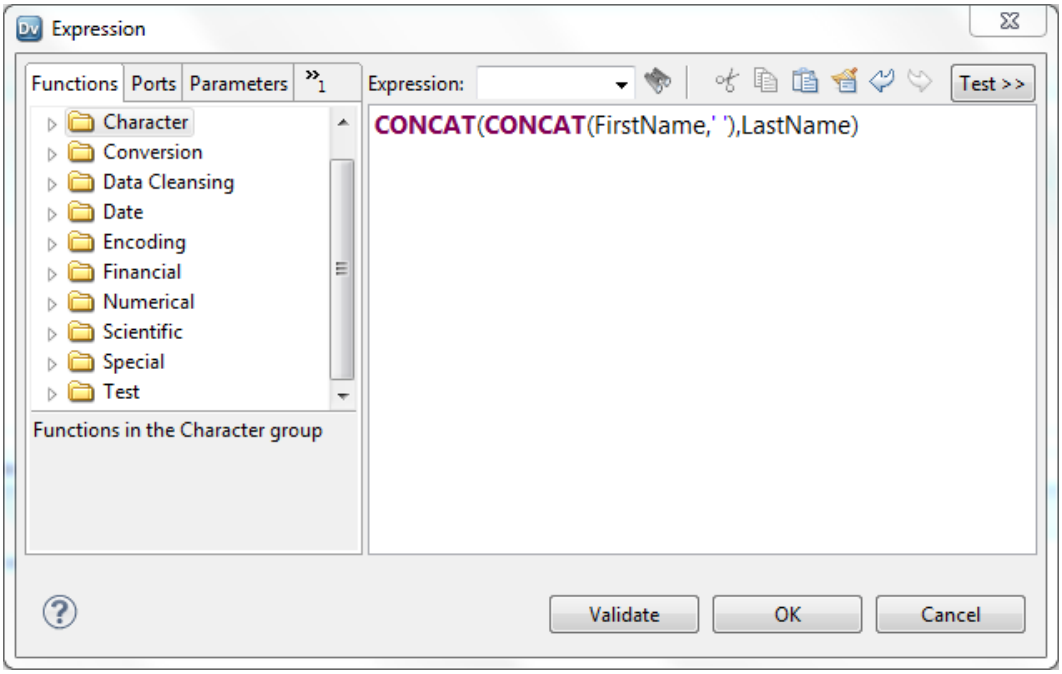
- Ausdrucksumwandlung - Übersicht, 241
- Ausdrucksumwandlungsports, 242
- Testausdrücke, 243
- Portauswahlen, 244
- Dynamische Ausdrücke, 248
- Ausdrucksumwandlung - Erweiterte Eigenschaften, 252

Ausdrucksumwandlung - Übersicht

Die Ausdrucksumwandlung ist eine passive Umwandlung, die Sie zur Durchführung von Berechnungen oder zum Testen von bedingten Anweisungen in einer Zeile verwenden können. In nicht wiederverwendbaren Ausdrucksumwandlungen können Sie beim Definieren von Mapping-Ausgaben einen zu aggregierenden Mapping-Ausgabeausdruck definieren.

In einer einzelnen Zeile müssen Sie möglicherweise einen Ausdruck zur Anpassung der Mitarbeitergehälter erstellen, um Vor- und Nachnamen zu verketten oder um Zeichenfolgen in Zahlen umzuwandeln.

Die folgende Abbildung zeigt einen Ausdruck in einer Ausdrucksumwandlung, in der der Vorname, ein Leerzeichen und der Nachname verkettet werden:



Sie können mehrere Ausdrücke in eine Ausdrucksumwandlung eingeben, indem Sie einen Ausdruck für jeden Ausgabeport erstellen. Sie möchten beispielsweise bei jedem Mitarbeitergehaltsscheck verschiedene Steuerarten berechnen, wie z. B. die Einkommensteuer auf Lokal- und Bundesebene. Beide Steuerberechnungen erfordern das Gehalt des Mitarbeiters und einen Steuersatz. Definieren Sie einen separaten Ausgabeport für jede Berechnung. Definieren Sie für jeden Ausgabeport einen anderen Ausdruck. Sie können Pass-Through-Ports für das Gehalt und den Steuersatz definieren, da sich die Portwerte nicht ändern.

Ausdrucksumwandlungsports

Eine Ausdrucksumwandlung verfügt über unterschiedliche Porttypen, auf die Sie beim Definieren von Ausdrücken verweisen können.

Eine Ausdrucksumwandlung weist die folgenden Porttypen auf:

Eingabe

Empfängt Daten aus vorangestellten Umwandlungen. Wenn die Ausdrucksumwandlung den Portwert nicht ändert, können Sie einen Pass-Through-Port anstelle eines Eingabeports definieren.

Ausgabe

Enthält den Rückgabewert des Ausdrucks. Sie geben Sie den Ausdruck als eine Konfigurationsoption für den Ausgabeport ein. Sie können auch einen Standardwert für jeden Port konfigurieren.

Pass-Through

Definieren Sie einen Pass-Through-Port, um die Daten über die Umwandlung ohne Änderung des Werts weiterzuleiten. Sie können in einer Berechnung auf einen Pass-Through-Port verweisen, aber Sie können den Datenwert im Pass-Through-Port nicht ändern.

Variable

Speichert Daten temporär zur Verwendung in Ausdrücken. Sie können Daten über mehrere Zeilen speichern. Sie können einen Ausdruck so definieren, dass ein Wert an einen variablen Port zurückgegeben wird.

Dynamischer Port

Empfängt oder gibt Ports in einem dynamischen Mapping zurück. Ein dynamischer Port kann eine oder mehrere Spalten aus einer vorgelagerten Umwandlung empfangen und für jede Spalte einen generierten Port erstellen. Ein dynamischer Ausgabeport kann einen oder mehrere generierte Ports zurückgeben. Sie können Eingaberegeln definieren, um festzulegen, welche Spalten ein dynamischer Port empfängt. Ein dynamischer Ausgabeport kann einen Ausdruck enthalten, der mehrere Ausgabeports generiert.

Generierter Port

Ein Port, der eine einzelne Spalte in einem dynamischen Port darstellt. Die generierten Ports in der Ausdrucksumwandlung ändern sich möglicherweise basierend auf den Spalten, die die Ausdrucksumwandlung von einer vorgelagerten Umwandlung empfängt.

Testausdrücke

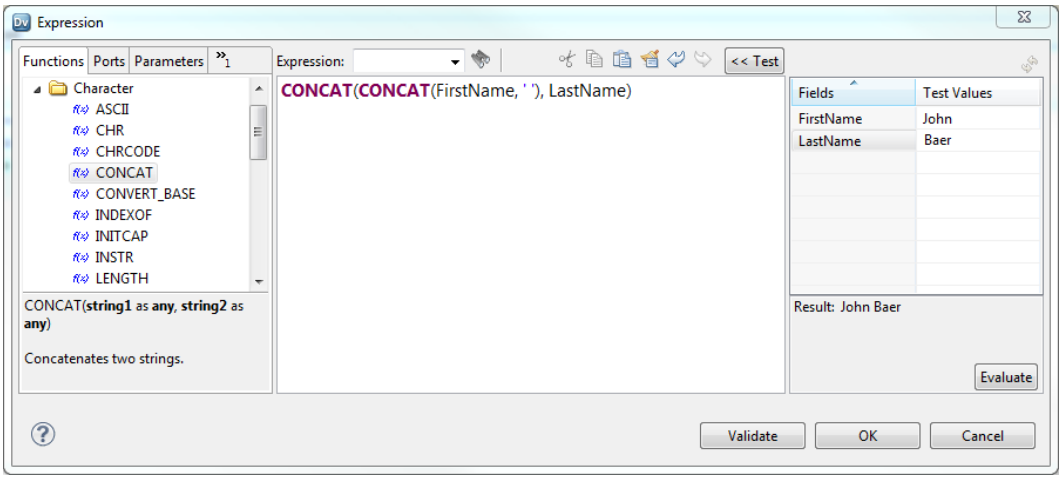
Sie können im Ausdruckseditor konfigurierte Ausdrücke testen. Wenn Sie einen Ausdruck testen, geben Sie Beispieldaten ein und werten den Ausdruck anschließend aus.

Sie können Ausdrücke testen, wenn Sie die Ausdrücke wie folgt konfigurieren:

- In einem Ausgabeport oder variablen Port in der Ausdrucksumwandlung
- In der Ansicht „Mapping-Ausgaben“ einer Ausdrucksumwandlung, nachdem die Umwandlung zu einem Mapping hinzugefügt wurde

Beispiel: Nach dem Konfigurieren eines Ausdrucks, der den Vornamen, ein Leerzeichen und den Nachnamen verkettet, können Sie Beispieldaten für die Ports eingeben und den Ausdruck anschließend auswerten, um das Ergebnis zu überprüfen.

Die folgende Abbildung zeigt die Ergebnisse eines Ausdrucks, bei dem ein Beispielvorname und -nachname verkettet werden:



Datumsformatzeichenfolgen für Beispieldaten

Wenn Sie einen Ausdruck testen, der einen Port des Datentyps „Datum/Uhrzeit“ oder „Zeitstempel mit Zeitzone“ verwendet, müssen Sie Beispieldaten für den Port mit der erforderlichen Zeichenfolge für das Datumsformat eingeben.

Für die Eingabe von Beispieldaten für einen Port des Datentyps „Datum/Uhrzeit“ verwenden Sie das Format MM/DD/YYYY HH24:MI:SS. Wenn Sie den Ausdruck auswerten, zeigt der Ausdruckseditor das Ergebnis im von Ihnen im Ausdruck angegebenen Format an. Wenn Sie die Formatzeichenfolge im Ausdruck weglassen, zeigt der Ausdruckseditor das Ergebnis in demselben Format MM/DD/YYYY HH24:MI:SS an.

Für die Eingabe von Beispieldaten für einen Port des Datentyps „Zeitstempel mit Zeitzone“ verwenden Sie das Format MM/DD/YYYY HH24:MI:SS TZR. Wenn Sie den Ausdruck auswerten, zeigt der Ausdruckseditor das Ergebnis im Format YYYY-MM-DD HH24:MI:SS.NS TZR an.

Testen eines Ausdrucks

Testen Sie einen Ausdruck im Ausdruckseditor, um den Ausdruck auszuwerten und das Ergebnis zu überprüfen.

1. Öffnen Sie den Ausdruckseditor auf eine der folgenden Arten:
   - Klicken Sie in einer Ausdrucksumwandlung auf die Schaltfläche **Öffnen** ( ) in der Spalte **Ausdruck** für einen Ausgabeport oder einen variablen Port.
   - Wählen Sie eine in einem Mapping enthaltene Ausdrucksumwandlung aus. Klicken Sie in der Ansicht **Mapping-Ausgaben** auf die Schaltfläche **Öffnen** ( ) in der Spalte **Ausdruck** für eine Ausgabe.
2. Konfigurieren Sie den Ausdruck.
3. Klicken Sie zum Öffnen des Testbereichs auf **Test >>**.
4. Geben Sie in der Spalte **Testwerte** Beispieldaten für jedes Feld ein.

   Sie können Testwerte für jeden in dem Ausdruck enthaltenen Port oder Parameter eingeben.
5. Klicken Sie auf **Überprüfen**.

   Das Ausdrucksergebnis wird unten im Testbereich angezeigt.

Portauswahlen

Wenn eine Umwandlung über generierte Ports verfügt, müssen Sie die Umwandlung so konfigurieren, dass sie bei Änderung der generierten Ports erfolgreich ausgeführt wird. Mithilfe eines Ports können Sie bestimmen, welche Ports in einem dynamischen Ausdruck, einer Lookup-Bedingung oder einer Joiner-Bedingung verwendet werden sollen.

Bei einer Portauswahl handelt es sich um eine sortierte Liste mit Ports, auf die Sie in einem Ausdruck verweisen können. Wenn sich die erzeugten Ports in einer dynamischen Zuordnung ändern, kann die Portauswahl unterschiedliche Ports enthalten.

Der folgende Ausdruck verweist beispielsweise auf einen erzeugten Port in einer dynamischen Zuordnung:

```
Salary * 12
```

Sie konfigurieren die Zuordnung so, dass dynamische Quellen verwendet werden. Die Spalte mit den Gehaltsinformationen weist jedoch in jeder Quelldatei einen anderen Namen auf. Die Spaltennamen lauten `Salary`, `Monthly_Salary` oder `Base_Salary`.

Sie führen die folgenden Aufgaben durch, um die verschiedenen Spaltennamen zu berücksichtigen:

1. Erstellen Sie eine Portauswahl mit der Bezeichnung „Salary_PortSelector“.
2. Erstellen Sie eine Auswahlregel, um alle Portnamen mit dem Suffix „Salary“ zur berücksichtigen.
3. Ändern Sie den Ausdruck so, dass der Name der Portauswahl anstelle des Spaltennamens „Salary“ eingefügt wird. Der Ausdruck verfügt über die folgende Syntax:

```
Salary_PortSelector * 12
```

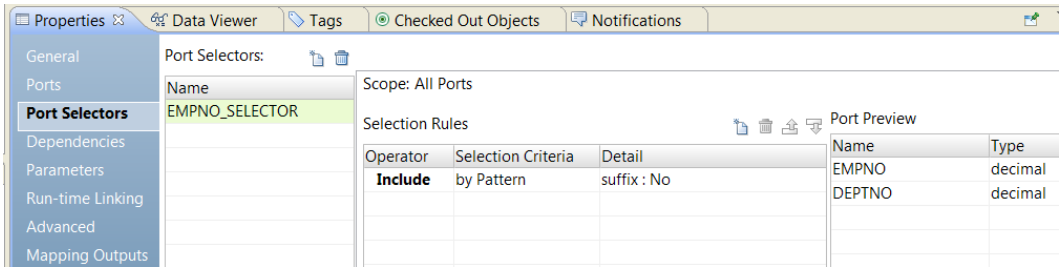
Der Ausdruck wird für jeden Gehaltsportnamen erfolgreich ausgeführt.

Portauswahl-Konfiguration

Wenn Sie eine Portauswahl konfigurieren, definieren Sie Auswahlregeln, um zu bestimmen, welche generierten Ports einbezogen werden sollen. Die Auswahlregeln ähneln den Eingaberegeln, die Sie für dynamische Ports konfigurieren können.

Eine Portauswahl kann statische oder generierte Ports enthalten. Konfigurieren Sie eine Portauswahl auf der Registerkarte **Portauswahl**.

Die folgende Abbildung zeigt die Registerkarte **Portauswahl**:



Konfigurieren Sie die folgenden Eigenschaften für eine Portauswahl:

Name

Gibt die Portauswahl an. Sie können mehrere Portauswahlen in einer Umwandlung erstellen und in Ausdrücken auf sie verweisen.

Bereich

Gibt eine Gruppe von Ports an, auf die die Portauswahl angewendet wird. Sie müssen den Bereich auswählen, wenn Sie eine Portauswahl für eine Joiner- oder Lookup-Umwandlung erstellen. Diese Umwandlungen verfügen über mehrere Eingabegruppen. Die Joiner-Umwandlung weist einen Master- oder einen Detail-Bereich auf. Die Lookup-Umwandlung weist einen Import- oder einen Lookup-Bereich auf. Die Ausdrucksumwandlung verfügt über eine Eingabegruppe. Der Bereich ist immer „Alle Ports“.

Auswahlregeln

Bestimmt die Ports, die in die Portauswahl einbezogen werden sollen. Beim Erstellen der Auswahlregeln werden im Bereich **Portvorschau** die Ports angezeigt, die sich aus den aktuellen Eingabeports qualifizieren. Diese Ports ändern sich möglicherweise. Konfigurieren Sie die Auswahlregeln, um Ports aus unterschiedlichen Quellen anzupassen.

Auswahlregeln

Die mit einer Portauswahl verbundenen Auswahlregeln bestimmen die Ports, die in die Portauswahl aufgenommen werden.

Beim Erstellen der Auswahlregeln werden im Bereich **Portvorschau** die Ports angezeigt, die sich aus den aktuellen Eingabeports qualifizieren. Diese Ports ändern sich möglicherweise. Konfigurieren Sie die Auswahlregeln, um Ports aus unterschiedlichen Quellen anzupassen.

Erstellen Sie Auswahlregeln auf Grundlage der folgenden Kriterien:

Operator

Schließt die Ports ein bzw. aus, die Auswahlregeln zurückgeben. Standardwert ist einschließen. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.

Auswahlkriterien

Der Auswahlregeltyp, den Sie erstellen möchten. Sie können eine Regel basierend auf dem Porttyp oder dem Spaltennamen erstellen. Um Ports basierend auf dem Spaltennamen einzuschließen, suchen Sie nach bestimmten Namen oder nach einem Muster von Zeichen im Namen.

Detail

Die auf die Auswahlkriterien anzuwendenden Werte. Wenn die Auswahlkriterien nach Spaltenname erfolgen, konfigurieren Sie die Zeichenfolge oder den Namen, nach der bzw. dem gesucht werden soll. Wenn die Auswahlkriterien nach Porttyp erfolgen, wählen Sie die einzuschließenden Porttypen aus.

In der folgenden Tabelle werden die Auswahlkriterien beschrieben und wie die Details für die Kriterien angegeben werden müssen:

Auswahlkriterien	Beschreibung	Detail
Alle	Schließt alle Ports ein.	Keine Details erforderlich.
Name	Filtert Ports basierend auf dem Portnamen.	Wählen Sie die Portnamen aus einer Liste mit Werten aus oder verwenden Sie einen Parameter des Typs „Port“ oder „Portliste“.
Typ	Filtert Ports basierend auf dem Datentyp jedes Ports.	Wählen Sie Datentypen aus einer Liste aus.
Muster	Filtert Ports nach einer Zeichenfolge von Zeichen im Namen oder nach einem regulären Ausdruck.	Wählen Sie ein Präfix, ein Suffix oder einen regulären Ausdruck als Mustertyp für den Portnamen aus. Geben Sie anschließend einen Wert für das Muster ein oder verwenden Sie einen Parameter des Typs „Zeichenfolge“.

Erstellen einer Portauswahl

Erstellen Sie einen Port zum Festlegen, welche Ports in einem dynamischen Ausdruck, einer Lookup-Bedingung oder einer Joiner-Bedingung verwendet werden sollen.

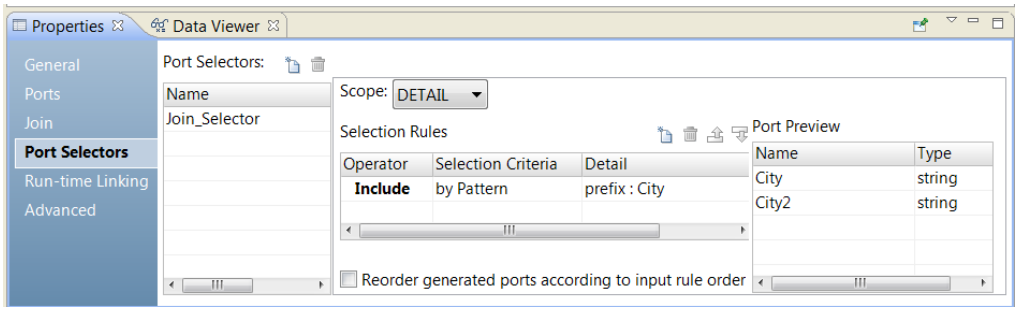
1. Klicken Sie auf die Registerkarte **Portauswahlen**.
2. Klicken Sie im Bereich **Portauswahlen** auf **Neu**.

   Das Developer Tool erstellt eine Portauswahl mit einer Standardauswahlregel, die alle Ports enthält.
3. Ändern Sie im Bereich **Portauswahlen** den Namen der Portauswahl in einen eindeutigen Namen.
4. Wenn Sie die Joiner-Umwandlung oder die Lookup-Umwandlung bearbeiten, wählen Sie den Bereich aus.

Die verfügbaren Ports ändern sich basierend auf der Gruppe der von Ihnen ausgewählten Ports.

5. Wählen Sie im Bereich **Auswahlregeln** einen **Operator** aus.
   - Einschließen. Erstellen Sie eine Regel, die Ports für die Portauswahl einschließt. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.
   - Ausschließen. Erstellen Sie eine Regel, die bestimmte Ports aus der Portauswahl ausschließt.
6. Wählen Sie die **Auswahlkriterien** aus.
   - Nach Name. Wählen Sie bestimmte Ports nach Namen aus. Sie können die Portnamen aus einer Liste mit Ports im Bereich auswählen.
   - Nach Typ. Wählen Sie Ports nach Typ aus. Sie können eine oder mehrere Datentypen auswählen.
   - Nach Muster. Wählen Sie Ports nach einem Muster von Zeichen im Portnamen aus. Sie können die Suche mit bestimmten Zeichen durchführen oder einen regulären Ausdruck erstellen.

   Die folgende Abbildung zeigt die Registerkarte „Portauswahl“:



7. Klicken Sie auf die Spalte **Detail**.

   Das Dialogfeld **Eingaberegel-Detail** wird angezeigt.
8. Wählen Sie die Werte aus, nach denen Ports gefiltert werden sollen.
   - Nach Name. Wählen Sie, ob eine Portliste nach Wert oder nach einem Parameter erstellt werden soll. Klicken Sie auf **Auswählen**, um die Ports auf der Liste auszuwählen.
   - Nach Typ. Wählen Sie mindestens einen Datentyp aus einer Liste aus. Im Bereich **Portvorschau** werden Ports des von Ihnen ausgewählten Typs angezeigt.
   - Nach Muster. Wählen Sie, ob das Präfix oder das Suffix des Portnamens nach einem bestimmten Muster von Zeichen durchsucht werden soll. Oder entscheiden Sie sich für das Erstellen eines regulären Ausdrucks für die Suche. Konfigurieren Sie einen Parameter oder konfigurieren Sie das Muster für die Suche.

   Im Bereich **Portvorschau** werden während der Konfiguration der Regeln die Ports in der Portauswahl angezeigt.
9. Um die Ports in der Portauswahl neu anzuordnen, wählen Sie **Erzeugte Ports gemäß der Reihenfolge der Eingaberegeln neu sortieren** aus.

Dynamische Ausdrücke

Wenn Sie einen Ausdruck in einem dynamischen Ausgabeport konfigurieren, wird der Ausdruck zu einem dynamischen Ausdruck. Ein dynamischer Ausdruck kann mehrere Ausgabeports generieren.

Sie können in einem dynamischen Ausdruck auf eine Portauswahl oder einen dynamischen Port verweisen. Wenn die Portauswahl bzw. der dynamische Port mehrere Ports enthält, wird der dynamische Ausdruck für jeden Port ausgeführt.

Wenn Sie einen dynamischen Ausdruck konfigurieren, validiert das Developer Tool nur dann, wenn es sich bei den generierten Ports nicht um für den Ausdruck gültige Typen handelt. Beispiel: Wenn Sie auf eine Portauswahl mit Ports des Typs „Dezimal“ in einem Ausdruck verweisen, der Zeichenfolgentypen erfordert, wird der Ausdruck zur Entwurfszeit als gültig angezeigt.

Beispiel

Eine Ausdrucksumwandlung verfügt über die folgenden generierten Eingabeports:

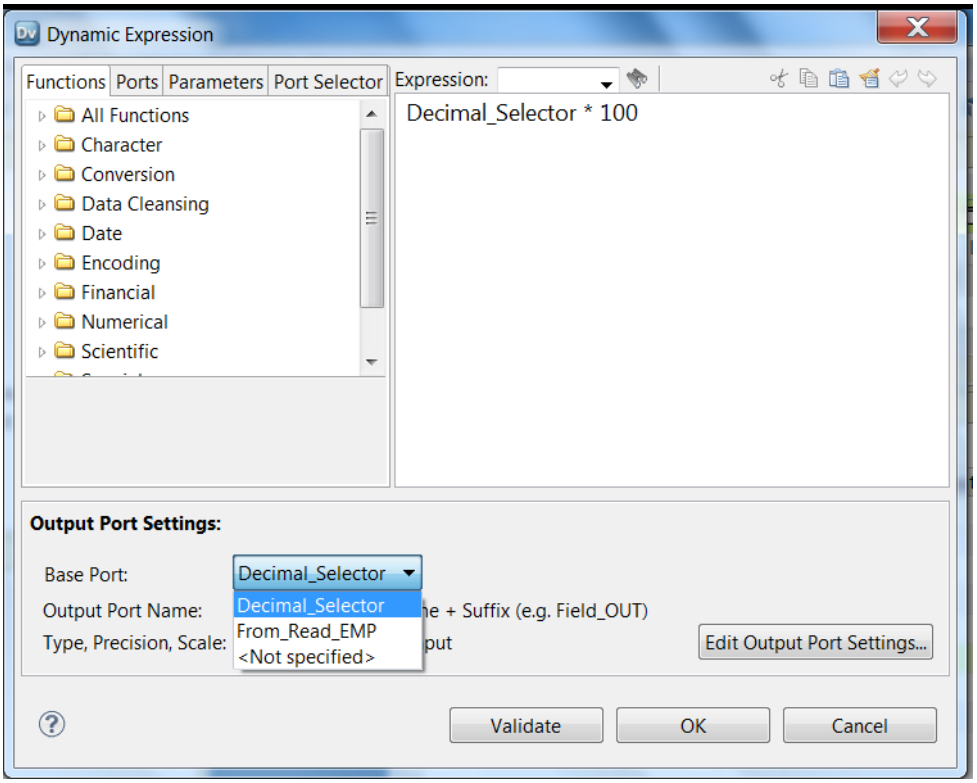
```
EMPNO   Decimal
NAME    String
SALARY  Decimal
DEPTNO  Decimal
```

Die Umwandlung enthält einen dynamischen Ausgabeport namens „MyDynamicPort“. Der Ausgabeport gibt die Ergebnisse eines dynamischen Ausdrucks zurück. Der dynamische Ausdruck multipliziert den Wert von jedem Port in einer Portauswahl mit 100. Der Ausdruck wird für jeden Port in der Portauswahl einmal ausgeführt. Jede Instanz kann ein anderes Ergebnis zurückgeben. Die Ausdrucksumwandlung generiert einen separaten Ausgabeport für jedes Ergebnis.

Die Portauswahl „Decimal_Selector“ verfügt über eine Auswahlregel, welche die Ports einschließt, die den Datentyp „Dezimal“ aufweisen:

```
EMPNO   Decimal
SALARY  Decimal
DEPTNO  Decimal
```

Die folgende Abbildung zeigt einen dynamischen Ausdruck, der auf die Portauswahl „Decimal_Selector“ verweist:



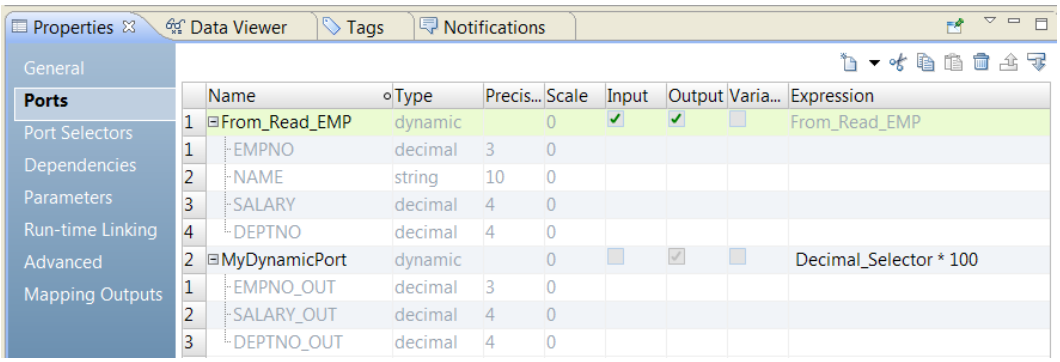
Bearbeiten Sie die Einstellungen für den Ausgabeport, um Namen und Eigenschaften des Ausgabeports zu ändern. Sie können zudem den Basisport auswählen.

Einstellungen für Ausgabeport

Sie können angeben, welche Ports Sie als Eingabe für einen dynamischen Ausdruck verwenden möchten. Wählen Sie die Ports im Bereich **Basisport** aus.

Wenn Sie die Portauswahl Decimal_Selector als Basisport auswählen, gibt der dynamische Ausdruck Ports des Typs „Dezimal“ zurück. Der dynamische Ausdruck generiert keinen Port für den NAME-Port, da es sich bei diesem um den Typ „Zeichenfolge“ handelt.

In der folgenden Abbildung werden die generierten Ports in der Umwandlung angezeigt:

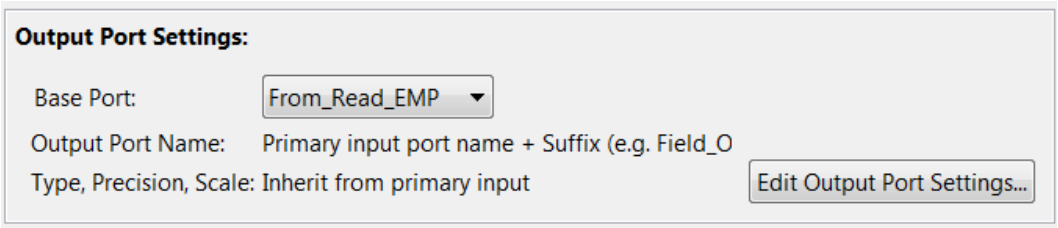


Obwohl es sich bei dem dynamischen Port „From_Read_Emp“ um einen Eingabe-/Ausgabeport handelt, gibt die Umwandlung nur die Ports im dynamischen Ausgabeport „MyDynamicPort“ zurück.

Sie können konfigurieren, wie die Ausgabeports benannt werden sollen. Der Standardname des Ausgabeports besteht aus dem Namen des Eingabeports und dem Suffix `_OUT`.

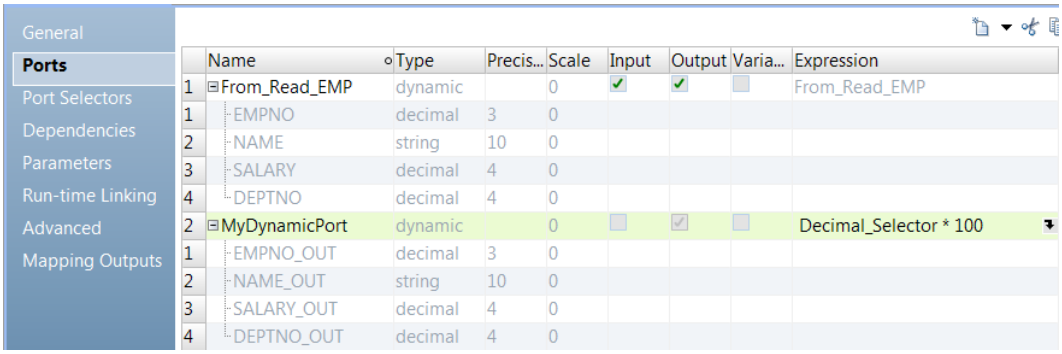
Sie können den Basisport in eine Portauswahl ändern.

Die folgende Abbildung zeigt die Einstellungen für den Ausgabeport im Ausdruckseditor:



Wenn Sie den Basisport als From_Read_EMP konfigurieren, wählen Sie den dynamischen Port aus, der alle generierten Eingabeports enthält. Der Datenintegrationsdienst führt den dynamischen Ausdruck für alle Ports in From_Read_EMP aus.

Die folgende Abbildung zeigt die generierten Ausgabeports basierend auf der From_Read_Emp-Eingabe:



Die generierten Ausgabeports enthalten einen Ausgabeport namens NAME_OUT, welcher vom Typ „Zeichenfolge“ ist.

Der Datenintegrationsdienst generiert Ausgabeports für jeden dynamischen Ausdruck. Wenn Sie einen dynamischen Ausdruck erstellen, bei dem 15 Ports generiert werden, und einen anderen dynamischen Ausdruck definieren, bei dem 5 Ports generiert werden, generiert der Datenintegrationsdienst 20 Ausgabeports. Jeder dynamische Ausgabeport generiert eine andere Gruppe von Ports.

Erstellen eines dynamischen Ausdrucks

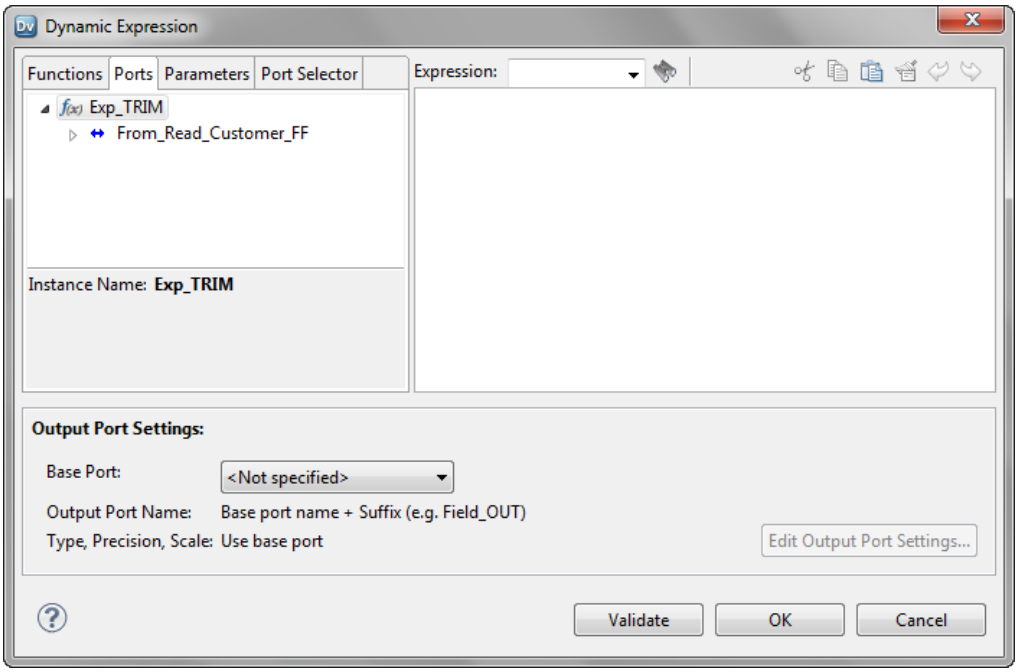
Erstellen Sie einen dynamischen Ausdruck in einer Ausdrucksumwandlung, um den Ausdruck einmal für jeden Port in einem dynamischen Port oder in einer Portauswahl auszuführen. Der dynamische Ausdruck gibt die Ergebnisse an einen separat generierten Port für jede Instanz zurück.

1. Navigieren Sie in der Ausdrucksumwandlung zur Ansicht **Eigenschaften** und klicken Sie auf die Registerkarte **Ports**.
2. Klicken Sie auf **Neuer dynamischer Port**.

   Das Developer Tool erstellt einen dynamischen Port mit Standardeigenschaften.
3. Benennen Sie den dynamischen Port um und deaktivieren Sie die Eingabeoption.

   Der dynamische Port muss ein Ausgabeport sein.
4. Klicken Sie in der Spalte **Ausdruck** für den dynamischen Ausgabeport auf die Schaltfläche **Öffnen** ( ).

Das Dialogfeld **Dynamischer Ausdruck** wird angezeigt:



5. Geben Sie im Ausdrucksseditor einen Ausdruck ein. Der Ausdruck kann eine Portauswahl oder einen dynamischen Port enthalten.

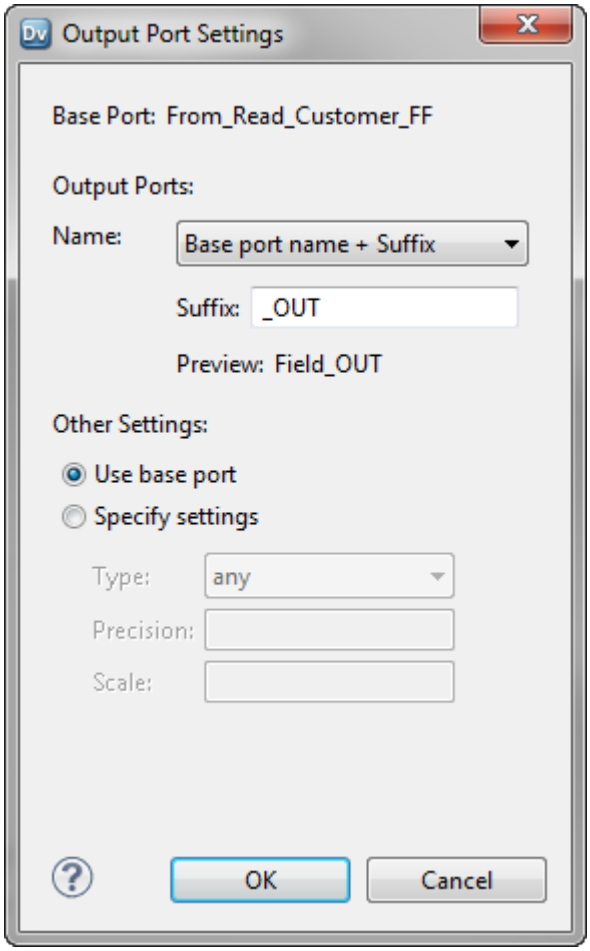
   Beispiel: `LTRIM(RTRIM(Dynamic_Customer))`, wobei Dynamic_Customer ein dynamischer Port ist.

6. Klicken Sie auf **Validieren**, um den Ausdruck zu validieren.
7. Klicken Sie auf **OK**, um das Dialogfeld **Ausdruck validieren** zu schließen.
8. Wählen Sie im Bereich **Einstellungen für Ausgabeport** den dynamischen Ausgabeport aus der Liste **Basisport** aus oder wählen Sie eine Portauswahl, auf die Sie im Ausdruck verwiesen haben.

   Das Developer Tool generiert Ausgabeports basierend auf Ihrer Auswahl.

9. Führen Sie die folgenden Schritte aus, um die Ausgabeports umzubenennen:
   a. Klicken Sie auf **Einstellungen für Ausgabeport bearbeiten**.

      Das Dialogfeld **Einstellungen für Ausgabeport** wird angezeigt.



   b. Wählen Sie auf der Liste **Name** eine der Optionen aus und geben Sie einen Wert für das Präfix oder Suffix ein. Wenn Sie **Feste Zeichenfolge + Auto-Nummerierung** ausgewählt haben, geben Sie den Text für den Ausgangsportnamen ein. Wenn Sie beispielsweise TRIM als Ausgangsportname eingeben, werden die Namen der Ausgangsports als TRIM1, TRIM2, TRIM3 angezeigt.
   c. Optional wählen Sie im Bereich **Andere Einstellungen** die Option **Einstellungen angeben**, um den Typ, die Gesamtstellenanzahl und die Dezimalstellen für die Ausgabeports zu ändern. Standardmäßig werden für die Ausgabeports die Einstellungen der Basisports verwendet.
   d. Klicken Sie auf **OK**.
10. Klicken Sie auf **OK**, um den Editor **Dynamischer Ausdruck** zu schließen.

Ausdrucksumwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Datenintegrationsdienst Daten für die Ausdrucksumwandlung verarbeitet.

Konfigurieren Sie die folgenden erweiterten Eigenschaften für eine Ausdrucksumwandlung:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Beibehalten der Zeilenreihenfolge

Beibehalten der Zeilenreihenfolge der Eingabedaten für die Umwandlung. Wählen Sie diese Option aus, wenn der Datenintegrationsdienst keine Optimierung durchführen soll, die die Zeilenreihenfolge verändern könnte.

Wenn der Datenintegrationsdienst Optimierungen durchführt, geht eine zu einem vorherigen Zeitpunkt im Mapping erstellte Reihenfolge möglicherweise verloren. Die Reihenfolge in einem Mapping können Sie mit einer Einfachdateiquelle, einer sortierten relationalen Quelle oder einer Sortierer-Umwandlung erstellen. Wenn Sie eine Umwandlung konfigurieren, um die Zeilenreihenfolge beizubehalten, berücksichtigt der Datenintegrationsdienst diese Konfiguration beim Durchführen von Optimierungen für das Mapping. Der Datenintegrationsdienst führt Optimierungen für die Umwandlung durch, wenn er die Reihenfolge beibehalten kann. Der Datenintegrationsdienst führt keine Optimierungen für die Umwandlung durch, wenn die Optimierungen die Zeilenreihenfolge verändern würden.

KAPITEL 15

Filterumwandlung

Dieses Kapitel umfasst die folgenden Themen:

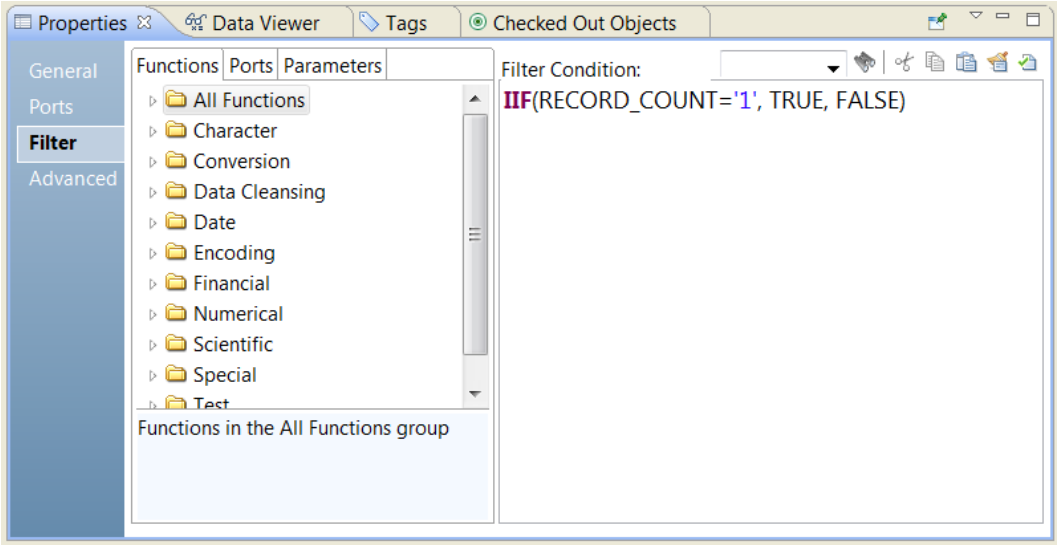
- Filterumwandlung - Übersicht, 254
- Filterumwandlungen in dynamischen Mappings, 255
- Filterbedingung, 255
- Filterumwandlung - Erweiterte Eigenschaften, 258
- Filterumwandlung - Tipps für die Leistung, 258

Filterumwandlung - Übersicht

Verwenden Sie die Filterumwandlung, um Zeilen aus einem Mapping herauszufiltern. Als aktive Umwandlung kann die Filterumwandlung die Anzahl der sie durchlaufenden Zeilen ändern.

Die Filterumwandlung ermöglicht die Übergabe von Zeilen, die die angegebenen Filterbedingung erfüllen. Sie löscht Zeilen, die diese Bedingung nicht erfüllen. Sie können Daten basierend auf einer oder mehreren Bedingungen filtern.

Die folgende Abbildung zeigt eine Filterbedingung in einer Filterumwandlung:



Eine Filterbedingung gibt TRUE oder FALSE für jede Zeile zurück, die der Datenintegrationsdienst basierend darauf bewertet, ob eine Zeile die angegebene Bedingung erfüllt. Für jede Zeile, die TRUE zurückgibt, übergibt der Datenintegrationsdient die Umwandlung. Für jede Zeile, die FALSE zurückgibt, fügt der Datenintegrationsdienst eine Meldung im Protokoll ein.

Sie können keine Ports aus mehr als einer Umwandlung in der Filterumwandlung verketten. Die Eingabeports für den Filter müssen aus einer einzelnen Umwandlung stammen.

Filterumwandlungen in dynamischen Mappings

Sie können eine Filterumwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports in der Filterbedingung verweisen.

Sie können die vollständige Filterbedingung parametrisieren. Konfigurieren Sie einen Ausdrucksparameter mit einem Standardwert, der den gesamten Ausdruck enthält. Das Developer Tool validiert keine Filterbedingung in einem Parameterstandardwert.

Sie können in einer Filterbedingung auf einen dynamischen Port verweisen. Der dynamische Port kann mehrere generierte Ports enthalten. Der Datenintegrationsdienst erweitert die Filterbedingung so, dass jeder generierte Port eingeschlossen ist. Jeder generierte Port muss einen gültigen Typ aufweisen, um in den Ausdruck eingeschlossen zu werden.

Sie können in einer Filterbedingung auf einen generierten Port verweisen. Wenn die generierten Ports zur Laufzeit jedoch nicht vorhanden sind, schlägt das Mapping fehl.

Filterbedingung

Die Filterbedingung ist ein Ausdruck, der TRUE oder FALSE zurückgibt.

Geben Sie Bedingungen im Ausdrucksseditor ein. Bei der Filterbedingung wird zwischen Groß- und Kleinschreibung unterschieden.

Sie können keinen Ausdruck verwenden, der einen einzelnen Wert als Filter zurückgibt. Wenn Sie beispielweise Zeilen für Mitarbeiter herausfiltern möchten, deren Gehalt unter oder bei 30.000 US-Dollar liegt, geben Sie die folgende Bedingung ein:

```
SALARY > 30000
```

Sie können mehrere Komponenten der Bedingung mit den logische Operatoren AND und OR angeben. Wenn Sie Mitarbeiter herausfiltern möchten, deren Gehalt unter 30.000 und über 100.000 US-Dollar liegt, geben Sie die folgende Bedingung ein:

```
SALARY > 30000 AND SALARY < 100000
```

Sie können Ports, Parameter, dynamische Ports und generierte Ports in der Filterbedingung verwenden. Wählen Sie die Ports und die Parameter im Ausdrucksseditor aus.

Wenn Sie einen dynamischen Port in der Filterbedingung verwenden, wird die Filterbedingung erweitert, um alle generierten Ports in den dynamischen Port einzubeziehen. Beispiel: Der dynamische Port „MyDynamicPort“ enthält drei Dezimal-Ports:

```
Salary
Bonus
Stock
```

Wenn Sie die folgende Filterbedingung konfigurieren:

```
MyDynamicPort > 100
```

Wird die Filterbedingung um den folgenden Ausdruck erweitert:

```
Salary > 100 AND Bonus > 100 AND Stock > 100
```

Sie können eine Konstante für die Filterbedingung eingeben. Das numerische Äquivalent von FALSE ist Null (0). Alle Nicht-Nullwerte sind das Äquivalent zu TRUE. Die Umwandlung enthält zum Beispiel einen Port namens NUMBER_OF_UNITS mit einem numerischen Datentyp. Konfigurieren Sie eine Filterbedingung, um FALSE zurückzugeben, wenn der Wert NUMBER_OF_UNITS gleich Null ist. Andernfalls gibt die Bedingung TRUE zurück.

Hinweis: Sie können keine einzelne Portauswahl und keinen dynamischen Port als einen booleschen Wert verwenden.

Sie brauchen TRUE oder FALSE nicht als Werte in einem Ausdruck angeben. TRUE und FALSE sind implizite Rückgabewerte aus einer beliebigen von Ihnen festgelegten Bedingung. Wenn die Filterbedingung mit NULL bewertet wird, ist die Zeile FALSE.

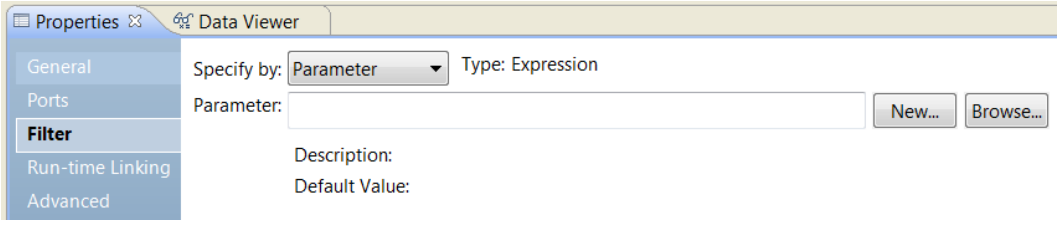
Die Filterbedingung parametrisieren

Sie können einen Ausdrucksparameter zum Definieren der Filterbedingung konfigurieren. Ein Ausdrucksparameter enthält den gesamten Ausdruck.

Möglicherweise müssen Sie die Filterbedingung parametrisieren, wenn sich die Filterumwandlung in einem dynamischen Mapping befindet. Die Filterbedingung ändert sich möglicherweise basierend auf den generierten Ports in der Umwandlung zur Laufzeit.

Wählen Sie zur Verwendung eines Ausdrucksparameters für die Filterbedingung die Option **Angeben nach Parameter** auf der Registerkarte **Filter** der Filterumwandlungseigenschaften aus.

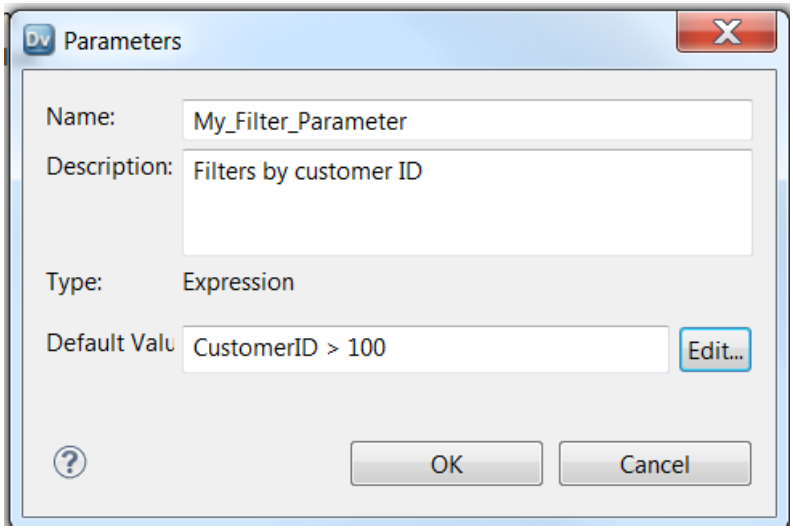
Die folgende Abbildung zeigt die Registerkarte **Filter**, wenn Sie die Filterbedingung mit einem Parameter angeben:



Sie können einen bereits erstellten Ausdrucksparameter suchen und auswählen. Oder Sie können einen Ausdrucksparameter erstellen.

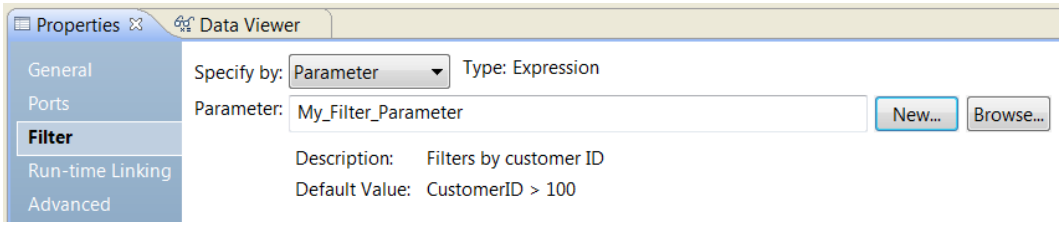
Klicken Sie zum Erstellen eines Ausdrucksparameters auf **Neu**. Geben Sie einen Namen für den Parameter sowie eine Beschreibung und den Standardausdruckswert ein.

Die folgende Abbildung zeigt, wo Sie die Parameter eingeben:



Sie können im Dialogfeld „Parameter“ einen Standardausdruck eingeben. Wenn Sie einen Ausdrucksеditor verwenden möchten, klicken Sie auf „Bearbeiten“. Wenn Sie den Ausdrucksеditor verwenden, können Sie Funktionen und Ports auswählen, die in dem Ausdruck verwendet werden sollen. Sie können den Ausdruck validieren.

Die folgende Abbildung zeigt die Registerkarte „Filter“ mit einem Parameter für die Filterbedingung:



Der Ausdrucksparameter ist ein Mapping-Parameter. Sie können den Parameter in einem Parametersatz oder in einer Parameterdatei zur Laufzeit überschreiben.

Filtern von Zeilen mit Nullwerten

Um Zeilen mit Nullwerten oder Leerzeichen zu filtern, verwenden Sie die Funktionen ISNULL und IS_SPACES, um den Wert des Ports zu testen.

Wenn Sie Zeilen mit einem Nullwert im FIRST_NAME-Port herausfiltern möchten, verwenden Sie die folgende Bedingung:

```
IIF(ISNULL(FIRST_NAME),FALSE,TRUE)
```

Wenn der FIRST_NAME-Port dieser Bedingung NULL ist, ist der Rückgabewert FALSE und die Zeile sollte verworfen werden. Andernfalls wird die Zeile an die nächste Umwandlung übergeben.

Filterumwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie bestimmen können, wie der Data Integration Service Daten für die Filterumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Filterumwandlung - Tipps für die Leistung

Verwenden Sie Tipps, um die Leistung der Filterumwandlung zu erhöhen.

Verwenden Sie die Filterumwandlung am Anfang des Mappings.

Bleiben Sie mit der Filterumwandlung so nah wie möglich an den Quellen im Mapping. Leiten Sie die Zeilen, die Sie löschen möchten, nicht durch das Mapping, sondern filtern Sie unerwünschte Daten zu einem frühen Zeitpunkt des Datenflusses von den Quellen zu den Targets heraus.

KAPITEL 16

Umwandlung von hierarchisch in relational

Dieses Kapitel umfasst die folgenden Themen:

- Umwandlung von hierarchisch in relational – Übersicht, 259
- Beispiel - Hierarchische-in-relational-Umwandlung, 259
- Relationale Ausgabeports und die Ansicht „Übersicht“, 261
- Umwandlungsports von hierarchisch in relational, 262
- Schemaverweise, 263
- Port-Konfiguration, 263
- Entwicklung der Umwandlung von hierarchisch in relational, 264

Umwandlung von hierarchisch in relational – Übersicht

Die Hierarchisch-in-relational-Umwandlung verarbeitet hierarchische XML- oder JSON-Eingaben und wandelt sie in relationale Ausgaben um. Bei einer Umwandlung von hierarchisch in relational werden hierarchische Eingaben aus Eingabeports gelesen und die Daten über die Umwandlungsausgabeports in relationale Ausgaben umgewandelt. Um hierarchische Eingaben in relationale Ausgaben umzuwandeln, verwenden Sie eine Schemadatei zum Definieren von hierarchischen Daten.

Sie können den Assistenten für die Umwandlung von hierarchisch in relational verwenden, um die Daten automatisch zuzuordnen. Sie können das Mapping über die relationalen Ausgabeports in der Ansicht **Übersicht** der Umwandlung konfigurieren.

Sobald der Assistent die Umwandlung generiert hat, können Sie die Daten aus den relationalen Ausgabeports an eine andere Umwandlung in einem Mapping übergeben.

Beispiel - Hierarchische-in-relational-Umwandlung

Die Logistikabteilung des Unternehmens Harrinder Shipping muss Lieferdaten verarbeiten. Bestands- und Kundendaten müssen aus dem hierarchischen Format in relationale Daten umgewandelt werden, die sich in Datenbanktabellen speichern lassen.

Dazu muss ein Mapping erstellt werden, mit dem hierarchische Daten in relationale Daten umgewandelt werden. Das Bestandsverwaltungssystem im Unternehmen generiert Lieferbestandsdaten in hierarchischem Format. Das Mapping muss eine Hierarchisch-in-relational-Umwandlung mit Lieferdaten als Eingabe und Details in einem verwertbaren relationalen Format als Ausgabe verwenden.

Die Lieferdateneingabe liegt in hierarchischem Format vor. Das Lieferdatenelement umfasst Unterelemente mit Kunden- und Bestandsdaten für jede Lieferung:

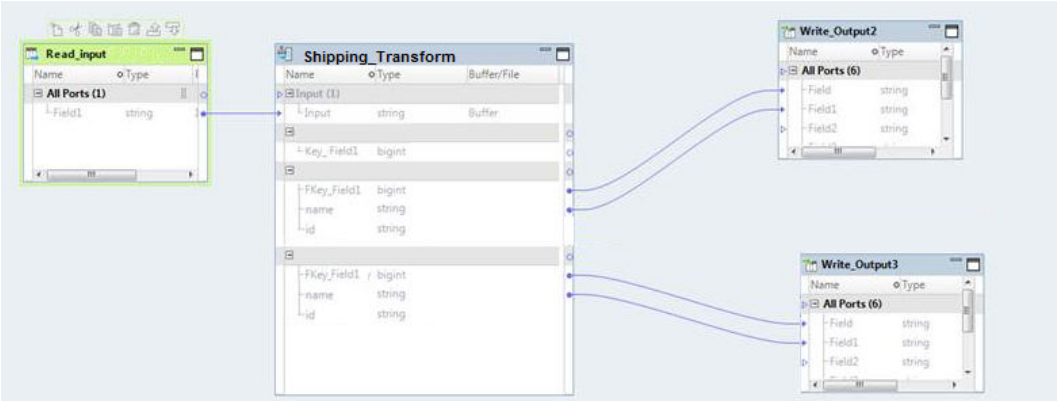
```
 Shipments
Shipment
    Items
     Item_Name
     Inventory_ID
    Customer
     Customer_Name
     Customer_ID
     Customer_Address
```

In der relationalen Ausgabe ist das Element „Customer_ID“ ein Primärschlüssel in der Kundentabelle und ein Fremdschlüssel in der Lieferungstabelle.

Customer_ID	Customer_Name	Customer_Address
3543766	Tony Birch	6 Moby Drive
6342562	Sujita Man	22 Dan Street
6471862	Dwayne Horace	7 Jafendar Boulevard
7265204	Carmela Perez	23 Dan Street
4559672	Delilah Soraya	28 Jafendar Boulevard

Shipment_ID	Inventory_Item	Customer_ID
9173327437	908274	7265204
9174562342	553439	7265204
8484526471	546584	3543766
7023847265	908274	3543766
9174596725	553439	3543766

Die folgende Abbildung verdeutlicht das Mapping in diesem Beispiel:



Das Mapping enthält die folgenden Objekte:

Read_input

Die Quelle mit dem Pfad zur Datei mit hierarchischen Daten. Liest Abrechnungsdaten aus einer XML-Datei.

Shipping_Transform

Eine Hierarchisch-in-relational-Umwandlung einer XML-Eingabe in einer relationale Ausgabe.

Write_Output2

Ein Ziel, in dem die Kundentabelle als Teil der umgewandelten Daten in relationalem Format gespeichert wird.

Write_Output3

Ein zweites Ziel, in dem die Lieferungstabelle als weiterer Teil der umgewandelten Daten in relationalem Format gespeichert wird.

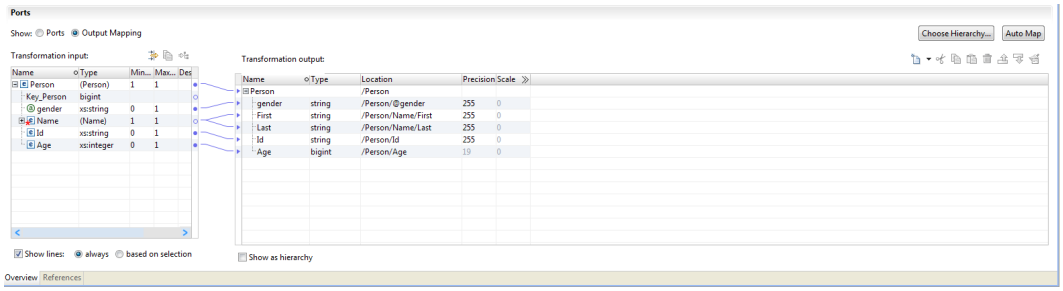
Für das Mapping wird die Einfachdatei **Read_input** zur Übernahme des Zielpfads für die hierarchische Eingabe herangezogen. Beim Mapping werden die Daten anhand der Umwandlung **Shipping_Transform** verarbeitet und umgewandelt. Danach speichert das Mapping die Ausgabe in den beiden Ausgabezielen.

Relationale Ausgabeports und die Ansicht „Übersicht“

Um mit der Hierarchisch-in-relational-Umwandlung hierarchische Daten in eine relationale Ausgabe umzuwandeln, generiert der Assistent Verknüpfungen zwischen hierarchischen Knoten und relationalen Ports. Sie können die Verknüpfungen zwischen relationalen und hierarchischen Ports in der Ansicht **Übersicht** anzeigen. Sie können auch Gruppen von Ausgabeports erstellen, indem Sie Knoten aus der hierarchischen Ausgabe mit Gruppen von Ports verknüpfen.

Verwenden Sie die Ansicht **Übersicht** zur Anzeige des Mappings von relationalen Gruppen. Wählen Sie **Mapping-Ausgabe** aus. Der Bereich **Ports** erscheint in der Ansicht **Übersicht**.

Die folgende Abbildung zeigt den Bereich **Ports**:



Auf der linken Seite finden Sie den Bereich **Umwandlungseingabe** mit dem hierarchischen Schema. Auf der rechten Seite finden Sie den Bereich **Umwandlungsausgabe** mit den relationalen Ausgabeports.

Sie können relationale Ausgabeports im Bereich **Umwandlungsausgabe** definieren und Knoten aus dem Schema mit den Ports verknüpfen. Sie können auch den Mauszeiger aus einem Knoten auf ein leeres Feld im Bereich **Umwandlungsausgabe** ziehen, um einen Port zu erstellen. Wenn Sie einen Knoten aus dem Ausgabeschema in einen Port ziehen, zeigt das Developer Tool eine Verbindung zwischen den beiden an.

Umwandlungsports von hierarchisch in relational

Die Umwandlungsports von hierarchisch in relational werden in der Ansicht **Übersicht** der Umwandlung definiert.

Eine Umwandlung von hierarchisch in relational kann Eingaben aus einer Datei oder einem Puffer lesen. Die Ausgabeports geben relationale Daten aus der Umwandlung zurück.

Wenn Sie eine Umwandlung von hierarchisch in relational erstellen, erstellt das Developer Tool einen Standardeingabeport. Der Eingabetyp bestimmt den Datentyp, der vom Datenintegrationsdienst an die Umwandlung von hierarchisch in relational übergeben wird. Der Eingabetyp legt fest, ob es sich bei der Eingabe um Daten oder einen Quelldateipfad handelt.

Konfigurieren Sie einen der folgenden Eingabetypen:

Puffer

Die Umwandlung von hierarchisch in relational empfängt Zeilen mit Quelldaten im Eingabeport. Verwenden Sie den Puffereingabetyp, wenn Sie die Umwandlung für das Empfangen von Daten von einer Informatica-Umwandlung konfigurieren.

Datei

Die Umwandlung von hierarchisch in relational empfängt den Quelldateipfad im Eingabeport. Die Umwandlung von hierarchisch in relational öffnet die Quelldatei. Sie können zur Verarbeitung mit einem Puffereingabeport auch den Datei-Eingabetyp für große Dateien verwenden, die möglicherweise viel Systemspeicher benötigen.

Beim Erstellen der Umwandlung mit dem Assistenten für neue Umwandlungen können Sie eine Beispieleingabedatei definieren. Eine Beispieleingabedatei ist ein kleines Muster für die Eingabedatei. Verweisen Sie auf eine Beispieleingabedatei, wenn Sie eine Umwandlung von hierarchisch in relational erstellen. Sie verwenden die Beispieleingabedatei auch, wenn Sie die Umwandlung in der Ansicht **Daten-Viewer** testen.

Die Umwandlung enthält eine oder mehrere Gruppen von Ports, die relationale Daten zurückgeben.

Schemaverweise

Eine Hierarchisch-in-relational-Umwandlung erfordert ein hierarchisches Schema zum Definieren der Eingabehierarchie in der Umwandlung. Zum Verwenden des Schemas in der Umwandlung definieren Sie einen Schemaverweis.

Sie können Schemaverweise für Umwandlungen in der Ansicht **Referenzen** der Umwandlung anzeigen.

Die Hierarchisch-in-relational-Umwandlung verweist auf Schemaobjekte im Modellrepository. Die Schemaobjekte können im Repository vorhanden sein, bevor Sie die Umwandlung erstellen. Sie können Schemas auch aus der Ansicht **Verweise** der Umwandlung importieren.

Ein Schema kann auf zusätzliche Schemas verweisen. Die Ansicht **Referenzen** zeigt den Namespace und das Präfix für jedes Schema an, auf das die Hierarchisch-in-relational-Umwandlung verweist. Wenn Sie mit leeren Namespaces auf mehrere Schemas verweisen, ist die Umwandlung ungültig.

Port-Konfiguration

Im Bereich **Ports** zeigt die Umwandlung das Mapping zwischen den hierarchischen Schemaknoten und den relationalen Ports an. Das Mapping verwendet ein Schema zum Definieren der hierarchischen Eingabe. Wenn das Schema mehr als ein Element aufweist, das ein Root-Element sein kann, wählen Sie einen Knoten als Root-Element aus.

Der Assistent erstellt Verknüpfungen zwischen hierarchischen Schemaknoten und relationalen Ports. Im Bereich **Ports** können Sie generierte Verknüpfungen hinzufügen, löschen oder bearbeiten. Sie können Knoten mit Ports verknüpfen und einen Port erstellen.

Wenn Sie Knoten im Bereich **Umwandlungsausgabe** verknüpfen, aktualisiert das Developer Tool das Speicherortfeld auf den Speicherort des Knotens in der Hierarchie. Wenn Sie Ports manuell erstellen, müssen Sie dem Knoten einen Code zuordnen. Aktualisieren Sie die Spalte **Speicherort** und wählen Sie einen Knoten aus der Liste aus.

Wenn Sie einen mehrfach vorkommenden Knoten mit einer Gruppe verknüpfen, die das übergeordnete Element enthält, können Sie die Anzahl der einzubeziehenden Vorkommen der untergeordneten Elemente konfigurieren. Sie können die übergeordnete Gruppe mit der mehrfach vorkommenden untergeordneten Gruppe in der Umwandlungsausgabe ersetzen.

Um eine Gruppe zu erstellen, verknüpfen Sie einen Knoten mit einer leeren Spalte im Bereich **Umwandlungsausgabe**. Wenn Sie einen mehrfach vorkommenden untergeordneten Knoten mit einer leeren Eingabe- oder Ausgabespalte verknüpfen, fordert Sie das Developer Tool auf, die Gruppe mit anderen Ausgabegruppen in Beziehung zu bringen. Wenn Sie eine Gruppe auswählen, erstellt das Developer Tool Schlüssel, um die Gruppen zu verknüpfen.

Konfigurieren Sie die zugehörigen Gruppen von Ausgabeports im Bereich **Umwandlungsausgabe**. Wenn das Developer Tool Sie auffordert, Ausgabegruppen zu verknüpfen, fügt es den Gruppen die Schlüssel hinzu. Sie können auch manuell Ports hinzufügen, um Schlüssel zu repräsentieren.

Entwicklung der Umwandlung von hierarchisch in relational

Verwenden Sie den Assistenten für neue Umwandlungen, um eine Umwandlung von hierarchisch in relational automatisch zu generieren. Wählen Sie ein Schema oder eine hierarchische Beispieldatei zum Definieren der Eingabehierarchie.

1. Erstellen Sie die Umwandlung im Developer Tool.
2. Konfigurieren Sie den Eingabeport und das Mapping.
3. Testen Sie die Umwandlung.

Erstellen der Umwandlung von hierarchisch in relational

1. Klicken Sie im Developer Tool auf **Datei** > **Neu** > **Umwandlung**.
2. Wählen Sie die Umwandlung von hierarchisch in relational aus und klicken Sie auf **Weiter**.
3. Geben Sie einen Namen für die Umwandlung ein und suchen Sie nach einem Modellrepository-Speicherort, um die Umwandlung abzulegen. Klicken Sie dann auf **Weiter**.
4. Halten Sie sich zur Auswahl eines Schemas an eine der folgenden Methoden:
   - Um ein Schema aus dem Modellrepository zum Definieren der Eingabehierarchie zu verwenden, suchen Sie die Schemadatei in der Nähe des Felds **Schemaobjekt**, um diese dann aus dem Repository auszuwählen.
   - Klicken Sie zum Importieren einer Schemadatei auf **Neues Schemaobjekt erstellen**. Im Fenster **Neues Schemaobjekt** können Sie zu einer Schemadatei navigieren und diese auswählen oder wahlweise ein Schema aus einer Beispiel-Hierarchiedatei erstellen.
5. Wählen Sie das Root-Element für die Ausgabehierarchie. Wählen Sie im Dialogfeld **Hierarchie-Root** das Element im Schema aus, bei dem es sich um das Root-Element für die Ausgabehierarchiedatei handelt. Sie können zur Auswahl des Root-Objekts eine Beispiel-Hierarchiedatei hinzufügen. Um eine Beispieldatei hinzuzufügen, navigieren Sie in der Nähe des Felds **Beispieldatei** zu der Datei im Dateisystem, die Sie auswählen möchten.
6. Klicken Sie auf **Fertig stellen**.

   Der Assistent erstellt die Umwandlung im Repository.

Konfigurieren der Ports und Mappings

Konfigurieren Sie in der Ansicht **Übersicht** die Eingabe- und Ausgabeports.

1. Wählen Sie den Datentyp des Eingabeports, den Porttyp, die Gesamtstellenanzahl und die Dezimalstellen aus.
2. Wählen Sie zur Anzeige des Mappings im Bereich **Ports** der Ansicht **Übersicht** die Option**Ausgabe-Mapping** aus.
3. Erweitern Sie die Bäume im Gitter **Ports**. Links wird im Bereich **Umwandlungseingabe** die erwartete hierarchische Eingabe und rechts im Bereich **Umwandlungsausgabe** die relationale Ausgabe angezeigt.
4. Wenn Sie einen Knoten als Root-Element definieren möchten, klicken Sie auf **Hierarchie auswählen**.

   Im Developer Tool werden nur die Knoten ab Stammebene angezeigt. Unterhalb der Stammebene werden diese im Bereich **Umwandlungseingabe** angezeigt.

5. Wenn Linien angezeigt werden sollen, die die Ports mit den hierarchischen Knoten verbinden, klicken Sie auf **Linien zeigen**. Legen Sie fest, ob alle Verbindungslinien angezeigt werden sollen oder nur die Linien für bestimmte Ports.
6. Verwenden Sie zum Hinzufügen einer Eingabegruppe oder eines Eingabeports zum Bereich **Umwandlungsausgabe** eine der folgenden Methoden:
   - Ziehen Sie ein einfaches oder komplexes Element aus dem Bereich **Umwandlungseingabe** in eine leere Spalte im Bereich **Umwandlungsausgabe**. Wenn es sich um einen Gruppenknoten handelt, fügt das Developer Tool eine relationale Gruppe ohne Ports hinzu.
   - Um eine relationale Gruppe hinzuzufügen, wählen Sie eine Zeile aus und klicken Sie mit der rechten Maustaste darauf. Wählen Sie dann **Neu** > **Gruppe** aus.
   - Um einen relationalen Port hinzuzufügen, klicken Sie mit der rechten Maustaste und wählen Sie **Neu** > **Feld** aus.
7. Verwenden Sie eine der folgenden Methoden, um die Einstellungen für den hierarchischen Knoten für Speicherorte von Ports zu löschen:
   - Wählen Sie mindestens einen Knoten im Bereich **Umwandlungseingabe** aus, klicken Sie mit der rechten Maustaste und wählen Sie **Löschen** aus.
   - Wählen Sie mindestens eine Verbindungslinie zwischen den relationalen Ports und den hierarchischen Knoten aus, klicken Sie mit der rechten Maustaste und wählen Sie **Löschen** aus.
8. Klicken Sie zum Anzeigen der Ausgabeports in einer Hierarchie auf **Als Hierarchie anzeigen**. Jede untergeordnete Gruppe wird unterhalb der übergeordneten Gruppe angezeigt.

Testen der Umwandlung

Testen Sie die Umwandlung von hierarchisch in relational in der Ansicht **Daten-Viewer**.

Stellen Sie vor dem Testen der Umwandlung sicher, dass der Eingabespeicherort der Datei definiert wurde. Definieren Sie den Eingabespeicherort auf dem DIS-Computer in der Spalte „Eingabespeicherort“ des Bereichs **Ports** in der Ansicht **Übersicht**.

1. Öffnen Sie die Ansicht **Daten-Viewer**.
2. Klicken Sie auf **Ausführen**

   Das Developer Tool validiert die Umwandlung. Wenn kein Fehler vorliegt, zeigt das Developer Tool den Inhalt der hierarchischen Datei im Bereich **Ausgabe** an.

KAPITEL 17

Java-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Java-Umwandlung – Übersicht, 266
- Entwickeln einer Java-Umwandlung, 268
- Java-Umwandlungsports, 269
- Java-Umwandlung – Erweiterte Eigenschaften, 270
- Entwickeln von Java-Code, 273
- Java-Umwandlung – Java-Eigenschaften, 277
- Filteroptimierung mit der Java-Umwandlung, 280
- Erstellen einer Java-Umwandlung, 283
- Kompilieren einer Java-Umwandlung, 284
- Fehlerbehebung bei einer Java-Umwandlung, 285

Java-Umwandlung – Übersicht

Verwenden Sie die Java-Umwandlung, um die Developer Tool-Funktionalität zu erweitern.

Die Java-Umwandlung bietet eine einfache integrierte Programmierschnittstelle zum Definieren von Umwandlungsfunktionen mit der Programmiersprache Java. Sie können die Java-Umwandlung zum Definieren einfacher oder mäßig komplexer Umwandlungsfunktionen verwenden. Dabei sind weder umfangreiche Kenntnisse der Programmiersprache Java noch eine externe Java-Entwicklungsumgebung erforderlich. Die Java-Umwandlung ist eine aktive oder passive Umwandlung.

Das Developer Tool verwendet Java Development Kit (JDK) zum Kompilieren des Java-Codes und Generieren von Bytecode für die Umwandlung. Das Developer Tool speichert den Bytecode im Model Repository.

Der Data Integration Service verwendet die Java Runtime Environment (JRE), um generierten Bytecode zur Laufzeit auszuführen. Wenn der Data Integration Service ein Mapping mit einer Java-Umwandlung ausführt, verwendet er die JRE zum Ausführen des Bytecode, Verarbeiten von Eingabezeilen und Generieren von Ausgabezeilen.

Erstellen Sie Java-Umwandlungen durch Schreiben von Java-Codeausschnitten, die Umwandlungslogik definieren. Definieren Sie das Umwandlungsverhalten für eine Java-Umwandlung basierend auf den folgenden Ereignissen:

- Die Umwandlung empfängt eine Eingabezeile.
- Die Umwandlung verarbeitet alle Eingabezeilen.

Wiederverwendbare und nicht wiederverwendbare Java-Umwandlungen

Sie können eine wiederverwendbare oder eine nicht wiederverwendbare Java-Umwandlung erstellen.

Wiederverwendbare Umwandlungen können in mehreren Mappings vorhanden sein. Nicht wiederverwendbare Umwandlungen befinden sich in einem einzelnen Mapping.

Die Editoransichten, in denen Sie Eigenschaften definieren und Java-Code erstellen, variieren abhängig davon, ob Sie eine wiederverwendbare oder nicht wiederverwendbare Java-Umwandlung erstellen.

Aktive und passive Java-Umwandlungen

Beim Erstellen einer Java-Umwandlung legen Sie fest, ob es sich um einen aktiven oder passiven Typ handelt.

Wenn der Umwandlungstyp festgelegt ist, kann er nicht mehr geändert werden.

Eine Java-Umwandlung führt den Java-Code aus, den Sie einmal pro Eingabedatenzeile auf der Registerkarte **Bei Eingabe** definieren.

Eine Java-Umwandlung verarbeitet Ausgabezeilen basierend auf dem Umwandlungstyp wie folgt:

- Eine passive Java-Umwandlung generiert nach dem Verarbeiten der einzelnen Eingabezeilen für jede Eingabezeile eine Ausgabezeile in der Umwandlung.
- Eine aktive Java-Umwandlung generiert mehrere Ausgabezeilen für jede Eingabezeile in der Umwandlung.

  Verwenden Sie die generateRow-Methode zum Generieren der einzelnen Ausgabezeilen. Wenn beispielsweise die Umwandlung zwei Eingabeports enthält, die für ein Start- und ein Enddatum stehen, können Sie mit der generateRow-Methode eine Ausgabezeile für jedes Datum zwischen dem Start- und dem Enddatum generieren.

Datentypkonvertierung

Eine Java-Umwandlung wandelt Developer Tool-Datentypen basierend auf dem Porttyp der Java-Umwandlung in Java-Datentypen um.

Wenn eine Java-Umwandlung Eingabezeilen liest, wandelt sie Eingabeport-Datentypen in Java-Datentypen um.

Wenn eine Java-Umwandlung Ausgabezeilen schreibt, wandelt sie Java-Datentypen in Ausgabeport-Datentypen um.

Folgende Verarbeitung wird beispielsweise bei einem Eingabeport mit Integer-Datentyp in einer Java-Umwandlung durchgeführt:

1. Die Java-Umwandlung wandelt den Integer-Datentyp des Eingabeports in den primitiven int-Java-Datentyp um.
2. In der Umwandlung behandelt die Umwandlung den Wert des Eingabeports als primitiven int-Java-Datentyp.
3. Wenn die Umwandlung die Ausgabezeile generiert, wandelt sie den primitiven int-Java-Datentyp in den Integer-Datentyp um.

Die folgende Tabelle zeigt, wie die Java-Umwandlung Developer Tool-Datentypen zu primitiven und komplexen Java-Datentypen zuordnet:

Developer Tool-Datentyp	Java-Datentyp
bigint	long
binary	byte[]
date/time	Bei aktivierter Nanosekunden-Verarbeitung BigDecimal mit Nanosekunden-Präzision Bei deaktivierter Nanosekunden-Verarbeitung long mit Millisekunden-Präzision (Anzahl Millisekunden seit 1. Januar 1970 00:00:00.000 GMT)
decimal	Bei Deaktivierung der Verarbeitung mit hoher Präzision double mit Präzision 15 Bei Aktivierung der Verarbeitung mit hoher Präzision BigDecimal
double	double
integer	int
string	String
text	String

In Java sind die Datentypen String, byte[] und BigDecimal komplexe Datentypen und die Datentypen double, int und long primitive Datentypen.

Hinweis: Die Java-Umwandlung setzt Nullwerte in primitiven Datentypen auf Null. Sie können mit den isNull- und setNull-API-Methoden auf der Registerkarte **Bei Eingabe** Nullwerte im Eingabeport auf Nullwerte im Ausgabeport setzen. Ein Beispiel finden Sie unter "setNull" auf Seite 295.

Entwickeln einer Java-Umwandlung

Wenn Sie eine Java-Umwandlung entwickeln, müssen Sie Faktoren wie z. B. den zu erstellenden Umwandlungstyp berücksichtigen.

Wenn Sie eine Java-Umwandlung entwickeln, müssen folgende Fragen berücksichtigt werden:

- Möchten Sie eine aktive oder passive Java-Umwandlung erstellen?

  Eine passive Java-Umwandlung generiert eine Ausgabezeile für jede Eingabezeile in der Umwandlung.

  Eine aktive Java-Umwandlung generiert mehrere Ausgabezeilen für jede Eingabezeile in der Umwandlung.
- Möchten Sie in der Java-Umwandlung Funktionen definieren? Wenn ja, welche Ausdrücke sollen in die einzelnen Funktionen eingebunden werden?

  Sie können beispielsweise eine Funktion zum Aufrufen eines Ausdrucks definieren, um die Werte von Eingabe- oder Ausgabeports oder die Werte von Java-Umwandlungsvariablen zu suchen.
- Möchten Sie eine wiederverwendbare oder nicht wiederverwendbare Java-Umwandlung erstellen?

  Eine wiederverwendbare Umwandlung kann in mehreren Mappings vorkommen.

  Eine nicht wiederverwendbare Umwandlung kann in einem einzelnen Mapping vorkommen.

Java-Umwandlungsports

Eine Java-Umwandlung kann Eingabe- und Ausgabeports haben.

Verwenden Sie zum Erstellen und Bearbeiten von Ports für eine nicht wiederverwendbare Java-Umwandlung die Registerkarte **Ports** im Editor. Verwenden Sie zum Erstellen und Bearbeiten von Ports für eine wiederverwendbare Java-Umwandlung die **Übersicht**-Ansicht im Editor.

Sie können Standardwerte für die Ports festlegen. Nachdem Sie einer Umwandlung Ports hinzugefügt haben, können Sie die Portnamen als Variablen in Java-Codeausschnitten verwenden.

Erstellen von Ports

Wenn Sie eine Java-Umwandlung erstellen, enthält diese eine Eingabegruppe und eine Ausgabegruppe.

Wenn Sie einen Port erstellen, fügt das Developer Tool diesen unter der aktuell ausgewählten Zeile oder Gruppe ein.

Festlegen von Standardportwerten

Sie können für die Ports in einer Java-Umwandlung Standardwerte definieren.

Die Java-Umwandlung initialisiert Portvariablen mit dem Standardportwert basierend auf dem Datentyp des Ports.

Eingabe- und Ausgabeports

Die Java-Umwandlung initialisiert den Wert von nicht verbundenen Eingabeports oder Ausgabeports, denen kein Wert in den Java-Codeausschnitten zugewiesen ist.

Java-Umwandlung initialisiert Ports, basierend auf den folgenden Port-Datentypen:

Einfach

Wenn Sie einen Standardwert für den Port definieren, der ungleich Null ist, initialisiert die Umwandlung den Wert der Portvariablen auf den Standardwert. Andernfalls wird der Wert der Portvariablen auf 0 initialisiert.

Komplex

Wenn Sie einen Standardwert für den Port definieren, erstellt die Umwandlung ein neues String- oder byte[]-Objekt und initialisiert das Objekt auf den Standardwert. Andernfalls initialisiert die Umwandlung die Portvariable auf Null.

Hinweis: Wenn Sie auf eine Eingabeportvariable mit einem Nullwert im Java-Code zugreifen, tritt eine NullPointerException auf.

Sie können einen Eingabeport als Partitionsschlüssel oder Sortierschlüssel aktivieren, und eine Sortierrichtung zuweisen, wenn Sie Mappings in einer Hive-Umgebung ausführen. Der Datenintegrationsdienst stellt sicher, dass Daten über die Reducer-Aufgaben im Hadoop-Cluster partitioniert werden, und sortiert die Ausgabe anhand des Sortierschlüssels und der Sortierrichtung. Partitionsschlüssel und Sortierschlüssel sind in einer Hive-Umgebung gültig, wenn der Umwandlungsumfang auf „Alle Eingaben“ eingestellt ist.

Verwenden Sie die folgenden Eigenschaften für die Partitionierung und das Sortieren von Daten, wenn Sie die Umwandlung in eine Hive-Umgebung verschieben:

Partitionsschlüssel

Eingabeport, der die Partition bestimmt, in denen eine bestimmte Zeile an Daten gespeichert ist. Wenn Sie Partitionsschlüssel und Sortiersschlüssel auswählen, generiert der Datenintegrationsdienst eine sortierte Ausgabe.

Sortierschlüssel

Eingabeport, der die Sortierkriterien bestimmt. Sie müssen Sortierschlüssel und Partitionsschlüssel auswählen, um sicherzustellen, dass Daten sortiert werden.

Richtung

Aufsteigende oder absteigende Reihenfolge. Standardwert ist aufsteigend

Java-Umwandlung – Erweiterte Eigenschaften

Die Java-Umwandlung enthält erweiterte Eigenschaften sowohl für den Umwandlungscode als auch für die Umwandlung.

Wenn Sie die Umwandlung in einem Mapping verwenden, können Sie die Umwandlungseigenschaften überschreiben.

Sie können die folgenden erweiterten Eigenschaften für die Java-Umwandlung auf der Registerkarte **Erweitert** definieren.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Partitionierbar

Die Umwandlung kann mit mehreren Threads verarbeitet werden. Löschen Sie diese Option, wenn der Datenintegrationsdienst einen Thread verwenden soll, um die Umwandlung zu verarbeiten. Der Datenintegrationsdienst kann mehrere Threads verwenden, um die verbleibenden Mapping-Pipeline-Stages zu verarbeiten.

Deaktivieren Sie die Partitionierung für eine Java-Umwandlung, wenn der Java-Code vorgibt, dass die Umwandlung mit einem Thread verarbeitet wird.

Hohe Genauigkeit aktivieren

Verarbeitet einen Port des Datentyps „Dezimal“ mit einer Gesamtstellenanzahl von größer als 15 und kleiner als oder gleich 38 als Port des Datentyps „Java BigDecimal“.

Aktivieren Sie die hohe Genauigkeit, um einen Port des Datentyps „Dezimal“ mit einer Gesamtstellenanzahl von größer als 15 und kleiner als oder gleich 38 als Port des Datentyps „Java Doppelt“ zu verarbeiten.

In der folgenden Tabelle wird veranschaulicht, wie eine Java-Umwandlung einen Wert in einem Eingabeport des Datentyps „Dezimal“ behandelt, je nachdem, ob die Option für hohe Genauigkeit aktiviert oder deaktiviert ist:

Beispiel	Verarbeitung mit hoher Genauigkeit aktiviert	Verarbeitung mit hoher Genauigkeit deaktiviert
Ein Eingabeport des Dezimaltyps empfängt den Wert 40012030304957666903.	Die Java-Umwandlung behält den Wert unverändert bei.	Die Java-Umwandlung wandelt den Wert um in folgenden Wert: `4.00120303049577 x 10^19`

Nanosekunden in date/time verwenden

Wandelt Ports mit Datentyp date/time in Java-Ports mit BigDecimal-Datentyp mit Nanosekunden-Genauigkeit um.

Deaktivieren Sie die Nanosekunden-Verarbeitung, damit der generierte Java-Code date/time-Datentypports in Java-Ports des Long-Datentyps mit Millisekunden-Genauigkeit umwandelt.

Klassenpfad

Legt den Klassenpfad für JAR- oder Klassendatei-Verzeichnisse fest, die mit nicht standardmäßigen Java-Paketen verbunden sind, die Sie auf der Registerkarte **Importe** importieren.

Die JAR- oder Klassendatei-Verzeichnisse müssen für die Kompilierung des Java-Codes auf dem Client-Computer des Developer tools zugänglich sein.

Trennen Sie abhängig vom Betriebssystem Klassenpfad-Einträge wie folgt:

- Verwenden Sie unter UNIX ein Komma, um Klassenpfad-Einträge zu trennen.
- Verwenden Sie unter Windows ein Semikolon, um Klassenpfad-Einträge zu trennen.

Wenn Sie beispielsweise das Java-Konverterpaket auf der Registerkarte **Importe** importieren und das Paket in converter.jar definieren, müssen Sie den Speicherort der Datei converter.jar zum Klassenpfad hinzufügen, bevor Sie den Java-Code für die Java-Umwandlung kompilieren.

Hinweis: Bei integrierten Java-Paketen muss der Klassenpfad nicht festgelegt werden. Bei java.io handelt es sich beispielsweise um ein integriertes Java-Paket, deshalb muss kein Klassenpfad für java.io festgelegt werden.

Ist aktiv

Die Umwandlung kann nicht mehr als eine Ausgabezeile für jede Eingabezeile generieren.

Sie können diese Eigenschaft nach dem Erstellen der Java-Umwandlung nicht mehr ändern. Wenn Sie diese Eigenschaft ändern möchten, müssen Sie eine neue Java-Umwandlung erstellen.

Umwandlungsbereich

Definiert die Methode, die der Datenintegrationsdienst verwendet, um die Umwandlungslogik auf eingehende Daten anzuwenden. Sie können einen der folgenden Werte auswählen:

- Zeile. Wendet die Umwandlungslogik auf jeweils eine Datenzeile gleichzeitig an. Wählen Sie "Zeile", wenn die Ergebnisse der Prozedur von einer einzigen Datenzeile abhängen.
- Transaktion. Wendet die Umwandlungslogik auf alle Zeilen in einer Transaktion an. Wählen Sie "Transaktion", wenn die Ergebnisse der Prozedur von allen Zeilen in derselben Transaktion abhängen, aber nicht von Zeilen in anderen Transaktionen. Wenn Sie "Transaktion" wählen, müssen Sie alle Eingabegruppen mit dem gleichen Transaktionskontrollpunkt verbinden.

- Alle Eingaben. Wendet die Umwandlungslogik auf alle eingehenden Daten an. Wenn Sie "Alle Eingaben" auswählen, löscht der Datenintegrationsdienst alle Transaktionsgrenzen. Wählen Sie "Alle Eingaben", wenn die Ergebnisse der Prozedur von allen Datenzeilen in der Quelle abhängen.

Hinweis: Die Eigenschaft "Umwandlungsbereich" ist nur in einer Hive-Umgebung gültig.

Ohne Status

Beibehalten der Zeilenreihenfolge der Eingabedaten für die Umwandlung. Wählen Sie diese Option aus, wenn der Datenintegrationsdienst keine Optimierung durchführen soll, die die Zeilenreihenfolge verändern könnte.

Wenn der Datenintegrationsdienst Optimierungen durchführt, geht eine zu einem vorherigen Zeitpunkt im Mapping erstellte Reihenfolge möglicherweise verloren. Die Reihenfolge in einem Mapping können Sie mit einer Einfachdateiquelle, einer sortierten relationalen Quelle oder einer Sortierer-Umwandlung erstellen. Wenn Sie eine Umwandlung konfigurieren, um die Zeilenreihenfolge beizubehalten, berücksichtigt der Datenintegrationsdienst diese Konfiguration beim Durchführen von Optimierungen für das Mapping. Der Datenintegrationsdienst führt Optimierungen für die Umwandlung durch, wenn er die Reihenfolge beibehalten kann. Der Datenintegrationsdienst führt keine Optimierungen für die Umwandlung durch, wenn die Optimierungen die Zeilenreihenfolge verändern würden.

Konfigurieren des Klassenpfads für den Developer Tool-Client

Sie können JAR-Dateien oder Klassendatei-Verzeichnisse dem Developer Tool-Client-Klassenpfad hinzufügen.

Zum Festlegen des Klassenpfads für den Computer, auf dem der Developer Tool-Client ausgeführt wird, führen Sie eine der folgenden Aufgaben durch:

- Konfigurieren Sie die Umgebungsvariable CLASSPATH. Legen Sie die Umgebungsvariable CLASSPATH auf dem Developer Tool-Client-Computer fest. Dies gilt für alle Java-Prozesse, die auf dem Computer ausgeführt werden.
- Bei einer nicht wiederverwendbaren Java-Umwandlung konfigurieren Sie den Klassenpfad in den Java-Umwandlungs- unter den erweiterten Eigenschaften. Dies gilt für Mappings, die diese Java-Umwandlung enthalten. Der Developer Tool-Client schließt Dateien innerhalb des Klassenpfad ein, wenn er den Java-Code kompiliert. Diese Eigenschaft kann nicht für eine wiederverwendbare Java-Umwandlung konfiguriert werden.

Zum Hinzufügen von JAR- oder Klassendatei-Verzeichnissen zum Klassenpfad in einer Java-Umwandlung gehen Sie folgendermaßen vor:

1. Klicken Sie auf der Registerkarte **Erweitert** auf den Abwärtspfeil in der Spalte **Wert** neben **Klassenpfad**.

   Das Dialogfeld **Klassenpfad bearbeiten** wird angezeigt.
2. Zum Hinzufügen eines Klassenpfads gehen Sie folgendermaßen vor:
   a. Klicken Sie auf **Hinzufügen**.

      Das Fenster **Speichern unter** wird angezeigt.
   b. Navigieren Sie im Fenster **Speichern unter** in das Verzeichnis, in dem die JAR-Datei gespeichert ist.
   c. Klicken Sie auf **OK**.

   Der Klassenpfad wird im Dialogfeld **Klassenpfad bearbeiten** angezeigt.
3. Um ein JAR- oder Klassendatei-Verzeichnis zu entfernen, wählen Sie das JAR- oder Klassendatei-Verzeichnis aus, und klicken Sie auf **Entfernen**.

   Das Verzeichnis wird in der Verzeichnisliste nicht mehr angezeigt.

Konfigurieren des Klassenpfads für den Data Integration Service

Sie können während der Laufzeit benötigte JAR- oder Klassendatei-Verzeichnisse zum Klassenpfad auf dem Data Integration Service-Knoten hinzufügen.

Speichern Sie die während der Laufzeit benötigten JAR-Dateien in folgendem Verzeichnis auf dem Data Integration Service-Knoten:

```
$INFA_HOME/services/shared/jars/thirdparty
```

Die JAR-Dateien an diesem Speicherort werden dynamisch geladen. Die von einzelnen Mappings während der Laufzeit benötigten Klassendateien werden in diesem Verzeichnis gesucht und daraus geladen.

Hinweis: Die Java-Umwandlung fügt JAR-Dateien in diesem Verzeichnis dem Klassenpfad auf Mapping-Ebene hinzu.

Entwickeln von Java-Code

Verwenden Sie die Codeeingabe-Registerkarten in der **Java**-Ansicht, um Java-Code zu schreiben und zu kompilieren, der Umwandlungsverhalten bei bestimmten Umwandlungsereignissen definiert.

Sie können Codeausschnitte auf den Codeeingabe-Registerkarten in beliebiger Reihenfolge entwickeln. Auf der Registerkarte **Full-Code** können Sie den vollständigen Java-Code anzeigen, aber nicht bearbeiten.

Nach dem Entwickeln der Code-Ausschnitte können Sie die Codeausschnitte oder den vollständigen Java-Code kompilieren und die Kompilierungsergebnisse im Fenster **Ergebnisse** unter den Eigenschaften der **Kompilierung** in der **Java**-Ansicht anzeigen.

Jede Codeeingabe-Registerkarte enthält Komponenten, die Sie zum Schreiben, Anzeigen und Kompilieren von Java-Code verwenden:

Code-Eigenschaften

Stellt Steuerelemente zur Verfügung, mit denen Sie Java-Code anzeigen und eingeben können, einschließlich Java-Umwandlungs-API-Methoden. In der folgenden Tabelle werden die Steuerelemente beschrieben, die in den **Code**-Eigenschaften zur Verfügung stehen:

Kontrolle	Beschreibung
Navigator	Zeigt Eingabeports, Ausgabeports und aufrufbare Java-Umwandlungs-API-Methoden. Klicken Sie auf ein Element im Navigator, um eine entsprechende Beschreibung anzuzeigen. Doppelklicken Sie auf ein Element, um es dem Fenster **Java-Code** hinzuzufügen. Alternativ können Sie ein Element aus dem Navigator in das Fenster **Java-Code** ziehen. Der Navigator steht auf folgenden Codeeingabe-Registerkarten zur Verfügung: - **Helfer** - **Bei Eingabe** - **Am Datenende**
Fenster **Java-Code**	Ermöglicht Ihnen das Anzeigen oder Eingeben von Java-Code für die Umwandlung. Das Fenster **Java-Code** zeigt Java-Code unter Verwendung der Java-Syntaxhervorhebung an. **Hinweis:** Auf der Registerkarte **Full-Code** können Sie den vollständigen Klassencode für die Java-Umwandlung anzeigen, aber nicht bearbeiten. Das Fenster **Java-Code** steht auf den folgenden Codeeingabe-Registerkarten zur Verfügung: - **Importe** - **Helfer** - **Bei Eingabe** - **Am Datenende** - **Funktionen** - **Optimierungsschnittstelle** - **Full-Code**
Befehl **Neue Funktion**	Öffnet das Dialogfeld **Funktion definieren**, in dem Sie Funktionen definieren, die Java-Ausdrücke aufrufen. Der Befehl **Funktion** steht auf der Registerkarte **Funktionen** zur Verfügung.
Bearbeiten-Symbolleiste	Ermöglicht Ihnen die Auswahl von Symbolen zum Bearbeiten von Java-Code, z. B. zum Ausschneiden, Kopieren oder Einfügen. Die Bearbeiten-Symbolleiste steht auf den folgenden Codeeingabe-Registerkarten zur Verfügung: - **Importe** - **Helfer** - **Bei Eingabe** - **Am Datenende** - **Funktionen** - **Optimierungsschnittstelle**

Kompilierungseigenschaften

Stellt Steuerelemente zum Kompilieren und Debuggen von Java-Code bereit. In der folgenden Tabelle werden die Steuerelemente in den Eigenschaften **Kompilierung** beschrieben:

Kontrolle	Beschreibung
Befehl **Kompilieren**	Kompiliert den Java-Code für die Umwandlung.
Fenster **Ergebnisse**	Zeigt die Kompilierungsergebnisse für die Java-Umwandlungsklasse an und ermöglicht das Suchen von Codefehlern. Um nach einem Fehler im Code zu suchen, klicken Sie im Fenster **Ergebnisse** mit der rechten Maustaste auf eine Fehlermeldung. Sie können auswählen, ob der Fehler im Ausschnittscode oder im vollständigen Code angezeigt wird. Sie können auch auf eine Fehlermeldung im Fenster **Ergebnisse** doppelklicken, um die Fehlerquelle zu suchen.

Erstellen von Java-Codeausschnitten

Um Java-Codeausschnitte zum Definieren von Umwandlungsverhalten zu erstellen, verwenden Sie das **Java-Code**-Fenster in den Codeeingabe-Registerkarten.

1. Klicken Sie auf die entsprechende Codeeingabe-Registerkarte.

   In der folgenden Tabelle werden die Aufgaben beschrieben, die Sie auf den Codeeingabe-Registerkarten in der **Java**-Ansicht ausführen können:

Tab	Beschreibung
Importe	Importiert Java-Pakete von Drittanbietern sowie integrierte und benutzerdefinierte Java-Pakete für eine aktive oder passive Java-Umwandlung. Nach dem Importieren von Paketen können Sie diese auf den anderen Codeeingabe-Registerkarten verwenden.
Helfer	Deklariert benutzerdefinierte Variablen und Methoden für die Java-Umwandlungsklasse in einer aktiven oder passiven Java-Umwandlung. Nachdem Sie Variablen und Methoden deklariert haben, können Sie diese auf den anderen Codeeingabe-Registerkarten verwenden, mit Ausnahme der Registerkarte **Importe**.
Bei Eingabe	Definiert, wie sich eine aktive oder passive Java-Umwandlung beim Empfang einer Eingabezeile verhält. Der Java-Code, den Sie auf dieser Registerkarte definieren, wird pro Eingabezeile einmal ausgeführt. Auf dieser Registerkarte haben Sie außerdem die Möglichkeit, auf Eingabe- und Ausgabeportdaten, Variablen und Java-Umwandlungs-API-Methoden zuzugreifen und diese zu verwenden.
Am Datenende	Definiert, wie sich eine aktive oder passive Java-Umwandlung nach der Verarbeitung sämtlicher Eingabedaten verhält. Auf dieser Registerkarte können Sie auch die Ausgabedaten für aktive Umwandlungen festlegen und Java-Umwandlungs-API-Methoden aufrufen.

Tab	Beschreibung
Funktionen	Definiert Funktionen, die Ausdrücke in einer Java-Umwandlung mit der Programmiersprache Java aufrufen. Beispielsweise können Sie eine Funktion zum Aufrufen eines Ausdrucks definieren, die Werte von Eingabe- oder Ausgabeports oder Werte von Java-Umwandlungsvariablen durchsucht. Auf der Registerkarte **Funktionen** können Sie Funktionen manuell definieren oder auf **Neue Funktion** klicken, um das Dialogfeld **Funktion definieren** aufzurufen. Dort können Sie auf einfache Weise eine Funktion definieren.
Optimierungsschnittstelle	Definiert "Early Selection"- oder Push-Into-Optimierung. Wählen Sie die Optimierungsmethode im Navigator aus. Aktualisieren Sie die Codeausschnitte zum Aktivieren der Optimierung. Definieren Sie die Eingabeports und zugeordneten Ausgabeports, durch die die Filterlogik geleitet werden soll.
Full-Code	Schreibgeschützt. Auf dieser Registerkarte können Sie den vollständigen Klassencode für die Java-Umwandlung anzeigen und kompilieren.

2. Um auf die Eingabe- oder Ausgabespalten-Variablen im Ausschnitt zuzugreifen, erweitern Sie die **Eingabe**- oder**Ausgabe**-Liste im Navigator, und doppelklicken Sie auf den Portnamen.
3. Um eine Java-Umwandlungs-API im Ausschnitt aufzurufen, erweitern Sie im Navigator die Liste **Aufrufbare APIs**, und doppelklicken Sie auf den Namen der Methode. Falls erforderlich, konfigurieren Sie die entsprechenden Eingabewerte für die Methode.
4. Schreiben Sie den entsprechenden Java-Code basierend auf dem Codeeingabe-Registerkartentyp.

   Zeigen Sie den vollständigen Klassencode für die Java-Umwandlung im **Java-Code**-Fenster auf der **Full-Code**-Registerkarte an.

Importieren von Java-Paketen

Auf der Registerkarte **Importe** können Sie Java-Pakete für aktive oder passive Java-Umwandlungen importieren.

Sie können Java-Pakete von Drittanbietern sowie integrierte oder benutzerdefinierte Java-Pakete importieren. Nachdem Sie Java-Pakete importiert haben, können Sie die importierten Pakete auf den anderen Codeeingabe-Registerkarten verwenden.

Hinweis: Auf der Registerkarte **Importe** können statische Variablen, Instanzvariablen und Benutzermethoden weder deklariert noch verwendet werden.

Wenn Sie bei Developer Tool Metadaten exportieren oder importieren, die eine Java-Umwandlung enthalten, werden die JAR- oder Klassendateien, die die für die Java-Umwandlung erforderlichen Drittanbieter- oder benutzerdefinierten Pakete beinhalten, nicht in den Export oder Import einbezogen.

Wenn Sie Metadaten importieren, die eine Java-Umwandlung enthalten, müssen Sie die JAR- oder Klassendateien, die die erforderlichen Drittanbieter- oder benutzerdefinierten Pakete enthalten, auf den Developer Tool-Client und Data Integration Service-Knoten kopieren.

Um beispielsweise das I/O-Java-Paket zu importieren, geben Sie den folgenden Code auf der Registerkarte **Importe** ein:

```
import java.io.*;
```

Beim Importieren von nicht standardmäßigen Java-Paketen fügen Sie das Paket oder die Klasse zum Klassenpfad in der Java-Umwandlung hinzu.

Definieren von Helfer-Code

Auf der Registerkarte **Helfer** können Sie benutzerdefinierte Variablen und Methoden für die Java-Umwandlungsklasse in aktiven oder passiven Java-Umwandlungen deklarieren.

Nachdem Sie Variablen und Methoden auf der Registerkarte **Helfer** deklariert haben, können Sie die Variablen und Methoden auf allen Codeeingabe-Registerkarten verwenden, mit Ausnahme der Registerkarte **Importe**.

Auf der Registerkarte **Helfer** können Sie die folgenden Typen von Code-, Variablen und Methoden deklarieren:

- Statischer Code und statische Variablen.

  Innerhalb eines statischen Blocks können Sie statische Variablen und statischen Code deklarieren. Alle Instanzen einer wiederverwendbaren Java-Umwandlung in einem Mapping nutzen statischen Code und Variablen gemeinsam. Statischer Code wird vor jedem anderen Code in einer Java-Umwandlung ausgeführt.

  Der folgende Code deklariert beispielsweise eine statische Variable zum Speichern des Fehlerschwellenwerts für alle Instanzen einer Java-Umwandlung in einem Mapping:

  ```
  static int errorThreshold;
  ```

  Verwenden Sie diese Variable zum Speichern des Fehlerschwellenwerts für die Umwandlung und greifen Sie darauf aus allen Instanzen der Java-Umwandlung in einem Mapping zu.

  Hinweis: Statische Variablen müssen in einer wiederverwendbaren Java-Umwandlung synchronisiert werden.

- Instanzvariablen.

  Instanzvariablen werden nicht von mehreren Instanzen einer wiederverwendbaren Java-Umwandlung in einem Mapping gemeinsam genutzt. Deklarieren Sie Instanzvariablen zur Vermeidung von Konflikten mit einem Präfix, und initialisieren Sie nicht-primitive Instanzvariablen.

  Der folgende Code verwendet beispielsweise eine boolesche Variable, um zu entscheiden, ob eine Ausgabezeile generiert wird:

  ```
  // boolean to decide whether to generate an output row
  // based on validity of input
  private boolean generateRow;
  ```

- Benutzerdefinierte statische oder Instanzmethoden.

  Erweitert die Funktionen der Java-Umwandlung. Auf der Registerkarte **Helfer** deklarierte Java-Methoden können Ausgabevariablen oder lokal deklarierte Instanzvariablen verwenden oder ändern. Der Zugriff auf Eingabevariablen von Java-Methoden auf der Registerkarte **Helfer** ist nicht möglich.

  Verwenden Sie zum Beispiel den folgenden Code auf der Registerkarte **Helfer**, um eine Funktion zu deklarieren, die zwei Ganzzahlen hinzufügt:

  ```
  private int myTXAdd (int num1,int num2)
  {
       return num1+num2;
  }
  ```

Java-Umwandlung – Java-Eigenschaften

Verwenden Sie die Codeeingabe-Registerkarten in der **Java**-Ansicht, um Java-Code zu schreiben und zu kompilieren, der Umwandlungsverhalten bei bestimmten Umwandlungsereignissen definiert.

Bei folgenden Registerkarten handelt es sich um Codeeingabe-Registerkarten:

- **Importe**

- **Helfer**
- **Bei Eingabe**
- **Am Datenende**
- **Funktionen**
- **Optimierungsschnittstelle**

Der vollständige Klassencode für die Java-Umwandlung wird auf der Registerkarte **Full-Code** angezeigt.

Registerkarte "Importe"

Auf der Registerkarte **Importe** können Sie Java-Pakete von Drittanbietern sowie integrierte oder benutzerdefinierte Java-Pakete für aktive oder passive Java-Umwandlungen importieren.

Geben Sie zum Importieren eines Java-Pakets den Code für den Paketimport im Fenster **Java-Code** in den **Code**-Eigenschaften auf der Registerkarte **Importe** ein.

Sie können beispielsweise folgenden Code eingeben, um das java.io-Paket zu importieren:

```
import java.io.*;
```

Um den Code zum Importieren der Java-Pakete zu kompilieren, klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Importe**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Importe** angezeigt.

Nach dem Import der Java-Pakete können Sie diese auf den anderen Codeeingabe-Registerkarten verwenden.

Registerkarte "Helfer"

Auf der Registerkarte **Helfer** können Sie benutzerdefinierte Variablen und Methoden für die Java-Umwandlungsklasse in einer aktiven oder passiven Java-Umwandlung deklarieren.

Um benutzerdefinierte Variablen und Methoden zu deklarieren, geben Sie den Code im Fenster **Java-Code** in den **Code**-Eigenschaften auf der Registerkarte **Helfer** ein.

Um Helfer-Code für die Java-Umwandlung zu kompilieren, klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Helfer**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Helfer** angezeigt.

Nachdem Sie Variablen und Methoden deklariert haben, können Sie diese auf den anderen Codeeingabe-Registerkarten verwenden, mit Ausnahme der Registerkarte **Importe**.

Registerkarte "Bei Eingabe"

Auf der Registerkarte **Bei Eingabe** definieren Sie, wie sich eine aktive oder passive Java-Umwandlung bei Empfang einer Eingabezeile verhält. Auf dieser Registerkarte haben Sie außerdem die Möglichkeit, auf Eingabe- und Ausgabeportdaten, Variablen und Java-Umwandlungs-API-Methoden zuzugreifen und diese zu verwenden.

Der Java-Code, den Sie auf dieser Registerkarte definieren, wird pro Eingabezeile einmal ausgeführt.

Um das Verhalten einer Java-Umwandlung bei Empfang einer Eingabezeile zu definieren, geben Sie den Code im Fenster **Java-Code** in den **Code**-Eigenschaften auf der Registerkarte **Bei Eingabe** ein.

Über den Navigator auf der Registerkarte **Bei Eingabe** können Sie auf folgende Variablen und API-Methoden zugreifen und diese definieren:

- Eingabeport- und Ausgabeport-Variablen. Für den Zugriff auf Eingabe- und Ausgabeportdaten als Variable verwenden Sie als Variablenname den Namen des Ports. Wenn beispielsweise "in_int" ein Ganzzahl-Eingabeport ist, können Sie auf die Daten für diesen Port unter Verwendung der Variablen "in_int" mit dem primitiven Datentyp int zugreifen. Es müssen keine Eingabe- oder Ausgabeports als Variablen deklariert werden.

  Weisen Sie Eingabeport-Variablen keinen Wert zu. Wenn Sie einer Eingabevariablen auf der Registerkarte **Bei Eingabe** einen Wert zuweisen, können Sie keine Eingabedaten für den entsprechenden Port in der aktuellen Zeile abrufen.

- Instanzvariablen und benutzerdefinierte Methoden. Verwenden Sie Instanz- oder statische Variablen bzw. benutzerdefinierte Methoden, die Sie auf der Registerkarte **Helfer** deklariert haben.

  Beispiel: Eine aktive Java-Umwandlung hat zwei Eingabeports, BASE_Salary und BONUSES, mit dem Datentyp Integer und einen einzelnen Ausgabeport, TOTAL_COMP, mit dem Datentyp Integer. Sie erstellen auf der Registerkarte **Helfer** eine benutzerdefinierte Methode, myTXAdd, die zwei Ganzzahlen hinzufügt und das Ergebnis zurückgibt. Verwenden Sie den folgenden Java-Code auf der Registerkarte **Bei Eingabe**, um die Gesamtwerte für die Eingabeports den Ausgabeports zuzuweisen und eine Ausgabezeile zu generieren:

  ```
  TOTAL_COMP = myTXAdd (BASE_SALARY,BONUSES);
  generateRow();
  ```

  Wenn die Java-Umwandlung eine Eingabezeile empfängt, fügt sie die Werte der Eingabeports BASE_SALARY und BONUSES hinzu, weist den Wert dem Ausgabeport TOTAL_COMP zu und generiert eine Ausgabezeile.

- Java-Umwandlungs-API-Methoden. Sie können API-Methoden aufrufen, die von der Java-Umwandlung zur Verfügung gestellt werden.

Zum Kompilieren des Codes für die Java-Umwandlung klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Bei Eingabe**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Bei Eingabe** angezeigt.

Registerkarte "Am Datenende"

Auf der Registerkarte **Am Datenende** definieren Sie, wie sich eine aktive oder passive Java-Umwandlung nach der Verarbeitung sämtlicher Eingabedaten verhält. Auf dieser Registerkarte können Sie auch die Ausgabedaten für aktive Umwandlungen festlegen und Java-Umwandlungs-API-Methoden aufrufen.

Um das Verhalten einer Java-Umwandlung nach Verarbeitung sämtlicher Eingabedaten zu definieren, geben Sie den Code im Fenster **Java-Code** in den **Code**-Eigenschaften auf der Registerkarte **Am Datenende** ein.

Sie können auf folgende Variablen und API-Methoden auf der Registerkarte **Am Datenende** zugreifen und diese definieren:

- Ausgabeport-Variablen. Sie können die Namen aller Ausgabeports verwenden, die Sie auf der Registerkarte **Ports** als Variablen definiert haben, oder Ausgabedaten für aktive Java-Umwandlungen festlegen.
- Instanzvariablen und benutzerdefinierte Methoden. Verwenden Sie die Instanzvariablen oder benutzerdefinierten Methoden, die Sie auf der Registerkarte **Helfer** deklariert haben.
- Java-Umwandlungs-API-Methoden. Rufen Sie API-Methoden auf, die von der Java-Umwandlung zur Verfügung gestellt werden.

  Verwenden Sie zum Beispiel den folgenden Java-Code, um Informationen in das Protokoll zu schreiben, wenn das Ende der Daten erreicht ist:

  ```
  logInfo("Number of null rows for partition is: " + partCountNullRows);
  ```

Zum Kompilieren des Codes für die Java-Umwandlung klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Am Datenende**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Am Datenende** angezeigt.

Registerkarte "Funktionen"

Auf der Registerkarte **Funktionen** definieren Sie Funktionen zum Aufrufen von Ausdrücken in einer Java-Umwandlung mit der Programmiersprache Java.

Beispielsweise können Sie eine Funktion zum Aufrufen eines Ausdrucks definieren, die Werte von Eingabe- oder Ausgabeports oder Werte von Java-Umwandlungsvariablen durchsucht.

Zum Definieren von Funktionen können Sie entweder im Fenster **Java-Code** in den **Code**-Eigenschaften auf der Registerkarte **Funktionen** Funktionen manuell definieren oder auf **Neue Funktion** klicken und im daraufhin angezeigten Dialogfeld **Funktion definieren** eine Funktion definieren.

Zum Kompilieren des Codes klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Funktionen**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Funktionen** angezeigt.

Registerkarte "Full-Code"

Auf der Registerkarte **Full-Code** haben Sie die Möglichkeit, den vollständigen Klassencode für die Java-Umwandlung anzuzeigen und zu kompilieren. Der Code kann jedoch nicht geändert werden.

Sie können den vollständigen Klassencode im Fenster **Java-Code** in den **Code**-Eigenschaften anzeigen.

Um den vollständigen Code für die Java-Umwandlung zu kompilieren, klicken Sie auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Full-Code**. Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** auf der Registerkarte **Full-Code** angezeigt.

Filteroptimierung mit der Java-Umwandlung

Der Datenintegrationsdienst kann die Filteroptimierung auf aktive Java-Umwandlungen anwenden. Beim Definieren der Java-Umwandlung fügen Sie Code für die Filteroptimierung auf der Registerkarte **Optimierungsschnittstellen** der Java-Umwandlung hinzu.

Frühe Auswahloptimierung mit der Java-Umwandlung

Sie können eine aktive oder passive Java-Umwandlung für die frühe Auswahloptimierung aktivieren, wenn die Java-Umwandlung keine Nebeneffekte aufweist. Der Optimierer übergibt der Filterlogik über die Java-Umwandlung und ändert die Filterbedingung nach Bedarf.

Zum Anzeigen der Codeausschnitte für die "Early Selection"-Optimierung wählen Sie PredicatePushOptimization im Navigator der Registerkarte **Optimierungsschnittstelle** aus.

allowPredicatePush

Boolescher Wert. Aktiviert "Early Selection". Ändern Sie die Funktion, um zur Aktivierung von "Early Selection" den Wahrheitswert "True" und eine Meldung zurückzugeben. Der Standardwert lautet "False" und die Funktion gibt eine Meldung mit dem Hinweis zurück, dass die Optimierung nicht unterstützt wird.

```
public ResultAndMessage allowPredicatePush(boolean ignoreOrderOfOp) {
    // To Enable PredicatePushOptimization, this function should return true
    //return new ResultAndMessage(true, "");
    return new ResultAndMessage(false, "Predicate Push Optimization Is Not
Supported");
}
```

canGenerateOutputFieldEvalError

Boolescher Wert. Gibt an, ob die Java-Umwandlung einen Ausgabefeldfehler zurückgeben kann, beispielsweise einen Division durch Null-Fehler. Ändern Sie die Funktion, um "False" zurückzugeben, wenn die Java-Umwandlung keine Ausgabefeldfehler erzeugt. Wenn die Java-Umwandlung Feldfehler erzeugen kann, kann "Early Selection" vom Data Integration Service nicht verwendet werden.

```
public boolean canGenerateOutputFieldEvalError() {
    // If this Java transformation can never generate an output field evaluation error,
    // return false.
    return true;
}
```

getInputExpr

Gibt einen Informatica-Ausdruck zurück, der die Eingabewerte aus Eingabefeldern beschreibt, die ein Ausgabefeld enthalten. Der Optimierer muss wissen, welche Eingabefelder ein Ausgabefeld enthalten, um die Filterlogik durch die Umwandlung zu leiten.

```
public InfaExpression getInputExpr(TransformationField field,
        TransformationDataInterface group) {
    // This should return an Informatica expression for output fields in terms of input
fields
    // We will only push predicate that use fields for which input expressions are
defined.
    // For example, if you have two input fields in0 and in1 and three output fields
out0, out1, out2
    // out0 is the pass-through of in1, out2 is sum of in1 and in2, and out3 is unknown,
the code should be:
    //if (field.getName().equals("out0"))
    //  return new InfaExpression("in0", instance);
    //else if (field.getName().equals("out1"))
    //    return new InfaExpression("in0 + in1", instance);
    //else if (field.getName().equals("out2"))
    //    return null;
    return null;
}
```

Ein Mapping enthält beispielsweise einen Filterausdruck, `"out0 > 8 "`. Out0 ist der Wert des out0-Ausgabeports in der Java-Umwandlung. Sie können den Wert out0 als den Wert des in0-Eingabeports + 5 definieren. Der Optimierer kann den folgenden Ausdruck `"(in0 + 5) > 8"` mit der "Early Selection"-Optimierung hinter die Java-Umwandlung verschieben. Sie können NULL zurückgeben, wenn ein Ausgabefeld nicht über einen Eingabefeldausdruck verfügt. Der Optimierer verschiebt Filterausdrücke nicht hinter Ausgabefelder ohne Eingabeausdruck.
Sie können gegebenenfalls folgenden Code aufnehmen:

```
if (field.getName().equals("out0"))
     return new InfaExpression("in0 + 5", instance);
else if (field.getName().equals("out2"))
     return null;
```

inputGroupsPushPredicateTo

Gibt eine Liste von Gruppen zurück, die die Filterlogik empfangen können. Die Java-Umwandlung verfügt über eine Eingabegruppe. Ändern Sie diese Funktion für die Java-Umwandlung nicht.

```
public List<TransformationDataInterface> inputGroupsPushPredicateTo(
        List<TransformationField> fields) {
    // This functions returns a list of input data interfaces to push predicates to.
    // Since JavaTx only has one input data interface, you should not have to modify
this function
    AbstractTransformation tx = instance.getTransformation();
    List<DataInterface> dis = tx.getDataInterfaces();
    List<TransformationDataInterface> inputDIs = new
ArrayList<TransformationDataInterface>();
    for (DataInterface di : dis){
        TransformationDataInterface tdi = (TransformationDataInterface) di;
        if (tdi.isInput())
            inputDIs.add(tdi);
    }
    if(inputDIs.size() == 1)
        return inputDIs;
    else
        return null;
}
```

Push-Into-Optimierung mit der Java-Umwandlung

Sie können eine aktive Java-Umwandlung für die push-into-Optimierung aktivieren, wenn sie keine Nebeneffekte aufweist und die Optimierung keine Auswirkungen auf die Mapping-Ergebnisse hat.

Wenn Sie die push-into-Optimierung für die Java-Umwandlung konfigurieren, definieren Sie für die Java-Umwandlung eine Möglichkeit zum Speichern der Filterbedingung, die sie aus dem Optimierer enthält. Fügen Sie Code hinzu, der die Filterbedingung überprüft. Wenn die Java-Umwandlung die Filterlogik absorbieren kann, gibt die Java-Umwandlung eine wahre Bedingung an den Optimierer zurück. Der Optimierer entfernt die Filterumwandlung aus dem optimierten Mapping.

Beim Konfigurieren der Java-Umwandlung schreiben Sie den Code, der die Filterbedingung während der Optimierung als Umwandlungsmetadaten speichert. Sie schreiben den Code auch zum Abrufen der Filterbedingung zur Laufzeit sowie zum Entfernen der Zeilen entsprechend der Filterlogik.

Beim Definieren der Java-Umwandlung fügen Sie Code für die Push-Into-Optimierung auf der Registerkarte **Optimierungsschnittstelle** der Java-Umwandlung hinzu. Wählen Sie für den Zugriff auf Codeausschnitte für die Push-Into-Optimierung "FilterPushdownOptimization" im Navigator auf der Registerkarte **Optimierungsschnittstelle** der Umwandlung aus.

Das Developer Tool zeigt Codeausschnitte an, um Push-Into-Optimierung zu aktivieren und die Filterbedingung aus dem Optimierer abzurufen. Aktualisieren Sie die Codeausschnitte zum Aktivieren der Optimierung sowie zum Speichern der Filterlogik als Umwandlungsmetadaten.

isFilterSupported

Gibt TRUE zurück, um Push-Into-Optimierung zu aktivieren. Gibt FALSE zurück, um Push-Into-Optimierung zu deaktivieren.
Ändern Sie die Funktion zur Rückgabe von TRUE, um Push-Into-Optimierung zu aktivieren.

```
public ResultAndMessage isFilterSupported() {
    // To enable filter push-into optimization this function should return true
    // return new ResultAndMessage(true, "");
    return new ResultAndMessage(false, "Filter push-into optimization is not supported");
}
```

pushFilter

Erhält die Filterbedingung vom Optimierer.

Fügen Sie Code hinzu, um den Filter zu überprüfen und anzugeben, ob die Filterlogik in der Umwandlung verwendet werden kann. Wenn die Umwandlung den Filter absorbieren kann, verwenden Sie die folgende Methode zum Speichern der Filterbedingung als Umwandlungsmetadaten:

```
storeMetadata(String key, String data)
```

Der Schlüssel ist ein Bezeichner für die Metadaten. Sie können jeden String als Schlüssel definieren. Die Daten sind die Daten, die Sie speichern möchten, um festzustellen, welche Zeilen zur Laufzeit entfernt werden sollen. Beispielsweise kann es sich bei den Daten um die Filterbedingung handeln, die die Java-Umwandlung vom Optimierer erhält.

```
public ResultAndMessage pushFilter(InfaExpression condition) {

    // Add code to absorb the filter
    // If filter is successfully absorbed return new ResultAndMessage(true, ""); and the
optimizer
    // will remove the filter from the mapping
    // If the filter is not absorbed, return new ResultAndMessage(false, msg);
    return new ResultAndMessage(false, "Filter push-into optimization is not supported");
}
```

Erstellen einer Java-Umwandlung

Im Developer Tool können Sie eine wiederverwendbare oder nicht wiederverwendbare Java-Umwandlung erstellen.

Erstellen einer wiederverwendbaren Java-Umwandlung

Wiederverwendbare Umwandlungen können in mehreren Mappings vorkommen.

Erstellen Sie eine wiederverwendbare Java-Umwandlung im Developer Tool.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die Java-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Geben Sie einen Namen für die Umwandlung ein.
6. Wählen Sie zum Erstellen einer aktiven Umwandlung die Option **Als aktiv erstellen** aus.
7. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.
8. Klicken Sie in der **Ports**-Ansicht auf die Schaltfläche **Neu**, um der Umwandlung einen Port hinzuzufügen.
9. Bearbeiten Sie den Port, um den Namen, den Datentyp und die Präzision zu bearbeiten.

   Verwenden Sie Portnamen als Variablen in Java-Codeausschnitten.
10. Verwenden Sie in der **Java**-Ansicht die Codeeingabe-Registerkarten zum Schreiben und Kompilieren des Java-Codes für die Umwandlung.
11. Verwenden Sie in der **Java**-Ansicht die Registerkarte **Funktionen**, um Funktionen zu definieren, die Ausdrücke aufrufen.

12. Doppelklicken Sie auf einer Codeeingabe-Registerkarte auf Fehlermeldungen im Fenster **Ergebnisse** in den Eigenschaften für **Kompilierung**, um Kompilierungsfehler im Java-Code für die Umwandlung zu suchen und zu beheben.
13. Bearbeiten Sie in der Ansicht **Erweitert** die Umwandlungseigenschaften.

Erstellen einer nicht wiederverwendbaren Java-Umwandlung

Nicht wiederverwendbare Umwandlungen sind in einem einzelnen Mapping vorhanden.

Erstellen Sie eine nicht wiederverwendbare Java-Umwandlung im Developer-Tool.

1. Ziehen Sie in einem Mapping oder Mapplet eine Java-Umwandlung aus der Umwandlungspalette auf den Editor.
2. Geben Sie im Dialogfeld **Neue Java-Umwandlung** einen Namen für die Umwandlung ein.
3. Wählen Sie zum Erstellen einer aktiven Umwandlung die Option **Als aktiv erstellen** aus.
4. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.
5. Auf der Registerkarte **Allgemein**, bearbeiten Sie den Umwandlungsnamen und die Beschreibung.
6. Klicken Sie auf der Registerkarte **Ports** auf die Schaltfläche **Neu**, um der Umwandlung einen Port hinzuzufügen.
7. Bearbeiten Sie den Port, um den Namen, den Datentyp und die Genauigkeit festzulegen.

   Verwenden Sie Portnamen als Variablen in Java-Codeausschnitten.
8. Verwenden Sie in der Ansicht **Java** die Codeeingabe-Registerkarten zum Schreiben und Kompilieren des Java-Codes für die Umwandlung.
9. Verwenden Sie in der Ansicht **Java** die Registerkarte **Funktionen**, um Funktionen zu definieren, die Ausdrücke aufrufen.
10. Doppelklicken Sie auf einer Codeeingabe-Registerkarte auf Fehlermeldungen im Fenster **Ergebnisse** in den Eigenschaften für **Kompilierung**, um Kompilierungsfehler im Java-Code für die Umwandlung zu suchen und zu beheben.
11. Bearbeiten Sie in der Ansicht **Erweitert** die Umwandlungseigenschaften.

Kompilieren einer Java-Umwandlung

Developer Tool verwendet den Java-Compiler zum Kompilieren von Java-Code und Generieren von Bytecode für die Umwandlung.

Der Java-Compiler kompiliert den Java-Code und zeigt die Ergebnisse der Kompilierung im Fenster **Ergebnisse** in den Eigenschaften für **Kompilierung** auf den Codeeingabe-Registerkarten an. Der Java-Compiler wird zusammen mit Developer Tool im Verzeichnis `java/bin` installiert.

Klicken Sie zum Kompilieren des vollständigen Codes für die Java-Umwandlung auf **Kompilieren** in den Eigenschaften für **Kompilierung** auf der Registerkarte **Full-Code**.

Beim Erstellen einer Java-Umwandlung enthält diese eine Java-Klasse, die die Basisfunktion für eine Java-Umwandlung definiert. Der vollständige Code für die Java-Klasse enthält den Vorlagen-Klassencode für die Umwandlung sowie den Java-Code, den Sie auf den Codeeingabe-Registerkarten definieren.

Wenn Sie eine Java-Umwandlung kompilieren, fügt Developer Tool den Code aus den Codeeingabe-Registerkarten der Vorlagenklasse für die Umwandlung hinzu, um den vollständigen Klassencode für die Umwandlung zu generieren. Developer Tool ruft anschließend den Java-Compiler auf, um den vollständigen Klassencode zu kompilieren. Der Java-Compiler kompiliert die Umwandlung und generiert den Bytecode für die Umwandlung.

Die Ergebnisse der Kompilierung werden im Fenster **Ergebnisse** angezeigt. Verwenden Sie die Ergebnisse der Kompilierung, um Java-Codefehler zu bestimmen und zu suchen.

Fehlerbehebung bei einer Java-Umwandlung

Im Fenster **Ergebnisse** in den **Kompilierung**seigenschaften auf allen Codeeingabe-Registerkarten können Sie Fehler im Java-Code finden und beheben.

Fehler in einer Java-Umwandlung können aufgrund eines Fehlers im Code auf einer Codeeingabe-Registerkarte oder im gesamten Code für die Java-Umwandlungsklasse auftreten.

Zur Behebung von Fehlern in einer Java-Umwandlung führen Sie die folgenden abstrakten Schritte aus:

1. Suchen Sie im Java-Ausschnittscode oder im vollständigen Klassencode für die Umwandlung nach der Fehlerquelle.
2. Identifizieren Sie den Fehlertyp. Verwenden Sie die Ergebnisse der Kompilierung im Fenster **Ergebnisse** und die Position des Fehlers, um den Fehlertyp zu identifizieren.
3. Beheben Sie den Java-Code auf der Codeeingabe-Registerkarte.
4. Kompilieren Sie die Umwandlung erneut.

Suchen der Quelle bei Kompilierungsfehlern

Verwenden Sie zum Suchen der Quelle von Kompilierungsfehlern die Ergebnisse der Kompilierung, die im Fenster **Ergebnisse** in den Eigenschaften für **Kompilierung** auf der Codeeingabe-Registerkarte oder der Registerkarte **Full-Code** angezeigt werden.

Wenn Sie im Fenster **Ergebnisse** auf eine Fehlermeldung doppelklicken, wird der Quellcode, der den Fehler verursacht hat, im Fenster **Java-Code** auf der Codeeingabe-Registerkarte oder der Registerkarte **Full-Code** angezeigt.

Sie können Fehler auf der Registerkarte **Full-Code** suchen, Java-Code kann allerdings nicht auf der Registerkarte **Full-Code** bearbeitet werden. Um die auf der Registerkarte **Full-Code** gefundenen Fehler zu beheben, ändern Sie den Code auf der jeweiligen Codeeingabe-Registerkarte. Möglicherweise müssen Sie die Registerkarte **Full-Code** verwenden, um Fehler anzuzeigen, die durch das Hinzufügen von Benutzercode zum Full-Class-Code für die Umwandlung verursacht wurden.

Suchen eines Fehlers auf einer Codeeingabe- oder Full-Code-Registerkarte

Sie können Kompilierungsfehler auf einer Codeeingabe-Registerkarte oder auf der Registerkarte **Full-Code** suchen.

1. Klicken Sie mit der rechten Maustaste im Fenster **Ergebnisse** in den Eigenschaften **Kompilierung** auf einer beliebigen Codeeingabe-Registerkarte oder auf der Registerkarte **Full-Code** auf eine Fehlermeldung.

2. Klicken Sie entweder auf **Zeigen in** > **Snippet** oder **Zeigen** > **Registerkarte "Full-Code"**.

   Das Developer Tool markiert die Quelle des Fehlers auf der ausgewählten Registerkarte.

   Hinweis: Sie können Fehler auf der Registerkarte **Full-Code** anzeigen, aber nicht korrigieren. Um Fehler zu korrigieren, müssen Sie zur entsprechenden Codeeingabe-Registerkarte navigieren.

Erkennen der Ursache von Kompilierungsfehlern

Kompilierungsfehler können auf Fehler im Benutzercode zurückzuführen sein.

Fehler im Benutzercode können ebenfalls einen Fehler im Nicht-Benutzercode für die Klasse generieren. Kompilierungsfehler treten im Benutzer- und Nicht-Benutzercode für die Java-Umwandlung auf.

Fehler im Benutzercode

Fehler können im Benutzercode auf den Codeeingabe-Registerkarten auftreten. Fehler im Benutzercode umfassen standardmäßige Java-Syntax- und -Sprachfehler.

Fehler im Benutzercode können auch auftreten, wenn das Developer Tool des Benutzercode aus den Codeeingabe-Registerkarten zum vollständigen Klassencode hinzufügt.

Eine Java-Umwandlung verfügt beispielsweise über einen Eingabeport mit der Bezeichnung "int1" und einem Ganzzahldatentyp. Der vollständige Code für die Klasse deklariert die Eingabeportvariable mit folgendem Code:

```
int int1;
```

Wenn Sie jedoch den gleichen Variablenname auf der Registerkarte **Bei Eingabe** verwenden, gibt der Java-Compiler einen Fehler für eine erneute Deklaration einer Variable aus. Benennen Sie zum Beheben des Fehlers die Variable auf der Registerkarte **Bei Eingabe** um.

Nicht-Benutzercode-Fehler

Der Benutzercode auf den Codeeingabe-Registerkarten kann zu Fehlern im Nicht-Benutzercode führen.

Beispielsweise hat eine Java-Umwandlung einen Eingabeport und einen Ausgabeport (int1 und out1) mit Ganzzahl-Datentypen. Schreiben Sie den folgenden Code in die Codeeingabe-Registerkarte **Bei Eingabe**, um "interest" für Eingabeport int1 zu berechnen und diesen zum Ausgabeport out1 hinzuzufügen:

```
int interest;
interest = CallInterest(int1); // calculate interest
out1 = int1 + interest;
}
```

Wenn Sie die Umwandlung kompilieren, fügt das Developer Tool den Code aus der Codeeingabe-Registerkarte **Bei Eingabe** dem Full-Class-Code für die Umwandlung hinzu. Wenn der Java-Compiler den Java-Code kompiliert, führt dies dazu, dass eine Methode im Full-Class-Code aufgrund der nicht übereinstimmenden Klammer vorzeitig abgebrochen wird und der Java-Compiler gibt einen Fehler aus.

KAPITEL 18

Java-Umwandlungs-API-Referenz

Dieses Kapitel umfasst die folgenden Themen:

- Java-Umwandlungs-API-Methoden - Übersicht, 287
- defineJExpression, 288
- failSession, 289
- generateRow, 289
- getInRowType, 290
- getMetadata, 291
- incrementErrorCount, 291
- invokeJExpression, 292
- isNull, 293
- logError, 294
- logInfo, 294
- resetNotification, 295
- setNull, 295
- storeMetadata, 296

Java-Umwandlungs-API-Methoden - Übersicht

Auf den Codeeingabe-Registerkarten in der Ansicht **Java** im Editor können Sie API-Methoden zum Java-Code hinzufügen, um das Umwandlungsverhalten zu definieren.

Um dem Code eine API-Methode hinzuzufügen, erweitern Sie die Liste **Aufrufbare APIs** im Navigator auf der Codeeingabe-Registerkarte und doppelklicken Sie auf den Namen der Methode, die Sie zum Code hinzufügen möchten.

Alternativ dazu können Sie die Methode aus dem Navigator in den Java-Codeausschnitt ziehen oder die API-Methode manuell in den Java-Codeausschnitt eingeben.

Sie können die folgenden API-Methoden zum Java-Code in einer Java-Umwandlung hinzufügen:

defineJExpression

Definiert einen Java-Ausdruck.

failSession

Gibt eine Ausnahme mit einer Fehlermeldung aus und das Mapping schlägt fehl.

generateRow

Generiert eine Ausgabezeile für aktive Java-Umwandlungen.

getInRowType

Gibt den Eingabetyp der aktuellen Zeile in der Umwandlung zurück.

incrementErrorCount

Erhöht den Fehlerzähler für das Mapping.

invokeJExpression

Ruft einen Java-Ausdruck auf, den Sie mit der defineJExpression-Methode definiert haben.

isNull

Sucht nach einem Nullwert in einer Eingabespalte.

logError

Schreibt eine Fehlermeldung in das sprotokoll.

logInfo

Schreibt eine Informationsmeldung in das sprotokoll.

resetNotification

Wenn der Data Integration Service-Computer im Neustartmodus ausgeführt wird, werden die Variablen, die Sie im Java-Code nach einer Mapping-Ausführung verwenden, zurückgesetzt.

setNull

Legt den Wert einer Ausgabespalte in einer aktiven oder passiven Java-Transaktion auf Null fest.

defineJExpression

Definiert einen Ausdruck, der unter anderem den Ausdrucksstring und Eingabeparameter enthält. Argumente für die defineJExpression-Methode umfassen ein Array von JExprParamMetadata-Objekten, die die Eingabeparameter und einen Stringwert enthalten, die die Ausdruckssyntax definieren.

Verwenden Sie folgende Syntax:

```
defineJExpression(
    String expression,
    Object[] paramMetadataArray
    );
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Typ	Datentyp	Beschreibung
expression	Eingabe	String	String, der den Ausdruck darstellt.
paramMetadataArray	Eingabe	Object[]	Array von JExprParaMetadata-Objekten, die den Eingabeparameter für den Ausdruck enthalten.

Sie können die defineJExpression-Methode zum Java-Code auf einer beliebigen Codeeingabe-Registerkarte hinzufügen. Ausnahmen dazu stellen die Registerkarten **Importe** und **Funktionen** dar.

Um die defineJExpression-Methode zu verwenden, müssen Sie ein Array von JExprParamMetadata-Objekten instanziieren, die die Eingabeparameter für den Ausdruck darstellen. Sie legen die Metadatenwerte für die Parameter fest und leiten das Array als Parameter an die defineJExpression-Methode weiter.

Der folgende Java-Code erstellt beispielsweise einen Ausdruck, um den Wert von zwei Strings nachzuschlagen:

```
JExprParaMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
defineJExpression(":lkp.mylookup(x1,x2)",params);
```

Hinweis: Sie müssen die Parameter nummerieren, die Sie nacheinander an den Ausdruck übergeben, und den Parameter mit dem Buchstaben x beginnen. Um beispielsweise drei Parameter an den Ausdruck zu übergeben, benennen Sie die Parameter mit x1, x2 und x3.

failSession

Gibt eine Ausnahme mit einer Fehlermeldung aus und das Mapping schlägt fehl.

Verwenden Sie folgende Syntax:

```
failSession(String errorMessage);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
errorMessage	Eingabe	String	Fehlermeldungsstring.

Verwenden Sie die failSession-Methode zum Beenden des Mappings. Verwenden Sie nicht die failSession-Methode in einem Try-/Catch-Block auf einer Codeeingabe-Registerkarte.

Sie können die failSession-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie der input1-Eingabeport hinsichtlich eines Nullwertes getestet und das Fehlschlagen des Mappings herbeigeführt wird, wenn es sich um einen Nullwert handelt.

```
if(isNull(”input1”)) {
     failSession(“Cannot process a null value for port input1.”);
}
```

generateRow

Generiert eine Ausgabezeile für aktive Java-Umwandlungen.

Verwenden Sie folgende Syntax:

```
generateRow();
```

Wenn Sie die generateRow-Methode aufrufen, generiert die Java-Umwandlung eine Ausgabezeile mit dem aktuellen Wert der Ausgabeportvariablen. Wenn Sie mehrere Zeilen entsprechend einer Eingabezeile generieren möchten, können Sie die generateRow-Methode mehr als ein Mal für jede Eingabezeile aufrufen.

Wenn Sie die generateRow-Methode nicht in einer aktiven Java-Umwandlung verwenden, generiert die Umwandlung keine Ausgabezeilen.

Sie können die generateRow-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Sie können die generateRow-Methode nur in aktiven Umwandlungen verwenden. Wenn Sie die generateRow-Methode in einer passiven Umwandlung verwenden, generiert der Data Integration Service einen Fehler.

Verwenden Sie den folgenden Java-Code, um eine Ausgabezeile zu generieren, die Werte der Ausgabeports zu ändern und eine weitere Ausgabezeile zu generieren.

```
// Generate multiple rows.
if(!isNull("input1") && !isNull("input2"))
{
     output1 = input1 + input2;
     output2 = input1 - input2;
}
generateRow();
// Generate another row with modified values.
output1 = output1 * 2;
output2 = output2 * 2;
generateRow();
```

getInRowType

Gibt den Eingabetyp der aktuellen Zeile in der Umwandlung zurück. Die Methode gibt einen Wert zum Einfügen, Aktualisieren oder Ablehnen zurück.

Verwenden Sie folgende Syntax:

```
rowType getInRowType();
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
rowType	Ausgabe	String	Gibt den Update-Strategie-Typ zurück, der einer der folgenden Werte ist: - DELETE - INSERT - REJECT - UPDATE

Sie können die getInRowType-Methode zum Java-Code auf der Codeeingabe-Registerkarte **Bei Eingabe** hinzufügen.

Sie können die getInRowType-Methode in aktiven Umwandlungen verwenden, die zum Festlegen der Update-Strategie konfiguriert wurden. Wenn Sie diese Methode in einer aktiven Umwandlung verwenden, die nicht zum Festlegen der Update-Strategie konfiguriert wurde, generiert der Data Integration Service einen Fehler.

getMetadata

Ruft Java-Umwandlungsmetadaten zur Laufzeit ab. Die Methode getMetadata ruft Metadaten ab, die mit der Methode storeMetadata gespeichert werden, wie z. B. eine Filterbedingung, die vom Optimierer in der Funktion pushFilter an die Java-Umwandlung übergeben wird.

Verwenden Sie folgende Syntax:

```
getMetadata (String key);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
key	Eingabe	String	Gibt die Metadaten an. Die Methode getmetadata verwendet den Schlüssel, um festzustellen, welche Metadaten abgerufen werden müssen.

Sie können die Methode getMetadata auf den folgenden Codeeingabe-Registerkarten zum Java-Code hinzufügen:

- Helper
- Bei Eingabe
- Am Datenende
- Optimierungsschnittstelle
- Funktionen

Sie können die Methode getMetadata konfigurieren, um Filterbedingungen für die Push-Into-Optimierung abzurufen. Die Methode getMetadata kann jede Filterbedingung abrufen, die Sie aus dem Optimierer gespeichert haben.

```
// Retrieve a filter condition
String mydata = getMetadata ("FilterKey");
```

incrementErrorCount

Erhöht den Fehlerzähler. Wenn der Fehlerzähler die Fehlerschwelle erreicht hat, schlägt das Mapping fehl.

Verwenden Sie folgende Syntax:

```
incrementErrorCount(int nErrors);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
nErrors	Eingabe	Integer	Zahl, um die der Fehlerzähler erhöht wird.

Sie können die incrementErrorCount-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie der Fehlerzähler erhöht wird, wenn ein Eingabeport für eine Umwandlung einen Nullwert hat:

```
// Check if input employee id and name is null.
if (isNull ("EMP_ID_INP") || isNull ("EMP_NAME_INP"))
{
     incrementErrorCount(1);
     // if input employee id and/or name is null, don't generate a output row for this
input row
     generateRow = false;
}
```

invokeJExpression

Ruft einen Ausdruck auf und gibt den Wert für den Ausdruck zurück.

Verwenden Sie folgende Syntax:

```
(datatype)invokeJExpression(
                      String expression,
                      Object[] paramMetadataArray);
```

Eingabeparameter für die invokeJExpression-Methode sind ein Stringwert, der den Ausdruck und ein Array von Objekten darstellt, in denen die Ausdruckseingabeparameter enthalten sind.

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
expression	Eingabe	String	String, der den Ausdruck darstellt.
paramMetadataArray	Eingabe	Object[]	Array von Objekten, in denen die Eingabeparameter für den Ausdruck enthalten sind.

Sie können die invokeJExpression-Methode zum Java-Code auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** hinzufügen.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie mit die invokeJExpression-Methode verwenden:

- Rückgabe-Datentyp. Der Rückgabe-Datentyp der invokeJExpression-Methode ist ein Objekt. Sie müssen den Rückgabewert der Funktion mit dem entsprechenden Datentyp umwandeln.

  Sie können Werte mit den Datentypen Integer, Double, String und byte[] zurückgeben.
- Zeilentyp. Der Zeilentyp für Rückgabewerte aus der invokeJExpression-Methode ist INSERT.

  Einen anderen Zeilentyp für den Rückgabewert können Sie mit der erweiterten Schnittstelle verwenden.
- Nullwerte. Wenn Sie einen Nullwert als Parameter weiterleiten oder der Rückgabewert für die invokeJExpression-Methode ist NULL, wird der Wert als Null-Indikator behandelt.

  Wenn der Rückgabewert eines Ausdrucks beispielsweise NULL ist und der Rückgabedatentyp ist String, wird ein String mit einem Wert von Null zurückgegeben.

- Date-Datentyp. Sie müssen Eingabeparameter mit einem Date-Datentyp in den String-Datentyp umwandeln.

  Um den String in einem Ausdruck als Date-Datentyp verwenden zu können, verwenden Sie die to_date()-Funktion, um den String in einen Date-Datentyp umzuwandeln.

  Außerdem müssen Sie den Rückgabetyp eines beliebigen Ausdrucks umwandeln, der einen Date-Datentyp als einen String-Datentyp zurückgibt.

Das folgende Beispiel verkettet die Strings "John" und "Smith" und gibt den String "John Smith" zurück:

```
(String)invokeJExpression("concat(x1,x2)", new Object [] { "John ", "Smith" });
```

Hinweis: Sie müssen die Parameter nummerieren, die Sie nacheinander an den Ausdruck übergeben, und den Parameter mit dem Buchstaben x beginnen. Um beispielsweise drei Parameter an den Ausdruck zu übergeben, benennen Sie die Parameter mit x1, x2 und x3.

isNull

Prüft den Wert einer Eingabespalte für einen Nullwert.

Verwenden Sie folgende Syntax:

```
Boolean isNull(String satrColName);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
strColName	Eingabe	String	Name einer Eingabespalte.

Sie können die isNull-Methode auf der Codeeingabe-Registerkarte **Bei Eingabe** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie Sie prüfen können, ob der Wert der SALARY-Eingabespalte Null ist, bevor er zur totalSalaries-Instanzvariablen hinzugefügt wird:

```
// if value of SALARY is not null
if (!isNull("SALARY")) {
     // add to totalSalaries
     TOTAL_SALARIES += SALARY;
}
```

Verwenden Sie alternativ dazu den folgenden Java-Code, um dieselben Ergebnisse zu erzielen:

```
// if value of SALARY is not null
String strColName = "SALARY";
if (!isNull(strColName)) {
     // add to totalSalaries
     TOTAL_SALARIES += SALARY;
}
```

logError

Schreibt eine Fehlermeldung in das sprotokoll.

Verwenden Sie folgende Syntax:

```
logError(String msg);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
msg	Eingabe	String	Fehlermeldungsstring.

Sie können die logError-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie ein Fehler protokolliert wird, wenn der Eingabeport Null ist:

```
// check BASE_SALARY
if (isNull("BASE_SALARY")) {
     logError("Cannot process a null salary field.");
}
```

Wenn der Code ausgeführt wird, wird die folgende Meldung im sprotokoll angezeigt:

```
[JTX_1013] [ERROR] Cannot process a null salary field.
```

logInfo

Schreibt eine Informationsmeldung in das sprotokoll.

Verwenden Sie folgende Syntax:

```
logInfo(String msg);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
msg	Eingabe	String	Informationsmeldungsstring.

Sie können die logInfo-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie eine Meldung in das sprotokoll geschrieben wird, nachdem die Java-Umwandlung einen Meldungsschwellenwert von 1000 Zeilen verarbeitet hat:

```
if (numRowsProcessed == messageThreshold) {
     logInfo("Processed " + messageThreshold + " rows.");
}
```

resetNotification

Wenn der Data Integration Service-Computer im Neustartmodus ausgeführt wird, werden die Variablen, die Sie im Java-Code nach einer Mapping-Ausführung verwenden, zurückgesetzt.

Im Neustartmodus ist der Data Integration Service nicht deinitialisiert, wird jedoch nach einer Anfrage zurückgesetzt, damit der Data Integration Service die nächste Anfrage verarbeiten kann.

Verwenden Sie für eine Java-Umwandlung die resetNotification-Methode, um Variablen im Java-Code nach einer Mapping-Ausführung zurückzusetzen.

Verwenden Sie folgende Syntax:

```
public int resetNotification(IGroup group) {
    return EStatus.value;
}
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
int	Ausgabe	EStatus.*value*	Rückgabewert, wobei *value* einer der folgenden Werte ist: - `SUCCESS`. Success. - `FAILURE`. Failure. - `NOIMPL`. Nicht implementiert.
group	Eingabe	IGroup	Die Eingabegruppe.

Sie können die resetNotification-Methode auf der Codeeingabe-Registerkarte auf der Registerkarte **Helfer** zum Java-Code hinzufügen.

Die resetNotification-Methode wird in der Liste "Aufrufbare APIs" nicht angezeigt.

Wenn der Java-Code beispielsweise eine statische Variable namens out5_static deklariert und sie auf 1 initialisiert, setzt der folgende Java-Code die out5_static-Variable nach der nächsten Mapping-Ausführung auf 1:

```
public int resetNotification(IGroup group) {
    out5_static=1;
    return EStatus.SUCCESS;
}
```

Diese Methode ist nicht erforderlich. Wenn der Data Integration Service jedoch im Neustartmodus ausgeführt wird und das Mapping eine Java-Umwandlung enthält, die die resetNotification-Methode nicht implementiert, wird die JSDK_42075-Warnmeldung im Protokoll angezeigt.

setNull

Setzt den Wert einer Ausgabespalte in einer aktiven oder passiven Java-Transaktion auf Null.

Verwenden Sie folgende Syntax:

```
setNull(String strColName);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
strColName	Eingabe	String	Name einer Ausgabespalte.

Die setNull-Methode setzt den Wert einer Ausgabespalte in einer aktiven oder passiven Java-Umwandlung auf Null. Nachdem Sie eine Ausgabespalte auf Null gesetzt haben, können Sie den Wert erst dann ändern, wenn Sie eine Ausgabezeile generieren.

Sie können die setNull-Methode auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** zum Java-Code hinzufügen.

Der folgende Java-Code zeigt, wie Sie den Wert einer Eingabespalte prüfen und den entsprechenden Wert der Ausgabespalte auf Null festlegen:

```
// check value of Q3RESULTS input column
if(isNull("Q3RESULTS")) {
     // set the value of output column to null
     setNull("RESULTS");
}
```

Alternativ dazu können Sie den folgenden Java-Code verwenden, um dieselben Ergebnisse zu erzielen:

```
// check value of Q3RESULTS input column
String strColName = "Q3RESULTS";
if(isNull(strColName)) {
     // set the value of output column to null
     setNull(strColName);
}
```

storeMetadata

Speichert Metadaten der Java-Umwandlung, die zur Laufzeit mit der getMetadata-Methode abgerufen werden können.

Verwenden Sie folgende Syntax:

```
storeMetadata (String key String data);
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
key	Eingabe	String	Gibt die Metadaten an. Die storeMetadata-Methode benötigt einen Schlüssel zum Angeben der Metadaten. Definieren Sie den Schlüssel als beliebigen Sting.
Daten	Eingabe	String	Die Daten, die als Metadaten der Java-Umwandlung gespeichert werden sollen.

Sie können dem Java-Code die storeMetadata-Methode auf folgenden Codeeingabe-Registerkarten hinzufügen:

- Helper
- Bei Eingabe

- Am Datenende
- Optimierungsschnittstelle
- Funktionen

Sie können die storeMetadata-Methode in einer aktiven Umwandlung so konfigurieren, dass Filterbedingungen für die Push-Into-Optimierung übernommen werden. Die storeMetadata-Methode speichert eine Filterbedingung, die vom Optimierer aus dem Mapping in die Java-Umwandlung verschoben wird.

```
// Store a filter condition
storeMetadata ("FilterKey", condition);
```

KAPITEL 19

Java-Ausdrücke

Dieses Kapitel umfasst die folgenden Themen:

- Java-Ausdrücke - Übersicht, 298
- Verwenden des Dialogfelds Funktion definieren zum Definieren eines Ausdrucks, 299
- Arbeiten mit der einfachen Schnittstelle, 301
- Arbeiten mit der erweiterten Schnittstelle, 303
- API-Referenz für JExpression-Klasse, 307

Java-Ausdrücke - Übersicht

Sie können -Ausdrücke in einer Java-Umwandlung mit der Java-Programmiersprache aufrufen.

Verwenden Sie Ausdrücke, um die Funktionalität einer Java-Umwandlung zu erweitern. Sie können beispielsweise ein Ausdruck in einer Java-Umwandlung aufrufen, um die Werte der Eingabe- oder Ausgabeports oder die Werte von Java-Umwandlungsvariablen zu suchen.

Zum Aufrufen von Ausdrücken in einer Java-Umwandlung generieren Sie die Java-Code- oder Java-Umwandlungs-API-Methoden. Sie rufen den Ausdruck auf und verwenden das Ergebnis des Ausdrucks auf der entsprechenden Codeeingabe-Registerkarte. Mithilfe der API-Methoden können Sie den Java-Code schreiben, der die Expression aufruft, oder Sie können den Java-Code generieren, der eine Expression aufruft.

In der folgenden Tabelle werden die Methoden beschrieben, die Sie zum Erstellen und Aufrufen von Ausdrücken in einer Java-Umwandlung verwenden können:

Methode	Beschreibung
Funktion definieren	Ermöglicht Ihnen das Erstellen einer Funktion, die einen Ausdruck aufruft, und das Generieren des Codes für einen Ausdruck.
Einfache Schnittstelle	Ermöglicht Ihnen das Aufrufen einer einzelnen API-Methode, um einen Ausdruck aufzurufen und das Ergebnis des Ausdrucks abzurufen.
Erweiterte Schnittstelle	Ermöglicht Ihnen, den Ausdruck zu definieren, den Ausdruck abzurufen und das Ergebnis des Ausdrucks zu verwenden. Wenn Sie mit objektorientierter Programmierung vertraut sind und mehr Kontrolle über das Aufrufen des Ausdrucks möchten, verwenden Sie die erweiterte Schnittstelle.

Ausdrucksfunktionstypen

Sie können Ausdrücke für eine Java-Umwandlung oder das Dialogfeld **Funktion definieren** oder die einfache oder erweiterte Schnittstelle verwenden.

Sie können Ausdrücke eingeben, die Eingabe- oder Ausgabeportvariablen oder Variablen im Java-Code als Eingabeparameter verwenden.

Wenn Sie das Dialogfeld **Funktion definieren** verwenden, können Sie den Ausdruck validieren, bevor Sie ihn in einer Java-Umwandlung verwenden.

Sie können die folgenden Typen von Ausdrucksfunktionen in einer Java-Umwandlung aufrufen:

Ausdrucksfunktionstyp	Beschreibung
Funktionen für die Umwandlungssprache	SQL-ähnliche Funktionen zur Verarbeitung von allgemeinen Ausdrücken.
Benutzerdefinierte Funktionen	Funktionen, die Sie in mit dem Developer Tool basierend auf den Funktionen für die Umwandlungssprache erstellen.
Benutzerspezifische Funktionen	Funktionen, die Sie mit der API für benutzerspezifische Funktionen erstellen.

Sie können auch nicht verbundene Umwandlungen und integrierte Variablen in Ausdrücke verwenden. Beispielsweise können Sie eine nicht verbundene Lookup-Umwandlung in einem Ausdruck verwenden.

Verwenden des Dialogfelds Funktion definieren zum Definieren eines Ausdrucks

Wenn Sie einen Java-Ausdruck definieren, konfigurieren Sie die Funktion, erstellen den Ausdruck und generieren den Code, der den Ausdruck aufruft.

Im Dialogfeld **Funktion definieren** können Sie die Funktion definieren und den Ausdruck erstellen.

Um eine Ausdrucksfunktion zu erstellen und den Ausdruck in einer Java-Umwandlung zu verwenden, führen Sie folgende hochrangigen Aufgaben aus:

1. Konfigurieren Sie die Funktion, die den Ausdruck aufruft, einschließlich Funktionsname, Beschreibung und Parameter. Verwenden Sie die Funktionsparameter, wenn Sie den Ausdruck erstellen.
2. Erstellen Sie die Ausdruckssyntax, und validieren Sie den Ausdruck.
3. Generieren Sie den Java-Code, der den Ausdruck aufruft.

   Der Developer speichert den Code auf der Codeeingabe-Registerkarte **Funktionen**.

Rufen Sie nach dem Generieren des Java-Codes die generierte Funktion auf der entsprechenden Codeeingabe-Registerkarte auf, um einen Ausdruck aufzurufen oder ein JExpression-Objekt abzurufen, abhängig davon, ob Sie die einfache oder erweiterte Schnittstelle verwenden.

Hinweis: Um einen Ausdruck beim Erstellen zu validieren, müssen Sie das Dialogfeld **Funktion definieren** verwenden.

Schritt 1. Konfigurieren der Funktion

Sie konfigurieren den Funktionsnamen, die Beschreibung sowie die Eingabeparameter für die Java-Funktion, die den Ausdruck aufruft.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie die Funktion konfigurieren:

- Verwenden Sie einen eindeutigen Funktionsnamen, der nicht mit einer vorhandenen Java-Funktion in der Umwandlung oder mit reservierten Java-Schlüsselwörtern in Konflikt steht.
- Müssen Sie den Parameternamen, den Java Datentyp, die Präzision und die Skalierung konfigurieren. Bei den Eingabeparametern handelt es sich um die Werte, die übergeben werden, wenn Sie die Funktion im Java-Code für die Umwandlung aufrufen.
- Verwenden Sie zum Übergeben des Date-Datentyps an einen Ausdruck, den String-Datentyp für den Eingabeparameter.

  Wenn ein Ausdruck den Date-Datentyp zurückgibt, können Sie den Rückgabewert als den String-Datentyp in der einfachen Schnittstelle und den String- oder langen Datentyp in der erweiterten Schnittstelle verwenden.

Schritt 2. Erstellen und Validieren des Ausdrucks

Verwenden Sie beim Erstellen des Ausdrucks die Parameter, die Sie für die Funktion konfiguriert haben.

Sie können auch Funktionen der Umwandlungssprache, benutzerdefinierte Funktionen oder andere benutzerdefinierte Funktionen im Ausdruck verwenden. Sie können den Ausdruck im Dialogfeld **Funktion definieren** im Dialogfeld .

Schritt 3. Generieren von Java-Code für den Ausdruck

Nach dem Konfigurieren der Funktion und Funktionsparameter sowie dem Definieren und Validieren des Ausdrucks können Sie den Java-Code generieren, der den Ausdruck aufruft.

Der Entwickler speichert den generierten Java-Code auf der Codeeingabe-Registerkarte **Funktionen**. Verwenden Sie den generierten Java-Code zum Ausführen der Funktionen, die den Ausdruck in den Codeeingabe-Registerkarten. Sie können einfachen oder erweiterten Java-Code generieren.

Nach dem Generieren des Java-Codes, der einen Ausdruck aufruft, können Sie den Ausdruck weder bearbeiten noch erneut validieren. Wenn Sie einen Ausdruck nach dem Generieren des Codes ändern möchten, müssen Sie den Ausdruck erneut erstellen.

Erstellen eines Ausdrucks und Generieren von Java-Code mit dem Dialogfeld Funktion definieren

Sie können eine Funktion erstellen, die einen Ausdruck im Dialogfeld **Funktion definieren** aufruft.

Führen Sie die folgenden Schritte zum Erstellen einer Funktion aus, die einen Ausdruck aufruft:

1. Öffnen Sie in Developer eine Java-Umwandlung oder erstellen Sie eine neue Java-Umwandlung.
2. Klicken Sie auf der Registerkarte **Java-Code** auf **Neue Funktion**.

   Das Dialogfeld **Funktion definieren** wird angezeigt.
3. Geben Sie einen Funktionsnamen ein.
4. Optional können Sie eine Beschreibung für den Ausdruck eingeben.

   Geben Sie bis zu 2.000 Zeichen ein.

5. Erstellen Sie die Argumente für die Funktion.

   Wenn Sie die Argumente erstellen, konfigurieren Sie das Argument, den Namen, den Datentyp, die Präzision und die Größenordnung.

6. Erstellen Sie auf der Registerkarte **Ausdruck** einen Ausdruck mit den von Ihnen erstellten Argumenten.
7. Klicken Sie zum Validieren des Ausdrucks auf **Validieren**.
8. Geben Sie optional den Ausdruck im Feld **Ausdruck** ein. Klicken Sie auf **Validieren**, um den Ausdruck zu validieren.
9. Wählen Sie zum Generieren von Java-Code mit der erweiterten Schnittstelle die Option **Erweiterten Code generieren** ein. Klicken Sie dann auf **Generieren**.

   Der Developer generiert die Funktion, um den Ausdruck auf der Codeeingabe-Registerkarte **Funktionen** aufzurufen.

Vorlagen für Java-Ausdrücke

Sie können Java-Code für einen Ausdruck mit dem einfachen oder erweiterten Java-Code für einen Ausdruck generieren.

Der Java-Code für den Ausdruck wird basierend auf den Vorlage für den Ausdruck generiert.

Das folgende Beispiel zeigt die Vorlage für einen Java-Ausdruck, der für einfachen Java-Code generiert wurde:

```
Object function_name (Java datatype x1[,
                                      Java datatype x2 ...] )
                                      throws SDK Exception
{
return (Object)invokeJExpression( String expression,
                                  new Object [] { x1[, x2, ... ]}  );
}
```

Das folgende Beispiel zeigt die Vorlage für einen Java-Ausdruck, der mit der erweiterten Schnittstelle erstellt wurde:

```
JExpression function_name ()  throws SDKException
{
    JExprParamMetadata params[] = new JExprParamMetadata[number of parameters];
    params[0] = new JExprParamMetadata (
                    EDataType.STRING,  // data type
                    20,  // precision
                    0    // scale
                    );
...
    params[number of parameters - 1] = new JExprParamMetadata (
                    EDataType.STRING,  // data type
                    20,  // precision
                    0    // scale
                    );
...
    return defineJExpression(String expression,params);
}
```

Arbeiten mit der einfachen Schnittstelle

Verwenden Sie die Java-API-Methode invokeJExpression, um einen Ausdruck in der einfachen Schnittstelle aufzurufen.

invokeJExpression

Ruft einen Ausdruck auf und gibt den Wert für den Ausdruck zurück.

Verwenden Sie folgende Syntax:

```
(datatype)invokeJExpression(
                      String expression,
                      Object[] paramMetadataArray);
```

Eingabeparameter für die invokeJExpression-Methode sind ein Stringwert, der den Ausdruck und ein Array von Objekten darstellt, in denen die Ausdruckseingabeparameter enthalten sind.

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Parametertyp	Datentyp	Beschreibung
expression	Eingabe	String	String, der den Ausdruck darstellt.
paramMetadataArray	Eingabe	Object[]	Array von Objekten, in denen die Eingabeparameter für den Ausdruck enthalten sind.

Sie können die invokeJExpression-Methode zum Java-Code auf einer beliebigen Codeeingabe-Registerkarte mit Ausnahme der Registerkarten **Importe** und **Funktionen** hinzufügen.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie mit die invokeJExpression-Methode verwenden:

- Rückgabe-Datentyp. Der Rückgabe-Datentyp der invokeJExpression-Methode ist ein Objekt. Sie müssen den Rückgabewert der Funktion mit dem entsprechenden Datentyp umwandeln.

  Sie können Werte mit den Datentypen Integer, Double, String und byte[] zurückgeben.
- Zeilentyp. Der Zeilentyp für Rückgabewerte aus der invokeJExpression-Methode ist INSERT.

  Einen anderen Zeilentyp für den Rückgabewert können Sie mit der erweiterten Schnittstelle verwenden.
- Nullwerte. Wenn Sie einen Nullwert als Parameter weiterleiten oder der Rückgabewert für die invokeJExpression-Methode ist NULL, wird der Wert als Null-Indikator behandelt.

  Wenn der Rückgabewert eines Ausdrucks beispielsweise NULL ist und der Rückgabedatentyp ist String, wird ein String mit einem Wert von Null zurückgegeben.
- Date-Datentyp. Sie müssen Eingabeparameter mit einem Date-Datentyp in den String-Datentyp umwandeln.

  Um den String in einem Ausdruck als Date-Datentyp verwenden zu können, verwenden Sie die to_date()-Funktion, um den String in einen Date-Datentyp umzuwandeln.

  Außerdem müssen Sie den Rückgabetyp eines beliebigen Ausdrucks umwandeln, der einen Date-Datentyp als einen String-Datentyp zurückgibt.

Hinweis: Sie müssen die Parameter nummerieren, die Sie nacheinander an den Ausdruck übergeben, und den Parameter mit dem Buchstaben x beginnen. Um beispielsweise drei Parameter an den Ausdruck zu übergeben, benennen Sie die Parameter mit x1, x2 und x3.

Einfache Schnittstelle - Beispiel

Sie können Expressions definieren und aufrufen, die die API-Methode invokeJExpression auf den Codeeingabe-Registerkarten **Helfer** und **Bei Eingabe** verwenden.

Das folgende Beispiel zeigt, wie Sie ein Lookup für die Eingabeports NAME und ADDRESS in einer Java-Umwandlung abschließen und den Rückgabewert dem Ausgabeport COMPANY_NAME zuweisen.

Geben Sie den folgenden Code auf der Codeeingabe-Registerkarte **Bei Eingabe** ein:

```
COMPANY_NAME = (String)invokeJExpression(":lkp.my_lookup(X1,X2)", new Object []
{str1 ,str2} );
generateRow();
```

Arbeiten mit der erweiterten Schnittstelle

In der erweiterten Schnittstelle können Sie objektorientierte API-Methoden zum Definieren, Aufrufen und Abfragen des Ergebnisses eines Ausdrucks verwenden.

Die folgende Tabelle beschreibt die Klassen und API-Methoden, die in der erweiterten Schnittstelle zur Verfügung stehen:

Klasse oder API-Methode	Beschreibung
EDataType-Klasse	Nummeriert die Datentypen für einen Ausdruck.
JExprParamMetadata-Klasse	Enthält die Metadaten für jeden Parameter in einem Ausdruck. Parametermetadaten umfassen den Datentyp, die Präzision und die Skalierung.
defineJExpression-API-Methode	Definiert den Ausdruck. Umfasst -Ausdrucksstrings und -Parameter.
invokeJExpression-API-Methode	Ruft einen Ausdruck auf.
JExpression-Klasse	Enthält die Methoden zum Erstellen, Aufrufen und Abfragen der Metadaten und des Ausdrucksergebnisses sowie zum Überprüfen des Rückgabedatentyps.

Aufrufen eines Ausdrucks mit der erweiterten Schnittstelle

Mit der erweiterten Schnittstelle können Sie einen Ausdruck definieren und aufrufen und die Ergebnisse eines Ausdrucks abrufen.

1. Erstellen Sie auf der Codeeingabe-Registerkarte **Helfer** oder **Bei Eingabe** eine Instanz der JExprParamMetadata-Klasse für jedes Argument für den Ausdruck und legen Sie den Wert der Metadaten fest. Optional können Sie das JExprParamMetadata-Objekt in der defineJExpression-Methode instanziieren.
2. Verwenden Sie die defineJExpression-Methode, um das JExpression-Objekt für den Ausdruck abzurufen.
3. Rufen Sie auf der entsprechenden Codeeingabe-Registerkarte den Ausdruck mit der invokeJExpression-Methode auf.
4. Prüfen Sie das Ergebnis des Rückgabewerts mit der isResultNull-Methode.
5. Sie können den Datentyp des Rückgabewerts oder die Metadaten des Rückgabewerts mit den Methoden getResultDataType und getResultMetadata abrufen.
6. Rufen Sie das Ergebnis des Ausdrucks mit der entsprechenden API-Methode ab. Sie können die Methoden getInt, getDouble, getStringBuffer und getBytes verwenden.

Regeln und Richtlinien für das Arbeiten mit der erweiterten Schnittstelle

Beim Arbeiten mit den erweiterten Schnittstellen müssen Sie bestimmte Regeln und Richtlinien beachten.

Verwenden Sie die folgenden Regeln und Richtlinien:

- Wenn Sie einen Nullwert als Parameter übergeben oder wenn das Ergebnis eine Ausdrucks Null ist, wird der Wert als Nullindikator behandelt. Wenn das Ergebnis eines Ausdrucks beispielsweise Null ist und der Rückgabedatentyp ist String, wird ein String mit einem Wert von Null zurückgegeben. Sie können das Ergebnis eines Ausdrucks mithilfe der isResultNull-Methode prüfen.
- Sie müssen Eingabeparameter mit einem Date-Datentyp in einen String umwandeln, bevor Sie diese in einem Ausdruck verwenden können. Um den String in einem Ausdruck als Date-Datentyp verwenden zu können, verwenden Sie die to_date()-Funktion, um den String in einen Date-Datentyp umzuwandeln.

  Sie können das Ergebnis eines Ausdrucks abrufen, der einen Date-Datentyp als String- oder long-Datentyp zurückgibt.

  Verwenden Sie die getStringBuffer-Methode, um das Ergebnis eines Ausdrucks abrufen, der einen Date-Datentyp als String-Datentyp zurückgibt. Verwenden Sie die getLong-Methode, um das Ergebnis eines Ausdrucks abrufen, der einen Date-Datentyp als long-Datentyp zurückgibt.

EDataType-Klasse

Zählt die in Ausdrücken verwendeten Java-Datentypen auf. Ruft den Rückgabedatentyp eines Ausdrucks ab oder weist den Datentyp für einen Parameter in einem JExprParamMetadata-Objekt zu. Sie brauchen die EDataType-Klasse nicht instanziieren.

In der folgenden Tabelle sind die Aufzählungswerte für Java-Datentypen in Ausdrücken aufgelistet:

Datentyp	Aufzählungswert
INT	1
DOUBLE	2
STRING	3
BYTE_ARRAY	4
DATE_AS_LONG	5

Der folgende Beispiel-Java-Code zeigt, wie die EDataType-Klasse verwendet wird, um einem JExprParamMetadata-Objekt einen String-Datentypen zuzuweisen:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata (
                EDataType.STRING,  // data type
                20,  // precision
                0  // scale
            );
...
```

JExprParamMetadata-Klasse

Instanziiert ein Objekt, das die Parameter für einen Ausdruck darstellt und die Metadaten für die Parameter festlegt.

Verwenden Sie ein Array von JExprParamMetadata-Objekten als Eingabe in die defineJExpression-Methode, um die Metadaten für die Eingabeparameter festzulegen. Sie können eine Instanz des JExprParamMetadata-Objekts auf der Codeeingabe-Registerkarte **Funktionen** oder in defineJExpression erstellen.

Verwenden Sie folgende Syntax:

```
JExprParamMetadata paramMetadataArray[] = new JExprParamMetadata[numberOfParameters];
paramMetadataArray[0] = new JExprParamMetadata(datatype, precision, scale);
...
paramMetadataArray[numberofParameters - 1] = new JExprParamMetadata(datatype, precision,
scale);;
```

In der folgenden Tabelle werden die Argumente beschrieben:

Argument	Argumenttyp	Argumentdatentyp	Beschreibung
Datentyp	Eingabe	EDataType	Datentyp des Parameters.
Genauigkeit	Eingabe	Ganzzahl	Parametergenauigkeit.
Größenordnung	Eingabe	Ganzzahl	Größenordnung des Parameters.

Verwenden Sie zum Beispiel den folgenden Java-Code, um ein Array von zwei JExprParamMetadata-Objekten mit zwei String-Datentypen, einer Präzision von 20 und einer Größenordnung von 0 zu instanziieren:

```
JExprParamMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
return defineJExpression(":LKP.LKP_addresslookup(X1,X2)",params);
```

defineJExpression

Definiert einen Ausdruck, der unter anderem den Ausdrucksstring und Eingabeparameter enthält. Argumente für die defineJExpression-Methode umfassen ein Array von JExprParamMetadata-Objekten, die die Eingabeparameter und einen Stringwert enthalten, die die Ausdruckssyntax definieren.

Verwenden Sie folgende Syntax:

```
defineJExpression(
    String expression,
    Object[] paramMetadataArray
    );
```

In der folgenden Tabelle werden die Parameter beschrieben:

Parameter	Typ	Datentyp	Beschreibung
expression	Eingabe	String	String, der den Ausdruck darstellt.
paramMetadataArray	Eingabe	Object[]	Array von JExprParaMetadata-Objekten, die den Eingabeparameter für den Ausdruck enthalten.

Sie können die defineJExpression-Methode zum Java-Code auf einer beliebigen Codeeingabe-Registerkarte hinzufügen. Ausnahmen dazu stellen die Registerkarten **Importe** und **Funktionen** dar.

Um die defineJExpression-Methode zu verwenden, müssen Sie ein Array von JExprParamMetadata-Objekten instanziieren, die die Eingabeparameter für den Ausdruck darstellen. Sie legen die Metadatenwerte für die Parameter fest und leiten das Array als Parameter an die defineJExpression-Methode weiter.

Der folgende Java-Code erstellt beispielsweise einen Ausdruck, um den Wert von zwei Strings nachzuschlagen:

```
JExprParaMetadata params[] = new JExprParamMetadata[2];
params[0] = new JExprParamMetadata(EDataType.STRING, 20, 0);
params[1] = new JExprParamMetadata(EDataType.STRING, 20, 0);
defineJExpression(":lkp.mylookup(x1,x2)",params);
```

Hinweis: Sie müssen die Parameter nummerieren, die Sie nacheinander an den Ausdruck übergeben, und den Parameter mit dem Buchstaben x beginnen. Um beispielsweise drei Parameter an den Ausdruck zu übergeben, benennen Sie die Parameter mit x1, x2 und x3.

JExpression-Klasse

Enthält die Methoden zum Erstellen und Aufrufen eines Ausdrucks, gibt den Wert eines Ausdrucks zurück und prüft den Rückgabedatentyp.

In der folgenden Tabelle sind die Methoden in der JExpression-Klasse aufgelistet:

Methodenname	Beschreibung
invoke	Ruft einen Ausdruck auf.
getResultDataType	Gibt den Datentyp des Ausdrucksergebnisses zurück.
getResultMetadata	Gibt die Metadaten des Ausdrucksergebnisses zurück.
isResultNull	Prüft den Ergebniswert eines Ausdrucksergebnisses.
getInt	Gibt den Wert eines Ausdrucksergebnisses als einen Integer-Datentyp zurück.
getDouble	Gibt den Wert eines Ausdrucksergebnisses als Double-Datentyp zurück.
getStringBuffer	Gibt den Wert eines Ausdrucksergebnisses als String-Datentyp zurück.
getBytes	Gibt den Wert eines Ausdrucksergebnisses als byte[]-Datentyp zurück.

Erweiterte Schnittstelle - Beispiel

Mit der erweiterten Schnittstelle können Sie eine Lookup-Expression in einer Java-Umwandlung erstellen und aufrufen.

Der folgende Beispiel-Java-Code zeigt, wie Sie eine Funktion erstellen, die eine Expression aufruft, und wie Sie die Expression zum Abrufen des Rückgabewerts aufrufen. Dieses Beispiel übergibt die Werte für zwei Eingabeports mit einem String-Datentyp, NAME und COMPANY an die Funktion myLookup. Die myLookup-Funktion verwendet einen Lookup-Ausdruck, um den Wert für den ADDRESS-Ausgabeport zu suchen.

Hinweis: Dieses Beispiel setzt voraus, dass Sie über eine nicht verbundene Lookup-Umwandlung im Mapping namens LKP_addresslookup verfügen.

Verwenden Sie den folgenden Java-Code auf der Registerkarte **Helfer**Registerkarte:

```
JExpression addressLookup()  throws SDKException
{
     JExprParamMetadata params[] = new JExprParamMetadata[2];
```

```
        params[0] = new JExprParamMetadata (
                        EDataType.STRING,         // data type
                        50,                   // precision
                        0                     // scale
                        );
        params[1] = new JExprParamMetadata (
                        EDataType.STRING,     // data type
                        50,                   // precision
                        0                     // scale
                        );
        return defineJExpression(":LKP.LKP_addresslookup(X1,X2)",params);
    }
    JExpression lookup = null;
    boolean isJExprObjCreated = false;
```

Verwenden Sie den folgenden Java-Code auf der Registerkarte **Bei Eingabe**, um den Ausdruck aufzurufen und den Wert des ADDRESS-Ports zurückzugeben:

```
...
if(!iisJExprObjCreated)
{
    lookup = addressLookup();
    isJExprObjCreated = true;
}
lookup = addressLookup();
lookup.invoke(new Object [] {NAME,COMPANY}, ERowType.INSERT);
EDataType addressDataType = lookup.getResultDataType();
if(addressDataType == EDataType.STRING)
{
    ADDRESS = (lookup.getStringBuffer()).toString();
} else {
    logError("Expression result datatype is incorrect.");
}
...
```

API-Referenz für JExpression-Klasse

Die JExpression-Klasse enthält API-Methoden, mit denen Sie einen Ausdruck erstellen und aufrufen, den Wert eines Ausdrucks zurückgeben und den Rückgabedatentyp überprüfen können.

Die JExpression-Klasse enthält die folgenden API-Methoden:

- getBytes
- getDouble
- getInt
- getLong
- getResultDataType
- getResultMetadata
- getStringBuffer
- invoke
- isResultNull

getBytes

Gibt den Wert eines Ausdrucksergebnisses als byte[]-Datentyp zurück. Ruft das Ergebnis eines Ausdrucks ab, der Daten mit der AES_ENCRYPT-Funktion verschlüsselt.

Verwenden Sie folgende Syntax:

```
objectName.getBytes();
```

Verwenden Sie den folgenden Java-Code, um das Ergebnis eines Ausdrucks abzurufen, das die binären Daten mit der AES_ENCRYPT-Funktion verschlüsselt, wobei AES_ENCRYPT ein JExpression-Objekt ist:

```
byte[] newBytes = JExprEncryptData.getBytes();
```

getDouble

Gibt den Wert eines Ausdrucksergebnisses als Double-Datentyp zurück.

Verwenden Sie folgende Syntax:

```
objectName.getDouble();
```

Verwenden Sie das folgende Java-Codebeispiel, um das Ergebnis eines Ausdrucks abzurufen, der einen Gehaltswert als ein Double zurückgibt, wobei JExprSalary das JExpression-Objekt ist:

```
double salary = JExprSalary.getDouble();
```

getInt

Gibt den Wert eines Ausdrucksergebnisses als einen Integer-Datentyp zurück.

Verwenden Sie folgende Syntax:

```
objectName.getInt();
```

Verwenden Sie beispielsweise den folgenden Java-Code, um das Ergebnis eines Ausdrucks abzurufen, der eine Mitarbeiter-ID-Nummer als Ganzzahl zurückgibt, wobei findEmpID ein JExpression-Objekt ist.

```
int empID = findEmpID.getInt();
```

getLong

Gibt den Wert eines Ausdrucksergebnisses als Long-Datentyp zurück. Ruft das Ergebnis eines Ausdrucks ab, der einen Date-Datentyp verwendet.

Verwenden Sie folgende Syntax:

```
objectName.getLong();
```

Verwenden Sie den folgenden Java-Beispielcode, um das Ergebnis eines Ausdrucks abzurufen, der einen Datumswert als einen Long-Datentyp zurückgibt, wobei JExprCurrentDate ein JExpression-Objekt ist.

```
long currDate = JExprCurrentDate.getLong();
```

getResultDataType

Gibt den Datentyp eines Ausdrucksergebnisses zurück. Gibt einen Wert von EDataType zurück.

Verwenden Sie folgende Syntax:

```
objectName.getResultDataType();
```

Verwenden Sie den folgenden Java-Code, um einen Ausdruck aufzurufen und den Datentyp des Ergebnisses zur dataType-Variable zuzuweisen:

```
myObject.invoke(new Object[] { NAME,COMPANY }, ERowType INSERT);
EDataType dataType = myObject.getResultDataType();
```

getResultMetadata

Gibt die Metadaten für ein Ausdrucksergebnis zurück. Sie können getResultMetadata verwenden, um die Präzision, Größenordnung und den Datentyp eines Ausdrucksergebnisses abzurufen. Sie können die Metadaten des Rückgabewerts aus einem Ausdruck zu einem JExprParamMetadata-Objekt zuweisen. Verwenden Sie die Objektmethoden getScale, getPrecision und getDataType , um die Ergebnismetadaten abzurufen.

Verwenden Sie folgende Syntax:

```
objectName.getResultMetadata();
```

Verwenden Sie den folgenden Java-Beispielcode, um die Größenordnung, Präzision und den Datentyp des Rückgabewerts von myObject zu Variablen zuzuweisen:

```
JExprParamMetadata myMetadata = myObject.getResultMetadata();
int scale = myMetadata.getScale();
int prec = myMetadata.getPrecision();
int datatype = myMetadata.getDataType();
```

Hinweis: Die getDataType-Objektmethode gibt den Ganzzahlwert des Datentyps wie in EDataType aufgelistet zurück.

getStringBuffer

Gibt den Wert eines Ausdrucksergebnisses als String-Datentyp zurück.

Verwenden Sie folgende Syntax:

```
objectName.getStringBuffer();
```

Verwenden Sie den folgenden Java-Beispielcode, um das Ergebnis eines Ausdrucks abzurufen, das zwei verkettete Strings zurückgibt, wobei JExprConcat ein JExpression-Objekt ist:

```
String result = JExprConcat.getStringBuffer();
```

invoke

Ruft einen Ausdruck auf. Argumente für invoke beinhalten ein Objekt, das die Eingabeparameter und den Zeilentyp definiert. Sie müssen ein JExpression-Objekt instanziieren, bevor Sie die invoke-Methode verwenden können. Verwenden Sie ERowType.INSERT, ERowType.DELETE und ERowType.UPDATE für den Zeilentyp.

Verwenden Sie folgende Syntax:

```
objectName.invoke(
    new Object[] { param1[, ... paramN ]},
    rowType
    );
```

In der folgenden Tabelle werden die Argumente beschrieben:

Argument	Datentyp	Eingabe/ Ausgabe	Beschreibung
objectName	JExpression	Eingabe	JExpression-Objektname.
Parameter	-	Eingabe	Objekt-Array, das die Eingabewerte für den Ausdruck enthält.

Sie erstellen beispielsweise eine Funktion auf der Codeeingabe-Registerkarte **Funktionen** namens address_lookup(), die ein JExpression-Objekt zurückgibt, das den Ausdruck darstellt. Verwenden Sie den folgenden Code, um den Ausdruck aufzurufen, der die Eingabeports NAME und COMPANY verwendet:

```
JExpression myObject = address_lookup();
myObject.invoke(new Object[] { NAME,COMPANY }, ERowType INSERT);
```

isResultNull

Prüft den Wert eines Ausdrucksergebnisses.

Verwenden Sie folgende Syntax:

```
objectName.isResultNull();
```

Verwenden Sie das folgende Java-Codebeispiel, um einen Ausdruck aufzurufen und den Rückgabewert des Ausdrucks zu einer Variablenadresse zuzuweisen, wenn der Rückgabewert nicht Null ist:

```
JExpression myObject = address_lookup();
myObject.invoke(new Object[] { NAME,COMPANY }, ERowType INSERT);
if(!myObject.isResultNull()) {
     String address = myObject.getStringBuffer();
}
```

KAPITEL 20

Joiner-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Joiner-Umwandlung - Übersicht, 311
- Joiner-Umwandlung – Erweiterte Eigenschaften, 312
- Joiner-Caches, 313
- Joiner-Umwandlungsports, 314
- Joiner-Umwandlungen in dynamischen Mappings, 315
- Portauswahlen in einer Joiner-Umwandlung, 315
- Definieren einer Join-Bedingung, 318
- Join-Typen, 322
- Sortierte Eingabe für eine Joiner-Umwandlung, 325
- Zusammenfügen von Daten aus derselben Quelle, 328
- Blockieren der Quellen-Pipelines, 330
- Leistungstipps für die Joiner-Umwandlung, 331
- Regeln und Richtlinien für eine Joiner-Umwandlung, 332

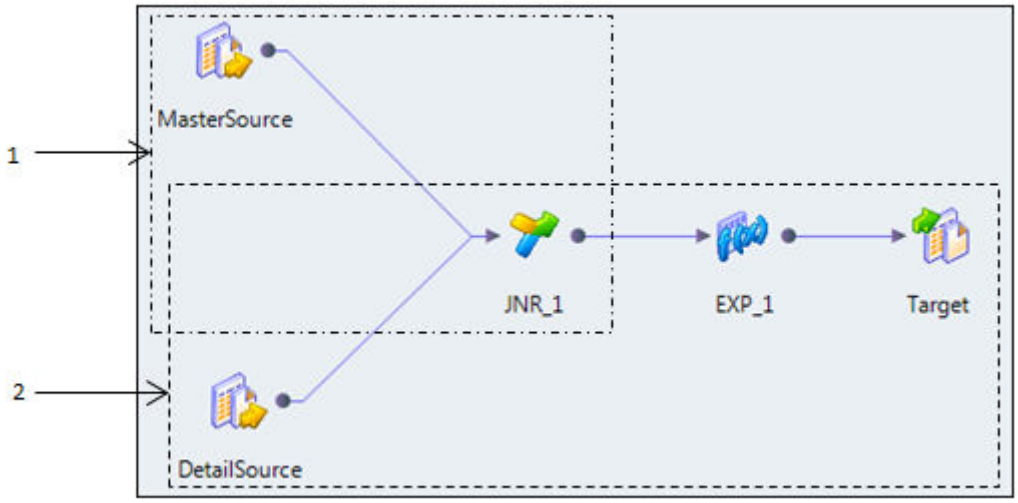
Joiner-Umwandlung - Übersicht

Die Joiner-Umwandlung verbindet Quelldaten aus zwei verwandten heterogenen Quellen aus verschiedenen Speicherorten oder aus verschiedenen Dateisystemen. Sie können auch Daten aus derselben Quelle zusammenfügen. Bei der Joiner-Umwandlung handelt es sich um eine aus mehreren Gruppen bestehende aktive Umwandlung.

Bei der Joiner-Umwandlung werden Quellen mit mindestens einer übereinstimmenden Spalte zusammengefügt. Die Joiner-Umwandlung verwendet eine Bedingung, die ein oder mehrere Spaltenpaare aus den beiden Quellen einander zuordnet.

Die beiden Eingabe-Pipelines enthalten eine Master-Pipeline und eine Detail-Pipeline oder einen Master- und einen Detail-Zweig. Der Master-Pipeline endet bei der Joiner-Umwandlung, während die Detail-Pipeline bis zum Ziel weiterführt.

In der folgenden Abbildung werden die Master- und Detail-Pipelines in einem Mapping mit einer Joiner-Umwandlung angezeigt:



1. Master-Pipeline
2. Detail-Pipeline

Um mehr als zwei Quellen in einem Mapping zusammenzufügen, verknüpfen Sie die Ausgabe aus der Joiner-Umwandlung mit einer anderen Quellen-Pipeline. Fügen Sie Joiner-Umwandlungen zum Mapping hinzu, bis alle Quellen-Pipelines miteinander verknüpft sind.

Joiner-Umwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, um zu bestimmen, wie der Data Integration Service Daten für die Joiner-Umwandlung verarbeitet.

Konfigurieren Sie die folgenden Eigenschaften auf der Registerkarte **Erweitert**:

Joiner-Daten-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Daten-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto" auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".

Joiner-Index-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Index-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto" auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".

Cache-Verzeichnis

Verzeichnis, in dem der Datenintegrationsdienst die Index-Cache- und Daten-Cachedateien erstellt. Stellen Sie sicher, dass das Verzeichnis vorhanden ist und genügend Speicherplatz für die Cachedateien enthält.

Geben Sie mehrere Verzeichnisse durch Semikolon getrennt ein, um die Leistung während der Cache-Partitionierung zu erhöhen. Die Cache-Partitionierung erstellt für jede Partition, die die Umwandlung verarbeitet, einen separaten Cache.

Standardwert ist der CacheDir-Systemparameter. Sie können für diese Eigenschaft einen anderen Systemparameter oder einen benutzerdefinierten Parameter konfigurieren.

Sortierte Eingabe

Gibt an, dass die Eingabedaten nach Gruppen vorsortiert sind. Wählen Sie "Sortierte Eingabe" aus, um sortierte Daten zusammenzufügen. Mithilfe der sortierten Eingabe kann die Leistung verbessert werden.

Master-Sortierreihenfolge

Gibt die Sortierreihenfolge der Master-Quelldaten an. Wählen Sie "Aufsteigend" aus, wenn die Master-Quelldaten aufsteigend sortiert sind. Wenn Sie "Aufsteigend" auswählen, aktivieren Sie auch "Sortierte Eingabe". Der Standardwert ist „Auto".

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz", „Normal", „Ausführliche Intialisierung" oder „Ausführliche Daten" wählen. Standardwert ist „Normal".

Verwandte Themen:

- "Cachegröße" auf Seite 662

Joiner-Caches

Wenn Sie ein Mapping ausführen, das eine Joiner-Umwandlung verwendet, erstellt der Datenintegrationsdienst einen Index- und einen Daten-Cache im Speicher, um die Umwandlung auszuführen. Wenn der Datenintegrationsdienst mehr Speicherplatz benötigt als im Speicher-Cache vorhanden ist, werden Überlaufdaten in Cachedateien gespeichert.

Wenn Sie ein Mapping ausführen, das eine Joiner-Umwandlung verwendet, liest der Datenintegrationsdienst die Zeilen aus den Master- und Detailquellen gleichzeitig und erstellt die Index- und Daten-Caches basierend auf den Masterzeilen. Der Datenintegrationsdienst führt den Join basierend auf den Detail-Quelldaten und den zwischengespeicherten Masterdaten aus.

Der Typ der Joiner-Umwandlung bestimmt die Anzahl der Zeilen, die der Datenintegrationsdienst im Cache speichert.

In der folgenden Tabelle werden die Informationen beschrieben, die der Datenintegrationsdienst in den Caches für die verschiedenen Typen von Joiner-Umwandlungen speichert:

Typ der Joiner-Umwandlung	Index-Cache	Daten-Cache
Unsortierte Eingabe	Speichert alle Masterzeilen in der Join-Bedingung mit eindeutigen Indexschlüsseln.	Speichert alle Masterzeilen.
Sortierte Eingabe mit verschiedenen Quellen	Speichert 100 Masterzeilen in der Join-Bedingung mit eindeutigen Indexschlüsseln.	Speichert Masterzeilen, die den im Index-Cache gespeicherten Zeilen entsprechen. Wenn die Masterdaten mehrere Zeilen mit dem gleichen Schlüssel enthalten, speichert der Datenintegrationsdienst mehr als 100 Zeilen im Daten-Cache.
Sortierte Eingabe mit derselben Quelle	Speichert alle Master- oder Detailzeilen in der Join-Bedingung mit eindeutigen Schlüsseln. Speichert Detailzeilen, wenn der Datenintegrationsdienst die Detail-Pipeline schneller als die Master-Pipeline verarbeitet. Andernfalls werden Masterzeilen gespeichert. Die Anzahl der gespeicherten Zeilen ist abhängig von der Verarbeitungsgeschwindigkeit der Master- und Detail-Pipelines. Wenn eine Pipeline die jeweiligen Reihen schneller als die andere verarbeitet, speichert der Datenintegrationsdienst alle bereits verarbeiteten Zeilen im Cache. Der Dienst lässt die Zeilen so lange im Cache, bis die andere Pipeline die Verarbeitung ihrer Zeilen abgeschlossen hat.	Speichert Daten für die im Index-Cache gespeicherten Zeilen. Wenn der Index-Cache Schlüssel für die Master-Pipeline speichert, speichert der Daten-Cache die Daten für die Master-Pipeline. Wenn der Index-Cache Schlüssel für die Detail-Pipeline speichert, speichert der Daten-Cache die Daten für die Detail-Pipeline.

Joiner-Umwandlungsports

Eine Joiner-Umwandlung hat verschiedene Porttypen, die bestimmen, wie der Datenintegrationsdienst das Join durchführt.

Eine Joiner-Umwandlung weist die folgenden Porttypen auf:

Master

Ports, die sich mit der Master-Quelle im Mapping verknüpfen.

Detail

Ports, die sich mit der Detail-Quelle im Mapping verknüpfen.

Dynamischer Port

Empfängt oder gibt Ports in einem dynamischen Mapping zurück. Ein dynamischer Port kann eine oder mehrere Spalten aus einer vorgelagerten Umwandlung empfangen und für jede Spalte einen generierten Port erstellen. Ein dynamischer Ausgabeport kann einen oder mehrere generierte Ports zurückgeben. Sie können Eingaberegeln definieren, um festzulegen, welche Spalten ein dynamischer Port empfängt.

Sie können einen Port aus einem Master-Port in einen Detail-Port ändern. Sie können einen Port auch aus einem Detail-Port in einen Master-Port ändern. Wenn Sie den Porttyp eines Ports ändern, wirkt sich diese Änderung auf alle Ports aus. Wenn Sie also einen Master-Port in einen Detail-Port ändern, ändern Sie alle Master-Ports in Detail-Ports und alle Detail-Ports in Master-Ports.

Joiner-Umwandlungen in dynamischen Mappings

Sie können eine Joiner-Umwandlung in einem dynamischen Mapping verwenden. Sie können auf dynamische Ports und generierte Ports in Join-Bedingungen verweisen.

Ein dynamisches Mapping ist ein Mapping, in dem sich Quellen, Ziele und Umwandlungslogik zur Laufzeit ändern können. Sie können Parameter und Regeln zum Ändern der Datenstruktur festlegen. Wenn Sie eine Joiner-Umwandlung in einem dynamischen Mapping verwenden, kann sich die Struktur der Quelle verändern. Die Eingabeports und die Ports in der Join-Bedingung ändern sich auch.

Sie können die folgenden Aufgaben durchführen, um eine Joiner-Umwandlung in einem dynamischen Mapping zu konfigurieren:

Dynamische Ports definieren.

Definieren Sie dynamische Ports und generierte Ports, um die unterschiedlichen Eingabespalten aus einer dynamischen Quelle anzupassen. Sie können dynamische Ports bzw. generierte Ports in die Join-Bedingungen einbeziehen.

Portauswahlen definieren.

Definieren Sie eine Portauswahl, die die in der Join-Bedingung zu verwendenden Ports enthält. Konfigurieren Sie Auswahlregeln, die die in die Portauswahl einzuschließenden Ports festlegt. Sie können die Portauswahl parametrisieren, um bestimmte Ports zur Laufzeit einzuschließen.

Die Join-Bedingung parametrisieren.

Sie können die gesamte Join-Bedingung parametrisieren. Konfigurieren Sie einen Ausdrucksparameter für jede möglicherweise benötigte Join-Bedingung.

Weitere Informationen zu dynamischen Mappings finden Sie im *Informatica Developer-Handbuch für Mappings*.

Portauswahlen in einer Joiner-Umwandlung

Wenn die Joiner-Umwandlung über generierte Ports verfügt, müssen Sie eine Join-Bedingung konfigurieren, die dann gültig ist, wenn die Umwandlung über unterschiedliche generierte Ports zur Laufzeit verfügt.

Beispiel: Ein dynamisches Mapping enthält eine Joiner-Umwandlung mit der folgenden Join-Bedingung:

```
CustomerID = CustomerNo
```

CustomerID ist ein generierter Port in der Joiner-Umwandlung. Da das Mapping eine dynamische Quelle aufweist, kann das Mapping in mehreren unterschiedlichen Quelldateiformaten ausgeführt werden. Die Spalte mit der Kundennummer weist in jeder Quelldatei einen anderen Namen auf: `CustomerID`,`CustomerNum` oder `CustNO`.

Sie können eine Portauswahl in der Joiner-Umwandlung erstellen, um die unterschiedlichen Namen der Kundenspalten aus der dynamischen Quelle anzupassen. Konfigurieren Sie eine Portauswahl mit einer Auswahlregel, die einen beliebigen Portnamen mit dem Präfix „Cust“ enthält.

Konfigurieren Sie anschließend die Join-Bedingung so, dass der Name der Portauswahl anstelle des Spaltennamens „CustomerID“ eingefügt wird:

```
Customer_PortSelector = CustomerNo
```

Die Join-Bedingung ist mit jedem Portnamen gültig, der mit „Cust“ beginnt.

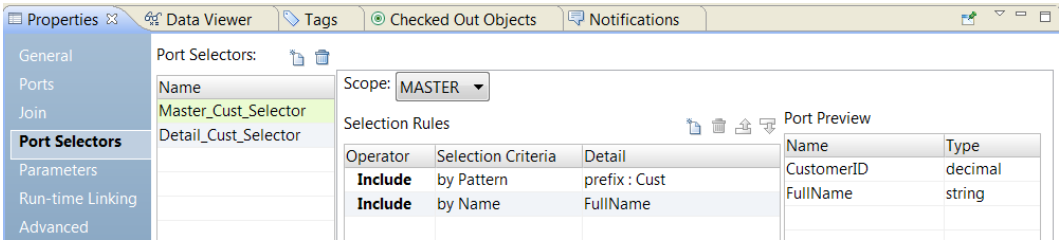
Eine Portauswahl kann einen oder mehrere Ports enthalten. Die Join-Bedingung kann mehrere Ports enthalten, wenn die Master-Gruppe und die Detail-Gruppe in der Join-Bedingung dieselbe Anzahl von Ports enthalten.

Auswahlregeln

Wenn Sie eine Portauswahl konfigurieren, definieren Sie Auswahlregeln, um zu bestimmen, welche generierten Ports einbezogen werden sollen. Die Auswahlregeln ähneln den Eingaberegeln, die Sie für dynamische Ports konfigurieren können.

Eine Portauswahl kann Ports oder generierte Ports enthalten. Konfigurieren Sie eine Portauswahl auf der Registerkarte **Portauswahl**.

Die folgende Abbildung zeigt die Registerkarte **Portauswahl**:



Konfigurieren Sie die folgenden Eigenschaften für eine Portauswahl:

Name

Gibt die Portauswahl an. Sie können mehrere Portauswahlen in einer Umwandlung erstellen und in Ausdrücken auf sie verweisen.

Bereich

Gibt eine Gruppe von Ports an, auf die die Portauswahl angewendet wird. Wählen Sie einen Master- oder einen Detailbereich.

Auswahlregeln

Bestimmt die Ports, die in die Portauswahl einbezogen werden sollen. Beim Erstellen der Auswahlregeln werden im Bereich **Portvorschau** die Ports angezeigt, die sich aus den aktuellen Eingabeports qualifizieren. Diese Ports ändern sich möglicherweise. Konfigurieren Sie die Auswahlregeln, um Ports aus unterschiedlichen Quellen anzupassen.

Sie können Auswahlregeln basierend auf den folgenden Kriterien erstellen:

Operator

Schließt die Ports ein bzw. aus, die Auswahlregeln zurückgeben. Standardwert ist einschließen. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.

Auswahlkriterien

Der Auswahlregeltyp, den Sie erstellen möchten. Sie können eine Regel basierend auf dem Porttyp oder dem Spaltennamen erstellen. Um Ports basierend auf dem Spaltennamen einzuschließen, suchen Sie nach bestimmten Namen oder nach einem Muster von Zeichen im Namen.

Detail

Die auf die Auswahlkriterien anzuwendenden Werte. Wenn die Auswahlkriterien nach Spaltenname erfolgen, konfigurieren Sie die Zeichenfolge oder den Namen, nach der bzw. dem gesucht werden soll. Wenn die Auswahlkriterien nach Porttyp erfolgen, wählen Sie die einzuschließenden Porttypen aus.

In der folgenden Tabelle werden die Auswahlkriterien beschrieben und wie die Details für die Kriterien angegeben werden müssen:

Auswahlkriterien	Beschreibung	Detail
Alle	Schließt alle Ports ein.	Keine Details erforderlich.
Name	Filtert Ports basierend auf dem Portnamen.	Wählen Sie die Portnamen aus einer Liste mit Werten aus oder verwenden Sie einen Parameter des Typs „Port“ oder „Portliste“.
Typ	Filtert Ports basierend auf dem Datentyp jedes Ports.	Wählen Sie Datentypen aus einer Liste aus.
Muster	Filtert Ports nach einer Zeichenfolge von Zeichen im Namen oder nach einem regulären Ausdruck.	Wählen Sie ein Präfix, ein Suffix oder einen regulären Ausdruck als Mustertyp für den Portnamen aus. Geben Sie anschließend einen Wert für das Muster ein oder verwenden Sie einen Parameter des Typs „Zeichenfolge“.

Erstellen einer Portauswahl

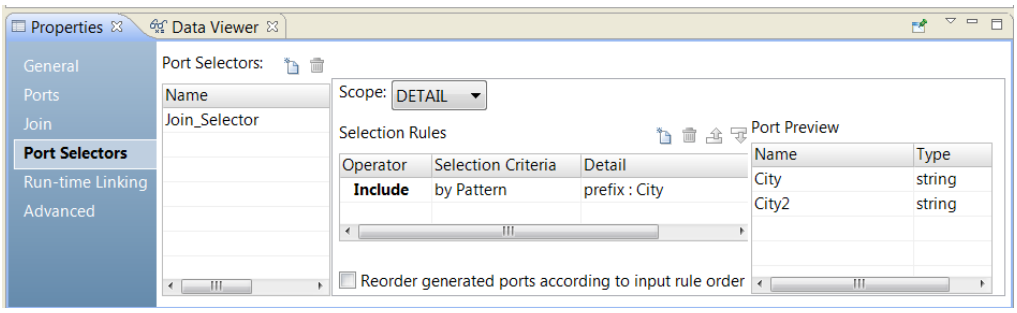
Erstellen Sie einen Port zum Festlegen, welche Ports in einem dynamischen Ausdruck, einer Lookup-Bedingung oder einer Joiner-Bedingung verwendet werden sollen.

1. Klicken Sie auf die Registerkarte **Portauswahlen**.
2. Klicken Sie im Bereich **Portauswahlen** auf **Neu**.

   Das Developer Tool erstellt eine Portauswahl mit einer Standardauswahlregel, die alle Ports enthält.
3. Ändern Sie im Bereich **Portauswahlen** den Namen der Portauswahl in einen eindeutigen Namen.
4. Wenn Sie die Joiner-Umwandlung oder die Lookup-Umwandlung bearbeiten, wählen Sie den Bereich aus.

   Die verfügbaren Ports ändern sich basierend auf der Gruppe der von Ihnen ausgewählten Ports.
5. Wählen Sie im Bereich **Auswahlregeln** einen **Operator** aus.
   - Einschließen. Erstellen Sie eine Regel, die Ports für die Portauswahl einschließt. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.
   - Ausschließen. Erstellen Sie eine Regel, die bestimmte Ports aus der Portauswahl ausschließt.
6. Wählen Sie die **Auswahlkriterien** aus.
   - Nach Name. Wählen Sie bestimmte Ports nach Namen aus. Sie können die Portnamen aus einer Liste mit Ports im Bereich auswählen.
   - Nach Typ. Wählen Sie Ports nach Typ aus. Sie können eine oder mehrere Datentypen auswählen.
   - Nach Muster. Wählen Sie Ports nach einem Muster von Zeichen im Portnamen aus. Sie können die Suche mit bestimmten Zeichen durchführen oder einen regulären Ausdruck erstellen.

   Die folgende Abbildung zeigt die Registerkarte „Portauswahl“:



7. Klicken Sie auf die Spalte **Detail**.

   Das Dialogfeld **Eingaberegel-Detail** wird angezeigt.

8. Wählen Sie die Werte aus, nach denen Ports gefiltert werden sollen.

   - Nach Name. Wählen Sie, ob eine Portliste nach Wert oder nach einem Parameter erstellt werden soll. Klicken Sie auf **Auswählen**, um die Ports auf der Liste auszuwählen.
   - Nach Typ. Wählen Sie mindestens einen Datentyp aus einer Liste aus. Im Bereich **Portvorschau** werden Ports des von Ihnen ausgewählten Typs angezeigt.
   - Nach Muster. Wählen Sie, ob das Präfix oder das Suffix des Portnamens nach einem bestimmten Muster von Zeichen durchsucht werden soll. Oder entscheiden Sie sich für das Erstellen eines regulären Ausdrucks für die Suche. Konfigurieren Sie einen Parameter oder konfigurieren Sie das Muster für die Suche.

   Im Bereich **Portvorschau** werden während der Konfiguration der Regeln die Ports in der Portauswahl angezeigt.

9. Um die Ports in der Portauswahl neu anzuordnen, wählen Sie **Erzeugte Ports gemäß der Reihenfolge der Eingaberegeln neu sortieren** aus.

Definieren einer Join-Bedingung

Die Join-Bedingung enthält Ports aus beiden Eingabequellen, die vom Datenintegrationsdienst zum Zusammenfügen zweier Zeilen verwendet werden.

Je nach ausgewähltem Join-Typ fügt der Datenintegrationsdienst die Zeile entweder zum Ergebnissatz hinzu oder verwirft die Zeile. Die Joiner-Umwandlung erzielt Ergebnissätze auf Basis des Join-Typs, der Bedingung und der Eingabedatenquellen.

Stellen Sie vor dem Definieren einer Join-Bedingung sicher, dass die Master- und Detail-Quellen für eine optimale Leistung konfiguriert sind. Während eines Mappings vergleicht der Datenintegrationsdienst jede Zeile der Master-Quelle mit der Detail-Quelle. Verwenden Sie zur Verbesserung der Leistung für eine unsortierte Joiner-Umwandlung die Quelle mit weniger Zeilen als die Master-Quelle. Verwenden Sie zur Verbesserung der Leistung für eine sortierte Joiner-Umwandlung die Quelle mit weniger doppelten Schlüsselwerten als die Master-Quelle.

Verwenden Sie mindestens einen Port aus den Eingabequellen einer Joiner-Umwandlung in der Join-Bedingung. Weitere Ports verlängern die Zeit, die zum Zusammenführen zweier Quellen benötigt wird. Die Reihenfolge der Ports in der Bedingung kann die Leistung der Joiner-Umwandlung beeinflussen. Wenn Sie mehrere Ports in der Join-Bedingung verwenden, vergleicht der Datenintegrationsdienst die Ports in der angegebenen Reihenfolge.

Wenn Sie Char- und Varchar-Datentypen zusammenfügen, zählt der Datenintegrationsdienst alle Leerzeichen, die Char-Werte als Teil der Zeichenfolge auffüllen:

```
Char(40) = "abcd"
Varchar(40) = "abcd"
```

Der Char-Wert lautet "abcd" aufgefüllt mit 36 Leerzeichen. Der Datenintegrationsdienst fügt die beiden Felder nicht zusammen, da das Char-Feld nachgestellte Leerzeichen enthält.

Hinweis: Die Joiner-Umwandlung stimmt nicht mit Nullwerten überein. Wenn beispielsweise sowohl EMP_ID1 als auch EMP_ID2 eine Zeile mit einem Nullwert enthalten, betrachtet der Datenintegrationsdienst sie nicht als Übereinstimmung und fügt die beiden Zeilen nicht zusammen. Um Zeilen mit Nullwerten zusammenzuführen, ersetzen Sie Null-Eingaben durch Standardwerte und fügen Sie dann die Standardwerte zusammen.

Sie können einen einfachen oder erweiterten Bedingungstyp definieren. Zudem können Sie einen Ausdrucksparameter definieren. Ein Ausdrucksparameter ist ein Parameter, der den Join-Ausdruck enthält. Sie können den Parameterwert zur Laufzeit mit einem Mapping-Parameter ändern.

Einfacher Bedingungstyp

Definieren Sie einen einfachen Bedingungstyp für eine sortierte oder unsortierte Joiner-Umwandlung.

Eine einfache Bedingung umfasst eine oder mehrere Bedingungen, die die angegebenen Master- und Detail-Quellen vergleicht. Eine einfache Bedingung muss folgendes Format verwenden:

```
<master_port> operator <detail_port>
```

Für eine sortierte Joiner-Umwandlung muss die Bedingung den Gleichheitsoperator verwenden.

Für eine unsortierte Joiner-Umwandlung kann die Bedingung einen der folgenden Operatoren verwenden: =, ! =, >, >=, <, <=.

Wenn beispielsweise zwei Quellen mit Tabellen mit der Bezeichnung MITARBEITER_ALTER und MITARBEITER_POSITION Mitarbeiter-IDs enthalten, gleicht die folgende Bedingung Zeilen mit Mitarbeitern ab, die in beiden Quellen aufgelistet werden:

```
EMP_ID1 = EMP_ID2
```

Das Developer Tool validiert die Datentypen in einer einfachen Bedingung. Beide Ports in der Bedingung müssen den gleichen Datentyp aufweisen. Wenn Sie in der Bedingung zwei Ports mit nicht übereinstimmenden Datentypen verwenden müssen, wandeln Sie die Datentypen so um, dass sie übereinstimmen.

Sie können eine Liste mit Join-Bedingungen in einer einfachen Bedingung konfigurieren. Wenn Sie mehrere Join-Bedingungen konfigurieren, müssen alle Bedingungen „True“ lauten, damit der Join durchgeführt werden kann.

Beispielsweise können Sie die folgenden Anweisungen in einer einfachen Bedingung konfigurieren:

```
StoreID = StoreNO
Dept    = Department
Salary  > Commission
```

Wenn Sie dieselben Anweisungen als eine erweiterte Bedingung anzeigen, wird die Join-Bedingung als folgender Ausdruck angezeigt:

```
StoreID = StoreNO AND Dept = Department AND (Salary > Commission)
```

Erweiterter Bedingungstyp

Definieren Sie einen erweiterten Bedingungstyp für eine unsortierte Joiner-Umwandlung.

Eine erweiterte Bedingung kann einen Ausdruck enthalten, der einen booleschen oder numerischen Wert liefert. Eine erweiterte Bedingung kann jeden der folgenden Operatoren enthalten: =, !=, >, >=, <, <=.

Sie können eine Konstante für die Join-Bedingung eingeben. Das numerische Äquivalent von FALSE ist Null (0). Alle Nicht-Nullwerte sind das Äquivalent zu TRUE. Die Umwandlung enthält zum Beispiel einen Port namens NUMBER_OF_UNITS mit einem numerischen Datentyp. Konfigurieren Sie eine Filterbedingung, um FALSE zurückzugeben, wenn der Wert NUMBER_OF_UNITS gleich Null ist. Andernfalls gibt die Bedingung TRUE zurück.

Hinweis: Ein einzelner dynamischer Port oder eine Portauswahl kann nicht als boolescher Wert für eine Join-Bedingung verwendet werden.

Wählen Sie zur Eingabe eines Ausdrucks in der Join-Bedingung auf der Registerkarte **Join** den Bedingungstyp "Erweitert" aus. Verwenden Sie den Ausdruckseditor, um Ports, Parameter, Ausdrücke, Portauswahlen und Operatoren in die Bedingung aufzunehmen. Generierte Ports lassen sich jedoch verwenden. Sie können einen einzelnen Port im Ausdruckseditor eingeben, sofern der Porttyp numerisch ist. Es kann jedoch keine Portauswahl als ein Ausdruck eingeben werden.

Sie möchten beispielsweise unter Verwendung des vollständigen Namens eines Mitarbeiters Quellen zusammenfügen. Die Master-Quelle enthält einen FirstName- und einen LastName-Port. Die Detail-Quelle enthält einen FullName-Port. Definieren Sie die folgende Bedingung, um die Master-Ports zu verketten und den vollständigen Namen in beiden Quellen zu finden:

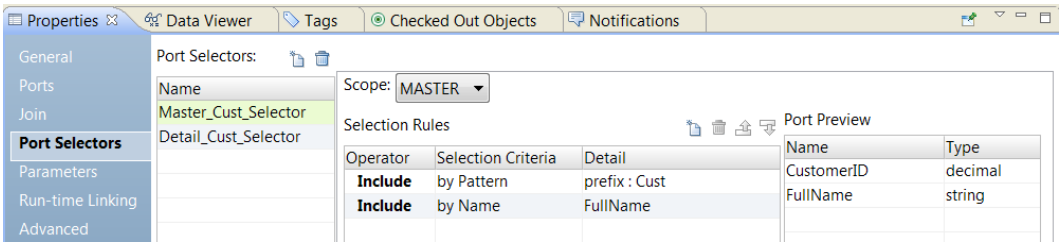
```
CONCAT(FirstName, LastName) = FullName
```

Portauswahlen in Join-Bedingungen

Sie können Portauswahlen in eine Join-Bedingung einschließen. Die Join-Bedingung muss auf eine Portauswahl aus der Master-Gruppe und auf eine Portauswahl aus der Detail-Gruppe verweisen.

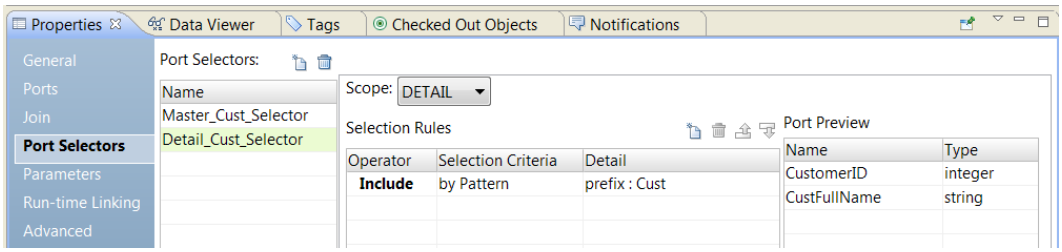
Beispiel: Die Joiner-Umwandlung verfügt über dynamische Ports. Möglicherweise müssen Sie mehrere generierte Ports in der Join-Bedingung vergleichen.

Die folgende Abbildung zeigt die Felder in der Portauswahl für die Master-Gruppe:



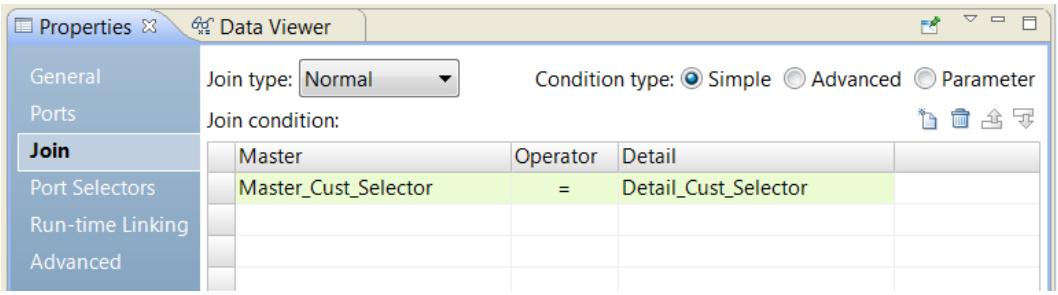
Die Portauswahl Master_Cust_Selector enthält die Ports „CustomerID“ und „FullName“.

Die folgende Abbildung zeigt die Felder in der Portauswahl für die Detail-Gruppe:



Die Portauswahl Detail_Cust_Selector enthält die Ports „CustomerNo“ und „CustFullName“. Diese Ports weisen das Präfix `Cust` auf.

Erstellen Sie die folgende einfache Join-Bedingung:



Die Join-Bedingung vergleicht jeden Port in der Portauswahl Master_Cust_Selector mit der Portauswahl Detail_Cust_Selector. Die Join-Bedingung lautet: `CustomerID = CustomerNo AND FullName = CustFullName`.

Jede Portauswahl muss dieselbe Anzahl von Ports enthalten. Die Ports müssen denselben Typ aufweisen.

Hinweis: Wenn Sie den Umfang einer Portauswahl ändern und eine Join-Bedingung des einfachen Typs nicht mehr gültig ist, wechselt das Developer Tool möglicherweise den Bedingungstyp in „Erweitert“. Auf der Registerkarte **Join** können Sie den Typ der Join-Bedingung zurück zu „Einfach“ wechseln.

Dynamische Ports in Join-Bedingungen

Sie können in einer Portauswahl auf einen dynamischen Port verweisen.

Ein dynamischer Port kann einen oder mehrere generierte Ports enthalten. Wenn die Join-Bedingung dynamische Ports enthält, muss die Anzahl der Master-Ports mit der Anzahl der Detail-Ports identisch sein.

Beispiel: Ein dynamischer Port A verfügt über 2 generierte Ports:

```
CustomerID
OrderID
```

Ein dynamischer Port B verfügt ebenfalls über 2 generierte Ports:

```
CustomerNo
OrderNo
```

Die folgende Join-Bedingung ist gültig:

DynamicPortA = DynamicPortB

Die Join-Bedingung wird um den folgenden Ausdruck erweitert:

```
CustomerID = CustomerNo AND OrderID = OrderNo
```

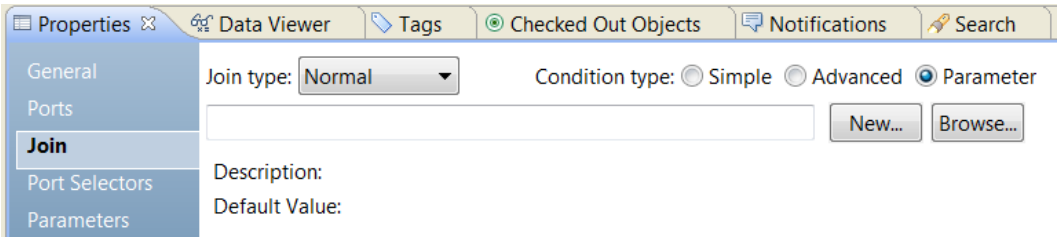
Sie können in einer Join-Bedingung auf eine Portauswahl und einen dynamischen Port verweisen, sofern die Portauswahl dieselbe Anzahl von Ports enthält wie der dynamische Port.

Ausdrucksparameter

Sie können einen Ausdrucksparameter definieren, der eine Join-Bedingung enthält. Sie können die Parameter als Join-Bedingung in der Joiner-Umwandlung auswählen.

Um einen Parameter für die Join-Bedingung zu verwenden, wählen Sie den Parameterbedingungstyp auf der Registerkarte „Join“ aus.

Die folgende Abbildung zeigt an, wo Sie den Parameterbedingungstyp auswählen können:



Sie können nach einem vorhandenen Parameter suchen oder einen Parameter erstellen. Klicken Sie zum Erstellen eines Parameters auf **Neu** und definieren Sie den Parameter. Erstellen Sie den Ausdruck im Ausdruckseditor.

Hinweis: Ein Ausdrucksparameter darf keine anderen Parameter enthalten. Wenn Sie einen Parameter in einen Ausdrucksparameter einbetten, gibt der Datenintegrationsdienst einen Laufzeitvalidierungsfehler aus.

Join-Typen

In einer Joiner-Umwandlung kann das Join aus unterschiedlichen Quelltypen stammen.

Die Joiner-Umwandlung unterstützt die folgenden Join-Typen:

- Normal
- Master-Outer
- Detail-Outer
- Full-Outer

Hinweis: Ein normaler oder Master Outer Join wird schneller durchgeführt als ein Full-Outer- oder Detail-Outer-Join.

Wenn ein Ergebnissatz Felder umfasst, die in keiner der Quellen Daten enthalten, füllt die Joiner-Umwandlung die leeren Felder mit Nullwerten auf. Wenn Sie wissen, dass ein Feld NULL zurückgibt, und Sie keine Nullen im Target einfügen möchten, können Sie einen Standardwert für den entsprechenden Port einrichten.

Normales Join

Mit einem normalen Join verwirft der Data Integration Service basierend auf der Bedingung alle Datenzeilen aus der Master- und Detail-Quelle, die nicht übereinstimmen.

Sie verfügen beispielsweise über zwei Datenquellen für automatische Teile mit der Bezeichnung TEILE_GRÖSSE und TEILE_FARBE.

Die Datenquelle TEILE_GRÖSSE ist die Master-Quelle und enthält die folgenden Daten:

PART_ID1	DESCRIPTION	SIZE
1	Seat Cover	Large
2	Ash Tray	Small
3	Floor Mat	Medium

Die Datenquelle TEILE_FARBE ist die Detail-Quelle und enthält die folgenden Daten:

PART_ID2	DESCRIPTION	COLOR
1	Seat Cover	Blue
3	Floor Mat	Black
4	Fuzzy Dice	Yellow

Um beide Tabellen durch Vergleichen der TEILE_IDs in beiden Quellen zusammenzufügen, setzen Sie die Bedingung wie folgt:

```
PART_ID1 = PART_ID2
```

Wenn Sie diese Tabellen mit einem normalen Join zusammenfügen, enthält der Ergebnissatz die folgenden Daten:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
3	Floor Mat	Medium	Black

Das folgende Beispiel zeigt die äquivalente SQL-Anweisung:

```
SELECT * FROM PARTS_SIZE, PARTS_COLOR WHERE PARTS_SIZE.PART_ID1 = PARTS_COLOR.PART_ID2
```

Master Outer Join

Ein Master Outer Join speichert alle Datenzeilen aus der Detail-Quelle und die übereinstimmenden Zeilen aus der Master-Quelle. Die nicht übereinstimmenden Zeilen aus der Master-Quelle werden verworfen.

Wenn Sie die Beispieltabellen mit einem Master Outer Join und derselben Bedingung zusammenfügen, enthält der Ergebnissatz die folgenden Daten:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
3	Floor Mat	Medium	Black
4	Fuzzy Dice	NULL	Yellow

Da für "Plüschwürfel" keine Größe angegeben wurde, füllt der Data Integration Service das Feld mit NULL aus.

Das folgende Beispiel zeigt die äquivalente SQL-Anweisung:

```
SELECT * FROM PARTS_SIZE RIGHT OUTER JOIN PARTS_COLOR ON (PARTS_COLOR.PART_ID2 =
PARTS_SIZE.PART_ID1)
```

Detail-Outer-Join

Ein Detail-Outer-Join speichert alle Datenzeilen aus der Master-Quelle und die übereinstimmenden Zeilen aus der Detail-Quelle. Die nicht übereinstimmenden Zeilen aus der Detail-Quelle werden verworfen.

Wenn Sie die Beispieltabellen mit einem Detail-Outer-Join und derselben Bedingung zusammenfügen, enthält der Ergebnissatz die folgenden Daten:

PART_ID	DESCRIPTION	SIZE	COLOR
1	Seat Cover	Large	Blue
2	Ash Tray	Small	NULL
3	Floor Mat	Medium	Black

Da für "Aschenbecher" keine Farbe angegeben wurde, füllt der Data Integration Service das Feld mit NULL auf.

Das folgende Beispiel zeigt die äquivalente SQL-Anweisung:

```
SELECT * FROM PARTS_SIZE LEFT OUTER JOIN PARTS_COLOR ON (PARTS_SIZE.PART_ID1 =
PARTS_COLOR.PART_ID2)
```

Full Outer Join

Ein Full Outer Join speichert alle Datenzeilen aus der Master- und der Detail-Quelle.

Wenn Sie die Beispieltabellen mit einem Full Outer Join und derselben Bedingung zusammenfügen, enthält der Ergebnissatz die folgenden Daten:

PARTED	DESCRIPTION	SIZE	Color
1	Seat Cover	Large	Blue
2	Ash Tray	Small	NULL
3	Floor Mat	Medium	Black
4	Fuzzy Dice	NULL	Yellow

Da für "Aschenbecher" keine Farbe und für "Plüschwürfel" keine Größe angegeben wurde, füllt der Data Integration Service die Felder mit NULL aus.

Das folgende Beispiel zeigt die äquivalente SQL-Anweisung:

```
SELECT * FROM PARTS_SIZE FULL OUTER JOIN PARTS_COLOR ON (PARTS_SIZE.PART_ID1 =
PARTS_COLOR.PART_ID2)
```

Sortierte Eingabe für eine Joiner-Umwandlung

Sie können die Leistung der Joiner-Umwandlung mit der Option "Sortierte Eingabe" steigern. Verwenden Sie die sortierte Eingabe, wenn die Daten sortiert sind.

Wenn Sie die Joiner-Umwandlung für die Verwendung sortierter Daten konfigurieren, erhöht der Data Integration Service die Leistung, indem er die Festplatteneingabe und -ausgabe minimiert. Sie erreichen den höchsten Leistungsanstieg, wenn Sie mit umfangreichen Datensätzen arbeiten.

Um ein Mapping für die Verwendung sortierter Daten zu konfigurieren, erstellen und verwalten Sie eine Sortierreihenfolge im Mapping, so dass der Data Integration Service die sortierten Daten beim Verarbeiten der Joiner-Umwandlung verwenden kann. Führen Sie die folgenden Schritte zum Konfigurieren des Mappings durch:

1. Konfigurieren Sie die Sortierreihenfolge der Daten, die Sie zusammenführen möchten.
2. Fügen Sie Umwandlungen hinzu, die die Reihenfolge der sortierten Daten beibehalten.
3. Konfigurieren Sie die Joiner-Umwandlung zur Verwendung sortierter Daten und die Join-Bedingung zur Verwendung der Sortierungsursprungsports. Der Sortierungsursprung stellt die Quelle der sortierten Daten dar.

Konfigurieren der Sortierreihenfolge

Konfigurieren Sie die Sortierreihenfolge, um sicherzustellen, dass der Data Integration Service sortierte Daten an die Joiner-Umwandlung weiterleitet.

Mithilfe einer der folgenden Methoden können Sie die Sortierreihenfolge konfigurieren:

- Verwenden Sie sortierte Einfachdateien. Wenn die Einfachdateien sortierte Daten enthalten, stellen Sie sicher, dass die Reihenfolge der Sortierspalten in jeder Quelldatei übereinstimmen.
- Verwenden Sie sortierte relationale Daten. Verwenden Sie sortierte Ports im relationalen Datenobjekt, um Spalten aus der Quelldatenbank zu sortieren. Konfigurieren Sie die Reihenfolge der sortierten Ports in jedem relationalen Datenobjekt auf die gleiche Weise.
- Verwenden Sie eine Sortierer-Umwandlung, um relationale oder Einfachdatei-Daten zu sortieren. Platzieren Sie eine Sortierer-Umwandlung in den Master- und Detail-Pipelines. Konfigurieren Sie jede Sortierer-Umwandlung, um dieselbe Reihenfolge der Sortierschlüsselports und der Sortierreihenfolgenrichtung zu verwenden.

Wenn Sie nicht oder falsch sortierte Daten an die Joiner-Umwandlung übergeben, die zur Verwendung von sortierten Daten konfiguriert ist, schlägt das Mapping fehl. Der Data Integration Service protokolliert den Fehler in der Protokolldatei.

Hinzufügen von Umwandlungen zum Mapping

Fügen Sie Umwandlungen zum Mapping hinzu, die die Reihenfolge der sortierten Daten in einer Joiner-Umwandlung beibehalten.

Sie können die Joiner-Umwandlung direkt hinter dem Sortierursprung platzieren, um die sortierten Daten beizubehalten.

Verwenden Sie beim Hinzufügen von Umwandlungen zwischen dem Sortierursprung und der Joiner-Umwandlung die folgenden , um sortierte Daten beizubehalten:

- Platzieren Sie keine der folgenden Umwandlungen zwischen dem Sortierursprung und der Joiner-Umwandlung:
  - Rang
  - Union
  - Nicht sortierter Aggregator
  - Mapplets, die eine der vorherigen Umwandlungen enthalten
- Sie können eine sortierte Aggregator-Umwandlung zwischen dem Sortierursprung und der Joiner-Umwandlung platzieren, wenn Sie die folgenden Richtlinien verwenden:
  - Konfigurieren Sie die Aggregat-Umwandlung für sortierte Eingabe.
  - Verwenden Sie dieselben Ports für die Gruppierungsspalten in der Aggregat-Umwandlung wie die Ports im Sortierursprung.
  - Der Gruppierungsport muss dieselbe Reihenfolge wie die Ports im Sortierursprung aufweisen.
- Wenn Sie den Ergebnissatz der Joiner-Umwandlung mit einer anderen Pipeline verbinden, überprüfen Sie, ob die Datenausgabe aus der ersten Joiner-Umwandlung sortiert ist.

Regeln und Richtlinien für Join-Bedingungen

Bestimmte Regeln und Richtlinien werden angewendet, wenn Sie Join-Bedingungen für eine sortierte Joiner-Umwandlung erstellen.

Verwenden Sie die folgenden Richtlinien, wenn Sie Join-Bedingungen erstellen:

- Sie müssen eine einfache Bedingung definieren, die den Gleichheitsoperator verwendet.
- Wenn Sie eine sortierte Aggregator-Umwandlung zwischen dem Sortierursprung und der Joiner-Umwandlung verwenden, behandeln Sie die sortierte Aggregator-Umwandlung als den Sortierursprung, wenn Sie die Join-Bedingung definieren.
- Die in der Join-Bedingung verwendeten Ports müssen müssen den Ports am Sortierursprung übereinstimmen.
- Wenn Sie mehrere Join-Bedingungen konfigurieren, müssen die Ports in der ersten Join-Bedingung mit den ersten Ports im Sortierursprung übereinstimmen.
- Wenn Sie mehrere Bedingungen konfigurieren, muss die Reihenfolge der Bedingungen mit der Reihenfolge der Ports im Sortierursprung übereinstimmen und Sie dürfen keine Ports überspringen.
- Die Anzahl der sortierten Ports im Sortierursprung kann größer als oder gleich der Anzahl der Ports in der Join-Bedingung sein.
- Wenn Sie Ports des Datentyps „Dezimal“ hinzufügen, muss die Gesamtstellenanzahl jedes Ports zu demselben Präzisionsbereich gehören.

  Sie können einen der folgenden gültigen Präzisionsbereiche anwenden:
  - Dezimal 0-18
  - Dezimal 19-28
  - Dezimal 29-38
  - Dezimal 39 und höher

  Wenn Sie beispielsweise die Bedingung `DecimalA = DecimalB` definieren, in der DecimalA die Präzision 15 und DecimalB die Präzision 25 aufweist, ist die Bedingung nicht gültig.

Beispiel für eine Join-Bedingung und Sortierreihenfolge

Dieses Beispiel zeigt eine Joiner-Umwandlung, die die Master- und Detail-Pipelines mit sortierten Ports verbindet.

Konfigurieren Sie Sortierer-Umwandlungen in den Master- und Detail-Pipelines mit den folgenden sortierten Ports:

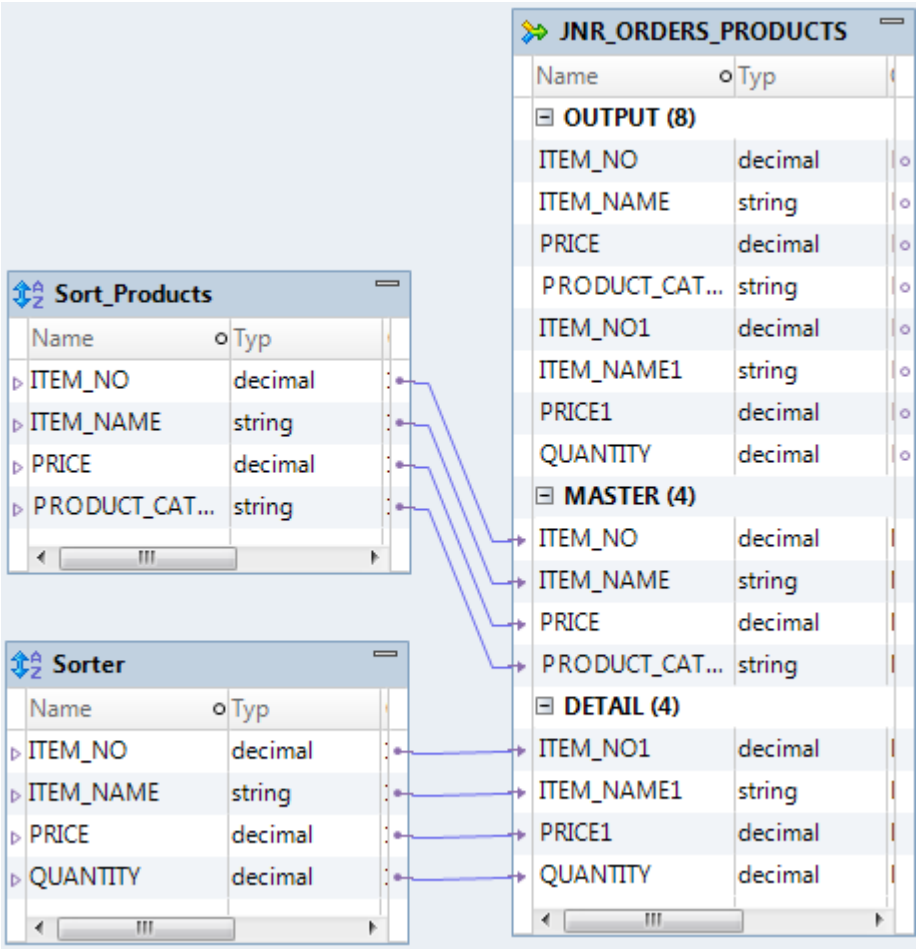
- ITEM_NO
- ITEM_NAME
- PRICE

Verwenden Sie beim Konfigurieren der Join-Bedingung die folgenden Richtlinien, um die Sortierreihenfolge beizubehalten:

- Sie müssen ITEM_NO in der ersten Join-Bedingung verwenden.
- Wenn Sie eine zweite Join-Bedingung hinzufügen, müssen Sie ITEM_NAME verwenden.
- Wenn Sie PRICE in einer Join-Bedingung verwenden möchten, müssen Sie ITEM_NAME ebenfalls in der zweiten Join-Bedingung verwenden.

Wenn Sie ITEM_NAME überspringen und die Optionen ITEM_NO und PRICE verbinden, verlieren Sie die Sortierreihenfolge und der Datenintegrationsdienst kann das Mapping nicht ausführen.

Die folgende Abbildung zeigt ein Mapping, das für Sortierungen und Joins an den Ports ITEM_NO, ITEM_NAME und PRICE konfiguriert wurde:



Wenn Sie mit der Joiner-Umwandlung die Master- und Detail-Pipelines verbinden, können Sie eine beliebige im Folgenden aufgelistete Join-Bedingungen konfigurieren:

```
ITEM_NO = ITEM_NO
```

oder

```
ITEM_NO = ITEM_NO1

ITEM_NAME = ITEM_NAME1
```

oder

```
ITEM_NO = ITEM_NO1

ITEM_NAME = ITEM_NAME1

PRICE = PRICE1
```

Zusammenfügen von Daten aus derselben Quelle

Sie können Daten aus derselben Quelle zusammenfügen, wenn Sie einen Teil der Daten berechnen und die umgewandelten Daten mit den ursprünglichen Daten zusammenfügen möchten.

Wenn Sie Daten aus derselben Quelle zusammenzufügen, können Sie die ursprünglichen Daten beibehalten und Teile dieser Daten innerhalb eines Mappings umwandeln. Zum Zusammenfügen von Daten aus derselben Quelle stehen folgende Möglichkeiten zur Verfügung:

- Zusammenfügen zweier Zweige derselben Pipeline.
- Zusammenfügen zweier Instanzen derselben Quelle.

Zusammenfügen zweier Zweige derselben Pipeline

Wenn Sie Daten aus derselben Quelle zusammenzufügen, können Sie zwei Zweige der Pipeline erstellen.

Wenn Sie eine Pipeline verzweigen, müssen Sie eine Umwandlung zwischen der Mapping-Eingabe und der Joiner-Umwandlung in mindestens einem Zweig der Pipeline hinzufügen. Sie müssen sortierte Daten zusammenfügen und die Joiner-Umwandlung für die sortierte Eingabe konfigurieren.

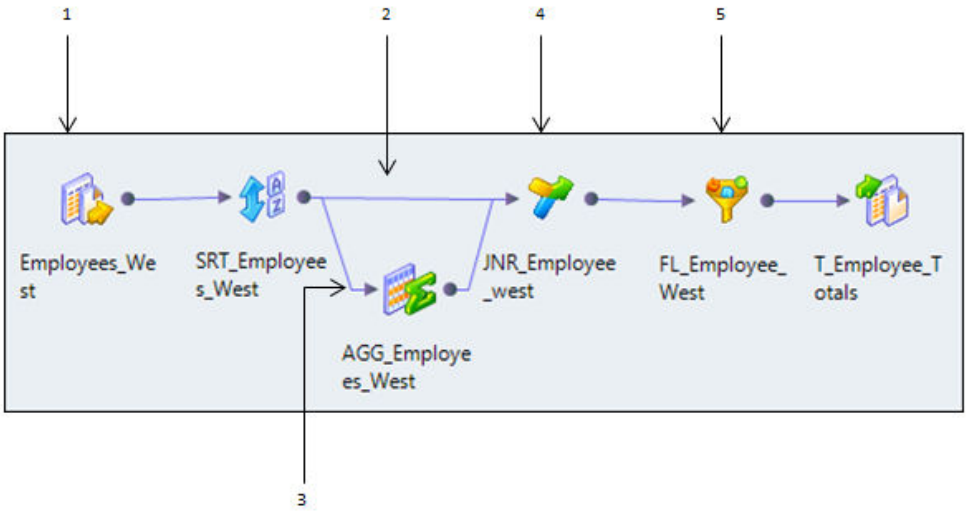
Sie verfügen beispielsweise über eine Quelle mit den folgenden Ports:

- Mitarbeiter
- Abteilung
- Gesamtumsatz

Sie möchten im Target die Mitarbeiter anzeigen, die Umsätze generiert haben, die größer sind als der Durchschnittsumsatz in ihren Abteilungen. Erstellen Sie hierzu ein Mapping mit den folgenden Umwandlungen:

- Sortierer-Umwandlung. Sortiert die Daten.
- Sortierte Aggregator-Umwandlung. Bildet den Mittelwert der Umsätze und gruppiert diese nach Abteilung. Wenn Sie diese Aggregation durchführen, verlieren Sie die Daten für einzelne Mitarbeiter. Zum Beibehalten der Mitarbeiterdaten müssen Sie einen Zweig der Pipeline an die Aggregator-Umwandlung übergeben. Zum Beibehalten der ursprünglichen Daten müssen Sie einen Zweig mit denselben Daten an die Joiner-Umwandlung übergeben. Wenn Sie beide Zweige der Pipeline zusammenfügen, fügen Sie die aggregierten Daten mit den ursprünglichen Daten zusammen.

- Sortierte Joiner-Umwandlung. Fügt die sortierten aggregierten Daten mit den ursprünglichen Daten zusammen.
- FilterUmwandlung. Vergleicht den Durchschnittsumsatz mit den Umsatzdaten jedes Mitarbeiters und filtert die Mitarbeiter mit niedrigeren Durchschnittsumsätzen heraus.



1. Employees_West-Quelle
2. Pipeline-Verzweigung 1
3. Pipeline-Verzweigung 2
4. Sortierte Joiner-Umwandlung
5. Herausfiltern von Mitarbeitern, deren Vertriebszahlen nicht über dem Durchschnitt liegen

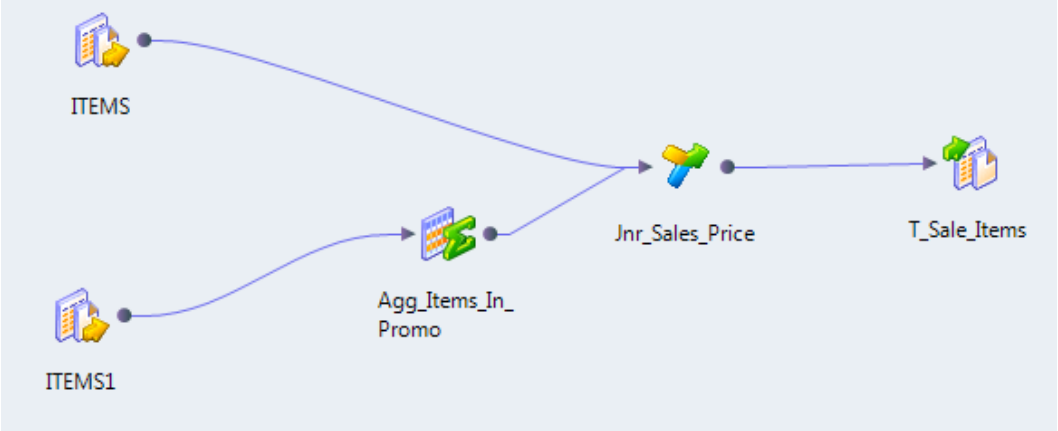
Das Zusammenfügen zweier Zweige kann die Leistung herabsetzen, wenn die Joiner-Umwandlung Daten aus einem Zweig zu einem wesentlich späteren Zeitpunkt als aus dem anderen Zweig empfängt. Bei der Joiner-Umwandlung werden alle Daten aus dem ersten Zweig zwischengespeichert und der Cache wird auf die Festplatte geschrieben, wenn sich der Cache füllt. Die Joiner-Umwandlung muss die Daten dann von der Festplatte lesen, wenn sie die Daten aus dem zweiten Zweig empfängt.

Zusammenfügen zweier Instanzen derselben Quelle

Sie können Daten aus derselben Quelle zusammenfügen, indem Sie eine zweite Instanz der Quelle erstellen.

Nach dem Erstellen der zweiten Quellinstanz können Sie die Pipelines aus den zwei Quellinstanzen zusammenfügen. Wenn Sie unsortierte Daten zusammenfügen möchten, müssen Sie zwei Instanzen derselben Quelle erstellen und die Pipelines zusammenfügen.

Die folgende Abbildung zeigt zwei Instanzen derselben Quelle, die mit einer Joiner-Umwandlung verbunden sind:



Wenn Sie zwei Instanzen derselben Quelle zusammenfügen, liest der Datenintegrationsdienst die Quelldaten für jede Quellinstanz. Im Vergleich zur Verbindung von zwei Pipeline-Zweigen kann die Leistung geringer sein.

Richtlinien für das Zusammenfügen von Daten aus derselben Quelle

Bestimmte Richtlinien gelten, wenn Sie darüber entscheiden, ob Zweige einer Pipeline oder zwei Instanzen einer Quelle zusammengefügt werden sollen.

Verwenden Sie die folgenden Richtlinien, wenn Sie darüber entscheiden, ob Zweige einer Pipeline oder zwei Instanzen einer Quelle zusammengefügt werden sollen:

- Fügen Sie zwei Zweige einer Pipeline zusammen, wenn Sie über eine große Quelle verfügen oder wenn Sie die Quelldaten nur einmal lesen können.
- Fügen Sie zwei Zweige einer Pipeline zusammen, wenn Sie sortierte Daten verwenden. Wenn die Quelldaten unsortiert sind und eine Sortierer-Umwandlung zum Sortieren der Daten verwendet wird, verzweigen Sie die Pipeline nach dem Sortieren der Daten.
- Fügen Sie zwei Instanzen einer Quelle zusammen, wenn Sie zwischen der Quelle und der Joiner-Umwandlung eine blockierende Umwandlung zur Pipeline hinzufügen müssen.
- Fügen Sie zwei Instanzen einer Quelle zusammen, wenn eine Pipeline langsamer verarbeitet wird als die andere Pipeline.
- Fügen Sie zwei Instanzen einer Quelle zusammen, wenn Sie unsortierte Daten zusammenfügen müssen.

Blockieren der Quellen-Pipelines

Beim Ausführen eines Mappings mit einem Joiner-Umwandlung sperrt und entsperrt der Data Integration Service die Quelldaten basierend auf der Mapping-Konfiguration und der Konfiguration der Joiner-Umwandlung für die sortierte Eingabe.

Unsortierte Joiner-Umwandlung

Wenn der Data Integration Service eine unsortierte Joiner-Umwandlung verarbeitet, liest er erst alle Master-Zeilen und anschließend die Detailzeilen. Der Data Integration Service sperrt die Detail-Quelle während der Zwischenspeicherung der Zeilen aus der Master-Quelle.

Nach dem Lesen und Zwischenspeichern der Master-Zeilen sperrt der Data Integration Service die Detail-Quelle und liest die Detailzeilen. Bestimmte Mappings mit unsortierten Joiner-Umwandlungen verletzen die Datenflussvalidierung.

Sortierte Joiner-Umwandlung

Wenn der Data Integration Service eine sortierte Joiner-Umwandlung verarbeitet, sperrt er Daten auf Basis der Mapping-Konfiguration. Die Logik kann gesperrt werden, wenn Master- und Detail-Eingabe in die Joiner-Umwandlung aus verschiedenen Quellen stammen.

Der Data Integration Service verwendet eine Sperrlogik zum Verarbeiten der Joiner-Umwandlung, wenn dies möglich ist, ohne dass alle Quellen in einer Gruppe für die Ladereihenfolge im Zielsystem gesperrt werden. Ist dies nicht möglich, wird keine Sperrlogik verwendet. Stattdessen werden mehrere Zeilen im Cache gespeichert.

Wenn der Data Integration Service die Sperrlogik zum Verarbeiten der Joiner-Umwandlung verwenden kann, werden weniger Zeilen im Cache gespeichert, was zu einer Steigerung der Leistung führt.

Caching von Master-Zeilen

Wenn der Data Integration Service-Prozesse eine Joiner-Umwandlung verarbeitet, liest er Zeilen aus beiden Datenquellen gleichzeitig und erstellt den Index- und Daten-Cache basierend auf den Master-Zeilen.

Der Data Integration Service führt dann das Join basierend auf den Detailquelldaten und den Cache-Daten aus. Die Anzahl der Zeilen, die der Data Integration Service im Cache speichert, hängt von der Datenquelle und davon ab, ob Sie die Joiner-Umwandlung für sortierte Eingabe konfigurieren.

Um die Leistung für eine nicht sortierte Joiner-Umwandlung zu erhöhen, verwenden Sie die Quelle, die weniger Zeilen als die Master-Quelle beinhaltet. Um die Leistung für eine sortierte Joiner-Umwandlung zu erhöhen, verwenden Sie die Quelle, die weniger doppelte Schlüsselwerte als die Master-Quelle beinhaltet.

Leistungstipps für die Joiner-Umwandlung

Verwenden Sie Tipps zur Steigerung der Leistung bei der Joiner-Umwandlung.

Joiner-Umwandlungen können eine geringe Leistung aufweisen, da sie zur Laufzeit zusätzlichen Speicher benötigen, um Zwischenergebnisse zu speichern. Sie können entsprechende Leistungszähler anzeigen, um zu bestimmen, ob die Joiner-Umwandlungen optimiert werden müssen.

Verwenden Sie die folgenden Tipps, um die Leistung bei Joiner-Umwandlungen zu verbessern:

Legen Sie die Master-Quelle als Quelle mit weniger doppelten Schlüsselwerten fest.

Wenn der Datenintegrationsdienst eine sortierte Joiner-Umwandlung verarbeitet, werden Zeilen für 100 eindeutige Schlüssel gleichzeitig zwischengespeichert. Wenn die Master-Quelle viele Zeilen mit demselben Schlüsselwert enthält, muss der Datenintegrationsdienst mehr Zeilen zwischenspeichern, was zu einer Verringerung der Leistung führen kann.

Festlegen der Master-Quelle als Quelle mit weniger Zeilen.

Die Joiner-Umwandlung vergleicht jede Zeile der Detail-Quelle mit der Master-Quelle. Je weniger Zeilen im Master enthalten sind, desto weniger Iterationen des Join-Vergleichs treten auf, was zu einer Beschleunigung des Join-Prozesses führt.

Durchführen von Joins in einer Datenbank, wenn möglich.

Das Durchführen eines Joins in einer Datenbank geht schneller als das Durchführen eines Joins während eines Mapping-Laufs. Der von Ihnen verwendete Typ des Datenbank-Joins kann sich auf die Leistung auswirken. Normale Joins sind schneller als Outer-Joins und resultieren in weniger Zeilen. In bestimmten Fällen kann in der Datenbank kein Join durchgeführt werden, beispielsweise das Zusammenführen von Tabellen aus zwei verschiedenen Datenbanken oder flachen Dateisystemen.

Zusammenfügen sortierter Daten, wenn möglich.

Konfigurieren Sie die Joiner-Umwandlung für die Verwendung sortierter Eingaben. Der Datenintegrationsdienst erhöht die Leistung durch Minimierung der Festplatteneingabe und Festplattenausgabe. Die größte Leistungserhöhung tritt auf, wenn Sie mit umfangreichen Datensätzen arbeiten. Legen Sie bei einer unsortierten Joiner-Umwandlung für die Quelle weniger Zeilen als für die Master-Quelle fest.

Optimieren Sie die Join-Bedingung.

Der Datenintegrationsdienst versucht, die Größe der Datenmenge eines Join-Operanden zu verringern, indem er die Zeilen aus der kleineren Gruppe liest, die übereinstimmenden Zeilen in der größeren Gruppe sucht und dann die Join-Operation durchführt. Das Verringern der Größe der Datenmenge verbessert die Mapping-Leistung, da der Datenintegrationsdienst nicht mehr unnötige Zeilen aus der größeren Quellgruppe liest. Der Datenintegrationsdienst verschiebt die Join-Bedingung in die größere Quellgruppe und liest nur die Zeilen, die mit der kleineren Gruppe übereinstimmen.

Verwenden Sie die Semi-Join-Optimierungsmethode.

Verbessern Sie mit der Semi-Join Optimierungsmethode die Mapping-Leistung, wenn eine Eingabegruppe über mehr Zeilen als die andere Gruppe verfügt und wenn die größere Gruppe viele Zeilen enthält, für die in der kleineren Gruppe keine Übereinstimmung gemäß der Join-Bedingung vorhanden ist.

Regeln und Richtlinien für eine Joiner-Umwandlung

Bestimmte Regeln und Richtlinien werden bei Verwendung einer Joiner-Umwandlung angewendet.

Die Joiner-Umwandlung akzeptiert Eingaben aus den meisten Umwandlungen. Sie können eine Joiner-Umwandlung jedoch nicht verwenden, wenn eine Eingabe-Pipeline eine Update-Strategie-Umwandlung enthält.

KAPITEL 21

Schlüsselgeneratorumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Schlüsselgeneratorumwandlung - Übersicht, 333
- Soundex-Strategie, 334
- String-Strategie, 334
- NYSIIS-Strategie, 335
- Schlüsselgenerator-Ausgabeports, 335
- Konfigurieren einer Gruppierungsstrategie, 336
- Schlüsselerstellungseigenschaften, 336
- Erweiterte Eigenschaften der Schlüsselgeneratorumwandlung, 337

Schlüsselgeneratorumwandlung - Übersicht

Die Schlüsselgeneratorumwandlung ist eine aktive Umwandlung, die Datensätze in Gruppen basierend auf Datenwerten in einer von Ihnen ausgewählten Spalte organisiert. Verwenden Sie diese Umwandlung, um Datensätze vor deren Weiterleitung an die Match-Umwandlung zu sortieren.

Die Schlüsselgeneratorumwandlung verwendet eine Gruppierungsstrategie zum Erstellen der Gruppenschlüssel für die von Ihnen ausgewählte Spalte. Die Strategien sind String, Soundex und NYSIIS. Datensätze mit allgemeinen Werten in dem ausgewählten Feld weisen einen allgemeinen Gruppenschlüsselwert auf. Die Match-Umwandlung verarbeitet Datensätze mit allgemeinen Gruppenschlüsselwerten zusammen. Dies ermöglicht eine schnellere Doppelanalyse in der Match-Umwandlung.

Die Anzahl der Vergleichsoperationen, die die Match-Umwandlung durchführen muss, wächst exponentiell mit der Anzahl der Datensätze im Dataset an. Dieses exponentielle Anwachsen verbraucht sehr viele Rechenressourcen. Beim Erstellen von Gruppenschlüsseln aktiviert die Schlüsselgeneratorumwandlung die Match-Umwandlung, um Datensätze in kleineren Gruppen zu vergleichen, wodurch die Verarbeitungszeit verringert wird.

Wenn Sie Feld-Matching durchführen, wählen Sie eine Spalte für die Erstellung der Gruppenschlüssel aus, die wahrscheinlich nützliche Gruppen für Ihre Matching-Anforderungen bietet. In einer Nachnamen-Spalte sind wahrscheinlich bedeutungsvollere Gruppenschlüsseldaten als in einer Vornamen-Spalte verfügbar. Verwenden Sie die Nachnamen-Spalte aber nicht, wenn Sie diese Spalte für die Doppelanalyse in der Match-Umwandlung auswählen möchten.

Die Schlüsselgeneratorumwandlung kann auch eine eindeutige ID für jeden Datensatz erstellen. Jeder Datensatz, der die Match-Umwandlung durchläuft, muss eine eindeutige ID aufweisen. Verwenden Sie die Schlüsselgeneratorumwandlung, um IDs (falls nicht vorhanden) für Ihre Daten zu erstellen.

Soundex-Strategie

Die Soundex-Strategie analysiert Wörter und erstellt Gruppenschlüssel von Codes, die Wortbetonungen darstellen.

Soundex-Codes beginnen mit dem ersten Buchstaben im Wort, gefolgt von einer Reihe von Zahlen, die sukzessive Konsonanten darstellen. Verwenden Sie die Soundex-Strategie, um denselben Code ähnlich klingenden Wörtern zuzuweisen. Konfigurieren Sie die Soundex-Tiefe, um die Anzahl der alphanumerischen Zeichen zu definieren, die die Strategie zurückgibt.

Diese Strategie fokussiert anstatt der Rechtschreibung den Klang der Wörter, sie kann alternierende Rechtschreibung und geringere Rechtschreibvarianten gruppieren. Die Soundex-Codes für `Smyth` und `Smith` sind beispielsweise identisch.

Die Soundex-Strategie kann auch falsch ausgesprochene Wörter gruppieren. Die Soundex-Codes für die Namen `Edmonton` und `Edmonson` sind beispielsweise identisch.

Soundex-Strategieeigenschaften

Konfigurieren Sie die Soundex-Strategieeigenschaften zum Festlegen der Soundex-Einstellungen, die von der Schlüsselgenerator-Umwandlung zum Erstellen eines Gruppenschlüssels verwendet werden.

In der folgenden Tabelle werden die Soundex-Strategieeigenschaften beschrieben:

Eigenschaft	Beschreibung
Soundex-Tiefe	Bestimmt die Anzahl der alphanumerischen Zeichen, die von der Soundex-Strategie zurückgegeben werden. Die Standardtiefe lautet `3`. Diese Tiefe erstellt eine Soundex-Code, der aus dem ersten Buchstaben in der Zeichenfolge und zwei Zahlen besteht, die die nächsten beiden unterschiedlichen Konsonanten darstellen.

Verwandte Themen:

- "Zeichenfolgen-Strategieeigenschaften" auf Seite 335
- "Schlüsselerstellungseigenschaften" auf Seite 336
- " Konfigurieren einer Gruppierungsstrategie" auf Seite 336

String-Strategie

Die String-Strategie erstellt Gruppenschlüssel aus Substrings in Eingabedaten.

Sie können die Länge und die Position des Substrings innerhalb der Eingabespalte angeben. Sie können diese Strategie beispielsweise zum Erstellen eines Schlüssel aus der ersten vier Zeichen im Eingabestring konfigurieren.

Zeichenfolgen-Strategieeigenschaften

Konfigurieren Sie die Zeichenfolgen-Strategieeigenschaften, um die Unterzeichenfolgen festzulegen, die von der Schlüsselgenerator-Umwandlung zum Erstellen eines Gruppenschlüssels verwendet werden.

In der folgenden Tabelle werden die Zeichenfolgen-Strategie-Eigenschaften beschrieben.

Eigenschaft	Beschreibung
Von links starten	Konfiguriert die Umwandlung zum Lesen des Eingabefelds von links nach rechts.
Von rechts starten	Konfiguriert die Umwandlung zum Lesen des Eingabefelds von rechts nach links.
Start position	Gibt die Anzahl der zu überspringenden Zeichen an. Wenn Sie beispielsweise `3` für die **Startposition** angeben, beginnt die Unterzeichenfolge entweder von links oder von rechts (je nach Angabe) beim vierten Zeichen im Eingabefeld.
Länge	Gibt die Länge der Zeichenfolge an, der als Gruppenschlüssel verwendet werden soll. Geben Sie `0` ein, um das gesamte Eingabefeld zu verwenden.

Verwandte Themen:

- "Soundex-Strategieeigenschaften" auf Seite 334
- "Schlüsselerstellungseigenschaften" auf Seite 336
- " Konfigurieren einer Gruppierungsstrategie" auf Seite 336

NYSIIS-Strategie

Die NYSIIS-Strategie analysiert Wörter und erstellt Gruppenschlüssel mithilfe von Buchstaben, die Wortbetonungen darstellen.

Während die Soundex-Strategie nur den ersten Vokal eines Strings betrachtet, analysiert die NYSIIS-Strategie alle Vokale eines Strings. Die NYSIIS-Strategie wandelt alle Buchstaben in ein von sechs Zeichen und die meisten Vokale in den Buchstaben `A` um.

Schlüsselgenerator-Ausgabeports

Die Ausgabeports der Schlüsselgeneratorumwandlung erstellen IDs und Gruppenschlüssel, die die Match-Umwandlung zur Verarbeitung von Datensätzen verwendet.

In der folgenden Tabelle werden die Ausgabeports für die Schlüsselgeneratorumwandlung erläutert:

Eigenschaft	Beschreibung
SequenceID	Erstellt eine ID, die jeden Datensatz im Quelldatensatz identifiziert.
GroupKey	Erstellt den Gruppenschlüssel, den die Match-Umwandlung zur Verarbeitung von Datensätzen verwendet.

Beim Erstellen einer wiederverwendbaren Schlüsselgeneratorumwandlung verwenden Sie die Ansicht **Übersicht**, um die Ports anzuzeigen. Beim Hinzufügen einer nicht wiederverwendbaren Umwandlung zu einem Mapping verwenden Sie die Registerkarte **Ports** der Ansicht **Eigenschaften**, um die Ports anzuzeigen.

Konfigurieren einer Gruppierungsstrategie

Bearbeiten Sie zum Konfigurieren einer Gruppierungsstrategie die Eigenschaften in der Ansicht **Strategien**.

Bevor Sie eine Schlüsselgenerator-Strategie konfigurieren, fügen Sie der Schlüsselgeneratorumwandlung Eingabeports hinzu.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Klicken Sie auf die Schaltfläche **Neu**.
3. Wählen Sie eine Gruppierungsstrategie aus.
4. Klicken Sie auf **OK**.
5. Wählen Sie im Abschnitt **Eingaben** einen Eingabeport aus.
6. Konfigurieren Sie die Strategieeigenschaften, indem Sie im Eigenschaftenfeld auf den Auswahlpfeil klicken.
7. Konfigurieren Sie die Eigenschaften für die Schlüsselerstellung.

Verwandte Themen:

- "Soundex-Strategieeigenschaften" auf Seite 334
- "Zeichenfolgen-Strategieeigenschaften" auf Seite 335
- "Schlüsselerstellungseigenschaften" auf Seite 336

Schlüsselerstellungseigenschaften

Konfigurieren Sie die Eigenschaften für die Schlüsselerstellung, die für die Analyse Ihrer Daten geeignet sind.

In der folgenden Tabelle werden die Eigenschaften für die Schlüsselerstellung beschrieben:

Eigenschaft	Beschreibung
Ergebnisse sortieren	Sortiert die Schlüsselgeneratorumwandlungsausgabe mit dem GroupKey-Feld. Wählen Sie diese Option für Feld-Matching-Operationen aus oder stellen Sie sicher, dass Sie die Match-Umwandlung mit sortierten Daten bereitstellen. Wählen Sie diese Option nicht für Identitäts-Matching-Operationen aus.
Sequenzschlüssel automatisch generieren	Generiert ein Sequenzschlüsselfeld unter Verwendung der Reihenfolge der Eingabedaten.
Feld als Sequenzschlüssel verwenden	Generiert ein Sequenzfeld für eine von Ihnen angegebene Spalte.
Sequenzschlüsselfeld	Gibt den Namen des Sequenzschlüsselfeldes an.

Verwandte Themen:

- "Soundex-Strategieeigenschaften" auf Seite 334
- "Zeichenfolgen-Strategieeigenschaften" auf Seite 335
- " Konfigurieren einer Gruppierungsstrategie" auf Seite 336

Erweiterte Eigenschaften der Schlüsselgeneratorumwandlung

Die Schlüsselgeneratorumwandlung enthält erweiterte Eigenschaften, die das Verhalten des Cache-Arbeitsspeichers und die Tracingebene bestimmen.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Cache-Dateiverzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst temporäre Daten für die aktuelle Umwandlung schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis, wenn das Volumen der Eingabedaten größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cache-Dateigröße

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst zum Sortieren der Eingabedaten bei der Umwandlung verwendet. Sie können einen Parameter verwenden, um die Cache-Dateigröße anzugeben.

Bevor die Daten sortiert werden, weist der Datenintegrationsdienst die von Ihnen angegebene Speichermenge zu. Wenn der Sortiervorgang eine größere Datenmenge generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Verzeichnis. Wenn der Speicherbedarf für den Sortiervorgang die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Wenn Sie keine Cache-Dateigröße angeben, wendet die Umwandlung den maximalen Wert für die Speichergröße auf die Ausführungsoptionen des Datenintegrationsdiensts an.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz", „Normal", „Ausführliche Intialisierung" oder „Ausführliche Daten" wählen. Standardwert ist „Normal".

KAPITEL 22

Labeler-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Labeler-Umwandlung - Übersicht, 338
- Einsatz der Labeler-Umwandlung, 339
- Referenzdatenverwendung in der Labeler-Umwandlung, 340
- Labeler-Umwandlung - Strategien, 342
- Beschriftungsumwandlung - Ports, 343
- Zeichenbeschriftungs-Eigenschaften, 344
- Tokenbeschriftungseigenschaften, 346
- Konfigurieren einer Zeichenbeschriftungs-Strategie, 349
- Konfigurieren einer Tokenbeschriftungsstrategie, 350
- Labeler-Umwandlung – Erweiterte Eigenschaften, 350

Labeler-Umwandlung - Übersicht

Die Labeler-Umwandlung ist eine passive Umwandlung, die Eingabeportfelder analysiert und Textbeschriftungen schreibt, die die Daten in jedem Feld beschreiben.

Sie verwenden eine Labeler-Umwandlung, wenn Sie die Typen von Informationen verstehen möchten, die ein Port enthält. Verwenden Sie eine Labeler-Umwandlung, wenn Sie die Informationstypen auf einem Port nicht kennen oder wenn Sie Datensätze ermitteln möchten, die nicht die erwarteten Informationstypen auf einem Port enthalten.

Eine Beschriftung ist ein String aus einem oder mehreren Zeichen zur Beschreibung eines Eingabestrings. Sie konfigurieren die Labeler-Umwandlung, um Eingabestrings basierend auf den Daten, die jeder String enthält, Beschriftungen zuzuweisen.

Beim Konfigurieren der Umwandlung legen Sie die zu suchenden Zeichen- oder Stringtypen fest und geben die Beschriftung an, die von der Umwandlung als Ausgabe geschrieben wird, wenn das entsprechende Zeichen oder der entsprechende String gefunden wird. Sie geben die zu suchenden Zeichen- und Stringtypen sowie die zu verwendenden Beschriftungen ein, wenn Sie eine Beschriftungsoperation konfigurieren. Sie können aber auch Referenzdatenobjekte zum Festlegen der Zeichen, Strings und Beschriftungen verwenden.

Sie konfigurieren die Umwandlung zum Durchführen der Zeichen- oder Tokenbeschriftung:

Zeichenbeschriftung

Schreibt eine Beschriftung, die die Zeichenstruktur des Eingabestrings beschreibt, einschließlich Satz- und Leerzeichen. Die Umwandlung schreibt eine einzelne Beschriftung für jede Zeile in einer Spalte. Die

Labeler-Umwandlung kann beispielsweise die Postleitzahl 10028 als "nnnnn" beschriften, wobei "n" ein numerisches Zeichen ist.

Tokenbeschriftung

Schreibt eine Beschriftung, die den Informationstyp im Eingabestring beschreibt. Die Umwandlung schreibt eine Beschriftung für jeden in den Eingabedaten erkannten Token. Sie können beispielsweise die Labeler-Umwandlung konfigurieren, um den String "John J. Smith" mit den Token "Word Init Word" zu beschriften.

Ein Token ist ein eingegrenzter Wert in einem Eingabestring.

Wenn die Labeler-Umwandlung ein Zeichen oder einen String findet, der einer von Ihnen festgelegten Beschriftung entspricht, wird der Beschriftungsname auf einen neuen Ausgabeport geschrieben.

Die Labeler-Umwandlung verwendet Referenzdaten, um Zeichen und Token anzugeben. Sie wählen das Referenzdatenobjekt aus, wenn Sie eine Operation in einer Labeler-Strategie konfigurieren.

Einsatz der Labeler-Umwandlung

Die Labeler-Umwandlung schreibt eine beschreibende Beschriftung für jeden Wert auf einem Port.

Die folgenden Beispiele beschreiben einige Analysetypen, die mit einer Labeler-Umwandlung durchgeführt werden können.

Auffinden von Datensätzen mit Kontaktdaten

Konfigurieren Sie die Umwandlung mit einer Referenztabelle, die eine Liste mit Vornamen enthält. Erstellen Sie eine Tokenbeschriftungsstrategie zum Beschriften eines beliebigen Strings, der mit einem Wert in der Referenztabelle übereinstimmt. Wenn Sie die Ausgabedaten überprüfen, gibt jeder Datensatz, der die Beschriftung enthält, wahrscheinlich eine Person an.

Auffinden von Unternehmensdatensätzen

Konfigurieren Sie die Umwandlung mit einem Tokensatz, der eine Liste der Unternehmenssuffixe enthält, z. B. Inc, Corp und Ltd. Erstellen Sie eine Tokenbeschriftungsstrategie zum Beschriften eines beliebigen Strings, der mit einem Wert in der Referenztabelle übereinstimmt. Wenn Sie die Ausgabedaten überprüfen, gibt jeder Datensatz, der die Beschriftung enthält, wahrscheinlich ein Unternehmen an.

Hinweis: Verwenden Sie einen Tokensatz der gewünschten Unternehmenssuffixe, um beliebige Unternehmensnamen anzugeben. Sie können eine Referenztabelle mit Unternehmensnamen verwenden, wenn Sie sicher sind, dass die Tabelle alle Unternehmen enthält, die angegeben werden sollen. Sie können beispielsweise eine Referenztabelle verwenden, in der die an der New Yorker Börse geführten Unternehmen aufgelistet werden.

Auffinden von Telefonnummerdaten

Konfigurieren Sie die Umwandlung mit Zeichensätzen, die die Zeichenstruktur einer Telefonnummer definieren. Sie können beispielsweise einen Zeichensatz verwenden, der verschiedene Muster aus Satzzeichen und Ziffern als US-amerikanische Telefonnummern erkennt. Sie können die Daten überprüfen und nach Datensätzen suchen, die nicht die richtigen Ziffern für eine Telefonnummer enthalten.

Die Zeichenbeschriftungen können gegebenenfalls die folgenden Zeichen zum Analysieren der Spaltendaten verwenden.

```
c=punctuation character n=digit s=space
```

Die folgende Tabelle zeigt Beispiele für Telefonnummerstrukturen:

Zeichenstruktur	Telefonnummer
cnnncsnnncnnncnnnnn	(212) 555-1212
nnnnnnnnnn	2125551212
cnnncnnncnnnn	+212-555-1212

Referenzdatenverwendung in der Labeler-Umwandlung

Informatica Developer installiert Referenzdatenobjekte unterschiedlichen Typs, die Sie mit der Bezeichnungsumwandlung verwenden können. Sie können auch Referenzdatenobjekte erstellen.

Beim Hinzufügen eines Referenzdatenobjekts zur Bezeichnungsumwandlungsstrategie durchsucht die Umwandlung die Eingabedaten in der Strategie nach den Werten im Referenzdatenobjekt. Die Umwandlung ersetzt alle Werte, die gefunden werden können, durch einen gültigen Wert aus dem Referenzdatenobjekt oder durch einen Wert, der von Ihnen festgelegt wird.

In der folgenden Tabelle werden die Referenzdatentypen beschrieben, die verwendet werden können:

Referenzdatentyp	Beschreibung
Zeichensätze	Gibt verschiedene Arten von Zeichen an, z. B. Buchstaben, Zahlen und Satzzeichen. Verwendung in Vorgängen für Zeichenbeschriftung.
Probabilistische Modelle	Fügt Tokenbeschriftungs-Operationen Fuzzy-Match-Funktionen hinzu. Die Umwandlung kann ein probabilistisches Modell verwenden, um den Informationstyp in einem String abzuleiten. Zum Aktivieren der Fuzzy-Match-Funktionen kompilieren Sie das probabilistische Modell im Developer Tool. Verwendung in Vorgängen für Tokenbeschriftung.
Referenztabellen	Sucht nach Strings, die mit den Einträgen in einer Datenbanktabelle übereinstimmen. Verwendung in Vorgängen für Tokenbeschriftung und Zeichenbeschriftung.
Reguläre Ausdrücke	Erkennt Strings, die mit von Ihnen definierten Bedingungen übereinstimmen. Sie können einen regulären Ausdruck verwenden, um nach einem String innerhalb eines längeren Strings zu suchen. Verwendung in Vorgängen für Tokenbeschriftung.
Tokensätze	Erkennt Strings basierend auf den enthaltenen Informationstypen. Verwendung in Vorgängen für Tokenbeschriftung. Informatica installiert mit Tokensätzen verschiedene Arten von Tokendefinitionen, wie z. B. Wörter, Telefonnummern, Postleitzahlen und Produktcodedefinitionen.

Zeichensätze

Ein Zeichensatz enthält Expressions, die bestimmte Zeichen und Zeichenbereiche kennzeichnen. Sie können Zeichensätze in Labeler- und in Parser-Umwandlungen im Tokenbeschriftungsmodus verwenden.

Zeichenbereiche geben Sie einen sequentielle Bereich von Zeichencodes an. Der Zeichenbereich "[A-C]" entspricht beispielsweise den Großbuchstaben "A," "B" und "C." Dieser Zeichenbereich stimmt nicht mit den Kleinbuchstaben "a," "b" oder "c" überein.

Verwenden Sie Zeichensätze zur Angabe eines Sonderzeichens oder von Zeichenbereichen als Teil von Token-Parsing- oder Beschriftungsoperationen. Sie können beispielsweise alle Nummernzeichen in einer Spalte kennzeichnen, die Telefonnummern beinhalten. Nach der Beschriftung der Zahlen können Sie Muster mit einer Parser-Umwandlung angeben und problematische Muster in separate Ausgaben schreiben.

Probabilistische Modelle

Ein probabilistisches Modell erkennt Token anhand des Typs der enthaltenen Informationen sowie deren Positionen in einer Eingabezeichenfolge.

Ein probabilistisches Modell enthält Referenzdatenwerte und Beschriftungswerte. Die Referenzdatenwerte stellen die Daten auf einem Eingabeport dar, den Sie mit der Umwandlung verbinden. Die Beschriftungswerte beschreiben die Typen von Informationen, die die Referenzdatenwerte enthalten. Sie weisen jedem Referenzdatenwert im Modell eine Beschriftung zu.

Sie kompilieren das Modell, um die Referenzdatenzeilen mit den Beschriftungen in einem probabilistischen Modell zu verknüpfen. Der Kompilierungprozess generiert eine Reihe von logischen Assoziationen zwischen den Datenwerten und den Beschriftungen. Wenn Sie ein Mapping ausführen, das das Modell liest, wendet der Datenintegrationsdienst die Logik des Modells auf die Eingabedaten der Umwandlung an. Der Datenintegrationsdienst gibt die Beschriftung zurück, die die Eingabedatenwerte am genauesten beschreibt.

Sie erstellen ein probabilistisches Modell im Developer Tool. Das Modellrepository speichert das probabilistische Modellobjekt. Das Developer-Tool schreibt die Datenwerte, die Beschriftungen und die Kompilierungsdaten in eine Datei in der Informatica-Verzeichnisstruktur.

Referenztabellen

Eine Referenztabelle ist eine Datenbanktabelle mit mindestens zwei Spalten. Eine Spalte enthält die Standard- oder die erforderliche Version eines Datenwerts. Andere Spalten enthalten alternative Versionen des Werts. Beim Hinzufügen einer Referenztabelle zu einer Umwandlung durchsucht die Umwandlung die Eingabeportdaten auf Werte, die auch in der Tabelle angezeigt werden. Sie können Tabellen mit beliebigen nützlichen Daten für das Datenprojekt erstellen, an dem Sie arbeiten.

Reguläre Ausdrücke

Im Kontext einer Beschriftungsoperation ist ein regulärer Ausdruck ein Ausdruck, den Sie verwenden können, um einen bestimmten String in Eingabedaten zu kennzeichnen. Sie können reguläre Ausdrücke in Labeler-Umwandlungen im Token-Beschriftungsmodus verwenden.

Labeler-Umwandlungen verwenden reguläre Ausdrücke, um ein Eingabemuster abzugleichen und eine einzelne Beschriftung zu erstellen. Reguläre Ausdrücke mit mehreren Ausgaben generieren keine Mehrfachbeschriftungen.

Tokensätze

Ein Tokensatz enthält Ausdrücke, die bestimmte Token kennzeichnen. Sie können Tokensätze in Labeler-Umwandlungen verwenden, die im Modus "Tokenbeschriftung" ausgeführt werden.

Verwenden Sie Tokensätze, um bestimmte Token als Teil der Tokenbeschriftungsoperationen zu kennzeichnen. Sie können einen Tokensatz beispielsweise zum Beschriften aller E-Mail Adressen verwenden, die das Format "Kontoname@Domänenname" aufweisen. Nach dem Beschriften der Token können Sie die Parser-Umwandlung verwenden, um E-Mail-Adressen auf angegebene Ausgabeports zu schreiben.

Das Developer Tool enthält vom System definierte Tokensätze, die zum Kennzeichnen eines großen Bereichs an Mustern verwendet werden können. Zu den Beispielen für vom System definierte Tokensätze gehören:

- Wörter
- Zahlen
- Telefonnummern
- E-Mail-Adressen
- Postleitzahlen
- Nationale Identifikationsnummern, z. B. Sozialversicherungsnummern
- Kreditkartennummern

Labeler-Umwandlung - Strategien

Verwenden Sie Beschriftungsstrategien, um Eingabedaten Beschriftungen zuzuweisen. Um eine Beschriftungsstrategie zu konfigurieren, bearbeiten Sie die Einstellungen in der Ansicht **Strategien** der Labeler-Umwandlung.

Beim Erstellen einer Beschriftungsstrategie fügen Sie eine oder mehrere Operationen hinzu. Jede Operation implementiert eine bestimmte Beschriftungsaufgabe.

Die Labeler-Umwandlung bietet einen Assistenten, den Sie zum Erstellen von Strategien verwenden können. Beim Erstellen einer Beschriftungsstrategie wählen Sie entweder den Zeichen- oder den Tokenbeschriftungsmodus aus. Anschließend fügen Sie Operationen hinzu, die diesem Beschriftungsmodus entsprechen.

Wichtig: Sie können die Reihenfolge der Operationen und Strategien ändern. Die Reihenfolge der Operationen in einer Strategie können die Ausgabe einer Strategie ändern, weil jede Operation die Ergebnisse der vorherigen Operation liest.

Zeichenkennzeichnungsoperationen

Verwenden Sie Zeichenbeschriftungsoperationen, um Beschriftungen zu erstellen, die die Zeichenmuster in Ihren Daten beschreiben.

Einer Zeichenbeschriftungsstrategie können Sie die folgenden Operationstypen hinzufügen:

Zeichen mithilfe von Zeichensätzen beschriften

Zeichen mithilfe von vordefinierten Zeichensätze beschriften, zum Beispiel Ziffern oder Buchstaben. Sie können Unicode- und Nicht-Unicode-Zeichensätze auswählen.

Zeichen mithilfe von Referenztabellen beschriften

Zeichen mit benutzerdefinierten Beschriftungen aus einer Referenztabelle beschriften.

Tokenbeschriftungs-Operationen

Verwenden Sie Tokenbeschriftungs-Operationen zum Erstellen von Beschriftungen, die Strings in Ihren Daten beschreiben.

Die Labeler-Umwandlung kann mehrere Token in einem Eingabestring erkennen und beschriften. Sie können die Labeler-Umwandlung beispielsweise so konfigurieren, dass sie die Tokensätze US-Telefonnummer und E-Mail-Adressen verwendet. Wenn die Labeler-Umwandlung den Eingabestring "555-555-1212 someone@somewhere.com" verarbeitet, lautet der Ausgabestring "USPHONE EMAIL".

Sie können die folgenden Typen von Tokenbeschriftungs-Operationen zu einer Beschriftungsstrategie hinzufügen:

Beschriften mit Referenztabelle

Beschriften von Strings, die mit Einträgen in der Referenztabelle übereinstimmen.

Beschriften von Token mit Tokensätzen

Beschriften Sie Stringmuster, die mit Tokensatzdaten oder Daten des probabilistischen Modells übereinstimmen.

Beschriftungsumwandlung - Ports

Sie wählen die Eingabe- und Ausgabeports aus, die Sie für die in der Umwandlung konfigurierten Beschriftungsoperationen benötigen.

Beschriftungsumwandlungen verwenden den folgenden Port:

Eingabeports

Liest die Zeichenfolgenausgabe aus vorangehenden Objekten.

Beschriftete Ausgabeports

Schreibt die Beschriftungen, die von Umwandlungsoperationen definiert wurden.

Tokenisierte Ausgabeports

Übergibt Eingabezeichenfolgen, die mit jeder Beschriftung in der Ausgabe übereinstimmen. Wählen Sie diesen Port aus, wenn Sie eine Parserumwandlung unterhalb der Beschriftungsumwandlung in einem Mapplet oder Mapping hinzufügen oder wenn Sie die Parserumwandlung zum Ausführen im musterbasierten Parsing-Modus konfigurieren. Die Parserumwandlung verbindet die Tokenbeschriftungsausgabe mit den Daten auf den tokenisierten Ausgabeports.

Score-Ausgabeports

Wählen Sie diese Option aus, um Score-Werte zu schreiben, die von probabilistischen Matching-Methoden in einer Tokenbeschriftungsoperation erzeugt wurden.

Wenn Sie eine Tokenbeschriftungsoperation ausführen, die ein probabilistisches Modell verwendet, erzeugt die Operation einen numerischen Score für jede beschriftete Zeichenfolge. Der Score spiegelt den Grad der Ähnlichkeit zwischen der Eingabezeichenfolge und den Mustern wider, die im probabilistischen Modell definiert wurden.

Zeichenbeschriftungs-Eigenschaften

Konfigurieren Sie die Eigenschaften für Zeichenbeschriftungs-Operationen in der Ansicht **Strategien** in der Labeler-Umwandlung.

Allgemeine Eigenschaften

Allgemeine Eigenschaften gelten für alle Zeichenbeschriftungs-Operationen, die in der Strategie definiert werden. Verwenden Sie die allgemeinen Eigenschaften, um die Strategie zu benennen und Eingabe- und Ausgabeports anzugeben.

In der folgenden Tabelle werden die allgemeinen Eigenschaften beschrieben:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Strategie bereit.
Eingaben	Gibt die Eingabeports an, die von den Strategieoperationen gelesen werden können.
Ausgaben	Gibt die Ausgabeports an, in die Strategieoperationen schreiben können.
Beschreibung	Stellt eine Textbeschreibung der Strategie bereit. Dies ist eine optionale Eigenschaft.

Referenztabelleneigenschaften

Wenn Sie eine Zeichenbeschriftungs-Strategie definieren, können Sie Operationen zur Beschriftung mit Zeichensätzen und Referenztabellen hinzufügen. Verwenden Sie die Referenztabelleneigenschaften, um anzugeben, wie die Umwandlung Referenztabellen verwendet.

Die folgende Tabelle beschreibt die Referenztabelleneigenschaften:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Referenztabelle	Gibt Referenztabellen an, die von der Umwandlung zur Beschriftung von Zeichen verwendet wird.
Beschriftung	Gibt den Ersetzungstext für Eingabezeichen an, die mit den Referenztabelleneinträgen übereinstimmen.
Andere Beschriftungen in der Strategie überschreiben	Bestimmt, ob diese Beschriftungsoperation andere Beschriftungsoperationen überschreibt.

Zeichensatzeigenschaften

Wenn Sie eine Zeichenbeschriftungs-Strategie definieren, können Sie Operationen zur Beschriftung mit Zeichensätzen und Referenztabellen hinzufügen. Verwenden Sie die Zeichensatzeigenschaften, um die Verwendung von Zeichensätzen durch die Transformation festzulegen.

Die folgende Tabelle beschreibt die Zeichensatzeigenschaften:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Zeichensätze wählen	Gibt die Zeichensätze an, die von der Umwandlung zum Beschriften von Zeichenfolgen verwendet werden. Sie können den Ersetzungstext für Eingabezeichenfolgen überschreiben, die dem Zeichensatz entsprechen. Klicken Sie auf den Auswahlpfeil in der Spalte **Beschriftung** , um benutzerdefinierten Ersetzungstext einzugeben.
Text filtern	Verwendet Zeichen oder Platzhalter, die zum Filtern der Liste mit Zeichensätzen eingegeben wurden.
Zeichensatz hinzufügen	Wählen Sie diese Option aus, um einen benutzerdefinierten Zeichensatz zu definieren.
Bearbeiten	Bearbeiten Sie den Inhalt eines benutzerdefinierten Zeichensatzes.
Importieren	Ermöglicht das Erstellen nicht wiederverwendbarer Kopien von Zeichensätzen, die in Inhaltssätzen gespeichert sind. Änderungen am ursprünglichen Zeichensatz führen nicht zu einer Aktualisierung der Kopien, die in der Beschriftungsumwandlung gespeichert wurden.
Löschen	Löscht einen benutzerdefinierten Zeichensatz.
Ausführungsreihenfolge angeben	Legt die Reihenfolge fest, in der die Operation die Tokensätze auf die Daten anwendet. Verwenden Sie die Aufwärts- und Abwärtspfeile, um die Reihenfolge zu ändern.

Filtereigenschaften

Sie können Werte festlegen, die während einer Beschriftungsoperation übersprungen werden sollen. Verwenden Sie die Eigenschaften unter **Text ignorieren**, um Werte festzulegen, auf die Beschriftungsoperationen nicht angewendet werden.

In der folgenden Tabelle werden die Filtereigenschaften beschrieben:

Eigenschaft	Beschreibung
Suchbegriff	Gibt Strings an, die die Umwandlung vor dem Ausführen der Beschriftung filtert. Verwenden Sie diese Funktion, um Ausnahmen in Ihrer definierten Beschriftungsstrategie anzugeben.
Groß-/ Kleinschreibung	Gibt an, ob gefilterte Strings mit der Groß-/Kleinschreibung des Suchbegriffs übereinstimmen müssen.
Großbuchstaben	Wandelt gefiltert Strings in Großbuchstaben um.

Eigenschaft	Beschreibung
Start	Gibt die Zeichenposition an, an der die Suche nach dem gefilterten String gestartet werden soll.
Ende	Gibt die Zeichenposition an, an der die Suche nach dem gefilterten String beendet werden soll.

Tokenbeschriftungseigenschaften

Konfigurieren Sie die Eigenschaften für Zeichenbeschriftungs-Operationen in der Ansicht **Strategien** in der Labeler-Umwandlung.

Allgemeine Eigenschaften

Allgemeine Eigenschaften gelten für alle in der Strategie definierten Tokenbeschriftungs-Operationen. Verwenden Sie die allgemeinen Eigenschaften, um die Strategie zu benennen, Eingabe- und Ausgabeports festzulegen und anzugeben, ob die Strategie probabilistische Matching-Methoden ermöglicht.

In der folgenden Tabelle werden die allgemeinen Eigenschaften beschrieben:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Strategie bereit.
Eingaben	Gibt die Ausgabeports an, die von den Strategieoperationen gelesen werden können.
Ausgaben	Gibt die Ausgabeports an, auf die die Strategieoperationen schreiben können.
Beschreibung	Beschreibt die Strategie. Die Eigenschaft ist optional.
Probabilistische Matching-Methoden verwenden	Gibt an, dass die Strategie ein probabilistisches Modell zur Angabe von Tokentypen verwenden kann.
Umkehr aktiviert	Gibt an, dass die Strategie Eingabedaten von rechts nach links liest. Diese Eigenschaft ist für probabilistisches Matching deaktiviert.
Delimiter	Gibt die Zeichen an, die von der Umwandlung beim Evaluieren von Substrings in Eingabedaten verwendet werden. Standardwert ist Leerzeichen. Die Eigenschaft ist bei probabilistischen Labels deaktiviert.
Tokenisiertes Ausgabefeld	Gibt an, dass die Strategie mehrere Beschriftungen auf einen Ausgabeport schreibt. Wählen Sie dieses Feld aus, um Eingabedaten für das musterbasierte Parsing in der Parserumwandlung zu erstellen.

Eigenschaft	Beschreibung
Score-Ausgabefeld	Gibt das Feld an, das die beim probabilistischen Matching erzeugten Score-Werte enthält. Legen Sie das Score-Ausgabefeld fest, wenn Sie die Option zum Verwenden probabilistischer Matching-Methoden verwenden.
Ausgabe-Delimiter	Gibt ein Zeichen an, das zum Trennen von Datenwerten auf dem Ausgabeport verwendet wird. Standardwert ist Doppelpunkt.

Eigenschaften von Tokensätzen

Tokensatzeigenschaften werden angewendet, wenn Sie eine Beschriftungsoperation zur Verwendung von Tokensätzen konfigurieren.

In der folgenden Tabelle werden die allgemeinen Eigenschaften beschrieben:

Eigenschaft	Beschreibung
Tokensätze auswählen	Gibt die Tokensätze an, die von der Umwandlung zum Beschriften von Zeichenfolgen verwendet werden.
Text filtern	Filtert die Liste von Tokensätzen oder regulären Ausdrücken. Verwenden Sie Textzeichen und Platzhalterzeichen als Filter.
Tokensatz hinzufügen	Verwenden Sie diese Option zum Definieren von benutzerdefinierten Tokensätzen.
Regulären Ausdruck hinzufügen	Verwenden Sie diese Option zum Definieren von regulären Ausdrücken, die mit einem Eingabemuster übereinstimmen.
Bearbeiten	Bearbeitet die Inhalte eines benutzerdefinierten Tokensatzes oder regulären Ausdrucks.
Importieren	Importiert eine nicht wiederverwendbare Kopie eines Tokensatzes oder regulären Ausdrucks aus einem Ordner im Modellrepository. Wenn Sie das Quellobjekt für den Tokensatz oder regulären Ausdruck aktualisieren, aktualisiert der Datenintegrationsdienst nicht die nicht wiederverwendbare Kopie.
Entfernen	Entfernt einen benutzerdefinierten Tokensatz oder regulären Ausdruck.
Ausführungsreihenfolge angeben	Legt die Reihenfolge fest, in der der Vorgang die Tokensätze oder regulären Ausdrücke auf die Daten anwendet. Verwenden Sie die Aufwärts- und Abwärtspfeile, um die Reihenfolge zu ändern.

Benutzerdefinierte Beschriftungseigenschaften

Wenn Sie eine Tokenbeschriftungsoperation konfigurieren, können Sie die Ansicht **Benutzerdefinierte Beschriftung** zum Erstellen von Beschriftungen für bestimmte Suchbegriffe auswählen.

In der folgenden Tabelle werden die benutzerdefinierten Beschriftungseigenschaften beschrieben:

Eigenschaft	Beschreibung
Suchbegriff	Gibt den String an, nach dem gesucht werden soll.
Groß-/Kleinschreibung	Gibt an, ob die Eingabedaten mit der Groß-/Kleinschreibung des Suchbegriffs übereinstimmen müssen.
Benutzerdefinierte Beschriftung	Gibt die anzuwendende benutzerdefinierte Beschriftung an.

Eigenschaften für das probabilistische Matching

Wenn Sie die Optionen zum Verwenden probabilistischer Matching-Methoden auswählen, können Sie der Beschriftungsoperation ein probabilistischen Modell hinzufügen. Einer Strategie, die einen Tokensatz oder eine Referenztabelle verwendet, kann kein probabilistisches Modell hinzugefügt werden.

In der folgenden Tabelle werden die dem probabilistischen Matching zugeordneten Eigenschaften beschrieben:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Text filtern	Verwendet Zeichen oder Platzhalter, die Sie zum Filtern der Liste mit probabilistischen Modellen im Repository eingegeben haben.
Probabilistisches Modell	Gibt das in der Operation zu verwendende probabilistische Modell an.

Referenztabelleneigenschaften

Referenztabelleneigenschaften werden angewendet, wenn Sie eine Beschriftungsoperation zur Verwendung einer Referenztabelle konfigurieren.

Die folgende Tabelle beschreibt die Referenztabelleneigenschaften:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Referenztabelle	Gibt die Referenztabelle an, die von der Operation zum Beschriften von Token verwendet wird.
Beschriftung	Gibt den Text an, der von der Operation auf einen neuen Port geschrieben wird, wenn ein Eingabestring mit einem Eintrag in der Referenztabelle übereinstimmt.
Groß-/ Kleinschreibung	Gibt an, ob Eingabestrings mit der Groß-/Kleinschreibung der Referenztabelleneinträge übereinstimmen müssen.

Eigenschaft	Beschreibung
Matches durch gültige Werte ersetzen	Ersetzt beschriftete Strings durch Einträge aus der Spalte "Gültig" in der Referenztabelle.
Modus	Gibt die Tokenbeschriftungsmethode an. Wählen Sie "Einschließlich" aus, um Eingabestrings zu beschriften, die mit Referenztabelleneinträgen übereinstimmen. Wählen Sie "Ausschließlich" aus, um Eingabestrings zu beschriften, die nicht mit Referenztabelleneinträgen übereinstimmen.
Priorität setzen	Gibt an, ob in einer Strategie Beschriftungsoperationen für Referenztabellen Vorrang vor Beschriftungsoperationen für Tokensätze haben sollen. Wenn Sie diese Eigenschaft festlegen, führt die Umwandlung die Referenztabellenbeschriftung vor der Tokensatzbeschriftung durch. Darüber hinaus kann die Tokensatzanalyse die Beschriftungsanalyse für Referenztabellen nicht überschreiben.

Konfigurieren einer Zeichenbeschriftungs-Strategie

Um eine Beschriftungsstrategie zu konfigurieren, bearbeiten Sie die Einstellungen in der Ansicht **Strategien** der Labeler-Umwandlung.

1. Wählen Sie die Ansicht **Strategien** aus und klicken Sie auf **Neu**, um eine Strategie zu erstellen.

   Der Assistent **Strategie** wird geöffnet.
2. Geben Sie einen Namen für die Strategie ein.
3. Klicken Sie auf die Felder **Eingaben** und **Ausgaben**, um Ports für die Strategie zu definieren.
4. Geben Sie optional eine Beschreibung der Strategie ein.
5. Wählen Sie den Zeichenbeschriftungs-Modus aus.
6. Klicken Sie auf **Weiter**.
7. Wählen Sie den Typ der zu konfigurierenden Zeichenbeschriftungs-Operation aus. Sie können folgende Operationen konfigurieren:
   - Zeichen mithilfe von Referenztabellen beschriften.
   - Zeichen mithilfe von Zeichensätzen beschriften.
8. Klicken Sie auf **Weiter**.
9. Konfigurieren Sie die Operationseigenschaften und klicken Sie auf **Weiter**.
10. Konfigurieren Sie optional Eigenschaften unter **Text ignorieren**.
11. Klicken Sie auf **Weiter**, um der Strategie weitere Operationen hinzuzufügen, oder klicken Sie auf **Fertig stellen**.

Sie können die Reihenfolge ändern, in der Strategien und Operationen von der Umwandlung verarbeitet werden. Wählen Sie in der Ansicht **Strategien** eine Strategie oder Operation aus und klicken Sie auf **Nach oben verschieben** oder **Nach unten verschieben**.

Konfigurieren einer Tokenbeschriftungsstrategie

Bearbeiten Sie zum Konfigurieren einer Beschriftungsstrategie die Einstellungen in der Ansicht **Strategien** der Labeler-Umwandlung.

1. Wählen Sie die Ansicht **Strategien** aus und klicken Sie auf **Neu**, um eine Strategie zu erstellen.

   Der Assistent **Strategie** wird geöffnet.
2. Geben Sie einen Namen für die Strategie ein.
3. Klicken Sie auf die Felder **Eingaben** und **Ausgaben**, um Ports für die Strategie festzulegen.
4. Geben Sie optional eine Beschreibung der Strategie ein.
5. Wählen Sie einen Tokenbeschriftungsmodus aus.

   Überprüfen oder bearbeiten Sie die Eigenschaften für den ausgewählten Modus.
6. Klicken Sie auf **Weiter**.
7. Wählen Sie den Typ der zu konfigurierenden Tokenbeschriftungsoperation aus. Sie können folgende Operationen konfigurieren:
   - Beschriften von Token mit einem Tokensatz.
   - Beschriften von Token mit einer Referenztabelle.
   - Beschriften von Token unter Verwendung von probabilistischem Matching.
8. Klicken Sie auf **Weiter**.
9. Konfigurieren Sie die Operationseigenschaften und klicken Sie auf **Weiter**.

   Geben Sie beim Konfigurieren der Strategie zur Verwendung von probabilistischem Matching eine Beschriftung ein, die für Token verwendet wird, die als Restdaten identifiziert wurden.
10. Konfigurieren Sie optional Eigenschaften unter **Benutzerdefinierte Beschriftung**.
11. Klicken Sie auf **Weiter**, um der Strategie weitere Operationen hinzuzufügen, oder klicken Sie auf **Fertig stellen**.

Sie können die Reihenfolge ändern, in der Strategien und Operationen von der Umwandlung verarbeitet werden. Wählen Sie in der Ansicht **Strategien** eine Strategie oder Operation aus und klicken Sie auf **Nach oben verschieben** oder **Nach unten verschieben**.

Labeler-Umwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Labeler-Umwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 23

Lookup-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Lookup-Umwandlung - Übersicht, 351
- Verbunden und nicht verbundene Lookups, 352
- Entwickeln einer Lookup-Umwandlung, 355
- Lookup-Abfrage, 355
- Lookup-Quellfilter, 358
- Lookup-Bedingung, 359
- Lookup-Cache, 362
- Abfrage-Eigenschaften, 362
- Lookup-Umwandlungen in dynamischen Mappings, 363
- Dynamische Ports definieren, 364
- Die Lookup-Quelle ändern, 364
- Portauswahlen, 368
- Laufzeiteigenschaften, 374
- Erweiterte Eigenschaften, 375
- Erstellen einer wiederverwendbaren Lookup-Umwandlung, 376
- Erstellen einer nicht wiederverwendbare Lookup-Umwandlung, 377
- Erstellen einer nicht verbundenen Lookup-Umwandlung, 378
- Nicht verbundenes Lookup - Beispiel, 379

Lookup-Umwandlung - Übersicht

Die Lookup-Umwandlung ist eine passive oder aktive Umwandlung, die in einer Einfachdatei, einem logischen Datenobjekt, einer Referenztabelle oder einer relationalen Tabelle Daten nachschlägt. Die Lookup-Umwandlung kann eine Zeile oder mehrere Zeilen aus einem Lookup zurückgeben.

Bevor Sie eine Lookup-Umwandlung erstellen, erstellen Sie zunächst die Lookup-Quelle. Importieren Sie eine Einfachdatei oder relationale Datenbanktabelle als physisches Datenobjekt. Oder erstellen Sie ein logisches Datenobjekt oder eine Referenztabelle, um diese als Lookup-Quelle zu verwenden. Beim Erstellen einer Lookup-Umwandlung fügt das Developer tool die Spalten aus dem Datenobjekt oder der Referenztabelle der Umwandlung als Lookup-Ports hinzu. Konfigurieren Sie nach dem Erstellen der Umwandlung einen oder

mehrere Ausgabeports, um die Lookup-Ergebnisse zurückzugeben. Konfigurieren Sie die Lookup-Bedingungen und weitere Lookup-Eigenschaften.

Wenn Sie ein Mapping ausführen oder Daten in der Vorschau anzeigen, fragt der Integrationsdienst die Lookup-Quelle ab. Der Integrationsdienst fragt die Lookup-Quelle basierend auf den Lookup-Ports in der Umwandlung, den Lookup-Eigenschaften und der Lookup-Bedingung ab. Die Lookup-Umwandlung gibt das Ergebnis des Lookups an das Ziel oder an eine andere Umwandlung zurück.

Sie können eine verbundene oder nicht verbundene Lookup-Umwandlung konfigurieren. Eine verbundene Umwandlung stellt eine Verbindung zu einer anderen Umwandlung im Mapping her. Eine nicht verbundene Umwandlung erhält Eingaben aus einem :LKP-Ausdruck in einer anderen Umwandlung. Wenn die Lookup-Umwandlung ein Lookup auf einem logischen Datenobjekt durchführt, müssen Sie eine verbundene Lookup-Umwandlung konfigurieren. Verbinden Sie die Eingabeports der Lookup-Umwandlung mit einer vorangehenden Umwandlung oder mit einer vorangehenden Quelle. Verbinden Sie die Ausgabeports mit einer nachgelagerten Umwandlung oder mit einem nachgelagerten Ziel.

Sie können mehrere Lookup-Umwandlungen in einem Mapping verwenden.

Sie können die folgenden Aufgaben mit einer Lookup-Umwandlung durchführen:

- Abrufen eines verwandten Wertes. Abrufen eines Werts aus der Lookup-Quelle, basierend auf einem Wert in den Eingabedaten. Die Eingabedaten enthalten beispielsweise eine Mitarbeiter-ID. Abrufen des Mitarbeiternamens aus der Lookup-Quelle nach Mitarbeiter-ID.
- Rufen Sie mehrere Zeilen aus einer Lookup-Quelle ab.
- Führen Sie eine Berechnung durch. Rufen Sie einen Wert aus einer Lookup-Tabelle ab und verwenden Sie ihn in einer Berechnung. Sie können beispielsweise den Steuerprozentsatz aus einem Verkauf abrufen, die Steuer berechnen und die Steuer an eine Zielspalte zurückgeben.
- Führen Sie ein nicht verbundenes Lookup mit einem :LKP-Ausdruck in einer Umwandlung durch, die Ausdrücke akzeptiert. Filtern Sie die Ergebnisse mit einem anderen Ausdruck in der Umwandlung.
- Parametrisieren Sie die Lookup-Quelle und die Lookup-Bedingung, um eine Lookup-Umwandlung in einem dynamischen Mapping verwenden zu können.

Verbunden und nicht verbundene Lookups

Sie können eine verbundene Lookup-Umwandlung oder eine nicht verbundene Lookup-Umwandlung konfigurieren. Eine verbundene Lookup-Umwandlung ist eine Umwandlung, die Eingabe- und Ausgabeports aufweist, die mit anderen Umwandlungen in einem Mapping verbinden. Eine nicht verbundene Lookup-Umwandlung erscheint im Mapping, wird aber nicht mit anderen Umwandlungen verbunden.

Eine nicht verbundene Lookup-Umwandlung erhält eine Eingabe aus dem Ergebnis eines :LKP-Ausdrucks in einer Umwandlung, zum Beispiel aus einer Ausdrucksumwandlung oder einer Aggregator-Umwandlung. Der :LKP-Ausdruck übergibt Argumente an die Lookup-Umwandlung und erhält ein Ergebnis aus der Lookup-Umwandlung zurück. Der :LKP-Ausdruck kann Lookup-Ergebnisse an einen anderen Ausdruck in der Ausdrucks- oder Aggregator-Umwandlung weitergeben, um Ergebnisse zu filtern.

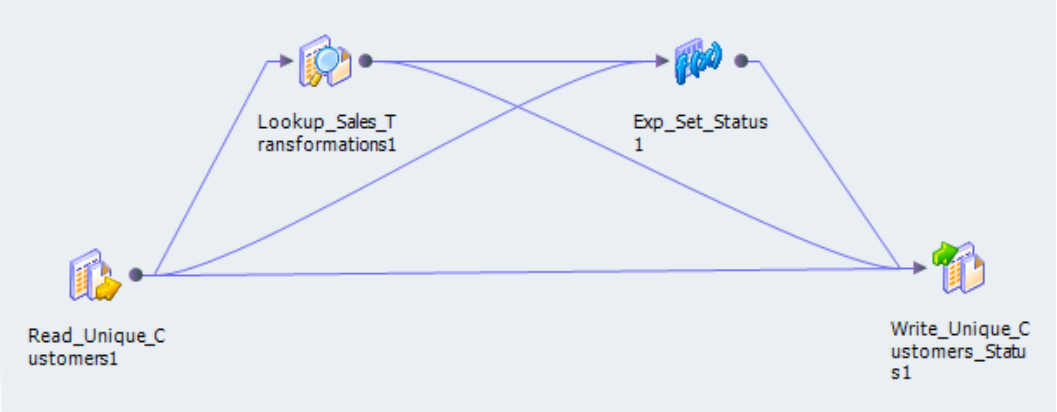
In der folgenden Tabelle sind die Unterschiede zwischen verbunden und nicht verbundene Lookups aufgelistet:

Verbundenes Lookup	Nicht verbundenes Lookup
Erhält Eingabewerte direkt aus der Pipeline.	Erhält Eingabewerte aus dem Ergebnis eines :LKP-Ausdrucks in einer anderen Umwandlung.
Verwenden Sie einen dynamischen oder statischen Cache.	Verwenden Sie einen statischen Cache.
Cache enthält die Lookup-Quellspalten in der Lookup-Bedingung und die Lookup-Quellspalten, die Ausgabeports sind.	Cache enthält die Lookup- und Ausgabeports in der Lookup-Bedingung und den Lookup- bzw. Rückgabeport.
Gibt mehrere Spalten aus derselben Zeile zurück oder fügt diese in den dynamischen Lookup-Cache ein.	Gibt eine Spalte aus jeder Zeile an einen Rückgabeport zurück.
Wenn die Lookup-Bedingung keinen Match enthält, gibt der Integrationsdienst den Standardwert für alle Ausgabeports zurück. Wenn Sie das dynamische Caching konfigurieren, fügt der Integrationsdienst Zeilen in den Cache ein oder nimmt keine Änderungen vor.	Wenn für die Lookup-Bedingung kein Match vorhanden ist, gibt der Integrationsdienst NULL zurück.
Wenn für die Lookup-Bedingung ein Match vorhanden ist, gibt der Integrationsdienst das Ergebnis der Lookup-Bedingung für alle Lookup- bzw. Ausgabeports zurück. Wenn Sie das dynamische Caching konfigurieren, aktualisiert der Integrationsdienst entweder die Zeile im Cache oder er nimmt keine Änderung an der Zeile vor.	Wenn für die Lookup-Bedingung ein Match vorhanden ist, gibt der Integrationsdienst das Ergebnis der Lookup-Bedingung an den Rückgabeport zurück.
Übergibt mehrere Ausgabewerte an eine andere Umwandlung. Verknüpfen Sie Lookup- bzw. Ausgabeports mit einer anderen Umwandlung.	Gibt einen Ausgabewert an eine andere Umwandlung zurück. Der Rückgabeport für die Lookup-Umwandlung gibt den Wert an den Port weiter, der den :LKP-Ausdruck in der anderen Umwandlung enthält.
Unterstützt benutzerdefinierte Standardwerte.	Unterstützt keine benutzerdefinierten Standardwerte.

Verbundene Lookups

Eine verbundene Lookup-Umwandlung ist eine Lookup-Umwandlung, die mit einer Quelle oder einem Ziel in einem Mapping verbunden ist.

In der folgenden Abbildung wird ein Mapping mit einer verbundenen Lookup-Umwandlung angezeigt:



Beim Ausführen eines Mappings mit einer verbundenen Lookup-Umwandlung führt der Integrationsdienst die folgenden Schritte aus:

1. Der Integrationsdienst übergibt Werte aus einer anderen Umwandlung an Eingabeports in der Lookup-Umwandlung.
2. Für jede Eingabezeile fragt der Integrationsdienst die Lookup-Quelle oder den Cache basierend auf den Lookup-Ports und Lookup-Bedingungen in der Umwandlung ab.
3. Wenn die Umwandlung nicht zwischengespeichert ist oder einen statischen Cache verwendet, gibt der Integrationsdienst Werte aus der Lookup-Abfrage zurück.

   Wenn die Umwandlung einen dynamischen Cache verwendet, fügt der Integrationsdienst die Zeile in den Cache ein, wenn er die Zeile im Cache nicht findet. Wenn der Integrationsdienst die Zeile im Cache findet, aktualisiert er sie im Cache oder lässt sie unverändert. Die Zeile wird wie folgt markiert: „Einfügen“, „Aktualisieren“ oder „Keine Änderung“.
4. Der Integrationsdienst gibt Daten aus der Abfrage zurück und übergibt sie an die nächste Umwandlung im Mapping.

   Wenn die Umwandlung einen dynamischen Cache verwendet, können Sie die Zeilen an eine Filter- oder Routerumwandlung übergeben, um neue Zeilen für das Ziel zu filtern.

Hinweis: Sofern nicht anders angegeben, werden in diesem Kapitel die verbundenen Lookup-Umwandlungen erörtert.

Nicht verbundene Lookups

Bei einer nicht verbundenen Lookup-Umwandlung handelt es sich um eine Lookup-Umwandlung, die nicht mit einer Quelle oder einem Ziel im Mapping verbunden ist. Rufen Sie das Lookup mit einem :LKP-Ausdruck in einer Umwandlung auf, die Ausdrücke zulässt.

Die Syntax für den Lookup-Ausdruck lautet `:LKP lookup_transformation_name(argument, argument, ...)`

Die Reihenfolge, in der Sie die einzelnen Argumente auflisten, muss mit der Reihenfolge der Lookup-Bedingungen in der Lookup-Umwandlung übereinstimmen. Die Lookup-Umwandlung gibt das Ergebnis der Abfrage über den Rückgabeport der Lookup-Umwandlung zurück. Die Umwandlung, die das Lookup aufruft, empfängt den Lookup-Ergebniswert an dem Port, der den :LKP-Ausdruck enthält. Wenn die Lookup-Abfrage keinen Wert zurückgibt, empfängt der Port einen Nullwert.

Wenn Sie ein nicht verbundenes Lookup durchführen, können Sie das gleiche Lookup mehrere Male in einem Mapping durchführen. Sie können die Ergebnisse des Lookups in einem anderen Ausdruck testen und Zeilen basierend auf den Ergebnissen filtern.

Beim Ausführen eines Mappings, das eine nicht verbundene Lookup-Umwandlung enthält, führt der Integrationsdienst die folgenden Schritte aus:

1. Eine nicht verbundene Lookup-Umwandlung erhält eine Eingabe aus dem Ergebnis eines :LKP-Ausdrucks in einer anderen Umwandlung, zum Beispiel aus einer Aggregator-Umwandlung oder einer Update-Strategie-Umwandlung.
2. Die Integrationsdienstabfragen erfolgen in der Lookup-Quelle oder dem Cache basierend auf den Lookup-Ports und der Bedingung in der Lookup-Umwandlung.
3. Der Integrationsdienst gibt einen Wert über den Rückgabeport der Lookup-Umwandlung zurück.
4. Der Integrationsdienst übergibt den Rückgabewert an den Port, der den :LKP-Ausdruck enthält.

Entwickeln einer Lookup-Umwandlung

Wenn Sie eine Lookup-Umwandlung entwickeln, müssen Sie einige Faktoren wie den Lookup-Quelltyp und die Lookup-Bedingung beachten.

Berücksichtigen Sie die folgenden Faktoren, wenn Sie eine Lookup-Umwandlung entwickeln:

- Ob Sie die Umwandlung aus einer Einfachdatei, einem logischen Datenobjekt, einer Referenztabelle oder einem relationalen Datenobjekt erstellen möchten. Bevor Sie eine Lookup-Umwandlung erstellen, erstellen Sie zunächst die Lookup-Quelle. Importieren Sie eine Einfachdatei oder relationale Datenbanktabelle als physisches Datenobjekt. Oder erstellen Sie ein logisches Datenobjekt oder eine Referenztabelle, um diese als Lookup-Quelle zu verwenden.
- Die Ausgabeports für die Umwandlung.
- Die Lookup-Bedingungen in der Umwandlung.
- Ob der Integrationsdienst die Lookup-Daten in einem Cache speichern soll. Der Integrationsdienst kann die Daten für Einfachdateien, Referenztabellen oder relationale Datenobjekte in einem Cache speichern.

Lookup-Abfrage

Der Integrationsdienst fragt die Lookup-Quelle basierend auf den Ports und Eigenschaften ab, die Sie in der Lookup-Umwandlung konfigurieren. Der Integrationsdienst führt eine Standard-Lookup-Abfrage aus, wenn die erste Zeile die Lookup-Umwandlung durchläuft.

Wenn Sie ein Lookup in einer relationalen Tabelle verwenden, können Sie die Lookup-Abfrage überschreiben. Sie können die Überschreibung verwenden, um die ORDER BY-Klausel zu ändern, eine WHERE-Klausel hinzuzufügen oder die Lookup-Daten umwandeln, bevor sie zwischengespeichert werden.

Wenn Sie eine SQL-Überschreibung und einen Filter für die Lookup-Abfrage konfigurieren, wird der Filter vom Integrationsdienst ignoriert.

Standard-Lookup-Abfrage

Die Standard-Lookup-Abfrage enthält die folgenden Anweisungen:

SELECT

Die SELECT-Anweisung enthält alle Lookup-Ports im Mapping. Um die SELECT-Anweisung für die Lookup-Abfrage anzuzeigen, wählen Sie die Eigenschaft Benutzerdefinierte Abfrage verwenden aus.

ORDER BY

Die ORDER BY-Klausel ordnet die Spalten in derselben Reihenfolge an, wie sie in der Lookup-Umwandlung angezeigt werden. Der Integrationsdienst generiert die ORDER BY-Klausel. Beim Generieren der Standard-SQL wird dies nicht angezeigt.

SQL-Überschreibung für eine Lookup-Abfrage

Sie können die Lookup-Abfrage für ein relationale Lookups überschreiben. Sie können die ORDER BY-Klausel ändern, eine WHERE-Klausel hinzufügen und die Lookup-Daten umwandeln, bevor sie zwischengespeichert werden.

Sie können reservierte Wörter und Schrägstriche in Tabellennamen und Spaltennamen verwenden.

Sie können eine Abfrage eingeben, um die Standard-Lookup-Abfrage vollständig zu überschreiben. Wahlweise können Sie die Standard-Lookup-Abfrage anzeigen und bearbeiten. Die Standard-Lookup-Abfrage beinhaltet die Lookup-Ports, Ausgabeports und die Rückgabeports.

SQL-Überschreibung für nicht zwischengespeicherte Lookups

Sie können für nicht zwischengespeicherte Lookups SQL-Überschreibungen definieren. Sie können SQL-Abfragen überschreiben, einschließlich der WHERE-Klausel und der ORDER BY-Klausel.

Das Developer-Tool generiert eine SELECT-Anweisung, die Lookup-Ports, Eingabe-/Ausgabeports und die WHERE-Klausel basierend auf der Lookup-Bedingung enthält. Wenn die Lookup-Umwandlung ein nicht verbundenes Lookup ist, enthält die SELECT-Anweisung die Lookup-Ports und den Rückgabeport.

Jede Spalte in der SELECT-Abfrage verwendet ein Alias zum Definieren der Ausgabespalte. Ändern Sie nicht die Syntax der Spaltenaliase in der SQL-Anweisung. Wenn Sie die Syntax ändern, schlägt die Abfrage fehl. Um Eingabeports in der WHERE-Klausel zu referenzieren, konfigurieren Sie Parameterbindung. Das folgende Beispiel enthält eine WHERE-Anweisung, die den Port „Name“ referenziert:

```
SELECT EMPLOYEE.NAME as NAME, max(EMPLOYEE.ID) as ID from EMPLOYEE WHERE EMPLOYEE.NAME=?
NAME1?
```

Sie können SQL-Funktionen in der SQL-Überschreibung verwenden. Wenn Sie der SQL-Anweisung eine Funktion hinzufügen, muss der Rückgabedatentyp mit dem Datentyp der Spalte übereinstimmen. Beispiel: Der Datentyp der ID stimmt mit dem Rückgabetyp der Funktion MAX überein:

```
SELECT EMPLOYEE.NAME as NAME, MAX(EMPLOYEE.ID) as ID FROM EMPLOYEE
```

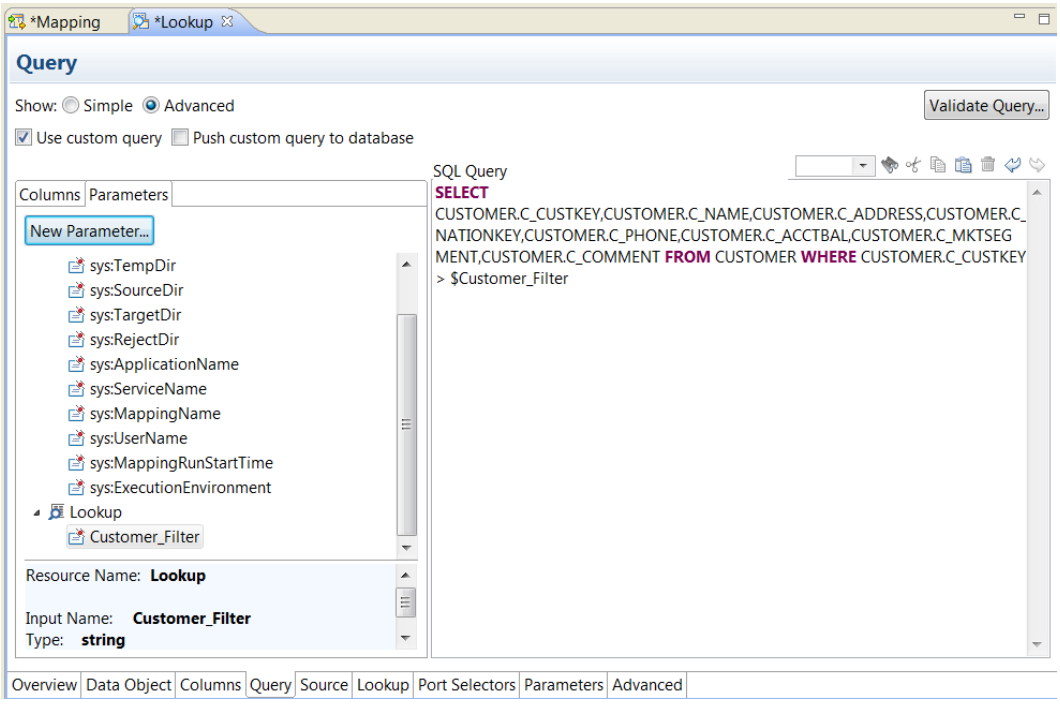
Hinweis: Sie können keine Unterabfragen in der SQL-Überschreibung für nicht zwischengespeicherte Lookups verwenden.

Parameter in einer SQL-Überschreibungsabfrage

Sie können Systemparameter oder benutzerdefinierte Parameter in der Lookup-Abfrage einer Lookup-Umwandlung verwenden. Der SQL-Editor stellt eine Liste mit den Systemparametern und benutzerdefinierten Parametern bereit, aus denen Sie auswählen können.

Auf der Registerkarte **Abfrage** einer Lookup-Umwandlung können Sie nach benutzerdefinierten Parametern suchen oder diese erstellen. Definieren Sie einen Standardwert für jeden Parameter. Sie können einen Standardwert überschreiben, indem Sie ein Mapplet oder einen Mapping-Parameter an den Umwandlungsparameter binden, nachdem Sie die Lookup-Umwandlung zu einem Mapping hinzugefügt haben.

Die folgende Abbildung zeigt die Registerkarte „Abfrage“ der Lookup-Umwandlung:



Reservierte Wörter

Wenn ein Lookup- oder ein Spaltenname ein für eine Datenbank reserviertes Wort enthält, zum Beispiel MONTH oder YEAR, schlägt das Mapping mit Datenbankfehlern fehl, wenn der Integrationsdienst SQL in der Datenbank ausführt.

Sie können die Datei mit reservierten Wörtern (reswords.txt) im Installationsverzeichnis des Integrationsdiensts erstellen und verwalten. Wenn der Integrationsdienst ein Mapping initialisiert, sucht er die reswords.txt-Datei, platziert reservierte Wörter in Anführungszeichen und führt dann SQL in Quell-, Ziel- und Lookup-Datenbanken aus.

Möglicherweise müssen Sie einige Datenbanken aktivieren, zum Beispiel Microsoft SQL Server und Sybase, um SQL-92 Standards bezüglich von in Anführungszeichen gesetzten Bezeichnern zu verwenden. Verwenden Sie Umgebungs-SQL, um den Befehl auszuführen. Verwenden Sie beispielsweise mit Microsoft SQL Server den folgenden Befehl:

```
SET QUOTED_IDENTIFIER ON
```

Richtlinien für das Überschreiben der Lookup-Abfrage

Bestimmte Regeln und Richtlinien werden beim Überschreiben einer Lookup-Abfrage angewendet.

Beachten Sie die folgenden Richtlinien, wenn Sie die Lookup-SQL-Abfrage überschreiben:

- Sie können die SQL-Lookup-Abfrage überschreiben für relationale Lookups überschreiben.
- Fügen Sie einen Quell-Lookup-Filter hinzu, um die zum Lookup-Cache hinzugefügten Zeilen zu filtern. Dies stellt sicher, dass der Integrationsdienst die Zeilen im dynamischen Cache und in der Zieltabelle einfügt, die mit der WHERE-Klausel übereinstimmen.
- Wenn mehrere Lookup-Umwandlungen einen Lookup-Cache gemeinsam verwenden, verwenden Sie dieselbe Lookup-SQL-Überschreibung für jede Lookup-Umwandlung.

- Wenn ein Tabellen- oder Spaltennamen in der Lookup-Abfrage ein reserviertes Wort enthält, setzen Sie das reservierte Wort in Anführungszeichen.
- Um eine Lookup-Abfrage für ein nicht zwischengespeichertes Lookup zu überschreiben, wählen Sie die Rückgabe eines beliebigen Werts aus, wenn der Integrationsdienst mehrere Übereinstimmungen findet.
- Sie können keine Spalten zur Standard-SQL-Anweisung hinzufügen oder aus dieser löschen.

Überschreiben der Lookup-Abfrage

Sie können die Standard-Lookup-SQL-Abfrage zum Erstellen einer benutzerdefinierten Abfrage in der Lookup-Quelle überschreiben.

1. Wählen Sie in der Ansicht **Eigenschaftten** die Registerkarte **Abfrage** aus.
2. Wählen Sie **Erweitert** aus.
3. Wählen Sie **Benutzerdefinierte Abfrage verwenden** aus.
4. Bearbeiten Sie die Lookup-Abfrage im Bereich „SQL-Abfrage“.

   Sie können auf einen Tabellen- oder Spaltennamen oder auf einen Parameter doppelklicken, um ihn zu der Abfrage hinzuzufügen.
5. Klicken Sie auf **Abfrage validieren**, um die Lookup-Abfrage zu validieren.
6. Wählen Sie **Benutzerdefinierte Abfrage in Datenbank verlagern**, um die Lookup-Abfrage in der Datenbank auszuführen.

Lookup-Quellfilter

Sie können einen Lookup-Quellfilter für eine relationale Lookup-Umwandlung mit aktiviertem Caching konfigurieren. Fügen Sie den Lookup-Quellfilter hinzu, um die Anzahl der Lookups zu begrenzen, die der Integrationsdienst für eine Lookup-Quelltabelle ausführt.

Wenn Sie einen Lookup-Quellfilter konfigurieren, führt der Integrationsdienst Lookups basierend auf den Ergebnissen der Filteranweisung durch. Sie müssen zum Beispiel den Nachnamen jedes Mitarbeiters abrufen, dess ID größer als 510 ist.

Sie konfigurieren den folgenden Lookup-Quellfilter in der EmployeeID-Spalte:

```
EmployeeID >= 510
```

EmployeeID ist ein Eingabeport in der Lookup-Umwandlung. Wenn der Integrationsdienst die Quellzeile liest, führt er Lookup für den Cache durch, wenn der Wert von EmployeeID größer als 510 ist. Wenn EmployeeID kleiner als oder gleich 510 ist, ruft die Lookup-Umwandlung den Nachnamen nicht ab.

Wenn Sie einen Lookup-Quellfilter zur Lookup-Abfrage für ein Mapping hinzufügen, das für Pushdown-Optimierung konfiguriert ist, erstellt der Integrationsdienst eine Ansicht, um die SQL-Überschreibung darzustellen. Der Integrationsdienst führt eine SQL-Abfrage für diese Ansicht aus, um die Umwandlungslogik in die Datenbank zu verschieben.

Filtern von Quellzeilen in einem Lookup

Sie können einen Lookup-Quellfilter für eine relationale Lookup-Umwandlung mit aktiviertem Caching konfigurieren. Filtern Sie Quellzeilen in einem Lookup, um die Anzahl der Lookups zu begrenzen, die der Integrationsdienst für eine Lookup-Quelltabelle ausführt.

1. Wählen Sie in der Ansicht **Eigenschaften** die Registerkarte **Abfrage** aus.
2. Klicken Sie in der Option **Filter** auf **Bearbeiten**.
3. Wählen Sie im SQL-Editor die Eingabeports aus oder geben Sie einen Lookup-Umwandlungsport ein, den Sie filtern möchten.
4. Geben Sie eine Filterbedingung ein.

   Geben Sie in der Filterbedingung nicht das Schlüsselwort WHERE ein. Schließen Sie die Zeichenfolgen-Mapping-Parameter und -Variablen in Zeichenfolgen-Bezeichnern ein.
5. Klicken Sie auf **Abfrage validieren**, um die Syntax der Filterbedingung zu validieren.

Lookup-Bedingung

Der Datenintegrationsdienst sucht nach Daten in der Lookup-Quelle basierend auf einer Lookup-Bedingung. Wenn Sie eine Lookup-Bedingung in einer Lookup-Umwandlung konfigurieren, vergleichen Sie den Wert in einer oder mehreren Spalten in den Quelldaten mit Werten in der Lookup-Quelle oder dem Cache.

Die Quelldaten enthalten beispielsweise eine Mitarbeiternummer (employee_number). Die Lookup-Quelltabelle enthält Mitarbeiter-ID (employee_ID), Vorname (first_name) und Nachname (last_name). Sie konfigurieren die folgende Lookup-Bedingung:

```
employee_ID = employee_number
```

Für jede Mitarbeiternummer (employee_number) gibt der Datenintegrationsdienst die Spalten Mitarbeiter-ID (employee_ID), den Nachnamen (last_name) und den Vornamen (first_name) aus der Lookup-Quelle zurück.

Der Datenintegrationsdienst kann mehr als eine Zeile aus der Lookup-Quelle zurückgeben. Sie konfigurieren die folgende Lookup-Bedingung:

```
employee_ID > employee_number
```

Der Datenintegrationsdienst gibt Zeilen für alle Nummern der Mitarbeiter-ID (employee_ID) zurück, die größer als die Quell-Mitarbeiternummer sind.

Nullwerte in einem Datenobjekt-Lookup

Wenn eine Eingabe in der Lookup-Bedingung NULL ist, gibt eine Umwandlung für ein Datenobjekt-Lookup eine einzelne Zeile mit Nullwerten für Nur-Ausgabe-Ports zurück und Werte aus Eingabezeilen für Pass-Through-Ports.

Beispiel: Die folgende Lookup-Bedingung führt ein Lookup auf einer Datenquelle durch, die eine oder mehrere Zeilen enthält, deren Wert für employee_ID NULL ist:

```
employee_ID = employee_number
```

In diesem Beispiel verwenden Sie eine Lookup-Tabelle mit den folgenden Daten:

EMPLOYEE_ID	LAST_NAME
1294765	Hara

EMPLOYEE_ID	LAST_NAME
1356356	Carver
1407207	NULL
1570348	Draper
NULL	Limonov

Sie vergleichen die folgenden Eingabewerte aus Ihrer Datenquelle mit der Lookup-Tabelle:

```
EMPLOYEE_NUMBER
---------------
1294765
1356356
1407207
1648246
NULL
```

In diesem Beispiel kommt die Lookup-Bedingung zu folgenden Ergebnissen:

```
1294765,Hara
1356356,Carver
1407207,NULL
NULL,NULL
NULL,NULL
```

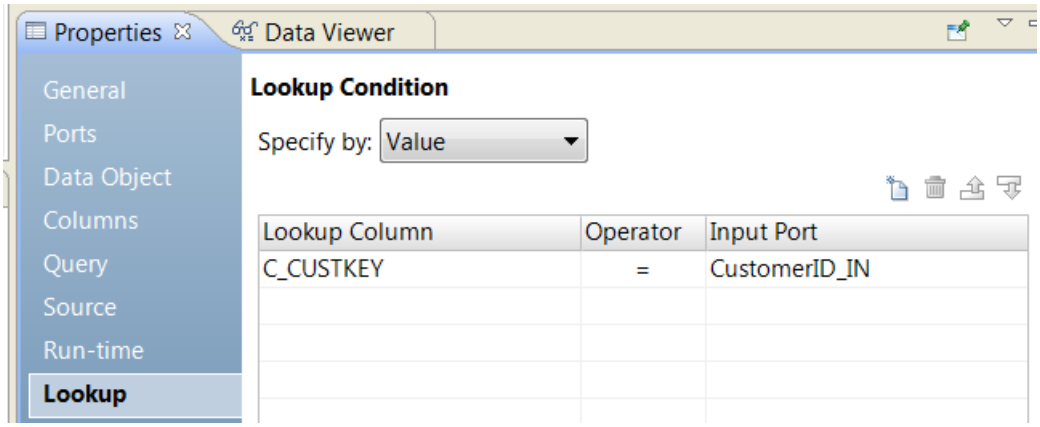
Die Lookup-Bedingung findet Matches zwischen EMPLOYEE_ID und EMPLOYEE_NUMBER für die ersten zwei Zeilen. Für die dritte Zeile enthält die Lookup-Quelle eine Zeile mit einem Wert NULL, der nicht an der Lookup-Bedingung teilnimmt. Er stimmt mit der Lookup-Bedingung überein und gibt ein Ergebnis mit dem Wert NULL für die Nicht-Lookup-Spalte zurück.

Für die vierte und fünfte Zeile findet die Lookup-Bedingung kein Match und gibt NULL für beide Werte zurück. Beachten Sie bei der fünften Zeile, dass die Lookup-Bedingung kein Match mehr findet, da NULL über keine Übereinstimmung verfügt, auch nicht mit NULL.

Die Lookup-Bedingung konfigurieren

Die Lookup-Bedingung ist ein Ausdruck, der bestimmt, welche Zeilen aus der Lookup-Quelle abgerufen werden sollen. Konfigurieren Sie die Lookup-Bedingung auf der Registerkarte **Lookup** der Ansicht **Eigenschaften**.

Die folgende Abbildung zeigt eine Lookup-Bedingung zur Durchführung eines Lookup nach Kundennummer:



Sie können die folgenden Optionen auf der Registerkarte **Lookup** konfigurieren:

Angeben nach

Wählen Sie **Wert**, um die Lookup-Spalte und die Namen der Eingabeports auszuwählen. Wählen Sie **Parameter** aus, um einen Ausdrucksparameter zum Definieren der Lookup-Bedingung zu konfigurieren.

Lookup-Spalte

Die Spalte in der Lookup-Quelle, die mit einer Spalte aus der Eingabezeile übereinstimmen muss. Sie können mehrere Spalten in die Lookup-Bedingung einschließen.

Operator

Der Operator, der die Bedingung bestimmt, nach der zwischen der Lookup-Spalte und dem Eingabeport gesucht werden soll. Operatoren beinhalten =, !=, >, < , >=, <=.

Eingabeport

Der Eingabeport, der den Wert enthält, nach dem in der Lookup-Quelle gesucht werden soll. Sie können mehr als einen Eingabeport mit den Ports in der Lookup-Quelle vergleichen.

Regeln und Richtlinien für Lookup-Umwandlungsbedingungen

Bestimmte Regeln und Richtlinien werden anwenden, wenn Sie eine Bedingung in eine Lookup-Umwandlung eingeben.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie eine Bedingung in eine Lookup-Umwandlung eingeben.

- Die Datentypen für Spalten in einer Lookup-Bedingung muss übereinstimmen.
- Verwenden Sie einen Eingabeport für jeden Lookup-Port in der Lookup-Bedingung. Sie können denselben Eingabeport in mehr als einer Bedingung in einer Umwandlung verwenden.
- Wenn Sie eine Portauswahl oder einen dynamischen Port in einer Lookup-Bedingung verwenden, berücksichtigt die Lookup-Bedingung alle Ports in dem Ausdruck.
- Sie können einen dynamischen Eingabeport oder eine Portauswahl als Eingabeport einer Lookup-Bedingung verwenden. Die Anzahl der generierten Ports im Eingabeport muss der Anzahl der Ports in der Lookup-Spalte entsprechen.
- Bei der Verarbeitung einer Lookup-Umwandlung mit mehreren Lookup-Bedingungen gibt der Integrationsdienst gibt Zeilen zurück, die allen Lookup-Bedingungen entsprechen.
- Sie können einen Ausdrucksparameter erstellen, um die Lookup-Bedingung in einer nicht wiederverwendbaren Lookup-Umwandlung zu parametrisieren.
- Um die Lookup-Leistung zu erhöhen, geben Sie Bedingungen in der folgenden Reihenfolge ein:
  - Gleich (=)
  - Kleiner als (<), größer als (>), kleiner oder gleich (< =), größer oder gleich (> =)
  - Nicht gleich (!=)
- Verwenden Sie einen der folgenden Operatoren beim Erstellen einer Lookup-Bedingung: =, >, <, >=, <= oder !=.
- Der Integrationsdienst verarbeitet die Matches basierend darauf, ob Sie die Lookup-Umwandlung mit einem dynamischen Lookup-Cache, statischen Lookup-Cache oder einem nicht zwischengespeicherten Lookup durchführen.
- Der Integrationsdienst gleicht Nullwerte ab. Der Integrationsdienst betrachtet beispielsweise einen Lookup-Port und einen Eingabeport gleich, wenn beide Nullwerte beinhalten.

- Wenn es sich bei den Spalten in einer Lookup-Bedingung um Datentypen „Dezimal“ handelt, muss die Gesamtstellenanzahl jeder Spalte zu demselben Präzisionsbereich gehören. Gültige Präzisionsbereiche:

  - Dezimal 0-18
  - Dezimal 19-28
  - Dezimal 29-38
  - Dezimal 39 und höher

  Wenn Sie beispielsweise die Bedingung `DecimalA = DecimalB` definieren, in der DecimalA die Präzision 15 und DecimalB die Gesamtstellenanzahl 25 aufweist, ist die Lookup-Bedingung nicht gültig.

Lookup-Cache

Sie können die Leistung erhöhen, indem Sie eine große Lookup-Quelle oder kleine Lookup-Tabellen zwischenspeichern. Wenn Sie eine Lookup-Quelle zwischenspeichern, fragt der Integrationsdienst den Lookup-Cache ab, anstatt die Lookup-Quelle für jede Eingabezeile abzufragen.

Basierend auf Ihren geschäftlichen Anforderungen können Sie verschiedene Arten von Lookup-Caches erstellen. Sie können einen statischen oder dynamischen Cache erstellen. Sie können einen persistenten oder nicht persistenten Cache erstellen. Sie können einen Cache für mehrere Lookup-Umwandlungen gemeinsam verwenden.

Wenn sich die Lookup-Umwandlung in einem dynamischen Mapping befindet, können Sie einen persistenten oder einen nicht persistenten Cache verwenden. Wenn Sie einen Cache beibehalten und die Lookup-Quelle mit einem Parameter ändern, schlägt das Mapping fehl. Das Mapping schlägt auch dann fehl, wenn Sie die Steuerungsdatei in eine Einfachdatei-Lookup-Quelle ändern.

Hinweis: Sie können keinen dynamischen Lookup-Cache oder persistenten Lookup-Cache verwenden, wenn die Lookup-Umwandlung einen dynamischen Port oder eine parametrisierte Lookup-Quelle enthält.

Abfrage-Eigenschaften

Konfigurieren Sie die Abfrage-Eigenschaften, um die Lookup-Abfrage in einer relationalen Lookup-Tabelle anzuzeigen oder zu ändern. Sie können einen Filter auf das Lookup anwenden oder die Lookup-Abfrage anpassen.

Die folgende Tabelle beschreibt die Abfrage-Eigenschaften für Lookup-Umwandlungen, die relationale Lookups durchführen:

Eigenschaft	Beschreibung
Einfach	Wählen Sie dies aus, um die Lookup-Abfrage anzuzeigen und einen Filter auf das Lookup anzuwenden.
Erweitert	Wählen Sie dies aus, um die Abfrage anzuzeigen, die Abfrage anzupassen oder die Abfrage in der Datenbank auszuführen, die die relationale Lookup-Tabelle enthält.
Filter	Geben Sie einen Filter ein, um die Anzahl der Zeilen, die der Integrationsdienst abfragt, zu reduzieren. Sie müssen die Option **Einfach** auswählen, um diese Option anzuzeigen.

Eigenschaft	Beschreibung
Benutzerdefinierte Abfrage verwenden	Wählen Sie dies aus, um die Lookup-Abfrage zu überschreiben. Sie müssen die Option **Einfach** auswählen, um diese Option anzuzeigen.
Benutzerdefinierte Abfrage in Datenbank verlagern	Wählen Sie dies aus, um die Abfrage in der Datenbank auszuführen, die die relationale Lookup-Tabelle enthält. Sie müssen die Option **Einfach** auswählen, um diese Option anzuzeigen.
SQL-Abfrage	Zeigt die SQL-Abfrage an, die im Lookup durchgeführt wird. Sie können die SQL-Abfrage anpassen. Sie müssen die Option **Einfach** auswählen, um diese Option anzuzeigen.

Lookup-Umwandlungen in dynamischen Mappings

Sie können eine Lookup-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports so konfigurieren, dass sie basierend auf den Quelldaten unterschiedliche Ports empfangen und zurückgeben. Sie können die Lookup-Quelle und die Lookup-Bedingung so parametrisieren, dass ein Lookup basierend auf den unterschiedlichen Ports durchgeführt wird.

Ein dynamisches Mapping ist ein Mapping, in dem sich Quellen, Ziele und Umwandlungslogik zur Laufzeit ändern können. Sie können Parameter und Regeln zum Ändern der Datenstruktur festlegen. Wenn Sie eine Lookup-Umwandlung in einem dynamischen Mapping verwenden, ändern sich möglicherweise die Eingabeports der Lookup-Umwandlung basierend auf den Quelldaten. Die Struktur der Lookup-Quelle und die Ports in der Lookup-Bedingung ändern sich möglicherweise.

Hinweis: Wenn die Lookup-Umwandlung einen dynamischen Port oder eine parametrisierte Lookup-Quelle enthält, können Sie den Lookup-Cache nicht beibehalten. Sie können auch keinen dynamischen Cache konfigurieren.

Sie können die folgenden Aufgaben für eine Lookup-Umwandlung durchführen, um die Umwandlung in einem dynamischen Mapping zu verwenden:

Dynamische Ports definieren

Definieren Sie dynamische Ports und generierte Ports, um Änderungen an den Eingabespalten anzupassen.

Die Lookup-Quelle parametrisieren

Weisen Sie dem Datenobjekt, das die Lookup-Quelle definiert, einen Parameter zu. Sie können die Lookup-Quelle in einer nicht wiederverwendbaren Lookup-Umwandlung parametrisieren.

Portauswahlen definieren

Definieren Sie eine Portauswahl, die die in der Lookup-Bedingung zu verwendenden Ports angibt. Sie können die Portauswahlports in einer nicht wiederverwendbaren Lookup-Umwandlung parametrisieren.

Die Lookup-Bedingung parametrisieren

Erstellen Sie einen Ausdrucksparameter und definieren Sie einen Standardwert, der einen vollständigen Ausdruck enthält.

Weitere Informationen zu dynamischen Mappings finden Sie im *Informatica Developer-Handbuch für Mappings*.

Dynamische Ports definieren

Sie können dynamische Ports in der Lookup-Umwandlung definieren.

Sie können auf einen dynamischen Port in der Eingabespalte einer Lookup-Bedingung verweisen. Wenn der dynamische Port mehrere generierte Ports enthält, können Sie eine Portauswahl für das Lookup-Spaltenelement der Lookup-Bedingung verwenden. Der dynamische Eingabeport muss dieselbe Anzahl von Ports wie die Portauswahl in der Lookup-Bedingung enthalten.

Wenn der dynamische Port einen Wert enthält, können Sie einen einzelnen Port im Lookup-Spaltenelement der Lookup-Bedingung verwenden.

Sie können auf generierte Ports in der Lookup-Bedingung verweisen. Wenn sich jedoch eine Quelle im dynamischen Mapping ändert, ist der generierte Port möglicherweise nicht vorhanden. Das Mapping schlägt fehl.

Die Lookup-Quelle ändern

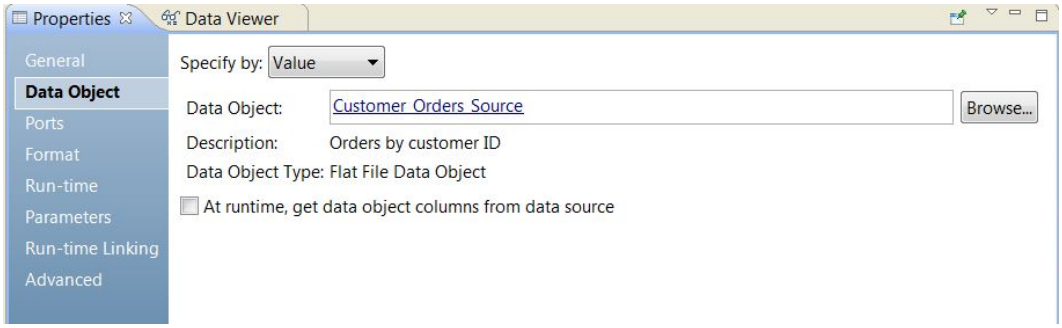
Sie können das Datenobjekt ändern, das die Lookup-Quelle für eine wiederverwendbare Lookup-Umwandlung darstellt. Konfigurieren Sie einen Parameter, der das als Lookup-Quelle zur Laufzeit zu verwendende Datenobjekt bestimmt.

Beim Erstellen einer Umwandlung aus einem physischen Datenobjekt werden Informationen zum Datenobjekt auf der Registerkarte **Datenobjekt** der Umwandlungseigenschaften angezeigt. Sie können auf den Namen des Datenobjekts klicken, um die Definition des physischen Datenobjekts aus dem Modellrepository anzuzeigen.

Sie können das Datenobjekt für die Umwandlung ändern, indem Sie nach einem anderen physischen Datenobjekt im Modellrepository suchen. Wenn Sie das Datenobjekt ändern, verwendet die Umwandlung die Laufzeiteigenschaften und die erweiterten Eigenschaften des von Ihnen ausgewählten Datenobjekts.

Sie können die Struktur des Datenobjekts zur Laufzeit basierend auf Änderungen in der Datenquelle aktualisieren. Die Datenquelle ist die physische Datei oder die Datenbanktabelle, die das Datenobjekt darstellt. Beim Aktivieren des Datenintegrationsdiensts zum Abrufen der Datenspalten aus der Datenquelle untersucht der Datenintegrationsdienst die Struktur der Datenquelle. Der Datenintegrationsdienst aktualisiert die Datenobjektports in der Umwandlungsinstanz basierend auf der Datenquelle. Der Datenintegrationsdienst ändert nicht die Definition des physischen Datenobjekts im Modellrepository.

Die folgende Abbildung zeigt die Registerkarte **Datenobjekt**:



Die Registerkarte **Datenobjekt** verfügt über die folgenden Felder:

Angeben nach

Wählen Sie **Wert**, um einen bestimmten Datenobjektnamen einzugeben. Wählen Sie **Parameter**, um das Datenobjekt zu parametrisieren.

Datenobjekt

Der Name des Datenobjekts im Modellrepository. Sie können auf den Link **Datenobjekt** klicken, um die Definition des Datenobjekts aus dem Repository zu öffnen. Sie können auch nach einem anderen Datenobjekt im Modellrepository suchen.

Beschreibung

Die Beschreibung des Datenobjekts im Repository. Schreibgeschützt.

Typ des Datenobjekts

Beschreibt den Typ des Datenobjekts, wie z. B. ein Einfachdatei-Datenobjekt, ein relationales Tabellenobjekt oder ein benutzerdefiniertes Datenobjekt.

Datenobjektspalten zur Laufzeit aus Datenquelle abrufen

Aktualisiert die Struktur des Datenobjekts für die Umwandlungsinstanz zur Laufzeit. Aktualisiert die Struktur basierend auf den Spalten aus der Datendatei oder der Tabelle, auf die das Datenobjekt verweist.

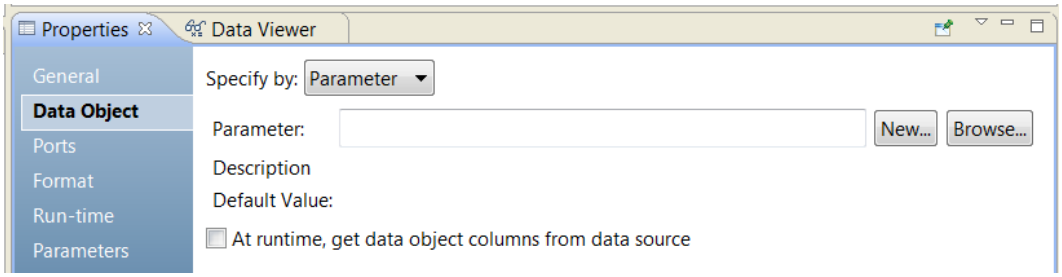
Die Lookup-Quelle parametrisieren

Sie können einen Parameter für die Lookup-Quelle in einer nicht wiederverwendbaren Lookup-Umwandlung konfigurieren.

Wählen Sie zum Parametrisieren eines Datenobjekts die Option **Angeben nach Parameter** auf der Registerkarte **Datenobjekt** aus. Die Eigenschaften auf der Registerkarte **Datenobjekt** ändern sich.

Um das Datenobjekt zu parametrisieren, erstellen Sie einen Ressourcentypparameter oder suchen Sie nach einem Ressourcenparameter, den Sie bereits erstellt haben. Der Standardwert des Parameters ist der Name des physischen Datenobjekts im Modellrepository. Wenn Sie einen Standardparameterwert erstellen, wählen Sie den Namen eines physischen Datenobjekts aus einer Liste mit Datenobjekten im Repository aus.

Die folgende Abbildung zeigt die Registerkarte **Datenobjekt**, wenn Sie das Datenobjekt nach einem Parameter angeben:



Die Registerkarte **Datenobjekt** verfügt über die folgenden Optionen nach Parameter:

Parameter

Der Name eines Ressourcenparameters, den Sie als Datenobjekt konfiguriert haben. Schreibgeschützt.

Beschreibung

Die Beschreibung des Parameters. Schreibgeschützt.

Neu

Erstellen Sie einen Ressourcenparameter. Suchen Sie im Modellrepository nach einem Datenobjekt und wählen Sie dieses als Standardwert des Parameters aus.

Durchsuchen

Suchen Sie nach einem Ressourcenparameter und wählen Sie den Parameter aus.

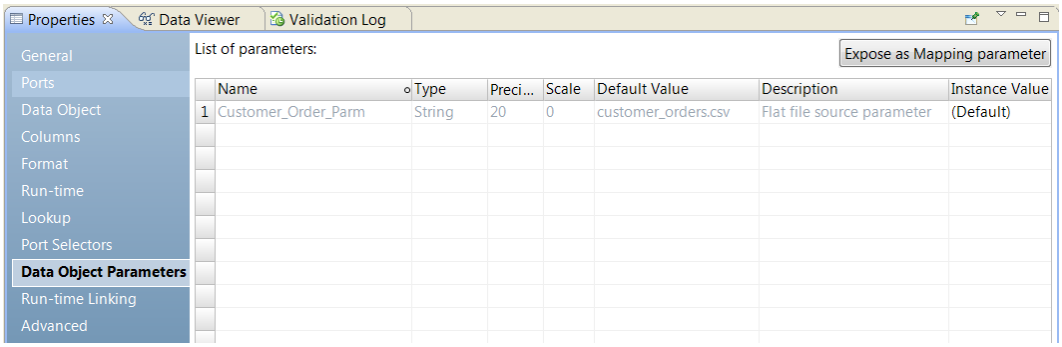
Standardwert

Der Standardwert des Ressourcenparameters, den Sie für das Datenobjekt konfiguriert haben. Der Standardwert ist der Name eines physischen Datenobjekts und der Pfad zu dem Objekt im Modellrepository. Schreibgeschützt.

Lookup-Quellen mit Parametern

Sie können eine Lookup-Quelle aus einem physischen Datenobjekt erstellen, das Parameter enthält. Wenn Sie das physische Datenobjekt zu einem Mapping hinzufügen, werden die Parameter auf der Registerkarte **Datenobjektparameter** angezeigt.

Die folgende Abbildung zeigt die Registerkarte **Datenobjektparameter** in der Lookup-Umwandlung:



Das Bild zeigt den Customer_Order_Parm. Customer_Order_Parm ist ein Parameter für den Quelldateinamen in einem Einfachdatei-Datenobjekt. Um den Quelldateinamen in dem Mapping zu überschreiben, binden Sie den Customer_Order_Parm an einen Parameter im Mapping. Klicken Sie auf **Als Mapping-Parameter anzeigen**, um einen doppelten Parameter im Mapping zu erstellen.

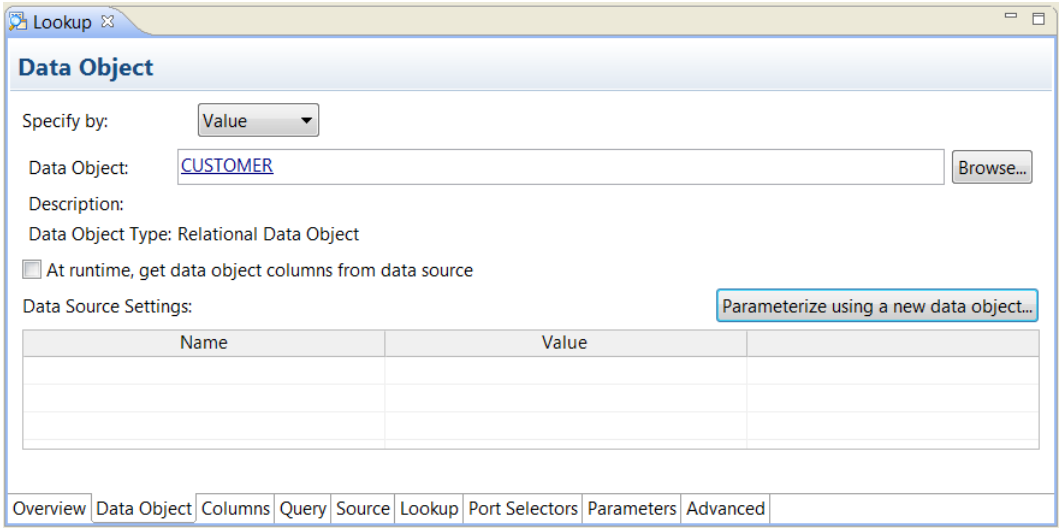
Parameter in einem doppelten Datenobjekt konfigurieren

Sie können ein doppeltes Datenobjekt im Repository erstellen und die Eigenschaften für dieses physische Datenobjekt parametrisieren. Definieren Sie Standardwerte für die Eigenschaften, wie z. B. die Verbindung, den Ressourcennamen, den Tabelleneigentümer oder den Namen der Steuerungsdatei.

Sie können ein doppeltes Datenobjekt für relationale Datenobjekte und Einfachdatei-Datenobjekte erstellen. Sie können ein doppeltes Datenobjekt in einer wiederverwendbaren Lookup-Umwandlung und in einer nicht wiederverwendbaren Lookup-Umwandlung erstellen.

Erstellen Sie ein doppeltes Datenobjekt auf der Registerkarte **Datenobjekt** der Lookup-Umwandlung. Sie können ein doppeltes Objekt erstellen, wenn Sie das Datenobjekt als Wert angeben. Beim Erstellen des doppelten Datenobjekts ersetzen Sie den Namen des Datenobjekts in der Lookup-Umwandlung durch den Namen des doppelten Datenobjekts. Das Developer Tool erstellt Parameter für die Datenobjekteigenschaften. Das Developer Tool fordert Sie zur Eingabe von Standardwerten für die Parameter auf. Die Namenssyntax des doppelten Datenobjekts lautet: `<Original object name>_Param`.

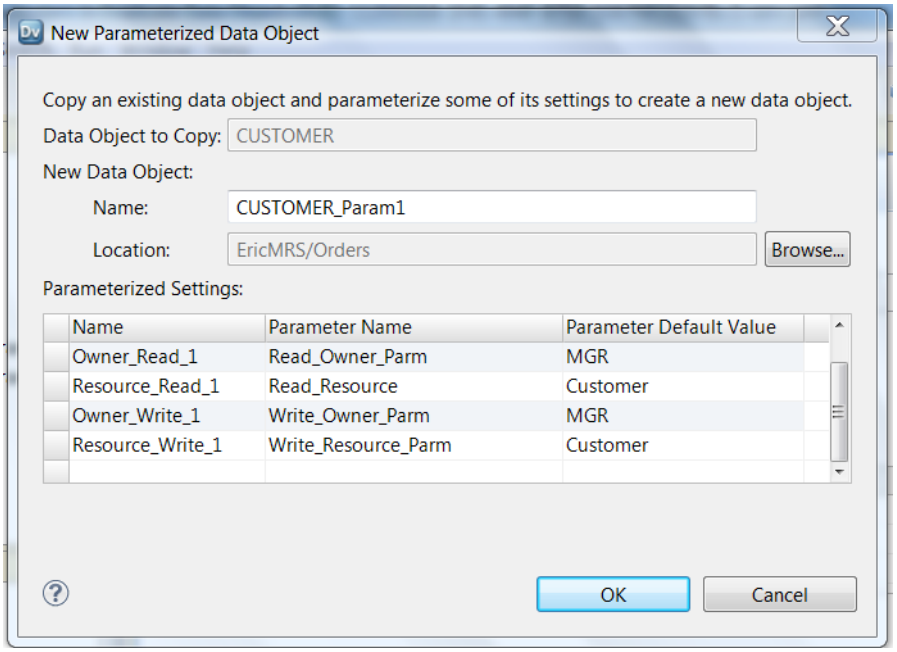
Die folgende Abbildung zeigt die Schaltfläche **Mithilfe eines neuen Datenobjekts parametrisieren** auf der Registerkarte **Datenobjekt** für relationale und Einfachdatei-Datenobjekte:



Wenn Sie ein Datenobjekt durch die Erstellung eines doppelten Datenobjekts parametrisieren, erstellt das Developer Tool einen Parametersatz für das Datenobjekt. Das Developer Tool erstellt verschiedene Parameter basierend darauf, ob das Datenobjekt eine Einfachdatei oder ein relationales Datenobjekt ist.

Konfigurieren Sie beim Erstellen des doppelten Datenobjekts die Standardparameterwerte im Dialogfeld **Neue parametrisierte Datenobjekte**.

Die folgende Abbildung zeigt das Dialogfeld **Neue parametrisierte Datenobjekte** für ein relationales Datenobjekt:

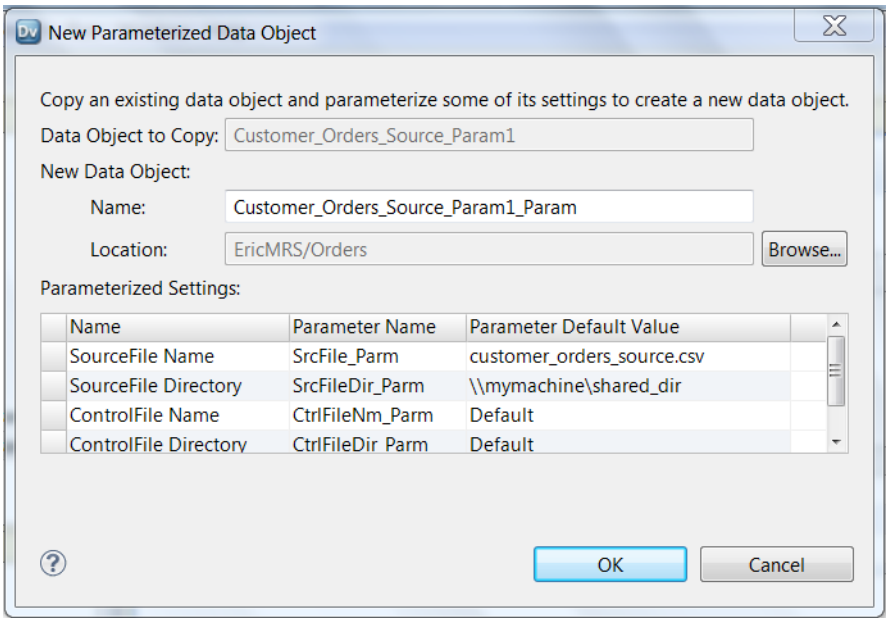


Sie können den Namen des Datenobjekts ändern. Geben Sie die Parameterstandardwerte für den Eigentümer und die Ressourcenparameter ein.

Wenn das ursprüngliche Datenobjekt parametrisiert ist, kopiert das Developer Tool die Parameter aus dem ursprünglichen Datenobjekt in das doppelte Datenobjekt. Wenn eine ursprüngliche Eigenschaft nicht parametrisiert ist, erstellt das Developer Tool für diese einen Parameter im doppelten Datenobjekt. Das Developer Tool verwendet den ursprünglichen Eigenschaftswert als Standardparameterwert im doppelten

Datenobjekt. Wenn das Developer Tool keinen ursprünglichen Eigenschaftswert ermitteln kann, erstellt das Developer Tool einen Parameter mit einem Standardwert basierend auf dem Parametertyp.

Das folgende Bild zeigt das Dialogfeld **Neue parametrisierte Datenobjekte** für ein Einfachdatei-Datenobjekt:



Konfigurieren Sie Standardparameterwerte für die Quelldatei und das Quelldateiverzeichnis. Wenn die Einfachdatei über eine Steuerdatei verfügt, konfigurieren Sie den Namen und das Verzeichnis der Steuerdatei.

Nachdem Sie die Standardwerte konfiguriert haben, erstellt das Developer Tool das doppelte Datenobjekt. Der Name des doppelten Datenobjekts wird auf der Registerkarte **Datenobjekt** der Lookup-Umwandlung angezeigt. Das doppelte Datenobjekt wird im **Objekt-Navigator** angezeigt.

Um die Parameterwerte für das Datenobjekt nach dessen Erstellung zu ändern, öffnen Sie das physische Datenobjekt im **Objekt-Navigator**. Klicken Sie auf die Registerkarte **Parameter**.

Portauswahlen

Sie können eine Lookup-Bedingung erstellen, wenn die Lookup-Umwandlung generierte Ports enthält. Sie können auf einen dynamischen Port oder eine Portauswahl in einer Lookup-Bedingung verweisen. Sie können auch mit einem Ausdrucksparameter den vollständigen Lookup-Ausdruck parametrisieren.

Wenn der dynamische Port mehrere generierte Ports enthält, können Sie eine Portauswahl für das Filtern der generierten Ports in der Lookup-Bedingung definieren. Die Lookup-Quelle kann sich in einem dynamischen Mapping ändern. Sie können eine Portauswahl so konfigurieren, dass nach den für die Lookup-Spalte zu verwendenden Ports gefiltert wird. Die Portauswahl der Lookup-Quelle muss dieselbe Anzahl von Ports wie die Portauswahl der Eingabespalten enthalten.

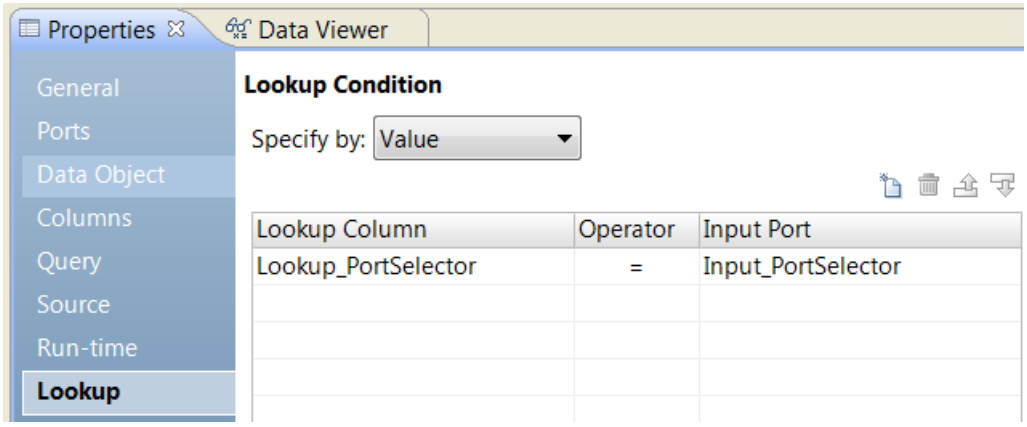
Beispiel: Lookup_PortSelector enthält die folgenden Ports:

```
C_CustKey
C_OrderKey
```

Input_PortSelector enthält die folgenden Eingabeports:

```
CustomerID_IN
OrderID_IN
```

Die folgende Abbildung zeigt eine Lookup-Bedingung, die Portauswahlen enthält:



Die Lookup-Bedingung wird um den folgenden Ausdruck erweitert:

```
C_CustKey = CustomerID_IN AND C_OrderKey = OrderID_IN
```

Wenn die Lookup-Bedingung mehrere Ports enthält, können Sie einen Operator konfigurieren. Beispielsweise können Sie den Operator in größer als (>) ändern. Die Lookup-Bedingung wird um den folgenden Ausdruck erweitert:

```
C_CustKey > CustomerID_IN AND C_OrderKey > OrderID_IN
```

Sie können eine Lookup-Bedingung erstellen, die einen dynamischen Port enthält:

```
Lookup_PortSelector = Dynamic_Input_Port
```

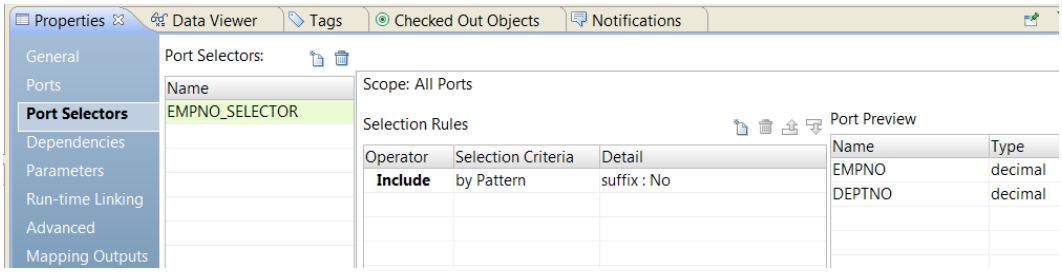
Der dynamische Port muss dieselbe Anzahl von Ports wie die Portauswahl enthalten.

Portauswahl-Konfiguration

Wenn Sie eine Portauswahl konfigurieren, definieren Sie Auswahlregeln, um zu bestimmen, welche generierten Ports einbezogen werden sollen. Die Auswahlregeln ähneln den Eingaberegeln, die Sie für dynamische Ports konfigurieren können.

Eine Portauswahl kann statische oder generierte Ports enthalten. Konfigurieren Sie eine Portauswahl auf der Registerkarte **Portauswahl**.

Die folgende Abbildung zeigt die Registerkarte **Portauswahl**:



Konfigurieren Sie die folgenden Eigenschaften für eine Portauswahl:

Name

Gibt die Portauswahl an. Sie können mehrere Portauswahlen in einer Umwandlung erstellen und in Ausdrücken auf sie verweisen.

Bereich

Gibt eine Gruppe von Ports an, auf die die Portauswahl angewendet wird. Sie müssen den Bereich auswählen, wenn Sie eine Portauswahl für eine Joiner- oder Lookup-Umwandlung erstellen. Diese Umwandlungen verfügen über mehrere Eingabegruppen. Die Joiner-Umwandlung weist einen Master- oder einen Detail-Bereich auf. Die Lookup-Umwandlung weist einen Import- oder einen Lookup-Bereich auf. Die Ausdrucksumwandlung verfügt über eine Eingabegruppe. Der Bereich ist immer „Alle Ports".

Auswahlregeln

Bestimmt die Ports, die in die Portauswahl einbezogen werden sollen. Beim Erstellen der Auswahlregeln werden im Bereich **Portvorschau** die Ports angezeigt, die sich aus den aktuellen Eingabeports qualifizieren. Diese Ports ändern sich möglicherweise. Konfigurieren Sie die Auswahlregeln, um Ports aus unterschiedlichen Quellen anzupassen.

Auswahlregeln

Die mit einer Portauswahl verbundenen Auswahlregeln bestimmen die Ports, die in die Portauswahl aufgenommen werden.

Beim Erstellen der Auswahlregeln werden im Bereich **Portvorschau** die Ports angezeigt, die sich aus den aktuellen Eingabeports qualifizieren. Diese Ports ändern sich möglicherweise. Konfigurieren Sie die Auswahlregeln, um Ports aus unterschiedlichen Quellen anzupassen.

Erstellen Sie Auswahlregeln auf Grundlage der folgenden Kriterien:

Operator

Schließt die Ports ein bzw. aus, die Auswahlregeln zurückgeben. Standardwert ist einschließen. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.

Auswahlkriterien

Der Auswahlregeltyp, den Sie erstellen möchten. Sie können eine Regel basierend auf dem Porttyp oder dem Spaltennamen erstellen. Um Ports basierend auf dem Spaltennamen einzuschließen, suchen Sie nach bestimmten Namen oder nach einem Muster von Zeichen im Namen.

Detail

Die auf die Auswahlkriterien anzuwendenden Werte. Wenn die Auswahlkriterien nach Spaltenname erfolgen, konfigurieren Sie die Zeichenfolge oder den Namen, nach der bzw. dem gesucht werden soll. Wenn die Auswahlkriterien nach Porttyp erfolgen, wählen Sie die einzuschließenden Porttypen aus.

In der folgenden Tabelle werden die Auswahlkriterien beschrieben und wie die Details für die Kriterien angegeben werden müssen:

Auswahlkriterien	Beschreibung	Detail
Alle	Schließt alle Ports ein.	Keine Details erforderlich.
Name	Filtert Ports basierend auf dem Portnamen.	Wählen Sie die Portnamen aus einer Liste mit Werten aus oder verwenden Sie einen Parameter des Typs „Port" oder „Portliste".

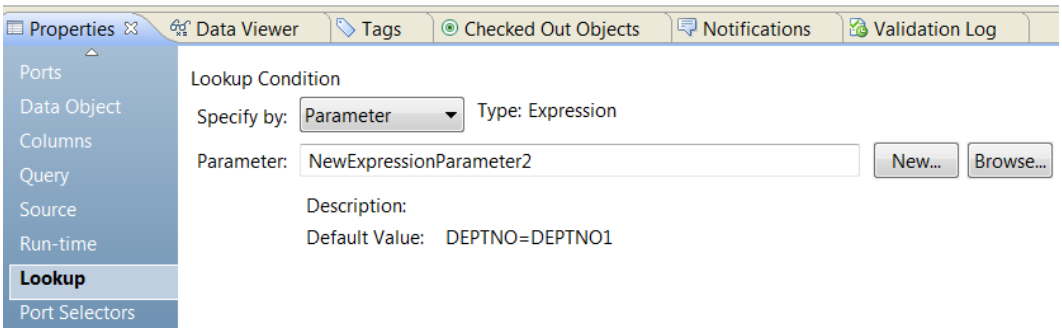
Auswahlkriterien	Beschreibung	Detail
Typ	Filtert Ports basierend auf dem Datentyp jedes Ports.	Wählen Sie Datentypen aus einer Liste aus.
Muster	Filtert Ports nach einer Zeichenfolge von Zeichen im Namen oder nach einem regulären Ausdruck.	Wählen Sie ein Präfix, ein Suffix oder einen regulären Ausdruck als Mustertyp für den Portnamen aus. Geben Sie anschließend einen Wert für das Muster ein oder verwenden Sie einen Parameter des Typs „Zeichenfolge“.

Die Lookup-Bedingung parametrisieren

Sie können einen Ausdrucksparameter zum Definieren der Lookup-Bedingung konfigurieren. Ein Ausdrucksparameter enthält einen vollständigen Ausdruck, den Sie in einem Ausdruckseditor erstellen können. Sie können einen Mapping-Parameter definieren, um den Ausdrucksparameter zur Laufzeit zu überschreiben.

Wenn Sie eine Lookup-Bedingung mithilfe eines Parameters angeben, können Sie nach einem Ausdrucksparameter suchen oder einen Parameter erstellen.

Die folgende Abbildung zeigt, wo ein Ausdrucksparameter für die Lookup-Bedingung konfiguriert werden kann:



Klicken Sie zum Erstellen eines Parameters auf **Neu**. Definieren Sie einen Namen für den Parameter und bearbeiten Sie den Standardwert. Der Standardwert für den Ausdrucksparameter ist der vollständige Ausdruck zum Definieren der Lookup-Bedingung. Sie können in dem Ausdruck generierte Ports, dynamische Ports und Portauswahlen verwenden.

Hinweis: Wenn Sie den Ausdruck erstellen, stellt die Lookup-Spalte immer den ersten Wert dar und die Eingabespalte den zweiten Wert.

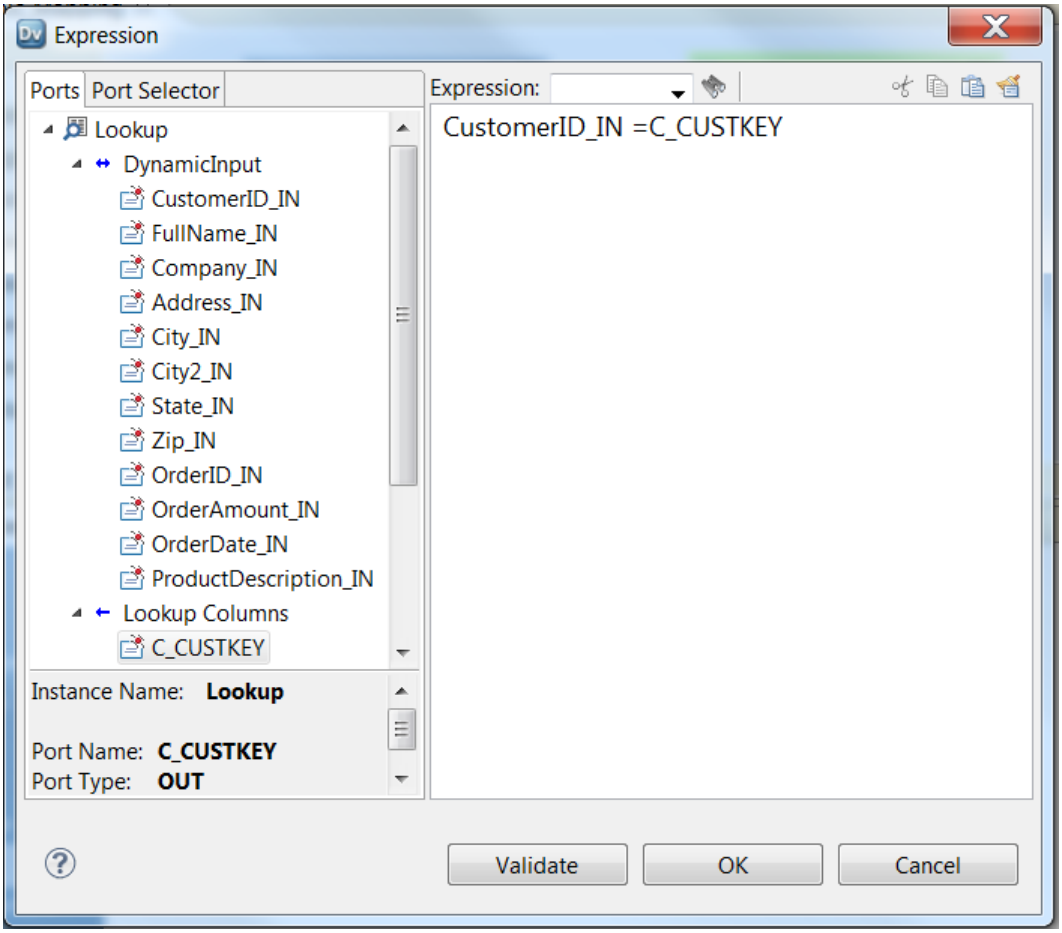
Beispiel: Sie erstellen die folgende Lookup-Bedingung in einem Ausdrucksparameter:

```
CustomerID_IN = C_CUSTKEY
```

`CustomerID_IN` ist die Lookup-Spalte.

`C_CUSTKEY` ist die Eingabespalte.

Die folgende Abbildung zeigt den Lookup-Ausdruck im Ausdruckseditor:



Erstellen einer Portauswahl

Erstellen Sie einen Port zum Festlegen, welche Ports in einem dynamischen Ausdruck, einer Lookup-Bedingung oder einer Joiner-Bedingung verwendet werden sollen.

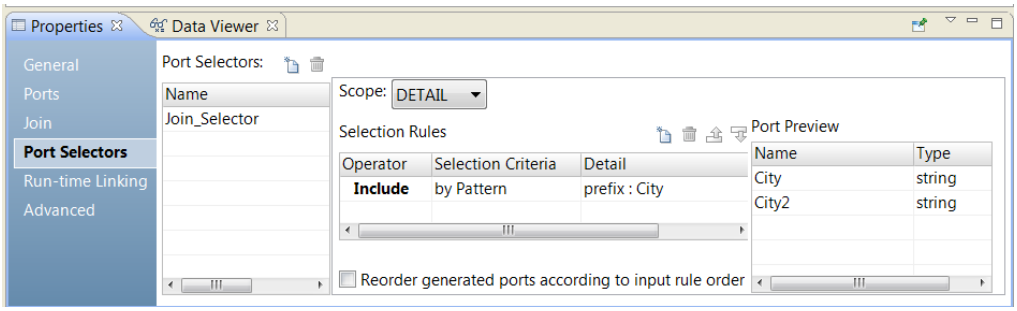
1. Klicken Sie auf die Registerkarte **Portauswahlen**.
2. Klicken Sie im Bereich **Portauswahlen** auf **Neu**.

   Das Developer Tool erstellt eine Portauswahl mit einer Standardauswahlregel, die alle Ports enthält.
3. Ändern Sie im Bereich **Portauswahlen** den Namen der Portauswahl in einen eindeutigen Namen.
4. Wenn Sie die Joiner-Umwandlung oder die Lookup-Umwandlung bearbeiten, wählen Sie den Bereich aus.

   Die verfügbaren Ports ändern sich basierend auf der Gruppe der von Ihnen ausgewählten Ports.
5. Wählen Sie im Bereich **Auswahlregeln** einen **Operator** aus.
   - Einschließen. Erstellen Sie eine Regel, die Ports für die Portauswahl einschließt. Sie müssen Ports einschließen, bevor Sie diese ausschließen können.
   - Ausschließen. Erstellen Sie eine Regel, die bestimmte Ports aus der Portauswahl ausschließt.
6. Wählen Sie die **Auswahlkriterien** aus.
   - Nach Name. Wählen Sie bestimmte Ports nach Namen aus. Sie können die Portnamen aus einer Liste mit Ports im Bereich auswählen.

- Nach Typ. Wählen Sie Ports nach Typ aus. Sie können eine oder mehrere Datentypen auswählen.
- Nach Muster. Wählen Sie Ports nach einem Muster von Zeichen im Portnamen aus. Sie können die Suche mit bestimmten Zeichen durchführen oder einen regulären Ausdruck erstellen.

Die folgende Abbildung zeigt die Registerkarte „Portauswahl“:



7. Klicken Sie auf die Spalte **Detail**.

   Das Dialogfeld **Eingaberegel-Detail** wird angezeigt.

8. Wählen Sie die Werte aus, nach denen Ports gefiltert werden sollen.
   - Nach Name. Wählen Sie, ob eine Portliste nach Wert oder nach einem Parameter erstellt werden soll. Klicken Sie auf **Auswählen**, um die Ports auf der Liste auszuwählen.
   - Nach Typ. Wählen Sie mindestens einen Datentyp aus einer Liste aus. Im Bereich **Portvorschau** werden Ports des von Ihnen ausgewählten Typs angezeigt.
   - Nach Muster. Wählen Sie, ob das Präfix oder das Suffix des Portnamens nach einem bestimmten Muster von Zeichen durchsucht werden soll. Oder entscheiden Sie sich für das Erstellen eines regulären Ausdrucks für die Suche. Konfigurieren Sie einen Parameter oder konfigurieren Sie das Muster für die Suche.

   Im Bereich **Portvorschau** werden während der Konfiguration der Regeln die Ports in der Portauswahl angezeigt.

9. Um die Ports in der Portauswahl neu anzuordnen, wählen Sie **Erzeugte Ports gemäß der Reihenfolge der Eingaberegeln neu sortieren** aus.

Laufzeiteigenschaften

Legen Sie die Laufzeiteigenschaften fest, um das Lookup-Caching zu aktivieren und konfigurieren. Sie müssen die Lookup-Umwandlung zu einem Mapping hinzufügen, bevor Sie die Laufzeit-Lookup-Eigenschaften konfigurieren können.

In der folgenden Tabelle werden die Laufzeiteigenschaften für Lookup-Umwandlungen beschrieben, die Einfachdateien, Referenztabellen bzw. relationale Lookups ausführen:

Eigenschaft	Beschreibung
Lookup-Zwischenspeicherung ist aktiviert	Gibt an, ob der Integrationsdienst Lookup-Werte im Cache zwischenspeichert. Wenn Sie die Lookup-Zwischenspeicherung aktivieren, fragt der Integrationsdienst die Lookup-Quelle einmal ab, speichert die Werte zwischen und sucht die Werte im Cache. Durch das Caching der Lookup-Werte kann die Leistung für umfangreiche Lookup-Tabellen verbessert werden. Wenn Sie die Zwischenspeicherung deaktivieren, gibt der Integrationsdienst jedes Mal, wenn eine Zeile an die Umwandlung weitergeleitet wird, eine SELECT-Anweisung an die Lookup-Quelle für Lookup-Werte aus. Einfachdatei-Lookups werden vom Integrationsdienst immer zwischengespeichert.
Lookup-Daten-Cachegröße	Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Daten-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto" auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".
Lookup-Index-Cachegröße	Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Index-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto" auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".
Cache-Dateinamenpräfix	Präfix für die Cachedatei. Sie können das Cache-Dateinamenpräfix für einen persistenten Lookup-Cache angeben.
Buildvorstufen-Lookup-Cache	Ermöglicht dem Integrationsdienst, den Lookup-Cache zu erstellen, bevor die Lookup-Umwandlung Daten empfängt. Der Integrationsdienst kann mehrere Lookup-Cachedateien gleichzeitig erstellen, um die Leistung zu erhöhen. Konfigurieren Sie eine der folgenden Optionen: - Auto. Der Integrationsdienst bestimmt den Wert. - Immer zugelassen. Ermöglicht dem Integrationsdienst, den Lookup-Cache zu erstellen, bevor die Lookup-Umwandlung Daten empfängt. Der Integrationsdienst kann mehrere Lookup-Cachedateien gleichzeitig erstellen, um die Leistung zu erhöhen. - Immer unzulässig. Der Integrationsdienst kann den Lookup-Cache nicht erstellen, bevor die Lookup-Umwandlung der erste Zeile empfängt.
Lookup-Cache-Verzeichnisname	Verzeichnis zum Erstellen der Lookup-Cachedateien, wenn Sie die Lookup-Umwandlung zum Cachen der Lookup-Quelle konfigurieren. Standardwert ist der CacheDir-Systemparameter. Sie können für diese Eigenschaft einen anderen Systemparameter oder einen benutzerdefinierten Parameter konfigurieren.
Aus Lookup-Quelle erneut im Cache zwischenspeichern	Erstellt den Lookup-Cache neu, um den persistenten Cache mit der Lookup-Tabelle zu synchronisieren. Speichern Sie das Lookup aus der Datenbank erneut im Cache, wenn die Lookup-Umwandlung einen persistenten Lookup-Cache aufweist und die Lookup-Tabelle gelegentlich geändert wird.

Verwandte Themen:

- "Cachegröße" auf Seite 662

Erweiterte Eigenschaften

Konfigurieren Sie den persistenten Lookup-Cache und die Verbindung zu einer relationalen Datenbank in den erweiterten Eigenschaften. Die angezeigten Eigenschaften basieren auf dem Typ der Lookup-Quelle.

In der folgenden Tabelle werden erweiterte Eigenschaften für jeden Typ der Lookup-Quelle beschrieben:

Eigenschaft	Typ der Lookup-Quelle	Beschreibung
Lookup-Cache persistent	Einfachdatei, Referenztabelle, relationales Lookup	Zeigt an, ob der Integrationsdienst einen persistenten Lookup-Cache verwendet, der aus mindestens zwei Cache-Dateien besteht. Wenn eine Lookup-Umwandlung für einen persistenten Lookup-Cache konfiguriert ist und keine persistenten Lookup-Cache-Dateien vorhanden sind, erstellt der Integrationsdienst die Dateien.
Zeichenfolgenvergleich unter Beachtung der Groß-/Kleinschreibung	Einfachdatei	Der Integrationsdienst verwendet auf Groß- und Kleinschreibung basierte Zeichenfolgenvergleiche, während Lookups in Zeichenfolgenspalten durchgeführt werden.
Null-Ordnungsfolge	Einfachdatei	Bestimmt, wie der Integrationsdienst Nullwerte anordnet. Sie können eine hohe oder niedrige Sortierung von Nullwerten auswählen. Standardmäßig verwendet der Integrationsdienst eine hohe Sortierung von Nullwerten. Dies überschreibt die Konfiguration des Integrationsdiensts, um Nullen in Vergleichsoperatoren als hoch, niedrig oder als Null zu behandeln. Bei relationalen Lookups basiert die Null-Ordnungsfolge auf dem Datenbankstandardwert.
Tracingebene	Einfachdatei, logisches Datenobjekt, Referenztabelle, relationales Lookup	Legt die Menge an Details fest, die im Protokoll für diese Umwandlung angezeigt wird. Sie können „Kurz", „Normal", „Ausführliche Initialisierung" oder „Ausführliche Daten" wählen. Standardwert ist „Normal".
Aktualisieren, andernfalls einfügen	Referenztabelle, relationales Lookup	Gilt nur für den dynamischen Lookup-Cache. Der Integrationsdienst aktualisiert die Zeile im Cache, wenn der Zeilentyp, der die Lookup-Umwandlung eingibt, aktualisiert wird, die Zeile im Index-Cache vorhanden ist und sich die Cache-Daten von der vorhandenen Zeile unterscheiden. Der Integrationsdienst fügt die Zeile in den Cache ein, sofern diese neu ist.
Einfügen, andernfalls aktualisieren	Referenztabelle, relationales Lookup	Gilt nur für den dynamischen Lookup-Cache. Der Integrationsdienst fügt die Zeile in den Cache ein, sofern der Zeilentyp, der die Lookup-Umwandlung eingibt, den Insert-Typ aufweist und neu ist. Wenn die Zeile im Index-Cache vorhanden ist, der Daten-Cache sich aber von der aktuellen Zeile unterscheidet, aktualisiert der Integrationsdienst die Zeile im Daten-Cache.

Eigenschaft	Typ der Lookup-Quelle	Beschreibung
Bei Aktualisierung alten Wert ausgeben	Referenztabelle, relationales Lookup	Der Integrationsdienst gibt den Wert aus, der vor der Aktualisierung der Zeile im Cache vorhanden war. Andernfalls gibt der Integrationsdienst den aktualisierten Wert aus, den er in den Cache schreibt.
Dynamische Cache-Bedingung aktualisieren	Referenztabelle, relationales Lookup	Gilt nur für den dynamischen Lookup-Cache. Ein Ausdruck, der angibt, ob der dynamische Cache aktualisiert werden soll. Der Integrationsdienst aktualisiert den Cache, wenn die Bedingung „True“ lautet und die Daten im Cache vorhanden sind. Standardwert ist „True“.
Verbindung	Referenztabelle, relationales Lookup	Verbindung zu der relationalen Datenbank, welche die relationale Lookup-Quelle enthält. Sie können einen Parameter für die Verbindung verwenden. Für die Lookups der Referenztabelle ist dieses Feld schreibgeschützt.
Sortierte Eingabe	Einfachdatei	Gibt an, dass die Eingabedaten nach Gruppen vorsortiert sind.
Datumzeit-Format	Einfachdatei	Definieren Sie ein Datumzeit-Format und die Feldbreite. Millisekunden-, Mikrosekunden- oder Nanosekunden-Formate haben eine Feldbreite von 29. Wenn Sie für einen Port kein Datumzeit-Format auswählen, können Sie ein beliebiges Datumzeit-Format eingeben. Der Standardwert ist „YYYY-MM-DD HH24:MI:SS“. Das Datumzeit-Format ändert nicht die Größe des Ports. Dieses Feld ist schreibgeschützt.
Tausendertrennzeichen	Einfachdatei	Der Wert ist „Keiner“. Dieses Feld ist schreibgeschützt.
Dezimaltrennzeichen	Einfachdatei	Der Wert ist ein Punkt. Dieses Feld ist schreibgeschützt.

Erstellen einer wiederverwendbaren Lookup-Umwandlung

Erstellen einer Lookup-Umwandlung zum Nachschlagen von Daten in einer Einfachdatei, einem logischen Datenobjekt, einer Referenztabelle oder einem relationalen Datenobjekt.

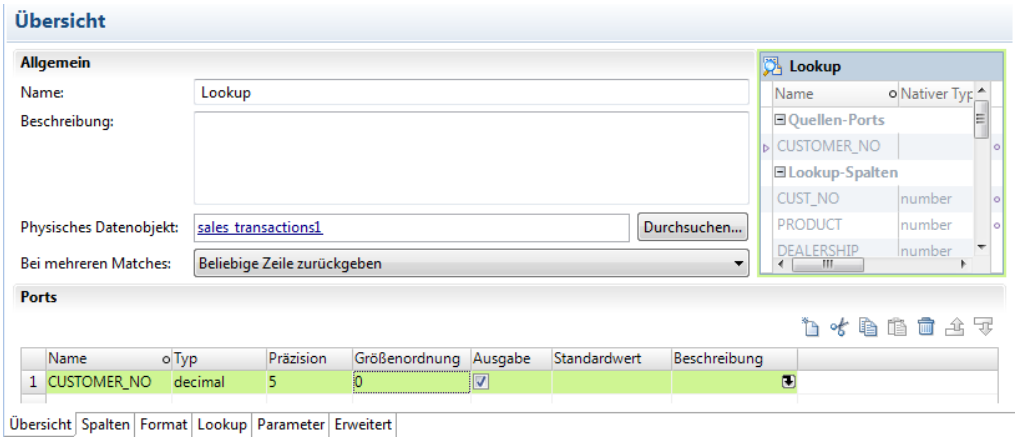
1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.
3. Navigieren Sie zum Lookup-Assistenten.
4. Wählen Sie **Lookup von Einfachdatei-Datenobjekt**, **Lookup von logischem Datenobjekt**, **Lookup von Referenztabelle** oder **Lookup von relationalem Datenobjekt**.
5. Klicken Sie auf **Weiter**.

   Das Dialogfeld **Neue Lookup-Umwandlung** wird angezeigt.
6. Wählen Sie ein physisches Datenobjekt oder eine Referenztabelle im Developer-Tool.

7. Geben Sie einen Namen für die Umwandlung ein.
8. Geben Sie für **Bei mehreren Matches** an, welche Zeilen von der Lookup-Umwandlung ausgegeben werden, wenn sie mehrere Zeilen findet, die mit der Lookup-Bedingung übereinstimmen.
9. Klicken Sie auf **Fertig stellen**.

   Die Lookup-Umwandlung wird im Editor angezeigt.
10. Fügen Sie der Umwandlung im Bereich **Ports** der Ansicht **Übersicht** Ausgabeports hinzu.

    Die folgende Abbildung zeigt den CUSTOMER_NO-Ausgabeport in einer Lookup-Umwandlung:



11. Wählen Sie auf der Registerkarte **Laufzeit** der Ansicht **Eigenschaften Lookup-Caching ist aktiviert** aus, um das Lookup-Caching zu aktivieren.

    Hinweis: Sie müssen die Lookup-Umwandlung zu einem Mapping hinzufügen, bevor Sie die Laufzeit-Lookup-Eigenschaften konfigurieren können.
12. Auf der Registerkarte **Lookup** der Ansicht **Eigenschaften** können Sie eine oder mehrere Lookup-Bedingungen hinzufügen.
13. Konfigurieren Sie auf der Registerkarte **Erweitert** der Ansicht **Eigenschaften** die Tracingebene, die dynamischen Lookup-Cache-Eigenschaften und die Laufzeitverbindung.
14. Speichern Sie die Umwandlung.

Erstellen einer nicht wiederverwendbare Lookup-Umwandlung

Erstellen Sie eine nicht wiederverwendbare Lookup-Umwandlung in einem Mapping oder Mapplet.

1. Ziehen Sie in einem Mapping oder Mapplet eine Lookup-Umwandlung aus der Umwandlungspalette in den Editor.

   Das Dialogfeld **Neu** wird eingeblendet.
2. Wählen Sie **Lookup von Einfachdatei-Datenobjekt**, **Lookup von logischem Datenobjekt**, **Lookup von Referenztabelle** oder **Lookup von relationalem Datenobjekt**.
3. Klicken Sie auf **Weiter**.

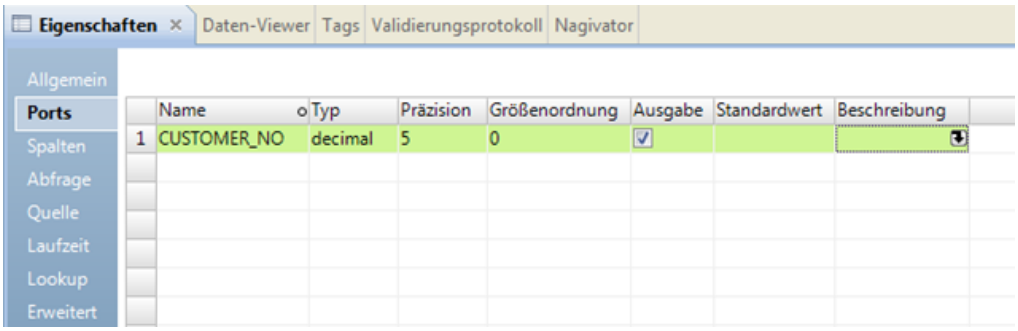
   Das Dialogfeld **Neue Lookup-Umwandlung** wird angezeigt.
4. Wählen Sie ein physisches Datenobjekt oder eine Referenztabelle im Developer-Tool.

5. Geben Sie einen Namen für die Umwandlung ein.
6. Geben Sie für **Bei mehreren Matches** an, welche Zeilen von der Lookup-Umwandlung ausgegeben werden, wenn sie mehrere Zeilen findet, die mit der Lookup-Bedingung übereinstimmen.
7. Klicken Sie auf **Fertig stellen**.

   Die Lookup-Umwandlung wird im Editor angezeigt.
8. Wählen Sie die Lookup-Umwandlung im Editor.

   Die Symbolleiste wird oberhalb der Umwandlung eingeblendet.
9. Fügen Sie der Umwandlung auf der Registerkarte **Ports** der Ansicht **Eigenschaften** Ausgabeports hinzu.

   Die folgende Abbildung zeigt den CUSTOMER_NO-Ausgabeport in einer Lookup-Umwandlung:



10. Wählen Sie auf der Registerkarte **Laufzeit** der Ansicht **Eigenschaften Lookup-Caching ist aktiviert** aus, um das Lookup-Caching zu aktivieren.

    Hinweis: Sie müssen die Lookup-Umwandlung zu einem Mapping hinzufügen, bevor Sie die Laufzeit-Lookup-Eigenschaften konfigurieren können.
11. Auf der Registerkarte **Lookup** der Ansicht **Eigenschaften** können Sie eine oder mehrere Lookup-Bedingungen hinzufügen.
12. Konfigurieren Sie auf der Registerkarte **Erweitert** der Ansicht **Eigenschaften** die Tracingebene, die dynamischen Lookup-Cache-Eigenschaften und die Laufzeitverbindung.
13. Speichern Sie die Umwandlung.

Erstellen einer nicht verbundenen Lookup-Umwandlung

Erstellen Sie eine nicht verbundene Lookup-Umwandlung, wenn Sie ein Lookup aus einem Ausdruck erstellen möchten. Sie können eine wiederverwendbare oder eine nicht wiederverwendbare nicht verbundene Lookup-Umwandlung in einer Einfachdatei, einer Referenztabelle oder einem relationalen Datenobjekt erstellen.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.
3. Navigieren Sie zum Lookup-Assistenten.
4. Wählen Sie **Lookup für Einfachdatei-Datenobjekt**, **Lookup für Referenztabelle** oder **Relationales Datenobjekt-Lookup** aus.
5. Klicken Sie auf **Weiter**.

   Das Dialogfeld **Neues Lookup** wird angezeigt.

6. Wählen Sie ein physisches Datenobjekt oder eine Referenztabelle im Developer-Tool aus.
7. Geben Sie einen Namen für die Umwandlung ein.
8. Geben Sie für **Bei mehreren Matches** an, welche Zeile von der Lookup-Umwandlung zurückgegeben wird, wenn mehrere Zeilen gefunden werden, die mit der Lookup-Bedingung übereinstimmen. Wählen Sie für ein nicht verbundenes Lookup nicht die Option **Alle zurückgeben** aus.
9. Klicken Sie auf **Fertig stellen**.

   Die Lookup-Umwandlung wird im Editor angezeigt.
10. Fügen Sie der Umwandlung im Bereich **Ports** in der Ansicht **Übersicht** Ports hinzu.

    Erstellen Sie einen Eingabeport für jedes Argument im :LKP-Ausdruck. Erstellen Sie einen Eingabeport für jede von Ihnen erstellte Lookup-Bedingung. Sie können einen Eingabeport in mehreren Bedingungen verwenden.
11. Konfigurieren Sie im Bereich **Ports** in der Ansicht **Übersicht** einen Port als Rückgabeport.
12. Fügen Sie in der Ansicht **Lookup** eine oder mehrere Lookup-Bedingungen hinzu, um die Eingabewerte der Umwandlung mit den Werten in der Lookup-Quelle oder im Cache zu vergleichen.

    Wenn die Bedingung TRUE ist, gibt das Lookup einen Wert im Rückgabeport zurück. Wenn die Lookup-Bedingung falsch ist, gibt Lookup NULL zurück.
13. Erstellen Sie einen :LKP-Ausdruck für einen Port in einer Umwandlung, die Ausdrücke zulässt, wie zum Beispiel eine Aggregatorumwandlung, Ausdrucksumwandlung oder Update-Strategie-Umwandlung.
14. Wenn Sie ein Mapping erstellen, fügen Sie die nicht verbundene Lookup-Umwandlung zum Mapping im Editor hinzu, verbinden Sie die Ports aber nicht mit anderen Umwandlungen im Mapping.

Nicht verbundenes Lookup - Beispiel

Ein Einzelhandelsgeschäft in Kalifornien fügt dem Preis für Artikel, die an Kunden innerhalb des Bundesstaates verkauft werden, eine Verkaufssteuer hinzu. Der Steuerbetrag richtet sich nach dem County, in dem der Kunde wohnt. Zum Abrufen der Verkaufssteuer erstellen Sie eine Lookup-Umwandlung, die einen County-Namen erhält und anschließend den Verkaufssteuerbetrag für das County zurückgibt. Wenn im County keine Verkaufssteuer erhoben wird, gibt die Lookup-Umwandlung den Wert NULL zurück. Rufen Sie das Lookup über eine Ausdrucksumwandlung ab.

Führen Sie die folgenden Schritte zum Konfigurieren eines nicht verbundenen Lookups von Verkaufssteuern nach County durch:

1. Importieren eines physischen Einfachdatei-Datenobjekts, das die Verkaufssteuerbeträge nach County enthält.
2. Erstellen Sie die nicht verbundene Lookup-Umwandlung.
3. Fügen Sie der Lookup-Umwandlung Eingabeports hinzu.
4. Definieren Sie den Rückgabeport.
5. Erstellen Sie die Lookup-Bedingung.
6. Rufen Sie das Lookup über eine Ausdrucksumwandlung ab.

Schritt 1. Importieren der Lookup-Quelle für die Verkaufssteuer in das Modellrepository

Die Verkaufssteuerdatei muss sich vor dem Erstellen der Lookup-Umwandlung bereits im Modellrepository befinden. In diesem Szenario enthält die Verkaufssteuerdatei zwei Felder: Sales_County und County_SalesTax (Umsatz_County und County_Verkaufssteuer). Bei County handelt es sich um eine

Zeichenfolge, die einen County-Namen enthält. Bei County_SalesTax handelt es sich um ein Dezimalfeld, das einen Steuersatz für das County enthält. Die Verkaufssteuerdatei fungiert als Lookup-Quelle.

Schritt 2. Erstellen der nicht verbundenen Lookup-Umwandlung

Erstellen Sie eine wiederverwendbare Einfachdatei-Lookup-Umwandlung mit dem Einfachdatei-Datenobjekt "Verkaufssteuer". In diesem Szenario lautet der Name der Umwandlung Sales_Tax_Lookup (Verkaufssteuer_Lookup). Wählen Sie **Erste Zeile zurückgeben** für mehrere Übereinstimmungen aus.

Schritt 3. Definieren des Lookup-Umwandlungsports

Definieren Sie die Lookup-Umwandlungsports auf der Registerkarte **Ports** der Ansicht **Eigenschaften**.

Porttyp	Name	Typ	Länge	Größenordnung
Eingabe	In_County	String	25	
Ausgabe	SalesTax	Decimal	3	3

Schritt 4. Konfigurieren des Lookup-Umwandlungsrückgabeports

Der Rückgabeport ist das Feld in der Einfachdatei, das vom Lookup abgerufen wird. Auf der Registerkarte **Spalten** fungiert die Spalte "County_SalesTax" als Rückgabeport.

Wenn der Lookup wahr ist, findet der Integrationsdienst das County in der Einfachdateiquelle. Der Integrationsdienst gibt einen Umatzsteuerwert im Rückgabeport zurück. Wenn der Integrationsdienst das County nicht findet, ist das Lookup-Ergebnis falsch. Der Integrationsdienst gibt dann NULL im Rückgabeport zurück.

Schritt 5. Definieren der Lookup-Bedingung

Definieren Sie in der Ansicht **Lookup** die Lookup-Bedingung, um den Eingabewert mit Werten in der Lookup-Quelle zu vergleichen.

Klicken Sie zum Hinzufügen der Lookup-Bedingung auf die Spalte **Lookup**.

Die Lookup-Bedingung hat folgende Syntax:

```
SALES_COUNTY = IN_COUNTY
```

Schritt 6. Erstellen einer Ausdrucksumwandlung

Erstellen Sie eine Ausdrucksumwandlung, die Umsatzdatensätze aus einer Einfachdatei empfängt. Die Ausdrucksumwandlung empfängt eine Kundennummer, einen Verkaufsbetrag und das County, in dem der Verkauf stattgefunden hat. Zurückgegeben wird die Kundennummer, der Verkaufsbetrag und eine Verkaufssteuer.

Die Ausdrucksumwandlung verfügt über die folgenden Ports:

Porttyp	Name	Typ	Länge	Genauigkeit	Standardwert
Eingabe	County	String	25	10	
Pass-Through	Kunde	String	10		
Pass-Through	SalesAmt	Decimal	10	2	
Ausgabe	SalesTax	Decimal	10	2	0

Der SalesTax-Port enthält einen :LKP-Ausdruck. Der Ausdruck ruft die Sales_Tax_Lookup-Umwandlung auf und übergibt den County-Namen als Parameter. Die Sales_Tax_Lookup-Umwandlung gibt den Verkaufssteuersatz an den Ausdruck zurück. Die Ausdrucksumwandlung multipliziert den Steuersatz mit dem Verkaufsbetrag.

Geben Sie folgenden Ausdruck für den SalesTax-Port ein:

```
(:LKP.Sales_Tax_Lookup(County) * SalesAmt)
```

Der SalesTax-Port enthält das Ausdrucksergebnis. Wenn der Lookup fehlschlägt, gibt die Lookup-Umwandlung NULL zurück und der SalesTax-Port enthält Nullwerte.

Sie können einen Ausdruck hinzufügen, um den SalesTax-Port auf Nullwerte zu überprüfen. Wenn SalesTax gleich NULL ist, können Sie den SalesTax-Port für die Rückgabe von Null konfigurieren. Fügen Sie dem Lookup-Ausdruck folgenden Text hinzu, um nach Nullwerten zu suchen und Null zurückzugeben:

```
IIF(ISNULL(:LKP.Sales_Tax_Lookup(County) * SalesAmt),0, SalesTax)
```

KAPITEL 24

Lookup-Caches

Dieses Kapitel umfasst die folgenden Themen:

- Lookup-Caches – Übersicht, 382
- Lookup-Cache-Typen, 383
- Nicht zwischengespeichertes Lookup, 384
- Statischer Lookup-Cache, 384
- Persistenter Lookup-Cache, 385
- Dynamischer Lookup-Cache, 386
- Gemeinsam genutzter Lookup-Cache, 386
- Cache-Vergleich, 388
- Cache-Partitionierung für Lookups, 388

Lookup-Caches – Übersicht

Sie können eine Lookup-Umwandlung zum Zwischenspeichern einer relationalen oder Einfachdatei-Lookup-Quelle konfigurieren. Aktivieren Sie das Lookup-Caching in einer großen Lookup-Tabelle oder -Datei, um die Lookup-Leistung zu erhöhen.

Der Integrationsdienst erstellt einen Cache im Speicher, wenn er die erste Zeile der Daten in einer zwischengespeicherten Lookup-Umwandlung verarbeitet. Der Integrationsdienst erstellt Caches, wenn die Quellzeilen die Lookup-Umwandlung eingeben. Er weist dem Cache Speicher basierend auf der Menge zu, die Sie in der Umwandlung konfigurieren. Der Integrationsdienst speichert Bedingungswerte im Index-Cache und Ausgabewerte im Daten-Cache. Der Integrationsdienst fragt den Cache für jede Zeile ab, die die Umwandlung eingibt.

Wenn die Daten nicht dem Speichercache entsprechen, speichert der Integrationsdienst die überzähligen Werte in Cache-Dateien. Der Integrationsdienst erstellt die Cache-Dateien im Cache-Verzeichnis. Standardmäßig erstellt der Integrationsdienst die Cache-Dateien in dem Verzeichnis, das im Systemparameter CacheDir angegeben ist. Wenn das Mapping abgeschlossen ist, gibt der Integrationsdienst Cachespeicher frei und löscht die Cache-Dateien, es sei denn, Sie konfigurieren die Lookup-Umwandlung zur Verwendung eines persistenten Caches.

Wenn Sie ein Einfachdatei-Lookup verwenden, speichert der Integrationsdienst die Lookup-Quelle zwischen. Wenn Sie ein Einfachdatei-Lookup für die sortierte Eingabe konfigurieren, kann der Integrationsdienst das Lookup nicht zwischenspeichern, wenn die Bedingungsspalten nicht gruppiert sind. Wenn die Spalten gruppiert, aber nicht sortiert sind, verarbeitet der Integrationsdienst das Lookup so, als hätten Sie die sortierte Eingabe nicht konfiguriert.

Wenn Sie die Lookup-Umwandlung nicht für das Caching konfigurieren, fragt der Integrationsdienst die Lookup-Quelle für jede Eingabezeile ab. Das Ergebnis der Lookup-Abfrage und -Verarbeitung ist unabhängig davon, ob Sie die Lookup-Quelle zwischenspeichern, dasselbe. Sie können jedoch die Lookup-Leistung für eine große Lookup-Quelle erhöhen, wenn Sie das Lookup-Caching aktivieren.

Lookup-Cache-Typen

Sie können verschiedene Typen von Lookup-Caches konfigurieren. Beispielsweise können Sie einen gemeinsam genutzten Cache konfigurieren, wenn Sie den Cache in mehreren Lookup-Umwandlungen in demselben Mapping gemeinsam nutzen möchten.

Sie können die folgenden Typen von Lookup-Caches konfigurieren:

Statischer Cache

Während der Integrationsdienst das Lookup verarbeitet, ändert sich der statische Cache nicht. Der Integrationsdienst erstellt einen statischen Cache jedes Mal neu, wenn er das Lookup verarbeitet. Standardmäßig erstellt der Integrationsdienst einen statischen Cache, wenn Sie Caching für eine Lookup-Umwandlung konfigurieren. Beim Verarbeiten der ersten Lookup-Anfrage erstellt der Integrationsdienst den Cache. Er sucht im Cache Werte für jede Zeile, die in der Lookup-Umwandlung einbezogen wird. Wenn die Lookup-Bedingung auf TRUE festgelegt ist, gibt der Integrationsdienst einen Wert aus dem Lookup-Cache zurück.

Verwenden Sie einen statischen Cache aus folgenden Gründen:

- Die Lookup-Quelle ändert sich nicht während der Ausführung des Mappings.
- Das Lookup ist ein nicht verbundenes Lookup. Sie müssen einen statischen Cache für ein nicht verbundenes Lookup verwenden.
- Zum Erhöhen der Leistung. Da der Integrationsdienst den Cache während der Verarbeitung der Lookup-Umwandlung nicht aktualisiert, verarbeitet der Integrationsdienst eine Lookup-Umwandlung mit einem statischen Cache schneller als eine Lookup-Umwandlung mit einem dynamischen Cache.
- Der Integrationsdienst soll den Standardwert für verbundene Umwandlungen oder NULL für nicht verbundene Umwandlungen zurückgeben, wenn die Lookup-Bedingung auf FALSE festgelegt ist.

Persistenter Cache

Während der Integrationsdienst das Lookup verarbeitet, ändert sich der persistente Cache nicht. Der Integrationsdienst speichert die Lookup-Cache-Dateien und verwendet sie wieder, wenn er das nächste Mal eine Lookup-Umwandlung zur Verwendung des Caches konfiguriert. Verwenden Sie einen persistenten Cache, wenn sich die Lookup-Quelle nicht ändert.

Sie können die Lookup-Umwandlung konfigurieren, um einen persistenten Lookup-Cache erneut zu erstellen.

Dynamischer Cache

Wenn der Integrationsdienst das Lookup verarbeitet, ändert sich der dynamische Lookup-Cache. Beim Verarbeiten der ersten Lookup-Anfrage erstellt der Integrationsdienst einen dynamischen Lookup-Cache. Beim Verarbeiten jeder Zeile fügt der Integrationsdienst Daten dynamisch im Lookup-Cache ein bzw. aktualisiert sie und übergibt die Daten an das Ziel. Der dynamische Cache wird mit dem Ziel synchronisiert.

Verwenden Sie einen dynamischen Cache, wenn Sie ein Ziel basierend auf neuen und geänderten Datensätze aktualisieren möchten. Sie können einen dynamischen Cache auch verwenden, wenn das Mapping ein Lookup für Zieldaten erfordert, die Verbindung zum Ziel jedoch langsam ist.

Gemeinsam genutzter Cache

Ein gemeinsam genutzter Cache kann von mehreren Lookup-Umwandlungen in demselben Mapping verwendet werden. Verwenden Sie einen gemeinsam genutzten Cache, um die Mapping-Leistung zu erhöhen. Anstatt einen separaten Lookup-Cache für jede Lookup-Umwandlung zu generieren, generiert der Integrationsdienst einen Cache.

Nicht zwischengespeichertes Lookup

Es liegt ein nicht zwischengespeichertes Lookup vor, wenn der Integrationsdienst die Lookup-Quelle nicht zwischenspeichert. Standardmäßig verwendet der Integrationsdienst keinen Lookup-Cache für eine Lookup-Umwandlung.

Der Integrationsdienst verarbeitet ein nicht zwischengespeichertes Lookup auf die gleiche Weise wie er ein zwischengespeichertes Lookup verarbeitet, außer dass er die Lookup-Quelle abfragt, anstatt den Lookup-Cache zu erstellen und abzufragen.

Wenn die Lookup-Bedingung auf TRUE festgelegt ist, gibt der Integrationsdienst die Werte aus der Lookup-Quelle zurück. Wenn der Integrationsdienst eine verbundene Lookup-Umwandlung verarbeitet, gibt er die Werte zurück, die von den Lookup- bzw. Ausgabeports dargestellt werden. Der Integrationsdienst gibt den durch den Rückgabeport dargestellten Wert beim Verarbeiten einer nicht verbundenen Lookup-Umwandlung zurück.

Wenn die Bedingung nicht auf TRUE festgelegt ist, gibt der Integrationsdienst entweder NULL oder Standardwerte zurück. Der Integrationsdienst gibt den Standardwert des Ausgabeports beim Verarbeiten einer verbundenen Lookup-Umwandlung zurück, wenn die Bedingung nicht erfüllt ist. Der Integrationsdienst gibt beim Verarbeiten einer nicht verbundenen Lookup-Umwandlung NULL zurück, wenn die Bedingung nicht erfüllt ist.

Statischer Lookup-Cache

Ein statischer Lookup-Cache ist ein Cache, den der Integrationsdienst beim Verarbeiten der Lookup-Umwandlung nicht aktualisiert. Standardmäßig erstellt der Integrationsdienst einen statischen Lookup-Cache, wenn Sie eine Lookup-Umwandlung für das Caching konfigurieren.

Beim Verarbeiten der ersten Lookup-Anfrage erstellt der Integrationsdienst den Cache. Er fragt den Cache basierend auf der Lookup-Bedingung für jede Zeile ab, die an die Umwandlung übergeben wird.

Wenn die Lookup-Bedingung auf TRUE festgelegt ist, gibt der Integrationsdienst die Werte aus dem statischen Lookup-Cache zurück. Wenn der Integrationsdienst eine verbundene Lookup-Umwandlung verarbeitet, gibt er die Werte zurück, die von den Lookup- bzw. Ausgabeports dargestellt werden. Der Integrationsdienst gibt den durch den Rückgabeport dargestellten Wert beim Verarbeiten einer nicht verbundenen Lookup-Umwandlung zurück.

Wenn die Bedingung nicht auf TRUE festgelegt ist, gibt der Integrationsdienst entweder NULL oder Standardwerte zurück. Der Integrationsdienst gibt den Standardwert des Ausgabeports beim Verarbeiten einer verbundenen Lookup-Umwandlung zurück, wenn die Bedingung nicht erfüllt ist. Der Integrationsdienst gibt beim Verarbeiten einer nicht verbundenen Lookup-Umwandlung NULL zurück, wenn die Bedingung nicht erfüllt ist.

Persistenter Lookup-Cache

Ein persistenter Lookup-Cache ist ein Cache, den der Integrationsdienst für mehrere Ausführungen desselben Mappings wiederverwendet. Verwenden Sie einen persistenten Lookup-Cache, wenn sich die Lookup-Quelle nicht zwischen Mapping-Ausführungen ändert.

Standardmäßig verwendet der Integrationsdienst einen nicht persistenten Cache, wenn Sie das Lookup-Caching in einer Lookup-Umwandlung aktivieren. Der Integrationsdienst löscht die Cache-Dateien nach Abschluss des Mappings. Wenn Sie das Mapping das nächste Mal ausführen, erstellt der Integrationsdienst den Speicher-Cache aus der Lookup-Quelle.

Wenn Sie die Lookup-Umwandlung konfigurieren, um einen persistenten Lookup-Cache zu verwenden, speichert und verwendet der Integrationsdienst die Cache-Dateien für mehrere Ausführungen des Mappings wieder. Ein persistenter Cache beseitigt die erforderliche Zeit, um die Lookup-Tabelle zu lesen und den Lookup-Cache erneut zu erstellen.

Wenn der Integrationsdienst zum ersten Mal ein Mapping mit einem persistenten Lookup-Cache ausführt, speichert der Integrationsdienst die Cache-Dateien auf der Festplatte. Wenn der Integrationsdienst das Mapping das nächste Mal ausführt, erstellt er den Speicher-Cache aus den Cache-Dateien.

Sie können den Integrationsdienst so konfigurieren, dass ein persistenter Lookup-Cache neu erstellt wird, wenn sich die ursprüngliche Lookup-Quelle ändert. Wenn Sie einen Cache erneut erstellen, erstellt der Integrationsdienst neue Cache-Dateien und schreibt eine Nachricht an den Integrationsdienst.

Neuerstellen eines persistenten Lookup-Caches

Sie können den Integrationsdienst so konfigurieren, dass er einen persistenten Lookup-Cache neu erstellt. In einigen Fällen erstellt der Integrationsdienst auch dann einen persistenten Lookup-Cache, wenn Sie ihn nicht entsprechend konfiguriert haben.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie einen persistenten Lookup-Cache erneut erstellen:

- Erstellen Sie einen persistenten Lookup-Cache erneut, wenn sich die Lookup-Quelle geändert hat, seitdem der Integrationsdienst den Cache zuletzt erstellt hat.
- Sie können den Cache erneut erstellen, wenn das Mapping eine oder mehrere Lookup-Umwandlungen enthält, die einen Cache gemeinsam nutzen.
- Wenn sich die Lookup-Tabelle zwischen Mapping-Ausführungen nicht ändert, konfigurieren Sie die Lookup-Umwandlung, um einen persistenten Lookup-Cache zu verwenden. Der Integrationsdienst speichert Cache-Dateien und verwendet sie wieder, wodurch die für das Lesen der Lookup-Tabelle erforderliche Zeit wegfällt.
- Wenn Sie folgende Lookup-Umwandlungen konfigurieren, um den Lookup-Cache erneut zu erstellen, nutzt der Integrationsdienst den Cache gemeinsam, anstatt den Cache bei der Verarbeitung der folgenden Lookup-Umwandlung erneut zu erstellen.
- Wenn ein Mapping zwei persistente Lookups enthält und Sie die zweite Lookup-Umwandlung zum erneuten Erstellen des Caches konfigurieren, erstellt der Integrationsdienst den persistenten Lookup-Cache für beide erneut.

Der Integrationsdienst erstellt den persistenten Lookup-Cache in den folgenden Szenarien erneut:

- Der Integrationsdienst kann die Cache-Dateien nicht finden.
- Der Integrationsdienst kann den Cache nicht wiederverwenden. In dieser Instanz wird der Lookup-Cache entweder neu erstellt oder das Mapping schlägt fehl.
- Sie aktivieren oder deaktivieren die hohe Genauigkeit für den Integrationsdienst.

- Sie bearbeiten die Lookup-Umwandlung oder das Mapping.

  Hinweis: Der Integrationsdienst erstellt den Cache nicht erneut, wenn Sie die Umwandlungsbeschreibung bearbeiten.
- Sie ändern die Anzahl der Partitionen.
- Sie ändern die Datenbankverbindung oder den Dateispeicherort, der für den Zugriff auf die Lookup-Quelle verwendet wird.
- Sie ändern die Sortierreihenfolge im Unicode-Modus.
- Sie ändern die Integrationsdienst-Codepage.

Dynamischer Lookup-Cache

Ein dynamischer Cache ist ein Cache, den der Integrationsdienst beim Verarbeiten jeder Zeile aktualisiert. Verwenden Sie einen dynamischen Lookup-Cache, um die Synchronisierung des Caches mit dem Ziel beizubehalten.

Sie können einen dynamischen Cache mit einem relationalen Lookup und einem Einfachdatei-Lookup verwenden. Beim Verarbeiten der ersten Lookup-Anfrage erstellt der Integrationsdienst den Cache. Er fragt den Cache basierend auf der Lookup-Bedingung für jede Zeile ab, die an die Lookup-Umwandlung übergeben wird. Beim Verarbeiten jeder Zeile aktualisiert der Integrationsdienst den Lookup-Cache.

Basierend auf den Ergebnissen der Lookup-Abfrage, dem Zeilentyp und den Lookup-Umwandlungseigenschaften aktualisiert oder fügt der Integrationsdienst Zeilen in den Cache ein bzw. nimmt keine Änderungen am Cache vor.

Gemeinsam genutzter Lookup-Cache

Ein gemeinsam genutzter Cache ist ein statischer Lookup-Cache, der von mehreren Lookup-Umwandlungen in einem Mapping genutzt wird. Verwenden Sie einen gemeinsam genutzten Lookup-Cache, um die erforderliche Zeit zum Erstellen des Caches zu verringern.

Standardmäßig nutzt der Integrationsdienst den Cache für Lookup-Umwandlungen in einem Mapping gemeinsam, die über kompatible Caching-Strukturen verfügen. Wenn ein Mapping beispielsweise über zwei Instanzen derselben wiederverwendbaren Lookup-Umwandlung verfügt und Sie verwenden dieselben Ausgabeports für beide Instanzen, nutzen die Lookup-Umwandlungen standardmäßig den Lookup-Cache gemeinsam.

Beim Verarbeiten der ersten Lookup-Umwandlung erstellt der Integrationsdienst den Cache. Er verwendet denselben Cache, um folgende Lookup-Umwandlungen zu verarbeiten, die den Cache gemeinsam nutzen. Wenn der Integrationsdienst ein Lookup-Cache gemeinsam nutzt, schreibt er eine Meldung an das Integrationsdienstprotokoll.

Der Integrationsdienst ordnet den Daten-Cachespeicher und Index-Cachespeicher für die erste Lookup-Umwandlung zu. Er ordnet keinen zusätzlichen Speicher für folgende Lookup-Umwandlungen zu, die den Lookup-Cache gemeinsam nutzen.

Wenn die Umwandlung oder die Cache-Struktur keine gemeinsame Nutzung zulässt, erstellt der Integrationsdienst einen neuen Cache.

Regeln und Richtlinien für die gemeinsame Nutzung eines Lookup-Caches

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie einen Lookup-Cache gemeinsam verwenden:

- Sie können einen oder mehrere statischen Caches mit einem dynamischen Lookup gemeinsam verwenden. Wenn ein dynamisches Lookup den Cache mit einem statischen Lookup in demselben Mapping gemeinsam nutzt, verwendet das statische Lookup den Cache wieder, der vom dynamischen Lookup erstellt wurde.
- Sie können einen Cache nicht für dynamische Lookups gemeinsam nutzen.
- Wenn Sie mehrere Lookup-Umwandlungen konfigurieren, um einen persistenten Lookup-Cache erneut zu erstellen, erstellt der Integrationsdienst den Cache für die erste Lookup-Umwandlung und nutzt dann den persistenten Lookup-Cache für die folgenden Lookup-Umwandlungen gemeinsam.
- Wenn Sie die erste Lookup-Umwandlung nicht konfigurieren, um einen persistenten Lookup-Cache erneut zu erstellen, aber eine folgende Lookup-Umwandlung zum erneuten Erstellen des Caches konfigurieren, können die Umwandlungen den Cache nicht gemeinsam nutzen. Beim Verarbeiten jeder Lookup-Umwandlung erstellt der Integrationsdienst den Cache.
- Die Lookup-/Ausgabeports für folgende Lookup-Umwandlungen müssen übereinstimmen oder eine Teilmenge der Ports in der Lookup-Umwandlung sein, die der Integrationsdienst zum Erstellen des Caches verwendet. Die Reihenfolge der Ports muss nicht übereinstimmen.
- Lookup-Umwandlungen, die den Cache gemeinsam nutzen, müssen folgende Merkmale aufweisen:
  - Die Lookup-Umwandlungen müssen dieselben Ports in der Lookup-Bedingung verwenden.
  - Die Lookup-Umwandlungen müssen dieselbe SQL-Überschreibung verwenden, wenn eine verwendet wird.
  - Das Lookup-Caching muss in allen Lookup-Umwandlungen aktiviert werden.
  - Die Lookup-Umwandlungen müssen denselben Typ der Lookup-Quelle verwenden.
  - Alle relationalen Lookup-Umwandlungen müssen dieselbe Datenbankverbindung verwenden.
  - Die Lookup-Umwandlungen müssen denselben Lookup-Tabellennamen verwenden.
  - Die Struktur des Caches muss für alle Lookup-Umwandlungen kompatibel sein.

Cache-Vergleich

Der Integrationsdienst wird basierend auf dem Typ des Lookup-Caches, den Sie konfigurieren, anders ausgeführt.

Die folgende Tabelle vergleicht die Lookup-Umwandlungen mit einem nicht zwischengespeicherten Lookup, einem statischen Cache und einem dynamischen Cache:

Nicht zwischengespeichert	Statischer Cache	Dynamischer Cache
Der Integrationsdienst fügt den Cache nicht ein bzw. aktualisiert ihn nicht.	Der Integrationsdienst fügt den Cache nicht ein bzw. aktualisiert ihn nicht.	Der Integrationsdienst kann Zeilen im Cache einfügen oder aktualisieren, wenn er Zeilen an das Ziel übergibt.
Sie können ein relationales Lookup verwenden.	Sie können ein relationales bzw. Einfachdatei-Lookup verwenden.	Sie können ein relationales bzw. Einfachdatei-Lookup verwenden.
Wenn die Bedingung auf TRUE festgelegt ist, gibt der Integrationsdienst einen Wert aus der Lookup-Tabelle oder dem Lookup-Cache zurück. Wenn die Bedingung nicht auf TRUE festgelegt ist, gibt der Integrationsdienst den Standardwert für verbundene Umwandlungen und NULL für nicht verbundene Umwandlungen zurück.	Wenn die Bedingung auf TRUE festgelegt ist, gibt der Integrationsdienst einen Wert aus der Lookup-Tabelle oder dem Lookup-Cache zurück. Wenn die Bedingung nicht auf TRUE festgelegt ist, gibt der Integrationsdienst den Standardwert für verbundene Umwandlungen und NULL für nicht verbundene Umwandlungen zurück.	Wenn die Bedingung auf TRUE festgelegt ist, aktualisiert der Integrationsdienst die Zeilen entweder im Cache oder lässt den Cache je nach Zeilentyp unverändert. Dies zeigt an, dass die Zeile im Cache und in der Zieltabelle vorhanden ist. Sie können die aktualisierten Zeilen an ein Ziel übergeben. Wenn die Bedingung nicht auf TRUE festgelegt ist, fügt der Integrationsdienst die Zeilen entweder in den Cache ein oder lässt den Cache je nach Zeilentyp unverändert. Dies zeigt an, dass die Zeile nicht im Cache bzw. Ziel vorhanden ist. Sie können eingefügte Zeilen an eine Zieltabelle übergeben.

Cache-Partitionierung für Lookups

Bei der Cache-Partitionierung wird ein separater Cache für jede Partition erstellt, die eine Aggregator-, Joiner-, Rang- oder Lookup-Umwandlung verarbeitet. Die Cache-Partitionierung erhöht die Mapping-Leistung, da jede Partition parallel einen separaten Cache abfragt.

Wenn der Integrationsdienst Partitionen für ein Mapping erstellt, verwendet er möglicherweise die Cache-Partitionierung für partitionierte Lookup-Umwandlungen.

Der Integrationsdienst verwendet die Cache-Partitionierung für verbundene Lookup-Umwandlungen, wenn die folgenden Bedingungen erfüllt sind:

- Die Lookup-Bedingung enthält nur Gleichheitsoperatoren.
- Wenn die verbundene Lookup-Umwandlung Daten in einer relationalen Tabelle sucht, ist die Datenbank für die Unterscheidung nach Groß- und Kleinschreibung konfiguriert.

Der Integrationsdienst verwendet keine Cache-Partitionierung für nicht verbundene Lookup-Umwandlungen.

Wenn der Integrationsdienst keine Cache-Partitionierung für eine Lookup-Umwandlung verwendet, nutzen alle Partitionen der Lookup-Umwandlung denselben Cache gemeinsam. Jede Partition fragt denselben Cache seriell ab.

KAPITEL 25

Dynamischer Lookup-Cache

Dieses Kapitel umfasst die folgenden Themen:

- Dynamischer Lookup-Cache – Übersicht, 389
- Verwendung für einen dynamischen Lookup-Cache, 390
- Dynamische Lookup-Cache-Eigenschaften, 390
- Dynamischer Lookup-Cache und Ausgabewerte, 392
- Lookup-Umwandlungswerte, 392
- SQL-Überschreibung und dynamischer Lookup-Cache, 395
- Mapping-Konfiguration für einen dynamischen Lookup-Cache, 396
- Konditionale dynamische Lookup-Cache-Updates, 399
- Dynamisches Cache-Update mit Ausdrucksergebnissen, 400
- Beispiel für einen dynamischen Lookup-Cache, 401
- Regeln und Richtlinien für dynamische Lookup-Caches, 402

Dynamischer Lookup-Cache – Übersicht

Verwenden Sie einen dynamischen Lookup-Cache, um die Synchronisierung des Caches mit dem Ziel beizubehalten. Sie können einen dynamischen Cache mit einem relationalen Lookup bzw. einem Einfachdatei-Lookup verwenden.

Beim Verarbeiten der ersten Lookup-Anfrage erstellt der Integrationsdienst den dynamischen Lookup-Cache. Er fragt den Cache basierend auf der Lookup-Bedingung für jede Zeile ab, die an die Umwandlung übergeben wird. Beim Verarbeiten jeder Zeile aktualisiert der Integrationsdienst den Lookup-Cache.

Basierend auf den Ergebnissen der Lookup-Abfrage, dem Zeilentyp und den Lookup-Umwandlungseigenschaften führt der Integrationsdienst eine der folgenden Aktionen im dynamischen Lookup-Cache aus, wenn er eine Zeile aus der Quelle liest:

Fügt die Zeile in den Cache ein

Der Integrationsdienst fügt die Zeile ein, wenn sich die Zeile nicht im Cache befindet und Sie die Lookup-Umwandlung konfiguriert haben, um Zeilen in den Cache einzufügen. Sie können die Umwandlung zum Einfügen von Zeilen in den Cache basierend auf Eingabeports bzw. generierten Sequenz-IDs konfigurieren. Der Integrationsdienst kennzeichnet die Zeile als „Einfügen“.

Aktualisiert die Zeile im Cache

Der Integrationsdienst aktualisiert die Zeile, wenn sich die Zeile nicht im Cache befindet und Sie die Lookup-Umwandlung zum Aktualisieren von Zeilen im Cache konfiguriert haben. Der Integrationsdienst

aktualisiert die Zeile im Cache basierend auf den Eingabeports. Der Integrationsdienst kennzeichnet die Zeile als Aktualisierungszeile.

Nimmt keine Änderung am Cache vor

Der Integrationsdienst nimmt keine Änderungen vor, wenn sich die Zeile im Cache befindet und Sie die Lookup-Umwandlung so konfiguriert haben, dass nur neue Zeilen eingefügt werden. Oder die Zeile befindet sich nicht im Cache und Sie haben angegeben, nur vorhandene Zeilen zu aktualisieren. Oder die Zeile befindet sich im Cache, basierend auf der Lookup-Bedingung ändert sich jedoch nichts. Der Integrationsdienst kennzeichnet die Zeile als unverändert.

Basierend auf dem Wert von NewLookupRow können Sie auch eine Router- bzw. Filterumwandlung mit der dynamischen Lookup-Umwandlung konfigurieren, um Einfüge- oder Aktualisierungszeilen an die Zieltabelle weiterzuleiten. Sie können unveränderte Zeilen an eine andere Zieltabelle bzw. Einfachdatei weiterleiten oder sie ablegen.

Verwendung für einen dynamischen Lookup-Cache

Sie können eine Lookup-Umwandlung mit einem dynamischen Lookup-Cache konfigurieren, um den Cache basierend auf Änderungen in der Lookup-Quelle zu aktualisieren.

Sie können einen dynamischen Lookup-Cache aus folgenden Gründen verwenden:

Aktualisieren einer Master-Kundentabelle mit neuen und aktualisierten Kundeninformationen

Sie können beispielsweise eine Lookup-Umwandlung zum Ausführen eines Lookups in der Kundentabelle verwenden, um zu bestimmen, ob ein Kunde im Ziel vorhanden ist. Der Cache stellt die Kundentabelle dar. Die Lookup-Umwandlung fügt Zeilen im Cache ein und aktualisiert sie, wenn sie Zeilen an das Ziel übergibt.

Verwenden Sie als Lookup-Quelle nicht eine relationale Tabelle, sondern eine exportierte Einfachdatei.

Wenn die Verbindung zur Datenbank langsam ist, können Sie die relationalen Tabelleninhalte in eine Einfachdatei exportieren und die Datei als Lookup-Quelle verwenden. Sie müssen diese Methode beispielsweise verwenden, wenn eine ODBC-Verbindung zu einer Datenbank langsam ist. Sie können die Datenbanktabelle als relationales Ziel im Mapping konfigurieren und die Lookup-Cache-Änderungen zur Datenbanktabelle zurückgeben.

Dynamische Lookup-Cache-Eigenschaften

Konfigurieren Sie die dynamischen Lookup-Eigenschaften, um einen dynamischen Lookup-Cache zu aktivieren und die Aktualisierung des Caches zu konfigurieren. Beispielsweise können Sie die Werte konfigurieren, die im dynamischen Cache eingefügt und aktualisiert werden.

Konfigurieren Sie die folgenden Eigenschaften, wenn Sie einen dynamischen Lookup-Cache aktivieren:

Bei mehreren Matches

Auf „Fehler melden“ festgelegt.

Dynamischer Lookup-Cache

Aktiviert einen dynamischen Lookup-Cache.

Diese Option ist nach der Aktivierung des Lookup-Caching verfügbar.

Aktualisieren, andernfalls einfügen

Gilt für Zeilen bei der Eingabe der Lookup-Umwandlung, in der der Zeilentyp „Aktualisieren“ ist. Bei entsprechender Aktivierung aktualisiert der Integrationsdienst vorhandene Zeilen im Cache und fügt eine Zeile ein, wenn sie neu ist. Bei entsprechender Aktivierung fügt der Integrationsdienst keine neuen Zeilen ein.

Diese Option ist nach der Aktivierung des dynamischen Caching verfügbar.

Einfügen, andernfalls aktualisieren

Gilt für Zeilen bei der Eingabe der Lookup-Umwandlung, in der der Zeilentyp „Einfügen“ ist. Bei entsprechender Aktivierung fügt der Integrationsdienst Zeilen im Cache ein und aktualisiert vorhandene Zeilen. Bei entsprechender Aktivierung aktualisiert der Integrationsdienst keine vorhandenen Zeilen.

Diese Option ist nach der Aktivierung des dynamischen Caching verfügbar.

Bei Aktualisierung alten Wert ausgeben

Die Lookup-Umwandlung kann die vorhandenen oder neuen Werte aus dem Cache ausgeben. Bei Aktivierung gibt der Integrationsdienst die vorhandenen Werte aus den Lookup- bzw. Ausgabeports aus, bevor er die Werte im Cache aktualisiert hat. Wenn der Integrationsdienst eine Zeile im Cache aktualisiert, gibt er den Wert im Lookup-Cache aus, bevor er die Zeile basierend auf den Eingabedaten aktualisiert hat. Wenn der Integrationsdienst eine Zeile im Cache einfügt, gibt er Nullwerte aus.

Deaktivieren Sie die Eigenschaft für den Integrationsdienst, um dieselben Werte aus den Lookup-/Ausgabeports und Eingabe-/Ausgabeports zu übergeben. Diese Eigenschaft ist standardmäßig aktiviert.

Diese Option ist nach der Aktivierung des dynamischen Caching verfügbar.

Dynamische Cache-Bedingung aktualisieren

Bei Aktivierung verwendet der Integrationsdienst den Bedingungsausdruck, um festzulegen, ob der dynamische Cache aktualisiert werden soll. Der Integrationsdienst aktualisiert den Cache, wenn die Bedingung auf TRUE festgelegt ist und die Daten im Cache vorhanden sind.

Erstellen Sie den Ausdruck mit Lookup-Ports oder Eingabeports. Der Ausdruck kann Eingabewerte oder Werte im Lookup-Cache enthalten. Standardwert ist TRUE.

Diese Option ist nach der Aktivierung des dynamischen Caching verfügbar.

NewLookupRow

Das Developer-Tool fügt diesen Port zu einer Lookup-Umwandlung hinzu, die mit einem dynamischen Cache konfiguriert ist.

Die NewLookupRow-Eigenschaft kann einen der folgenden Werte enthalten:

- 0 = Kein Update des Caches.
- 1 = Zeile in den Cache einfügen.
- 2 = Zeile im Cache aktualisieren.

Um die Synchronisierung des Lookup-Caches und der Zieltabelle beizubehalten, übergeben Sie Zeilen an das Ziel, wenn der NewLookupRow-Wert gleich 1 oder 2 ist.

Zugeordneter Port

Der Integrationsdienst verwendet den Wert des zugeordneten Ports, wenn er die Daten im Cache aktualisiert. Der Integrationsdienst ordnet die in der Lookup-Bedingung angegebenen Eingabeports und Lookup-Quellen-Ports zu. Sie müssen die zugeordneten Ports für die verbleibenden Lookup-Quellen-Ports im dynamischen Lookup konfigurieren. Wenn Sie einen zugeordneten Port für alle Lookup-Quellen-Ports in einem dynamischen Lookup nicht konfigurieren, schlägt die Zuordnungsvalidierung fehl.

Sie können einen Lookup-Quellen-Port mit den folgenden Objekten zuordnen:

Objekt	Beschreibung
Eingabeport	Aktualisiert den Cache basierend auf dem Wert eines Eingabeports.
Zugeordneter Ausdruck	Auswählen, um einen Ausdruck einzugeben. Der Integrationsdienst aktualisiert den Cache basierend auf dem Ergebnis des Ausdrucks.
Sequenz-ID	Generiert einen Primärschlüssel für Zeilen, die in den Lookup-Cache eingefügt wurden. Sie können eine Sequenz-ID nur mit bigint- und int-Spalten zuordnen.

Null-Eingaben bei Updates ignorieren

Das Developer-Tool aktiviert diese Port-Eigenschaft für Lookup-/Ausgabeports, wenn Sie die Lookup-Umwandlung für die Verwendung eines dynamischen Caches konfigurieren. Wählen Sie diese Eigenschaft aus, wenn der Integrationsdienst die Spalte im Cache nicht mit einem Null-Eingabewert aktualisieren soll.

In Vergleich ignorieren

Das Developer-Tool aktiviert diese Port-Eigenschaft für Lookup-/Ausgabeports, die nicht in der Lookup-Bedingung verwendet werden, wenn Sie die Lookup-Umwandlung für die Verwendung eines dynamischen Caches konfigurieren. Der Integrationsdienst vergleicht standardmäßig die Werte in allen Lookup-Ports mit den Werten in deren zugeordneten Ports. Wählen Sie diese Eigenschaft aus, wenn der Integrationsdienst beim Vergleichen der Werte vor der Aktualisierung einer Zeile den Port ignorieren soll. Mit dieser Eigenschaft verbessern Sie die Vergleichsleistung.

Dynamischer Lookup-Cache und Ausgabewerte

Wenn Sie einen dynamischen Lookup-Cache aktivieren, variieren die Ausgabeport-Werte je nach Konfiguration des dynamischen Lookup-Caches. Der Ausgabewert des Lookup-/Ausgabeports hängt davon ab, ob Sie alte oder neue Werte ausgeben möchten, wenn der Integrationsdienst eine Zeile aktualisiert.

Sie können die Eigenschaft **Bei Aktualisierung alten Wert ausgeben** konfigurieren, um einen der folgenden Typen von Ausgabewerten für einen Lookup-/Ausgabeport anzugeben:

- Bei Aktualisierung alte Werte ausgeben. Der Integrationsdienst gibt den Wert aus, der vor der Aktualisierung der Zeile im Cache vorhanden war.
- Bei Aktualisierung neue Werte ausgeben. Der Integrationsdienst gibt den aktualisierten Wert aus, den er in den Cache schreibt. Der Lookup-/Ausgabeport-Wert stimmt mit dem Ausgabeport-Wert überein.

Lookup-Umwandlungswerte

Die Lookup-Umwandlung enthält Werte für die Eingabeports sowie Lookup- und Ausgabeports. Wenn Sie einen dynamischen Lookup-Cache aktivieren, variieren die Ausgabeport-Werte je nach Konfiguration des dynamischen Lookup-Caches.

Die Lookup-Umwandlung enthält die folgenden Wertetypen:

Eingabewert

Wert, den der Integrationsdienst an die Lookup-Umwandlung übergibt.

Lookup-Wert

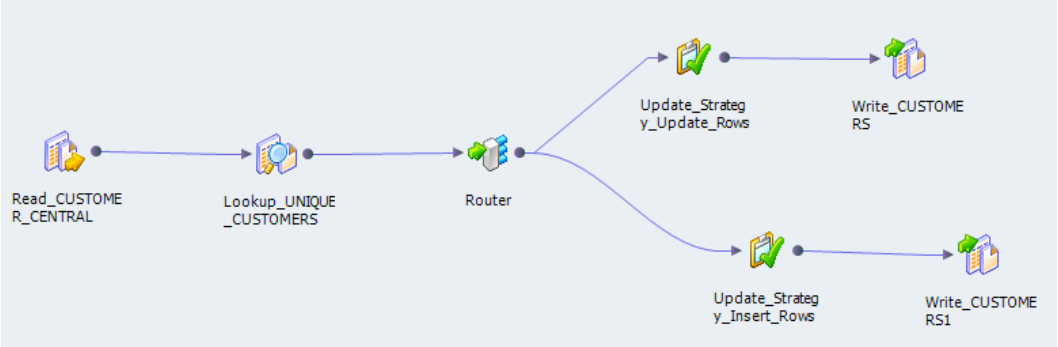
Wert, den der Integrationsdienst in den Cache einfügt.

Ausgabewert

Wert, den der Integrationsdienst aus dem Ausgabeport der Lookup-Umwandlung übergibt. Der Ausgabewert des Lookup-/Ausgabeports hängt davon ab, ob Sie alte oder neue Werte ausgeben möchten, wenn der Integrationsdienst eine Zeile aktualisiert.

Lookup-Umwandlungswerte – Beispiel

Beispielsweise können Sie ein Mapping mit den folgenden Objekten erstellen:



Für die Lookup-Umwandlung aktualisieren Sie dynamisches Lookup-Caching und definieren die folgende Lookup-Bedingung:

```
IN_CUST_ID = CUST_ID
```

Standardmäßig ist der Zeilentyp aller Zeilen bei der Eingabe der Lookup-Umwandlung „Einfügen“. Wählen Sie zum Einfügen und Aktualisieren im Cache und in der Zieltabelle die Eigenschaft **Einfügen, andernfalls aktualisieren** in der Lookup-Umwandlung aus.

Ursprüngliche Cache-Werte

Wenn Sie das Mapping ausführen, erstellt der Integrationsdienst den Lookup-Cache aus der Zieltabelle.

Die folgende Tabelle zeigt die ursprünglichen Werte des Lookup-Cache:

PK_PRIMARYKEY	CUST_ID	CUST_NAME	ADDRESS
100001	80001	Marion James	100 Main St.
100002	80002	Laura Jones	510 Broadway Ave.
100003	80003	Shelley Lau	220 Burnside Ave.

Eingabewerte

Die Quelle enthält Zeilen, die in der Zieltabelle vorhanden sind, und Zeilen, die nicht in der Zieltabelle vorhanden sind. Der Integrationsdienst übergibt die Quellzeilen an die Lookup-Umwandlung.

In der folgenden Tabelle werden die Quellzeilen angezeigt:

SQ_CUST_ID	SQ_CUST_NAME	SQ_ADDRESS
80001	Marion Atkins	100 Main St.
80002	Laura Gomez	510 Broadway Ave.
99001	Jon Freeman	555 6th Ave.

Lookup-Werte

Der Integrationsdienst sucht Werte im Cache basierend auf der Lookup-Bedingung. Er aktualisiert Zeilen im Cache für die vorhandenen Kunden-IDs 80001 und 80002. Er fügt eine Zeile in den Cache für die Kunden-ID 99001 ein. Der Integrationsdienst generiert einen neuen Schlüssel (PK_PRIMARYKEY) für die neue Zeile.

Die folgende Tabelle zeigt die Zeilen und Werte, die vom Lookup zurückgegeben wurden:

PK_PRIMARYKEY	CUST_ID	CUST_NAME	ADDRESS
100001	80001	Marion Atkins	100 Main St.
100002	80002	Laura Gomez	510 Broadway Ave.
100004	99001	Jon Freeman	555 6th Ave.

Ausgabewerte

Der Integrationsdienst kennzeichnet die Zeilen in der Lookup-Umwandlung basierend auf den Einfügungen und Aktualisierungen, die er im dynamischen Cache durchführt. Der Integrationsdienst übergibt die Zeilen möglichweise an eine Router-Umwandlung, die eine Verzweigung für Einfügezeilen und eine andere Verzweigung für Aktualisierungszeilen erstellt. Jede Verzweigung enthält eine Update-Strategie-Umwandlung. Die Update-Strategie-Umwandlung kennzeichnet die Zeilen für Einfügungen und Aktualisierungen basierend auf dem Wert des NewLookupRow-Ports.

Die Ausgabewerte der Lookup-/Ausgabe- und Eingabe-/Ausgabeports hängen davon ab, ob Sie alte oder neue Werte ausgeben möchten, wenn der Integrationsdienst eine Zeile aktualisiert. Die Ausgabewerte des NewLookupRow-Ports und jeder Lookup-/Ausgabeport, der die Sequenz-ID verwendet, sind für neue und aktualisierte Zeilen gleich.

Wenn Sie neue Werte ausgeben möchten, geben die Lookup-/Ausgabeports die folgenden Werte aus:

NewLookupRow	PK_PRIMARYKEY	CUST_ID	CUST_NAME	ADDRESS
2	100001	80001	Marion Atkins	100 Main St.
2	100002	80002	Laura Gomez	510 Broadway Ave.
1	100004	99001	Jon Freeman	555 6th Ave.

Wenn Sie alte Werte ausgeben möchten, geben die Lookup-/Ausgabeports die folgenden Werte aus:

NewLookupRow	PK_PRIMARYKEY	CUST_ID	CUST_NAME	ADDRESS
2	100001	80001	Marion James	100 Main St.
2	100002	80002	Laura Jones	510 Broadway Ave.
1	100004	99001	Jon Freeman	555 6th Ave.

Wenn der Integrationsdienst Zeilen im Lookup-Cache aktualisiert, verwendet er die Primärschlüsselwerte (PK_PRIMARYKEY) für Zeilen im Cache und in der Zieltabelle.

Der Integrationsdienst verwendet die Sequenz-ID zum Generieren eines Primärschlüssels für den Kunden, den er nicht im Cache findet. Der Integrationsdienst fügt den Primärschlüsselwert in den Lookup-Cache ein und gibt den Wert für den Lookup-/Ausgabeport zurück.

Die Integrationsdienst gibt die Werte aus den Eingabe-/Ausgabeports aus, die mit den Eingabewerten übereinstimmen.

Hinweis: Wenn der Eingabewert NULL ist und Sie die Eigenschaft „NULL ignorieren“ für den verbundenen Eingabeport auswählen, entspricht der Eingabewert nicht dem Suchwert oder dem Wert außerhalb des Eingabe-/Ausgabeports. Wenn Sie die Eigenschaft „Null ignorieren“ auswählen, wird die Synchronisierung des Lookup-Caches und der Zieltabelle möglicherweise aufgehoben, wenn Sie Nullwerte an das Ziel übergeben. Sie müssen sicherstellen, dass Sie Nullwerte nicht an das Ziel übergeben.

SQL-Überschreibung und dynamischer Lookup-Cache

Sie können eine WHERE-Klausel in der Lookup-Abfrage hinzufügen, um die Datensätze zu filtern, die zum Erstellen des Caches und zum Durchführen eines Lookups in der Datenbanktabelle für ein nicht zwischengespeichertes Lookup verwendet werden. Der Integrationsdienst verwendet die WHERE-Klausel jedoch nicht, wenn Zeilen in einen dynamischen Cache eingefügt werden.

Wenn Sie eine WHERE-Klausel in einer Lookup-Umwandlung bei der Verwendung eines dynamischen Caches hinzufügen, verbinden Sie eine Filterumwandlung vor der Lookup-Umwandlung, um die Zeilen zu filtern, die Sie nicht in den Cache oder in die Zieltabelle einfügen möchten. Wenn Sie die Filterumwandlung nicht hinzufügen, erhalten Sie möglicherweise inkonsistente Ergebnisse zwischen dem Cache und der Zieltabelle.

Beispiel: Sie können eine Lookup-Umwandlung konfigurieren, um ein dynamisches Lookup für die Mitarbeitertabelle, EMP und übereinstimmende Zeilen nach EMP_ID durchzuführen. Sie können die folgende Lookup-SQL-Überschreibung definieren:

```
SELECT EMP_ID, EMP_STATUS FROM EMP ORDER BY EMP_ID, EMP_STATUS WHERE EMP_STATUS = 4
```

Beim erstmaligen Ausführen von Mapping erstellt der Integrationsdienst den Lookup-Cache aus der Zieltabelle basierend auf der Lookup-SQL-Überschreibung. Alle Zeilen im Cache entsprechen der Bedingung `EMP_STATUS = 4` in der WHERE-Klausel.

Der Integrationsdienst beispielsweise liest eine Quellzeile, die die von Ihnen angegebene Lookup-Bedingung erfüllt, der Wert von EMP_STATUS ist jedoch 2. Obwohl das Ziel die Zeile enthalten kann, in der EMP_STATUS 2 beträgt, findet der Integrationsdienst aufgrund der SQL-Überschreibung die Zeile nicht im Cache. Der Integrationsdienst fügt die Zeile in den Cache ein und übergibt die Zeile an die Zieltabelle. Wenn der Integrationsdienst diese Zeile in die Zieltabelle einfügt, erhalten Sie möglicherweise inkonsistente Ergebnisse, wenn die Zeile bereits vorhanden ist. Zudem entsprechen nicht alle Zeilen im Cache der Bedingung in der WHERE-Klausel in der SQL-Überschreibung.

Um sicherzustellen, dass Sie nur Zeilen in den Cache einfügen, die mit der WHERE-Klausel übereinstimmen, fügen Sie vor der Lookup-Umwandlung eine Filterumwandlung hinzu und definieren Sie die Filterbedingung als Bedingung in der WHERE-Klausel in der Lookup-SQL-Überschreibung.

Geben Sie beim obigen Beispiel die folgende Filterbedingung in der Filterumwandlung und die WHERE-Klausel in der SQL Überschreibung ein:

```
EMP_STATUS = 4
```

Mapping-Konfiguration für einen dynamischen Lookup-Cache

Wenn Sie ein Lookup mit einem dynamischen Cache verwenden, müssen Sie das Mapping konfigurieren, um den dynamischen Lookup-Cache zu aktualisieren und die geänderten Zeilen in das Ziel zu schreiben.

Führen Sie die folgenden Schritte zum Konfigurieren eines Mappings mit einem dynamischen Lookup-Cache durch:

Kennzeichnen Sie die Eingabezeilen der Lookup-Umwandlung für das Einfügen bzw. Aktualisieren.

Standardmäßig ist der Zeilentyp aller Eingabezeilen „Einfügen“. Fügen Sie eine Update-Strategie-Umwandlung vor der Lookup-Umwandlung hinzu, um andere Zeilentypen für die Eingabezeilen anzugeben.

Geben Sie an, wie der Integrationsdienst die Eingabezeilen für den dynamischen Cache behandelt.

Wählen Sie die Optionen **Einfügen, andernfalls aktualisieren** oder **Aktualisieren, andernfalls einfügen** aus, um Zeilen zu verarbeiten, die zum Einfügen oder Aktualisieren gekennzeichnet sind.

Erstellen Sie separate Mapping-Pipelines für Zeilen, die in das Ziel eingefügt und im Ziel aktualisiert werden sollen.

Fügen Sie nach der Lookup-Umwandlung eine Filter- oder Routerumwandlung hinzu, um Einfüge- und Aktualisierungszeilen an separate Mapping-Verzweigungen weiterzuleiten. Verwenden Sie den Wert von NewLookupRow, um die entsprechende Verzweigung für jede Zeile zu bestimmen.

Konfigurieren Sie den Zeilentyp für Ausgabezeilen der Lookup-Umwandlung.

Fügen Sie eine Update-Strategie-Umwandlung hinzu, um Zeilen zum Einfügen oder Aktualisieren zu kennzeichnen.

Einfügen, andernfalls aktualisieren

Verwenden Sie die Eigenschaft **Einfügen, andernfalls aktualisieren**, um vorhandene Zeilen im dynamischen Lookup-Cache zu aktualisieren, wenn der Zeilentyp „Einfügen“ ist.

Diese Eigenschaft gilt nur für Zeilen bei der Eingabe der Lookup-Umwandlung, in der der Zeilentyp „Einfügen“ ist. Wenn eine Zeile an einen anderen Zeilentyp wie z. B. „Aktualisieren“ die Lookup-Umwandlung eingibt, hat die Eigenschaft **Einfügen, andernfalls aktualisieren** keinen Einfluss darauf, wie der Integrationsdienst die Zeile behandelt.

Wenn Sie **Einfügen, andernfalls aktualisieren** auswählen und der Zeilentyp bei der Eingabe der Lookup-Umwandlung „Einfügen“ ist, fügt der Integrationsdienst die Zeile in den Cache ein, wenn sie neu ist. Wenn die Zeile im Index-Cache vorhanden ist, der Daten-Cache sich aber von der aktuellen Zeile unterscheidet, aktualisiert der Integrationsdienst die Zeile im Daten-Cache.

Wenn Sie **Einfügen, andernfalls aktualisieren** nicht auswählen und der Zeilentyp bei der Eingabe der Lookup-Umwandlung „Einfügen“ ist, fügt der Integrationsdienst die Zeile in den Cache ein, wenn sie neu ist, und nimmt keine Änderungen am Cache vor, wenn die Zeile vorhanden ist.

Die folgende Tabelle beschreibt, wie der Integrationsdienst den Lookup-Cache ändert, wenn der Zeilentyp der Zeilen bei der Eingabe der Lookup-Umwandlung „Einfügen“ ist:

Option „Einfügen, andernfalls aktualisieren“	Zeile im Cache gefunden	Datencache unterscheidet sich	Lookup-Cache-Ergebnis	NewLookupRow-Wert
Gelöscht - nur einfügen	Ja	-	Keine Änderung	0
Gelöscht - nur einfügen	Nein	-	Einfügen	1
Ausgewählt	Ja	Ja	Aktualisieren	2[1]
Ausgewählt	Ja	Nein	Keine Änderung	0
Ausgewählt	Nein	-	Einfügen	1

1. Wenn Sie „Null ignorieren“ für alle Lookup-Ports auswählen, die sich nicht in der Lookup-Bedingung befinden, und wenn diese Ports Nullwerte enthalten, ändert der Integrationsdienst den Cache nicht und der NewLookupRow-Wert entspricht 0.

Aktualisieren, andernfalls einfügen

Verwenden Sie die Eigenschaft **Aktualisieren, andernfalls einfügen**, um neue Zeilen im dynamischen Lookup-Cache einzufügen, wenn der Zeilentyp „Aktualisieren“ ist.

Sie können die Eigenschaft **Aktualisieren, andernfalls einfügen** in der Lookup-Umwandlung auswählen. Diese Eigenschaft gilt nur für Zeilen bei der Eingabe der Lookup-Umwandlung, in der der Zeilentyp „Aktualisieren“ ist. Wenn eine Zeile an einen anderen Zeilentyp wie z. B. „Einfügen“ die Lookup-Umwandlung eingibt, hat diese Eigenschaft keinen Einfluss darauf, wie der Integrationsdienst die Zeile behandelt.

Wenn Sie diese Eigenschaft auswählen und der Zeilentyp bei der Eingabe der Lookup-Umwandlung „Aktualisieren“ ist, aktualisiert der Integrationsdienst die Zeile im Cache, wenn die Zeile im Index-Cache vorhanden ist und sich die Cache-Daten von der vorhandenen Zeile unterscheiden. Der Integrationsdienst fügt die Zeile im Cache ein, wenn sie neu ist.

Wenn Sie diese Eigenschaft und den Zeilentyp nicht bei der Eingabe der Lookup-Umwandlung auswählen, aktualisiert der Integrationsdienst die Zeile im Cache, wenn sie vorhanden ist, und nimmt keine Änderungen am Cache vor, wenn die Zeile neu ist.

Wenn Sie **Null ignorieren** für alle Lookup-Ports auswählen, die sich nicht in der Lookup-Bedingung befinden, und wenn diese Ports Nullwerte enthalten, ändert der Integrationsdienst den Cache nicht und der NewLookupRow-Wert entspricht 0.

Die folgende Tabelle beschreibt, wie der Integrationsdienst den Lookup-Cache ändert, wenn der Zeilentyp der Zeilen bei der Eingabe der Lookup-Umwandlung „Aktualisieren“ ist:

Option „Aktualisieren, andernfalls einfügen“	Zeile im Cache gefunden	Datencache unterscheidet sich	Lookup-Cache-Ergebnis	NewLookupRow-Wert
Gelöscht (nur aktualisieren)	Ja	Ja	Aktualisieren	2
Gelöscht (nur aktualisieren)	Ja	Nein	Keine Änderung	0
Gelöscht (nur aktualisieren)	Nein	-	Keine Änderung	0
Ausgewählt	Ja	Ja	Aktualisieren	2
Ausgewählt	Ja	Nein	Keine Änderung	0
Ausgewählt	Nein	-	Einfügen	1

Dynamischer Lookup-Cache und Ziel-Synchronisierung

Konfigurieren Sie nachgelagerte Umwandlungen, um sicherzustellen, dass der dynamische Lookup-Cache und das Ziel synchronisiert sind.

Wenn Sie einen dynamischen Lookup-Cache verwenden, schreibt der Integrationsdienst in den Lookup-Cache, bevor er in die Zieltabelle schreibt. Die Synchronisierung des Lookup-Caches und der Zieltabelle wird aufgehoben, wenn der Integrationsdienst die Daten nicht in das Ziel schreibt. Beispiel: Die Zieldatenbank weist die Daten möglicherweise zurück.

Beachten Sie die folgenden Richtlinien, um die Synchronisierung des Lookup-Caches mit der Lookup-Tabelle beizubehalten:

- Verwenden Sie eine Routerumwandlung, um Zeilen an das zwischengespeicherte Ziel zu übergeben, wenn der NewLookupRow-Wert gleich Eins oder Zwei ist.
- Verwenden Sie die Routerumwandlung, um Zeilen zu löschen, wenn der NewLookupRow-Wert gleich Null ist. Geben Sie die Zeilen alternativ in ein anderes Ziel aus.
- Verwenden Sie Update-Strategie-Umwandlungen nach der Lookup-Umwandlung, um Zeilen zum Einfügen bzw. Aktualisieren im Ziel zu markieren.
- Stellen Sie sicher, dass die Lookup-Umwandlung dieselben Werte an das Ziel ausgibt, die der Integrationsdienst in den Lookup-Cache schreibt. Wenn Sie bei der Aktualisierung neue Werte ausgeben, verbinden Sie nur Lookup-/Ausgabeports mit der Zieltabelle anstatt mit Ausgabe-Ports. Wenn Sie bei der Aktualisierung alte Werte ausgeben möchten, fügen Sie nach der Lookup-Umwandlung und vor der Routerumwandlung eine Ausdrucksumwandlung hinzu. Fügen Sie Ausgabeports in die Ausdrucksumwandlung für jeden Port in der Zieltabelle hinzu und erstellen Sie Ausdrücke, um sicherzustellen, dass Sie keine Null-Eingabewerte an das Ziel ausgeben.
- Wählen Sie „Einfügen und Aktualisieren“ aus, wenn Sie die Optionen der Update-Strategie-Zieltabelle definieren. Dies stellt sicher, dass der Integrationsdienst zum Aktualisieren gekennzeichnete Zeilen aktualisiert und zum Einfügen gekennzeichnete Zeilen einfügt.

Konditionale dynamische Lookup-Cache-Updates

Sie können den dynamischen Lookup-Cache basierend auf den Ergebnissen eines booleschen Ausdrucks aktualisieren. Der Integrationsdienst aktualisiert den Cache, wenn der Ausdruck auf TRUE festgelegt ist.

Beispiel: Sie verfügen möglicherweise über eine Produktanzahl-, Lagerbestand- und Zeitstempel-Spalte in einer Zieltabelle. Sie müssen den Lagerbestand mit den neuesten Quellwerten aktualisieren. Sie können den Lagerbestand aktualisieren, wenn die Quelldaten einen Zeitstempel größer als der Zeitstempel im dynamischen Cache enthalten. Erstellen Sie einen Ausdruck in der Lookup-Umwandlung ähnlich dem folgenden Ausdruck:

```
lookup_timestamp < input_timestamp
```

Der Ausdruck kann die Lookup- und Eingabeports enthalten. Sie können auf integrierte, Mapping- und Parameter-Variablen zugreifen. Sie können benutzerdefinierte Funktionen hinzufügen und sich auf nicht verbundene Umwandlungen beziehen.

Der Ausdruck gibt „TRUE“, „FALSE“ bzw. „NULL“ zurück. Wenn das Ergebnis des Ausdrucks „NULL“ ist, ist der Ausdruck „FALSE“. Der Integrationsdienst aktualisiert den Cache nicht. Zusätzlich können Sie eine Prüfung für NULL-Werte im Ausdruck durchführen, wenn Sie das Ausdrucksergebnis in „TRUE“ ändern. Der Standard-Ausdruckswert ist „TRUE“.

Konditionale dynamische Lookup-Cache-Verarbeitung

Sie können eine Bedingung erstellen, die bestimmt, ob der Integrationsdienst den dynamischen Lookup-Cache aktualisiert. Wenn die Bedingung auf FALSE oder NULL festgelegt ist, aktualisiert der Integrationsdienst den dynamischen Lookup-Cache nicht.

Wenn die Bedingung auf FALSE oder NULL festgelegt ist, ist der NewLookupRow-Wert unabhängig von den Lookup-Umwandlungseigenschaften Null und der Integrationsdienst aktualisiert den dynamischen Lookup-Cache nicht mit allen Zeileneinfügungen bzw. -aktualisierungen.

Wenn die Zeile im Cache vorhanden ist und sie „Einfügen, andernfalls aktualisieren“ bzw. „Aktualisieren, andernfalls einfügen“ aktivieren, ist der NewLookupRow-Wert 1 und der Integrationsdienst aktualisiert die neue Zeile im Cache.

Wenn die Zeile nicht im Cache vorhanden ist und sie „Einfügen, andernfalls aktualisieren“ bzw. „Aktualisieren, andernfalls einfügen“ aktivieren, ist der NewLookupRow-Wert 2 und der Integrationsdienst fügt die neue Zeile im Cache ein.

Konfigurieren eines konditionalen dynamischen Lookup-Cache

Sie können einen Ausdruck konfigurieren, der bestimmt, ob der Integrationsdienst einen dynamischen Lookup-Cache aktualisiert.

1. Erstellen Sie die Lookup-Umwandlung.
2. Wählen Sie auf der Registerkarte **Laufzeit** der Ansicht **Eigenschaften Lookup-Caching ist aktiviert** aus.
3. Wählen Sie auf der Registerkarte **Erweitert** der Ansicht **Eigenschaften Dynamischer Lookup-Cache** aus.
4. Um die Bedingung einzugeben, klicken Sie für die Eigenschaft **Dynamische Cache-Bedingung aktualisieren** auf den Pfeil nach unten.

   Der Ausdruckseditor wird angezeigt.
5. Definieren Sie eine Ausdrucksbedingung.

   Sie können Eingabeports, Lookup-Ports und Funktionen für den Ausdruck auswählen.
6. Klicken Sie auf **Validieren**, um sicherzustellen, dass der Ausdruck gültig ist.

7. Klicken Sie auf **OK**.
8. Konfigurieren Sie ggf. andere erweiterte Eigenschaften, die auf den dynamischen Lookup-Cache angewendet werden.

Dynamisches Cache-Update mit Ausdrucksergebnissen

Die Lookup-Umwandlung kann die dynamischen Lookup-Cache-Werte mit den Ergebnissen eines Ausdrucks aktualisieren.

Beispiel: Ein Produkt-Tabellenziel enthält eine numerische Spalte mit einer Anzahl an Aufträgen. Jedes Mal, wenn die Lookup-Umwandlung eine Reihenfolge für das Produkt erhält, aktualisiert er den dynamischen Cache order_count mit den Ergebnissen des folgenden Ausdrucks:

```
order_count = order_count + 1
```

Die Lookup-Umwandlung gibt order_count zurück.

Sie können konfigurieren, wie der Integrationsdienst die Groß- und Kleinschreibung behandelt, in der der Ausdruck den Wert Null ergibt.

Null-Ausdruckswerte

Der Ausdruck gibt NULL zurück, wenn einer der Werte im Ausdruck auf NULL festgelegt ist. Sie können jedoch einen Ausdruck konfigurieren, um einen Wert zurückzugeben, der nicht Null ist.

Wenn der Ausdruck auf einen Lookup-Port verweist, die Quelldaten aber neu sind, enthält der Lookup-Port einen Standardwert. Standardwert ist möglicherweise NULL. Sie können einen IsNull-Ausdruck für die Prüfung auf Nullwerte konfigurieren.

Der folgende Ausdruck prüft beispielsweise, ob lookup_column NULL ist:

```
iif (isnull(lookup_column), input_port, user_expression)
```

Wenn die Spalte Null ist, geben Sie den input_port-Wert zurück. Geben Sie andernfalls den Wert des Ausdrucks zurück.

Ausdrucksverarbeitung

Der Integrationsdienst kann Zeilen im dynamischen Lookup-Cache basierend auf Ausdrucksergebnissen einfügen und aktualisieren. Die Ausdrucksergebnisse variieren möglicherweise basierend darauf, ob der Lookup-Port-Wert auf NULL festgelegt ist und im Ausdruck enthalten ist.

Wenn Sie „Einfügen, andernfalls aktualisieren“ aktivieren, fügt der Integrationsdienst eine Zeile mit dem Ausdrucksergebnis ein, wenn die Daten nicht im Cache vorhanden sind. Der Lookup-Port-Wert ist NULL, wenn die Daten nicht im Cache vorhanden sind. Wenn sich der Ausdruck auf den Lookup-Port-Wert bezieht, ersetzt der Integrationsdienst den Standardportwert im Ausdruck. Wenn Sie „Einfügen, andernfalls aktualisieren“ aktivieren und die Daten im Cache vorhanden sind, aktualisiert der Integrationsdienst den Cache mit dem Ausdrucksergebnis.

Wenn Sie „Aktualisieren, andernfalls einfügen“ aktivieren, aktualisiert der Integrationsdienst den Cache mit dem Ausdrucksergebnis, wenn die Daten im Cache vorhanden sind. Wenn die Daten nicht im Cache vorhanden sind, fügt der Integrationsdienst eine Zeile ein, die das Ausdrucksergebnis enthält. Wenn sich der

Ausdruck auf einen Lookup-Port-Wert bezieht, ersetzt der Integrationsdienst den Standardportwert im Ausdruck.

Konfigurieren eines Ausdrucks für dynamische Cache-Updates

Sie können einen Ausdruck für ein dynamisches Cache-Lookup-Update konfigurieren.

Sie müssen die Lookup-Umwandlung zum Ausführen der dynamischen Lookups aktivieren, bevor Sie einen konditionalen Ausdruck erstellen können.

1. Erstellen Sie die Lookup-Umwandlung.
2. Wählen Sie auf der Registerkarte **Laufzeit** der Ansicht **Eigenschaften Lookup-Caching ist aktiviert** aus.
3. Wählen Sie auf der Registerkarte **Erweitert** der Ansicht **Eigenschaften Dynamischer Lookup-Cache** aus.
4. Konfigurieren Sie ggf. andere erweiterte Eigenschaften, die auf den dynamischen Lookup-Cache angewendet werden.
5. Um einen Ausdruck zu erstellen, wählen Sie die Registerkarte **Spalten** in der Ansicht **Eigenschaften** aus.
6. Klicken Sie in der Spalte **Zugeordneter Port** für die Lookup-Umwandlung, die Sie aktualisieren möchten, auf den Dropdownpfeil.
7. Wählen Sie **Zugeordneter Ausdruck** aus der Dropdown-Liste aus und klicken Sie auf die **EINGABETASTE**.

   Der Ausdruckseditor wird angezeigt.
8. Definieren Sie den Ausdruck.

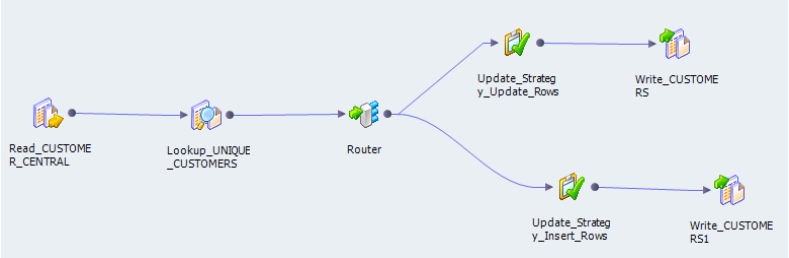
   Sie können Eingabeports, Lookup-Ports und Funktionen für den Ausdruck auswählen. Der Ausdrucks-Rückgabewert muss mit dem Datentyp des Lookup-Ports übereinstimmen.
9. Klicken Sie auf **Validieren**, um sicherzustellen, dass der Ausdruck gültig ist.
10. Klicken Sie auf **OK**.

Beispiel für einen dynamischen Lookup-Cache

Sie können einen dynamischen Lookup-Cache verwenden, um Zeilen im Ziel einzufügen und zu aktualisieren. Wenn Sie einen dynamischen Lookup-Cache verwenden, können Sie dieselben Zeilen im Cache einfügen und aktualisieren, die Sie im Ziel einfügen und aktualisieren.

Beispielsweise müssen Sie eine Tabelle aktualisieren, die Kundendaten enthält. Die Quelldaten enthalten Zeilen von im Ziel einzufügenden oder zu aktualisierenden Kundendaten. Erstellen Sie einen dynamischen Cache, der das Ziel darstellt. Konfigurieren Sie eine Lookup-Umwandlung, um Kunden im dynamischen Cache nachzuschlagen.

Die folgende Abbildung zeigt ein Mapping mit einer Lookup-Umwandlung, die einen dynamischen Lookup-Cache verwendet:



Die Routerumwandlung teilt sich in zwei Verzweigungen. Die Routerumwandlung übergibt Einfügezeilen an eine Verzweigung und Aktualisierungszeilen an eine andere Verzweigung. Jede Verzweigung enthält eine Update-Strategie-Umwandlung, die Zeilen in das Ziel schreibt. Beide Verzweigungen enthalten dasselbe Ziel.

Wenn das Mapping startet, erstellt der Integrationsdienst den Lookup-Cache aus der Kunden-Zieltabelle. Wenn der Integrationsdienst eine Zeile liest, die sich nicht im Lookup-Cache befindet, fügt er die Zeile im Cache ein.

Die Lookup-Umwandlung gibt jede Zeile in einer Routerumwandlung zurück. Basierend darauf, ob die Zeile zum Einfügen oder Aktualisieren markiert ist, leitet die Routerumwandlung die Zeile an die Update-Strategie-Umwandlungen. Die Routerumwandlung bestimmt, ob Zeilen basierend auf der NewLookupRow-Eigenschaft zum Einfügen oder Aktualisieren markiert sind. Die Update-Strategie-Umwandlungen markieren jede Zeile als Einfüge- oder Aktualisierungszeile, bevor sie an das Ziel übergeben werden.

Die Kunden-Zieltabelle ändert sich, wenn das Mapping ausgeführt wird. Der Integrationsdienst fügt neue Zeilen ein und aktualisiert vorhandene Zeilen im Lookup-Cache. Der Integrationsdienst behält die Synchronisierung des Lookup-Caches und der Kunden-Zieltabelle bei.

Verwenden Sie zum Generieren der Schlüssel für das Ziel die Sequenz-ID im zugeordneten Port. Der Integrationsdienst verwendet die Sequenz-ID als Primärschlüssel für jede in die Zieltabelle eingefügte neue Zeile.

Durch die Verwendung eines dynamischen Lookup-Caches erhöhen Sie die Sitzungsleistung, da Sie den Cache einmal aus der Datenbank erstellen.

Regeln und Richtlinien für dynamische Lookup-Caches

Beachten Sie beim Verwenden eines dynamischen Lookup-Caches die folgenden Richtlinien:

- Sie müssen die Eigenschaft **Bei mehreren Matches** auf „Fehler melden“ festlegen, wenn Sie einen dynamischen Lookup-Cache verwenden. Um die Eigenschaft zurückzusetzen, ändern Sie das dynamische Lookup in ein statisches Lookup, ändern Sie die Eigenschaft und ändern Sie dann das statische Lookup in ein dynamisches Lookup.
- Die gemeinsame Nutzung einer dynamischen und statischen Lookup-Umwandlung in derselben Gruppe mit der Reihenfolge der Ziellast ist nicht zulässig.
- Sie können einen dynamischen Lookup-Cache für ein relationales Lookup bzw. ein Einfachdatei-Lookup verwenden.
- Bei der Lookup-Umwandlung muss es sich um eine verbundene Umwandlung handeln.
- Sie können einen persistenten oder nicht persistenten Cache verwenden.
- Wenn der dynamische Cache nicht persistent ist, erstellt der Integrationsdienst den Cache aus der Datenbank immer neu, auch wenn Sie die Option **Aus Lookup-Quelle zwischenspeichern** nicht aktivieren.
- Sie können nur eine Gleichheits-Lookup-Bedingung erstellen. Sie können einen Datenbereich nicht in einem dynamischen Cache suchen.
- Sie müssen jeden Lookup-Port, der nicht in der Lookup-Bedingung vorhanden ist, einem Eingabeport, einer Sequenz-ID bzw. einem zugeordneten Ausdruck zuweisen.
- Verwenden Sie eine Routerumwandlung, um Zeilen an das zwischengespeicherte Ziel zu übergeben, wenn der NewLookupRow-Wert gleich Eins oder Zwei ist.

- Verwenden Sie die Routerumwandlung, um Zeilen zu löschen, wenn der NewLookupRow-Wert gleich Null ist. Alternativ können Sie die Zeilen an ein anderes Ziel ausgeben.
- Stellen Sie sicher, dass der Integrationsdienst am Ziel dieselben Werte ausgibt, die er in den Lookup-Cache schreibt. Wenn Sie bei der Aktualisierung neue Werte ausgeben, verbinden Sie nur Lookup-/Ausgabeports mit der Zieltabelle anstatt mit den Eingabe/Ausgabeports. Wenn Sie bei der Aktualisierung alte Werte ausgeben möchten, fügen Sie nach der Lookup-Umwandlung und vor der Routerumwandlung eine Ausdrucksumwandlung hinzu. Fügen Sie Ausgabeports in die Ausdrucksumwandlung für jeden Port in der Zieltabelle hinzu und erstellen Sie Ausdrücke, um sicherzustellen, dass Sie keine Null-Eingabewerte an das Ziel ausgeben.
- Wenn Sie eine Lookup-SQL-Überschreibung verwenden, ordnen Sie die richtigen Spalten den richtigen Zielen für das Lookup zu.
- Verwenden Sie beim Hinzufügen einer WHERE-Klausel zur Lookup-SQL-Überschreibung eine Filterumwandlung vor der Lookup-Umwandlung. Dies stellt sicher, dass der Integrationsdienst die Zeilen im dynamischen Cache und in der Zieltabelle einfügt, die mit der WHERE-Klausel übereinstimmen.
- Wenn Sie eine wiederverwendbare Lookup-Umwandlung zur Verwendung eines dynamischen Caches konfigurieren, können Sie die Bedingung nicht bearbeiten oder die Eigenschaft **Dynamischer Lookup-Cache** nicht in einem Mapping deaktivieren.
- Verwenden Sie Update-Strategie-Umwandlungen nach der Lookup-Umwandlung, um die Zeilen zum Einfügen bzw. Aktualisieren für das Ziel zu markieren.
- Verwenden Sie eine Update-Strategie-Umwandlung vor der Lookup-Umwandlung, um einige oder alle Zeilen als Aktualisierung zu definieren, wenn Sie die Eigenschaft „Aktualisieren, andernfalls einfügen“ in der Lookup-Umwandlung verwenden möchten.

KAPITEL 26

Match-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Match-Umwandlung - Übersicht, 404
- Match-Analyse, 405
- Übereinstimmungswert-Berechnungen, 409
- Analyse von Master-Daten, 413
- Identitätsvergleichsanalyse und persistente Indexdaten, 414
- Match-Mapping-Leistung, 415
- Match-Leistung in der Identitätsanalyse, 418
- Match-Umwandlungsansichten, 420
- Match-Umwandlungsports, 422
- Match-Mapplets, 426
- Konfigurieren eines Match-Analyse-Vorgangs, 428

Match-Umwandlung - Übersicht

Die Match-Umwandlung ist eine aktive Umwandlung, die den Ähnlichkeitsgrad zwischen Datensätzen analysiert. Verwenden Sie die Match-Umwandlung, um Datensätze ausfindig zu machen, die duplizierte Informationen in einem Dataset oder in zwei Datasets enthalten.

Die Match-Umwandlung analysiert die Werte in einem Eingabeport und generiert einen Satz von numerischen Scores, die den Grad der Ähnlichkeit zwischen den Werten darstellen. Sie können mehrere Ports zum Ermitteln des gesamten Ähnlichkeitsgrads zwischen den Eingabedatensätzen auswählen. Sie geben einen minimalen Score als Schwellenwert an, um die Datensätze zu identifizieren, die wahrscheinlich duplizierte Informationen enthalten.

Sie können die Match-Umwandlung in folgenden Datenprojekten verwenden:

- Customer Relationship Management. Beispiel: Ein Geschäft erstellt eine Mailaktion und muss die Kundendatenbank auf duplizierte Kundendatensätze überprüfen.
- Fusionen und Übernahmen. Beispiel: Eine Bank kauft eine andere Bank in derselben Region, und manche Kunden sind Kunde bei beiden Banken.
- Initiativen zur Einhaltung gesetzlicher Bestimmungen. Beispiel: Ein Unternehmen ist mit vom Staat oder der Industrie vorgegebenen Bestimmungen behaftet, die besagen, dass kein Datensystem duplizierte Datensätze enthalten darf.

- Finanzielles Risikomanagement. Beispiel: Eine Bank möchte Beziehungen zwischen Kontoinhabern ausfindig machen.
- Master Data Management Beispiel: Eine Einzelhandelskette hat eine Master-Datenbank mit Kundendatensätzen und jedes Einzelhandelsgeschäft in der Kette übermittelt regelmäßig Datensätze an die Master-Datenbank.
- Jedes Projekt, das duplizierte Datensätze identifizieren muss, ist ein Dataset.

Match-Analyse

Sie können verschiedene Arten der Duplikatsanalyse in der Match-Umwandlung definieren. Die Duplikatsanalyse-Vorgänge, die Sie definieren können, hängen von der Anzahl der Datenquellen im Mapping und vom Typ der Informationen ab, die in den Quellen enthalten sind.

Beachten Sie die folgenden Faktoren beim Konfigurieren einer Match-Umwandlung:

- Sie können eine einzelne Spalte oder mehrere Spalten aus einem Dataset auswählen.
- Sie können Spalten aus einer einzelnen Datenquelle oder zwei Datenquellen analysieren.
- Sie können die Match-Umwandlung zum Analysieren der Rohdaten in den Eingabeport-Feldern konfigurieren oder die Umwandlung zur Analyse der Identitätsinformationen in den Daten konfigurieren.
- Sie können die Match-Umwandlung zum Schreiben von verschiedenen Ausgabetypen konfigurieren. Der Typ der von Ihnen ausgewählten Ausgabe bestimmt die Anzahl der Datensätze, die die Umwandlung schreibt, sowie die Reihenfolge der Datensätze.
- Um die Leistung zu erhöhen, sortieren Sie die Eingabe-Datensätze in Gruppen, bevor Sie die Match-Analyse durchführen.

Spaltenanalyse

Beim Konfigurieren einer Vergleichsumwandlung wählen Sie eine oder mehrere Spalten für die Analyse aus.

Bei der Vergleichsumwandlung werden Spalten paarweise analysiert. Wenn Sie eine einzelne Spalte zur Analyse auswählen, erstellt die Umwandlung eine temporäre Kopie der Spalte und vergleicht die Quellspalte mit der temporären Spalte. Wenn Sie zwei Spalten zur Analyse auswählen, vergleicht die Umwandlung die Werte in den beiden von Ihnen ausgewählten Spalten. Die Umwandlung vergleicht alle Werte in einer Spalte mit allen Werten in der anderen Spalte. Die Umwandlung gibt einen Vergleichswert für jedes analysierte Wertepaar zurück.

Sie wählen die zu analysierenden Spalten aus, wenn Sie eine Strategie in einer Vergleichsumwandlung konfigurieren. Die Strategie gibt die zu analysierenden Spalten und den auf die Spalten anzuwendenden Algorithmus an. Der Algorithmus berechnet den Ähnlichkeitsgrad zwischen den Wertepaaren. Die unterschiedlichen Algorithmen in der Umwandlung verwenden verschiedene Kriterien zum Bestimmen des Ähnlichkeitsgrads zwischen den Werten. Sie können mehrere Strategien in einer Umwandlung definieren und jeder Strategie verschiedene Spalten zuweisen.

Beispiel für Spaltenanalyse

Sie möchten die Werte in einer Spalte mit Nachnamen-Daten vergleichen. Sie erstellen ein Mapping mit einer Datenquelle und einer Vergleichsumwandlung. Sie verbinden den Port *Nachname* mit der Vergleichsumwandlung. Die Umwandlung erstellt während der Zuordnungsausführung eine temporäre Kopie der Daten im Port *Nachname*.

Die folgende Abbildung zeigt ein Fragment der Nachnamedaten:

	A	B
1	**Surname**	**Surname_1**
2	Annan	Annan
3	Baker	Baker
4	Barker	Barker
5	Edwards	Edwards
6	Parker	Parker
7	Smith	Smith
8	Smith	Smith
9	Zhang	Zhang

Die Zuordnung erzeugt einen Satz an Vergleichswerten, die angeben, dass es sich bei den folgenden Werten unter Umständen um Duplikate handelt:

- Baker, Barker
- Barker, Parker
- Smith, Smith

Wenn Sie die Daten überprüfen, legen Sie fest, dass *Baker, Barker* und *Parker* keine doppelten Werte sind. Sie legen fest, dass *Smith* und *Smith* doppelte Werte sind.

Analyse mit Einzelquelle und Analyse mit Doppelquelle

Sie können die Match-Umwandlung zum Analysieren von Daten aus einer oder zwei Datenquellen konfigurieren. Sie wählen die Ports aus allen Datenquellen aus, wenn Sie eine Strategie in der Umwandlung definieren.

Wenn Sie die Umwandlung konfigurieren, um die Analyse mit Einzelquelle durchführen, wählen Sie einen oder mehrere Ports aus einem einzelnen Dataset aus. Wenn Sie die Umwandlung konfigurieren, um die Analyse mit Doppelquelle durchzuführen, wählen Sie einen oder mehrere Ports aus jedem Dataset aus. Sie wählen die Ports in Paaren aus. Für jedes von Ihnen ausgewählte Port-Paar vergleicht die Umwandlung alle Werte eines Ports mit allen Werten eines anderen Ports. Wenn Sie eine Einzelquellenanalyse für Daten aus einer einzelnen Spalte durchführen, erzeugt die Umwandlung eine temporäre Kopie des von Ihnen ausgewählten Ports.

Hinweis: Wenn Sie eine Identitätsvergleichsanalyse durchführen, können Sie eine Datenquelle mit einem persistenten Index der Identitätsdaten vergleichen, den Sie in einem früheren Mapping erstellt haben. Verwenden Sie die Optionen **Match-Typ**, um eine Identitätsanalyse mit einem persistenten Index anzugeben.

Feldvergleichsanalysen und Identitätsvergleichsanalysen

Sie können eine Vergleichsumwandlung für Feldvergleichsanalysen oder Identitätsvergleichsanalysen konfigurieren.

Bei der Feld-Match-Analyse analysiert die Vergleichsumwandlung die Quelldaten, die die Umwandlung eingibt. Sie können eine Feldvergleichsanalyse für jeden Datentyp durchführen. In der Identitätsvergleichsanalyse erzeugt die Vergleichsumwandlung einen Index alternativer Datenwerte aus den Eingabedaten und analysiert die Indexdaten. Konfigurieren Sie die Vergleichsumwandlung für Identitätsvergleichsanalysen, wenn die Eingabeports Identitätsdaten enthalten. Eine Identität ist eine Gruppe von Datenwerten, die eine Person oder ein Unternehmen identifiziert.

Ein Datensatz kann eine Identität auf verschiedene Arten darstellen. Beispiel: Die folgenden Datenwerte stellen den Namen John Smith dar:

- John Smith
- Smith, John
- jsmith@email.com
- SMITHJMR

Die Vergleichsumwandlung liest die Daten in einem Datensatz und berechnet die möglichen alternativen Versionen der Identität. Die Umwandlung erstellt einen Index, der die aktuellen Versionen und alternativen Versionen der Identitäten enthält. Die Vergleichsumwandlung analysiert die Indexwerte und nicht die Werte in den Eingabedatensätzen.

Identitätspopulationsdateien

In Identitätsvergleichsvorgängen werden Referenzdatendateien gelesen, die als Populationen bezeichnet werden. Die Populationsdateien definieren die potenziellen Abweichungen in den Identitätsdaten. Diese Dateien werden nicht mit Informatica-Anwendungen installiert. Sie erwerben die Populationsdatendateien zum Download bei Informatica.

Installieren Sie die Dateien in einem Speicherort, auf den der Content-Managementdienst zugreifen kann. Legen Sie den Speicherort im Content-Managementdienst mithilfe von Informatica Administrator fest.

Gruppen in der Match-Analyse

Die Ausführung der Vergleichsanalysezuordnung kann wegen der Anzahl der Datenvergleiche, die von der Umwandlung durchgeführt werden müssen, einige Zeit in Anspruch nehmen. Die Anzahl der Vergleiche bezieht sich auf die Anzahl der Datenwerte in den von Ihnen ausgewählten Ports.

In der folgenden Tabelle wird die Anzahl der Berechnungen angezeigt, die eine Zuordnung für eine unterschiedliche Anzahl an Datenwerten in einem einzelnen Port durchführt:

Anzahl der Datenwerte	Anzahl der Vergleiche
10.000	50 Millionen
100.000	5 Milliarden
1 Million	500 Milliarden

Weisen Sie die Eingabedatensätze Gruppen zu, um die Ausführungsdauer der Zuordnung zu verringern. Eine Gruppe besteht aus mehreren Datensätzen, die identische Werte auf einem von Ihnen ausgewählten Port enthalten. Wenn Sie eine Vergleichsanalyse für gruppierte Daten durchführen, analysiert die Vergleichsumwandlung die Datensätze innerhalb jeder Gruppe. Die Umwandlung vergleicht nicht die Datensätze in einer Gruppe mit den Datensätzen in einer anderen Gruppe. Durch die Gruppen wird die Gesamtzahl der Vergleiche, die von der Umwandlung durchgeführt werden müssen, ohne Genauigkeitsverlust in der Zuordnungsanalyse reduziert.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie Daten in Gruppen zusammenfassen:

- Der Port, auf dem Sie die Daten gruppieren, ist der Gruppenschlüsselport. Ein Gruppenschlüsselport muss einen Bereich von doppelten Werten enthalten, wie z. B. den Namen einer Stadt oder eines Bundeslands in einem Adressdatensatz. Wenn die Zuordnungsdaten keinen verwendbaren Gruppenschlüsselport enthalten, verwenden Sie den Schlüsselgenerator zum Erstellen des Ports anhand der aktuellen Zuordnungsdaten. Verbinden Sie den Gruppenschlüsselausgabeport aus der Schlüsselgeneratorumwandlung mit der Vergleichsumwandlung.

  Sie können die Schlüsselgeneratorumwandlung auch zum Hinzufügen von Sequenzbezeichnern zu den Zuordnungsdaten verwenden.
- Feldvergleichsvorgänge müssen einen Gruppenschlüsselport angegeben. Wenn Sie die Vergleichsumwandlung für die Identitätsanalyse konfigurieren, wählen Sie keinen Gruppenschlüsselport aus. Die Identitätsanalyse erzeugt Gruppenschlüssel für die Identitätsindexdaten.
- Geben Sie keinen Gruppenschlüsselport an, den Sie in der Vergleichsanalyse verwenden möchten.
- Wenn Sie Gruppen erstellen, müssen Sie sicherstellen, dass die Gruppen eine gültige Größe aufweisen. Wenn die Gruppen zu klein sind, findet die Vergleichsanalyse unter Umständen nicht alle duplizierten Daten im Datensatz. Wenn die Gruppen zu groß sind, gibt die Vergleichsanalyse unter Umständen falsche Duplikate zurück. Wählen Sie Gruppenschlüssel aus, die Gruppen mit einer durchschnittlichen Größe von 10.000 Datensätzen erstellen.
- Gruppen ändern die Position der Datensätze im Zuordnungsdatensatz nicht.

Match-Paare und Cluster

Die Match-Umwandlung kann unterschiedlich viele Eingabe- und Ausgabezeilen lesen und schreiben sowie die Reihenfolge der Ausgabezeilen ändern. Sie bestimmen das Ausgabeformat für die Ergebnisse der Match-Analyse.

Die Umwandlung kann Zeilen in den folgenden Formaten schreiben:

Gematchte Paare

Die Umwandlung schreibt eine Zeile für jedes Datensatzpaar, das mit einem Score übereinstimmt, der dem Match-Schwellenwert entspricht. Die Umwandlung schreibt jedes Datensatzpaar in eine einzelne Zeile.

Da ein Datensatz u. U. mit mehr als einem anderen Datensatz übereinstimmt, kann ein Datensatz in mehr als nur einer Ausgabezeile erscheinen.

Bester Match

Die Umwandlung schreibt eine Zeile für jeden Datensatz in einem Dataset und fügt den ähnlichsten Datensatz aus einem anderen Dataset in dieselbe Zeile hinzu.

Cluster

Die Umwandlung weist Clustern die Ausgabedatensätze basierend auf dem Ähnlichkeitsgrad zwischen den Datensätzen zu. Ein Cluster ist eine Datensatzgruppe, in der jeder Datensatz mit mindestens einem anderen Datensatz übereinstimmt, dessen Score dem Match-Schwellenwert entspricht. Die Umwandlung schreibt jeden Datensatz in eine einzelne Zeile.

Jeder Datensatz in einem Cluster muss mit mindestens einem anderen Datensatz in dem Cluster übereinstimmen. Deshalb kann ein Cluster Datensatzpaare enthalten, die nicht miteinander übereinstimmen. Ein Cluster kann einen einzelnen Datensatz enthalten, sofern dieser mit keinem anderen Datensatz übereinstimmt.

Hinweis: Die Option „Cluster“ in einer Feldanalyse entspricht der Option „Cluster – Alle Matches“ in einer Identitätsanalyse. Die Option „Cluster – Bester Match“ in einer Identitätsanalyse kombiniert Cluster-Berechnungen mit Berechnungen von gematchten Paaren.

Konfigurieren Sie die Ausgabeoptionen in der Ansicht **Match-Ausgabe** der Umwandlung.

Verwandte Themen:

- “Cluster-Ausgabeoptionen” auf Seite 410

Übereinstimmungswert-Berechnungen

Die Übereinstimmungswert ist ein numerischer Wert, der den Grad der Ähnlichkeit zwischen zwei Spaltenwerten darstellt. Ein Algorithmus berechnet einen Übereinstimmungswert als Dezimalwert im Bereich 0 bis 1. Ein Algorithmus weist einen Score von 1 zu, wenn zwei Spaltenwerte identisch sind.

Wenn Sie mehrere Spaltenpaare zur Analyse auswählen, berechnet die Umwandlung auf der Grundlage der Scores in den ausgewählten Spalten einen Durchschnittsscore. Die Umwandlung weist standardmäßig Scores aus jedem Spaltenpaar gleiche Gewichtungen zu. Die Umwandlung leitet die relative Bedeutung der Spaltendaten im Dataset nicht ab.

Sie können die Gewichtungswerte bearbeiten, die die Umwandlung zum Berechnen des Übereinstimmungswerts verwendet. Bearbeiten Sie die Gewichtungswerte, wenn Sie Spalten im Dataset eine höhere oder niedrigere Priorität zuweisen möchten.

Außerdem können Sie die Scores festlegen, die die Umwandlung anwendet, wenn sie einen Null-Wert in einer Spalte vorfindet. Die Umwandlung behandelt Null-Werte standardmäßig als Datenfehler und weist jedem Wertepaar, das eine Null enthält, einen niedrigen Übereinstimmungswert zu.

Hinweis: Der Algorithmus, den Sie auswählen, bestimmt den Übereinstimmungswert zwischen zwei Werten. Die Algorithmus generiert einen einzigen Score für die beiden Werte. Die Übereinstimmungswerte richten sich nicht nach dem von Ihnen ausgewählten Typ der Match-Ausgabe oder Typ der Scoring-Methode.

Gewichtete Scores

Wenn Sie mehrere Spalten für die Match-Analyse auswählen, berechnet die Umwandlung auf der Grundlage der Scores in den Spalten einen Durchschnittsscore für jeden Datensatz. Die Durchschnittsscore enthält alle Gewichtungswerte, die Sie auf die Vergleichsalgorithmen für jede Spalte anwenden.

Standardmäßig verwenden alle Algorithmen den Gewichtungswert 0,5. Sie können den Wert erhöhen, wenn die Wahrscheinlichkeit größer ist, dass die ausgewählten Spalten duplizierte Informationen enthalten. Sie können den gewichteten Wert verringern, wenn die Wahrscheinlichkeit geringer ist, dass duplizierte Werte in den ausgewählten Spalten auf echte duplizierte Informationen zwischen Datensätzen hinweisen. Die Match-Umwandlung verwendet den Durchschnittsscore als einzigen Übereinstimmungswert für jedes Paar von Datensätzen.

Null-Übereinstimmungswerte

Ein Match-Algorithmus wendet einen vordefinierten Match-Score für ein Wertepaar an, wenn einer oder beide Werte null sind. Sie können den Übereinstimmungswert bearbeiten, den ein Feld-Match-Algorithmus auf Null-Werte anwendet.

Null-Übereinstimmungswerte und Feld-Match-Algorithmen

Beim Konfigurieren eines Feld-Match-Algorithmus überprüfen Sie die Werte der Übereinstimmungswerte, die der Algorithmus auf Null-Daten anwendet. Ein Feld-Match-Algorithmus wendet den Standardscore 0,5 an, wenn er zwei Werte vergleicht und ein Wert oder beide null sind. Der Score 0,5 gibt einen geringen Ähnlichkeitsgrad zwischen Datenwerten an.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie die Null-Match-Scores überprüfen:

- Wenn der Algorithmus Spalten analysiert, die Primärschlüssel oder andere wichtige Daten enthalten, bearbeiten Sie die Standardscores nicht. In diesem Fall stellt ein Null-Wert einen Datenfehler dar, und die Standardscores sind für die Daten geeignet.
- Wenn der Algorithmus Spalten analysiert, die optional Daten enthalten können, aktualisieren Sie die Werte der Null-Übereinstimmungswerte auf den Wert des Match-Schwellenwerts. Sie können die Auswirkung der Null-Werte auf die Match-Analyse aufheben, wenn Sie die Null-Übereinstimmungswerte auf den Wert des Match-Schwellenwerts setzen.

Null-Übereinstimmungswerte und Identitätsvergleichsalgorithmen

Ein Identitätsvergleichsalgorithmus wendet den Übereinstimmungswert 0 an, wenn er zwei Werte vergleicht und ein Wert oder beide null sind. Die Identitätsvergleichsanalyse weist einem eindeutigen Datensatzcluster einen Datensatz mit einem Null-Übereinstimmungswert zu und zeichnet den Wert 1 als Clustergröße auf. Sie können den Score bearbeiten, den ein Identitätsvergleichsalgorithmus auf Null-Daten anwendet.

Cluster-Ausgabeoptionen

Wählen Sie eine Cluster-Ausgabeoption aus, wenn ähnliche oder identische Datensätze in den Ausgabedaten angeordnet werden sollen.

Wenn Sie eine Cluster-Ausgabeoption auswählen, fügt die Umwandlung einen Cluster-ID-Wert zu jedem Datensatz der Ausgabe hinzu. Sie können die Datensätze nach den Cluster-ID-Werten sortieren. Die Umwandlungsausgabe enthält eine Zeile für jeden Datensatz. Wenn ein Datensatz nicht mit einem anderen Datensatz übereinstimmt, dessen Score dem Match-Schwellenwert entspricht, weist die Umwandlung dem Datensatz eine eindeutige Cluster-ID zu. Verwenden Sie die Ansicht **Match-Ausgabe**, um die Cluster-Ausgabeoptionen auszuwählen oder zu aktualisieren.

Sie können die folgenden Cluster-Ausgabeoptionen auswählen:

Cluster

Wählen Sie die Option aus, um Cluster-ID-Werte den Datensätzen der Ausgabe zuzuweisen.

Cluster – Bester Match

Wählen Sie die Option aus, um das Datensatzpaar mit dem höchsten Übereinstimmungswert zu einem Cluster hinzuzufügen. Da ein Datensatz den besten Match möglicherweise mit mehr als einem anderen Datensatz darstellt, kann mehr als ein Datensatzpaar einen Cluster-ID-Wert gemeinsam nutzen.

Cluster – Alle Matches

Die Option **Cluster – Alle Matches** funktioniert auf dieselbe Weise wie die Option **Cluster**.

Die Umwandlung verwendet **Cluster – Alle Matches** und **Cluster – Bester Match** als Optionsnamen in Identitätsvergleichsanalysen.

Hinweis: Wenn ein Datenintegrationsdienst mehrere Match-Umwandlungen gleichzeitig ausführt, generiert er eindeutige Cluster-ID-Werte für die Ausgabe aus jeder Umwandlung. Die Cluster-ID-Werte für die Datensätze, die jede Umwandlung generiert, sind daher nicht unbedingt aufeinander folgend.

Die Optionen „Cluster“ und „Cluster – Alle Matches“

Wählen Sie die Option „Cluster“ in der Feld-Match-Analyse aus. Wählen Sie die Option „Cluster – Alle Matches“ in der Identitätsvergleichsanalyse aus.

Die Match-Umwandlung verwendet zum Erstellen des Clusters die folgenden Regeln:

- Wenn zwei Datensätze einen Übereinstimmungswert besitzen, der dem Match-Schwellenwert entspricht, fügt die Match-Umwandlung die Datensätze zu einem Cluster hinzu.
- Wenn ein Datensatz im Dataset mit einem beliebigen Datensatz im Cluster übereinstimmt, fügt die Umwandlung den Datensatz zum Cluster hinzu.
- Wenn ein Datensatz in einem Cluster mit einem Datensatz in einem anderen Cluster übereinstimmt, führt der Prozess die Cluster zusammen.
- Die Umwandlung verarbeitet die Match-Ergebnisse so lange, bis alle Datensätze einem Cluster zugewiesen werden.
- Wenn ein Datensatz mit keinem anderen Datensatz im Dataset übereinstimmt, weist die Umwandlung dem Datensatz einen eindeutigen Cluster-ID-Wert zu.

Die Option „Cluster – Bester Match“

Wählen Sie die Option „Cluster – Bester Match“ in der Identitätsvergleichsanalyse aus.

Die Umwandlung verwendet die folgenden Regeln zum Erstellen des Clusters:

- Die Umwandlung identifiziert den Datensatz, der den höchsten Übereinstimmungswert mit dem aktuellen Datensatz besitzt. Wenn der Übereinstimmungswert dem Schwellenwert entspricht, fügt die Umwandlung das Datensatzpaar zu einem Cluster hinzu.
- Wenn sich einer der übereinstimmenden Datensätze in einem Cluster befindet, fügt die Umwandlung den anderen Datensatz zum aktuellen Cluster hinzu.
- Die Umwandlung verarbeitet die Match-Score-Ergebnisse so lange, bis alle Datensätze einem Cluster zugewiesen werden.
- Ein Cluster kann einen einzelnen Datensatz enthalten, sofern dieser mit keinem anderen Datensatz in den Daten übereinstimmt.

Hinweis: Mithilfe der Eigenschaft **Match** in der Ansicht **Match-Ausgabe** können Sie angeben, wie die Umwandlung eine einzelne Datenquelle mit einem persistenten Datenspeicher vergleichen soll. Die Eigenschaft **Match** legt fest, ob die Umwandlung die Quelldaten oder den persistenten Datenspeicher nach Duplikaten durchsuchen soll.

Verwandte Themen:

- “Match-Paare und Cluster” auf Seite 408

Treiber-Scores und Verknüpfungs-Scores in der Cluster-Analyse

Wenn Sie eine Cluster-Ausgabeoption in der Match-Umwandlung auswählen, können Sie Verknüpfungs- und Treiber-Scoredaten zu der Ausgabe hinzufügen.

Der Verknüpfungs-Score ist der Score zwischen zwei Datensätzen, der diese als Mitglieder desselben Clusters identifiziert. Die Verknüpfungen zwischen Datensätzen bestimmen die Zusammensetzung des Clusters. Jeder Datensatz kann mit jedem anderen Datensatz im selben Cluster verknüpft werden.

Der Treiber-Score ist der Score zwischen dem Datensatz mit dem höchsten Sequenz-ID-Wert in einem Cluster und einem anderen Datensatz im selben Cluster. Treiber-Scores bieten die Möglichkeit, alle Datensätze in einem Cluster anhand eines einzelnen Datensatzes zu bewerten. Wenn Sie Treiber-Scores zur Match-Ausgabe hinzufügen, wird das Mapping langsamer ausgeführt, da die Match-Umwandlung die Treiber-Scores erst dann berechnen kann, wenn alle Cluster abgeschlossen sind.

Hinweis: Die Match-Analyse generiert einen einzelnen Score-Satz für jede Strategie, die Sie definieren. Der Treiber-Score und der Verknüpfungs-Score zeigen die Übereinstimmungswerte für verschiedene Paare von Datensätzen in jedem Cluster an. Die Treiber-Scores und Verknüpfungs-Scores können von der Reihenfolge abhängen, in der Datensätze in der Umwandlung erfasst werden. Der Treiber-Score ist möglicherweise niedriger als der Match-Schwellenwert.

Beispiel für Clusteranalyse

Konfigurieren Sie eine Feld-Match-Strategie, um eine Spalte mit Nachnamen-Daten zu analysieren. Legen Sie einen Match-Schwellenwert von 0.825 in der Strategie fest. Wählen Sie ein geclustertes Ausgabeformat und führen Sie den Daten-Viewer in der Umwandlung aus.

In der folgenden Tabelle werden die Daten angezeigt, die im Daten-Viewer angezeigt werden:

Nachname	Sequenz-ID	Cluster-ID	Cluster-Größe	Treiber-ID	Treiber-Score	Verknüpfungs-ID	Score verlinken
SMITH	1	1	2	1 - 6	1	1 - 1	1
SMYTH	2	2	2	1 - 3	0.83333	1 - 2	1
SMYTHE	3	2	2	1 - 3	1	1 - 2	0.83333
SMITT	4	3	1	1 - 4	1	1 - 4	1
SMITS	5	4	1	1 - 5	1	1 - 5	1
SMITH	6	1	2	1 - 6	1	1 - 1	1

Der Daten-Viewer enthält die folgenden Informationen über die Nachnamen-Daten:

- SMITT und SMITS stimmen mit keinem Datensatz überein, der über einen dem Match-Schwellenwert entsprechenden Score verfügt. Die Match-Umwandlung legt fest, dass die Datensätze im Dataset eindeutig sind.

  Die Cluster-Größe von SMITT und SMITS beträgt 1. Um eindeutige Datensätze in der Cluster-Ausgabe zu finden, suchen Sie nach Clustern mit einem einzigen Datensatz.
- Der Verknüpfungs-Score von SMITH und SMITH beträgt 1. Die Match-Umwandlung legt fest, dass die Datensätze identisch sind. Die Umwandlung fügt die Datensätze einem einzelnen Cluster hinzu.
- SMYTH und SMYTHE verfügen über einen Verknüpfungs-Score von 0,83333. Der Score überschreitet den Match-Schwellenwert. Die Umwandlung fügt daher die Datensätze einem einzelnen Cluster hinzu.

Analyse von Master-Daten

Wenn Sie zwei Datenquellen in der Match-Umwandlung analysieren, müssen Sie eine Quelle als Master-Dataset angeben. Die Umwandlung vergleicht Datenwerte aus jedem Datensatz in dem von Ihnen angegebenen Dataset mit den entsprechenden Werten in jedem Datensatz des zweiten Dataset.

In vielen Organisationen besteht ein Master-Dataset aus einem permanenten, hochwertigen Datenspeicher. Bevor Sie einem Master-Dataset Datensätze hinzufügen, verwenden Sie die Match-Umwandlung, um sicherzustellen, dass die Datensätze den Master-Daten keine duplizierten Informationen hinzufügen.

Masterdaten - Beispiel

Eine Bank unterhält ein Master-Dataset mit Kundenkonten-Datensätzen. Die Bank aktualisiert das Master-Dataset jeden Tag mit Datensätzen, die neue Kundenkonten angeben. Die Bank verwendet ein Mapping zur Duplikatsanalyse, um sicherzustellen, dass die neuen Datensätze die Kundeninformationen im Master-Dataset nicht duplizieren. Das Master-Dataset und die neuen Benutzerkontotabellen weisen eine gemeinsame Struktur auf und die Tabellen verwenden denselben Datenbanktyp. Die Bank kann daher das Mapping zur Duplikatsanalyse jedes Mal wiederverwenden, wenn das Master-Dataset aktualisiert werden muss.

Direktionalität bei Analysen von Master-Datasets

Die Match-Umwandlung vergleicht Datensätze aus einem Dataset mit einem anderen in einer einzigen Richtung. Die Umwandlung vergleicht jeden Datensatz im Master-Dataset mit allen Datensätzen im zweiten Dataset. Die Umwandlung vergleicht nicht jeden Datensatz im zweiten Dataset mit allen Datensätzen im Master-Dataset. Deshalb kann die Auswahl des Master-Datasets die Ergebnisse der Match-Analyse beeinflussen.

In der folgenden Tabelle werden zwei Datasets angezeigt, die Sie bei der Identitätsvergleichsanalyse vergleichen können:

Data Set 1	Dataset 2
Alex Bell	Alexander Bell
Alexander Graham Bell	Thomas Edison
Alva Edison	Nicola Tesla
Marie Curie	Irene Joliot Curie
Dorothy Crowfoot	Dorothy Hodgkin

Wenn Sie Dataset 1 als Master-Dataset festlegen und die Ausgabeoption **Beste Übereinstimmung** auswählen, enthält die Ausgabe die folgenden Datensätze:

- Alex Bell, Alexander Bell
- Alexander Graham Bell, Alexander Bell

Wenn Sie Dataset 2 als Master-Dataset festlegen und die Ausgabeoption **Beste Übereinstimmung** auswählen, enthält die Ausgabe die folgenden Datensätze:

- Alexander Bell, Alex Bell

Wenn Dataset 2 das Master-Dataset ist, kann die Umwandlung Alexander Bell nicht Alexander Graham Bell zuordnen, da Alexander Bell in den Ausgabedaten bereits mit Alex Bell übereinstimmt.

Mapping erneut verwenden

Wenn Sie einem Master-Dataset regelmäßig Daten hinzufügen, konfigurieren Sie ein Mapping zur Duplikatsanalyse, das Sie wiederverwenden können. Sie können die Zuordnung wiederverwenden, wenn sich die Portkonfigurationen in der Datenquelle, die Sie mit dem Masterdatensatz vergleichen, nicht ändern.

Stellen Sie beim Ausführen der Zuordnung sicher, dass die Umwandlung die Masterdaten und die neueren Daten angibt. Sie können das Mapping ausführen, ohne alle anderen Konfigurationen zu aktualisieren.

Identitätsvergleichsanalyse und persistente Indexdaten

Wenn Sie ein Mapping ausführen, um die Identitätsinformationen zu analysieren, generiert die Vergleichsumwandlung einen Index, der die alternativen Versionen der Identitäten im Dataset speichert. Standardmäßig schreibt die Vergleichsumwandlung die Indexdaten in temporäre Dateien. Sie können die Umwandlung zum Speichern der Indexdaten in Datenbanktabellen konfigurieren.

Die von Ihnen erzeugten Indextabellen stellen einen Datenspeicher dar, den Sie in nachfolgenden Zuordnungen wiederverwenden können. Sie können die Indextabellen mit einer Datenquelle vergleichen und optional die Indextabellen mit Indexdaten aus der Datenquelle aktualisieren. Da die Indextabellen von der Umwandlung nicht erneut erzeugt werden, werden die nachfolgenden Zuordnungen schneller ausgeführt. Weiterhin können die Indextabellen einen vertrauenswürdigen Datenspeicher mit Identitätsdaten darstellen.

Verwandte Themen:

- "Match-Leistung in der Identitätsanalyse" auf Seite 418

Regeln und Richtlinien für persistente Indexdaten

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie eine Match-Umwandlung konfigurieren, um ein Master-Dataset mit Identitätsinformationen zu analysieren:

- Um einen wiederverwendbaren Index für ein Master-Dataset zu generieren, konfigurieren Sie die Umwandlung zum Schreiben der Indexdaten in Datenbanktabellen. Die Datenbanktabellen bilden einen persistenten Speicher der Indexdaten.
- Konfigurieren Sie den Datensatz als Mapping-Datenquelle, um die Identitäten in einem anderen Datensatz mit dem Indexspeicher zu vergleichen. Konfigurieren Sie die Match-Umwandlung zum Lesen der Datenquelle und des Indexdatenspeichers. Wählen Sie die Indextabellen aus dem Standardschema der von Ihnen angegebenen Datenbankverbindung aus.
- Die Match-Umwandlung fügt den Sequenzbezeichnerwert aus dem Eingabedatensatz zu den Indexdatenzeilen hinzu, die mit dem Datensatz übereinstimmen. Der Eingabeport *SequenceID* enthält die Sequenzbezeichner. Die Umwandlung verwendet die Sequenzbezeichner zum Verfolgen der Indexdaten mittels der verschiedenen Schritte in der Vergleichsanalyse. Trennen Sie nicht den Sequenz-ID-Port.
- Wenn Sie eine Match-Umwandlung mit einem Indexspeicher verbinden, verwendet die Umwandlung erneut die Eigenschaftswerte der Population, der Schlüsselebene, des Schlüsseltyps und des Schlüsselfelds aus der Umwandlung, die den Speicher erstellt hat. Die Umwandlung verwendet auch die Portkonfiguration aus der Umwandlung wieder, die den Speicher erstellt hat.

  Wenn die Umwandlungseigenschaften nicht übereinstimmen, kann die Identitätsanalyse die Mapping-Quelldaten und die Indexdaten nicht ordnungsgemäß vergleichen.

- Die Match-Umwandlung verwendet die Daten auf dem Eingabeport, den Sie als Schlüsselfeld zum Generieren des Identitätsindex auswählen. Die Umwandlung kann auch Daten aus anderen Ports in den Index schreiben. Wenn Sie die Felddatenports ohne Schlüssel von der Umwandlung trennen, werden beim Ausführen des Mappings sämtliche Daten in den entsprechenden Indexspalten gelöscht. Um die Daten des Eingabeports in den Indextabellen beizubehalten, heben Sie die Verbindung der Eingabedatenports nicht auf.
- Sie können die Vergleichsanalyse in der Match-Umwandlung deaktivieren, wenn Sie die Indextabellendaten für einen Datensatz erzeugen. Sie möchten die Vergleichsanalyse beispielsweise deaktivieren, wenn Sie einen Indexspeicher für einen Datensatz erstellen. Wenn Sie die Vergleichsanalyse deaktivieren, wird das Mapping schneller ausgeführt.

  Wenn Sie die Match-Analyse deaktivieren, kann die Match-Umwandlung Persistenz-Statuscodes und Persistenz-Statusbeschreibungen generieren und anzeigen. Match-Scores oder andere Daten in Verbindung mit den Ergebnissen der Match-Analyse werden von der Umwandlung weder generiert noch angezeigt. Wenn Sie beispielsweise die Umwandlung so konfigurieren, dass Datensätze Clustern zugewiesen werden, und Sie die Match-Analyse deaktivieren, werden von der Umwandlung keine Cluster-ID-Werte generiert oder angezeigt.
- Sie bestimmen, ob die Match-Umwandlung den Indexspeicher mit den Daten aus der Mapping-Quelle aktualisiert. Die Match-Umwandlung verwendet Sequenzbezeichner, um anzugeben, ob die Zeilen im Indexspeicher und in den Mapping-Daten dieselben Datensätze darstellen.

Match-Mapping-Leistung

Sie können eine Vorschau der Datenfaktoren anzeigen, die die Leistung der Vergleichsumwandlung bestimmen, bevor Sie das Mapping ausführen, das die Umwandlung enthält. Sie können überprüfen, ob das System über die Ressourcen zum Ausführen des Mappings verfügt. Sie können darüber hinaus sicherstellen, dass Sie die Umwandlung zur Bestimmung des Ähnlichkeitsgrads in den Eingabedaten ordnungsgemäß konfiguriert haben.

Verwenden Sie die Option **Match-Performance-Analyse**, um sicherzustellen, dass das System über die benötigten Ressourcen verfügt. Verwenden Sie die Option **Vergleichsclusteranalyse**, um sicherzustellen, dass die Zuordnung die Ähnlichkeitsgrade im Datensatz genau bestimmen kann.

Führen Sie die Vergleichsleistungs- und Vergleichsclusteranalyse für alle Vergleichsumwandlungen aus, die eine einzelne Datenquelle lesen. Führen Sie die Match-Performance-Analyse für alle Vergleichsumwandlungen aus, die eine Feldübereinstimmungsanalyse aus zwei Quellen durchführen. Führen Sie keine Vergleichsleistungs- oder Vergleichsclusteranalyse für eine Identitätsvergleichsstrategie aus, die eine Verbindung zu Indexdatenbanktabellen herstellt.

Drilldown für Match-Performance-Analyse

Sie können einen Drilldown in Vergleichsanalysedaten durchführen, um die Datensatzpaare anzuzeigen, die dem Vergleichsschwellenwert entsprechen oder diesen überschreiten. Doppelklicken Sie auf einen Datensatz in der Ansicht **Details** und verwenden Sie den Daten-Viewer zum Anzeigen der Datensätze, die dem von Ihnen ausgewählten Datensatz entsprechen. Der Daten-Viewer zeigt die Daten für jedes Datensatzpaar in einer einzelnen Zeile an. Die Zeile enthält den Zeilenbezeichner jedes Datensatzes im Paar.

Drilldown für Match-Cluster-Analyse

Sie können einen Drilldown in Clusteranalysedaten durchführen, um die Datensätze in jedem Cluster anzuzeigen. Doppelklicken Sie auf einen Cluster in der Ansicht **Details** und lesen Sie die Daten im Daten-Viewer. Der Daten-Viewer zeigt nur jeweils einen Cluster an. Zu den Clusterdaten gehören die von Ihnen

ausgewählten Score-Optionen, z. B. Treiber-Score, Verknüpfungs-Score, Treiber- oder Verknüpfungsbezeichner.

Protokollierung der Vergleichsumwandlung

Wenn Sie ein Mapping ausführen, das eine Vergleichsumwandlung verwendet, verfolgt das Developer-Tool-Protokoll die Anzahl der Vergleichsberechnungen, die das Mapping durchführt. Um die Protokolldaten anzuzeigen, wählen Sie im Daten-Viewer die Option **Protokoll anzeigen** aus.

Das Mapping aktualisiert das Protokoll nach jeweils 100.000 Berechnungen.

Anzeigen von Match-Cluster-Analysedaten

Sie können statistische Daten in den Clustern anzeigen, die die Umwandlung erstellen kann. In der Clusterstatistik wird die Ebene der Datensatzduplizierung im Dataset basierend auf der aktuellen Mapping-Konfiguration zusammengefasst.

Um die Daten anzuzeigen, klicken Sie mit der rechten Maustaste auf die Match-Umwandlung in der Mapping-Arbeitsfläche und wählen Sie **Match-Cluster-Analyse** aus.

Vor dem Ausführen der Analyse validieren Sie das Mapping, das die Umwandlung enthält.

Die Match-Cluster-Analyse zeigt die Daten für die folgenden Eigenschaften an:

Eigenschaft	Beschreibung
Quelle	Die Anzahl der Eingabedatenzeilen.
Letzte Ausführung	Das Datum und die Uhrzeit der Analyse.
Gesamtanzahl erkannter Cluster	Die Anzahl der Cluster, die die Match-Analyse generiert, wenn das Mapping ausgeführt wird.
Min. Cluster-Größe	Die Anzahl der Datensätze im Cluster oder die Cluster, die die wenigsten Datensätze enthalten. Wenn die minimale Cluster-Größe 1 ist, enthält das Dataset mindestens einen eindeutigen Datensatz.
Max. Cluster-Größe	Die Anzahl der Datensätze im Cluster oder die Cluster, die die meisten Datensätze enthalten. Wenn dieser Wert die durchschnittliche Cluster-Größe erheblich überschreitet, enthält das größte Cluster möglicherweise falsche Duplikate.
Anzahl der eindeutigen Datensätze	Die Anzahl der Datensätze im Dataset, die nicht mit einem anderen Datensatz mit einem Score übereinstimmt, der dem Match-Schwellenwert entspricht.
Anzahl der duplizierten Datensätze	Die Anzahl der Datensätze im Dataset, die mit einem anderen Datensatz mit einem Score übereinstimmt, der dem Match-Schwellenwert entspricht.
Gesamtsumme Vergleiche	Die Anzahl der Vergleichsoperationen, die das Mapping ausführt.
Durchschnittliche Cluster-Größe	Die durchschnittliche Anzahl der Datensätze in einem Cluster.

Anzeigen von Match-Performance-Analysedaten

Sie können statistische Daten in den Datensatzgruppen anzeigen, die das Mapping als Eingabedaten liest.

Um die Daten anzuzeigen, klicken Sie mit der rechten Maustaste auf die Match-Umwandlung in der Mapping-Arbeitsfläche und wählen Sie **Match-Performance-Analyse** aus.

Vor dem Ausführen der Analyse validieren Sie das Mapping, das die Umwandlung enthält.

Die Match-Performance-Analyse zeigt die Daten für die folgenden Eigenschaften an:

Eigenschaft	Beschreibung
Quelle	Die Anzahl der Eingabedatenzeilen.
Letzte Ausführung	Das Datum und die Uhrzeit der Analyse.
Gesamtanzahl erkannter Gruppen	Die Anzahl der für das Dataset definierten Gruppen basierend auf dem ausgewählten Gruppenschlüsselwert.
Durchsatz (Datensätze pro Minute)	Variablenwert, der die Geschwindigkeit der Match-Analyse schätzt. Sie legen diesen Wert fest. Verwenden Sie den Wert, um die zur Ausführung der Match-Analyse erforderliche Zeit zu schätzen.
Geschätzte Zeit für Datensatzabgleich	Die zur Analyse aller Datensätze im Dataset in Anspruch genommene Zeit basierend auf der Konfiguration der Match-Umwandlung.
Gesamtanzahl generierter Paare	Die Anzahl der Vergleiche, die die Umwandlung basierend auf der Anzahl der Eingabedatenzeilen und der Anzahl der Gruppen durchführen muss.
Min.-Gruppengröße	Variablenwert, der die minimale Anzahl von Datensätzen angibt, die eine Gruppe enthalten kann. Sie legen diesen Wert fest. Verwenden Sie den Wert, um sicherzustellen, dass das Mapping Gruppen mit einer verwendbaren Größe erstellt. **Hinweis:** Der Wert für die Mindestgruppengröße bestimmt nicht die Größe der bei der Ausführung des Mappings erstellten Gruppen.
Anzahl der Gruppen unter Min.-Schwellenwert	Die Anzahl der Gruppen, die im Vergleich zum Wert für die Mindestgruppengröße weniger Datensätze enthalten. Wenn die Größe vieler Gruppen unterhalb der Mindestgruppengröße liegt, müssen Sie die Umwandlung möglicherweise bearbeiten und einen anderen Gruppenschlüssel auswählen.
Max. Gruppengröße	Variablenwert, der die maximale Anzahl von Datensätzen angibt, die eine Gruppe enthalten kann. Sie legen diesen Wert für die Performance-Analyse fest. Verwenden Sie den Wert, um sicherzustellen, dass das Mapping Gruppen mit einer verwendbaren Größe erstellt. **Hinweis:** Der Wert bestimmt nicht die Größe der bei der Ausführung des Mappings erstellten Gruppen.
Anzahl der Gruppen über Max.-Schwellenwert:	Die Anzahl der Gruppen, die im Vergleich zum Wert für die maximale Gruppengröße mehr Datensätze enthalten. Wenn die Größe vieler Gruppen über der maximalen Gruppengröße liegt, müssen Sie die Umwandlung möglicherweise bearbeiten und einen anderen Gruppenschlüssel auswählen.

Match-Leistung in der Identitätsanalyse

Zur Steigerung der Zuordnungsleistung bei der Durchführung einer Identitätsanalyse für zwei Datensätze konfigurieren Sie die Vergleichsumwandlung zum Lesen von Identitätsindexdaten aus Datenbanktabellen. Führen Sie eine Zuordnung aus, um die Indextabellen für den Masterdatensatz zu erstellen. Führen Sie die Zuordnung erneut aus, um die Indexdaten mit einer anderen Datenquelle zu vergleichen.

Verwenden Sie die Optionen in der Ansicht **Vergleichstyp**, um die Datenbanktabellen anzugeben, in denen die Indexdaten gespeichert sind. Verwenden Sie dieselben Optionen zur Auswahl der Indextabellen, wenn Sie die Umwandlung für den Vergleich von Indexdaten mit Daten aus einer anderen Quelle konfigurieren.

Führen Sie zum Schreiben der Indexdaten in Datenbanktabellen die folgenden Aufgaben durch:

1. Erstellen Sie ein Mapping, das eine Datenquelle mit Identitätsinformationen liest.
2. Konfigurieren Sie eine Vergleichsumwandlung im Mapping, um die Indexdaten in die Datenbank zu schreiben.
3. Führen Sie das Mapping aus, um die Indexdaten zu generieren. Die Indexdaten stellen einen Datenspeicher dar, den Sie wiederverwenden können.

Führen Sie zum Lesen der Indexdaten aus den Datenbanktabellen die folgenden Aufgaben durch:

1. Erstellen Sie eine Zuordnung, die eine andere Identitätsdatenquelle liest.
2. Konfigurieren Sie eine Vergleichsumwandlung in der Zuordnung, um die Indexdaten aus der von Ihnen zuvor angegebenen Datenbank zu lesen.

   Wenn die Zuordnungsdatenquelle und die Indexdaten eine gemeinsame Struktur aufweisen, können Sie die Zuordnung, die die Indexdaten erzeugt hat, wiederverwenden.
3. Führen Sie das Mapping aus, um die Datenquelle mit den Indexdaten zu vergleichen.

   Das Mapping generiert Indexdaten für die Datenquelle. Das Mapping muss keine Indexdaten für das größere Dataset generieren. Das Mapping wird daher im Vergleich zu einem Mapping mit Doppelquelle, das Indexdaten für beide Datasets ausführt, schneller ausgeführt.

Verwandte Themen:

- "Identitätsvergleichsanalyse und persistente Indexdaten" auf Seite 414

Erstellen eines Datenspeichers für Identitätsindexdaten

Konfigurieren Sie ein Mapping, das eine Datenquelle mit Identitätsinformationen liest. Verwenden Sie eine Vergleichsumwandlung, um die Indexdaten in die Datenbank zu schreiben.

1. Erstellen Sie ein Mapping und fügen Sie die Datenquelle zur Mapping-Arbeitsfläche hinzu.
2. Fügen Sie eine Vergleichsumwandlung zur Mapping-Arbeitsfläche hinzu.
3. Wählen Sie in der Datenquelle Ports aus, die die Identitätsinformationen enthalten.
   - Verbinden Sie die Ports mit den Identitätsinformationen der Vergleichsumwandlung.
   - Verbinden Sie den Port, der die Sequenzbezeichnerwerte enthält, mit der Vergleichsumwandlung.
4. Wählen Sie in der Vergleichsumwandlung die Ansicht **Match-Typ** aus.
5. Legen Sie den Vergleichstyp auf **Identitätsvergleich mit persistenter Datensatz-ID** fest.
6. Konfigurieren Sie die folgenden Optionen zum Erstellen des Indexdatenspeichers:
   - Legen Sie die Persistenzmethode auf **Datenbank mit neuen IDs aktualisieren** fest.

- Überprüfen Sie den Wert des Vergleichsprozesses. Als Standardwert wird **Aktivieren** verwendet. Wenn Sie den Vergleichsprozess deaktivieren, erstellt die Zuordnung die Identitätsindextabellen, führt aber keine Vergleichsanalyse für die Daten durch.
- Wählen Sie im Menü „Datenbankverbindung“ eine Datenbank für die Indextabellen aus.
- Wählen Sie im Menü „Persistenter Speicher“ die Option **Neu erstellen** aus. Geben Sie im Dialogfeld **Speichertabellen erstellen** einen Namen für den Index ein.

7. Führen Sie alle anderen Konfigurationsschritte aus, die für die Vergleichsumwandlung erforderlich sind. Konfigurieren Sie beispielsweise eine Umwandlungsstrategie.
8. Fügen Sie der Zuordnung ein Zieldatenobjekt hinzu.
9. Verbinden Sie die Ausgabeports der Vergleichsumwandlung mit dem Zieldatenobjekt.
10. Führen Sie das Mapping aus.

Das Mapping schreibt die Indexdaten für die Datenquelle in die von Ihnen angegebenen Datenbanktabellen.

Verwenden des Indexdatenspeichers in der Analyse mit Einzelquelle

Konfigurieren Sie ein Mapping, das eine Datenquelle mit Identitätsinformationen liest. Verwenden Sie eine Vergleichsumwandlung, um die Datenquelle mit dem Indexdatenspeicher für das Master-Dataset zu vergleichen.

Bevor Sie die Zuordnung konfigurieren, stellen Sie sicher, dass die Datenquelle eine Spalte mit Sequenzbezeichnerwerten enthält.

Um Zeit zu sparen, können Sie die Zuordnung, die den Indexdatenspeicher erstellt hat, kopieren oder wiederverwenden.

1. Öffnen Sie das Mapping, das den Indexdatenspeicher generiert hat.

   Alternativ dazu können Sie eine Kopie des Mappings öffnen.
2. Überprüfen Sie die Datenquelle in der Zuordnung.

   Ersetzen Sie gegebenenfalls die Datenquelle durch ein Quelle, die die aktuellen Daten enthält.

   Hinweis: Wenn Sie die Datenquelle löschen, löschen Sie ebenfalls die Portverbindungen in der Vergleichsumwandlung.
3. Geben Sie die Datenquellenports an, die die Identitätsinformationen enthalten.
   - Verbinden Sie die Ports mit den Identitätsinformationen der Vergleichsumwandlung.
   - Verbinden Sie den Port, der die Sequenzbezeichnerwerte enthält, mit der Vergleichsumwandlung. Die Eingabeports und die Reihenfolge der Eingabeports muss mit den Eingabeports in der Umwandlung übereinstimmen, die die Indextabellen erstellt hat.
4. Wählen Sie in der Vergleichsumwandlung die Ansicht **Match-Typ** aus.
5. Legen Sie den Vergleichstyp auf **Identitätsvergleich mit persistenter Datensatz-ID** fest.
6. Überprüfen Sie die Population, die Schlüsselebene, den Schlüsseltyp und die Schlüsselfeldwerte.
7. Konfigurieren Sie die Optionen zum Identifizieren des Indexdatenspeichers:
   - Richten Sie die Persistenzmethode ein. Wählen Sie beispielsweise die Option **Datenbank nicht aktualisieren** aus, um die aktuellen Daten in der Indextabelle beizubehalten.
   - Setzen Sie den Vergleichsprozess auf **Aktivieren**.
   - Wählen Sie im Menü „Datenbankverbindung“ die Datenbank aus, die die Indextabellen enthält.
   - Navigieren Sie im Menü „Persistenter Speicher“ zu den Tabellen, die die Indexdaten enthalten.

8. Konfigurieren Sie die folgenden Eigenschaften in der Ansicht „Match-Ausgabe“:
   - Match. Identifiziert die zu analysierenden Datensätze, wenn die Umwandlung Indexdaten aus den Datenbanktabellen liest.
   - Ausgabe. Filtert die Datensätze, die die Umwandlung als Ausgabe schreibt.
9. Führen Sie alle anderen Konfigurationsschritte aus, die für die Vergleichsumwandlung erforderlich sind.

   Konfigurieren Sie beispielsweise eine Umwandlungsstrategie.
10. Überprüfen Sie das Datenziel in der Zuordnung.

    Ersetzen Sie das Datenziel gegebenenfalls durch ein anderes Zielobjekt.
11. Verbinden Sie die Ausgabeports der Vergleichsumwandlung mit dem Zieldatenobjekt.
12. Führen Sie das Mapping aus.

Das Mapping vergleicht die Datenquellen-Datensätze mit dem Indexdatenspeicher. Die Umwandlung schreibt die Indexdaten für die Datenquelle in den Datenspeicher.

Match-Umwandlungsansichten

Die Match-Umwandlung organisiert die Optionen, die Sie in einer Reihe von Ansichten konfigurieren können. In einer wiederverwendbaren Umwandlung werden die Ansichten als Registerkarten auf der Developer tool-Arbeitsfläche angezeigt. Um eine Ansicht zu öffnen, klicken Sie auf eine Registerkarte. In einer nicht wiederverwendbaren Umwandlung werden die Ansichten als eine Liste auf den Arbeitsflächen angezeigt. Um eine Ansicht zu öffnen, klicken Sie auf ein Element in der Liste.

Beim Konfigurieren eines Matching-Analyse-Vorgangs können Sie die folgenden Ansichten konfigurieren:

Allgemein

Verwenden Sie die Ansicht „Allgemein“, um den Namen und die Beschreibung einer nicht wiederverwendbaren Match-Umwandlung zu aktualisieren. Die Beschreibung ist optional.

Ports

Verwenden Sie die Ansicht „Ports“, um die Eingabe- und Ausgabeports in einer nicht wiederverwendbaren Match-Umwandlung zu überprüfen.

Übersicht

Verwenden Sie die Ansicht „Übersicht“, um den Namen und die Beschreibung einer wiederverwendbaren Match-Umwandlung zu aktualisieren. Die Beschreibung ist optional. Verwenden Sie ebenfalls die Ansicht „Übersicht“, um die Eingabe- und Ausgabeports in einer wiederverwendbaren Umwandlung zu erstellen.

Match-Typ

Verwenden Sie die Ansicht „Match-Typ“, um den Typ der Duplikatsanalyse auszuwählen, die die Umwandlung durchführt. Sie können die Feld-Match-Analyse oder die Identitäts-Match-Analyse auswählen. Sie können eine einzelne Datenquelle oder zwei Datenquellen angeben.

Strategien

Verwenden Sie die Ansicht „Strategien“, um eine oder mehrere Strategien für die Analyse der Eingabedaten zu definieren. Sie wählen in jeder Strategie zwei Datenspalten aus und weisen den Spalten einen Match-Analyse-Algorithmus zu.

Match-Ausgabe

Verwenden Sie die Ansicht „Match-Ausgabe“, um die Struktur und das Format der Ausgabedaten anzugeben.

Parameter

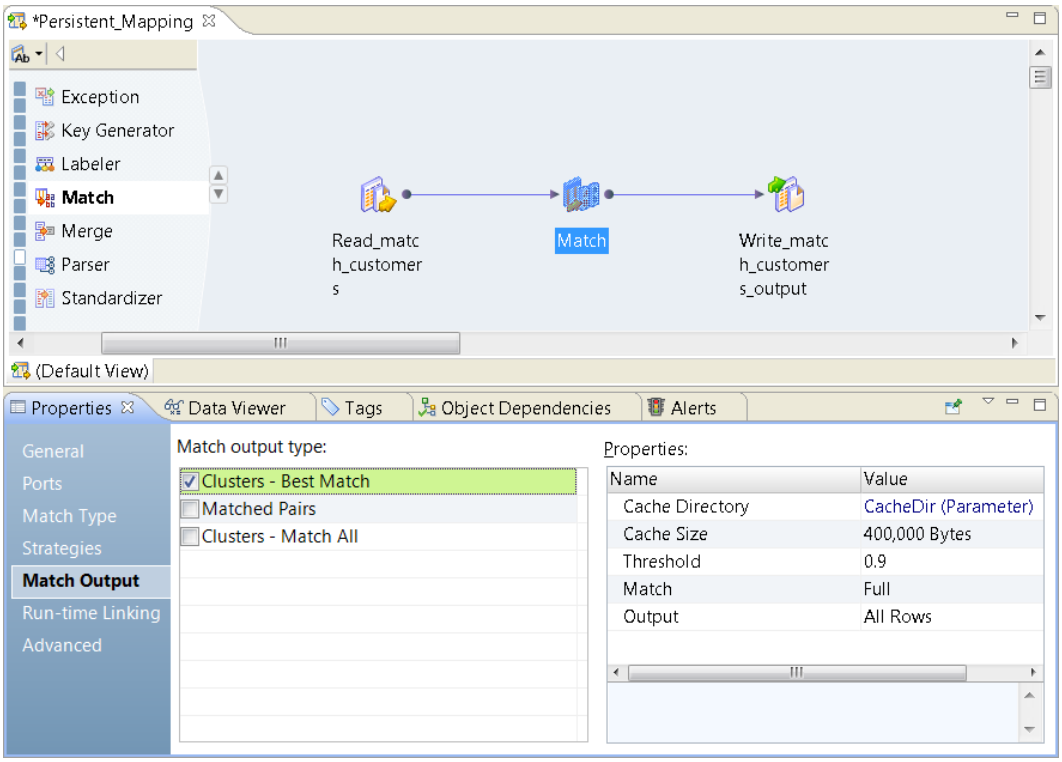
Verwenden Sie die Ansicht „Parameter“, um alle Parameter zu definieren, die der Datenintegrationsdienst auf die Umwandlung anwenden kann, wenn Sie ein Mapping ausführen, das die Umwandlung enthält.

Erweitert

Verwenden Sie die Ansicht „Erweitert“, um die folgenden Eigenschaften anzugeben:

- Die Detailebene in allen Protokollmeldungen, die das Mapping für die Umwandlung schreibt.
- Die Anzahl der Prozesse, die Identitäts-Match-Vorgänge bei Ausführung eines Mappings verwenden, das die Umwandlung enthält.
- Die Umwandlung übergibt Daten aus identischen Datensätzen direkt an die Ausgabeports. Sie können identische Datensätze filtern, wenn Sie die Umwandlung zum Schreiben von geclusterter Ausgabe konfigurieren.

Die folgende Abbildung zeigt die Ansichten einer Match-Umwandlung, die Sie für die Identitätsanalyse mit einem persistenten Indexspeicher konfigurieren:



Match-Umwandlungsports

Die Match-Umwandlung enthält eine Reihe von vordefinierten Eingabe- und Ausgabeports, die Metadaten für die von Ihnen definierten Match-Analyse-Vorgänge enthalten. Beim Konfigurieren des Match-Typs und der Match-Ausgabe-Optionen aktiviert oder deaktiviert die Umwandlung die Ports.

Wenn Sie die Umwandlung konfigurieren, überprüfen Sie die Metadaten-Ports. Wenn Sie die Umwandlung einem Mapping hinzufügen, stellen Sie sicher, dass die Metadaten-Ports mit den richtigen Ports für die vorangehenden und nachfolgenden Mapping-Objekte verbunden sind.

Match-Umwandlungs-Eingabeports

Die vordefinierten Eingabeports enthalten die Metadaten, die die Umwandlung für die Match-Analyse erfordert.

Nach dem Erstellen einer Match-Umwandlung können Sie die folgenden Eingabeports konfigurieren:

SequenceId

Eindeutiger Bezeichner für jeden Datensatz im Mapping-Quelldatensatz. Jeder Datensatz in einem Eingabe-Dataset muss über einen eindeutigen Sequenzbezeichner verfügen. Wenn ein Dataset doppelte Sequenzbezeichner enthält, kann die Match-Umwandlung duplizierte Datensätze nicht ordnungsgemäß identifizieren. Verwenden Sie die Schlüsselgeneratorumwandlung, um eindeutige Bezeichner zu erstellen, wenn im Datensatz keine Bezeichner vorhanden sind.

Wenn Sie einen Indexdatenspeicher für Identitätsdaten erstellen, fügt die Match-Umwandlung den Sequenzbezeichner für jeden Datensatz zum Datenspeicher hinzu. Wenn Sie die Umwandlung zum Vergleichen einer Datenquelle mit dem Indexdatenspeicher konfigurieren, findet die Umwandlung möglicherweise in beiden Datasets einen allgemeinen Sequenzbezeichner. Die Umwandlung kann die Sequenzbezeichner analysieren, wenn sie in den jeweiligen Datasets eindeutig sind.

GroupKey

Schlüsselwert, der die Gruppe identifiziert, zu der der Datensatz gehört.

Hinweis: Um die Matching-Leistung zu erhöhen, konfigurieren Sie sowohl den GroupKey-Eingabeport als auch den Ausgabeport, der eine Verbindung mit demselben Wert für die Gesamtstellenanzahl zu diesem Port herstellt.

Vergleichsumwandlungs-Ausgabeports

Die vordefinierte Ausgabeports enthalten Metadaten über die Analyse, die die Umwandlung durchführt.

Nach dem Erstellen einer Vergleichsumwandlung können Sie die folgenden Ausgabeports erstellen:

GroupKey

Schlüsselwert, der die Gruppe angibt, zu der der Datensatz gehört.

Nachgelagerte Umwandlungen wie die Zuordnungsumwandlung können den Gruppenschlüsselwert lesen.

ClusterId

Die Bezeichner des Clusters, zu dem der Datensatz gehört. Wird in der Cluster-Ausgabe verwendet.

ClusterSize

Die Anzahl der Datensätze in dem Cluster, zu dem der Datensatz gehört. Wenn ein Cluster einen eindeutigen Datensatz enthält, ist die Cluster-Größe 1. Wird in der Cluster-Ausgabe verwendet.

RowId und RowId1

Ein eindeutiger Zeilenbezeichner für den Datensatz. Die Vergleichsumwandlung verwendet den Zeilenbezeichner zur Identifizierung der Zeile während der Match-Analyse-Vorgänge. Dieser Bezeichner stimmt möglicherweise nicht mit der Zeilennummer in den Eingabedaten überein.

DriverId

Der Zeilenbezeichner des Treiberdatensatzes in einem Cluster. Wird in der Cluster-Ausgabe verwendet. Der Treiber-Datensatz ist der Datensatz im Cluster, der den höchsten Wert im SequenceID-Eingabeport aufweist.

DriverScore

Die Umwandlung weist der Ausgabe für abgeglichene Paare und der geclusterten Ausgabe einen Treiber-Score zu. In einem abgeglichenen Paar ist der Treiber-Score der Match-Typ zwischen den Datensatzpaaren. In einem Cluster ist der Treiber-Score der Match-Score zwischen dem aktuellen Datensatz und dem Treiberdatensatz im Cluster.

LinkId

Der Zeilenbezeichner des Datensatzes, der mit dem aktuellen Datensatz übereinstimmt und diesen mit dem Cluster verknüpft. Wird in der Cluster-Ausgabe verwendet.

LinkScore

Der Match-Score zwischen zwei Datensätzen, der dazu führt, dass ein Cluster erstellt oder ein Datensatz zu einem Cluster hinzugefügt wird. Der LinkID-Port identifiziert den Datensatz, mit dem der aktuelle Datensatz den Verknüpfungs-Score teilt. Wird in der Cluster-Ausgabe verwendet.

PersistenceStatus

Ein achtstelliger Code, der die Ergebnisse der Match-Analyse in einem Eingabeport darstellt. Wird in der Identitätsanalyse mit Einzelquelle verwendet, wenn die Umwandlung die Datenquelle mit einem Indexdatenspeicher vergleicht.

Die Umwandlung füllt die ersten drei Zeichen im Code. Die Umwandlung kann an jeder Position unterschiedliche Zeichen zurückgeben. Die Umwandlung gibt 0 für die Positionen 4 bis 8 zurück.

Wenn Sie die Umwandlung zum Generieren der Ausgabe für Match-Paare konfigurieren, erstellt die Umwandlung einen PersistenceStatus-Port und einen PersistenceStatus1-Port.

PersistenceStatusDesc

Eine Textbeschreibung der Werte für den Persistenzstatuscode. Wird in der Identitätsanalyse mit Einzelquelle verwendet, wenn die Umwandlung die Datenquelle mit einem Indexdatenspeicher vergleicht.

Wenn Sie die Umwandlung zum Generieren der Ausgabe für Match-Paare konfigurieren, erstellt die Umwandlung einen PersistenceStatusDesc-Port und einen PersistenceStatusDesc1-Port.

Persistenzstatuscodes und Persistenzstatusbeschreibungen

Die Persistenzstatuscodes und Persistenzstatusbeschreibungen erläutern die Beziehung zwischen den verschiedenen Typen von Indexdaten, die bei der Vergleichsumwandlung analysiert werden. Die Umwandlung erzeugt die Statuscodes und die Statusbeschreibungen, wenn Sie die Umwandlung zum Lesen eines persistenten Identitätsdatenspeichers konfigurieren.

Die Umwandlung schreibt den Persistenzstatuscode auf den PersistenceStatus-Port. Der Code besteht aus acht Zeichen. Die Umwandlung befüllt die ersten drei Positionen in der Zeichenfolge mit Codewerten. Die Umwandlung gibt 0 für die Positionen 4 bis 8 zurück.

Die Umwandlung schreibt die Persistenzstatusbeschreibung auf den PersistenceStatusDesc-Port. Die Beschreibung enthält drei kommagetrennte Textzeichenfolgen, die die Werte an den ersten drei Positionen im Persistenzstatuscode beschreiben.

Die Umwandlung verwendet die Sequenzbezeichnerwerte aus den Quelldatensätzen, um die Indexdaten für die beiden Datensätze zu vergleichen.

In der folgenden Tabelle werden die Informationstypen beschrieben, die die Umwandlung an jeder Position in der Statusbeschreibung und im Statuscode schreibt:

Position	Beschreibung
1	Gibt den Datensatz an, der den Datensatz enthält.
2	Gibt den Duplikatsstatus des Datensatzes an. Die Umwandlung sucht nach gemeinsamen Sequenzbezeichnern zwischen den Eingabedaten der Umwandlung und dem Indexdatenspeicher.
3	Beschreibt eine beliebige Aktion, die die Umwandlung für die Daten durchführt.
4-8	Der Statuscode enthält 0 an jeder Position. Die Statusbeschreibung enthält keinen Text für die Positionen.

Werte für Statuscode und Statusbeschreibung

Die Persistenzstatuscodes und die Persistenzstatusbeschreibungen erläutern die Beziehung zwischen den Eingabedatensätzen der Umwandlung und den Datensätzen, die der Datenspeicher angibt. Die Umwandlung verwendet Sequenzbezeichnerwerte zur Angabe der Datensätze und zur Bestimmung der Beziehung zwischen den Datensätzen in den Datensätzen.

Der Persistenzstatuscode und die Persistenzstatusbeschreibungen weisen eine gemeinsame Struktur auf. Die Statuscodes und Statusbeschreibungen enthalten an jeder Position in der Ausgabezeichenfolge dieselben Informationen.

Datensatzstatus

Der erste Wert im Statuscode und in der Statusbeschreibung gibt den Datensatz an, der den Datensatz enthält.

In der folgenden Tabelle werden die Statuscodes und Statusbeschreibungen erläutert, die die Umwandlung an der ersten Position zurückgeben kann:

Statuscode	Statusbeschreibung
S	Speicher. Der aktuelle Datensatz stammt aus dem Indexdatenspeicher.
I	Eingabe. Der aktuelle Datensatz stammt aus den Eingabedaten der Umwandlung.

Duplizierter Datensatz - Status

Der zweite Wert im Statuscode und in der Statusbeschreibung erläutert die Beziehung zwischen den Indexdaten der Umwandlung und dem persistenten Datenspeicher.

In der folgenden Tabelle werden die Statuscodes und Statusbeschreibungen erläutert, die die Umwandlung an der zweiten Position zurückgeben kann:

Statuscode	Statusbeschreibung
A	Nicht vorhanden. Der Indexdatenspeicher enthält keine Daten für den aktuellen Datensatz.
E	Vorhanden. Der aktuelle Datensatz ist im Indexdatenspeicher und in den Eingabedaten der Umwandlung vorhanden.
I	Ungültig. Die Umwandlung kann den aktuellen Datensatz nicht analysieren. Die Umwandlung kann beispielsweise keine Indexdaten für den Datensatz erzeugen, weil das Schlüsselfeld auf der Registerkarte „Vergleichstyp“ nicht mit den Daten im Datensatz kompatibel ist.
N	Neu. Der Datensatz ist in der Datenquelle vorhanden.
0	[Gedankenstrich] Der Datensatz ist im Indexdatenspeicher vorhanden.

Datenspeicherstatus

Der dritte Wert im Statuscode und in der Statusbeschreibung erläutert eine beliebige Aktion, die die Umwandlung in den Indexdatentabellen durchführt.

In der folgenden Tabelle werden die Statuscodes und Statusbeschreibungen erläutert, die die Umwandlung an der dritten Position zurückgeben kann:

Statuscode	Statusbeschreibung
A	Hinzugefügt. Die Umwandlung fügt die Indexdaten für den aktuellen Eingabedatensatz zum persistenten Datenspeicher hinzu. Die Eingabedaten der Umwandlung sowie die persistenten Indexdaten weisen unterschiedliche Sequenzbezeichner auf.
I	Ignoriert. Die Umwandlung fügt keine Indexdaten für den aktuellen Eingabedatensatz zum persistenten Datenspeicher hinzu.
N	Die Umwandlung gibt eine der folgenden Beschreibungen zurück: - Keine Änderung. Der aktuelle Datensatz stammt aus dem persistenten Datenspeicher und die Umwandlung führt keine Aktion durch. - Nicht hinzugefügt. Die Umwandlung aktualisiert den persistenten Datenspeicher aufgrund der von Ihnen definierten Vergleichsrichtlinie nicht mit Daten für den aktuellen Eingabedatensatz.

Statuscode	Statusbeschreibung
R	Entfernt. Die Umwandlung entfernt die Indexdaten für den Datensatz aus dem Indexdatenspeicher.
U	Aktualisiert. Die Umwandlung aktualisiert die Zeilen im persistenten Datenspeicher mit Indexdaten aus dem Eingabedatensatz der Umwandlung. Die Eingabedaten der Umwandlung sowie die persistenten Indexdaten weisen gemeinsame Sequenzbezeichner auf.

Persistenzstatusbeschreibung - Beispiel

Der Persistenzstatuscode INA00000 hat die folgende Persistenzstatusbeschreibung:

Eingabe, Neu, Hinzugefügt

Der Statuscode und die Statusbeschreibung enthalten die folgenden Informationen über den Datensatz:

- Der Datensatz stammt aus den Eingabedaten der Umwandlung.
- Der persistente Datenspeicher enthält keine Kopie des Datensatzes.
- Die Umwandlung fügt dem persistenten Datenspeicher die Indexdaten für den Datensatz hinzu.

Ausgabeports und Match-Ausgabe-Auswahl

Die von Ihnen ausgewählten Match-Ausgabeoptionen bestimmen die Ausgabeports für die Umwandlung. Die Umwandlung erstellt einen ClusterId-Port und einen ClusterSize-Port, wenn Sie einen geclusterten Ausgabetyp auswählen.

Wählen Sie den gewünschten Typ der Umwandlungsausgabe aus und überprüfen Sie die Ports in der Umwandlung.

Wenn Sie den Match-Ausgabetyp aktualisieren, überprüfen Sie im Anschluss die Ausgabeportkonfiguration für die Umwandlung. Wenn Sie die Umwandlung in einem Mapping verwenden, müssen Sie möglicherweise die Ausgabeports erneut mit den nachfolgenden Objekten im Mapping verbinden.

Match-Mapplets

Ein Match-Mapplet ist ein Mapplet, das Sie erstellen und in Match-Umwandlungen einbetten können.

Sie erstellen Match-Mapplets, indem Sie die Konfiguration einer Match-Umwandlung als Match-Mapplet speichern. Wenn Sie ein Match-Mapplet erstellen, konvertieren Sie die Einstellungen für die Match-Umwandlung in Vergleichs- und , Sie in Match-Umwandlung Einstellungen in Vergleichs-Umwandlungen und in Umwandlungen für gewichteten Durchschnitt.

Wenn Sie ein Match-Mapplet erstellt haben, können Sie die Umwandlungen hinzufügen, um die Match-Analyse anzupassen. Sie können beispielsweise eine Ausdrucks-Umwandlung hinzufügen, um die Link-Scores von zwei Strategien zu bewerten und den höchsten Score auszuwählen.

Im Gegensatz zu Match-Umwandlungen sind Mapplets passiv und können als Regeln im Analyst Tool verwendet werden. Verwenden Sie Match-Mapplets im Analyst Tool, um Datensätze als Teil des Daten-Profilingprozesses abzugleichen.

Die Match-Umwandlung kann Match-Mapplets nur lesen, die Sie innerhalb einer Match-Umwandlung erstellt haben.

Erstellen eines Match-Mapplets

Erstellen Sie ein Match-Mapplet zum Definieren eines Match-Analyse-Vorgangs, das mehrere Umwandlungen verwendet.

1. Öffnen Sie eine Match-Umwandlung im Editor und wählen Sie die Ansicht **Strategien** aus.
2. Wählen Sie **Match-Regel verwenden** aus.
3. Wählen Sie im Feld **Name** die Option **Neu erstellen** aus.

   Das Fenster **Neues Mapplet** wird geöffnet.
4. Geben Sie im Fenster **Neues Mapplet** einen Namen für das Mapplet ein und wählen Sie einen Speicherort für das Mapplet aus.
5. Wählen Sie optional **Strategien aus Match-Umwandlung erneut verwenden** aus, um Eingaben, Strategien und Gewichtungen aus der aktuellen Match-Umwandlung in das Match-Mapplet zu kopieren.

   Hinweis: Informatica empfiehlt die Verwendung dieser Einstellung, um Mapplets umgehend zu erstellen, die die in den Match-Umwandlungen aktuell definierten Match-Funktionen replizieren.
6. Klicken Sie auf **Fertig stellen**.

   Das Match-Mapplet wird im Editor geöffnet.
7. Optional können Sie Match-Vorgänge erstellen, indem Sie Vergleichs-Umwandlungen und Umwandlungen für gewichteten Durchschnitt zum Match-Mapplet hinzufügen und konfigurieren.
8. Um das Mapplet zu speichern, klicken Sie auf **Datei** > **Speichern**.
9. Schließen Sie das Mapplet und wählen Sie den Editor aus, der die Match-Umwandlung enthält. Überprüfen Sie das von Ihnen erstellte Mapplet im Feld **Name**.
10. Konfigurieren Sie optional die Match-Felder im Mapplet, indem Sie auf die Schaltfläche **Match-Felder** klicken.

    Das Fenster **Match-Regeln konfigurieren** wird geöffnet.
11. Doppelklicken Sie auf die Felder in den Spalten **Eingabefelder** und **Verfügbare Eingaben**, um den Match-Eingaben Eingabeports zuzuweisen.
12. Um die Umwandlung zu speichern, klicken Sie auf **Datei** > **Speichern**.

Mit einem Match-Mapplet

Sie können ein zuvor in der Match-Umwandlung definiertes Match-Mapplet auswählen und konfigurieren.

1. Öffnen Sie eine Match-Umwandlung im Editor und wählen Sie die Ansicht **Strategien** aus.
2. Wählen Sie **Match-Regel verwenden** aus.
3. Wählen Sie im Feld **Name** die Option **Verwende vorhandene** aus.

   Das Fenster **Match-Regeln konfigurieren** wird geöffnet.
4. Klicken Sie auf **Durchsuchen**, um ein Match-Mapplet im Repository zu suchen.

   Wichtig: Sie können nur Mapplets auswählen, die von der Match-Umwandlung erstellt wurden.

   Das Fenster **Match-Mapplet auswählen** wird geöffnet.

5. Wählen Sie ein Match-Mapplet aus und klicken Sie auf **OK**.
6. Doppelklicken Sie auf die Felder in den Spalten **Eingabefelder** und **Verfügbare Eingaben**, um den Match-Eingaben Eingabeports zuzuweisen.
7. Klicken Sie auf **OK**.

   Das Fenster **Match-Regeln konfigurieren** wird geschlossen.
8. Um die Match-Umwandlung zu speichern, klicken Sie auf **Datei** > **Speichern**.

Konfigurieren eines Match-Analyse-Vorgangs

Um einen Match-Vorgang zu konfigurieren, verbinden Sie die Quelldaten mit der Match-Umwandlung und bearbeiten Sie die Eigenschaften in den Umwandlungsansichten.

1. Erstellen Sie eine Match-Umwandlung und verbinden Sie die Quelldaten mit der Umwandlung.
2. Wählen Sie die Ansicht **Match-Typ** und einen Match-Typ aus.
3. Konfigurieren Sie die Eigenschaften für den von Ihnen ausgewählten Typ des Match-Vorgangs.

   Wenn Sie einen Match-Typ mit doppelter Quelle auswählen, konfigurieren Sie die Eigenschaft **Master-Dataset**.
4. Wählen Sie die Ansicht **Strategien** und die Option **Match-Strategien definieren** aus.
5. Klicken Sie auf **Neu**.

   Der Assistent **Neue Match-Strategie** wird geöffnet.
6. Wählen Sie eine Match-Strategie aus und klicken Sie auf **Weiter**.
7. Bearbeiten Sie optional die Gewichtungs- und Null-Match-Einstellungen. Klicken Sie auf **Weiter**.
8. Doppelklicken Sie auf die Zellen in der Spalte „Verfügbar“, um die zu analysierenden Eingabeports auszuwählen.

   Klicken Sie auf **Weiter**, um eine andere Strategie zu konfigurieren, oder klicken Sie auf **Fertig stellen**, um den Assistenten zu beenden.

   Hinweis: Um die Konfiguration für die Strategie zu bearbeiten, klicken Sie in der Ansicht **Strategien** auf den Pfeil in der Zelle für diese Strategie.
9. Wählen Sie die Ansicht **Match-Ausgabe** aus.

   Wählen Sie einen Match-Ausgabetyp aus und konfigurieren Sie die Eigenschaften.

Hinweis: Sie können auch Match-Strategien konfigurieren, indem Sie in der Ansicht **Strategien** ein Match-Mapplet auswählen oder erstellen. Ein Match-Mapplet ist ein Mapplet, das Sie in eine Match-Umwandlungen einbetten können.

KAPITEL 27

Match-Umwandlungen in der Feldanalyse

Dieses Kapitel umfasst die folgenden Themen:

- Feld-Match-Analyse, 429
- Prozessablauf für Feld-Match-Analysen, 429
- Optionen für den Feld-Match-Typ, 430
- Feld-Match-Strategien, 430
- Optionen der Feld-Match-Ausgabe, 434
- Erweiterte Feld-Match-Eigenschaften, 436
- Beispiel für die Feld-Match-Analyse, 436

Feld-Match-Analyse

Führen Sie eine Feld-Match-Analyse durch, um ähnliche oder doppelte Datensätze in Datasets oder zwischen zwei Datasets zu suchen.

Wenn Sie eine Match-Umwandlung für Feld-Match-Analysen konfigurieren, legen Sie die Optionen für die folgenden Ansichten fest:

- Match-Typ
- Strategien
- Match-Ausgabe

Legen Sie optional die Optionen für die Parameter und erweiterten Ansichten fest.

Prozessablauf für Feld-Match-Analysen

Der folgende Verarbeitungsfluss fasst die Schritte zusammen, die Sie zum Konfigurieren einer Match-Umwandlung für Feld-Match-Analysen verwenden. Sie können einen Prozess definieren, der nur die Match-Umwandlung oder mehrere Umwandlungen oder andere Umwandlungen verwendet.

Hinweis: Wenn Sie einem Mapping in einer Feld-Match-Analyse eine Match-Umwandlung hinzufügen, fügen Sie dem Mapping eine vorangehende Schlüsselgeneratorumwandlung hinzu.

Um die Daten für die Match-Umwandlung vorzubereiten, führen Sie die folgenden Schritte aus:

1. Organisieren Sie die Quelldaten-Datensätze in Gruppen.

   Verwenden Sie eine Schlüsselgeneratorumwandlung, um jedem Datensatz einen Gruppenschlüsselwert zuzuweisen. Die Gruppenzuordnungen verringern Sie die Anzahl der Berechnungen, die die Match-Umwandlung durchführen muss.

2. Stellen Sie sicher, dass die Datenquellen-Datensätze eindeutige Werte für den Sequenzbezeichner enthalten. Mit einer Schlüsselgeneratorumwandlung können Sie die Werte erstellen.

Führen Sie in der Match-Umwandlung die folgenden Schritte durch:

1. Geben Sie die Feldanalyse als Match-Typ und die Anzahl der Datenquellen an.

   Wenn Sie die Umwandlung zur Analyse von zwei Datasets konfigurieren, wählen Sie ein Master-Dataset aus.

   Verwenden Sie die Ansicht **Match-Typ**, um den Typ und die Anzahl der Datenquellen festzulegen.

2. Definieren Sie eine Match-Analysestrategie. Wählen Sie einen Algorithmus aus und weisen Sie dem Algorithmus ein Spaltenpaar zu.

   Verwenden Sie die Ansicht **Strategien**, um die Strategie zu definieren.

3. Geben Sie die Methode an, die die Umwandlung zum Generieren der Match-Analyse-Ergebnisse verwendet.

4. Legen Sie den Match-Schwellenwert fest. Der Schwellenwert ist der minimale Score, der zwei Datensätze als Duplikate des jeweils anderen identifizieren kann.

   Verwenden Sie die Ansicht **Match-Ausgabe**, um die Ausgabemethode und den Match-Schwellenwert auszuwählen.

   Hinweis: Sie können den Match-Schwellenwert in einer Match-Umwandlung oder einer Umwandlung für gewichteten Durchschnitt festlegen. Verwenden Sie die Umwandlung für gewichteten Durchschnitt, wenn Sie ein Match-Mapplet erstellen.

Optionen für den Feld-Match-Typ

Der Match-Typ enthält eine einzelne Option, die sich auf eine Match-Analyse mit Doppelquelle bezieht. Die Option identifiziert das Master-Dataset. Die Ansicht „Match-Typ“ enthält keine Option für die Match-Analyse mit Einzelquelle.

Wenn Sie zwei Datasets analysieren, müssen Sie ein Dataset als Master-Dataset festlegen. Wenn kein Dataset einen Master-Dataset im Projekt oder Unternehmen darstellt, wählen Sie das größere Dataset als Master-Dataset aus.

Verwenden Sie die Option **Master-Dataset**, um das Master-Dataset anzugeben.

Feld-Match-Strategien

Die Strategieansicht listet die von Ihnen für die Eingabedaten definierten Strategien auf.

Die Strategien bestimmen, wie die Umwandlung die Ähnlichkeiten und Unterschiede zwischen den Datenquellen-Datensätzen bewertet.

Feld-Match-Algorithmen

Die Match-Umwandlung enthält Algorithmen, die Datenwerte in zwei Spalten vergleichen. Jede Algorithmus berechnet den Grad der Unterschiede zwischen Datenwerten unterschiedlich.

Wählen Sie einen Algorithmus aus, der die unterschiedlichen Datentypen analysiert, von denen Sie annehmen, dass sie sich in den von Ihnen ausgewählten Spalten befinden.

Bigramm

Verwenden Sie den Bigramm-Algorithmus, um lange Textzeichenfolgen zu vergleichen, wie z. B. Anschriften, die in ein einzelnes Feld eingegeben wurden.

Der Bigramm-Algorithmus berechnet einen Übereinstimmungswert für zwei Datenzeichenfolgen basierend auf dem Vorkommen aufeinander folgender Zeichen in beiden Zeichenfolgen. Der Algorithmus sucht nach Paaren aufeinander folgender Zeichen, die in beiden Zeichenfolgen identisch sind. Er dividiert die Anzahl der Paare, die in beiden Zeichenfolgen übereinstimmen, durch die Gesamtanzahl der Zeichenpaare.

Bigramm - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `larder`
- `lerder`

Diese Zeichenfolgen ergeben die folgenden Bigramm-Gruppen:

```
l a, a r, r d, d e, e r
l e, e r, r d, d e, e r
```

Beachten Sie, dass für das zweite Vorkommen der Zeichenfolge "`e r`" innerhalb der Zeichenfolge "`lerder`" keine Übereinstimmung gefunden wird, da kein zweites Vorkommen von "`e r`" in der Zeichenfolge "`larder`" vorhanden ist.

Zur Berechnung des Bigramm-Übereinstimmungswerts dividiert die Umwandlung die Anzahl der übereinstimmenden Paare (6) durch die Gesamtzahl der Paare in beiden Zeichenfolgen (10). In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 60% auf und der Übereinstimmungswert lautet `0,60`.

Hamming-Abstand

Verwenden Sie den Algorithmus für den Hamming-Abstand, wenn die Position der Datenzeichen einen wichtigen Faktor darstellt, beispielsweise in numerischen oder Codefeldern, wie Telefonnummern, Postleitzahlen oder Produktcodes.

Der Algorithmus für den Hamming-Abstand berechnet die Übereinstimmung für zwei Zeichenfolgen, indem die Anzahl der Positionen ermittelt wird, an denen sich die Zeichen zwischen den Zeichenfolgen unterscheiden. Bei Zeichenfolgen mit unterschiedlicher Länge wird jedes zusätzliche Zeichen in der längsten Zeichenfolge als Unterschied zwischen den Zeichenfolgen gewertet.

Hamming-Abstand - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `Morlow`
- `M`**`a`**`rlow`**`es`**

Die markierten Zeichen geben die Positionen an, die vom Hamming-Algorithmus als verschieden angegeben werden.

Zur Berechnung des Hamming-Übereinstimmungswerts wird bei der Umwandlung die Anzahl der übereinstimmenden Zeichen (5) durch die Länge der längsten Zeichenfolge (8) dividiert. In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 62,5% auf und der Übereinstimmungswert lautet `0.625`.

Abstand bearbeiten

Verwenden Sie den Algorithmus "Abstand bearbeiten", um Wörter oder kurze Textzeichenfolgen, wie z. B. Namen, zu vergleichen.

Der Algorithmus "Abstand bearbeiten" berechnet die minimalen "Kosten" der Umwandlung einer Zeichenfolge in eine andere Zeichenfolge durch Einfügen, Löschen, oder Ersetzen von Zeichen.

Abstand bearbeiten - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `Levenston`
- `Levens**h**t**ei**n`

Die markierten Zeichen geben die Operationen an, die zum Umwandeln einer Zeichenfolge in eine andere Zeichenfolge benötigt werden.

Der Algorithmus "Abstand bearbeiten" dividiert die Anzahl der unveränderten Zeichen (8) durch die Länge der längsten Zeichenfolge (11). In diesem Beispiel weisen die Zeichenfolgen eine Ähnlichkeit von 72,7% auf und der Übereinstimmungswert lautet `0.727`.

Jaro-Abstand

Verwenden Sie den Algorithmus für den Jaro-Abstand, um zwei Zeichenfolgen miteinander zu vergleichen, wenn die Ähnlichkeit der ersten Zeichen in den Zeichenfolgen eine Priorität darstellt.

Der Übereinstimmungswert „Jaro-Abstand“ gibt den Grad der Übereinstimmung zwischen den ersten vier Zeichen beider Zeichenfolgen und der Anzahl erkannter Zeichentranspositionen an. Die Umwandlung wichtet die Bedeutung der Übereinstimmung zwischen den ersten vier Zeichen mithilfe des Werts, den Sie in der Eigenschaft Strafe eingegeben haben.

Jaro-Abstand - Eigenschaften

Beim Konfigurieren des Algorithmus für den Jaro-Abstand können folgende Eigenschaften konfiguriert werden:

Strafe

Bestimmt die Strafe für den Übereinstimmungswert, wenn die ersten vier Zeichen in zwei verglichenen Zeichenfolgen nicht identisch sind. Wenn das erste Zeichen nicht übereinstimmt, wird bei der Umwandlung der komplette Strafwert abgezogen. Bei der Umwandlung werden Teile des Strafwerts basierend auf der Position der anderen nicht übereinstimmenden Zeichen abgezogen. Der Standardstrafwert lautet `0,20`.

Groß-/Kleinschreibung

Bestimmt, ob der Algorithmus für den Jaro-Abstand beim Vergleichen der Zeichen die Groß-/Kleinschreibung beachtet.

Jaro-Abstand - Beispiel

Betrachten Sie die folgenden Zeichenfolgen:

- `391859`
- `813995`

Wenn Sie den standardmäßigen Straf-wert von `0,20` zum Analysieren dieser Zeichenfolgen verwenden, gibt der Algorithmus für den Jaro-Abstand einen Übereinstimmungswert von `0,513` zurück. Dieser Übereinstimmungswert gibt an, dass die Zeichenfolgen eine Ähnlichkeit von 51,3% aufweisen.

Hamming-Abstand umkehren

Verwenden Sie den Algorithmus "Hamming-Abstand umkehren", um den Prozentsatz der Zeichenpositionen zu berechnen, die zwischen zwei Zeichenfolgen von rechts nach links unterschiedlich sind.

Der Algorithmus für den Hamming-Abstand berechnet die Übereinstimmung für zwei Zeichenfolgen, indem die Anzahl der Positionen ermittelt wird, an denen sich die Zeichen zwischen den Zeichenfolgen unterscheiden. Bei Zeichenfolgen unterschiedlicher Länge wertet der Algorithmus alle zusätzlichen Zeichen in der längsten Zeichenfolge als Unterschied zwischen den Zeichenfolgen.

Hamming-Abstand umkehren - Beispiel

Beachten Sie die folgenden Zeichenfolgen, die von rechts nach links angeordnet sind, um den Algorithmus "Hamming-Abstand umkehren" zu simulieren:

- 1-999-9999
- **011-0**1-999-999**1**

Die markierten Zeichen zeigen die Positionen an, die vom Algorithmus "Hamming-Abstand umkehren" als unterschiedlich erkannt wurden.

Zur Berechnung des Übereinstimmungswerts für "Hamming-Abstand umkehren" wird bei der Umwandlung die Anzahl der übereinstimmenden Zeichen (9) durch die Länge der längsten Zeichenfolge (15) dividiert. In diesem Beispiel liegt der Übereinstimmungswert bei `0,6`, was angibt, dass die Zeichenfolgen eine Ähnlichkeit von 60% aufweisen.

Eigenschaften für Feld-Match-Strategien

Öffnen Sie den **Strategie**-Assistenten in der Ansicht **Strategien** und konfigurieren Sie die Eigenschaften für jede Feld-Match-Strategie.

Beim Konfigurieren einer Feld-Match-Strategie können Sie die folgenden Eigenschaften konfigurieren:

Name

Identifiziert die Strategie nach dem Namen.

Gewicht

Bestimmt die relative Priorität, die dem Übereinstimmungswert bei der Berechnung des Gesamtscores für den Datensatz zugewiesen wird. Standardwert ist 0,5.

Einzelnes Feld Null

Definiert den Übereinstimmungswert, den der Algorithmus auf ein Paar von Datenwerten anwendet, wenn ein Wert null ist. Standardwert ist 0,5.

Beide Felder Null

Definiert den Übereinstimmungswert, den der Algorithmus auf ein Paar von Datenwerten anwendet, wenn beide Werte null sind. Standardwert ist 0,5.

Hinweis: Ein Match-Algorithmus berechnet keinen Übereinstimmungswert, wenn ein verglichener Spaltenwert oder beide null sind. Die Algorithmus wendet die in den Null-Match-Eigenschaften definierten Scores an. Sie können die Null-Match-Eigenschaften nicht löschen.

Optionen der Feld-Match-Ausgabe

Konfigurieren Sie die Optionen der **Match-Ausgabe**, um das Ausgabeformat für die Feld-Match-Analyse zu definieren.

Sie konfigurieren diese Optionen im Bereich **Match-Ausgabetyp** und im Bereich **Eigenschaften**.

Match-Ausgabetypen

Die Ansicht „Match-Ausgabe“ enthält Optionen, die das Format der Ausgabedaten angeben. Sie können die Umwandlung so konfigurieren, dass Datensätze in Cluster oder in abgeglichene Paare geschrieben werden.

Wählen Sie einen der folgenden Match-Ausgabetypen aus:

Bester Match

Schreibt jeden Datensatz in das Master-Dataset mit dem Datensatz, der die genaueste Übereinstimmung im sekundären Dataset darstellt. Der Match-Vorgang wählt den Datensatz im zweiten Dataset mit dem höchsten Übereinstimmungswert mit dem Master-Datensatz aus. Wenn mindestens zwei Datensätze den höchsten Score zurückgeben, wählt der Match-Vorgang den ersten Datensatz im zweiten Dataset aus. Bester Match schreibt jedes Datensatzpaar in eine einzelne Zeile.

Sie können **Bester Match** auswählen, wenn Sie die Umwandlung für eine Analyse mit doppelter Quelle konfigurieren.

Cluster

Schreibt Cluster, die miteinander verknüpfte Datensätze enthalten, mit Vergleichs-Scores, die mit dem Vergleichsschwellenwert übereinstimmen. Jeder Datensatz muss mit mindestens einem anderen Datensatz im Cluster übereinstimmen, dessen Score dem Schwellenwert entspricht.

Sie können die Option **Cluster** auswählen, wenn Sie die Umwandlung für eine Analyse mit Einzelquelle oder mit Doppelquelle konfigurieren.

Gematchte Paare

Schreibt alle Datensatzpaare, die miteinander mit einem Score übereinstimmen, der dem Match-Schwellenwert entspricht. Die Umwandlung schreibt jedes Paar in eine einzelne Zeile und fügt den Übereinstimmungswert für jedes Paar in jede Zeile hinzu. Wenn ein Datensatz mit mehr als einem anderen Datensatz übereinstimmt, schreibt die Umwandlung eine Zeile für jedes Datensatzpaar.

Sie können die Option **Gematchte Paare** auswählen, wenn Sie die Umwandlung für eine Analyse mit Einzelquelle oder mit Doppelquelle konfigurieren.

Eigenschaften für Match-Ausgabe

Die Ansicht „Match-Ausgabe“ enthält Eigenschaften, die das Verhalten des Cache-Arbeitsspeichers, den Match-Score-Schwellenwert sowie die Match-Scores angeben, die in der Umwandlungsausgabe angezeigt werden.

Sie können die Eigenschaften für Match-Ausgabe auch verwenden, um anzugeben, wie die Umwandlung Match-Score-Werte zu Ausgabedatensätzen hinzufügt.

Konfigurieren Sie nach Auswahl eines Match-Ausgabetyps die folgenden Eigenschaften:

Cache-Verzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst die temporären Daten während der Feld-Match-Analyse schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis,

wenn das von der Match-Analyse generierte Datenvolumen größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cache-Größe

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst für die Feld-Match-Analyse zuweist. Der Standardwert ist 400.000 Byte.

Bevor die Daten sortiert werden, weist der Datenintegrationsdienst die von Ihnen angegebene Speichermenge zu. Wenn die Match-Analyse eine größere Menge an Daten generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Verzeichnis. Wenn der Speicherbedarf für die Match-Analyse die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Schwellenwert

Legt den minimalen Match-Score fest, der zwei Datensätze als potenzielle Duplikate voneinander identifiziert.

Sie können dem Schwellenwert einen Parameter zuweisen. Legen Sie einen Dezimalwert im Bereich von 0 bis 1 fest.

Scoring-Methode

Bestimmt die Match-Score-Werte, die in der Umwandlungsausgabe angezeigt werden. Wählen Sie eine Scoring-Methode für Cluster-Ausgaben aus.

In der folgenden Tabelle werden die Optionen für die Scoring-Methode beschrieben:

Option für Scoring-Methode	Beschreibung
Beide	Fügt jedem Datensatz im Cluster den Link-Score und den Treiber-Score hinzu.
Verknüpfungs-Score	Fügt jedem Datensatz im Cluster den Verknüpfungs-Score hinzu. Standardoption.
Treiber-Score	Fügt jedem Datensatz im Cluster den Treiber-Score hinzu.
Keiner	Fügt keinem Datensatz im Cluster einen Match-Score hinzu.

Hinweis: Wenn Sie den Datensätzen den Treiber-Score hinzufügen, erhöhen Sie die Mapping-Laufzeit. Das Mapping wartet, bis alle Cluster abgeschlossen sind, bevor es den Datensätzen die Treiber-Score-Werte hinzufügt.

Erweiterte Feld-Match-Eigenschaften

Die Umwandlung enthält erweiterte Eigenschaften, die die Anzahl der Ausführungsinstanzen, die Art der Analyse von identischen Zeilen durch die Umwandlung und die Tracingebene für die Protokolldaten bestimmt.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Ausführungsinstanzen

Bestimmt die Anzahl der Threads, die die Umwandlung zur Laufzeit verwendet.

Die Match-Umwandlung verwendet eine einzelne Ausführungsinstanz in Feld-Match-Analysen. Sie können die Anzahl der Ausführungsinstanzen bearbeiten, wenn Sie die Umwandlung für die Identitäts-Match-Analyse konfigurieren.

Filter für genaue Übereinstimmung

Legt fest, ob die Umwandlung den Vergleichsalgorithmus in einer Match-Strategie auf Paare mit identischen Datensätzen in den Eingabedaten anwendet.

Wenn die Umwandlung ein Paar mit identischen Datensätzen findet, muss der Algorithmus den Grad der Ähnlichkeit zwischen den Datensätzen nicht analysieren. Die Umwandlung kann die Datensätze direkt ohne weitere Analyse an die Ausgabestufe übergeben. Um die Umwandlung für die Übergabe der identischen Datensätze direkt an die Ausgabestufe zu konfigurieren, wählen Sie die Option „Nach exakter Übereinstimmung filtern“ aus. Wenn die Eingabedaten viele identische Zeilen enthalten, führt der Vergleichsalgorithmus weniger Berechnungen durch und das Mapping wird schneller ausgeführt.

Wählen Sie die Option aus, wenn die Eingabedaten viele identische Zeilen enthalten. Wählen Sie die Option nicht aus, wenn die Eingabedaten nicht viele identische Zeilen enthalten, da die Umwandlung langsamer ausgeführt werden könnte.

Hinweis: Die Umwandlungsausgabe enthält dieselben Datensatzdaten, wenn Sie die Option auswählen oder deaktivieren. Die Umwandlung kann den Ausgabedatensätzen unterschiedliche Verknüpfungs-Scores zuweisen, wenn Sie diese Option auswählen oder deaktivieren.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Beispiel für die Feld-Match-Analyse

Sie sind ein Datenverwalter in einer Retailbank. Sie erhalten eine Reihe von Kontodatensätzen für die Kunden, die in den vergangenen sieben Tagen ein Bankkonto eröffnet haben. Sie möchten sicherstellen, dass das Dataset keine duplizierten Datensätze enthält. Sie entwerfen ein Mapping, um doppelte Daten in Datensätzen zu suchen.

Erstellen des Mappings

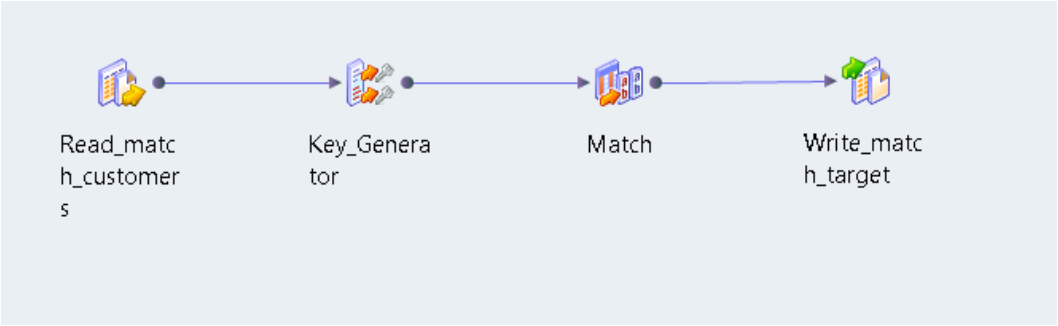
Erstellen Sie ein Mapping, das in mehreren Feldern nach doppelten Daten sucht.

Das Mapping führt die folgenden Aufgaben durch:

- Es liest eine Datenquelle.

- Es fügt einen Gruppenschlüsselwert und einen Wert für den Sequenzbezeichner zu den Quelldatensätzen hinzu.
- Es analysiert die Felddaten in den Datensätzen.
- Es schreibt die Ergebnisse in ein Datenziel.

Die folgende Abbildung zeigt das Mapping im Developer-Tool:



Das von Ihnen erstellte Mapping enthält die folgenden Objekte:

Objektname	Beschreibung
Read_Match_Customers	Datenquelle. Enthält Informationen zum Kontoinhaber, einschließlich Namen, Adressen und Kontonummern.
Key_Generator	Schlüsselgeneratorumwandlung. Fügt der Datenquelle Gruppenschlüsselwerte und Werte für den Sequenzbezeichner hinzu.
Match	Match-Umwandlung. Analysiert das Ausmaß der Duplizierung in den Kontodaten.
Write_match_target	Datenziel. Enthält die Ergebnisse der Feldanalyse.

Eingabedaten - Beispiel

Der Datensatz enthält die Kontonummer, den Namen, die Adresse und den Arbeitgeber für jeden Kunden. Sie erstellen eine Datenquelle aus dem Dataset im Modellrepository. Sie fügen die Datenquelle zum Mapping hinzu.

Der folgende Datenausschnitt zeigt ein Beispiel der Kundenkontodaten:

Kunden-ID	Nachname	Ort	Bundesland	PLZ
15954467	JONES	SCARSDALE	NY	10583
10110907	JONES	MINNEAPOLIS	MN	55437
19131127	JONES	INDIANAPOLIS	IN	46240
10112097	JONES	HOUSTON	TX	77036

Kunden-ID	Nachname	Ort	Bundesland	PLZ
19133807	JONES	PLANTATION	FL	33324
10112447	JONES	SCARSDALE	NY	10583
15952487	JONES	HOUSTON	TX	77002
10112027	JONES	OAKLAND	CA	94623

Konfiguration der Schlüsselgeneratorumwandlung

Wenn Sie die Schlüsselgeneratorumwandlung konfigurieren, verbinden Sie die Datenquellenports, die Sie analysieren möchten. Geben Sie den Port an, der die Gruppenschlüsseldaten enthält. Wenn die Datensätze keinen eindeutigen Bezeichner enthalten, verwenden Sie den Sequenz-ID-Port, um dem Datensatz eindeutige Bezeichner hinzuzufügen.

Wenn Sie den Gruppenschlüssel-Port angeben, beachten Sie die folgenden Richtlinien:

- Wählen Sie einen Port aus, der Werte enthält, die sich in den Portdaten regelmäßig wiederholen. Wählen Sie vorzugsweise einen Port aus, der ähnlich große Gruppen erstellt.
- Wählen Sie einen Port aus, der für die Duplikatsanalyse nicht relevant ist.

Im aktuellen Beispiel wählen Sie den Stadt-Port als Gruppenschlüssel aus. Wenn ein Kontoname mehr als einmal in einer Stadt vorkommt, enthalten die Konten möglicherweise doppelte Daten. Wenn ein Kontoname mehr als einmal in einer anderen Stadt vorkommt, sind die Konten höchstwahrscheinlich keine Duplikate.

Tipp: Führen Sie ein Spaltenprofil für die Datenquelle aus, bevor Sie den Gruppenschlüssel-Port auswählen. Die Profilergebnisse können angeben, wie oft jeder Wert in einem Port angezeigt wird.

Konfiguration der Match-Umwandlung

Fügen Sie dem Mapping eine nicht wiederverwendbare Match-Umwandlung hinzu, um die Feldanalyse durchzuführen.

Führen Sie die folgenden Aufgaben durch, um die Match-Umwandlung zu konfigurieren:

1. Wählen Sie den Typ der durchzuführenden Match-Analyse aus:
2. Verbinden Sie die Eingabeports mit der Umwandlung.
3. Konfigurieren Sie Strategien, um die Datensatzdaten zu vergleichen.
4. Wählen Sie den Typ der Match-Ausgabedaten aus, die die Umwandlung erstellt.
5. Verbinden Sie die Ausgabeports mit einem Datenziel.

Auswählen des Typs des Match-Vorgangs

Verwenden Sie die Optionen in der Ansicht **Match-Typ**, um den Match-Vorgang auszuwählen. Um die Kontodaten zu vergleichen, konfigurieren Sie die Umwandlung zur Ausführung einer Feldanalyse mit Einzelquelle.

Verbinden der Eingabeports

Verbinden Sie die Kundenkontodaten-Ports mit der Match-Umwandlung.

Die Match-Umwandlung verwendet voreingestellte Eingabeports, um die Reihenfolge festzulegen, in denen die Datensätze verarbeitet werden. Die Umwandlung verwendet einen Sequenzbezeichner, um die Datensätze von den Eingabeports bis zu den abgeglichenen Paaren oder Clustern zu verfolgen, in die die Ausgabe geschrieben wird. Die Umwandlung verwendet einen Gruppenschlüssel zum Sortieren der Datensätze, die verarbeitet werden.

Verbinden Sie die folgenden voreingestellten Ports in der Match-Umwandlung mit den voreingestellten Ausgabeports in der Schlüsselgeneratorumwandlung:

- SequenceID
- GroupKey

Konfigurieren der Strategien für die Feldanalyse

Verwenden Sie die Optionen in der Ansicht **Strategien**, um die Strategien zu konfigurieren. Die Strategien bestimmen die Analysetypen, die die Umwandlung für die Datensatzdaten durchführt.

Erstellen Sie die folgenden Strategien:

- Erstellen Sie eine Strategie, die den Algorithmus „Abstand bearbeiten" zwecks Analyse der Kunden-Identifikationsnummern verwendet. Wählen Sie die Ports „CustomerID_1“ und „CustomerID_2“ aus.
- Erstellen Sie eine Strategie, die den Algorithmus „Jaro-Abstand“ zwecks Analyse der Nachnamen-Daten verwendet. Wählen Sie die Ports „Lastname_1“ und „Lastname_2“ aus.

  Hinweis: Der Algorithmus „Jaro-Abstand“ wendet einen zusätzlichen Strafwert für ähnliche Zeichenfolgen an, die mit verschiedenen Zeichen beginnen. Der Algorithmus „Jaro-Abstand“ kann daher einen hohen Match-Score für PATTON und PATTEN anwenden, aber einen niedrigen Match-Score für BAYLOR und TAYLOR.
- Erstellen Sie eine Strategie, die den Algorithmus „Hamming-Abstand umkehren“ zwecks Analyse der Postleitzahl-Daten verwendet. Wählen Sie die Ports „Zip_1“ und „Zip_2“ aus.

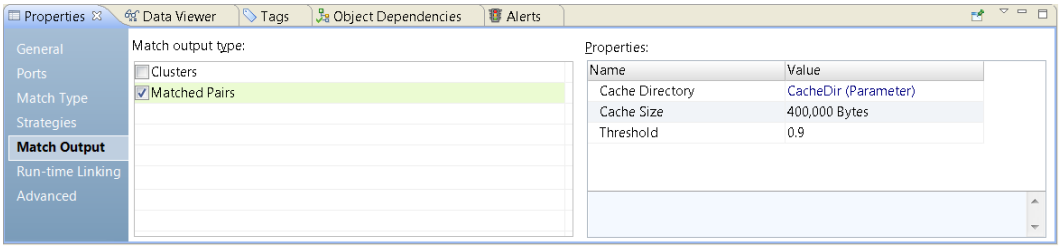
Verwandte Themen:

- “Feld-Match-Algorithmen” auf Seite 431

Auswählen des Match-Ausgabetyps

Verwenden Sie die Optionen in der Ansicht **Match-Ausgabe**, um das Ausgabeformat für die Ergebnisse der Match-Analyse zu definieren.

Die folgende Abbildung enthält die Ansicht „Match-Ausgabe“ für die Feldanalyse mit Einzelquelle:



Sie konfigurieren die Umwandlung, um die Ausgabedatensätze in abgeglichenen Paaren zu organisieren. Da die Umwandlung Datensätze in Paaren zurückgibt, enthalten die Analyse-Ergebnisse keine eindeutigen Datensätze.

Verbinden der Ausgabeports

Verbinden Sie die Ausgabeports der Match-Umwandlung mit dem Datenziel im Mapping. Wählen Sie die Ports aus, die die Datensatzdaten enthalten, die Sie ins Datenziel schreiben möchten.

Die Umwandlung enthält voreingestellte Ports, die die Datensätze in jedem abgeglichenen Paar identifizieren. Die voreingestellten Portnamen lauten **RowId** und **RowId1**. Jeder Zeilen-ID-Wert identifiziert einen Datensatz in den Ausgabedaten.

Die Zeilen-ID-Werte entsprechen den Ports, die Sie in einer Match-Strategie auswählen. Wählen Sie beim Konfigurieren einer Strategie Portnamen mit dem Suffix _1 oder _2 aus. Der RowId-Wert identifiziert den Datensatz, der einen Port mit dem Suffix _1 enthält. Der RowId1-Wert identifiziert den Datensatz, der einen Port mit dem Suffix _2 enthält.

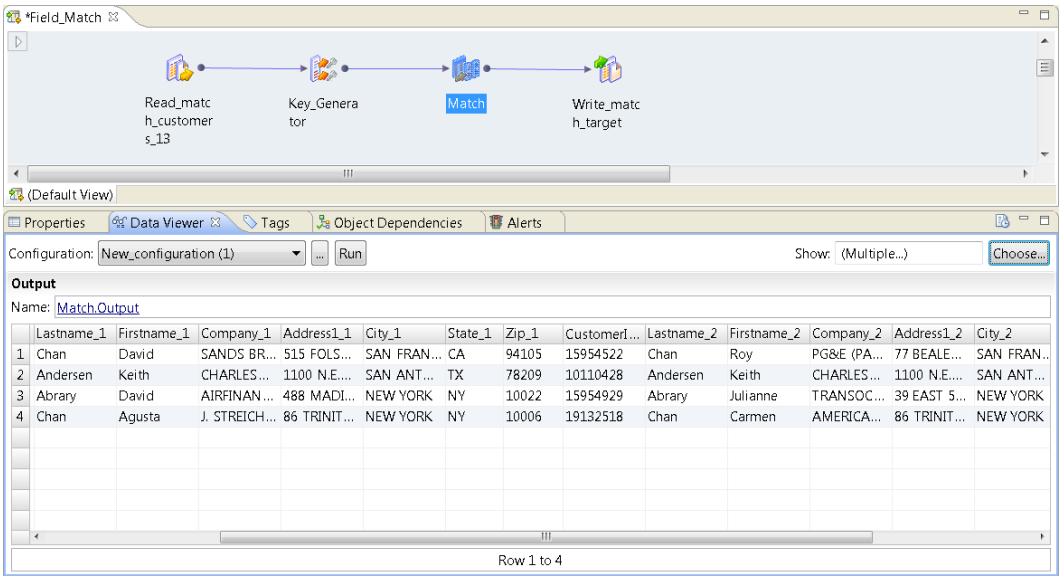
Sie können andere Ausgabeports verwenden, um die Beziehungen zwischen den Datensätzen zu überprüfen. Die Verknüpfungs- und Treiber-Portwerte geben den Grad der Ähnlichkeit zwischen den Datensätzen in jedem Cluster an.

Im aktuellen Beispiel verbinden Sie alle Ports mit dem Datenziel. Um Ausgabedaten in den Ports anzuzeigen, führen Sie den Daten-Viewer aus.

Ausführen des Daten-Viewers

Führen Sie den Daten-Viewer aus, um die Ergebnisse der Match-Analyse zu überprüfen. Der Daten-Viewer zeigt alle Ausgabeports in der Match-Umwandlung an. Wenn Sie das Mapping ausführen, können Sie das Datenziel mit den Daten aus den Ausgabeports aktualisieren.

Die folgende Abbildung zeigt die Ausgabedaten im Daten-Viewer:



Der Daten-Viewer überprüft, ob die Kundenkontendaten einen oder mehrere duplizierte Datensätze enthalten.

Beachten Sie die folgenden Daten im Daten-Viewer:

- Die Umwandlung bestimmt, dass die Datensätze für Augusta Chan und Carmen Chan möglicherweise die gleichen Informationen enthalten, weil sie dieselben Nachnamen- und Adressdaten enthalten. Wenn Sie die Datensätze überprüfen, legen Sie fest, dass die Datensätze im Dataset eindeutig sind. Sie stellen jedoch fest, dass die Datensätze einen gemeinsamen Kunden-ID-Wert verwenden. Da die Kunden-ID-Spalte ein Primärschlüssel im Dataset ist, wenden Sie sich an die Niederlassung in New York. Die Niederlassung in New York behebt den Fehler.

- Die Umwandlung bestimmt, dass die Datensätze für Keith Anderson möglicherweise die gleichen Informationen enthalten. Wenn Sie die Datensätze überprüfen, überprüfen Sie, ob die zwei Datensätze dasselbe Konto darstellen. Sie stellen jedoch fest, dass die Datensätze unterschiedliche Kunden-ID-Werte enthalten. Da jedes Kundenkonto einen einzelnen ID-Wert enthalten muss, wenden Sie sich an die Niederlassung in San Antonio. Die Niederlassung in San Antonio behebt den Fehler.

Ergebnis

Die Ergebnisse der Match-Analyse zeigen, dass das Kundenkonto-Dataset mindestens ein Paar mit doppelten Datensätzen enthält. Sie setzen sich mit dem lokalen Standort für die Konten in Verbindung, die Sie überprüfen müssen. Sie überprüfen, ob die anderen Datensätze im Dataset ein Kundenkonto eindeutig identifizieren.

KAPITEL 28

Match-Umwandlungen in der Identitätsanalyse

Dieses Kapitel umfasst die folgenden Themen:

- Identitäts-Match-Analyse, 442
- Prozessablauf für Identitätsvergleichs-Analysen, 443
- Eigenschaften für Identitätsvergleichstyp, 443
- Identitäts-Match-Strategien, 447
- Optionen für Identitäts-Match-Ausgabe, 450
- Erweiterte Identitäts-Match-Eigenschaften, 453
- Persistenter Index - Fallstudie, 454
- Identitätsvergleichsanalyse - Beispiel, 456

Identitäts-Match-Analyse

Führen Sie eine Identitäts-Match-Analyse durch, um ähnliche oder doppelte Identitäten in einem Dataset oder zwischen zwei Datasets zu suchen.

Ähnliche Identitäten in zwei oder mehr Datensätzen deuten eine Datensatzduplizierung an. Andere Identitäten können auf eine Verbindung zwischen Datensätzen hinweisen, zum Beispiel eine gemeinsame Familien- oder Arbeitgeberidentität.

Wenn Sie eine Match-Umwandlung für Identitäts-Match-Analysen konfigurieren, legen Sie die Optionen für die folgenden Ansichten fest:

- Match-Typ
- Strategien
- Match-Ausgabe

Legen Sie optional die Optionen für die Parameter und erweiterten Ansichten fest.

Beim Konfigurieren einer Match-Umwandlung für Identitäts-Match-Analysen müssen Sie Eingabeports mit allen primär erforderlichen Feldern in einer Identitätsstrategie verbinden. Die meisten Identitätsstrategien enthalten primäre erforderliche Felder. Einige Strategien enthalten ebenfalls sekundär erforderliche Felder. Verbinden Sie einen Eingabeport mit mindestens einem sekundär erforderlichen Feld.

Prozessablauf für Identitätsvergleichs-Analysen

Der folgende Verarbeitungsfluss fasst die Schritte zusammen, die Sie zum Konfigurieren einer Match-Umwandlung für Identitäts-Match-Analysen verwenden. Sie können einen Prozess definieren, der nur die Match-Umwandlung oder mehrere Umwandlungen oder andere Umwandlungen verwendet.

Bevor Sie die Match-Umwandlung mit den vorangehenden Datenobjekten verbinden, stellen Sie sicher, dass die Datensätze eindeutige Werte für den Sequenzbezeichner enthalten. Mit einer Schlüsselgeneratorumwandlung können Sie die Werte erstellen. Beim Ausführen der Identitäts-Match-Analyse können Sie optional die Eingabedaten in Gruppen organisieren.

Führen Sie in der Match-Umwandlung die folgenden Schritte durch:

1. Geben Sie die Identitätsanalyse als Match-Typ und die Anzahl der Datenquellen an.

   Wenn Sie die Umwandlung zur Analyse von zwei Datasets konfigurieren, wählen Sie ein Master-Dataset aus.

   Verwenden Sie die Ansicht **Match-Typ**, um den Typ und die Anzahl der Datenquellen festzulegen.
2. Geben Sie den Speicherort zum Speichern der Indexdaten an. Die Umwandlung kann die Indexdaten in temporäre Dateien schreiben oder in Datenbanktabellen speichern.

   Mithilfe der Ansicht **Match-Typ** können Sie den Index-Datenspeicher angeben.
3. Definieren Sie eine Match-Analysestrategie. Wählen Sie eine Population und einen Vergleichsalgorithmus aus und weisen Sie dem Algorithmus ein Spaltenpaar zu.

   Die Population gibt die auszuwählenden Spaltenpaare an.

   Verwenden Sie die Ansicht **Strategien**, um die Strategie zu definieren.
4. Geben Sie die Methode an, die die Umwandlung zum Generieren der Match-Analyse-Ergebnisse verwendet.
5. Legen Sie den Match-Schwellenwert fest. Der Schwellenwert ist der minimale Score, der zwei Datensätze als Duplikate des jeweils anderen identifizieren kann.

   Verwenden Sie die Ansicht **Match-Ausgabe**, um die Ausgabemethode und den Match-Schwellenwert auszuwählen.

   Hinweis: Sie können den Match-Schwellenwert in einer Match-Umwandlung oder einer Umwandlung für gewichteten Durchschnitt festlegen. Verwenden Sie die Umwandlung für gewichteten Durchschnitt, wenn Sie ein Match-Mapplet erstellen.

Eigenschaften für Identitätsvergleichstyp

Verwenden Sie die Ansicht „Vergleichstyp“, um den von der Match-Umwandlung durchgeführten Analysetyp anzugeben und die Eigenschaften festzulegen, die die Analyse definieren. Sie können eine Einzel- oder Doppelquellenanalyse festlegen. Sie können auch einen persistenten Datenspeicher für die Identitätsindexdaten angeben.

Die von Ihnen konfigurierten Eigenschaften richten sich nach dem von Ihnen ausgewählten Analysetyp. Viele der Optionen gelten für alle Analysetypen.

Allgemeine Eigenschaften

Die folgenden Eigenschaften gelten für alle Typen von Identitätsanalysen:

Population

Gibt die Populationsdatei an, die die Umwandlung verwendet. Die Populationsdatei enthält die schlüsselbildende Algorithmen, die die Indexschlüssel generieren.

Schlüsselebene

Bestimmt die Anzahl der Schlüssel, die die Identitätsalgorithmen generieren. Die Standardeinstellung ist Standard. Mit der Einstellung Begrenzt wird eine geringere Anzahl von Schlüsseln, eine höherer Genauigkeit und eine längere Verarbeitungszeit erzielt. Mit der Einstellung Erweitert wird eine höhere Anzahl von Schlüsseln, eine niedrigere Genauigkeit und eine kürzere Verarbeitungszeit erzielt.

Schlüsseltyp

Beschreibt den Typ von Informationen, die das Schlüsselfeld enthält. Die Identitätsanalyse kann Schlüssel für Personennamen, Organisationen und Adressen generieren. Wählen Sie den Schlüsseltyp aus, der die von Ihnen in der Eigenschaft **Schlüsselfeld** angegebenen Spalte am ehesten entspricht.

Suchebene

Gibt das Verhältnis zwischen der Suchtiefe und der Suchgeschwindigkeit an, die die Umwandlung auf die Match-Analyse anwendet. Die Suchtiefe korreliert inversiv mit der Anzahl der zurückgegebenen Übereinstimmungen. Die Option **Umfassend** gibt beispielsweise weniger Matches zurück.

Schlüsselfeld

Gibt die Spalte an, die die Match-Umwandlung zum Generieren der Indexschlüsseldaten verwendet. Stellen Sie sicher, dass die von Ihnen ausgewählte Spalte den Typ von Informationen enthält, den Sie für die Eigenschaft **Schlüsseltyp** angegeben haben.

Indexverzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst Indexschlüsseldaten für die aktuelle Umwandlung schreibt. Standardmäßig ist diese Eigenschaft leer. Wenn Sie kein Indexverzeichnis angeben, verwendet der Datenintegrationsdienst den von Ihnen für den Content-Management-Dienst festgelegten Speicherort.

Sie können einen Pfad zum Verzeichnis eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben.

Cache-Verzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst während der Indexerstellung bei der Identitätsvergleichsanalyse temporäre Daten schreibt. Aktualisieren Sie die Eigenschaft, um einen Speicherort für Daten aus der aktuellen Umwandlung anzugeben. Standardmäßig ist diese Eigenschaft leer. Wenn Sie kein Cache-Verzeichnis angeben, verwendet der Datenintegrationsdienst den von Ihnen im Content-Management-Dienst festgelegten Speicherort.

Sie können einen Pfad zum Verzeichnis eingeben oder einen Parameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben.

Cache-Größe

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst für die Identitätsindexerstellung zuweist. Der Standardwert ist 400.000 Byte.

Wenn der Vorgang der Indexerstellung eine größere Datenmenge generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in ein Cache-Verzeichnis. Wenn der Speicherbedarf für den Vorgang die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Doppelquelleneigenschaften

Legen Sie die folgende Eigenschaft zusätzlich zu den allgemeinen Eigenschaften fest, wenn Sie die Umwandlung für eine Doppelquellenanalyse konfigurieren:

Master-Dataset

Gibt die Datenquelle an, die die Masterdaten enthält. Geben Sie einen Master-Datensatz in der Analyse mit Doppelquelle an.

Eigenschaften des persistenten Datenspeichers

Legen Sie die folgenden Eigenschaften zusätzlich zu den allgemeinen Eigenschaften fest, wenn Sie die Umwandlung zur Verwendung des persistenten Indexdatenspeichers konfigurieren:

Persistenz-Methode

Gibt an, ob die Umwandlung die aktuellen Indextabellen mit Indexdaten aus der Mapping-Datenquelle aktualisiert. Wählen Sie eine der folgenden Optionen aus:

- Datenbank mit neuen IDs aktualisieren.
  Die Umwandlung fügt alle Zeilen zu den Indexdaten hinzu, die keinen Sequenzbezeichner in den Indexdaten duplizieren. Die aktuellen Zeilen im Index werden bei der Umwandlung nicht aktualisiert.

  Standardmäßig führt die Umwandlung eine Vergleichsanalyse durch, wenn Sie die Option auswählen. Sie können die Option „Vergleichsprozess“ verwenden, um die Vergleichsanalyse zu aktivieren oder zu deaktivieren.
- Datenbank nicht aktualisieren.
  Die Umwandlung aktualisiert die Indextabellen nicht mit Indexdaten aus der Mapping-Datenquelle.

  Die Umwandlung führt eine Vergleichsanalyse durch, wenn Sie die Option auswählen.
- IDs aus der Datenbank entfernen.
  Die Umwandlung löscht Zeilen aus den Indextabellen, wenn die Zeilen Sequenzbezeichner gemeinsam mit den Zuordnungsquelldaten nutzen.

  Die Umwandlung führt keine Vergleichsanalyse durch, wenn Sie die Option auswählen.
- Aktuelle IDs in der Datenbank aktualisieren.
  Die Umwandlung ersetzt Zeilen in den Indextabellen durch Zeilen aus den Zuordnungsquelldaten, wenn die Zeilen Sequenzbezeichner gemeinsam nutzen. Bei der Umwandlung werden keine Zeilen zum Index hinzugefügt.

  Standardmäßig führt die Umwandlung eine Vergleichsanalyse durch, wenn Sie die Option auswählen. Sie können die Option „Vergleichsprozess“ verwenden, um die Vergleichsanalyse zu aktivieren oder zu deaktivieren.

Die Standardpersistenzmethode lautet **Datenbank mit neuen IDs aktualisieren**.

Vergleichsprozess

Legt fest, ob die aktuelle Umwandlung Identitätsanalysen durchführt.

Die Option, die Sie in der Eigenschaft „Persistenzmethode“ ausgewählt haben, bestimmt die Optionen in der Eigenschaft „Vergleichsprozess“.

Datenbankverbindung

Identifiziert die Datenbank, die die Indextabellen enthält.

Persistenter Speicher

Identifiziert die Indextabellen innerhalb der von Ihnen angegebenen Datenbank.

Verwandte Themen:

- "Persistenter Index - Fallstudie" auf Seite 454
- "Parameter der Persistenz-Methode" auf Seite 447

Eigenschaften für Indexverzeichnis und Cache-Verzeichnis

Das Indexverzeichnis und das Cache-Verzeichnis speichern die temporären Daten, die die Match-Umwandlung während der Analyse generiert.

Der Datenintegrationsdienst schreibt Daten in das Indexverzeichnis, wenn Sie die Umwandlung nicht konfigurieren, um die Indexdaten in Datenbanktabellen zu schreiben. Der Datenintegrationsdienst schreibt Daten in das Cache-Verzeichnis, wenn die Angabe der Cachegröße nicht der Speichermenge entspricht, die für die Identitätsanalyse erforderlich ist. Standardmäßig sind die Eigenschaften in der Ansicht „Match-Typ" leer. Wenn Sie kein Indexverzeichnis oder Cache-Verzeichnis angeben, liest der Datenintegrationsdienst die Verzeichnispfade aus dem Content-Management-Dienst.

Wenn Sie ein Indexverzeichnis oder ein Cache-Verzeichnis in der Ansicht „Match-Typ" angeben, geben Sie einen lokalen Pfad zum Datenintegrationsdienst-Computer ein. Sie können einen vollständig qualifizierten Pfad oder einen relativen Pfad eingeben. Wenn Sie einen relativen Pfad eingeben, beginnen Sie die Pfadangabe mit einem Punkt. Der Pfad bezieht sich auf das Verzeichnis `tomcat/bin` auf dem Datenintegrationsdienst-Computer.

In der folgenden Tabelle werden ein relativer Pfad in einer Cache-Verzeichnis- oder Indexverzeichnis-Eigenschaft sowie der vollständig qualifizierte Pfad angegeben, den die Eigenschaft darstellt:

Relativer Pfad	Vollständig qualifizierter Pfad
`./ch`	`[Informatica_installation_directory]/tomcat/bin/ch`

Sie können die Indexverzeichniseigenschaft und die Cache-Verzeichniseigenschaft zur Angabe desselben Verzeichnisses konfigurieren. Wenn Sie dasselbe Verzeichnis für jede Eigenschaft angeben, erstellt der Datenintegrationsdienst Verzeichnisse in dem von Ihnen angegebenen Verzeichnis. Der Datenintegrationsdienst erstellt ein `Index`-Verzeichnis für die Indexdaten und ein `Cache`-Verzeichnis für die Cache-Daten. Wenn Sie unterschiedliche Verzeichnisse für jede Eigenschaft angeben, schreibt der Datenintegrationsdienst die Daten in die von Ihnen angegebenen Verzeichnisse.

Nach der Ausführung des Mappings löscht der Datenintegrationsdienst die Indexdateien und die Cache-Dateien aus den Verzeichnissen. Der Datenintegrationsdienst löscht die Verzeichnisse nicht. Der Datenintegrationsdienst kann die Verzeichnisse in anderen Mappings wiederverwenden, wenn die Match-Umwandlung die Verzeichnisse angibt.

Eigenschaften des Content-Management-Diensts

Der Content-Management-Dienst gibt die folgenden Standard-Speicherorte für das Indexverzeichnis und das Cache-Verzeichnis an:

- ./identityIndex. Standardverzeichnis für die Identitätsindexdaten.
- ./identityCache. Standardverzeichnis für die Identitäts-Cache-Daten.

Wenn Sie die Eigenschaften für die Match-Umwandlung nicht festlegen, erstellt der Datenintegrationsdienst die Verzeichnisse, wenn Sie ein Identitäts-Match-Mapping ausführen. Der Datenintegrationsdienst erstellt die Verzeichnisse im Verzeichnis `tomcat/bin`.

Parameter der Persistenz-Methode

Wenn Sie einen Persistenzdatenindex in der Identitätsvergleichsanalyse auswählen, können Sie einen Parameter zum Angeben der Persistenz-Methode verwenden. Verwenden Sie zur Bestimmung des Parameterwerts eine Zeichenfolge.

In der folgenden Tabelle werden die Persistenz-Methoden aufgelistet, die Sie auf der Registerkarte **Match-Typ** auswählen können, sowie die Parameterwerte, die Sie für die Methoden festlegen können:

Persistenz-Methode	Parameter
Datenbank mit neuen IDs aktualisieren	ignore
Datenbank nicht aktualisieren	addNone
IDs aus der Datenbank entfernen	remove
Aktuelle IDs in der Datenbank aktualisieren	update

Bei den Parameterwerten muss auf Groß-/Kleinschreibung geachtet werden.

Verwandte Themen:

- "Eigenschaften für Identitätsvergleichstyp" auf Seite 443

Identitäts-Match-Strategien

Die Ansicht „Strategien" enthält eine Liste der von Ihnen für die Identitätsdaten definierten Strategien. Die Strategien bestimmen, wie die Umwandlung die Ähnlichkeiten und Unterschiede zwischen den Daten im Identitätsindex bewertet.

Identitäts-Match-Algorithmen

Die Match-Umwandlung enthält vordefinierte Identitätsalgorithmen, die Datenwerte im Identitätsindex vergleichen. Wählen Sie den Algorithmus aus, der den Typ der Identitätsdaten im Dataset am besten darstellt.

In der folgenden Tabelle werden die Algorithmen und die für jeden Algorithmus ausgewählten Eingaben beschrieben:

Identitätsalgorithmus	Beschreibung
Adresse	Kennzeichnet Datensätze, die eine Adresse gemeinsam verwenden. Der Algorithmus erfordert die folgende primäre Eingabe: - Adresse Der Algorithmus erfordert keine sekundäre Eingabe.
Ansprechpartner	Kennzeichnet Datensätze, die einen Kontakt in einem einzelnen Unternehmensstandort gemeinsam verwenden. Der Algorithmus erfordert die folgenden primären Eingaben: - Person_Name - Organization_Name - Address_Part1 Der Algorithmus erfordert keine sekundäre Eingabe.
Unternehmenseinheit	Kennzeichnet Datensätze für gemeinsame Unternehmensidentifikationsdaten. Optional können Sie diesen Algorithmus zum Analysieren von Adress- und Telefondaten auswählen. Der Algorithmus erfordert die folgende primäre Eingabe: - Organization_Name Der Algorithmus erfordert keine sekundäre Eingabe.
Division	Kennzeichnet Datensätze für einen gemeinsamen Bürostandort in einem Unternehmen. Der Algorithmus erfordert die folgenden primären Eingaben: - Organization_Name - Address_Part1 Der Algorithmus erfordert keine sekundäre Eingabe.
Familie	Kennzeichnet Personen, die zu einer Familie gehören. Analysiert Namen, Adressen und Telefonnummern. Der Algorithmus erfordert die folgende primäre Eingabe: - Person_Name Der Algorithmus erfordert eine der folgenden sekundären Eingaben: - Address_Part1 - Telephone_Number
Felder	Kennzeichnet Datensätze mit gemeinsamen Daten auf von Ihnen ausgewählten Ports. Der Algorithmus gibt keine erforderlichen Eingaben an. Wählen Sie den Port oder Ports aus, die möglicherweise duplizierte Identitätsdaten enthalten.
Haushalt	Kennzeichnet Personen, die zu einem Haushalt gehören. Analysiert Namen und Adressen. Der Algorithmus erfordert die folgenden primären Eingaben: - Person_Name - Address_Part1

Identitätsalgorithmus	Beschreibung
Person	Kennzeichnet doppelt vorkommende Personen. Analysiert Namen, Geburtsdaten und persönliche Identifizierungsdaten wie Sozialversicherungsnummern, Kontonummern und Fahrgestellnummern. Der Algorithmus erfordert die folgende primäre Eingabe: - Person_Name Der Algorithmus erfordert eine der folgenden sekundären Eingaben: - Datum - ID
Unternehmen	Kennzeichnet Datensätze, die Organisationsdaten gemeinsam verwenden. Der Algorithmus erfordert die folgende primäre Eingabe: - Organization_Name Der Algorithmus erfordert keine sekundäre Eingabe.
Personenname	Kennzeichnet Datensätze mit gemeinsamen Informationen über Personen. Der Algorithmus erfordert die folgende primäre Eingabe: - Person_Name Der Algorithmus erfordert keine sekundäre Eingabe.
Bewohner	Kennzeichnet doppelt vorkommende Personen unter einer Adresse. Konfigurieren Sie optional diese Strategie, um persönliche Identifikationsdaten zu analysieren. Der Algorithmus erfordert die folgende primäre Eingabe: - Person_Name - Address_Part1 Der Algorithmus erfordert keine sekundäre Eingabe.
Wide Contact	Kennzeichnet Datensätze, die einen Kontakt in einem Unternehmen gemeinsam verwenden. Der Algorithmus erfordert die folgenden primären Eingaben: - Person_Name - Organization_Name Der Algorithmus erfordert keine sekundäre Eingabe.
Wide Household	Kennzeichnet Personen, die zum selben Haushalt gehören. Der Algorithmus erfordert die folgende primäre Eingabe: - Address_Part1 Der Algorithmus erfordert keine sekundäre Eingabe.

Eigenschaften für Identitäts-Match-Strategien

Konfigurieren Sie die Eigenschaften für jede Identitäts-Strategie.

Beim Konfigurieren einer Identitätsstrategie können Sie die folgenden Strategieeigenschaften konfigurieren:

Population

Bestimmt die auf die Identitätsanalyse anzuwendende Population. Populationen enthalten schlüsselbildende Algorithmen für bestimmte Gebietsschemata und Sprachen.

Match-Ebene

Bestimmt das Gleichgewicht von Suchqualität und Suchgeschwindigkeit. Die Suchgeschwindigkeit bezieht sich umgekehrt auf die Anzahl der zurückgegebenen Matches. Suchvorgänge mit der Einstellung

`Lose` geben weniger Matches zurück, während Suchvorgänge mit der Einstellung `Konservativ` mehr Matches zurückgeben.

Optionen für Identitäts-Match-Ausgabe

Die Ansicht „Match-Ausgabe“ enthält Optionen, die das Format der Ausgabedaten angeben. Sie können die Umwandlung so konfigurieren, dass Datensätze in Cluster oder in abgeglichene Paare geschrieben werden. Außerdem können Sie die Umwandlung zum Einbeziehen oder Ausschließen von anderen Identitätskategorien konfigurieren, wenn Sie eine Identitätsanalyse für einen persistenten Indexdatenspeicher durchführen.

Sie konfigurieren diese Optionen im Bereich **Match-Ausgabetyp** und im Bereich **Eigenschaften**.

Match-Ausgabetypen

Die Ansicht „Match-Ausgabe“ enthält Optionen, die das Format der Ausgabedaten angeben. Sie können die Umwandlung so konfigurieren, dass Datensätze in Cluster oder in abgeglichene Paare geschrieben werden.

Wählen Sie einen der folgenden Match-Ausgabetypen aus:

Bester Match

Schreibt jeden Datensatz in das Master-Dataset mit dem Datensatz, der die genaueste Übereinstimmung im sekundären Dataset darstellt. Der Match-Vorgang wählt den Datensatz im zweiten Dataset mit dem höchsten Übereinstimmungswert mit dem Master-Datensatz aus. Wenn mindestens zwei Datensätze den höchsten Score zurückgeben, wählt der Match-Vorgang den ersten Datensatz im zweiten Dataset aus. Bester Match schreibt jedes Datensatzpaar in eine einzelne Zeile.

Sie können **Bester Match** auswählen, wenn Sie die Umwandlung für eine Analyse mit doppelter Quelle konfigurieren.

Cluster – Bester Match

Schreibt Cluster, die die größte Übereinstimmung zwischen zwei Datensätzen in demselben Dataset oder in zwei Datasets darstellen. Der Übereinstimmungswert zwischen den beiden Datensätzen muss dem Match-Schwellenwert entsprechen. Cluster von besten Matches können mehr als zwei Datensätze enthalten, wenn ein Datensatz die größte Übereinstimmung mit mehreren anderen Datensätzen darstellt.

Sie können **Cluster – Bester Match** in einem beliebigen Typ der Identitätsanalyse auswählen.

Hinweis: Die von Ihnen ausgewählte Methode zum Speichern von Indexdaten kann sich auf den Inhalt der Clusterausgabe im Modus **Cluster – Bester Match** auswirken. Eine Umwandlung, die Verbindungen mit Index-Tabellen herstellt, kann andere Cluster als eine Umwandlung erstellen, die Indexdaten für dieselben Datensätze in temporären Dateien speichert. Die Methode zum Speichern von Indexdaten wirkt sich nicht auf die Match-Scores aus, die die Umwandlung für Datensatzpaare generiert.

Cluster – Alle Matches

Schreibt Cluster von Datensätzen, die mit einem Score übereinstimmen, der dem Match-Schwellenwert entspricht. Jeder Datensatz muss mit mindestens einem anderen Datensatz im Cluster übereinstimmen.

Sie können **Cluster – Alle Matches** in einem beliebigen Typ der Identitätsanalyse auswählen.

Gematchte Paare

Schreibt alle Datensatzpaare, die miteinander mit einem Score übereinstimmen, der dem Match-Schwellenwert entspricht. Die Umwandlung schreibt jedes Paar in eine einzelne Zeile und fügt den Übereinstimmungswert für jedes Paar in jede Zeile hinzu. Wenn ein Datensatz mit mehr als einem anderen Datensatz übereinstimmt, schreibt die Umwandlung eine Zeile für jedes Datensatzpaar.

Sie können **Gematchte Paare** in einem beliebigen Typ der Identitätsanalyse auswählen.

Eigenschaften für Match-Ausgabe

Die Ansicht „Match-Ausgabe“ enthält Eigenschaften, die das Verhalten des Cache-Arbeitsspeichers und den Match-Score-Schwellenwert angeben. Sie können die Eigenschaften auch verwenden, um zu bestimmen, wie die Umwandlung Datenspeicher-Datensätze für die Analyse auswählt und Datenspeicher-Datensätze als Ausgabe schreibt.

Konfigurieren Sie nach Auswahl eines Match-Ausgabetyps die folgenden Eigenschaften:

Cache-Verzeichnis

Gibt das Verzeichnis an, in das der Datenintegrationsdienst während der Identitäts-Match-Analyse temporäre Daten schreibt. Der Datenintegrationsdienst schreibt temporäre Dateien in das Verzeichnis, wenn das von der Match-Analyse generierte Datenvolumen größer als der verfügbare Systemspeicher ist. Nach dem Ausführen des Mappings löscht der Datenintegrationsdienst die temporären Dateien.

Sie können einen Verzeichnispfad für die Eigenschaft eingeben oder einen Systemparameter verwenden, um das Verzeichnis anzugeben. Geben Sie einen lokalen Pfad auf dem Datenintegrationsdienst-Computer an. Der Datenintegrationsdienst muss in der Lage sein, in das Verzeichnis zu schreiben. Der Standardwert ist der CacheDir-Systemparameter.

Cachegröße

Bestimmt die Menge des Systemspeichers, die der Datenintegrationsdienst für die Identitäts-Match-Analyse zuweist. Der Standardwert ist 400.000 Byte.

Wenn die Match-Analyse eine größere Menge an Daten generiert, schreibt der Datenintegrationsdienst die überschüssigen Daten in das Cache-Verzeichnis. Wenn der Speicherbedarf für die Match-Analyse die Größe des Systemspeichers und des Dateispeichers überschreitet, schlägt das Mapping fehl.

Hinweis: Wenn Sie den Wert 65536 oder einen höheren Wert eingeben, liest die Umwandlung den Wert in Byte. Wenn Sie einen niedrigeren Wert eingeben, liest die Umwandlung den Wert in Megabyte.

Match

Identifiziert die zu analysierenden Datensätze, wenn die Umwandlung Indexdaten aus den Datenbanktabellen liest. Verwenden Sie die Optionen in der Ansicht **Match-Typ**, um die Indextabellen zu identifizieren.

Standardmäßig analysiert die Umwandlung alle Datensätze in der Datenquelle und den Index-Datenbanktabellen. Konfigurieren Sie die Match-Eigenschaft, um eine Teilmenge der Datensätze für die Duplikatsanalyse anzugeben.

Ausgabe

Filtert die Datensätze, die die Umwandlung als Ausgabe schreibt, wenn Sie die Umwandlung zum Lesen der Index-Datenbanktabellen konfigurieren. Verwenden Sie die Optionen in der Ansicht **Match-Typ**, um die Indextabellen zu identifizieren.

Standardmäßig schreibt die Match-Umwandlung alle Datensätze aus der Datenquelle und den Index-Datenbanktabellen als Ausgabe. Konfigurieren Sie die Eigenschaft **Ausgabe**, wenn Sie nicht alle Datensätze in den Eingabedaten überprüfen müssen.

Schwellenwert

Legt den minimalen Match-Score fest, der zwei Datensätze als potenzielle Duplikate voneinander identifiziert.

Sie können dem Schwellenwert einen Parameter zuweisen. Legen Sie einen Dezimalwert im Bereich von 0 bis 1 fest.

Konfiguration der Match-Eigenschaft

Verwenden Sie die **Match**-Eigenschaft in der Ansicht **Match-Ausgabe**, um anzugeben, wie die Umwandlung Eingabedaten für die Analyse auswählt. Konfigurieren Sie die Eigenschaft, wenn Sie die Vergleichsumwandlung zum Lesen eines persistenten Speichers mit Indexdaten konfigurieren. Mit der Eigenschaft „Vergleich“ verfeinern Sie die Optionen, die Sie in der Ansicht **Vergleichstyp** eingerichtet haben.

Sie können die Match-Eigenschaft zum Ausführen der folgenden Analysetypen konfigurieren:

Vergleichen Sie die Datensätze der Datenquelle mit den Datensätzen der Indexdaten.

Um nach duplizierten Datensätzen zwischen der Datenquelle und den Indexdatentabellen zu suchen, wählen Sie **Exklusiv** aus.

Bei Auswahl der Option „Exklusiv“ vergleicht die Vergleichsumwandlung die Datenquellen-Datensätze mit dem Indexdatenspeicher. Die Umwandlung analysiert keine Datensätze innerhalb der Datenquelle oder innerhalb des Datenspeichers.

Wählen Sie **Exklusiv** aus, wenn Sie wissen, dass der Indexdatenspeicher und der Datenspeicher keine doppelten Datensätze enthalten.

Vergleichen Sie die Datenquellen-Datensätze mit den Indexdatensätzen und die Datenquellen-Datensätze untereinander.

Um nach Duplikaten in der Datenquelle sowie nach Duplikaten zwischen der Datenquelle und den Indextabellen zu suchen, wählen Sie **Teilweise** aus.

Die Umwandlung vergleicht die Datenquellen-Datensätze mit dem Indexdatenspeicher. Die Umwandlung vergleicht auch die Datensätze untereinander innerhalb Datenquelle.

Wählen Sie **Teilweise** aus, wenn Sie wissen, dass der Indexdatenspeicher keine doppelten Datensätze enthält, Sie aber keine Duplikatsanalyse in der Datenquelle durchgeführt haben.

Vergleichen Sie alle Datensätze in der Datenquelle und den Indextabellen als ein einzelnes Dataset.

Um nach Duplikaten zwischen der Datenquelle und den Indextabellen sowie nach Duplikaten innerhalb der Datenquelle und innerhalb der Indextabellen zu suchen, wählen Sie **Vollständig** aus. Die Standardoption ist „Vollständig“.

Die Umwandlung analysiert die Datenquelle und den Datenspeicher als ein einzelnes Dataset und vergleicht alle Datensätze im Dataset.

Wählen Sie **Vollständig** aus, wenn nicht überprüft werden kann, ob ein Datensatz doppelte Datensätze aufweist.

Konfiguration der Ausgabe-Eigenschaft

Verwenden Sie die Eigenschaft **Ausgabe** in der Ansicht **Match-Ausgabe**, um die Datensätze zu filtern, die die Umwandlung als Ausgabe schreibt. Konfigurieren Sie die Eigenschaft, wenn Sie Indextabellen angeben und ein geclustertes Ausgabeformat auswählen. Filtern Sie die Datensätze, um die Ausgabe auf Cluster zu begrenzen, die einen oder mehrere Datensätze aus der Datenquelle enthalten.

Sie können die Ausgabedaten wie folgt filtern:

Schreiben aller Cluster, die einen Datensatz aus der Datenquelle oder aus den Indextabellen enthalten

Wählen Sie **Alle Zeilen** aus. Die Umwandlung schreibt alle Cluster, die mindestens einen Datensatz aus der Datenquelle oder aus dem Indexdatenspeicher enthalten. Der Standardwert ist „Alle Zeilen“.

Da ein Cluster einen einzelnen Datensatz enthalten kann, enthält die Ausgabe alle Datensätze.

Schreiben aller Cluster, die einen Datensatz aus der Datenquelle enthalten

Wählen Sie **Neue und zugehörige Zeilen** aus. Die Umwandlung schreibt alle Cluster, die mindestens einen Datensatz aus der Datenquelle enthalten.

Da ein Cluster einen einzelnen Datensatz enthalten kann, enthält die Ausgabe alle Datensätze in der Datenquelle. Die Cluster können auch Datensätze aus den Indextabellen enthalten.

Schreiben aller Cluster aus der Datenquelle

Wählen Sie „Nur neue Zeilen“ aus. Die Umwandlung schreibt die Cluster, die Datensätze aus der Datenquelle enthalten. Die Ausgabe enthält keine Datensätze aus den Indextabellen.

Erweiterte Identitäts-Match-Eigenschaften

Die Umwandlung enthält erweiterte Eigenschaften, die die Anzahl der Ausführungsinstanzen und die Tracingebene für Protokolldaten bestimmen und festlegen, ob die Umwandlung identische Zeilen analysiert.

Sie können die folgenden erweiterten Eigenschaften konfigurieren:

Ausführungsinstanzen

Gibt die Anzahl der Threads an, die der Datenintegrationsdienst für die aktuelle Umwandlung zur Laufzeit zu erstellen versucht. Der Datenintegrationsdienst berücksichtigt den Wert „Ausführungsinstanzen“, wenn Sie die Laufzeiteigenschaft „Maximaler Parallelismus“ auf demjenigen Mapping überschreiben, das die Umwandlung enthält. Der Standardwert für Ausführungsinstanzen ist „Auto“.

Der Datenintegrationsdienst berücksichtigt mehrere Faktoren bei der Ermittlung der Anzahl von Threads, die der Umwandlung zugewiesen werden sollen. Die wichtigsten Faktoren sind der Wert „Ausführungsinstanzen“ und die Werte für das Mapping und für die zugeordneten Anwendungsdienste in der Domäne.

Der Datenintegrationsdienst, der das Mapping ausführt, liest die folgenden Werte, um die Anzahl der für die Umwandlung zu verwendenden Threads zu ermitteln:

- Der Wert *Maximaler Parallelismus* für den Datenintegrationsdienst. Standardwert ist 1.
- Alle *Maximaler Parallelismus*-Werte, die Sie auf der Mapping-Ebene festlegen. Standardwert ist „Auto“.
- Der Wert *Ausführungsinstanzen* für die Umwandlung. Standardwert ist „Auto“.

Wenn Sie den Wert „Maximaler Parallelismus“ auf der Mapping-Ebene überschreiben, versucht der Datenintegrationsdienst, den niedrigsten Wert innerhalb der Eigenschaften für das Ermitteln der Anzahl der Threads zu verwenden.

Wenn Sie den Standardwert für „Maximaler Parallelismus“ auf der Mapping-Ebene verwenden, ignoriert der Datenintegrationsdienst den Wert „Ausführungsinstanzen“.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie die Anzahl der Ausführungsinstanzen festlegen:

- Mehrere Benutzer führen möglicherweise gleichzeitige Mappings auf einem Datenintegrationsdienst aus. Um die korrekte Anzahl der Threads zu berechnen, dividieren Sie die Anzahl der CPUs, auf die der Datenintegrationsdienst zugreifen kann, durch die Anzahl gleichzeitiger Mappings.
- Wenn Sie die Standardwerte für „Ausführungsinstanzen“ und „Maximaler Parallelismus“ verwenden, sind die Umwandlungsoperationen nicht partitionierbar.

Filter für genaue Übereinstimmung

Legt fest, ob die Umwandlung den Vergleichsalgorithmus in einer Match-Strategie auf Paare mit identischen Datensätzen in den Eingabedaten anwendet.

Wenn die Umwandlung ein Paar mit identischen Datensätzen findet, muss der Algorithmus den Grad der Ähnlichkeit zwischen den Datensätzen nicht analysieren. Die Umwandlung kann die Datensätze direkt

ohne weitere Analyse an die Ausgabestufe übergeben. Um die Umwandlung für die Übergabe der identischen Datensätze direkt an die Ausgabestufe zu konfigurieren, wählen Sie die Option „Nach exakter Übereinstimmung filtern“ aus. Wenn die Eingabedaten viele identische Zeilen enthalten, führt der Vergleichsalgorithmus weniger Berechnungen durch und das Mapping wird schneller ausgeführt.

Wählen Sie die Option aus, wenn die Eingabedaten viele identische Zeilen enthalten. Wählen Sie die Option nicht aus, wenn die Eingabedaten nicht viele identische Zeilen enthalten, da die Umwandlung langsamer ausgeführt werden könnte.

Hinweis: Die Umwandlungsausgabe enthält dieselben Datensatzdaten, wenn Sie die Option auswählen oder deaktivieren. Die Umwandlung kann den Ausgabedatensätzen unterschiedliche Verknüpfungs-Scores zuweisen, wenn Sie diese Option auswählen oder deaktivieren.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Persistenter Index - Fallstudie

Sie arbeiten als Datenverwalter in einer Privatkundenbank mit mehreren Niederlassungen. Sie verwalten einer Mastersatz mit Kundenkonten-Datensätzen aus allen Niederlassungen. Sie verwenden einen Satz an Indexdatenbanktabellen, um sicherzustellen, dass die Kundenkonten-Datenbank keine redundanten oder doppelten Datensätze enthält.

Zum Erstellen und Verwalten des Indexdatenspeichers führen Sie die folgenden Vorgänge durch:

- Sie erstellen den Datenspeicher.
- Sie aktualisieren den Datenspeicher mit den aktuellen Daten aus den Niederlassungen der Bank.

  Sie fügen dem Datenspeicher unter Umständen Kontodaten hinzu oder aktualisieren die gegenwärtigen Daten im Datenspeicher.
- Sie entfernen veraltete Datensätze aus dem Datenspeicher.

Sie wissen, dass jeder Vorgang unter Umständen doppelte Datensätze im Datenspeicher erstellt. Sie entwickeln eine Richtlinie zur Analyse der Daten aus den Niederlassungen. Nach der Analyse fügen Sie die Daten dann zu den Daten im Masterdatenspeicher hinzu. Sie verwenden die Identitätsvergleichsanalyse zum Analysieren der Daten aus den Niederlassungen und um sicherzustellen, dass die Daten keine doppelten Identitäten im Datenspeicher erstellen. Sie konfigurieren die Optionen des persistenten Indexes in der Vergleichsumwandlung, um die Daten aus den Niederlassungen und den Datenspeicher zu analysieren.

Entwickeln einer Richtline zur Verwaltung persistenter Indexdaten

Als Datenverwalter definieren Sie eine Geschäftsregel, die besagt, dass der Datenspeicher für die Kundenkontendaten keine doppelten Identitäten enthalten darf. Sie entwerfen eine Identitätsvergleichszuordnung zum Analysieren der Niederlassungsdaten in einer Staging-Datenbank. Nach der Analyse fügen Sie die Daten dann zum Datenspeicher hinzu.

Die Vorgänge zum Hinzufügen der Daten aus den Niederlassungen zum Datenspeicher können in folgenden Fällen doppelte Identitäten erstellen:

- Die Daten aus den Niederlassungen enthalten doppelte Identitäten.
- Die Daten aus den Niederlassungen enthalten eine Identität, die im Index ebenfalls enthalten ist.
- Die Daten aus den Niederlassungen enthalten eine neuere Version einer Identität im Datenspeicher und die neuere Version entspricht einer anderen Identität im Index.

Wenn Sie die Staging-Datenbank mit dem Datenspeicher vergleichen, wählen Sie die Optionen des persistenten Indexes aus, die den Status der doppelten Datensätze der Niederlassungsdaten widerspiegeln. Vor dem Aktualisieren des Datenspeichers möchten Sie die Daten aus den Niederlassungen unter Umständen mit den Indexdaten vergleichen.

Hinweis: Sie können die Vergleichsanalyse für bestimmte Optionen aktivieren oder deaktivieren. Aktivieren Sie die Vergleichsanalyse, um die Zuordnungsdaten zu analysieren oder den Indexdatenspeicher mit den Zuordnungsdaten zu vergleichen. Deaktivieren Sie die Vergleichsanalyse, wenn Sie die Daten nicht vergleichen müssen. Sie können auch die Eigenschaften unter „Vergleich“ auf der Registerkarte „Vergleichsausgabe“ verwenden, um Daten in Vergleichsanalysen zu berücksichtigen oder aus diesen auszuschließen.

Vergleichen einer Zuordnungsdatenquelle mit dem Indexdatenspeicher

Wählen Sie die folgende Option aus, um die Zuordnungseingabedaten mit dem Indexdatenspeicher zu vergleichen und keine Änderungen am Datenspeicher vorzunehmen:

- Datenbank nicht aktualisieren

Die Zuordnung vergleicht die Eingabedaten mit dem Indexdatenspeicher. Die Zuordnung fügt keine Daten zum Indexdatenspeicher hinzu, noch entfernt oder aktualisiert sie Daten im Indexdatenspeicher.

Sie können die Identitätsvergleichsanalyse nicht deaktivieren, wenn Sie die Option auswählen.

Da Sie die Indexdaten nicht aktualisieren, können Sie keine doppelten Zeilen im Speicher erstellen. Wählen Sie aus den Eigenschaften unter „Vergleich“ auf der Registerkarte „Vergleichsausgabe“ die Option aus, die die aktuellen Bedürfnisse des Datenprojekts erfüllt. Wählen Sie beispielsweise die Option **Vollständig** aus. Mit der Option **Vollständig** wird sichergestellt, dass die Zuordnungsdaten keine Duplikate enthalten und dass die Zuordnungsdaten keine Duplikate zum Datenspeicher hinzufügen.

Hinweis: Verwenden Sie die Option zum Vergleichen der Zuordnungsdaten und des Datenspeichers, bevor Sie den Datenspeicher aktualisieren. Wenn die Zuordnungsausgabe angibt, dass die Zuordnungsdaten keine Duplikate zum Datenspeicher hinzufügen, führen Sie die Zuordnung erneut aus. Wählen Sie die Option zum Aktualisieren der Datenbank aus, wenn Sie die Zuordnung erneut ausführen.

Erstellen des Datenspeichers und Hinzufügen von Zeilen zum Datenspeicher

Wählen Sie die folgende Option aus, um einen Datenspeicher zu erstellen oder Zeilen aus den Zuordnungsdaten zum Datenspeicher hinzuzufügen:

- Datenbank mit neuen IDs aktualisieren

Die Zuordnung fügt eine Zeile zum Datenspeicher hinzu, wenn die Zeile keinen Sequenzbezeichner mit einer Zeile im Datenspeicher gemeinsam nutzt. Die Zuordnung überschreibt keine Zeilen in den Indextabellen. Wenn Sie leere Datenbanktabellen angeben, schreibt die Zuordnung alle Zuordnungsindexdaten in die Tabellen.

Sie können die Identitätsvergleichsanalyse aktivieren oder deaktivieren, wenn Sie die Option auswählen. Mit dieser Option werden Vergleichsanalysen standardmäßig aktiviert.

Da Sie die Indexzeilen nicht aktualisieren, wählen Sie die Option **Exklusiv** oder die Option **Teilweise** aus den Eigenschaften unter „Vergleich“ auf der Registerkarte „Vergleichsanalyse“ aus. Verwenden Sie die Option **Exklusiv**, wenn Sie die Eindeutigkeit der Zuordnungsdatenzeilen in einem früheren Prozess überprüft haben.

Aktualisieren der Zeilen im Datenspeicher

Wählen Sie die folgende Option aus, um eine aktuelle Zeile im Datenspeicher mit den Zuordnungsdaten zu aktualisieren:

- Aktuelle IDs in der Datenbank aktualisieren

Die Zuordnung aktualisiert einen aktuellen Datensatz im Datenspeicher, wenn der Datensatz einen Sequenzbezeichner mit einem Datensatz in den Zuordnungsdaten gemeinsam nutzt. Die Zuordnung fügt den Indextabellen keine Zeilen hinzu.

Sie können die Identitätsvergleichsanalyse aktivieren oder deaktivieren, wenn Sie die Option auswählen. Mit dieser Option werden Vergleichsanalysen standardmäßig deaktiviert.

Da Sie den Indextabellen keine Indexzeilen hinzufügen, wählen Sie die Option **Vollständig** aus den Eigenschaften unter „Vergleich“ auf der Registerkarte „Vergleichsanalyse“ aus.

Hinweis: Wenn Sie die Zeilen im Datenspeicher aktualisieren, gehen Sie davon aus, dass Duplikate zwischen den Zuordnungsquelldaten und dem Datenspeicher gefunden werden. Wählen Sie die Option **Vollständig** aus, um sicherzustellen, dass die von Ihnen zum Speicher hinzugefügten Identitätsdaten nicht mit den aktuellen Daten im Speicher übereinstimmen.

Entfernen von Zeilen aus dem Datenspeicher

Wählen Sie die folgende Option aus, um Zeilen aus dem Datenspeicher zu löschen:

- IDs aus der Datenbank entfernen

Die Zuordnung löscht eine Zeile aus dem Datenspeicher, wenn die Zeile einen Sequenzbezeichner mit einem Datensatz in den Zuordnungsdaten gemeinsam nutzt.

Sie können die Identitätsvergleichsanalyse aktivieren oder deaktivieren, wenn Sie die Option auswählen. Mit dieser Option werden Vergleichsanalysen standardmäßig deaktiviert.

Hinweis: Wenn Sie Daten aus einem Datenspeicher entfernen, ändern Sie die Beziehungen zwischen den Zeilen im Speicher. Wenn der Speicher doppelte Identitäten enthält, entfernen Sie gegebenenfalls Daten für einen Treiberdatensatz oder einen verknüpften Datensatz in einem Cluster. Sie können aber auch Daten für die genaueste Übereinstimmung in einem Vergleichspaar entfernen. Wenn Sie die Zuordnung erneut ausführen, erstellt die Zuordnung unter Umständen verschiedene Cluster oder doppelte Paare. Wenn Sie Zeilen aus einem Datenspeicher entfernen, der keine doppelten Datensätze enthält, können Sie den Duplikatsstatus der Datensätze nicht ändern. Wenn Sie die Zuordnung nach dem Löschen der Zeilen ausführen, erzeugt die Zuordnung dieselben Vergleichswerte für die Identitäten, die im Datensatz verbleiben.

Identitätsvergleichsanalyse - Beispiel

Sie arbeiten in der Personalabteilung eines Software-Unternehmens mit Entwicklungsabteilungen in verschiedenen Städten. Das Unternehmen speichert persönliche Datensätze für Mitarbeiter in einer Datenbank am Hauptsitz des Unternehmens. Die Entwicklungsabteilungen rekrutieren in regelmäßigen Abständen neue Mitarbeiter und senden die persönlichen Daten der Mitarbeiter an die Personalabteilung.

Sie geben die persönlichen Datensätze in eine Tabelle ein und verwenden die Daten aus der Datei zum Aktualisieren der Mitarbeiterdatenbank. Sie vermuten, dass die aktuelle Datei doppelte Identitäten enthält.

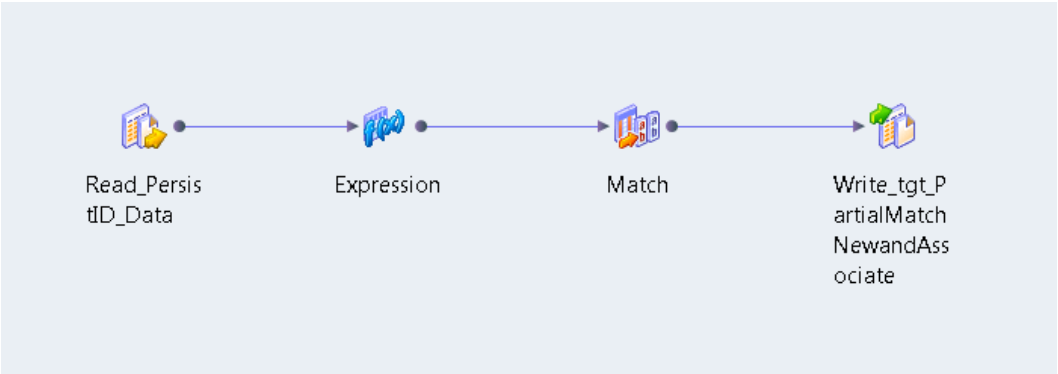
Sie entwerfen ein Mapping, um eine Identitätsanalyse für die Mitarbeiterdatensätze durchzuführen. Sie konfigurieren eine Vergleichsumwandlung, um nach doppelten Identitäten in der Tabelle zu suchen. Außerdem müssen Sie sicherstellen, dass die Dateidaten keine Mitarbeiterdaten in der Masterdatenbank duplizieren. Sie konfigurieren die Vergleichsumwandlung, um die Dateidaten mit den Masterdaten zu vergleichen, die Sie für die Mitarbeiter des Unternehmens speichern.

Da es sich bei der Datenbank um einen Masterdatensatz handelt, speichern Sie die Indexdaten für die Mitarbeiterdatensätze in einem persistenten Datenspeicher.

Erstellen des Mappings

Erstellen Sie ein Mapping, das nach duplizierten Identitäten sucht. Das Mapping liest eine Datenquelle, fügt einen Sequenzbezeichner für die Quelldatensätze hinzu, führt Identitätsanalysen aus und schreibt die Ergebnisse in ein Datenziel.

Die folgende Abbildung zeigt das Mapping im Developer-Tool:



Das von Ihnen erstellte Mapping enthält die folgenden Objekte:

Objektname	Beschreibung
Read_PersistID_Data	Datenquelle. Enthält den Mitarbeiternamen und Details.
Ausdruck	Ausdrucksumwandlung. Fügt den Quelldaten Werte für den Sequenzbezeichner hinzu.
Match	Match-Umwandlung. Analysiert das Ausmaß der Duplizierung in den Quelldatenidentitäten.
Write_tgt_PartialMatchNewandAssociate	Datenziel. Enthält die Ergebnisse der Identitätsanalyse.

Hinweis: Das Mapping verwendet nicht die Schlüsselgeneratorumwandlung. In der Identitätsvergleichsanalyse ist die Schlüsselgeneratorumwandlung optional.

Eingabedaten - Beispiel

Die Mitarbeiterdatei enthält den Namen des Mitarbeiters, die Stadt, in der der Mitarbeiter arbeitet, sowie die dem Mitarbeiter zugewiesene Rolle. Sie erstellen eine Datenquelle aus der Mitarbeiterdatei im Modellrepository. Sie fügen die Datenquelle zum Mapping hinzu.

Der folgende Datenausschnitt zeigt ein Beispiel der Mitarbeiterdaten in der Mitarbeiterdatei:

Name	Ort	Zuordnung
Chaithra	Bangalore	SE
Ramanan	Chennai	SSE

Name	Ort	Zuordnung
Ramesh	Chennai	SSE
Ramesh	Chennai	Lead
Sunil	Bangalore	Prinzipal
Venu	Hyderabad	Prinzipal
Harish	Bangalore	SE
Sachin	Bangalore	SSE

Konfiguration der Ausdrucksumwandlung

Beim Konfigurieren der Ausdrucksumwandlung verbinden Sie alle Datenquellenports, die Sie in der Mapping-Ausgabe einbeziehen möchten. Verbinden Sie die Ports als Pass-Through-Ports. Erstellen Sie einen Ausdruck zum Hinzufügen eines Werts für den Sequenzbezeichner mit den Ports.

Der folgende Ausdruck erstellt die Variable *Init3* und fügt jedem Sequenzbezeichner den Ganzzahlwert 1267 hinzu:

Init3+1267

In der folgenden Tabelle werden die Ports in der Ausdrucksumwandlung beschrieben, die die Mitarbeiterdatenquelle lesen:

Name	Porttyp	Portgruppe
SEQID	bigint	Nur Ausgabe
Name	string	Pass-Through
Ort	string	Pass-Through
Zuordnung	string	Pass-Through
Init3	integer	Variable

Konfiguration der Match-Umwandlung

Fügen Sie dem Mapping eine nicht wiederverwendbare Match-Umwandlung hinzu, um die Identitätsanalyse durchzuführen.

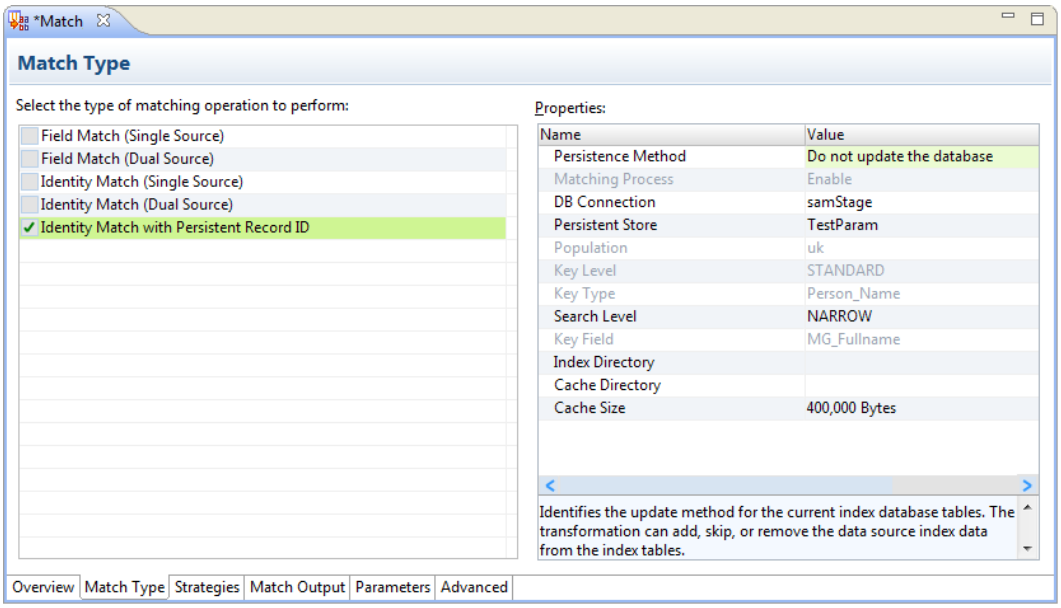
Führen Sie die folgenden Aufgaben durch, um die Umwandlung zu konfigurieren:

1. Wählen Sie den Typ der durchzuführenden Match-Analyse aus:
2. Verbinden Sie die Eingabeports mit der Umwandlung.
3. Konfigurieren Sie eine Strategie, um die Daten in den Datensätzen zu vergleichen.
4. Wählen Sie den Typ der Match-Ausgabedaten aus, die die Umwandlung erstellt.
5. Verbinden Sie die Ausgabeports mit einem Datenziel.

Auswählen des Typs des Match-Vorgangs

Verwenden Sie die Optionen in der Ansicht „Match-Typ“, um den Match-Vorgang auszuwählen.

Die folgende Abbildung zeigt die Ansicht „Match-Typ“:



Wählen Sie **Identitätsvergleich mit persistenter Datensatz-ID** aus, um die Indexdaten aus der Datenquelle mit den Indexdaten aus dem Masterdatensatz zu vergleichen. Aktualisieren Sie die Beibehaltungsmethode, sodass die Match-Analyse den Indextabellen keine Daten hinzufügt. Sie können angeben, ob die Indextabellen nach der Überprüfung der Zuordnungsergebnisse aktualisiert werden sollen.

Verwenden Sie die Option **Datenbankverbindung**, um die Datenbank zu identifizieren, die die Indextabellen enthält. Verwenden Sie die Option **Persistenter Cache**, um die Indextabellen auszuwählen.

Hinweis: Die Match-Umwandlung liest die Eigenschaftswerte von Identitätspopulation, Schlüsselebene, Schlüsseltyp und Schlüsselfeld aus den Metadaten in den Indexdatenbanktabellen. Die Werte stimmen mit den entsprechenden Eigenschaften in der Umwandlung überein, die den Indexdatenspeicher erstellt hat.

Verbinden der Eingabeports

Verbinden Sie die Datenports mit der Umwandlung. Stellen Sie sicher, dass die Portnamen, Portreihenfolge, Datentypen und Gesamtstellenanzahl des Ports den Portkonfigurationen der Umwandlung entsprechen, die den Datenspeicher erstellt hat.

Die Match-Umwandlung verwendet voreingestellte Eingabeports, um die Reihenfolge festzulegen, in denen die Datensätze verarbeitet werden. Die Umwandlung verwendet einen Sequenzbezeichner, um die Datensätze von den Eingabeports bis zu den abgeglichenen Paaren oder Clustern zu verfolgen, in die die Ausgabe geschrieben wird. Die Umwandlung verwendet einen Gruppenschlüssel zum Sortieren der Datensätze, die verarbeitet werden.

Verbinden Sie die voreingestellten Ports mit den folgenden Ports in der Ausdrucksumwandlung:

- SequenceID. Verbindung zum SEQID-Port aus der Ausdrucksumwandlung.
- GroupKey. Verbindung zum Ort-Port aus der Ausdrucksumwandlung.

Konfigurieren einer Strategie für die Identitätsanalyse

Verwenden Sie die Optionen in der Ansicht „Strategien“, um eine Strategie zu konfigurieren. Die Strategie bestimmt den Typ der Analyse, die die Umwandlung für die Datensatzdaten durchführt.

Wählen Sie den Person_Name-Algorithmus für die Datensatzdaten aus. Wählen Sie den Eingabeport „Name“ für die Analyse aus. Da die Umwandlung eine Kopie der Portdaten erstellt, wählen Sie die Ports Name_1 und Name_2 aus.

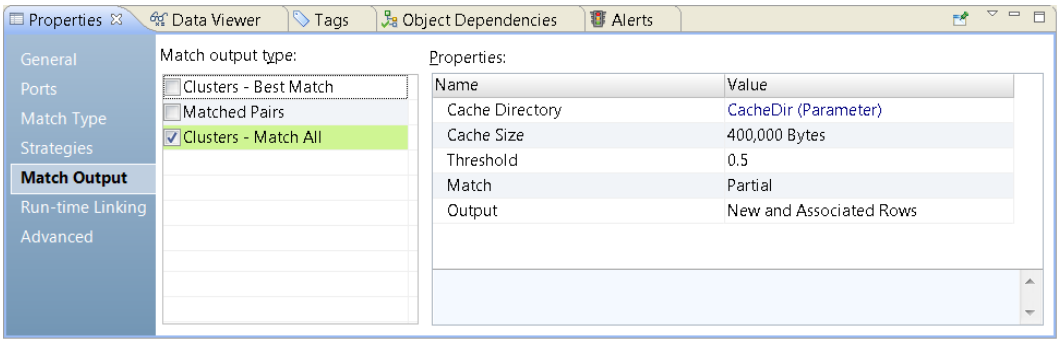
Verwandte Themen:

- “Identitäts-Match-Algorithmen” auf Seite 448

Auswählen des Match-Ausgabetyps

Verwenden Sie die Optionen in der Ansicht „Match-Ausgabe“, um das Ausgabeformat für die Ergebnisse der Match-Analyse zu definieren.

Die folgende Abbildung enthält die Ansicht „Match-Ausgabe“ für die Identitätsanalyse mit Einzelquelle:



Sie konfigurieren die Umwandlung, um die Ausgabedatensätze in Clustern zu organisieren. Jedes Cluster enthält alle Datensätze, die mit mindestens einem anderen Datensatz basierend auf den von Ihnen angegebenen Match-Eigenschaften übereinstimmen. Die Match-Eigenschaften bestimmen, wie die Umwandlung die Datenquellen-Datensätze mit den Indexdatensätzen vergleicht.

In der folgenden Tabelle werden die Optionen der Match-Eigenschaft beschrieben, die Sie zum Analysieren der Daten des Mitarbeiterdatensatzes angeben:

Match-Eigenschaft	Option	Optionsbeschreibung
Match	Teilweise	Die Umwandlung vergleicht die Datenquellen-Datensätze mit dem Indexdatenspeicher. Die Umwandlung vergleicht auch die Datensätze untereinander innerhalb Datenquelle.
Ausgabe	Neue und zugehörige Zeilen	Die Umwandlung schreibt alle Cluster, die mindestens einen Datensatz aus der Datenquelle enthalten. Die Cluster können Datensätze aus dem Indexdatenspeicher enthalten. Da ein Cluster einen einzelnen Datensatz enthalten kann, enthält die Ausgabe alle Datensätze in der Datenquelle.

Verwandte Themen:

- “Konfiguration der Match-Eigenschaft” auf Seite 452
- “Konfiguration der Ausgabe-Eigenschaft” auf Seite 452

Verbinden der Ausgabeports

Verbinden Sie die Ausgabeports der Vergleichsumwandlung mit dem Datenziel im Mapping. Wählen Sie die Ports aus, die die Datensatzdaten enthalten, die Sie ins Datenziel schreiben möchten.

Die Umwandlung enthält eine Reihe von voreingestellten Ports für die geclusterten Daten. Wählen Sie die voreingestellten Ports aus, die einen Duplikatsstatus der Datensätze angeben, und identifizieren Sie die Datenquelle, in denen die Datensätze gespeichert sind.

Die folgenden Ports enthalten Daten, die Sie verwenden können, um duplizierte Datensätze zu suchen und um die Quelle oder die Datensätze zu bestimmen:

- Der Port **ClusterSiz** gibt die Anzahl der Datensätze in einem Cluster an. Wenn ein Datensatz zu einem Cluster mit einer Clustergröße größer als 1 gehört, betrachtet die Umwandlung den Datensatz als Duplikat eines anderen Datensatzes.
- Der Port **ClusterID** identifiziert den Cluster, zu dem ein Datensatz gehört. Verwenden Sie die ClusterID-Daten, um Datensätze zu suchen, die Duplikate des aktuellen Datensatzes sind.
- Der Port **PersistenceStatus** verwendet einen Codewert, um die Beziehung zwischen den Indexdaten aus der Zuordnungsquelle und den Indexdaten im Datenspeicher zu beschreiben.
- Der Port **PersistenceStatusDesc** gibt eine Textbeschreibung der Werte im PersistenceStatus-Portcode zurück.

Sie können andere Ports verwenden, um die Beziehungen zwischen den Cluster-Datensätzen zu überprüfen. Die Verknüpfungs- und Treiber-Portwerte geben den Grad der Ähnlichkeit zwischen den Datensätzen in jedem Cluster an.

Im aktuellen Beispiel verbinden Sie alle Ports mit dem Datenziel. Um Ausgabedaten in den Ports anzuzeigen, führen Sie den Daten-Viewer aus.

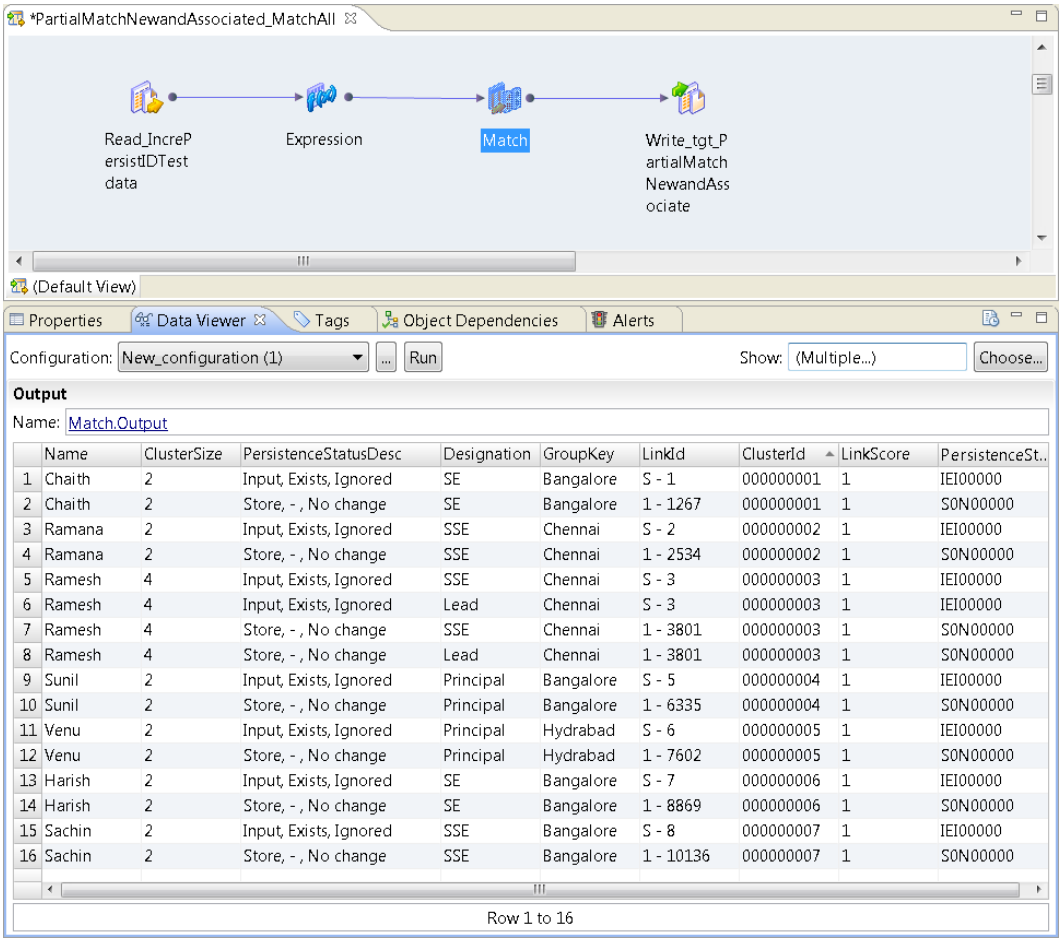
Verwandte Themen:

- "Persistenzstatuscodes und Persistenzstatusbeschreibungen" auf Seite 423
- "Werte für Statuscode und Statusbeschreibung" auf Seite 424

Ausführen des Daten-Viewers

Führen Sie den Daten-Viewer aus, um die Ergebnisse der Match-Analyse zu überprüfen. Der Daten-Viewer zeigt alle Ausgabeports in der Match-Umwandlung an. Wenn Sie das Mapping ausführen, können Sie das Datenziel mit den Daten aus den Ausgabeports aktualisieren.

Die folgende Abbildung zeigt die Ausgabedaten im Daten-Viewer:



	Name	ClusterSize	PersistenceStatusDesc	Designation	GroupKey	LinkId	ClusterId	LinkScore	PersistenceSt..
1	Chaith	2	Input, Exists, Ignored	SE	Bangalore	S - 1	000000001	1	IEI00000
2	Chaith	2	Store, - , No change	SE	Bangalore	1 - 1267	000000001	1	S0N00000
3	Ramana	2	Input, Exists, Ignored	SSE	Chennai	S - 2	000000002	1	IEI00000
4	Ramana	2	Store, - , No change	SSE	Chennai	1 - 2534	000000002	1	S0N00000
5	Ramesh	4	Input, Exists, Ignored	SSE	Chennai	S - 3	000000003	1	IEI00000
6	Ramesh	4	Input, Exists, Ignored	Lead	Chennai	S - 3	000000003	1	IEI00000
7	Ramesh	4	Store, - , No change	SSE	Chennai	1 - 3801	000000003	1	S0N00000
8	Ramesh	4	Store, - , No change	Lead	Chennai	1 - 3801	000000003	1	S0N00000
9	Sunil	2	Input, Exists, Ignored	Principal	Bangalore	S - 5	000000004	1	IEI00000
10	Sunil	2	Store, - , No change	Principal	Bangalore	1 - 6335	000000004	1	S0N00000
11	Venu	2	Input, Exists, Ignored	Principal	Hydrabad	S - 6	000000005	1	IEI00000
12	Venu	2	Store, - , No change	Principal	Hydrabad	1 - 7602	000000005	1	S0N00000
13	Harish	2	Input, Exists, Ignored	SE	Bangalore	S - 7	000000006	1	IEI00000
14	Harish	2	Store, - , No change	SE	Bangalore	1 - 8869	000000006	1	S0N00000
15	Sachin	2	Input, Exists, Ignored	SSE	Bangalore	S - 8	000000007	1	IEI00000
16	Sachin	2	Store, - , No change	SSE	Bangalore	1 - 10136	000000007	1	S0N00000

Der Daten-Viewer überprüft, ob die Datei Datensätze enthält, die die Daten im Master-Dataset duplizieren.

Beachten Sie die folgenden Datenwerte im Daten-Viewer:

- Jeder Datensatz im Dataset gehört zu einem Cluster, das zwei oder mehr Datensätze enthält. Daher ist jeder Datensatz ist ein Duplikat von mindestens einem anderen Datensatz. Die Umwandlung weist eindeutigen Datensätzen eine Cluster-Größe von 1 zu. Die Datenquelle enthält keine Datensätze, die nicht im Master-Dataset enthalten sind.
- Die **PersistenceStatusDesc**-Daten identifizieren den Ursprung des Datensatzes und geben an, wenn die Match-Umwandlung den Indextabellen den Datensatz hinzufügt. Die Spalte gibt an, dass jeder Eingabedatensatz im Master-Dataset enthalten ist. Die Umwandlung fügt keine Daten zum Master-Datenindex hinzu.

Ergebnis

Die Ergebnisse der Match-Analyse zeigen an, dass die Datei mit den Mitarbeiterdatensätzen keine Datensätze enthält, die nicht im Master-Dataset enthalten sind. Die Beschreibungen des Persistenzstatus zeigen an, dass

das Mapping die Indextabellen nicht mit Daten aus der Datenquelle aktualisiert. Sie verwerfen die Datei mit den Mitarbeiterdatensätzen.

Wenn Sie ein anderes Update von regionalen Niederlassungen erhalten, können Sie eine andere Datei erstellen und diese mit dem Master-Dataset vergleichen. Sie können das Mapping und die Indextabellen wiederverwenden. Da Sie die Indexdaten für das Master-Dataset in den Datenbanktabellen speichern, müssen Sie die Indexdaten nicht erneut erstellen.

KAPITEL 29

Merge-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Merge-Umwandlung - Übersicht, 464
- Konfigurieren einer Merge-Strategie, 464
- Merge-Umwandlung – Erweiterte Eigenschaften, 465

Merge-Umwandlung - Übersicht

Die Merge-Umwandlung ist eine passive Umwandlung, die die Datenwerte aus mehreren Eingabespalten liest und eine einzelne Ausgabespalte erstellt.

Verwenden Sie die Merge-Umwandlung zum Erstellen von Daten in einem bevorzugten Format. Sie können beispielsweise die Felder Customer_Firstname und Customer_Surname kombinieren, um ein Feld namens Customer_FullName zu erstellen.

Innerhalb einer Merge-Umwandlung können Sie mehrere Merge-Strategien erstellen. Die Merge-Umwandlung bietet einen Assistenten, den Sie zum Erstellen von Strategien verwenden.

Konfigurieren einer Merge-Strategie

Um eine Merge-Strategie zu konfigurieren, bearbeiten Sie die Einstellungen in der Ansicht **Strategien** der Standardisierungs-Umwandlung.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Klicken Sie auf **Neu**.

   Der Assistent **Neue Strategie** wird geöffnet.
3. Klicken Sie auf das Feld **Eingaben**, um Eingabeports für die Strategie auszuwählen.
4. Um die Merge-Zeichen zur Platzierung zwischen den zusammengeführten Elementen zu definieren, klicken Sie auf **Auswählen**. Wenn Sie kein Merge-Zeichen wählen, verwendet die Merge-Umwandlung standardmäßig ein Leerzeichen.
5. Wählen Sie optional **Leerstrings einschließen in zusammengeführter Ausgabe** aus, um leere Eingabestrings in die Ausgabe aufzunehmen.
6. Klicken Sie auf **Fertig stellen**.

Merge-Umwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Merge-Umwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 30

Normalizer-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Normalizer-Umwandlung - Übersicht, 466
- Mehrfach vorkommende Felder, 467
- Mehrfach vorkommende Datensätze, 468
- Definition der Eingabehierarchie, 469
- Ausgabegruppen und Ports der Normalizer-Umwandlung, 472
- Schlüsselgenerierung für Ausgabegruppen, 476
- Normalizer-Umwandlung - Erweiterte Eigenschaften, 476
- Erstellen einer Normalizer-Umwandlung, 477
- Erstellen einer Normalizer-Umwandlung aus einer vorgelagerten Quelle, 478
- Normalizer-Mapping - Beispiel, 478

Normalizer-Umwandlung - Übersicht

Die Normalizer-Umwandlung ist eine aktive Umwandlung, die eine Quellzeile in mehrere Zielzeilen umwandelt. Wenn eine Normalizer-Umwandlung eine Zeile empfängt, die mehrfach vorkommenden Daten enthält, gibt sie eine Zeile für jede Instanz der mehrfach vorkommenden Daten zurück.

Die Normalizer-Umwandlung parst mehrfach vorkommende Daten in separate Ausgabezeilen. Beispiel: Eine relationale Quellzeile kann vier Umsatzquartale enthalten. Die Normalizer-Umwandlung generiert eine separate Ausgabezeile für jedes Umsatzvorkommen.

Wenn Sie eine Normalizer-Umwandlung definieren, konfigurieren Sie eine Eingabezeilenhierarchie, die die Quelldatenstruktur beschreibt. Optional können Sie Datensätze in der Umwandlungseingabehierarchie definieren. Der Datensatz ist ein Container für eine Gruppe von Feldern. Definieren Sie einen Datensatz, wenn eine Gruppe von Feldern mehrmals in der Datenquelle vorkommt. Der Eingabehierarchie bestimmt, wie Sie die Umwandlungsausgabegruppen konfigurieren.

Die Normalizer-Umwandlung wandelt Daten aus relationalen Tabellen oder aus Einfachdateiquellen um.

Mehrfach vorkommende Felder

Wenn sich ein Feld in den Quelldaten mehrmals wiederholt, können Sie das Feld als mehrfach vorkommendes Feld in der Eingabezeilenhierarchie definieren. Die Normalizer-Umwandlung kann eine Zeile für jedes Vorkommen eines mehrfach vorkommenden Felds oder einer mehrfach vorkommenden Gruppe von Feldern in einer Quelle zurückgeben.

Eine Zeile enthält möglicherweise vier Umsatzquartale nach Filiale:

Filiale	Umsätze(1)	Umsätze(2)	Umsätze(3)	Umsätze(4)
Store1	100	300	500	700
Filiale2	250	450	650	850

Wenn Sie die Normalizer-Eingabehierarchie definieren, können Sie die vier Umsatzfelder ein einem mehrfach vorkommenden Feld kombinieren. Definieren Sie einen Feldnamen wie Qtr_Sales und konfigurieren Sie ihn so, dass er vier Mal in der Quelle vorkommt.

Wenn die Ausgabegruppe die Filialdaten und die Umsatzdaten enthält, gibt die Normalizer-Umwandlung eine Zeile für jede Kombination aus „Filiale“ und „Qtr_Sales“ zurück. Die Ausgabezeile enthält einen Index, der angibt, welche Instanz von Qtr_Sales sich in der Ausgabezeile befindet.

Die Umwandlung gibt die folgenden Zeilen zurück:

Filiale	Qtr_Sales	Qtr (GCID)
Store1	100	1
Store1	300	2
Store1	500	3
Store1	700	4
Filiale2	250	1
Filiale2	450	2
Filiale2	650	3
Filiale2	850	4

Wenn eine Ausgabegruppe einzeln vorkommende Spalten und eine mehrfach vorkommende Spalte enthält, gibt der Normalizer doppelte Daten für einzeln vorkommende Spalten in jeder Ausgabezeile zurück. Beispiel: „Filiale1“ und „Filiale2“ werden in jeder Umsatzinstanz wiederholt.

Eine Quellzeile kann mehr als eine Ebene mit mehrfach vorkommenden Daten enthalten. Sie können die Normalizer-Umwandlung so konfigurieren, dass separate Zeilen auf jeder Ebene basierend auf der Definition der Eingabehierarchie zurückgegeben werden.

Generierte Spalten-ID

Die Normalizer-Umwandlung gibt einen Ausgabeport mit generierter Spalten-ID (GCID) für jede Instanz eines mehrfach vorkommenden Felds zurück.

Der Port für die generierte Spalten-ID ist ein Index für die Instanz der mehrfach vorkommenden Daten. Wenn ein Wert beispielsweise vier Mal in einem Quelldatensatz vorkommt, gibt das Developer-Tool einen Wert von 1, 2, 3 oder 4 im Port für die generierte Spalten-ID basierend auf der Instanz der in der Zeile mehrfach vorkommenden Daten zurück.

Mehrfach vorkommende Datensätze

Sie können mehrfach vorkommende Datensätze in den Quelldaten der Normalizer-Umwandlung definieren. Datensätze sind Gruppen von Feldern. Definieren Sie Datensätze in der Normalizer-Umwandlung, wenn Sie Gruppen von mehrfach vorkommenden Quellfeldern definieren müssen.

Mehrfach vorkommende Datensätze – Beispiel

Die folgende Kundenzeile enthält Kundendaten mit Informationen zur Privat- und Geschäftsadresse.

```
CustomerID
FirstName
LastName
Home_Street
Home_City
Home_State
Home_Country
Business_Street
Business_City
Business_State
Business_Country
```

Wenn Sie die Normalizer-Umwandlung konfigurieren, können Sie eine Eingabestruktur definieren, die die Kundenfelder und einen mehrfach vorkommenden Adressdatensatz enthält. Der Adressdatensatz kommt zwei Mal vor. Wenn Sie die Ausgabegruppen für die Normalizer-Umwandlung konfigurieren, können Sie den Adressdatensatz an ein anderes Ziel als die Felder „CustomerID“, „FirstName“ und „LastName“ zurückgeben.

Das folgende Beispiel zeigt eine Eingabestruktur mit einem mehrfach vorkommenden Adressdatensatz:

```
CustomerID
FirstName
LastName
Address (occurs twice)
   Street
   City
   State
   Country
```

Teildatensätze sind Datensätze in Datensätzen. Wenn Sie Datensätze und Teildatensätze definieren, definieren Sie eine Hierarchie von Feldern in der Quellzeile. Jeder Datensatz ist ein Knoten in einer Hierarchie, auf den Sie beim Definieren der Umwandlungsausgabe verweisen können.

Beispiel: Der Quellzeile enthält möglicherweise mehrere Telefonnummern für jeden Adresstyp:

```
CustomerID
FirstName
LastName
Home_Street
Home_City
Home_State
Home_Country
Telephone_No
Cell_Phone_No
Alternate_Phone_No
Business_Street
Business_City
Business_State
Business_Country
Business_Telephone_No
Business_Cell_Phone_No
Business-Alternate_Phone1
```

Sie definieren eine Eingabehierarchie, in der die Adresse das übergeordnete Element der Telefonnummer ist. Wenn Sie die Ausgabe für die Normalizer-Umwandlung definieren, können Sie die Adressen und Telefonnummern an andere Ziele als die Kundendaten zurückgeben.

Definieren Sie eine Eingabehierarchie ähnlich dem folgenden Beispiel:

```
CustomerID
FirstName
LastName
Address  (occurs twice)
  Street
  City
  State
  Country
  Phone
    Telephone_No (occurs three times)
```

Definition der Eingabehierarchie

Wenn Sie eine Normalizer-Umwandlung erstellen, definieren Sie eine Eingabehierarchie, die Datensätze und Felder in der Quelle beschreibt. Definieren Sie die Eingabehierarchie in der Ansicht **Normalizer** der Umwandlung.

Das Developer Tool erstellt die Umwandlungseingabeports basierend auf den Feldern, die Sie in der Eingabehierarchie definieren. Definieren Sie die Eingabegruppenstruktur, bevor Sie die Umwandlungsausgabegruppen definieren.

Wenn Sie eine Eingabehierarchie definieren, müssen Sie eine Eingabestruktur definieren, die der Struktur der Quelldaten entspricht. Die Quelldaten können mehr als eine Gruppe mit mehrfach vorkommenden Feldern enthalten. Um die Struktur zu definieren, können Sie einen Datensatz konfigurieren, der auf derselben Ebene wie ein anderer Datensatz in der Quelle vorkommt. Wahlweise können Sie Datensätze definieren, die in anderen Datensätzen vorkommen.

Eingabehierarchie – Beispiel

Die folgende Quellzeile enthält benutzerdefinierte Felder und einen Adressdatensatz, der zwei Mal vorkommt:

```
CustomerID
  FirstName
  LastName
  Address
    Street
    City
    State
    Country
  Address1
    Street1
    City1
    State1
    Country1
```

Wenn Sie die Eingabestruktur in der Ansicht **Normalizer** definieren, können Sie „CustomerID“, „FirstName“ und „LastName“ als Felder hinzufügen. Definieren Sie einen Adressdatensatz und beziehen Sie die Felder „Street“, „State“ und „Country“ in die Adresse ein. Ändern Sie den Wert für „Address Occurs“ in 2.

In der folgenden Abbildung wird die Eingabehierarchie in der Ansicht **Normalizer** angezeigt:

Normalizer

Name	Level	Occurs	Type	Precision	Scale
CustomerID	1	1	string	10	0
FirstName	1	1	string	10	0
LastName	1	1	string	10	0
⊟ Address	1	2			
Street	2	1	string	10	0
City	2	1	string	10	0
State	2	1	string	10	0
Country	2	1	string	10	0

Die Spalte **Occurs** in der Ansicht **Normalizer** kennzeichnet die Anzahl der Instanzen eines Felds oder Datensatzes in einer Quellzeile. Ändern Sie den Wert in der Spalte **Occurs** für mehrfach vorkommende Felder oder Datensätze. In diesem Beispiel kommen die benutzerdefinierten Felder einmal und der Adressdatensatz zweimal vor.

Die Spalte **Level** in der Ansicht **Normalizer** gibt an, wo ein Feld oder ein Datensatz in der Eingabehierarchie angezeigt wird. Die benutzerdefinierten Felder befinden sich auf Ebene 1 in der Hierarchie. Der Adressdatensatz befindet sich ebenfalls auf Ebene 1.

Eingabeports für die Normalizer-Umwandlung

Das Developer-Tool erstellt die Eingabeports für die Normalizer-Umwandlung, wenn sie die Eingabehierarchie in der Ansicht **Normalizer** definieren. Wenn Sie die Felder in der Eingabehierarchie ändern, ändert das Developer-Tool die Eingabeports.

Zeigen Sie die Eingabeports für die Normalizer-Umwandlung in der Ansicht **Übersicht** an. Sie können die Eingabeports in der Ansicht **Übersicht** neu anordnen. Um die Eingabeports zu ändern, aktualisieren Sie die Eingabehierarchie in der Ansicht **Normalizer**.

Wenn Sie ein Feld als mehrfach vorkommend in der Eingabehierarchie definieren, erstellt das Developer-Tool einen Eingabeport für jede Instanz des mehrfach vorkommenden Felds. Wenn ein Datensatz mehrfach vorkommt, erstellt das Developer-Tool einen Eingabeport für jede Instanz des Felds im Datensatz.

Eingabeports – Beispiel

In der folgenden Abbildung werden die Eingabeports angezeigt, die das Developer-Tool für Kundendaten und mehrfach vorkommende Adressdaten erstellt:

Ports

	Name	Type	Precision	Scale	Location
	Input				
1	CustomerID	string	10	0	CustomerID
2	FirstName	string	10	0	FirstName
3	LastName	string	10	0	LastName
4	Street	string	10	0	Address.Street
5	Street1	string	10	0	Address.Street
6	City	string	10	0	Address.City
7	City1	string	10	0	Address.City
8	State	string	10	0	Address.State
9	State1	string	10	0	Address.State
10	Country	string	10	0	Address.Country
11	Country1	string	10	0	Address.Country

Felder zusammenführen

Sie können Felder mit ähnlichen Daten in einem einzelnen mehrfach vorkommenden Feld in der Ansicht **Normalizer** zusammenführen. Möglicherweise müssen Sie Felder zusammenführen, wenn Sie Ports aus einem anderen Objekt ziehen, um eine Normalizer-Umwandlung in einem Mapping zu erstellen.

Eine Quellzeile kann mehrere Felder mit verschiedenen Gehaltsdatentypen enthalten, zum Beispiel „Base_Salary“, „Bonus_Pay“ und „Sales_Commissions“. Sie können die Felder zusammenführen, um ein Salary-Feld zu erstellen, das dreimal vorkommt.

In der folgenden Abbildung wird eine Mitarbeiterzeile mit drei in der Normalizer-Ansicht ausgewählten Gehaltstypen dargestellt:

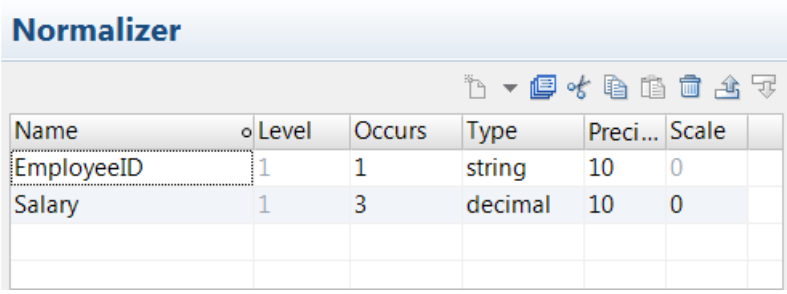
Normalizer

Name	Level	Occurs	Type	Preci...	Scale
EmployeeID	1	1	string	10	0
Base_Salary	1	1	decimal	10	0
Bonus_Pay	1	1	decimal	10	0
Sales_Commissions	1	1	decimal	10	0

Overview | Normalizer | Advanced

Sie können die drei Gehaltsdatentypen in einem Salary-Feld zusammenführen, das dreimal vorkommt.

In der folgenden Abbildung wird das Salary-Feld dargestellt:



Zusammenführen von Feldern

Führen Sie die Felder desselben Typs in ein mehrfach vorkommendes Feld in der Normalizer-Ansicht zusammen.

1. Klicken Sie auf die Ansicht **Normalizer**.
2. Wählen Sie die Felder aus, die Sie zusammenführen möchten.
3. Klicken Sie auf die Schaltfläche **Zusammenführen**.

   Das Dialogfeld **Felder zusammenführen** wird angezeigt.
4. Geben Sie einen Namen für das Zusammenführungsfeld, den Typ, die Genauigkeit, die Größenordnung und das Vorkommen des mehrfach vorkommenden Felds ein.
5. Klicken Sie auf **OK**.

Ausgabegruppen und Ports der Normalizer-Umwandlung

Definieren Sie Ausgabegruppen und Ports in der Ansicht „Übersicht“ der Normalizer-Umwandlung. Sie können die Ausgabegruppen nach dem Definieren der Umwandlungseingabehierarchie definieren.

Das Developer-Tool generiert standardmäßig mindestens eine Ausgabegruppe. Die Ausgabegruppe enthält alle Felder der Ebene 1 aus den Eingabeports und das erste mehrfach vorkommende Feld. Wenn Sie mehr als ein mehrfach vorkommendes Feld in der Normalizer-Ansicht definieren, erstellt das Developer-Tool eine Ausgabegruppe für jedes weitere mehrfach vorkommende Feld. Sie können verhindern, dass das Developer-Tool weitere Ausgabegruppen generiert, indem Sie die erweiterte Eigenschaft `Ausgabegruppen der ersten Ebene generieren` deaktivieren.

Die folgende Quellzeile enthält die Kundendaten, das mehrfach vorkommende Umsatzfeld und das mehrfach vorkommende Telefonfeld:

```
CustomerID
LastName
FirstName
Sales (occurs 4 times)
Phone (occurs 3 times)
```

Das Developer-Tool erstellt zwei Ausgabegruppen aus der Eingabestruktur:

```
Output
CustomerID
LastName
```

```
FirstName
Sales
```

```
Output1
Phone
GCID_Phone
```

Das Developer-Tool erstellt eine Ausgabe1-Gruppe, weil die Quelldaten mehr als ein mehrfach vorkommendes Feld beinhalten. Das Developer-Tool erstellt keine Gruppen für Felder, die Sie in Datensätzen definieren. Wenn Sie Datensätze definieren, müssen Sie die Ausgabegruppen definieren, die Felder in den Datensätzen enthalten.

Die folgende Quellzeile enthält benutzerdefinierte Felder und einen Adressdatensatz, der zwei Mal vorkommt:

```
CustomerID
  FirstName
  LastName
  Address
    Street
    City
    State
    Country
  Address1
    Street1
    City1
    State1
    Country1
```

Das Developer-Tool erstellt eine Ausgabegruppe, die die folgenden Felder enthält.

```
CustomerID
FirstName
LastName
```

Das Developer-Tool erstellt eine Standardausgabegruppe für benutzerdefinierte Felder der Ebene 1. Die standardmäßige Ausgabegruppe enthält den Adressdatensatz nicht. Sie müssen konfigurieren, wie die Adressdaten in die Ausgabe zurückgegeben werden sollen.

Erstellen Sie die Ausgabegruppen basieren darauf, wie Sie die Ausgabezeilen strukturieren müssen. Wenn die Quellzeile Kundendaten und Adressdaten enthält, können Sie eine Ausgabegruppe für benutzerdefinierte Felder erstellen. Sie können eine andere Ausgabegruppe für die Adressfelder erstellen. Wahlweise können Sie die Standardausgabegruppe aktualisieren und ihr die Adressfelder hinzufügen. Die folgenden Beispiele zeigen unterschiedliche Ausgabeergebnisse basierend auf der Konfiguration der Ausgabegruppe.

Erstellen einer Ausgabegruppe

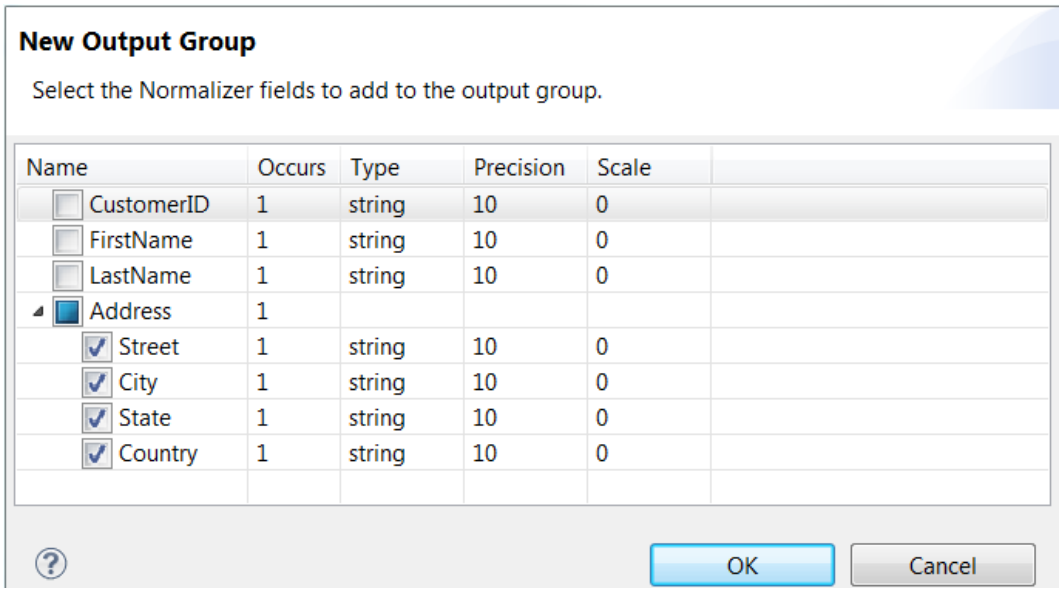
Erstellen Sie eine Ausgabegruppe in der Ansicht **Übersicht** der Normalizer-Umwandlung.

Wenn Sie die Ansicht **Übersicht** der Normalizer-Umwandlung öffnen, zeigt das Developer-Tool die Standardgruppen an, die es aus der Eingabehierarchie erstellt.

Wenn Sie eine neue Ausgabegruppe erstellen, wird in einem Dialogfeld eine Liste der Felder und Datensätze in der Eingabehierarchie angezeigt. Wählen Sie aus, welche Felder oder Datensätze Sie in die Gruppe einbeziehen möchten.

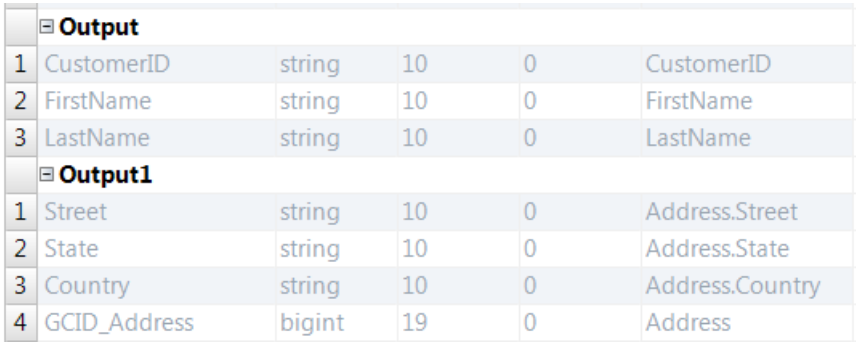
Ausgabegruppe – Beispiel

In der folgenden Abbildung wird das Dialogfeld **Neue Ausgabegruppe** angezeigt:



Wenn Sie den Adressdatensatz auswählen, erstellt das Developer-Tool eine Gruppe von Ausgabeports, die den Feldern im Adressdatensatz entsprechen. Die Gruppe „Ausgabe1“ enthält die Ports „Street“, „City“, „State“ und „Country“. Sie können die Ports in der Ausgabegruppe ändern.

In der folgenden Abbildung werden die Gruppe „Ausgabe“ und die Gruppe „Ausgabe1“ in der Ansicht **Übersicht** angezeigt:



	Output				
1	CustomerID	string	10	0	CustomerID
2	FirstName	string	10	0	FirstName
3	LastName	string	10	0	LastName
	Output1				
1	Street	string	10	0	Address.Street
2	State	string	10	0	Address.State
3	Country	string	10	0	Address.Country
4	GCID_Address	bigint	19	0	Address

Sie können die Normalizer-Umwandlung so konfigurieren, dass die Zeilen aus der Gruppe „Ausgabe“ an eine Kundentabelle zurückgegeben werden.

Die Kundentabelle empfängt Daten ähnlich der folgenden Zeilen:

```
100, Robert, Bold
200, James, Cowan
```

Sie können die Zeilen aus der Gruppe „Ausgabe1“ an die Adresstabelle zurückgeben. Die Adresstabelle empfängt die Straße, die Stadt, das Bundesland, das Land und die GCID.

Die Adresstabelle empfängt Daten ähnlich der folgenden Zeilen:

```
100 Summit Dr, Redwood City, CA, United States,1
41 Industrial Way, San Carlos, CA, United States,2
85 McNulty Way, Los Angeles, CA, United States,1
55 Factory Street, Los Vegas, NV, United States,2
```

Die GCID gibt an, welche Instanz der Kundenadresse sich in der Ausgabezeile befindet. In diesem Beispiel gibt die Normalizer-Umwandlung zwei Instanzen des Adressdatensatzes zurück. Jede Ausgabezeile enthält einen GCID-Wert von 1 oder 2.

Aktualisieren einer Ausgabegruppe

Sie können eine Ausgabegruppe der Normalizer-Umwandlung aktualisieren. Sie können die Felder in der Gruppe hinzufügen oder entfernen.

Standardmäßig erstellt das Developer-Tool Ausgabegruppen der Stufe 1, wenn Sie eine Eingabehierarchie definieren. Das Developer-Tool enthält keine Datensätze in den Gruppen. Sie können die Standardausgabegruppen aktualisieren und diesen Datensätze hinzufügen.

Um eine Ausgabegruppe zu aktualisieren, heben Sie den Gruppennamen hervor und klicken Sie auf **Neu** > **Gruppe aktualisieren**. Im Dialogfeld **Ausgabegruppe bearbeiten** werden die Felder in der Eingabehierarchie angezeigt. Wählen Sie aus, welche Felder die Gruppe enthalten soll.

Aktualisieren der Ausgabegruppe – Beispiel

Im vorherigen Beispiel hat das Developer-Tool eine Standardausgabegruppe mit den Feldern „CustomerID“, „FirstName“ und „LastName“ erstellt.

In der folgenden Abbildung wird die Standardausgabegruppe angezeigt:

	Output				
1	CustomerID	string	10	0	CustomerID
2	FirstName	string	10	0	FirstName
3	LastName	string	10	0	LastName

Sie können die Standardausgabegruppe aktualisieren und ihr den Adressdatensatz hinzufügen.

In der folgenden Abbildung wird das Dialogfeld **Ausgabegruppe bearbeiten** angezeigt:

Edit Output Group

Select the Normalizer fields to include in the output group.

Name	Occurs	Type	Precision	Scale
CustomerID	1	string	10	0
FirstName	1	string	10	0
LastName	1	string	10	0
Address	2			
Street	1	string	10	0
State	1	string	10	0
Country	1	string	10	0

In diesem Beispiel handelt es sich bei den Knoten der Ebene 1 um „CustomerID“, „FirstName“ und „LastName“. Der Adressdatensatz ist ebenfalls ein Knoten der Ebene 1. Die Normalizer-Umwandlung kann „Adresse“ in derselben Zeile wie die Benutzerdaten zurückgeben. Da „Adresse“ mehrfach vorkommt, fügt das Developer-Tool den GCID_Address-Index zur Ausgabegruppe hinzu.

In der folgenden Abbildung werden die Ports in der Ausgabegruppe angezeigt:

	Output				
1	CustomerID	string	10	0	CustomerID
2	FirstName	string	10	0	FirstName
3	LastName	string	10	0	LastName
4	Street	string	10	0	Address.Street
5	State	string	10	0	Address.State
6	Country	string	10	0	Address.Country
7	GCID_Address	bigint	19	0	Address

Wenn sich die benutzerdefinierten Felder und die mehrfach vorkommenden Adressfelder in der Ausgabegruppe befinden, gibt die Normalizer-Umwandlung dieselben benutzerdefinierten Felder für jede Instanz der Adressdaten zurück.

Im folgenden Beispiel werden die Zeilen angezeigt, die die Normalizer-Umwandlung aus der Ausgabegruppe generiert.

```
100, Robert, Bold, 100 Summit Dr, Redwood City, CA, United States,1
100, Robert, Bold, 41 Industrial Way, San Carlos, CA, United States,2
200, James, Cowan, 85 McNulty Way, Los Angeles, CA, United States,1
200, James, Cowan, 55 Factory Street, Los Vegas, NV, United States,2
```

Der GCID-Port enthält die Adressinstanznummer. Der GCID-Wert lautet 1 oder 2.

Schlüsselgenerierung für Ausgabegruppen

Sie können eine Sequenzgeneratorumwandlung zum Generieren von Schlüsseln konfigurieren, die jede Ausgabezeilen verknüpft, die die Normalizer-Umwandlung aus derselben Quellzeile zurückgibt.

Sie können eine Sequenzgeneratorumwandlung vor der Normalizer-Umwandlung zu einem Mapping hinzufügen. Die Sequenzgeneratorumwandlung fügt jeder Quellzeile eine Sequenznummer hinzu. Wenn die Normalizer-Umwandlung mehrere Ausgabegruppen oder Zeilen aus derselben Quellezeile zurückgibt, erhält jede Ausgabezeile dieselbe Sequenznummer. Sie können die Anzahl als einen Schlüssel in einer Beziehung zwischen Primärschlüssel und Fremdschlüssel zwischen Zieltabellen verwenden.

Beispiel: Die Normalizer-Umwandlung gibt Kundendaten in einer Ausgabegruppe und Auftragsinformationen in einer anderen Ausgabegruppe zurück. Sie können die Sequenznummer verwenden, um die Kundendaten in einer Tabelle mit den Auftragsdaten in einer anderen Tabelle zu verknüpfen.

Normalizer-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften für die Normalizer-Umwandlung auf der Registerkarte **Erweitert**.

Konfigurieren Sie die folgenden Eigenschaften auf der Registerkarte **Erweitert**:

Ausgabegruppen der ersten Ebene automatisch generieren

Generiert die Ausgabegruppen automatisch, wenn die Normalizer-Umwandlung mehr als eine mehrfach vorkommende Gruppe enthält.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Generieren von Ausgabegruppen der ersten Ebene

Die Ausgabegruppe der ersten Ebene der Normalizer-Umwandlung wird vom Developer-Tool standardmäßig generiert. Das Developer Tool generiert weitere Ausgabegruppen, wenn Sie mehr als ein mehrfach vorkommendes Feld in der Normalizer-Ansicht definieren.

Das Developer-Tool erstellt eine Ausgabegruppe, die alle einzeln vorkommenden Felder der Ebene 1 und alle mehrfach vorkommenden Felder der Ebene 1 enthält. Das Developer-Tool erstellt eine zusätzliche Ausgabegruppe, wenn die Umwandlung mehr als ein mehrfach vorkommendes Feld enthält.

Das Developer-Tool erstellt keine Ausgabegruppe für Datensätze der Ebene 1. Wenn Sie einen Datensatz der Ebene 1 definieren, müssen Sie die jeweilige Platzierung in der Ausgabe konfigurieren.

Um zu verhindern, dass das Developer-Tool eine Standardausgabegruppe erstellt, deaktivieren Sie die erweiterte Eigenschaft **Ausgabegruppen der ersten Ebene automatisch generieren**.

Erstellen einer Normalizer-Umwandlung

Sie können eine wiederverwendbare oder nicht wiederverwendbare Normalizer-Umwandlung erstellen. Wiederverwendbare Umwandlungen können in mehreren Mappings vorhanden sein. Nicht wiederverwendbare Umwandlungen sind in einem einzelnen Mapping vorhanden.

1. Mithilfe einer der folgenden Methoden können Sie eine Umwandlung erstellen:

Option	Beschreibung
Wiederverwendbar	Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**. Wählen Sie die Normalizer-Umwandlung aus und klicken Sie auf **Weiter**.
Nicht wiederverwendbar	Ziehen Sie in einem Mapping oder Mapplet eine Normalizer-Umwandlung aus der Palette **Umwandlung** in den Editor. Sie können Ports aus Quelldatenobjekten oder aus Umwandlungen im Mapping ziehen, um die Umwandlung zu definieren.

   Der Assistent **Neue Normalizer-Umwandlung** wird angezeigt.
2. Geben Sie einen Namen für die Umwandlung ein.
3. Klicken Sie auf **Weiter**.

   Die Seite **Normalizer-Definition** wird angezeigt.
4. Um einen Datensatz hinzuzufügen, klicken Sie auf die Schaltfläche **Neu** und wählen Sie dann die Option **Datensatz** aus.
5. Um ein Feld hinzuzufügen, klicken Sie auf die Schaltfläche **Neu** und wählen Sie dann die Option **Feld** aus.

   Um einem Datensatz ein Feld hinzuzufügen, müssen Sie den Datensatz auswählen, bevor Sie das Feld hinzufügen.
6. Doppelklicken Sie optional auf den Wert in der Spalte **Häufigkeit**, um das Vorkommen eines Felds oder Datensatzes zu ändern.

7. Klicken Sie auf **Weiter**.

   Die Seite **Normalizer-Ports** wird angezeigt.
8. Um eine Ausgabegruppe hinzuzufügen, klicken Sie auf die Schaltfläche **Neu** und wählen Sie dann **Neue Ausgabegruppe** aus.
9. Um eine Ausgabegruppe zu bearbeiten, wählen Sie die Ausgabegruppe aus, die Sie bearbeiten möchten. Klicken Sie auf die Schaltfläche **Neu** und wählen Sie dann die Option **Ausgabegruppe bearbeiten** aus.
10. Klicken Sie auf **Fertig stellen**.

    Die Umwandlung wird im Editor angezeigt.

Erstellen einer Normalizer-Umwandlung aus einer vorgelagerten Quelle

Sie können eine leere Normalizer-Umwandlung erstellen und die Ports aus einem Quelldatenobjekt oder aus einer Umwandlung in die Normalizer-Umwandlung ziehen, um Eingabe und Ports zu erstellen.

1. Erstellen Sie ein Mapping, das die Quelle oder Umwandlung enthält, um die Quelldaten an die Normalizer-Umwandlung zu übergeben.
2. Um die Normalizer-Umwandlung zu erstellen, wählen Sie die Normalizer-Umwandlung aus der Umwandlungspalette aus und ziehen Sie die Umwandlung in den Editor.

   Das Dialogfeld „Normalizer“ wird angezeigt.
3. Klicken Sie auf **Fertig stellen**, um eine leere Umwandlung zu erstellen.
4. Wählen Sie Ports aus einer Quelle oder Umwandlung im Mapping aus und ziehen Sie sie in die Normalizer-Umwandlung.

   Die Eingabe- und Ausgabeports werden in der Normalizer-Umwandlung angezeigt. Das Developer-Tool erstellt eine Eingabegruppe und eine Ausgabegruppe.
5. Öffnen Sie die Ansicht **Normalizer**, um die Standardgruppe zu aktualisieren und die Felder je nach Bedarf in Datensätzen zu organisieren.
6. Um mehrere Felder in einem einzelnen mehrfach vorkommenden Feld zusammenzuführen, wählen Sie die Felder in der Ansicht **Normalizer** aus und klicken Sie auf die Option **Zusammenführen**.

   Wählen Sie einen Namen für das mehrfach vorkommende Feld aus.

Normalizer-Mapping - Beispiel

Eine Einzelhandelsorganisation erhält die Gesamtumsätze für die Unternehmensfilialen. Das Unternehmen erhält eine Zeile mit Daten, die die Filialdaten und vier Verkaufsbeträge enthält. Jeder Verkaufsbetrag stellt den Gesamtumsatz für ein Quartal des Jahres dar.

In folgendem Beispiel wird dargelegt, wie eine Normalizer-Umwandlung definiert werden muss, um die Umsatzdaten an ein Filialziel und ein Umsatzziel zurückzugeben. Das Filialziel empfängt eine Zeile für jede Filiale. Das Umsatzziel empfängt vier Zeilen von jeder Filiale. Jede Zeile enthält ein Quartal mit Umsatzdaten.

Eine Sequenzgeneratorumwandlung erzeugt eine eindeutige ID für jede Filiale. Die Normalizer-Umwandlung gibt die StoreID mit jeder Ausgabezeile zurück.

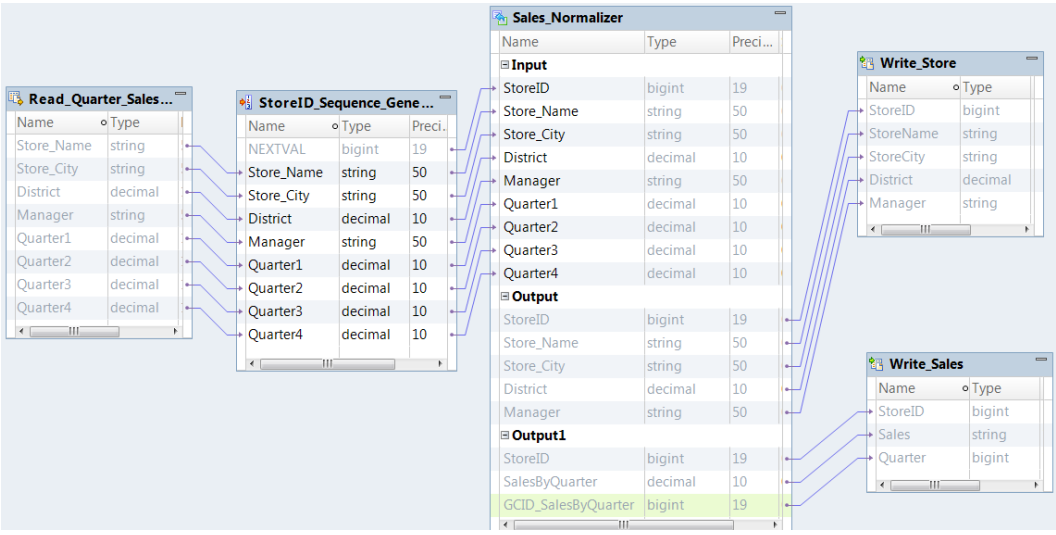
Erstellen Sie ein Mapping mit einem Lesedatenobjekt, einer Sequenzgeneratorumwandlung, der Normalizer-Umwandlung und zwei Schreibdatenobjekten.

Beispiel für Normalizer-Mapping

Erstellen Sie ein Mapping, das die Normalizer-Umwandlung enthält, um mehrfach vorkommenden Quartalsumsatz aus einer Einfachdateiquelle zu normalisieren.

Die Normalizer-Umwandlung generiert eine separate Ausgabezeile für alle Quartalsumsätze und schreibt die normalisierten Verkaufsbeträge in ein Umsatzziel. Die Normalizer-Umwandlung schreibt die Filialdaten in ein Filialziel.

Die folgende Abbildung zeigt das Mapping der Normalizer-Umwandlung:



Das Mapping enthält die folgenden Objekte:

Read_STORE

Eine Datenquelle, die mehrfach vorkommende Felder enthält.

StoreID-Sequenzgeneratorumwandlung

Eine Sequenzgeneratorumwandlung, die einen storeID-Schlüssel generiert, um eine Verknüpfung zwischen der Filialtabelle und der Umsatztabelle herzustellen.

Sales_Normalizer

Eine Normalizer-Umwandlung, die die mehrfach vorkommenden Umsatzdaten normalisiert.

Write_Store

Ein Ziel, das die Filialdaten aus der Normalizer-Umwandlung empfängt.

Write_Sales

Ein Ziel, das die Umsatzzahlen aus der Normalizer-Umwandlung erhält.

Beispiel für die Normalizer-Definition

Die Quelle ist eine Einfachdatei, die Informationen und Quartalsumsatzdaten enthält. Definieren Sie die Struktur der Quelldaten in der Ansicht **Normalizer**.

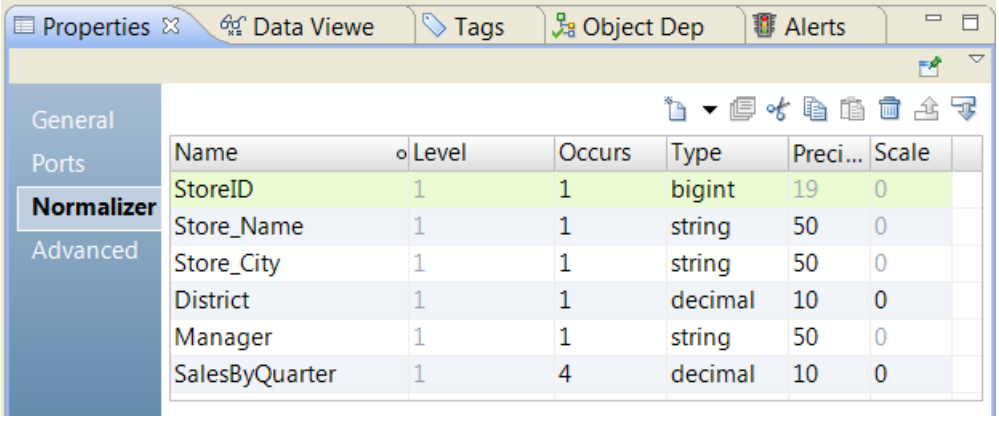
Die SPEICHER-Einfachdatei enthält die folgenden Quelldaten:

StoreID	Store_Name	Store_City	Kreis	Manager	Quartal1	Quartal2	Quartal3	Quartal4
1	BigStore	New York	East	Robert	100	300	500	700
2	SmallStore	Phoenix	West	Radhika	250	450	650	850

Fügen Sie die Einfachdatei zu einem Mapping als Lese-Datenobjekt hinzu und erstellen Sie eine leere Normalizer-Umwandlung. Ziehen Sie die Ports aus dem Read_STORE-Datenobjekt in die Normalizer-Umwandlung, um die Normalizer-Definition zu erstellen.

Die Ansicht **Normalizer** enthält eine Instanz der Felder „Store_Name“, „Store_City“, „Bezirk“ und „Manager“. Die Ansicht **Normalizer** enthält vier Instanzen eines Felds namens QUARTAL. Führen Sie die QUARTAL-Felder zusammen, um ein SalesByQuarter-Feld zu erstellen, das viermal vorkommt.

In der folgenden Abbildung wird die Normalizer-Definition mit zusammengeführten Quartalsfeldern angezeigt:



Beispiel für Normalizer-Eingabe- und Ausgabegruppen

Nachdem Sie die Eingabehierarchie geändert haben, weist die Normalizer-Umwandlung eine Eingabegruppe und eine Standardausgabegruppe auf. Sie müssen die Ausgabeports in zwei Gruppen neu organisieren. Sie benötigen eine Gruppe mit den Filialinformationen und eine Gruppe mit den Umsatzinformationen.

Die Eingabegruppe enthält einen Port für jedes Feld in der Quelle. Die Ausgabegruppe enthält Ports für die Filialfelder und einen Port für das mehrfach vorkommende SalesByQuarter-Feld. Die Ausgabegruppe enthält auch die generierte Spalten-ID `GCID_SalesByQuarter`, die dem mehrfach vorkommenden SalesByQuarter-Feld entspricht.

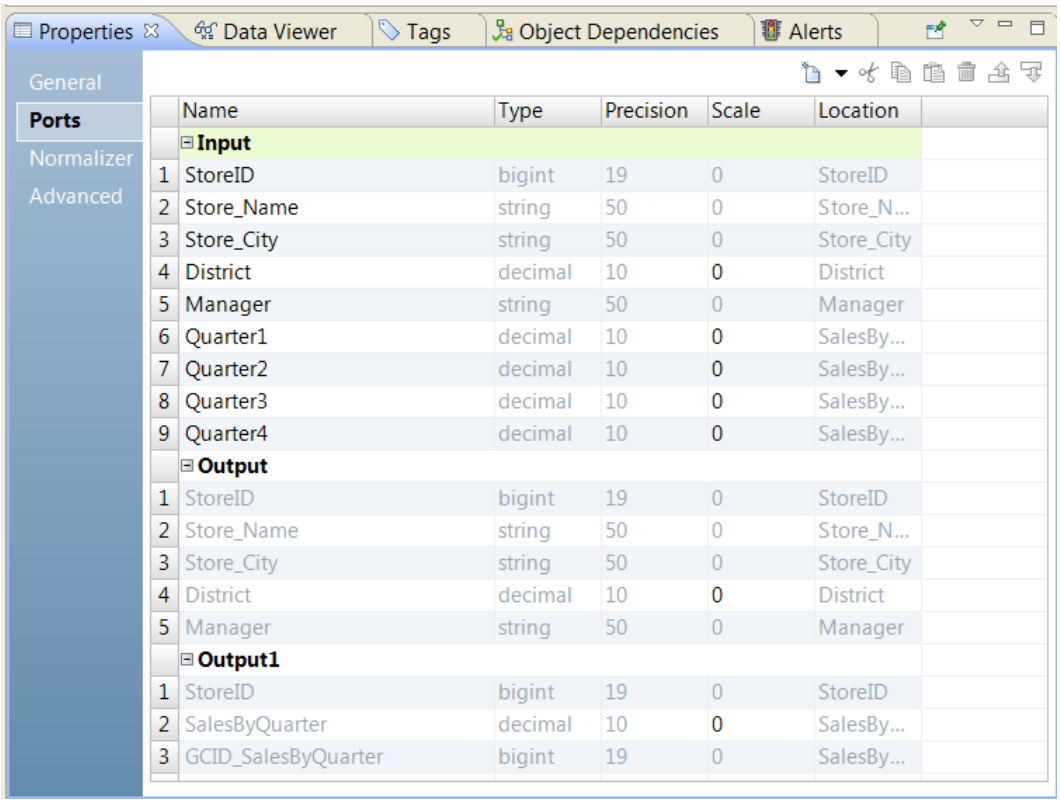
Um die Quartalsumsätze an ein anderes Ziel zurückzugeben, erstellen Sie eine neue Gruppe in der Ansicht **Übersicht**. Fügen Sie in der Ausgabe1-Gruppe die folgenden Felder hinzu:

```
StoreID
SalesByQuarter
GCID_SalesByQuarter
```

Aktualisieren Sie die Standardausgabegruppe. Entfernen Sie die folgenden Felder:

```
SalesByQuarter
GCID_SalesByQuarter
```

In der folgenden Abbildung werden die Eingabegruppe und die Ausgabegruppen in der Ansicht **Übersicht** angezeigt:



	Name	Type	Precision	Scale	Location
	Input				
1	StoreID	bigint	19	0	StoreID
2	Store_Name	string	50	0	Store_N...
3	Store_City	string	50	0	Store_City
4	District	decimal	10	0	District
5	Manager	string	50	0	Manager
6	Quarter1	decimal	10	0	SalesBy...
7	Quarter2	decimal	10	0	SalesBy...
8	Quarter3	decimal	10	0	SalesBy...
9	Quarter4	decimal	10	0	SalesBy...
	Output				
1	StoreID	bigint	19	0	StoreID
2	Store_Name	string	50	0	Store_N...
3	Store_City	string	50	0	Store_City
4	District	decimal	10	0	District
5	Manager	string	50	0	Manager
	Output1				
1	StoreID	bigint	19	0	StoreID
2	SalesByQuarter	decimal	10	0	SalesBy...
3	GCID_SalesByQuarter	bigint	19	0	SalesBy...

Die StoreID ist der generierte Schlüssel, der die Filialdaten mit den Umsatzdaten verknüpft. Stellen Sie sicher, dass beide Ausgabegruppen die StoreID zurückgeben.

Beispiel für die Normalizer-Mapping-Ausgabe

Fügen Sie dem Mapping die Schreibdatenobjekte hinzu und verbinden Sie die Normalizer-Umwandlungsausgabe-Ports mit den Datenobjekten.

Wenn Sie das Mapping ausführen, schreibt die Normalizer-Umwandlung die folgenden Zeilen in das Filialziel:

StoreID	Store_Name	Store_City	Kreis	Manager
1	BigStore	New York	East	Robert
2	SmallStore	Phoenix	West	Radhika

Die Normalizer-Umwandlung schreibt die folgenden Ziele in das Umsatzziel:

StoreID	SalesByQuarter	GCID_SalesByQuarter
1	100	1
1	300	2
1	500	3
1	700	4
2	250	1
2	450	2
2	650	3
2	850	4

KAPITEL 31

Parser-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Parser-Umwandlung - Übersicht, 482
- Modi der Parserumwandlung, 483
- Einsatz der Parserumwandlung, 483
- Referenzdatenverwendung in der Parserumwandlung, 484
- Token-Parsing-Operationen, 486
- Token-Parsing-Ports, 487
- Token-Parsing-Eigenschaften, 488
- Musterbasierter Parsing-Modus, 490
- Konfigurieren einer Token-Parsing-Strategie, 491
- Konfigurieren einer Muster-Parsing-Strategie, 492
- Erweiterte Eigenschaften für Parserumwandlung, 493

Parser-Umwandlung - Übersicht

Die Parserumwandlung ist eine passive Umwandlung, die Eingabedatenwerte in neue Ports parst. Die Umwandlung schreibt die Werte entsprechend der in den Werten enthaltenen Informationstypen und der Position der Werte im Eingabestring in neue Ports.

Sie verwenden eine Parserumwandlung, wenn Sie die Struktur eines Datensatzes ändern möchten. Mit der Parserumwandlung können Spalten zu einem Datensatz hinzugefügt und Datenwerte in neue Spalten geschrieben werden. Verwenden Sie eine Parserumwandlung, wenn eine Datenspalte mehrere Werte in einer einzelnen Spalte enthält und die Datenwerte basierend auf den enthaltenen Informationstypen in einzelne Spalten geschrieben werden sollen.

Die Parserumwandlung parst Datenwerte in die von Ihnen festgelegten Ausgabeports. Wenn die Umwandlung einen Eingabedatenwert erkennen kann, ein definierter Ausgabeport jedoch nicht verfügbar ist, schreibt die Umwandlung den Wert auf einen Überlaufport. Wenn die Umwandlung keinen Eingabedatenwert erkennen kann, wird der Wert auf einen Port für nicht geparste Daten geschrieben.

Modi der Parserumwandlung

Beim Erstellen einer Parserumwandlung wählen Sie entweder den Token-Parsing-Modus oder den musterbasierten Parsing-Modus aus.

Wählen Sie einen der folgenden Modi aus:

- Token-Parsing-Modus. Verwenden Sie diesen Modus zum Parsen von Eingabewerten, die Werten in den Referenzdatenobjekten entsprechen, wie Tokensätze, reguläre Ausdrücke, probabilistische Modelle und Referenztabellen. Sie können mehrere Token-Parsing-Strategien in einer Umwandlung verwenden.
- Musterbasierter Parsing-Modus. Verwenden Sie diesen Modus zum Parsen von Eingabewerte, die Werten in Mustersätzen entsprechen.

Einsatz der Parserumwandlung

Verwenden Sie die Parserumwandlung, wenn die Datenfelder in einer Spalte mehr als einen Informationstyp enthalten und die Feldwerte in neue Spalten verschoben werden sollen. Mit der Parserumwandlung können Sie neue Spalten für jeden Informationstyp in einem Datensatz erstellen.

Die folgenden Beispiele beschreiben bestimmte Arten struktureller Änderungen, die Sie mit einer Parserumwandlung durchführen können.

Erstellen neuer Spalten für Kontaktdaten

Sie können eine Datenstruktur erstellen, die Namensdaten aus einer einzelnen Spalte in mehrere Spalten parst. Sie können beispielsweise Spalten für Begrüßungen, Vornamen, zweite Vornamen und Nachnamen erstellen.

Sie konfigurieren die Umwandlung mit einem probabilistischen Modell, das die Strukturen der Personennamen auf dem Eingabeport darstellt. Sie verwenden eine Stichprobe der Eingabeportdaten, um das Modell zu definieren.

Sie erstellen eine Token-Parsing-Strategie, die ein probabilistisches Modell auf den Eingabeport anwendet und die Namenwerte in neue Spalten schreibt. Die Umwandlung schreibt die Namenwerte basierend auf der Position jedes Werts im Eingabestring und dem Namenstyp, den der Wert repräsentiert, in die neuen Spalten.

Hinweis: Außerdem können Sie eine musterbasierte Parsing-Strategie zum Parsen von Kontaktdaten verwenden. Definieren Sie beim Konfigurieren einer musterbasierten Parsing-Strategie die Muster, die die Strukturen der Namen auf dem Eingabeport darstellen.

Erstellen von Adressspalten

Sie können eine Datenstruktur erstellen, die eine einzelne Spalte mit Adressdaten in mehrere Spalten parst, die eine zustellfähige Adresse beschreiben.

Konfigurieren Sie die Umwandlung mit Referenztabellen, die erkennbare Adresselemente enthalten, wie z. B. Postleitzahlen sowie Namen von Bundesstaaten und Städten. Erstellen Sie eine Token-Parsing-Strategie, die jedes Adresselement auf einen neuen Port schreibt.

Sie können eine Referenztabelle nicht verwenden, um Straßenadressdaten aus einem Eingabestring zu parsen, da Straßenname und Nummer zu allgemein sind, um in einer Referenztabelle erfasst zu werden. Sie können jedoch den Überlaufport verwenden, um diese Daten zu erfassen. Wenn Sie alle Stadt-, Bundesstaat- und Postleitzahlendaten aus einer Adresse geparst haben, enthalten die verbleibenden Daten Straßeninformationen.

Verwenden Sie zum Beispiel eine Token-Parsing-Strategie, um die folgende Adresse in Adresselemente aufzuteilen:

```
123 MAIN ST NW STE 12 ANYTOWN NY 12345
```

Die Parsing-Strategie kann die Adresselemente in die folgenden Spalten schreiben:

Spaltenname	Daten
Überlauf	123 MAIN ST NW STE 12
Stadt	Beliebige Stadt
Bundesstaat	NY
PLZ	12345

Erstellen von Produktdatenspalten

Sie können eine Datenstruktur erstellen, die eine einzelne Spalte mit Produktdaten in mehrere Spalten parst, die die Details des Produktbestands beschreiben.

Konfigurieren Sie die Umwandlung mit Tokensätzen, die Bestandselemente enthalten, z. B. Maß, Farbe und Gewicht. Erstellen Sie eine Token-Parsing-Strategie, die jedes Bestandselement auf einen neuen Port schreibt.

Verwenden Sie zum Beispiel eine Token-Parsing-Strategie, um die folgende Farbbeschreibung in getrennte Bestandselemente aufzuteilen:

```
500ML Red Matt Exterior
```

Die Parsing-Strategie kann die Adresselemente in die folgenden Spalten schreiben:

Spaltenname	Daten
Größe	500ML
Farbe	Rot
Stil	Matt
Außenfarbe	J

Referenzdatenverwendung in der Parserumwandlung

Informatica Developer wird mit mehreren Referenzdatenobjekten installiert, die Sie mit der Parserumwandlung verwenden können. Sie können Referenzdatenobjekte auch im Developer Tool erstellen.

Beim Hinzufügen eines Referenzdatenobjekts zu einer Parserumwandlung schreibt die Umwandlung die Strings, die einem Wert im Objekt entsprechen, in neue von Ihnen angegebene Spalten.

In der folgenden Tabelle werden die Referenzdatentypen beschrieben, die verwendet werden können:

Referenzdatentyp	Beschreibung
Mustersätze	Erkennt Datenwerte basierend auf der relativen Position jedes Werts in der Zeichenfolge.
Probabilistische Modelle	Fügt Token-Parsing-Operationen Fuzzy-Match-Funktionen hinzu. Die Umwandlung kann ein probabilistisches Modell verwenden, um den Informationstyp in einer Zeichenfolge abzuleiten. Zum Aktivieren der Fuzzy-Match-Funktionen kompilieren Sie das probabilistische Modell im Developer-Tool.
Referenztabellen	Sucht nach Zeichenfolgen, die mit den Einträgen in einer Datenbanktabelle übereinstimmen.
Reguläre Ausdrücke	Erkennt Zeichenfolgen, die mit den von Ihnen definierten Bedingungen übereinstimmen. Sie können einen regulären Ausdruck verwenden, um nach einer Zeichenfolge innerhalb einer längeren Zeichenfolge zu suchen.
Tokensätze	Erkennt Zeichenfolgen basierend auf den enthaltenen Informationstypen. Informatica installiert mit Tokensätzen verschiedene Arten von Tokendefinitionen, wie z. B. Wörter, Telefonnummern, Postleitzahlen und Produktcodedefinitionen.

Mustersätze

Ein Mustersatz enthält Expressions, die Datensätze in der Ausgabe einer Token-Bezeichnung kennzeichnen. Sie können Mustersätze verwenden, um den Datenausgabeport zu analysieren, der mit einem Token versehen ist, und um Matching-Strings in einen oder mehrere Ausgabeports zu schreiben. Verwenden Sie Mustersätze in Parser-Umwandlungen, die den Muster-Parsing-Modus verwenden.

Sie können beispielsweise eine Parser-Umwandlung konfigurieren, um Mustersätze zu verwenden, die Namen und Anfangsbuchstaben angeben. Diese Umwandlung verwendet die Mustersätze, um die Ausgabe einer Labeler-Umwandlung im Tokenbeschriftungsmodus zu analysieren. Sie können die Parser-Umwandlung konfigurieren, um Namen und Anfangsbuchstaben in der Ausgabe an verschiedene Ports zu schreiben.

Probabilistische Modelle

Ein probabilistisches Modell erkennt Token anhand des Typs der enthaltenen Informationen sowie deren Positionen in einer Eingabezeichenfolge.

Ein probabilistisches Modell enthält Referenzdatenwerte und Beschriftungswerte. Die Referenzdatenwerte stellen die Daten auf einem Eingabeport dar, den Sie mit der Umwandlung verbinden. Die Beschriftungswerte beschreiben die Typen von Informationen, die die Referenzdatenwerte enthalten. Sie weisen jedem Referenzdatenwert im Modell eine Beschriftung zu.

Sie kompilieren das Modell, um die Referenzdatenzeilen mit den Beschriftungen in einem probabilistischen Modell zu verknüpfen. Der Kompilierungprozess generiert eine Reihe von logischen Assoziationen zwischen den Datenwerten und den Beschriftungen. Wenn Sie ein Mapping ausführen, das das Modell liest, wendet der Datenintegrationsdienst die Logik des Modells auf die Eingabedaten der Umwandlung an. Der Datenintegrationsdienst gibt die Beschriftung zurück, die die Eingabedatenwerte am genauesten beschreibt.

Sie erstellen ein probabilistisches Modell im Developer Tool. Das Modellrepository speichert das probabilistische Modellobjekt. Das Developer-Tool schreibt die Datenwerte, die Beschriftungen und die Kompilierungsdaten in eine Datei in der Informatica-Verzeichnisstruktur.

Hinweis: Wenn Sie ein probabilistisches Modell zu einer Token-Parsing-Operation hinzufügen und die Beschriftungskonfiguration anschließend im probabilistischen Modell ändern, wird die Operation ungültig.

Wenn Sie die Beschriftungskonfiguration in einem probabilistischen Modell aktualisieren, erstellen Sie jede Parsing-Operation, die das Modell verwendet, neu.

Referenztabellen

Eine Referenztabelle ist eine Datenbanktabelle mit mindestens zwei Spalten. Eine Spalte enthält die Standard- oder die erforderliche Version eines Datenwerts. Andere Spalten enthalten alternative Versionen des Werts. Beim Hinzufügen einer Referenztabelle zu einer Umwandlung durchsucht die Umwandlung die Eingabeportdaten auf Werte, die auch in der Tabelle angezeigt werden. Sie können Tabellen mit beliebigen nützlichen Daten für das Datenprojekt erstellen, an dem Sie arbeiten.

Reguläre Ausdrücke

Im Kontext einer Parsing-Operation ist eine reguläre Expression eine Expression, die Sie verwenden können, um eine oder mehrere Strings in Eingabedaten zu kennzeichnen. Die Parser-Umwandlung schreibt gekennzeichnete Strings in einen oder mehrere Ausgabeports. Sie können reguläre Ausdrücke in Parser-Umwandlungen im Token-Parsing-Modus verwenden.

Parser-Umwandlungen verwenden reguläre Ausdrücke, um Muster in Eingabedaten abzugleichen und alle Matching-Strings in einen oder mehrere Ausgabeports zu parsen. Beispielsweise können Sie mit einem regulären Ausdruck alle E-Mail-Adressen in Eingabedaten kennzeichnen und jede E-Mail-Adresskomponente in einen anderen Ausgang parsen.

Tokensätze

Ein Tokensatz enthält Ausdrücke, die bestimmte Token kennzeichnen. Sie können Tokensätze in Parser-Umwandlungen verwenden, die den Modus "Token-Parsing" verwenden.

Verwenden Sie Tokensätze, um bestimmte Token als Teil der Parsing-Operationen zu kennzeichnen. Sie können einen Tokensatz beispielsweise zum Parsen aller E-Mail Adressen verwenden, die das Format "Kontoname@Domänenname" aufweisen.

Token-Parsing-Operationen

Im Token-Parsing-Modus parst die Parserumwandlung Strings, die mit Daten in Tokensätzen, regulären Ausdrücken und probabilistischen Modellen oder mit Referenztabelleneinträgen übereinstimmen.

Zum Durchführen des Token-Parsings fügen Sie in der Ansicht **Strategien** der Umwandlung Strategien hinzu. Sie können jeder Strategie eine oder mehrere Operationen hinzufügen. Die Umwandlung bietet einen Assistenten, der zum Erstellen von Strategien verwendet wird.

Sie können einer Token-Parsing-Strategie die folgenden Operationstypen hinzufügen:

Parsen mittels Tokensatz

Verwenden Sie vordefinierte oder benutzerdefinierte Tokendefinitionen zum Parsen von Eingabedaten. Tokensatzoperationen können benutzerdefinierte reguläre Ausdrücke verwenden, die Daten in eine oder mehrere Ausgaben schreiben.

Sie können auch probabilistische Modelle zum Identifizieren und Parsen von Eingabedaten verwenden.

Parsen mittels Referenztabelle

Verwenden Sie Referenztabellen zum Parsen von Eingabedaten.

Die Umwandlung führt die Operationen in der Reihenfolge durch, in der sie in der Strategie angezeigt werden.

Token-Parsing-Ports

Konfigurieren Sie die Token-Parsing-Ports mit für Ihre Daten geeigneten Einstellungen.

Eine Parser-Umwandlung im Token-Parsing-Modus weist die folgenden Porttypen auf:

Eingabe

Enthält Daten, die Sie an die Parser-Umwandlung übergeben. Die Umwandlung führt mithilfe des auf der Registerkarte **Strategien** festgelegten **Join-Zeichens für die Eingabe** alle Eingabeports in einem kombinierten Datenstring zusammen. Wenn Sie kein Join-Zeichen für die Eingabe festlegen, verwendet die Umwandlung standardmäßig ein Leerzeichen.

Geparste Ausgabeports

Benutzerdefinierte Ausgabeports, die erfolgreich geparste Strings enthalten. In Fällen, in denen mehrere Parsing-Strategien dieselbe Ausgabe verwenden, führt die Umwandlung mithilfe des auf der Registerkarte **Strategien** festgelegten **Join-Zeichens für die Ausgabe** die Ausgabe in einem kombinierten Datenstring zusammen. Wenn Sie kein Join-Zeichen für die Ausgabe festlegen, verwendet die Umwandlung standardmäßig ein Leerzeichen.

Overflow

Enthält erfolgreich geparste Strings, die nicht in die Anzahl der in der Umwandlung festgelegten Ausgaben passen. Wenn die Umwandlung beispielsweise nur zwei "Wort"-Ausgaben aufweist, resultiert der String "John James Smith" in der Überlaufausgabe "Smith." Die Parser-Umwandlung erstellt einen Überlaufport für jede hinzugefügte Strategie.

Wenn Sie die Option für den detaillierten Überlauf auswählen, erstellt die Umwandlung einen Überlaufport für jede Beschriftung im Modell.

Nicht geparst

Enthält Strings, die von der Umwandlung nicht erfolgreich geparst werden können. Die Parser-Umwandlung erstellt einen nicht geparsten Port für jede hinzugefügte Strategie.

Ausgabeports im probabilistischen Matching

Wenn Sie eine Parsing-Strategie zur Verwendung probabilistischer Matching-Methoden konfigurieren, fügt die Parserumwandlung einen Port hinzu, um die Übereinstimmungswerte für jeden Ausgabeport zu speichern.

In der folgenden Tabelle werden die Porttypen beschrieben:

Porttyp	Beim probabilistischen Matching erstellter Port
Geparster Ausgabeport	[Beschriftungsname] Ausgabe [Beschriftungsname] Score-Ausgabe
Port für Überlaufdaten	[Überlaufdaten] Ausgabe [[Überlaufdaten] Score-Ausgabe
Nicht geparster Datenport	[Nicht geparste Daten] Ausgabe [Nicht geparste Daten] Score-Ausgabe

Token-Parsing-Eigenschaften

Konfigurieren Sie Eigenschaften für Token-Parsing-Operationen in der Ansicht **Strategien** in der Parserumwandlung.

Allgemeine Eigenschaften

Allgemeine Eigenschaften gelten für alle Token-Parsing-Operationen, die in der Strategie definiert werden. Verwenden Sie die allgemeinen Eigenschaften, um die Strategie zu benennen, Eingabe- und Ausgabeports festzulegen und anzugeben, ob die Strategie probabilistische Matching-Methoden ermöglicht.

In der folgenden Tabelle werden die allgemeinen Eigenschaften beschrieben:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Strategie bereit.
Eingaben	Gibt die Eingabeports an, die von den Strategieoperationen gelesen werden können.
Ausgaben	Gibt die Ausgabeports an, in die Strategieoperationen schreiben können.
Beschreibung	Beschreibt die Strategie. Die Eigenschaft ist optional.
Probabilistische Matching-Methoden verwenden	Gibt an, dass die Strategie ein probabilistisches Modell zur Angabe von Token verwenden kann.
Join-Zeichen für Eingabe	Gibt das Zeichen an, das zum Zusammenführen von Eingabedatenports verwendet wird. Die Umwandlung führt alle Eingabeports in einem kombinierten Datenstring zusammen und parst den gesamten String.
Join-Zeichen für Ausgabe	Gibt das Zeichen an, dass zum Zusammenführen von Ausgabedaten verwendet wird, wenn mehrere Parsing-Operationen dieselbe Ausgabe verwenden.
Umkehr aktiviert	Konfiguriert die Strategie zum Parsen von Daten von rechts nach links. Diese Eigenschaft ist für probabilistisches Matching deaktiviert.
Overflow-Umkehr aktiviert	Konfiguriert die Strategie zum Parsen von Overflow-Daten von rechts nach links. Diese Eigenschaft ist für probabilistisches Matching deaktiviert.
Detaillierter Overflow aktiviert	Erstellt ein eindeutiges Overflow-Feld für jede Parsing-Operation.
Delimiter	Gibt die Delimiter an, die die Eingabedaten in separate Token trennen. Standardwert ist Leerzeichen.

Eigenschaften des probabilistischen Modells

Sie können ein probabilistisches Modell anstelle eines Tokensatzes auswählen, wenn Sie eine Token-Parsing-Strategie konfigurieren. Wählen Sie die Operation **Parsen mittels Tokensatz** sowie die Option zur Verwendung probabilistischer Matching-Methoden aus.

Die folgende Tabelle beschreibt Eigenschaften des probabilistischen Modells:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Text filtern	Verwendet Zeichen oder Platzhalter, die von Ihnen zum Filtern der Liste mit Tokensätzen, probabilistischen Modellen oder regulären Ausdrücken eingegeben wurden.
Probabilistisches Modell	Gibt das ausgewählte probabilistische Modell an.

Referenztabelleneigenschaften

Referenztabelleneigenschaften werden angewendet, wenn Sie eine Beschriftungsoperation zur Verwendung einer Referenztabelle konfigurieren.

Die folgende Tabelle beschreibt die Referenztabelleneigenschaften:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Referenztabelle	Gibt die Referenztabellen an, die von der Operation zum Parsen von Eingabewerten verwendet werden.
Groß-/Kleinschreibung	Gibt an, ob Eingabestrings mit der Groß-/Kleinschreibung der Referenztabelleneinträge übereinstimmen müssen.
Matches durch gültige Werte ersetzen	Ersetzt geparste Daten durch Daten aus der Spalte "Gültig" in der Referenztabelle.
Ausgaben	Gibt die Ausgabeports für die geparsten Daten an.

Eigenschaften von Tokensätzen

Tokensatzeigenschaften werden angewendet, wenn Sie eine Parsing-Operation für die Verwendung von Tokensätzen konfigurieren.

Wählen Sie die Operation **Parsen mittels Tokensatz** aus, um die Eingabe mit Tokensätzen zu parsen. Deaktivieren Sie die Option zum Verwenden probabilistischer Matching-Methoden.

Die folgenden Tabellen beschreiben die Tokensatzeigenschaften:

Eigenschaft	Beschreibung
Name	Stellt einen Namen für die Operation bereit.
Tokensätze (Nur Einzel-Ausgabe)	Gibt den Tokensatz an, den der Vorgang zum Parsen von Daten verwendet. Der Vorgang schreibt die Daten in einen einzelnen Port.
Regulärer Ausdruck (Einzel- oder Mehrfach-Ausgabe)	Gibt den regulären Ausdruck an, den der Vorgang zum Parsen von Daten verwendet. Der Vorgang schreibt Daten in mehrere Ports, wenn mehrere Zeichenfolgen im Eingabefeld gefunden werden.
Ausgaben	Gibt die Ausgabeports an, auf die die Vorgänge schreiben.

Sie können einen Tokensatz oder regulären Ausdruck hinzufügen, bearbeiten, importieren oder entfernen. Außerdem können Sie die Liste der Tokensätze filtern.

In der folgenden Tabelle werden die Eigenschaften beschrieben, die Sie zur Durchführung dieser Aufgaben verwenden:

Eigenschaft	Beschreibung
Text filtern	Filtert die Liste von Tokensätzen oder regulären Ausdrücken. Verwenden Sie Textzeichen und Platzhalterzeichen als Filter.
Hinzufügen	Verwenden Sie diese Option zum Definieren eines benutzerdefinierten Tokensatzes oder regulären Ausdrucks.
Bearbeiten	Bearbeitet den Inhalt eines benutzerdefinierten Tokensatzes.
Importieren	Importiert eine nicht wiederverwendbare Kopie eines Tokensatzes oder regulären Ausdrucks aus einem Ordner im Modellrepository. Wenn Sie das Quellobjekt für den Tokensatz oder regulären Ausdruck aktualisieren, aktualisiert der Datenintegrationsdienst nicht die nicht wiederverwendbare Kopie.
Entfernen	Löscht einen benutzerdefinierten Tokensatz oder regulären Ausdruck.

Musterbasierter Parsing-Modus

Im musterbasierten Parsing-Modus parst die Parser-Umwandlung Muster, die aus mehreren Strings bestehen.

Sie können die folgenden Methoden verwenden, um Muster im musterbasierten Parsing-Modus zu definieren:

- Parsen von Eingabedaten mit Mustern, die in Referenztabellen definiert sind. Sie können eine Musterreferenztabelle aus der profilierten Ausgabe einer Labeler-Umwandlung erstellen, die den Tokenbeschriftungsmodus verwendet.
- Parsen von Eingabedaten mit Mustern, die Sie definieren.
- Parsen Sie Eingabedateien unter Verwendung von Mustern, die Sie aus einem wiederverwendbaren Mustersatz im Model Repository importiert haben. Änderungen am wiederverwendbaren Mustersatz aktualisieren nicht die Daten, die Sie in der Parserumwandlung hinzugefügt haben.

Mit den Platzhaltern "+" und "*" können Sie ein Muster definieren. Verwenden Sie "*"-Zeichen, um einen beliebigen String abzugleichen, und "+"-Zeichen, um eine oder mehrere Instanzen des vorherigen Strings abzugleichen. Verwenden Sie zum Beispiel "WORD+", um mehrere aufeinander folgende Instanzen eines Wort-Tokens zu suchen, und verwenden Sie "WORD *", um einen Wort-Token gefolgt von einem oder mehreren Token eines beliebigen Typs zu suchen.

Sie können mehrere Instanzen dieser Methoden innerhalb der Parser-Umwandlung verwenden. Die Umwandlung verwendet die Instanzen in der Reihenfolge, in der sie in der Ansicht **Konfiguration** aufgelistet sind.

Hinweis: Im musterbasierten Parsing-Modus benötigt die Parser-Umwandlung die Ausgabe aus einer Labeler-Umwandlung, die den Tokenbeschriftungsmodus verwendet. Erstellen und konfigurieren Sie die Labeler-Umwandlung vor dem Erstellen einer Parserumwandlung, die den musterbasierten Parsing-Modus verwendet.

Musterbasierte-Parsing-Ports

Konfigurieren Sie die die musterbasierte Parsing-Ports mit Einstellungen, die für Ihre Daten geeignet sind.

Eine Parser-Umwandlung, die den musterbasierten Parsing-Modus verwendet, weist die folgenden Porttypen auf:

Label_Data

Verbinden Sie diesen Port mit dem `Labeled_Output`-Port einer Labeler-Umwandlung, die den Tokenbeschriftungsmodus verwendet.

Tokenized_Data

Verbinden Sie diesen Port mit dem `Tokenized_Data`-Ausgabeport einer Labeler-Umwandlung, die den Tokenbeschriftungsmodus verwendet.

Parse_Status

Wenn ein Match für die Eingabemuster gefunden wird, gibt dieser Port den Wert `Matched` aus. Wenn kein Match gefunden, wird `Unmatched` ausgegeben.

Overflow

Erfolgreich geparste Strings, die nicht in die Anzahl der in der Umwandlung definierten Ausgaben passen. Wenn beispielsweise nur zwei "WORD"-Ausgaben definiert sind, resultiert der String "John Smith" standardmäßig in der Overflow-Ausgabe "Smith".

Parsed

Erfolgreich geparste Strings in benutzerdefinierten Ports.

Konfigurieren einer Token-Parsing-Strategie

Öffnen Sie zum Konfigurieren einer Token-Parsing-Strategie eine Parserumwandlung im Token-Parsing-Modus und wählen Sie die Ansicht **Strategien** aus.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Klicken Sie auf **Neu**.

   Der Assistent **Neue Strategie** wird geöffnet.
3. Klicken Sie auf das Feld **Eingaben**, um Ports für die Strategie auszuwählen.
4. Konfigurieren Sie die Strategieeigenschaften und klicken Sie auf **Weiter**.

5. Wählen Sie eine Operation aus und klicken Sie auf **Weiter**.
6. Konfigurieren Sie die Operationseigenschaften und wählen Sie Ausgabeports für erfolgreich geparste Daten aus.
7. Optional können Sie auf **Weiter** klicken, um der Strategie mehrere Vorgänge hinzuzufügen.
8. Nachdem Sie der Strategie alle Operationen hinzugefügt haben, klicken Sie auf **Fertig stellen**.
9. Optional können Sie mehrere Strategien zur Umwandlung hinzufügen.
10. Ändern Sie wahlweise die Reihenfolge, in der die Umwandlung Strategien und Operationen verarbeitet. Wählen Sie eine Strategie oder Operation aus und klicken Sie auf **Nach oben verschieben** oder **Nach unten verschieben**.

Konfigurieren einer Muster-Parsing-Strategie

Öffnen Sie zum Konfigurieren einer Muster-Parsing-Strategie eine Parserumwandlung im Muster-Parsing-Modus und wählen Sie die Ansicht **Muster** aus.

Bevor Sie die Umwandlung zum Parsen von Mustern konfigurieren, stellen Sie sicher, dass in der Ansicht **Muster** die Namen der erwarteten Ausgabeports angezeigt werden. Bei der Parserumwandlung werden Token für ausgewählte Ausgabeports geparst. Erstellen Sie bei Bedarf zusätzliche Ausgabeports.

1. Wählen Sie die Ansicht **Muster** aus.
2. Fügen Sie der Strategie ein oder mehrere Muster hinzu. Zum Hinzufügen von Mustern stehen folgende Möglichkeiten bereit:
   - Eingeben von Datenwerten zum Erstellen eines Musters. Klicken Sie auf **Neu** und wählen Sie **Neues Muster** aus.

     Wenn Sie **Neues Muster,** auswählen, klicken Sie auf **Muster hier eingeben** und geben Sie einen oder mehrere Tokentypen ein. Die eingegebenen Token müssen der Tokenstruktur eines Eingabedatenfelds entsprechen. Fügen Sie Muster hinzu, die zum Beschreiben der Tokenstrukturen auf dem Eingabeport benötigt werden.
   - Importieren von Datenwerten aus einer Referenztabelle. Klicken Sie auf **Neu** und wählen Sie **Neue Referenztabelle** aus.

     Wenn Sie **Neue Referenztabelle** auswählen, durchsuchen Sie das Repository auf eine Referenztabelle und wählen Sie eine Spalte aus, die eine Liste der Tokenstrukturen enthält.
   - Importieren von Datenwerten aus einem Mustersatz. Klicken Sie auf **Importieren** und wählen Sie einen wiederverwendbaren Mustersatz im Modellrepository aus.

     Wenn Sie **Importieren** auswählen, durchsuchen Sie die im Repository definierten Inhaltssätze und wählen Sie einen wiederverwendbaren Mustersatz aus.

   Hinweis: Sie können das Feld **Filtertext** verwenden, um die Listen mit Referenztabellen und Mustersätzen zu filtern.

   Sie können Mustersätze und Referenztabellen in der Spalte "Muster" kombinieren.
3. Ordnen Sie jeden Token in der Spalte "Muster" einem Ausgabeport zu.
   - Zum Zuweisen eines Token zu einem Ausgabeport doppelklicken Sie auf die Spalte "Port" und wählen Sie den Tokennamen aus dem Menü aus.
   - Zum Parsen mehrerer Token in eine einzige Ausgabe doppelklicken Sie auf die Spalte "Port" und wählen Sie **Benutzerdefiniert** aus. Weisen Sie dem Port Token zu und wählen Sie den zu verwendenden Delimiter aus.

Weisen Sie die Token in jeder Spalte des Musters zu einem oder mehreren Ausgabeports zu.

4. Speichern Sie die Umwandlung.

Erweiterte Eigenschaften für Parserumwandlung

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Parserumwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 32

Rang-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Rang-Umwandlung - Übersicht, 494
- Rang-Umwandlungen in dynamischen Mappings, 495
- Rang-Umwandlungsports, 496
- Rang-Port, 497
- Ports „Gruppieren nach" definieren, 497
- Rang-Caches, 499
- Rangumwandlung - Erweiterte Eigenschaften, 499

Rang-Umwandlung - Übersicht

Die Rang-Umwandlung ist eine aktive Umwandlung, die Datensätze auf einen oberen oder unteren Bereich beschränkt. Verwenden Sie eine Rang-Umwandlung, um den größten oder kleinsten numerischen Wert in einem Port oder einer Gruppe zurückzugeben. Sie können eine Rang-Umwandlung auch verwenden, um Strings im oberen oder unteren Bereich einer Mapping-Sortierreihenfolge zurückzugeben.

Bei der Ausführung eines Mappings speichert der Data Integration Service Eingabedaten so lange zwischen, bis er die Rang-Berechnungen durchführen kann.

Die Rang-Umwandlung unterscheidet sich von den Umwandlungsfunktionen MAX und MIN. Die Rang-Umwandlung gibt eine Gruppe von Werten aus den oberen und unteren Bereichen zurück, nicht nur einen Wert. Verwenden Sie zum Beispiel eine Rang-Umwandlung, um die Top 10 unter den Vertriebsmitarbeitern eines bestimmten Gebiets auszuwählen. Wenn Sie alternativ dazu einen Finanzbericht generieren möchten, verwenden Sie eine Rang-Umwandlung, um die drei Abteilungen mit den niedrigsten Kosten in Bezug auf Löhne und Overhead zu ermitteln. Während die SQL-Sprache viele Funktionen zur Verarbeitung von Gruppen von Daten bietet, können mit Standard-SQL-Funktionen keine oberen oder unteren Bereiche in einer Gruppe von Zeilen ermittelt werden.

Sie verbinden alle Ports, die denselben Zeilensatz für die Umwandlung darstellen. Die Zeilen, die zu diesem auf einigen Maßeinheiten basierten Rang gehören, die Sie beim Konfigurieren der Umwandlung festgelegt haben, durchlaufen die Rang-Umwandlung.

Als aktive Umwandlung kann die Rang-Umwandlung die Anzahl der sie durchlaufenden Zeilen ändern. Sie können 100 Zeilen an die Rang-Umwandlung übergeben, aber nur das Ranking der Top 10 der Zeilen auswählen. Die Top 10 der Zeilen werden aus der Rang-Umwandlung an eine andere Umwandlung übergeben.

Sie können Ports aus einer Umwandlung mit der Rang-Umwandlung verbinden. Sie können auch lokale Variablen erstellen und Nicht-Aggregat-Ausdrücke schreiben.

Ranking von Stringwerten

Sie können die Rang-Umwandlung so konfigurieren, dass die oberen oder unteren Werte eines Stringports zurückgegeben werden. Der Data Integration Service sortiert Strings basierend auf der für das bereitgestellte Mapping ausgewählten Sortierreihenfolge.

Beim Konfigurieren der Anwendung, die das Mapping enthält, wählen Sie die Sortierreihenfolge, die der Data Integration Service zum Ausführen des Mappings verwendet. Sie können einen binären Wert oder eine bestimmte Sprache wie Französisch oder Deutsch auswählen. Wenn Sie einen binären Wert auswählen, berechnet der Data Integration Service den binären Wert jedes Strings und sortiert die Strings mit den binären Werten. Wenn Sie eine Sprache auswählen, sortiert der Data Integration Service die Strings alphabetisch mit der Sortierreihenfolge für die Sprache.

Rang-Umwandlungseigenschaften

Beim Erstellen einer Rang-Umwandlung können Sie die folgenden Eigenschaften konfigurieren:

- Geben Sie ein Cache-Verzeichnis ein.
- Wählen Sie den oberen oder unteren Rang aus.
- Wählen Sie den Eingabe-/Ausgabeport aus, der die zum Bestimmen eines Rangs verwendeten Werte enthält. Sie können nur einen Port zum Definieren eines Rangs auswählen.
- Wählen Sie die Anzahl der Zeilen für das Ranking aus.
- Definieren Sie Gruppen für Ränge, zum Beispiel die 10 billigsten Produkte für jeden Hersteller.

Rang-Umwandlungen in dynamischen Mappings

Sie können eine Rang-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports verweisen.

Wenn Sie in der Rang-Umwandlung auf einen generierten Port verweisen und der generierte Port zur Laufzeit nicht vorhanden ist, schlägt das Mapping fehl.

Wenn Sie einen dynamischen Port als Rang-Port festlegen, darf der dynamische Port höchstens einen generierten Port aufweisen.

Wenn Sie einen dynamischen Port als den Port „Gruppieren nach“ festlegen, betrachtet der Datenintegrationsdienst alle generierten Ports als Ports „Gruppieren nach“. Das Mapping ist nicht gültig, wenn Sie einen generierten Port als einen Port „Gruppieren nach“ und den übergeordneten dynamischen Port als Rang-Port oder als Port „Gruppieren nach“ festlegen.

Sie können den Rang-Port und die Ports „Gruppieren nach“ parametrisieren. Verwenden Sie einen Parameter des Typs „Port“ für den Rang-Port. Verwenden Sie einen Parameter des Typs „Portliste“ für die Ports „Gruppieren nach“.

Rang-Umwandlungsports

Die Rang-Umwandlung enthält Eingabe-, Eingabe-/Ausgabe- oder Ausgabeports, die mit einer anderen Umwandlung im Mapping verbunden sind. Die Umwandlung enthält zudem Pass-Through-Ports und variable Ports.

Eine Rang-Umwandlung weist die folgenden Porttypen auf:

Eingabe

Empfängt Daten aus vorangestellten Umwandlungen. Sie können Eingabeports als Eingabe-/Ausgabeports angeben. Die Umwandlung muss mindestens einen Eingabeport haben.

Dynamischer Port

Ein Port kann mehrere Spalten empfangen, um eine dynamische Anzahl von generierten Ports zu erstellen. Ein generierter Port ist ein Port innerhalb eines dynamischen Ports, der eine einzelne Spalte darstellt. Sie können dynamische Eingabe-, Ausgabe- und variable Ports erstellen.

Ausgabe

Übergibt Daten an nachgelagerte Umwandlungen. Sie können Ausgabeports als Eingabe-/Ausgabeports angeben. Die Umwandlung muss mindestens einen Ausgabeport haben.

Pass-Through

Übergibt Daten in unverändertem Zustand.

Variable

Wird für lokale Variablen verwendet. Sie können einen variablen Port zum Speichern von Werten oder Berechnungen zur Verwendung in einem Ausdruck verwenden. Variable Ports können keine Eingabe- oder Ausgabeports sein. Übergeben Sie Daten in der Umwandlung.

Rang-Index

Das Developer Tool erstellt einen RANKINDEX-Port für jede Rang-Umwandlung. Der Data Integration Service verwendet den Rang-Index-Port zum Speichern der Ranking-Position für jede Zeile in einer Gruppe.

Sie können beispielsweise eine Rang-Umwandlung erstellen, um die 50 bestbezahltesten Mitarbeiter im Unternehmen zu ermitteln. Sie geben die SALARY-Spalte als Eingabe-/Ausgabeport zum Messen der Ränge an und konfigurieren die Umwandlung, um bis auf die Top 50-Zeilen alle Zeilen herauszufiltern.

Nachdem alle Zeilen, die zu einem oberen oder unteren Rang gehören, von der Rang-Umwandlung ermittelt wurden, werden die Rang-Index-Werte zugewiesen. Im Falle der nach Gehalt eingestuften Top 50-Mitarbeiter wird dem bestbezahltesten Mitarbeiter ein Rang-Index von 1 zugeordnet. Dem nächsten bestbezahltesten Mitarbeiter wird ein Rang-Index von 2 zugeordnet usw. Beim Ranking eines niedrigen Bereichs, wie zum Beispiel die 10 billigsten Produkte im Bestand, weist die Rang-Umwandlung einen Rang-Index von der niedrigsten bis zur höchsten Stufe zu. Der billigste Artikel erhält somit einen Rang-Index von 1.

Wenn zwei Rang-Werte übereinstimmen, erhalten Sie denselben Wert im Rang-Index und die Umwandlung überspringt den nächsten Wert. Wenn Sie zum Beispiel die Top 5 unter den Einzelhandelsgeschäften im Land und zwei Filialen mit demselben Umsatz anzeigen möchten, werden möglicherweise die folgenden Daten zurückgegeben:

RANKINDEX	SALES	STORE
1	10000	Orange
1	10000	Brea
3	90000	Los Angeles

RANKINDEX	SALES	STORE
4	80000	Ventura

Die RANKINDEX ist nur ein Ausgabeport. Sie können den Rang-Index an eine andere Umwandlung im Mapping oder direkt an ein Target übergeben.

Rang-Port

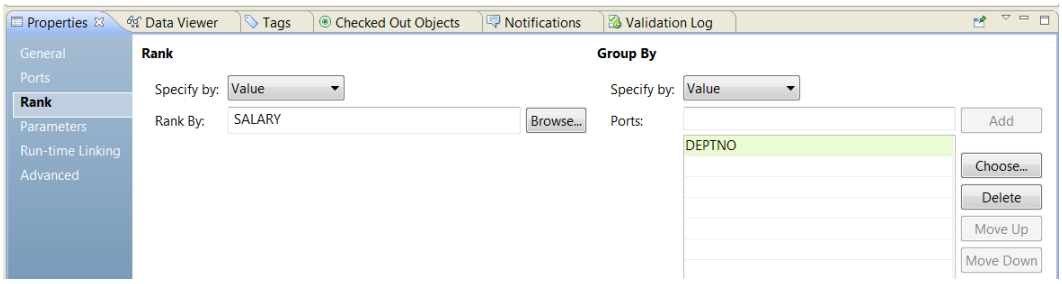
Der Rang-Port bestimmt die Spalte, nach der die Werte angeordnet werden.

Sie müssen einen Eingabe-/Ausgabeport oder Ausgabeports als Rangport angeben. Angenommen, Sie erzeugen eine Rang-Umwandlung, um die besten Mitarbeiter in jeder Abteilung basierend auf dem Gehalt anzuordnen. Der Port „Salary“ enthält das Gehalt für jeden Mitarbeiter. Sie legen den Eingabe-/Ausgabeport „Salary“ als Rang-Port fest.

Wählen Sie den Rang-Port auf der Registerkarte **Rang** der Ansicht **Eigenschaften** aus. Sie können einen Parameter für den Rang-Port verwenden. Um einen Parameter zu verwenden, wählen Sie **Angeben nach Parameter** aus. Suchen Sie nach einem Portparameter oder erstellen Sie einen Portparameter. Der Standardwert des Parameters ist der Name eines Ports bzw. generierten Ports.

Sie müssen den Rang-Port mit einer anderen Umwandlung verknüpfen.

Die folgende Abbildung zeigt die Registerkarte **Rang**:



Ports „Gruppieren nach” definieren

Sie können die Rang-Umwandlung konfigurieren, um Gruppen für Zeilen zu erstellen, denen ein Rang zugewiesen wurde.

Wenn Sie beispielsweise die 10 teuersten Artikel nach Hersteller auswählen, definieren Sie zunächst eine Gruppe für jeden Hersteller. Im Bereich **Gruppieren nach** der Registerkarte **Rang** können Sie einen der Eingabe-, Eingabe-/Ausgabe- oder Ausgabeports als Port „Gruppieren nach” festlegen.

Für jeden eindeutigen Wert im Gruppenport erstellt die Umwandlung eine Gruppe von Zeilen, die in der Rang-Definition enthalten ist (oben oder unten und eine bestimmte Anzahl in jedem Rang).

Die Rang-Umwandlung ändert die Anzahl der Zeilen auf zwei verschiedene Arten. Wenn Sie mit Ausnahme der Zeilen, die in einen oberen oder unteren Rang fallen, alle Zeilen filtern, reduzieren Sie die Anzahl der Zeilen,

die die Umwandlung durchlaufen. Wenn Sie Gruppen definieren, erstellen Sie Zeilen für jede Gruppe, die mit einem Rang versehenen.

Wenn Sie beispielsweise eine Rang-Umwandlung erstellen, die den nach Quartal gruppierten Top 5 unter den Vertriebsmitarbeitern einen Rang zuordnet, nummeriert der Rang-Index die Vertriebsmitarbeiter von 1 bis 5 für jedes Quartal.

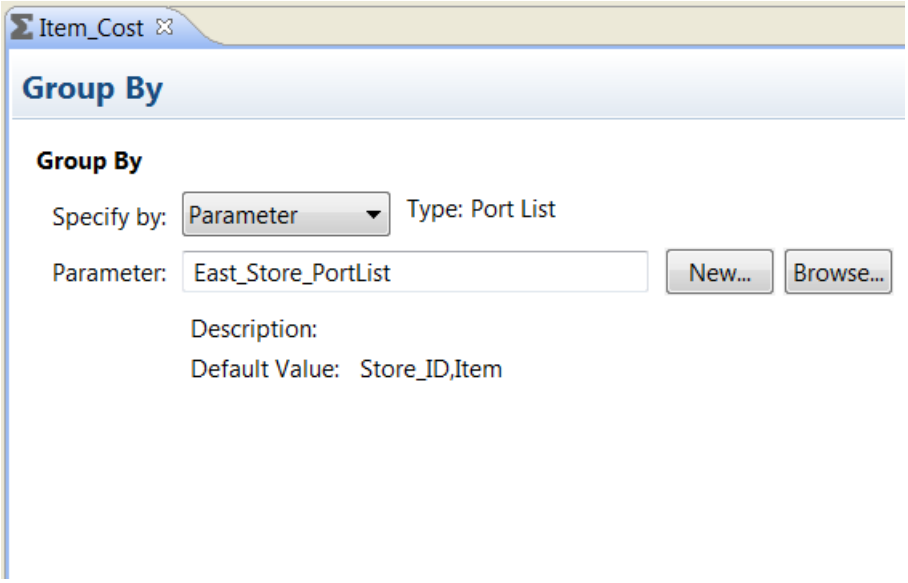
RANKINDEX	SALES_PERSON	SALES	QUARTER
1	Sam	10,000	1
2	Mary	9,000	1
3	Alice	8,000	1
4	Ron	7,000	1
5	Alex	6,000	1

Legen Sie auf der Registerkarte **Erweitert** in der Ansicht **Eigenschaften** die Anzahl der Zeilen fest, die in ein Ranking eingeschlossen werden soll.

Parameter „Gruppieren nach"

Sie können einen Portlistenparameter konfigurieren, der mindestens einen in die Gruppe einzuschließenden Port enthält. Erstellen Sie einen Portlistenparameter, indem Sie Ports aus einer Liste mit den Ports in der Umwandlung auswählen.

Die folgende Abbildung zeigt die Registerkarte **Gruppieren nach**, wenn Sie einen Parameter zur Angabe der Ports in der Gruppe verwenden:



Sie können nach einem Portlistenparameter suchen oder auf **Neu** klicken, um einen Portlistenparameter zu erstellen. Wenn Sie einen Portlistenparameter erstellen möchten, können Sie die Ports aus einer Liste mit den Ports in der Umwandlung auswählen.

Rang-Caches

Wenn Sie ein Mapping ausführen, das eine Rangumwandlung verwendet, erstellt der Datenintegrationsdienst einen Index- und Daten-Cache im Speicher, um die Umwandlung zu verarbeiten. Wenn der Datenintegrationsdienst mehr Speicherplatz benötigt als im Speicher-Cache vorhanden ist, werden Überlaufdaten in Cachedateien gespeichert.

Wenn Sie ein Mapping ausführen, das eine Rangumwandlung verwendet, vergleicht der Datenintegrationsdienst eine Eingabezeile mit den Zeilen im Daten-Cache. Wenn die Eingabezeile über den Rang einer zwischengespeicherten Zeile hinausgeht, ersetzt der Datenintegrationsdienst die zwischengespeicherte Zeile mit der Eingabezeile. Wenn Sie die Rang-Umwandlung in Gruppenzeilen konfigurieren, weist der Datenintegrationsdienst den Zeilen innerhalb jeder Gruppe einen Rang zu.

Der Datenintegrationsdienst erstellt die folgenden Caches für die Rangumwandlung:

- Index-Cache. Speichert die Gruppenwerte so, wie sie in „Nach Ports gruppieren” konfiguriert wurden.
- Daten-Cache. Speichert Informationen basierend auf „Nach Ports gruppieren”.

Rangumwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen festgelegt wird, wie der Data Integration Service Daten für die Rang-Umwandlung verarbeitet.

Konfigurieren Sie die folgenden Eigenschaften auf der Registerkarte **Erweitert**:

Oben/Unten

Gibt an, ob Sie ein Ranking im oberen oder unteren Bereich einer Spalte vornehmen möchten.

Anzahl der Ränge

Anzahl der Zeilen für ein Ranking im oberen oder unteren Bereich.

Zeichenfolgenvergleich unter Beachtung der Groß-/Kleinschreibung

Gibt an, ob der Data Integration Service beim Ranking von Zeichenfolgen Zeichenfolgenvergleiche unter Beachtung der Groß-/Kleinschreibung verwendet. Deaktivieren Sie diese Option, wenn der Data Integration Service die Groß-/Kleinschreibung für Zeichenfolgen ignorieren soll. In der Voreinstellung ist diese Option aktiviert.

Cache-Verzeichnis

Verzeichnis, in dem der Datenintegrationsdienst die Index-Cache- und Daten-Cachedateien erstellt. Stellen Sie sicher, dass das Verzeichnis vorhanden ist und genügend Speicherplatz für die Cachedateien enthält.

Geben Sie mehrere Verzeichnisse durch Semikolon getrennt ein, um die Leistung während der Cache-Partitionierung zu erhöhen. Die Cache-Partitionierung erstellt für jede Partition, die die Umwandlung verarbeitet, einen separaten Cache.

Standardwert ist der CacheDir-Systemparameter. Sie können für diese Eigenschaft einen anderen Systemparameter oder einen benutzerdefinierten Parameter konfigurieren.

Rang-Daten-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Daten-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto” auswählen, berechnet der

Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".

Rang-Index-Cachegröße

Menge an Arbeitsspeicher, die der Datenintegrationsdienst dem Index-Cache für die Umwandlung zu Beginn der Mapping-Ausführung zuweist. Wenn Sie „Auto" auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte an. Der Standardwert ist „Auto".

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz", „Normal", „Ausführliche Intialisierung" oder „Ausführliche Daten" wählen. Standardwert ist „Normal".

Verwandte Themen:

- "Cachegröße" auf Seite 662

KAPITEL 33

Leseumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Leseumwandlung – Übersicht, 501
- Eigenschaften von Leseumwandlungen , 501
- Relationale Datenobjekte synchronisieren, 505
- Das Quelldatenobjekt ändern, 506
- Parameter von Leseumwandlungen, 508
- Einschränkungen , 508
- Eine Leseumwandlung erstellen, 509

Leseumwandlung – Übersicht

Die Leseumwandlung ist eine passive Umwandlung, bei der Daten aus einer Quelle gelesen werden. Sie können eine Leseumwandlung aus einem physischen Datenobjekt oder einem logischen Datenobjekt erstellen. Die Leseumwandlung ist nicht wiederverwendbar.

Sie können für eine Leseumwandlung verschiedene Eigenschaften konfigurieren, basierend auf dem Typ des Datenobjekts, das Sie zum Erstellen der Umwandlung verwendet haben. Wenn Sie beispielsweise eine Leseumwandlung aus einem relationalen Datenobjekt erstellen, können Sie SQL-Überschreibungen konfigurieren und Einschränkungen definieren. Die Eigenschaften, die Sie konfigurieren können, hängen auch davon ab, ob Sie Parameter für die Umwandlung konfiguriert haben.

Leseumwandlungen können dynamische Quellen enthalten. Sie können eine Leseumwandlung so konfigurieren, dass deren Ports, Metadaten und andere Eigenschaften dynamisch aktualisiert werden. Informationen zum Konfigurieren von dynamischen Quellen finden Sie im Kapitel „Dynamische Mappings“ im *Informatica Developer-Handbuch für Mappings*.

Eigenschaften von Leseumwandlungen

Nach dem Erstellen einer Leseumwandlung können Sie Eigenschaften für die Umwandlung konfigurieren.

Konfigurieren Sie die Eigenschaften von Leseumwandlungen auf den Eigenschaften-Registerkarten. Die Registerkarten, die Sie verwenden können, hängen vom Typ der Quelle ab, die die Leseumwandlung darstellt.

In der folgenden Tabelle wird jede Eigenschaften-Registerkarte beschrieben und der Quelltyp angegeben, für den die Registerkarte verwendet wird:

Eigenschaften-Registerkarte	Beschreibung	Quelltyp
Allgemein	Geben Sie Umwandlungseigenschaften und -verhalten an. Synchronisieren Sie für relationale und benutzerdefinierte Datenobjektquellen die Eingabeports der Umwandlung mit der Quelle.	Alle
Datenobjekt	Geben Sie die Datenquelle der Umwandlung an.	- Relational - Einfachdatei - Benutzerdefiniertes Datenobjekt
Ports	Legen Sie die Portdefinition nach dem zugehörigen Datenobjekt fest.	- Relational - Benutzerdefiniertes Datenobjekt - Logisches Datenobjekt
Format	Eingabeeinstellungen für eine Einfachdatei-Datenquelle	Einfachdatei
Abfrage	Geben Sie eine Abfrage an die Quelle an.	- Relational - Benutzerdefiniertes Datenobjekt - Logisches Datenobjekt
Laufzeit	Definieren Sie das Laufzeitverhalten.	- Relational - Einfachdatei - Benutzerdefiniertes Datenobjekt
Quellen	Wählen Sie Quelltabellen aus und konfigurieren Sie Quelldetails.	- Relational - Benutzerdefiniertes Datenobjekt
Datenobjektparameter	Legen Sie Parametereigenschaften fest.	- Einfachdatei - Benutzerdefiniertes Datenobjekt - Logisches Datenobjekt
Verknüpfung zur Laufzeit	Konfigurieren Sie eine Verknüpfung von Gruppen zwischen Umwandlungen, bei der ein Parameter, eine Verknüpfungsrichtlinie oder beides verwendet wird, um zu bestimmen, welche Ports zur Laufzeit verknüpft werden sollen.	Alle
Erweitert	Legen Sie die Tracingebene und Zeilenreihenfolge fest. Legen Sie für eine relationale Quelle die Option zum Erstellen oder Ersetzen der Zieltabelle zur Laufzeit fest.	Alle

Allgemeine Eigenschaften

Sie können den Namen und die Beschreibung der Leseumwandlung konfigurieren. Sie können zudem die folgenden Eigenschaften konfigurieren:

Bei Änderung der Spaltenmetadaten

Verfügbar für relationale Quellen. Wählen Sie eine der folgenden Optionen aus:

- Ausgabeports synchronisieren. Das Developer Tool aktualisiert die Ausgabeports der Leseumwandlung mit den Änderungen der Metadaten, die das Modellrepository für das Datenobjekt speichert.
- Nicht synchronisieren. Die Leseumwandlung zeigt keine Änderungen der Metadaten im Datenobjekt an.

Physisches Datenobjekt

Verfügbar für Einfachdateiquellen und benutzerdefinierte Quellen. Das zum Erstellen der Umwandlung verwendete Objekt.

Sie können den Dateiobjektnamen auswählen und dessen Eigenschaften konfigurieren.

Datenobjekteigenschaften

Auf der Registerkarte „Datenobjekt“ können Sie die Quelle der Leseumwandlung angeben bzw. ändern sowie Einfachdatei-Datenobjektquellen, relationale und benutzerdefinierte Datenobjektquellen in dynamische Datenobjektquellen ändern.

Sie können die folgenden Eigenschaften konfigurieren:

Angeben nach

Wählen Sie für die Angabe von Quellspalten und Metadaten für die Leseumwandlung eine der folgenden Optionen aus:

- Wert. Die Leseumwandlung verwendet das zugehörige Datenobjekt, um Quellspalten und Metadaten anzugeben.
- Parameter. Die Leseumwandlung verwendet einen Parameter, um Quellspalten und Metadaten anzugeben.

Datenobjekt

Wenn Sie die Leseumwandlung aus einem vorhandenen Datenobjekt erstellt haben, wird im Feld der Name des Objekts angezeigt. Klicken Sie auf **Durchsuchen**, um das Datenobjekt zu ändern, das der Leseumwandlung zugeordnet werden soll.

Datenobjektspalten zur Laufzeit aus Datenquelle abrufen

Wenn Sie diese Option aktivieren, ruft der Datenintegrationsdienst Metadaten und Änderungen bei der Datendefinition aus den Quelltabellen in die Leseumwandlung ab.

Abfrageeigenschaften

Konfigurieren Sie eine SQL-Abfrage aus einer relationalen Ressource oder einem benutzerdefinierten Datenobjekt.

Beim Konfigurieren der Eigenschaften auf der Registerkarte **Abfrage** können Sie einfache oder erweiterte Eigenschaften konfigurieren.

In der Eigenschaftenansicht **Einfach** konfigurieren Sie die Standard-SQL-Anweisung als Define-Distinct-Anweisung sowie zum Bearbeiten der Hinweise und der Join-, Filter- und Sortierbedingungen für die Anweisung.

In der Eigenschaftenansicht **Erweitert** können Sie eine benutzerdefinierte SQL-Abfrage definieren. Sie können aus Spalten im zugehörigen Datenobjekt oder aus Parametern auswählen oder aber einen neuen Parameter erstellen, der ein Datenobjekt darstellt.

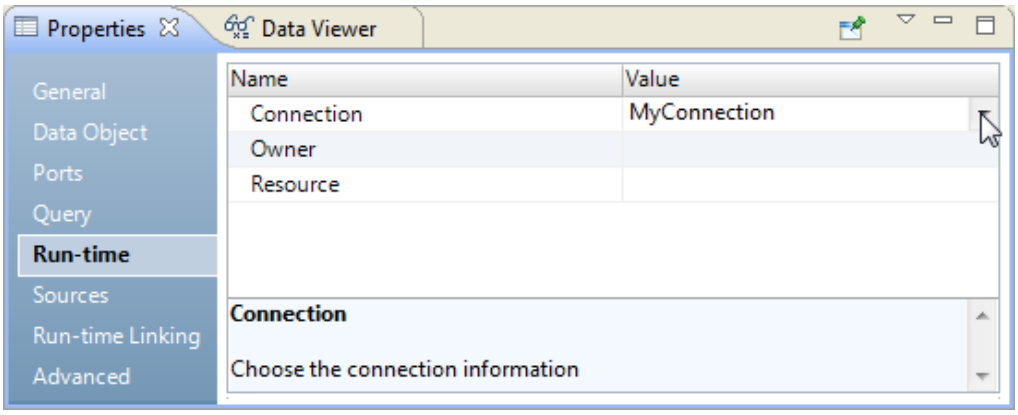
Laufzeiteigenschaften

Sie können die folgenden Eigenschaften für Leseumwandlungen auf der Registerkarte **Laufzeit** konfigurieren:

Verbindung

Verfügbar für relationale Quellen. Die von der Umwandlung verwendete Verbindung. Klicken Sie in die rechte Seite des Felds, um die Verbindung zu ändern.

Die folgende Abbildung zeigt die Position der Dropdown-Schaltfläche, auf die Sie klicken können:



Quelleneigenschaften

Konfigurieren Sie die Details von Quellen für relationale Ressourcen und benutzerdefinierte Datenobjekte. Sie können die Definition des relationalen Datenobjekts nach dem Importieren in das Repository ändern. Sie können Ports hinzufügen und entfernen, Primärschlüssel definieren und Beziehungen zwischen mehreren relationalen Datenobjekten im Repository konfigurieren.

Die Registerkarte **Quellen** ermöglicht Ihnen das Konfigurieren der folgenden Einstellungen:

Alle Quellen

Verwenden Sie die Schaltflächen „Hinzufügen“ und „Entfernen“, um zusätzliche Quellen für die Umwandlung hinzuzufügen bzw. zu entfernen.

Registerkarte „Allgemein“

Ändern Sie den Namen und die Beschreibung für die ausgewählte Quelle. Klicken Sie auf den Namen der Quelle, um weitere Details zu ändern.

Registerkarte „Schlüssel“

Legen Sie Ressourcenspalten als Schlüssel fest.

Registerkarte „Beziehungen“

Fügen Sie Beziehungen zwischen mehreren relationalen Ressourcen hinzu oder entfernen Sie diese.

Erweiterte Eigenschaften

Der Datenintegrationsdienst kann SQL-Befehle ausführen.

Der Datenintegrationsdienst führt Pre-Mapping-SQL-Befehle für die Quelldatenbank aus, bevor er die Quelle liest. Er führt Post-Mapping-SQL-Befehle für die Quelldatenbank aus, nachdem er die Quelle gelesen hat. Sie können die folgenden Eigenschaften für Lesetransaktionen auf der Registerkarte „Erweitert“ konfigurieren:

Tracingebene

Steuert die Detailmenge in der Mapping-Protokolldatei.

PreSQL

SQL-Befehl, den der Datenintegrationsdienst für die Quelldatenbank ausführt, bevor er die Quelle liest.

Das Developer Tool validiert die SQL nicht.

PostSQL

SQL-Befehl, den der Datenintegrationsdienst für die Quelldatenbank ausführt, nachdem er die Quelle gelesen hat.

Das Developer Tool validiert die SQL nicht.

Einschränkungen

SQL-Anweisungen für Einschränkungen der referenziellen Integrität auf Tabellenebene. Gilt nur für relationale Quellen.

Relationale Datenobjekte synchronisieren

Sie können physische Datenobjekte synchronisieren, wenn sich deren Quellen ändern. Wenn Sie ein physisches Datenobjekt synchronisieren, importiert das Developer Tool die Objektmetadaten aus der von Ihnen ausgewählten Quelle erneut.

Sie können alle physischen Datenobjekte synchronisieren. Wenn Sie relationale Datenobjekte oder benutzerdefinierte Datenobjekte synchronisieren, können Sie die von Ihnen im Developer Tool definierten Schlüsselbeziehungen beibehalten oder überschreiben.

Wählen Sie aus mehreren Methoden zum Synchronisieren der Mapping-Objekte aus:

Eine relationale Ressource synchronisieren.

Um ein physisches Datenobjekt zu synchronisieren, klicken Sie in der Ansicht **Objekt-Explorer** mit der rechten Maustaste auf das Objekt und wählen Sie **Synchronisieren** aus.

Umwandlungsports mit dem physischen Datenobjekt synchronisieren.

Wählen Sie auf der Registerkarte „Datenobjekt“ einer Umwandlung die Option **Datenobjektspalten aus Datenquelle abrufen** aus.

Wenn das Mapping ausgeführt wird, ruft der Datenintegrationsdienst Spalteninformationen aus der Datenquelle ab und aktualisiert die Datenobjektdefinition im Modellrepository.

Bei Änderung der Metadaten Ports synchronisieren

Wählen Sie auf der Registerkarte „Allgemein“ die Option zum Synchronisieren von Ports aus. Die genaue Bezeichnung dieser Option hängt vom Typ der konfigurierten Umwandlung ab. Beispiel: Für eine Leseumwandlung lautet die Option **Bei Änderung der Metadaten Ausgabeports synchronisieren**.

Wenn das Mapping ausgeführt wird, synchronisiert der Datenintegrationsdienst die Spaltenmetadaten in der Umwandlung mit den Metadaten in der Datenquelle.

Das Quelldatenobjekt ändern

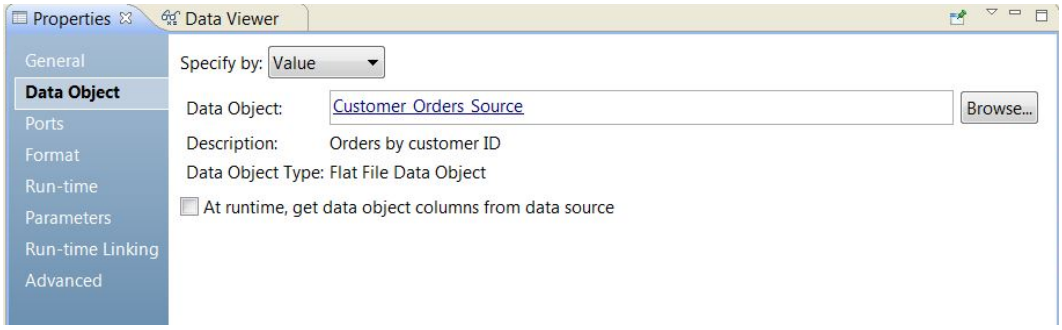
Eine Leseumwandlung basiert auf einem physischen Datenobjekt oder einem logischen Datenobjekt im Modellrepository. Sie können das Datenobjekt beim Konfigurieren einer Leseumwandlung ändern. Sie können das Datenobjekt so parametrisieren, dass es zur Laufzeit geändert wird. Sie können beispielsweise ein Mapping mit einer anderen Quelldatei als derjenigen Quelldatei testen, die Sie für die Ausführung eines Produktions-Mappings verwenden.

Beim Erstellen einer Umwandlung aus einem physischen Datenobjekt werden Informationen zum Datenobjekt auf der Registerkarte **Datenobjekt** der Umwandlungseigenschaften angezeigt. Sie können auf den Namen des Datenobjekts klicken, um die Definition des physischen Datenobjekts aus dem Modellrepository anzuzeigen.

Sie können das Datenobjekt für die Umwandlung ändern, indem Sie nach einem anderen physischen Datenobjekt im Modellrepository suchen. Wenn Sie das Datenobjekt ändern, verwendet die Umwandlung die Laufzeiteigenschaften und die erweiterten Eigenschaften des von Ihnen ausgewählten Datenobjekts.

Sie können die Struktur des Datenobjekts zur Laufzeit basierend auf Änderungen in der Datenquelle aktualisieren. Die Datenquelle ist die physische Datei oder die Datenbanktabelle, die das Datenobjekt darstellt. Beim Aktivieren des Datenintegrationsdiensts zum Abrufen der Datenspalten aus der Datenquelle untersucht der Datenintegrationsdienst die Struktur der Datenquelle. Der Datenintegrationsdienst aktualisiert die Datenobjektports in der Umwandlungsinstanz basierend auf der Datenquelle. Der Datenintegrationsdienst ändert nicht die Definition des physischen Datenobjekts im Modellrepository.

Die folgende Abbildung zeigt die Registerkarte **Datenobjekt**:



Die Registerkarte **Datenobjekt** verfügt über die folgenden Felder:

Angeben nach

Wählen Sie **Wert**, um einen bestimmten Datenobjektnamen einzugeben. Wählen Sie **Parameter**, um das Datenobjekt zu parametrisieren.

Datenobjekt

Der Name des Datenobjekts im Modellrepository. Sie können auf den Link **Datenobjekt** klicken, um die Definition des Datenobjekts aus dem Repository zu öffnen. Sie können auch nach einem anderen Datenobjekt im Modellrepository suchen.

Beschreibung

Die Beschreibung des Datenobjekts im Repository. Schreibgeschützt.

Typ des Datenobjekts

Beschreibt den Typ des Datenobjekts, wie z. B. ein Einfachdatei-Datenobjekt, ein relationales Tabellenobjekt oder ein benutzerdefiniertes Datenobjekt.

Datenobjektspalten zur Laufzeit aus Datenquelle abrufen

Aktualisiert die Struktur des Datenobjekts für die Umwandlungsinstanz zur Laufzeit. Aktualisiert die Struktur basierend auf den Spalten aus der Datendatei oder der Tabelle, auf die das Datenobjekt verweist.

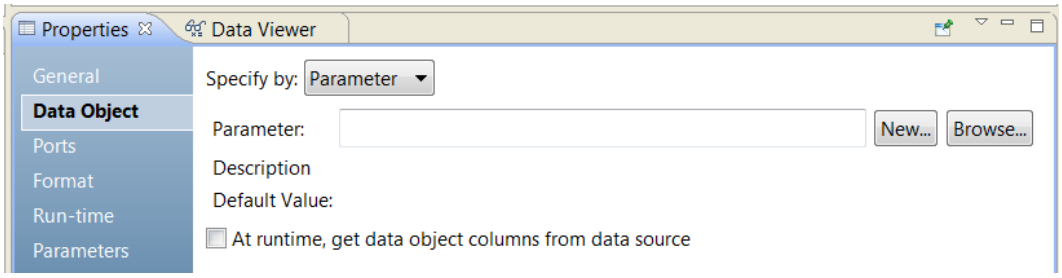
Das Lesedatenobjekt parametrisieren

Sie können das Datenobjekt der Leseumwandlung parametrisieren und das Datenobjekt zur Laufzeit ändern.

Wählen Sie zum Parametrisieren eines Datenobjekts die Option **Angeben nach Parameter** auf der Registerkarte **Datenobjekt** aus. Die Eigenschaften auf der Registerkarte **Datenobjekt** ändern sich.

Um das Datenobjekt zu parametrisieren, erstellen Sie einen Ressourcentypparameter oder suchen Sie nach einem Ressourcenparameter, den Sie bereits erstellt haben. Der Standardwert des Parameters ist der Name des physischen Datenobjekts im Modellrepository. Wenn Sie einen Standardparameterwert erstellen, wählen Sie den Namen eines physischen Datenobjekts aus einer Liste mit Datenobjekten im Repository aus.

Die folgende Abbildung zeigt die Registerkarte **Datenobjekt**, wenn Sie das Datenobjekt nach einem Parameter angeben:



Die Registerkarte **Datenobjekt** verfügt über die folgenden Optionen nach Parameter:

Parameter

Der Name eines Ressourcenparameters, den Sie als Datenobjekt konfiguriert haben. Schreibgeschützt.

Beschreibung

Die Beschreibung des Parameters. Schreibgeschützt.

Neu

Erstellen Sie einen Ressourcenparameter. Suchen Sie im Modellrepository nach einem Datenobjekt und wählen Sie dieses als Standardwert des Parameters aus.

Durchsuchen

Suchen Sie nach einem Ressourcenparameter und wählen Sie den Parameter aus.

Standardwert

Der Standardwert des Ressourcenparameters, den Sie für das Datenobjekt konfiguriert haben. Der Standardwert ist der Name eines physischen Datenobjekts und der Pfad zu dem Objekt im Modellrepository. Schreibgeschützt.

Parameter von Leseumwandlungen

Sie können einige der Eigenschaften einer Leseumwandlung und einige der Eigenschaften des wiederverwendbaren physischen Datenobjekts parametrisieren, aus dem Sie die Leseumwandlung erstellen.

Wenn Sie ein physisches Datenobjekt erstellen, konfigurieren Sie die Lese- und Schreibzugriffseigenschaften. Die Parameter, die Sie für Lesezugriffseigenschaften in einem physischen Datenobjekt konfigurieren, werden auf der Registerkarte **Datenobjektparameter** der Leseumwandlung angezeigt, wenn Sie das Datenobjekt zu einem Mapping hinzufügen.

Sie können Parameter für die folgenden Lesezugriffseigenschaften im physischen Datenobjekt konfigurieren:

- Verzeichnis der Steuerungsdatei
- Steuerungsdateiname
- Standardskalierung
- Delimiter
- Einfachdatei-Delimiter
- Merge-Dateiverzeichnis
- Quelldateiname
- Quelldateiverzeichnis

Nachdem Sie das physische Datenobjekt zu einem Mapping hinzugefügt haben, können Sie die Parameter auf der Registerkarte **Datenobjektparameter** der Leseumwandlung anzeigen. Sie können diese Parameter als Mapping-Parameter anzeigen, um die Parameterwerte zur Laufzeit zu überschreiben.

Sie können die folgenden Mapping-Parameter für die Leseumwandlung konfigurieren:

- Verbindung (relational)
- Datenobjekt
- Reihenfolge der Verknüpfungsauflösung
- Ressourcenname (relational)
- Tabelleneigentümername (relational)

Sie können diese Parameter auf der Registerkarte **Datenobjektparameter** des Mappings anzeigen.

Einschränkungen

Eine Einschränkung ist ein Bedingungsausdruck, dem die Werte in einer Datenzeile entsprechen müssen.

Wenn Sie eine Einschränkung einrichten, geben Sie einen Ausdruck ein, der für jede Datenzeile mit TRUE bewertet wird.

Der Datenintegrationsdienst kann Einschränkungen aus relationalen Quellen, logischen Datenobjekten, physischen Datenobjekten oder virtuellen Tabellen lesen. Erstellen Sie zum Einrichten einer Einschränkung für ein wiederverwendbares physisches Objekt ein benutzerdefiniertes Datenobjekt.

Wenn der Datenintegrationsdienst Einschränkungen liest, lässt er basierend auf der angewendeten Optimierungsmethode unter Umständen die Zeilen außer Acht, die nicht mit TRUE bewertet werden.

Vor dem Einrichten einer Einschränkung müssen Sie sicherstellen, dass die Quelldaten der von der Einschränkung festgelegten Bedingung entsprechen. Beispiel: Eine Quelldatenbank verfügt über eine AGE-Spalte, die zwei Zeilen mit AGE < 70 aufweist. Sie können eine Einschränkung mit AGE < 70 in der

Quelldatenbank einrichten. Der Datenintegrationsdienst liest Datensätze aus der Quelldatenbank mit der Einschränkung AGE < 70. Wenn der Datenintegrationsdienst Datensätze mit AGE >= 70 liest, lässt er unter Umständen die Zeilen mit AGE >= 70 außer Acht.

In der Datenbank können Sie SQL-Befehle verwenden, um Einschränkungen für die Datenbankumgebung einzurichten, wenn Sie eine Verbindung zur Datenbank herstellen. Der Datenintegrationsdienst führt die Verbindungsumgebungs-SQL bei jeder Verbindung mit der Datenbank aus.

Eine Leseumwandlung erstellen

Wenn Sie eine Leseumwandlung erstellen, wählen Sie eine der folgenden Methoden basierend auf der Ressource aus, aus der die Umwandlung erstellt wird:

Die Umwandlung aus einem Datenobjekt im Modellrepository erstellen.

Führen Sie die folgenden Schritte zum Erstellen einer Leseumwandlung aus einem Datenobjekt im Modellrepository aus:

1. Öffnen Sie das Mapping im Editor.
2. Ziehen Sie ein Datenobjekt aus dem **Objekt-Explorer** in die Editoransicht.
3. Wählen Sie **Lesen** aus und klicken Sie auf **OK**.
   Die Leseumwandlung im Mapping enthält die Ports und Eigenschaften des Datenobjekts.

Erstellen Sie die Umwandlung mithilfe des Zuordnungseditors.

Verwenden Sie diese Methode, wenn Sie detaillierte Einstellungen für die Leseumwandlung konfigurieren möchten. Sie können diese Methode zudem verwenden, wenn Sie einer Leseumwandlung einen Parameter zugrunde legen möchten.

Informationen zum Erstellen einer Leseumwandlung im Zuordnungseditor finden Sie unter “Erstellen einer Leseumwandlung im Zuordnungseditor” auf Seite 509.

Erstellen einer Leseumwandlung im Zuordnungseditor

Sie können eine Leseumwandlung erstellen, um die Quelle der Daten und Spaltenmetadaten sowie Eigenschaften in einem Mapping darzustellen.

Führen Sie die folgenden Schritte durch:

1. Wählen Sie eine der folgenden Methoden zum Erstellen einer Leseumwandlung aus:
   - Klicken Sie mit der rechten Maustaste in den Zuordnungseditor und wählen Sie **Umwandlung hinzufügen** aus.
     Das Dialogfeld **Umwandlung hinzufügen** wird geöffnet.

     Wählen Sie „Leseumwandlung“ aus und klicken Sie auf **Weiter**.
   - Scrollen Sie in der Mapping-Palette nach unten, um das Symbol für die Leseumwandlung zu suchen und zu doppelklicken.

Die folgende Abbildung zeigt das Symbol für die Leseumwandlung:

Das Dialogfeld **Neue Leseumwandlung** wird geöffnet.

2. Zur Verwendung einer Einfachdatei, einer relationalen Ressource oder eines benutzerdefinierten Datenobjekts als Quelle, führen Sie die folgenden Schritte aus:
   a. Wählen Sie **Physisches Datenobjekt** als Datenobjekttyp aus.
   b. Klicken Sie auf **Durchsuchen**, um eine Einfachdatei, eine relationale Ressource oder ein benutzerdefiniertes Datenobjekt auszuwählen.

      Das Fenster **Datenobjekt auswählen** wird geöffnet.
   c. Wählen Sie ein Datenobjekt aus und klicken Sie auf **OK**.
   d. Konfigurieren Sie optional die Umwandlung zum Abrufen von Datenobjektspalten aus der Quelle zur Laufzeit aus. Wählen Sie **Datenobjektspalten zur Laufzeit aus Datenquelle abrufen** aus.

      Der Datenintegrationsdienst aktualisiert die Spaltenmetadaten für die Leseumwandlung, wenn das Mapping ausgeführt wird.
3. Führen Sie zum Verwenden eines Parameters als Quelle die folgenden Schritte aus:
   a. Wählen Sie **Mithilfe eines Parameters erstellen** aus.
   b. Klicken Sie auf **Neu**, um einen neuen Parameter zu erstellen, oder klicken Sie auf **Durchsuchen**, um einen vorhandenen Parameter auszuwählen.
   c. Wählen Sie einen Parameter aus und klicken Sie auf **OK**.
   d. Konfigurieren Sie optional die Umwandlung zum Abrufen von Datenobjektspalten aus der Quelle zur Laufzeit aus. Wählen Sie **Datenobjektspalten zur Laufzeit aus Datenquelle abrufen** aus.

      Der Datenintegrationsdienst aktualisiert die Spaltenmetadaten für die Leseumwandlung, wenn das Mapping ausgeführt wird.
4. Führen Sie zum Verwenden eines logischen Datenobjekts als Quelle die folgenden Schritte aus:
   a. Wählen Sie **Logisches Datenobjekt** als Datenobjekttyp aus.
   b. Klicken Sie auf **Durchsuchen**, um ein Datenobjekt auszuwählen, und klicken Sie anschließend auf **OK**.
5. Geben Sie optional einen Namen für die Leseumwandlung ein.
6. Klicken Sie auf **Fertig stellen**.

KAPITEL 34

Relational-in-hierarchisch-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Übersicht über Relational-in-Hierarchisch-Umwandlungen, 511
- Beispiel - Relational-in-hierarchisch-Umwandlung, 512
- Relationale Eingabeports und die Ansicht „Übersicht“, 514
- Ports für Relational-in-hierarchisch-Umwandlungen, 515
- Schemaverweise, 515
- Entwicklung der Relational-in-hierarchisch-Umwandlung, 515

Übersicht über Relational-in-Hierarchisch-Umwandlungen

Die Relational-in-hierarchisch-Umwandlung verarbeitet relationale Eingaben und wandelt sie in hierarchische Ausgaben um. Eine Relational-in-hierarchisch-Umwandlung liest relationale Eingaben von Eingabeports und wandelt die Daten in eine hierarchische Ausgabe am Umwandlungsausgabeport um. Um relationale Eingaben in hierarchische Ausgaben umzuwandeln, verwenden Sie ein Schemaobjekt zum Definieren der hierarchischen Struktur.

Sie können die Assistenten für Relational-in-hierarchisch-Umwandlungen zur Erstellung einer hierarchischen Struktur verwenden, die die relationalen Eingabeports wiedergeben. Sie können das Mapping für die hierarchischen Ausgabeports in der Ansicht **Übersicht** der Umwandlung anzeigen.

Nach Erstellung der Umwandlung können Sie die Daten aus dem hierarchischen Ausgabeport an eine andere Umwandlung in einem Mapping übergeben.

Im relationalen Modell gibt jedes Tabellenschema eine Spalte an, um jede Zeile eindeutig anzugeben. Diese Spalte wird als Primärschlüssel bezeichnet. Sie geben die Beziehung zwischen jeder Zeile in der Tabelle und einer Zeile in einer anderen Tabelle mit einem Fremdschlüssel an. Der Assistent generiert Schlüssel, wenn er die Umwandlung erstellt. Sie können eine automatisch generierte Umwandlung ändern und Ports hinzufügen, bearbeiten oder löschen.

Sie können einen relationalen Eingabeport mit einem Knoten in der Hierarchie verknüpfen. Verknüpfen Sie einen Primärschlüssel aus dem jeweiligen Element oder Attribut in der Hierarchie mit einer relationalen Gruppe in der Eingabe. Der Primärschlüssel gibt jede Zeile in den relationalen Tabellen an.

Verknüpfen Sie einen Fremdschlüssel aus dem jeweiligen Element oder Attribut in der Hierarchie mit einer relationalen Gruppe in der Eingabe. Ein Fremdschlüssel in der relationalen Eingabe ist eine Spalte in einer Tabelle, die auf den Primärschlüssel einer anderen Tabelle zeigt.

Der relationale Eingabeport und der hierarchische Ausgabeknoten müssen kompatible Datentypen aufweisen.

Beispiel - Relational-in-hierarchisch-Umwandlung

Die Finanzabteilung des Unternehmens „Electronics Superstore muss Gehaltszahlungen für die Unternehmensmitarbeiter verarbeiten. Dazu müssen in einer relationalen Datenbank gespeicherte Mitarbeiterdaten in ein hierarchisches Format umgewandelt werden, das für das Zahlungssystem geeignet ist.

Beim Mapping müssen anhand einer Relational-in-hierarchisch-Umwandlung Mitarbeiterdetails wie Mitarbeiternamen, Mitarbeiter-IDs, Mitarbeiteradressen und Bankkontoangaben eingegeben und die Details in einem geeigneten hierarchischen Format ausgegeben werden.

In der relationalen Eingabe ist das Element „Bank_ID“ ein Primärschlüssel in der Mitarbeitertabelle und ein Fremdschlüssel in der Banktabelle:

Employee_ID	Last_Name	First_Name	Address	Bank_ID	Bank_Account
9173327437	Sandrine	Jacques	74 Mobile Avenue	74845	8723487234
9174562342	Race	Tom	266 Crouse St.	9234734	45324734
8484526471	Jones	Charles	3815 LaValle Boulevard	389236	234638437
7023847265	Smith	Delilah	193 Short Drive	74845	8723463432
9174596725	Frederick	George	17 Serenity Road	9234734	6342636699

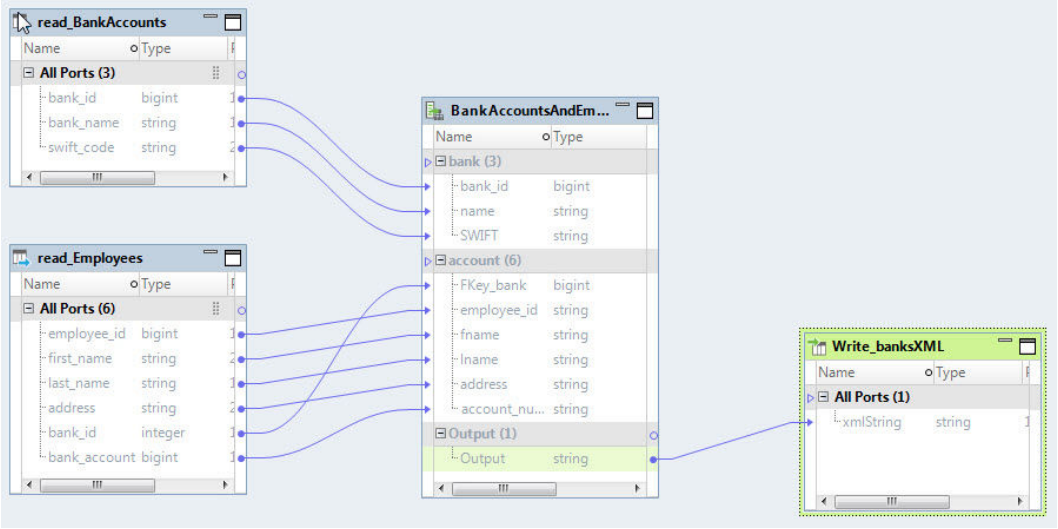
Bank_ID	Bank_Name	SWIFT_Code
74845	National Bank	9173327
9234734	International Bank	9174562
389236	Star National Bank	8484526

In der Zahlungsausgabe in hierarchischem Format werden die Elemente aus den Tabellen miteinander kombiniert:

```
<banks>
    <bank name="National Bank" SWIFT="9173327">
        <account id="8723487234">
            <employee_id>9173327437</employee_id>
            <fname>Sandrine</fname>
            <lname>Jacques</lname>
            <address>74 Mobile Avenue</address>
        </account>
        <account id="8723463432">
            <employee_id>9082745558</employee_id>
            <fname>Delilah</fname>
            <lname>Smith</lname>
            <address>193 Short Drive</address>
        </account>
    </bank>
    <bank name="International Bank" SWIFT="9174562">
        <accounts>
            <account id="45324734">
                <employee_id>5534398889</employee_id>
                <fname>Race</fname>
                <lname>Tom</lname>
                <address>266 Crouse St.</address>
            </account>
            <account id="6342636699">
                <employee_id>9174596725</employee_id>
                <fname>Frederick</fname>
```

```
                <lname>George</lname>
                <address>17 Serenity Road</address>
            </account>
        </accounts>
    </bank>
    <bank name="Star National Bank" SWIFT="8484526">
        <accounts>
            <account id="234638437">
                <employee_id>8484526471</employee_id>
                <fname>Jones</fname>
                <lname>Charles</lname>
                <address>3815 LaValle Boulevard</address>
            </account>
        </accounts>
    </bank>
</banks>
```

Die folgende Abbildung verdeutlicht das Mapping in diesem Beispiel:



Das Mapping enthält die folgenden Objekte:

Read_BankAccounts

Die Quelle, die die Bankdaten enthält.

Read_Employees

Die Quelle mit den Mitarbeiterdaten.

BankAccountsAndEmployees_To_PaymentsSystemXML

Eine Relational-in-hierarchisch-Umwandlung, mit der die relationale Eingabe mit den Angaben zu den Mitarbeitern und Bankverbindungen in ein XML-Format umgewandelt wird, das für das Zahlungssystem geeignet ist.

Write_BanksXML

Ein Zielpfad zur Datei, in der die umgewandelten Daten bei jeder Ausführung des Mappings gespeichert werden.

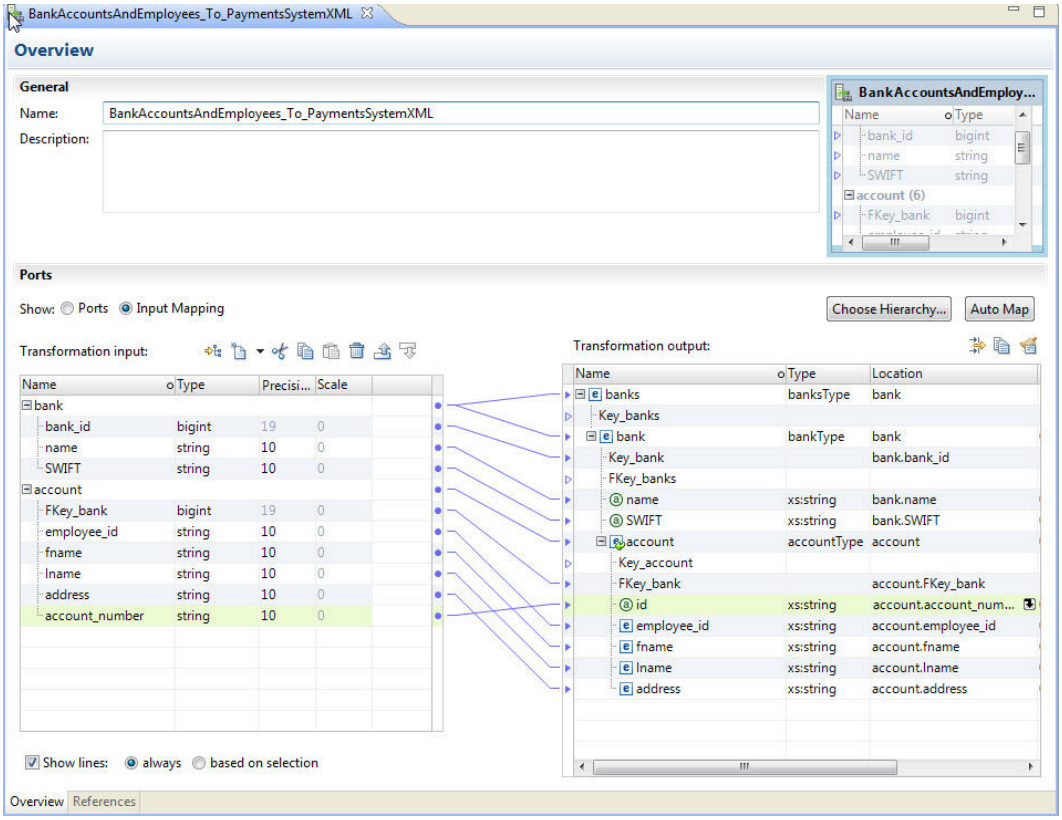
Beim Mapping werden die Dateien „Read_BankAccount“ und „Read_Employees“ zur Bereitstellung der relationalen Eingabe herangezogen. Beim Mapping werden die Daten anhand der Umwandlung „BankAccountsAndEmployees_To_PaymentsSystemXML“ verarbeitet und umgewandelt. Danach speichert das Mapping die Ausgabe unter dem Zielpfad, der in der Einfachdatei „Write_BanksXML“ angegeben ist.

Relationale Eingabeports und die Ansicht „Übersicht“

Zum Umwandeln von relationalen Daten in hierarchische Daten erstellt der Assistent eine hierarchische Struktur, die die relationalen Eingabeports wiedergibt. Sie können in der Ansicht **Übersicht** relationale Ports mit hierarchischen Ports verknüpfen.

Verwenden Sie die Ansicht **Übersicht** zur Anzeige der Verknüpfungen zwischen relationaler Eingabe und hierarchischer Ausgabe. Wählen Sie **Eingabe-Mapping** aus. Der Bereich **Ports** erscheint in der Ansicht **Übersicht**.

Die folgende Abbildung zeigt den Bereich **Ports**:



Links neben dem Bereich **Ports** befindet sich der Bereich **Umwandlungseingabe**, der die relationalen Elemente und Gruppen enthält. Rechts befindet sich der Bereich **Umwandlungsausgabe**, der die hierarchischen Schema-Knoten enthält.

Sie können Ports in der **Umwandlungseingabe** erstellen und relationale Elemente mit den Schema-Knoten verknüpfen. Sie können auch den Mauszeiger aus einem Knoten auf ein leeres Feld im Bereich **Umwandlungseingabe** ziehen, um einen Port zu erstellen. Wenn Sie einen relationalen Port mit einem Schema-Knoten verbinden, zeigt das Developer Tool eine Verknüpfung zwischen ihnen an.

Ports für Relational-in-hierarchisch-Umwandlungen

Die Ports für Relational-in-hierarchisch-Umwandlungen werden in der Ansicht **Übersicht** der Umwandlung definiert.

Eine Relational-in-hierarchisch-Umwandlung kann relationale Datenausgaben aus einem Puffer lesen. Die Ausgabeports geben hierarchische Daten in einen Puffer aus.

Schemaverweise

Eine Relational-in-hierarchisch-Umwandlung erfordert ein hierarchisches Schema zum Definieren der Ausgabehierarchie in der Umwandlung. Zum Verwenden des Schemas in der Umwandlung definieren Sie einen Schemaverweis.

Sie können Schemaverweise für Umwandlungen in der Ansicht **Referenzen** der Umwandlung anzeigen.

Die Relational-in-hierarchisch-Umwandlung verweist auf Schemaobjekte im Modellrepository. Die Schemaobjekte können im Repository vorhanden sein, bevor Sie die Umwandlung erstellen.

Ein Schema kann auf zusätzliche Schemas verweisen. Die Ansicht **Referenzen** zeigt den Namespace und das Präfix für jedes Schema an, auf das die Relational-in-hierarchisch-Umwandlung verweist.

Entwicklung der Relational-in-hierarchisch-Umwandlung

Verwenden Sie den Assistenten „Neue Umwandlung“ zur automatischen Generierung einer Relational-in-hierarchisch-Umwandlung. Wählen Sie ein Schema oder eine hierarchische Beispieldatei zum Definieren der Ausgabehierarchie.

Erstellen der Relational-in-hierarchisch-Umwandlung

1. Klicken Sie im Developer Tool auf **Datei** > **Neu** > **Umwandlung**.
2. Wählen Sie die Relational-in-hierarchisch-Umwandlung aus und klicken Sie auf **Weiter**.
3. Geben Sie einen Namen für die Umwandlung ein und suchen Sie nach einem Modellrepository-Speicherort, um die Umwandlung abzulegen. Klicken Sie dann auf **Weiter**.
4. Halten Sie sich zur Auswahl eines Schemas an eine der folgenden Methoden:
   - Um ein Schema aus dem Modellrepository zum Definieren der Ausgabehierarchie zu verwenden, suchen Sie die Schemadatei in der Nähe des Felds **Schemaobjekt**, um diese dann aus dem Repository auszuwählen.
   - Klicken Sie zum Importieren einer Schemadatei auf **Neues Schemaobjekt erstellen**. Im Fenster **Neues Schemaobjekt** können Sie zu einer Schemadatei navigieren und diese auswählen oder wahlweise ein Schema aus einer Beispiel-Hierarchiedatei erstellen.
5. Wählen Sie das Root-Element für die Ausgabehierarchie. Wählen Sie im Dialogfeld **Hierarchie-Root** das Element im Schema aus, bei dem es sich um das Root-Element für die Ausgabehierarchiedatei handelt.

Sie können zur Auswahl des Root-Objekts eine Beispiel-Hierarchiedatei hinzufügen. Um eine Beispieldatei hinzuzufügen, navigieren Sie in der Nähe des Felds **Beispieldatei** zu der Datei im Dateisystem, die Sie auswählen möchten.

6. Klicken Sie auf **Fertig stellen**.

   Der Assistent erstellt die Umwandlung im Repository.

Erstellen der Ports

Konfigurieren Sie die Ports in der Ansicht **Übersicht**.

1. Wählen Sie zur Anzeige des Mappings im Bereich **Ports** der Ansicht **Übersicht** die Option **Eingabe-Mapping** aus.
2. Wählen Sie den Typ des Eingabeports, die Präzision und die Größenordnung aus.
3. Erweitern Sie die Bäume im Gitter **Ports**. Links wird im Bereich **Umwandlungseingabe** die relationale Eingabe und rechts im Bereich **Umwandlungsausgabe**die erwartete hierarchische Ausgabe angezeigt.
4. Wenn Sie einen Knoten als Root-Element definieren möchten, klicken Sie auf **Hierarchie auswählen**.

   Im Developer Tool werden nur die Knoten ab Stammebene angezeigt. Unterhalb der Stammebene werden diese im Bereich **Umwandlungseingabe** angezeigt.
5. Wenn Linien angezeigt werden sollen, die die Ports mit den hierarchischen Knoten verbinden, klicken Sie auf **Linien zeigen**. Legen Sie fest, ob alle Verbindungslinien angezeigt werden sollen oder nur die Linien für bestimmte Ports.
6. Verwenden Sie zum Hinzufügen einer Eingabegruppe oder eines Eingabeports zum Bereich **Umwandlungseingabe** eine der folgenden Methoden:
   - Ziehen Sie ein einfaches oder komplexes Element aus dem Bereich **Umwandlungsausgabe** in eine leere Spalte im Bereich **Umwandlungseingabe**. Wenn es sich um einen Gruppenknoten handelt, fügt das Developer Tool eine relationale Gruppe ohne Ports hinzu.
   - Um eine relationale Gruppe hinzuzufügen, wählen Sie eine Zeile aus und klicken Sie mit der rechten Maustaste darauf. Wählen Sie dann **Neu** > **Gruppe** aus.
   - Um einen relationalen Port hinzuzufügen, klicken Sie mit der rechten Maustaste und wählen Sie **Neu** > **Feld** aus.
7. Verwenden Sie eine der folgenden Methoden, um die Einstellungen für den hierarchischen Knoten für Speicherorte von Ports zu löschen:
   - Wählen Sie einen oder mehrere Knoten im Bereich **Umwandlungsausgabe** aus, klicken Sie mit der rechten Maustaste und wählen Sie **Löschen** aus.
   - Wählen Sie mindestens eine Verbindungslinie zwischen den relationalen Eingabeports und den hierarchischen Knoten aus, klicken Sie mit der rechten Maustaste und wählen Sie **Löschen** aus.
8. Klicken Sie zum Anzeigen der Ausgabeports in einer Hierarchie auf **Als Hierarchie anzeigen**. Jede untergeordnete Gruppe wird unterhalb der übergeordneten Gruppe angezeigt.

KAPITEL 35

REST-Webdienst-Verbraucher-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- REST-Webdienst-Verbraucher-Umwandlung - Übersicht, 517
- REST-Webdienst-Verbraucher-Umwandlung - Konfiguration, 519
- HTTP-Methoden, 521
- REST-Webdienst-Verbraucher-Umwandlung - Ports, 524
- REST-Webdienst-Verbraucher-Umwandlung - Eingabe-Mapping, 527
- REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping, 529
- REST-Webdienst-Verbraucher-Umwandlung - Erweiterte Eigenschaften, 531
- REST-Webdienst-Verbraucher-Umwandlung - Erstellung, 532

REST-Webdienst-Verbraucher-Umwandlung - Übersicht

Die REST-Webdienst-Verbraucher-Umwandlung ist eine aktive Verbindung, die sich mit einem REST-Webdienst als Webdienst-Client für den Zugriff auf oder die Umwandlung von Daten verbindet. Verwenden Sie eine REST-Webdienst-Verbraucher-Umwandlung, um eine Verbindung zu einem REST-Webdienst herzustellen. Die REST-Webdienst-Verbraucher-Umwandlung kann eine Anfrage an einen REST-Webdienst senden und eine Antwort von einem REST-Webdienst empfangen.

Die REST-Webdienst-Verbraucher-Umwandlung stellt eine Verbindung zu einem Webdienst über eine URL her, die Sie in der Umwandlung oder in einer HTTP-Verbindung definieren. Sie können auch eine HTTPS-Verbindung verwenden.

Ein REST-Webdienst enthält eine HTTP-Methode für jede Aktion, die der Webdienst unterstützt. Wenn der Datenintegrationsdienst eine Verbindung zu einem REST-Webdienst herstellt, kann er eine Anfrage zum Abrufen, Übertragen, Einfügen oder Löschen von Daten senden. Die Anfrage kann auf einzelne Ressourcen oder Ressourcensammlungen reagieren. Nachdem der Datenintegrationsdienst eine Anfragemeldung gesendet hat, empfängt er eine Antwortnachricht vom Webdienst.

Die Anfrage- und Antwortmeldungen enthalten XML- oder JSON-Daten mit Elementen, die eine Hierarchie bilden können. Wenn eine Anfrage oder eine Antwortmeldung mehrfach vorkommende Elemente enthält,

bilden die Elementgruppen Ebenen in der XML- oder JSON-Hierarchie. Die Gruppen werden verbunden, wenn eine Ebene mit einer anderen geschachtelt wird.

In der REST-Webdienst-Verbraucher-Umwandlung definieren die Methodeneingabe und die Methodenausgabe die Struktur der Anfrage- und Antwortmeldungen. Die Methodeneingabe und die Methodenausgabe beinhalten Mappings, die definieren, wie die Meldungselemente in den Eingabe- und Ausgabeports zugeordnet werden.

Die REST-Webdienst-Verbraucher-Umwandlung unterstützt Proxy-Server. Sie können auch eine Verbindung zu einer Microsoft SharePoint-Anwendung mit der REST-Webdienst-Verbraucher-Umwandlung herstellen.

Beispiel

Ein Online-Speicher definiert Ressourcen für eine Produktdatenbank. Die Datenbank identifiziert jedes Produkt nach Teilenummer.

Webdienst-Clients greifen auf die Produktdetails über einen REST-Webdienst zu. Der Webdienst verwendet die folgende URL:

```
http://www.HypoStores.com/products/ProductDetails
```

Sie müssen Details zu einem bestimmten Produkt abrufen, wie zum Beispiel die Beschreibung und der Stückpreis, und die Details an eine nachgelagerte Umwandlung in einem Mapping übergeben. Erstellen Sie eine REST-Webdienst-Verbraucher-Umwandlung, um Details zu einem Produkt abzurufen und diese an eine andere Umwandlung zu übergeben.

In der folgenden Tabelle werden die von Ihnen konfigurierten Konfigurationsdetails angezeigt:

Umwandlungsdetails	Wert
HTTP-Methode	Get
Basis-URL	`http://www.HypoStores.com/products/ProductDetails`
Eingabeargumentport	Part_No
Ausgabeports	Beschreibung, Unit_Price
Methodenausgabe	<Die Struktur der Antwortmeldung.>

Die Methodenausgabe enthält ein Ausgabe-Mapping, das definiert, wie die Elemente in der Antwortmeldung zu den Ausgabeports zugeordnet werden.

Wenn der Datenintegrationsdienst die Anfrage an den Webdienst sendet, hängt er den Wert im Argumentport an die Basis-URL an. Beispiel: Zum Abrufen von Details zu Teil 0716 verwendet der Datenintegrationsdienst die folgende URL:

```
http://www.HypoStores.com/products/ProductDetails?Part_No=0716
```

Wenn der Datenintegrationsdienst eine Antwort erhält, wandelt er die Produktbeschreibung und den Stückpreis in der Antwortmeldung in Daten für die Ausgabeports um.

Sie können auch Part_No als ein Parameter übergeben und den zentral gesteuerten Wert bei der Ausführung des Mappings übergeben.

REST-Webdienst-Verbraucher-Umwandlung - Prozess

Die REST-Webdienst-Verbraucher-Umwandlung erstellt eine Anfragemeldung basierend auf den Daten in den Eingabeports und der Methodeneingabe. Sie wandelt Elemente in der Antwortmeldung in Daten für die Ausgabeports basierend auf der Methodenausgabe um.

Die Eingabeports der REST-Webdienst-Verbraucher-Umwandlung enthalten relationale Daten aus vorangestellten Umwandlungen in einem Mapping. Der Datenintegrationsdienst verwendet die Methodeneingabe zum Konvertieren von Daten aus den Eingabeports in Elemente in der Anfragemeldung.

Um eine Verbindung zum Webdienst herzustellen, liest der Datenintegrationsdienst die HTTP-Verbindung bzw. die Basis-URL, die Sie in den Umwandlungseigenschaften konfiguriert haben. Er identifiziert die Ressource, die Sie abrufen, übertragen, eingeben oder löschen möchten, indem er Werte aus den URL-Ports oder Argumentports an die Basis-URL anhängt.

Wenn der Datenintegrationsdienst eine Antwort erhält, übergibt er die Daten in der Antwortmeldung an die Ausgabeports der Umwandlung. Der Datenintegrationsdienst übergibt Daten basierend auf der Konfiguration der Methodenausgabe. Die Ausgabeports enthalten relationale Daten. Der Datenintegrationsdienst sendet die Daten an die Ausgabeports an nachgelagerte Umwandlungen im Mapping oder an das Ziel.

REST-Webdienst-Verbraucher-Umwandlung - Konfiguration

Beim Erstellen einer REST-Webdienst-Verbraucher-Umwandlung wählen Sie die HTTP-Methode aus und definieren die Methodeneingabe und Methodenausgabe. Wenn Sie die GET-Methode auswählen, definieren Sie keine Methodeneingabe.

Die Eingabe-Elemente in der HTTP-Anfragemeldung werden Eingabeports zugeordnet. Die Ausgabe-Elemente in der HTTP-Antwortmeldung werden Ausgabeports zugeordnet. Das Developer-Tool erstellt Ports für die Elemente der ersten Ebene.

Wenn Sie die Umwandlung konfigurieren, führen Sie die folgenden Aufgaben durch:

1. Wählen Sie die HTTP-Methode aus.
2. Konfigurieren Sie Ports, um Elemente in der Kopfzeile und im Hauptteil der Anfrage- und Antwortmeldungen darzustellen.
3. Konfigurieren Sie das Eingabe-Mapping.
4. Konfigurieren Sie das Ausgabe-Mapping.
5. Konfigurieren Sie erweiterte Eigenschaften wie die Verbindung und die Basis-URL für den Webdienst.

Wenn der REST-Webdienst Authentifizierung erfordert, erstellen Sie ein HTTP-Verbindungsobjekt.

Meldungskonfiguration

Der Datenintegrationsdienst generiert Anfragemeldungen und interpretiert Antwortmeldungen basierend auf der Methodeneingabe und Methodenausgabe und auf den Ports, die Sie in der REST-Webdienst-Verbraucher-Umwandlung konfigurieren.

Eingabeports stellen verschiedene Teile der Anfragemeldung dar. Sie können Eingabeports hinzufügen, die die Ressource identifizieren, die Sie abrufen oder ändern möchten. Sie können auch Eingabeports hinzufügen, die HTTP-Header, Cookie-Informationen und Elemente in der Anfragemeldung darstellen.

Ausgabeports stellen Elemente in der Antwortmeldung dar, die Sie an nachgelagerte Umwandlungen oder an das Ziel in einem Mapping senden möchten. Sie können Eingabeports hinzufügen, die HTTP-Header, Cookie-Informationen, den Antwortcode und Elemente in der Antwortmeldung darstellen.

Ressourcenkennung

Um eine Ressource in einer HTTP-Anfrage zu identifizieren, hängt der Datenintegrationsdienst Werte in bestimmten Eingabeports an die Basis-URL an. Sie definieren die URL in der HTTP-Verbindung oder in den Umwandlungseigenschaften. Verwenden Sie URL- oder Argumentports, um eine bestimmte Ressource zu identifizieren.

Verwenden Sie die URL-Ports, wenn der Webdienst eine Ressource mit einer eindeutigen Zeichenfolge identifiziert.

Beispiel: Der HypoStores REST-Webdienst identifiziert Teile nach Teilenummer über die folgende URL:

```
http://www.HypoStores.com/products/ProductDetails/<Part_No>
```

Um ein Teil zu identifizieren, definieren Sei die folgenden Umwandlungsdetails:

1. Legen Sie die Basis-URL für die folgende URL fest:

   ```
   http://www.HypoStores.com/products/ProductDetails
   ```

2. Definieren Sie einen URL-Port und übergeben Sie die Teilenummer an die Umwandlung über den URL-Port.

Wenn das Mapping Teilenummer 500 an den URL-Port übergibt, verwendet der Datenintegrationsdienst die folgende URL in der Anfragemeldung:

```
http://www.HypoStores.com/products/ProductDetails/500
```

Verwenden Sie Argumentports, wenn der Webdienst den Speicherort einer Ressource über Argumente identifiziert.

Beispiel: Sie möchten eine Teilenummer an den HypoStores REST-Webdienst über das Argument „Part_No“ übergeben.

Um ein Teil zu identifizieren, definieren Sei die folgenden Umwandlungsdetails:

1. Legen Sie die Basis-URL für die folgende URL fest:

   ```
   http://www.HypoStores.com/products/ProductDetails
   ```

2. Erstellen Sie einen Argumentport mit dem Argumentnamen „Part_No“ und übergeben Sie die Teilenummer an die Umwandlung über den Argumentport.

Wenn das Mapping Teilenummer 600 an den Argumentport übergibt, verwendet der Datenintegrationsdienst die folgende URL in der Anfragemeldung:

```
http://www.HypoStores.com/products/ProductDetails?Part_No=600
```

Erstellen Sie mehrere Argumentports, um mehrere Argumente zu definieren. Der Datenintegrationsdienst trennt jedes Argument mit einem kaufmännischen Und-Zeichen (&).

Beispiel: Sie möchten Mitarbeiterinformationen von einem REST-Webdienst abrufen und den Vornamen und Nachnamen des Mitarbeiters über die Argumente „First_Name“ und „Last_Name“ übergeben. Erstellen Sie die Argumentports mit den Argumentnamen „First_Name“ und „Last_Name“. Wenn das Mapping den Namen „John Smith“ an die Umwandlung übergibt, verwendet der Datenintegrationsdienst eine URL wie die folgende in der Anfragemeldung:

```
http://www.HypoStores.com/employees/EmpDetails?First_Name=John&Last_Name=Smith
```

Wenn Sie keine URL oder keinen Argumentport angeben, verwendet der Datenintegrationsdienst die Basis-URL aus den Umwandlungseigenschaften oder die HTTP-Verbindung zum Identifizieren der Ressource. Die Basis-URL in der HTTP-Verbindung überschreibt die Basis-URL in der Umwandlung.

HTTP-Methoden

Wählen Sie beim Erstellen einer Rest-Webdienst-Verbraucher-Umwandlung die HTTP-Methode aus, die der Datenintegrationsdienst in der Anfragemeldung verwendet. Sie können die HTTP-Methode nach dem Erstellen der Umwandlung nicht mehr ändern.

Sie konfigurieren die Umwandlung zur Verwendung einer der folgenden HTTP-Methoden:

Get

Ruft eine Ressource oder eine Sammlung von Ressourcen aus dem Webdienst ab. Beispiel: Sie können eine Produkttabelle oder Informationen zu einem Produkt abrufen.

Post

Sendet Daten an einen Webdienst. Verwenden Sie die Post-Methode, um eine Ressource oder eine Sammlung von Ressourcen zu erstellen. Beispiel: Sie können die Details einer neuen Speichertransaktion hinzufügen.

Put

Ersetzt eine Ressource oder eine Sammlung von Ressourcen. Wenn die Daten nicht vorhanden sind, werden die Daten von der Put-Methode eingefügt. Beispiel: Sie können die Lieferadresse eines Kunden aktualisieren.

Delete

Löscht eine Ressource oder eine Sammlung von Ressourcen. Beispiel: Sie können den Datensatz eines Mitarbeiters löschen, der nicht mehr für eine Organisation arbeitet.

HTTP-Get-Methode

Der Datenintegrationsdienst verwendet die HTTP-Get-Methode zum Abrufen von Daten aus einem REST-Webdienst. Verwenden Sie die Get-Methode zum Abrufen einer Ressource oder einer Sammlung von Ressourcen.

Wenn Sie die REST-Webdienst-Verbraucher-Umwandlung zur Verwendung der Get-Methode konfigurieren, konfigurieren Sie die Eingabeports, die Methodenausgabe und die Ausgabeports. Sie konfigurieren keine Methodeneingabe.

Beispiel

Sie möchten die Beschreibung und den Preis für die Teilenummer 500 in der HypoStores-Produktdatenbank abrufen. Der Webdienst verwendet die folgende URL, um ein Teil zu identifizieren:

```
http://www.HypoStores.com/products/ProductDetails?Part_No=<Part_No>
```

Geben Sie die folgende Basis-URL ein:

```
http://www.HypoStores.com/products/ProductDetails
```

In der folgenden Tabelle wird der Eingabeport angezeigt, den Sie möglicherweise definieren:

Porttyp	Argumentname	Eingabewert
Argument	Part_No	500

In der folgenden Tabelle werden die Ausgabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Rückgabewert
Ausgabe	Part_Desc	…<desc>ACME-Kugelschreiber, 12-pk, schwarz, 0,7 mm</desc>…
Ausgabe	Price_USD	…<price>9,89</price>…

HTTP-Post-Methode

Der Datenintegrationsdienst verwendet die HTTP-Post-Methode zum Senden von Daten an einen REST-Webdienst. Der Webdienst bestimmt die aktuelle Funktion, die die POST-Methode ausführt. Verwenden Sie die Post-Methode zum Erstellen einer Ressource oder einer Sammlung von Ressourcen.

Wenn Sie die REST-Webdienst-Verbraucher-Umwandlung zur Verwendung der Post-Methode konfigurieren, konfigurieren Sie die Eingabeports, die Methodeneingabe, die Methodenausgabe und die Ausgabeports.

Beispiel

Sie möchten das neue Teil 501 an die HypoStores-Produktdatenbank übermitteln. Der Webdienst verwendet die folgende URL für Teil 501:

```
http://www.HypoStores.com/products/ProductDetails/501
```

Geben Sie die folgende Basis-URL ein:

```
http://www.HypoStores.com/products/ProductDetails
```

In der folgenden Tabelle werden die Eingabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Eingabewert
URL	URL_Part_No	501
Eingabe	Part_Desc	ACME-Kugelschreiber, 12-pk, schwarz, 0,5 mm
Eingabe	Price_USD	9.89

In der folgenden Tabelle werden die Ausgabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Rückgabewert
Ausgabe	Antwort	<Vom Webdienst zurückgegebene Antwort>

HTTP-Put-Methode

Der Datenintegrationsdienst verwendet die HTTP-Put-Methode zum Aktualisieren von Daten über einen REST-Webdienst. Verwenden Sie die Post-Methode, um eine Ressource oder eine Sammlung von Ressourcen zu

aktualisieren. Wenn die Daten nicht vorhanden sind, erstellt der Datenintegrationsdienst die Ressource oder die Sammlung von Ressourcen.

Wenn Sie die REST-Webdienst-Verbraucher-Umwandlung zur Verwendung der Put-Methode konfigurieren, konfigurieren Sie die Eingabeports, die Methodeneingabe, die Methodenausgabe und die Ausgabeports.

Beispiel

Sie möchten den Stückpreis für Teil 501 in der HypoStores-Produktdatenbank aktualisieren. Der Webdienst verwendet die folgende URL für Teil 501:

```
http://www.HypoStores.com/products/ProductDetails/501
```

Geben Sie die folgende Basis-URL ein:

```
http://www.HypoStores.com/products/ProductDetails
```

In der folgenden Tabelle werden die Eingabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Eingabewert
URL	URL_Part_No	501
Eingabe	Price_USD	9.99

In der folgenden Tabelle werden die Ausgabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Rückgabewert
Ausgabe	Antwort	<Vom Webdienst zurückgegebene Antwort>

HTTP-Delete-Methode

Der Datenintegrationsdienst verwendet die HTTP-Delete-Methode zum Entfernen von Daten über einen REST-Webdienst. Verwenden Sie die Delete-Methode, um eine Ressource oder eine Sammlung von Ressourcen zu entfernen.

Wenn Sie die REST-Webdienst-Verbraucher-Umwandlung zur Verwendung der Delete-Methode konfigurieren, konfigurieren Sie die Eingabeports, die Methodeneingabe, die Methoden-Ausgabe und die Ausgabeports.

Beispiel

Sie möchten die Teilenummer 502 aus der HypoStores-Produktdatenbank löschen. Der Webdienst verwendet die folgende URL, um ein Teil zu identifizieren:

```
http://www.HypoStores.com/products/ProductDetails?Part_No=<Part_No>
```

Geben Sie die folgende Basis-URL ein:

```
http://www.HypoStores.com/products/ProductDetails
```

In der folgenden Tabelle wird der Eingabeport angezeigt, den Sie möglicherweise definieren:

Porttyp	Argumentname	Eingabewert
Argument	Part_No	502

In der folgenden Tabelle werden Ausgabeports angezeigt, die Sie möglicherweise definieren:

Porttyp	Portname	Rückgabewert
Ausgabe	Antwort	<Vom Webdienst zurückgegebene Antwort>

REST-Webdienst-Verbraucher-Umwandlung - Ports

Eine REST-Webdienst-Verbraucher-Umwandlung kann mehrere Eingabeports und mehrere Ausgabeports enthalten. Sie erstellen Ports in Gruppen basierend auf der Struktur der XML- oder JSON-Hierarchie.

Wenn Sie die Umwandlungsports sehen, blenden Sie die Ports ein, sofern die XML- oder JSON-Hierarchie nicht angezeigt werden muss. Wenn die Ports angezeigt werden, können Sie Gruppen und Ports definieren und den Eingabe- und Ausgabeports Elemente aus der Methodeneingabe und -ausgabe zuordnen.

Eine REST-Webdienst-Verbraucher-Umwandlung kann mehrere Eingabe- und mehrere Ausgabegruppen enthalten. Erstellen Sie beim Anlegen von Ports Gruppen und fügen Sie den Gruppen die Ports hinzu. Definieren Sie die Ports in einer Gruppenhierarchie basierend auf der Struktur der Eingabe- oder Ausgabe-Hierarchie in der XML- oder JSON-Hierarchie. Fügen Sie einen Schlüssel hinzu, um eine untergeordnete Gruppe mit einer übergeordneten Gruppe zu verbinden.

Alle Gruppen mit Ausnahme der untersten Gruppe in der Hierarchie müssen Primärschlüssel aufweisen. Alle Gruppen in der Hierarchie mit Ausnahme der Root-Gruppe müssen Fremdschlüssel aufweisen.

Die Umwandlung verfügt über eine Root-Eingabegruppe mit der Bezeichnung RequestInput. Sie müssen der Root-Eingabegruppe einen Primärschlüssel hinzufügen. Bei dem Schlüssel muss es sich um einen String, Bigint- oder Integer-Datentyp handeln. Sie können einen Port in der Root-Eingabegruppe als Pass-Through-Port konfigurieren.

Um einem Port ein Element zuzuweisen, klicken Sie auf das Feld in der Spalte **Speicherort** und erweitern Sie die Hierarchie im Dialogfeld **Speicherort auswählen**. Wählen Sie anschließend ein Element aus der Hierarchie aus.

Eingabeports

Eingabeports stellen Daten aus einer vorgelagerten Umwandlung oder einer Quelle dar, die Sie an den Webdienst übergeben möchten. Sie können mehrere Eingabeports konfigurieren. Jeder Eingabeport verweist auf ein Element in der Anfragemeldung.

Um einen Eingabeport hinzuzufügen, wählen Sie eine Eingabegruppe aus, klicken Sie auf den Pfeil neben der Schaltfläche **Neu** und wählen Sie **Feld** aus.

Ausgabeports

Ausgabeports stellen Elemente in der Antwortmeldung dar, die Sie an eine nachgelagerte Umwandlung oder an das Ziel übergeben möchten. Sie können mehrere Ausgabeports konfigurieren. Jeder Ausgabeport verweist auf ein Element in der Antwortmeldung.

Um einen Ausgabeport hinzuzufügen, wählen Sie eine Ausgabegruppe aus, klicken Sie auf den Pfeil neben der Schaltfläche **Neu** und wählen Sie **Feld** aus.

Pass-Through-Ports

Pass-Through-Ports übergeben Daten über die Umwandlung, ohne die Daten zu ändern. Sie können einen Port in der Root-Eingabegruppe als Pass-Through-Port konfigurieren.

Um einen Pass-Through-Port hinzuzufügen, fügen Sie der Root-Eingabegruppe einen Port hinzu. Klicken Sie dann mit der rechten Maustaste auf den Port und wählen Sie **Zuordnen** aus.

Argumentports

Mit Argumentports können Sie eine Ressource identifizieren, wenn die URL für die Ressource ein Argument annimmt. Fügen Sie der Root-Eingabegruppe Argumentports hinzu.

Ein Argumentport enthält einen Portnamen und einen Argumentnamen. Wenn ein Argumentname ein Zeichen enthält, das in einem Portnamen nicht zulässig ist, geben Sie einen Argumentnamen ein, der sich vom Portnamen unterscheidet. Beispiel: Sie möchten das Argument „Cust-ID“ an den Webdienst übergeben, der Datenintegrationsdienst lässt aber keine Gedankenstriche (-) in Portnamen zu. Geben Sie „Cust-ID“ als Argumentname ein, aber geben Sie „CustID“ als Portname ein.

Der Datenintegrationsdienst hängt die Argumentnamen und -werte für jeden Argumentport an die Basis-URL als „Name=Value“-Paar an. Sie können mehrere Argumentports konfigurieren. Der Datenintegrationsdienst trennt mehrere Argumente in der Anfrage mit einem kaufmännischen Und-Zeichen (&).

Beispiel:

```
http://www.HypoStores.com/customers/CustDetails?Last_Name=Jones&First_Name=Mary
```

Wenn Sie Argumentports und URL-Ports in der Umwandlung definieren, hängt der Datenintegrationsdienst die URL-Portwerte an die Basis-URL gefolgt von den Argumentnamen und -werten an.

Um einen Argumentport hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **Argumentports** aus. Geben Sie den Argumentnamen und den Portnamen ein.

URL-Ports

Mit URL-Ports können Sie eine Ressource über eine statische URL identifizieren. Zur Identifizierung einer Ressource hängt der Datenintegrationsdienst den Wert des URL-Ports an die Basis-URL an.

Beispiel:

```
http://www.HypoStores.com/products/ProductDetails/<URL_port_value>
```

Fügen Sie der Root-Eingabegruppe URL-Ports hinzu.

Sie können mehrere URL-Ports konfigurieren. Der Datenintegrationsdienst trennt die Werte in jedem URL-Port mit Schrägstrichen (/). Wenn Sie URL-Ports und Argumentports in der Umwandlung definieren, hängt der Datenintegrationsdienst die URL-Portwerte an die Basis-URL gefolgt von den Argumentnamen und -werten an.

Um einen URL-Port hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **URL-Ports** aus.

HTTP-Header-Ports

HTTP-Header-Ports stellen HTTP-Header in der Anfrage- oder Antwortmeldung dar. Sie können mehrere HTTP-Header-Ports konfigurieren.

Um Header-Informationen an den Webdienst in der Anfrage zu übergeben, fügen Sie den Port zur Root-Eingabegruppe hinzu. Sie können einen HTTP-Header-Port für die Root-Eingabegruppe konfigurieren. Wenn

Sie einen HTTP-Header zu einer Root-Eingabegruppe hinzufügen, können Sie ihn als Pass-Through-Port konfigurieren.

Um Header-Informationen aus der Antwort zu extrahieren, fügen Sie den Port zu einer Ausgabegruppe hinzu. Sie können einen HTTP-Header-Port für jede Ausgabegruppe konfigurieren.

Ein HTTP-Header-Port enthält einen Portnamen und einen HTTP-Header-Namen. Wenn ein HTTP-Header-Name ein Zeichen enthält, das in einem Portnamen nicht zulässig ist, geben Sie einen HTTP-Header-Namen ein, der sich vom Portnamen unterscheidet. Beispiel: Sie möchten den Header-Namen „Content-Type“ an den Webdienst übergeben, der Datenintegrationsdienst lässt aber keine Gedankenstriche (-) in Portnamen zu. Geben Sie „Content-Type“ als HTTP-Header-Name ein, aber geben Sie „ContentType“ als Portname ein.

Um einen HTTP-Header-Port hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **HTTP-Header** aus. Geben Sie einen Kopfzeilen- und Portnamen ein.

Cookie-Ports

Sie können die REST-Webdienst-Verbraucher-Umwandlung zur Verwendung der Cookie-Authentifizierung konfigurieren. Der Remote-Webserver verfolgt die Webdienst-Verbraucher-Benutzer anhand der Cookies. Sie können die Leistung erhöhen, wenn ein Mapping einen Webdienst mehrmals aufruft.

Um Cookie-Informationen an den Webdienst in der Anfrage zu übergeben, fügen Sie den Port der Root-Eingabegruppe hinzu. Sie können einen Cookie-Port für die Root-Eingabegruppe konfigurieren. Wenn Sie der Root-Eingabegruppe einen Cookie-Port hinzufügen, können Sie ihn als Pass-Through-Port konfigurieren.

Um Cookie-Informationen aus der Antwort zu extrahieren, fügen Sie den Cookie-Port einer Ausgabegruppe hinzu. Sie können einen Cookie-Port für jede Ausgabegruppe konfigurieren.

Wenn Sie den Cookie-Port für eine Webdienst-Anfragemeldung projektieren, gibt der Webdienstanbieter einen Cookie-Wert in der Antwortmeldung zurück. Sie können den Cookie-Wert an eine andere im Mapping nachgelagerte Umwandlung übergeben oder den Cookie-Wert in einer Datei speichern. Beim Speichern des Cookie-Werts in einer Datei können Sie den Cookie als Eingabe in die REST-Webdienst-Verbraucher-Umwandlung konfigurieren.

Um einen Cookie-Port hinzuzufügen, klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **Andere Ports** aus. Wählen Sie anschließend **Cookie** aus und klicken Sie auf **OK**.

Ausgabe-XML-Ports

Ausgabe-XML-Ports stellen Antworten vom Webdienst dar. Ausgabe-XML Ports sind Zeichenfolgen-Ports.

Fügen Sie eine XML-Ausgabe einer Ausgabegruppe hinzu. Sie können einen XML-Ausgabeport für jede Ausgabegruppe konfigurieren.

Klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **Andere Ports** aus. Wählen Sie anschließend **Ausgabe-XML** aus und klicken Sie auf **OK**.

Antwortcode-Ports

Antwortcode-Ports stellen die HTTP-Antwortcodes aus dem Webdienst dar. Antwortcode-Ports sind Ganzzahl-Ports.

Fügen Sie einen Antwortcode einer Ausgabegruppe hinzu. Sie können einen Antwort-Port für jede Ausgabegruppe konfigurieren.

Um einen Antwortcode-Port hinzuzufügen, wählen Sie eine Ausgabegruppe aus, klicken Sie mit der rechten Maustaste auf die Root-Eingabegruppe und wählen Sie **Neu** > **Andere Ports** aus. Wählen Sie anschließend **Antwortcode** aus und klicken Sie auf **OK**. .

REST-Webdienst-Verbraucher-Umwandlung - Eingabe-Mapping

Wenn Sie die Umwandlungsports sehen, blenden Sie das Eingabe-Mapping ein, um die Methodeneingabe-Hierarchie anzuzeigen. Wenn Sie das Eingabe-Mapping anzeigen, können Sie Eingabegruppen und Eingabeports definieren und die Eingabeports zu Methodeneingabe-Elementen zuordnen.

Das Eingabe-Mapping enthält die folgenden Bereiche:

Ports

Erstellen Sie die Umwandlungseingabegruppen und -eingabeports im Bereich **Ports**.

Methodeneingabe

Im Bereich **Methodeneingabe** werden die Elemente in der Anfragemeldung angezeigt, die von der Webdienst-Verbraucher-Umwandlung an den Webdienst gesendet werden. Wenn Sie ein Schema-Objekt zum Erstellen der Umwandlung verwenden, definiert das Schema-Objekt die Methodeneingabe-Hierarchie.

Ordnen Sie nach dem Erstellen der Eingabeports die Eingabeports aus dem Bereich **Ports** den Elementen im Bereich **Methodeneingabe** zu. Wenn Sie einem Element in der Methodeneingabe einen Eingabeport zuordnen, wird der Speicherort des Ports in der Spalte „Speicherort“ im Bereich **Methodeneingabe** angezeigt.

Das Developer-Tool ordnet Eingabeports Elemente auf der ersten Ebene der Methodeneingabe zu, wenn die erste Ebene der Eingabe-Hierarchie zugeordnet werden soll. Das Developer-Tool erstellt auch die Ports zum Durchführen des Mappings. Wenn die erste Ebene der Hierarchie ein mehrfach auftretendes übergeordnetes Element mit einem oder mehreren mehrfach auftretenden untergeordneten Elementen enthält, ordnet das Developer-Tool die erste Ebene der Hierarchie nicht zu.

Sie können die Zeilen anzeigen, die die Eingabeports mit den Elementen in der Methodeneingabe verbinden.

Regeln und Richtlinien zum Zuordnen von Eingabeports zu Elementen

Überprüfen Sie die folgenden Regeln, wenn Sie Eingabeports zu Elementen in der Methodeneingabe-Hierarchie zuordnen:

- Sie können einen Eingabeport einem Element in der Hierarchie zuordnen. Sie können den gleichen Port einer beliebigen Anzahl von Schlüsseln in der Hierarchie zuordnen.
- Der Eingabeport und das Element müssen kompatible Datentypen aufweisen.
- Sie können Ports aus einer Eingabegruppe zu mehreren Hierarchie-Ebenen in der Methodeneingabe zuordnen.
- Sie müssen Eingabeports zu den Schlüsseln in der Methodeneingabe zuordnen. Alle Ports, die einem Schlüssel zugeordnet werden, müssen den String-, Integer- oder Bigint-Datentyp aufweisen. Ordnen Sie Daten den Schlüsseln auf allen Ebenen der Methodeneingabe über der Hierarchie-Ebene zu, die Sie in der Anfragenachricht einschließen. Schließen Sie die Fremdschlüssel für alle Ebenen ein, die sich über der zuzuordnenden Ebene befinden, einschließlich dieser Ebene selbst.

  Hinweis: Sie müssen Schlüsseln keine Eingabeports zuordnen, wenn Sie nur die niedrigste Ebene der Methodeneingabe-Hierarchie zuordnen.
- Sie müssen das RequestInput-Root-Element zum untergeordneten Element der Rest_Consumer_input-Gruppe für die Methodeneingabedefinition zuordnen.
- Sie können mehrere String-, Bigint- oder Integer-Eingabeports einem Schlüssel im Bereich **Methodeneingabe** zuordnen, um einen zusammengesetzten Schlüssel zu erstellen. Wenn Sie bei einem

zusammengesetzten Schlüssel auf das Feld **Speicherort** klicken, können Sie die Eingabeports neu anordnen oder einen der Ports entfernen.

- Wenn der Webdienst ein JSON-Dokument erzeugt, stellen Sie sicher, dass xmlRoot der erste Knoten in der Antworthierarchie ist. Wenn xmlRoot nicht der erste Knoten für einen Webdienst mit JSON-Antwort ist, werden unter Umständen Nullwerte angezeigt.

Zuordnen von Eingabeports zur Methodeneingabe

Wenn Sie das Umwandlungseingabe-Mapping anzeigen, können Sie Eingabegruppen definieren, Eingabeports definieren und Eingabeports zu Operationseingabeknoten zuordnen.

1. Öffnen Sie eine REST-Webdienst-Verbraucher-Umwandlung.
2. Zeigen Sie in der Ansicht **Ports** das Eingabe-Mapping an.
3. Definieren Sie einen Primärschlüssel für die Root-Eingabegruppe.
4. Verwenden Sie zum Hinzufügen einer Eingabegruppe oder eines Eingabeports zum Bereich **Ports** eine der folgenden Methoden:

Methode	Beschreibung
Ziehen Sie ein Element.	Ziehen Sie eine Gruppe oder ein untergeordnetes Element aus dem Bereich **Methodeneingabe** in eine leere Spalte im Bereich **Ports**. Wenn Sie eine Gruppe in den Bereich **Ports** ziehen, fügt das Developer-Tool eine Gruppe ohne Ports hinzu.
Fügen Sie eine Gruppe oder einen Port manuell hinzu.	Klicken Sie zum Hinzufügen einer Gruppe auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Gruppe** Klicken Sie zum Hinzufügen eines Ports auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Feld**.
Ziehen Sie einen Port aus einer anderen Umwandlung.	Ziehen Sie im Editor einen Port aus einer anderen Umwandlung in die REST-Webdienst-Verbraucher-Umwandlung.
Kopieren Sie einen Port.	Wählen Sie Ports einer anderen Umwandlung aus und kopieren Sie sie in den Bereich **Ports**. Zum Kopieren von Ports können Sie Tastenkombinationen oder die Schaltflächen **Kopieren** und **Einfügen** im Developer-Tool verwenden.
Wählen Sie **Zuordnen der ersten Hierarchieebene** aus.	Das Developer-Tool weist Elemente auf der ersten Ebene der Methodeneingabe zu Eingabeports und Gruppen zu. Das Developer-Tool erstellt auch die Eingabeports und Eingabegruppen, um das Mapping durchzuführen.

5. Wenn Sie einen Port manuell erstellen oder aus einer anderen Umwandlung kopieren, klicken Sie auf die Spalte **Speicherort** im Bereich **Methodeneingabe** und wählen Sie einen Port aus der Liste aus.
6. Um Eingabeports als zusammengesetzten Schlüssel zuzuordnen, verwenden Sie eine der folgenden Methoden:

Methode	Beschreibung
Ziehen Sie Eingabeports.	Wählen Sie zwei oder mehr Eingabeports und ziehen Sie sie zu einem Schlüssel in der Methodeneingabe-Hierarchie.
Wählen Sie Eingabeports aus dem Dialogfeld **Speicherort auswählen** aus.	Klicken Sie auf die Spalte **Speicherort** eines Schlüssels in der Methodeneingabe-Hierarchie und wählen Sie anschließend die Eingabeports aus.

7. Um die Elementspeicherorte zu löschen, verwenden Sie eine der folgenden Methoden:

Methode	Beschreibung
Klicken Sie auf **Löschen**.	Wählen Sie im Bereich **Methodeneingabe** mindestens ein Element aus und klicken Sie auf **Löschen**.
Löschen Sie die Zeilen, die Ports mit Elementen verbinden.	Wählen Sie mindestens eine Zeile aus, die die Eingabeports mit den Elementen in der Methodeneingabe verbindet, und klicken Sie auf **Löschen**.

REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping

Wenn Sie die Umwandlungsports sehen, blenden Sie das Ausgabe-Mapping ein, um die Methoden-Ausgabe-Hierarchie anzuzeigen. Wenn Sie das Ausgabe-Mapping anzeigen, können Sie Ausgabegruppen und Ausgabeports definieren und den Ausgabeports Methoden-Ausgaben-Elemente zuordnen.

Das Ausgabe-Mapping enthält die folgenden Bereiche:

Methodenausgabe

Der Bereich **Methoden-Ausgabe** zeigt die Knoten in der Antwortmeldung an, die vom Webdienst an die REST-Webdienst-Verbraucher-Umwandlung zurückgegeben werden. Wenn Sie ein Schema-Objekt zum Erstellen der Umwandlung verwenden, definiert das Schema-Objekt die Methoden-Ausgabe-Hierarchie.

Ports

Erstellen Sie die Umwandlungsausgabegruppen und -ports im Bereich **Ports**.

Ordnen Sie nach dem Erstellen der Ausgabeports die Elemente aus dem Bereich **Methoden-Ausgabe** den Ports im Bereich **Ports** zu. Wenn Sie einem Ausgabeport ein Element aus der Methoden-Ausgabe zuordnen, wird der Speicherort des Elements in der Spalte **Speicherort** im Bereich **Ports** angezeigt.

Das Developer-Tool ordnet Elemente auf der ersten Ebene der Methoden-Ausgabe zu, wenn die erste Ebene der Ausgabe-Hierarchie zugeordnet werden soll. Das Developer-Tool erstellt auch die Ports zum Durchführen des Mappings. Wenn die erste Ebene der Hierarchie ein mehrfach auftretendes übergeordnetes Element mit einem oder mehreren mehrfach auftretenden untergeordneten Elementen enthält, ordnet das Developer-Tool die erste Ebene der Hierarchie nicht zu.

Sie können die Ausgabeports in einer Hierarchie anzeigen. Jede untergeordnete Gruppe wird unter der übergeordneten Gruppe angezeigt. Sie können die Zeilen anzeigen, die die Elemente in der Methoden-Ausgabe mit den Ausgabeports verbinden.

Wenn das zugehörige Schema-Objekt aus dem Repository gelöscht wird, behält das Developer-Tool den Speicherort der Methoden-Elemente im Ausgabe-Mapping bei. Wenn Sie das Ausgabe-Mapping anzeigen, wird im Bereich **Ports** weiterhin der Speicherort der Methoden-Elemente in der Spalte **Speicherort** für die Ausgabeports angezeigt. Wenn Sie ein anderes Schema mit der Umwandlung verknüpfen, überprüft das Developer-Tool die Gültigkeit aller Speicherorte. Das Developer-Tool löscht den Speicherort der Methoden-Elemente im Bereich **Ports** des Ausgabe-Mappings, wenn der Speicherort nicht mehr gültig ist.

Regeln und Richtlinien zum Zuordnen von Elementen zu Ausgabeports

Überprüfen Sie die folgenden Regeln, wenn Sie Elemente in der Methoden-Ausgabe-Hierarchie zu Ausgabeports zuordnen:

- Das Methoden-Ausgabe-Element und der Ausgabeport müssen kompatible Datentypen aufweisen.
- Sie können ein Element nicht mehr als einem Ausgabeport in einer Gruppe zuordnen.
- Jeder Ausgabeport muss einen gültigen Speicherort aufweisen, es sei denn, es handelt sich um einen Pass-Through-Port.
- Wenn Sie ein mehrfach vorkommendes untergeordnetes Element in einen leeren Ausgabeport ziehen, müssen Sie die Gruppe mit anderen Ausgabegruppen verknüpfen. Wenn Sie eine Gruppe auswählen, erstellt das Developer-Tool Schlüssel, um die Gruppen zu verknüpfen.
- Wenn Sie ein mehrfach vorkommendes Element in eine Gruppe ziehen, das das übergeordnete Element enthält, können Sie die Anzahl der einzubeziehenden untergeordneten Elemente konfigurieren. Sie können die übergeordnete Gruppe mit der mehrfach vorkommenden untergeordneten Gruppe in der Umwandlungsausgabe ersetzen.
- Wenn der Webdienst ein JSON-Dokument erzeugt, stellen Sie sicher, dass xmlRoot der erste Knoten in der Antworthierarchie ist. Wenn xmlRoot nicht der erste Knoten für einen Webdienst mit JSON-Antwort ist, werden unter Umständen Nullwerte in den Ausgabeports angezeigt.

Anpassen von Anzeigeoptionen

Sie können die Methoden-Ausgabe-Hierarchie so ändern, dass Cookie-Ports, Pass-Through-Ports und Schlüssel im Bereich **Methoden-Ausgabe** angezeigt werden. Außerdem können Sie den Gruppierungsaufbau anzeigen, der die Reihenfolge von Elementen definiert.

Klicken Sie auf **Ansicht anpassen** im Bereich **Methoden-Ausgabe**. Aktivieren Sie eine der folgenden Optionen:

Sequenz, Auswahl und Alles

Zeigen Sie eine Zeile an, die angibt, ob als Elementdefinition „Sequenz“, „Auswahl“ oder „Alle“ ausgewählt ist.

Elemente in einer Sequenz-Gruppe müssen in der in der Hierarchie festgelegten Reihenfolge angeordnet sein.

Mindestens ein Element in einer Auswahlgruppe muss in der Antwortmeldung angezeigt werden.

Alle Elemente in einer Alle-Gruppe müssen in der Antwortmeldung enthalten sein.

Schlüssel

Zeigen Sie die Schlüssel im Bereich **Methoden-Ausgabe** an. Der Bereich **Methoden-Ausgabe** enthält Schlüssel für jede Gruppe. Sie können einen Schlüssel einem Ausgabeport im Bereich **Ports** hinzufügen.

Pass-Through-Ports

Der Bereich **Methoden-Ausgabe** zeigt die Pass-Through-Ports an. Pass-Through-Ports sind Ports, die Daten ohne Änderung an die Umwandlung übergeben. Sie können Pass-Through-Ports von der Methoden-Ausgabe auf beliebige Ausgabegruppen der REST-Webdienst-Verbraucher-Umwandlung projizieren. Ein Pass-Through-Port empfängt Daten einmalig, damit der Port sich in den Antwortmeldungen auf der Root-Ebene befindet.

Zuordnen der Methodenausgabe zu Ausgabeports

Wenn Sie das Ausgabe-Mapping der Umwandlung anzeigen, können Sie Ausgabegruppen und Ausgabeports definieren und den Ausgabeports Elemente der Methoden-Ausgabe zuordnen.

1. Öffnen Sie eine REST-Webdienst-Verbraucher-Umwandlung.
2. Zeigen Sie in der Ansicht **Ports** das Ausgabe-Mapping an.
3. Verwenden Sie zum Hinzufügen einer Ausgabegruppe oder eines Ausgabeports zum Bereich **Ports** eine der folgenden Methoden:

Methode	Beschreibung
Ziehen Sie ein Element.	Ziehen Sie eine Gruppe oder ein untergeordnetes Element aus dem Bereich **Methoden-Ausgabe** in eine leere Spalte im Bereich **Ports**. Wenn Sie eine Gruppe in den Bereich **Ports** ziehen, fügt das Developer-Tool eine Gruppe ohne Ports hinzu.
Fügen Sie eine Gruppe oder einen Port manuell hinzu.	Klicken Sie zum Hinzufügen einer Gruppe auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Gruppe**. Klicken Sie zum Hinzufügen eines Ports auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Feld**.
Ziehen Sie einen Port aus einer anderen Umwandlung.	Ziehen Sie im Editor einen Port aus einer anderen Umwandlung in die REST-Webdienst-Verbraucher-Umwandlung.
Kopieren Sie einen Port.	Wählen Sie Ports einer anderen Umwandlung aus und kopieren Sie sie in den Bereich **Ports**. Zum Kopieren von Ports können Sie Tastenkombinationen oder die Schaltflächen **Kopieren** und **Einfügen** im Developer-Tool verwenden.

4. Wenn Sie einen Port manuell erstellen oder aus einer anderen Umwandlung kopieren, klicken Sie auf die Spalte **Speicherort** im Bereich **Ports** und wählen Sie ein Element aus der Liste aus.
5. Um die Portspeicherorte zu löschen, verwenden Sie eine der folgenden Methoden:

Methode	Beschreibung
Klicken Sie auf **Löschen**.	Wählen Sie einen oder mehrere Ports im Bereich **Ports** aus und klicken Sie auf **Löschen**.
Löschen Sie die Zeilen, die Elemente mit Ports verbinden.	Wählen Sie eine oder mehrere Zeilen aus, die die Elemente in der Methoden-Ausgabe mit den Ausgabeports verbinden, und klicken Sie auf **Löschen**.

REST-Webdienst-Verbraucher-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, mit denen Sie festlegen können, wie der Datenintegrationsdienst Daten für die REST-Webdienst-Verbraucher-Umwandlung verarbeitet.

Konfigurieren Sie die folgenden Eigenschaften auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Verbindung

Identifiziert das HTTP-Verbindungsobjekt zum Herstellen einer Verbindung zum Webdienst. Erstellen und bearbeiten Sie die HTTP-Verbindung im Developer-Tool. Wenn Sie eine HTTP-Verbindung konfigurieren, konfigurieren Sie die Basis-URL, den vom Webdienst benötigten Sicherheitstyp sowie eine Verbindungstimeoutperiode.

Die REST-Webdienst-Verbraucher-Umwandlung stellt mit einer URL eine Verbindung zum Webdienst her. Sie können die URL in den Umwandlungseigenschaften oder in der HTTP-Verbindung definieren.

Konfigurieren Sie in den folgenden Fällen eine HTTP-Verbindung:

- Sie verwenden keinen URL-Eingabeport.
- Der Webdienst erfordert HTTP-Authentifizierung oder SSL-Zertifikate.
- Sie möchten die Standardverbindungs-Timeoutperiode ändern.

XML-Schemavalidierung

Validiert die Antwortmeldung zur Laufzeit. **Wählen Sie „Fehler bei ungültigem XML“** oder **Keine Validierung** aus.

Sortierte Eingabe

Ermöglicht dem Datenintegrationsdienst das Generieren der Ausgabe, ohne alle Eingabedaten zu verarbeiten. Aktivieren Sie sortierte Eingabe, wenn die Eingabedaten durch die Schlüssel in der XML-Eingabe-Hierarchie sortiert werden.

URL

Die Basis-URL für den REST-Webdienst. Die Basis-URL in der HTTP-Verbindung überschreibt diesen Wert.

Format

Das Format der Webdienst-Antwort. Wählen Sie je nach Webdienst-Antwort **XML** oder **JSON** aus.

REST-Webdienst-Verbraucher-Umwandlung - Erstellung

Sie können eine wiederverwendbare oder nicht wiederverwendbare REST-Webdienst-Verbraucher-Umwandlung erstellen. Wiederverwendbare Umwandlungen können in mehreren Mappings vorhanden sein. Nicht wiederverwendbare Umwandlungen sind in einem einzelnen Mapping vorhanden.

Beim Erstellen einer REST-Webdienst-Verbraucher-Umwandlung können Sie die Elemente und die XML-Hierarchie manuell definieren oder Sie können die Elemente und die Hierarchie aus einem Schema-Objekt importieren. Das Schema-Objekt kann eine XML-Datei oder eine Textdatei sein.

Erstellen einer REST-Webdienst-Verbraucher-Umwandlung

Wenn Sie eine REST-Webdienst-Verbraucher-Umwandlung erstellen, wählen Sie eine Methode aus und definieren Sie die Methodeneingabe und Methodenausgabe basierend auf der von Ihnen ausgewählten Methode.

1. Zum Erstellen einer Rest-Webdienst-Verbraucher-Umwandlung verwenden Sie eine der folgenden Methoden:

Methode	Beschreibung
Wiederverwendbar	Wählen Sie in der Ansicht „Objekt-Explorer“ ein Projekt oder einen Ordner aus. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**. Wählen Sie die REST-Webdienst-Verbraucher-Umwandlung aus und klicken Sie auf **Weiter**.
Nicht wiederverwendbar	Ziehen Sie in einem Mapping oder Mapplet eine REST-Webdienst-Verbraucher-Umwandlung aus der Umwandlungspalette in den Mapping- oder Mapplet-Editor.

2. Geben Sie den Umwandlungsnamen ein und wählen Sie den Speicherort und die HTTP-Methode aus.
3. Klicken Sie auf **Weiter**.
4. Zum Definieren der Methodeneingabe verwenden Sie eine der folgenden Methoden:

Methode	Beschreibung
Als leer erstellen	Definieren Sie die XML-Elemente und -Hierarchie manuell.
Aus einem Element in einem Schema-Objekt erstellen	Importieren Sie die XML-Elemente und -Hierarchie aus einem Schemaobjekt.

   Im Bereich **Definition der Methoden-Eingabe** werden die Eingabegruppen und Eingabeports der Umwandlung angezeigt. Im Bereich **Eingabe-Mapping** wird die Anfragenachrichtenhierarchie angezeigt.
5. Definieren Sie die Eingabegruppen sowie Eingabeports und ordnen Sie die Eingabeports den Eingabeelementen zu.
6. Klicken Sie auf **Weiter**.
7. Wählen Sie zum Definieren der Methodenausgabe **Als leer erstellen** bzw. **Aus einem Element in einem Schema-Objekt erstellen** aus.

   Im Bereich **Definition der Methoden-Ausgabe** werden die Ausgabegruppen und Ausgabeports der Umwandlung angezeigt. Im Bereich **Ausgabe-Mapping** wird die Anfragenachrichtenhierarchie angezeigt.
8. Definieren Sie die Ausgabegruppen sowie Ausgabeports und ordnen Sie die Elemente den Ausgabeports zu.
9. Klicken Sie auf **Fertig stellen**.

KAPITEL 36

Router-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Router-Umwandlung - Übersicht, 534
- Router-Umwandlungen in dynamischen Mappings, 535
- Arbeiten mit Gruppen, 536
- Arbeiten mit Ports, 540
- Verbinden von Router-Umwandlungen in einem Mapping, 540
- Router-Umwandlung - Erweiterte Eigenschaften, 541

Router-Umwandlung - Übersicht

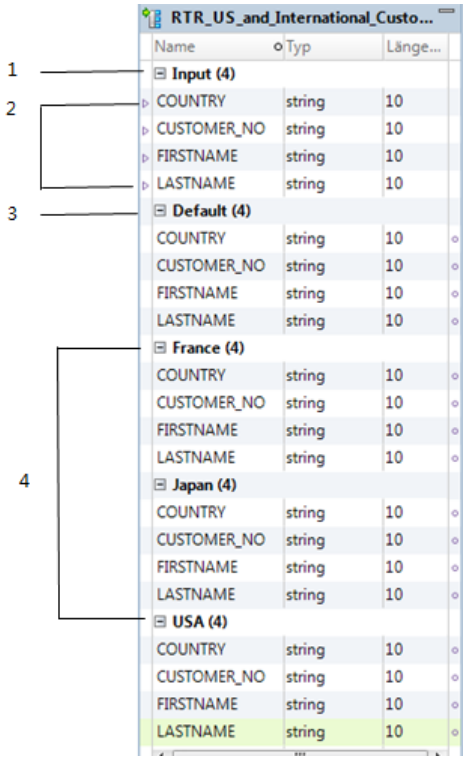
Die Router-Umwandlung ist eine aktive Umwandlung, die Daten in mehrere Umwandlungen basierend auf einer oder mehreren Gruppenbedingungen weiterleitet. Leiten Sie die Ausgabegruppen an verschiedene Umwandlungen oder verschiedene Ziele im Mapping weiter.

Eine Router-Umwandlung ähnelt einer Filterumwandlung, weil beide Umwandlungen eine Bedingung zum Testen der Daten verwenden. Eine Filterumwandlung testet Daten für eine Bedingung und löscht die Datenzeilen, die die Bedingung nicht erfüllen. Eine Router-Umwandlung testet Daten für eine oder mehrere Bedingungen und kann Datenzeilen umleiten, die die Bedingungen nicht erfüllen.

Wenn Sie dieselben Eingabedaten basierend auf mehreren Bedingungen testen müssen, verwenden Sie eine Router-Umwandlung in einem Mapping, anstatt mehrere Filterumwandlungen zur Durchführung derselben Aufgabe zu erstellen. Die Router-Umwandlung ist effizienter. Zum Testen von Daten basierend auf drei Bedingungen können Sie beispielsweise eine Router-Umwandlung anstelle von drei Filterumwandlungen verwenden. Wenn Sie eine Router-Umwandlung in einem Mapping verwenden, verarbeitet der Datenintegrationsdienst die eingehenden Daten ein Mal. Wenn Sie mehrere Filterumwandlungen in einem Mapping verwenden, verarbeitet der Datenintegrationsdienst die eingehenden Daten für jede Umwandlung.

Eine Router-Umwandlung besteht aus Eingabe- und Ausgabegruppen, die Eingabe- und Ausgabeports, Gruppenfilterbedingungen und erweiterten Eigenschaften, die Sie im Developer tool konfigurieren können.

Die folgende Abbildung zeigt eine Beispiel-Router-Umwandlung und ihre Komponenten:



1. Eingabegruppe
2. Eingabeports
3. Standardmäßige Ausgabegruppe
4. Benutzerdefinierte Ausgabegruppen

Router-Umwandlungen in dynamischen Mappings

Sie können eine Router-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports in der Gruppenfilterbedingung verweisen.

Wenn Sie einen dynamischen Port in der Gruppenfilterbedingung verwenden, enthält der dynamische Port möglicherweise mehr als einen generierten Port. Die Gruppenfilterbedingung wird so erweitert, dass jeder generierte Port eingeschlossen wird. Jeder generierte Port muss einen gültigen Typ für den Ausdruck aufweisen.

Sie können die Gruppenfilterbedingung parametrisieren. Verwenden Sie einen Ausdruckstypparameter, um den Filter anzugeben.

Arbeiten mit Gruppen

Eine Router-Umwandlung verfügt über die folgenden Gruppentypen:

- Eingabe
- Ausgabe

Eingabegruppe

Die Rang-Umwandlung enthält eine einzelne Eingabegruppe. Die Eingabegruppe enthält alle Eingabeports, die Sie der Umwandlung hinzufügen.

Ausgabegruppen

Die Rang-Umwandlung enthält die folgenden Ausgabegruppentypen:

Benutzerdefinierte Gruppen

Sie können eine benutzerdefinierte Gruppe erstellen, um eine Bedingung basierend auf eingehenden Daten zu testen. Eine benutzerdefinierte Gruppe besteht aus Ausgabeports und einer Gruppenfilterbedingung. Sie können benutzerdefinierte Gruppen auf der Registerkarte **Gruppen** mit dem Developer-Tool erstellen und bearbeiten. Erstellen Sie eine benutzerdefinierte Gruppe für jede Bedingung, die Sie festlegen möchten.

Der Data Integration Service verwendet die Bedingung, um jede Zeile der eingehenden Daten zu überprüfen. Vor der Verarbeitung der Standardgruppe testet er die Bedingungen jeder einzelnen benutzerdefinierten Gruppe. Der Data Integration Service bestimmt die Reihenfolge der Bewertung für jede Bedingung basierend auf der Reihenfolge der verbundenen Ausgabegruppen. Der Data Integration Service verarbeitet benutzerdefinierte Gruppen, die mit einer Umwandlung oder einem Target in einem Mapping verbunden sind.

Wenn eine Zeile mehr als eine Gruppenfilterbedingung erfüllt, übergibt der Data Integration Service diese Zeile mehrfach.

Die Standardgruppe

Das Developer Tool erstellt die Standardgruppe, nachdem Sie eine benutzerdefinierte Gruppe erstellt haben. Das Developer Tool lässt das Bearbeiten und Löschen der Standardgruppe nicht zu. Dieser Gruppe ist keine verknüpfte Gruppenfilterbedingung zugeordnet. Wenn alle Gruppenbedingungen als FALSE bewertet werden, übergibt der Data Integration Service die Zeile an die Standardgruppe. Wenn der Data Integration Service alle Zeilen in die Standardgruppe löschen soll, verbinden Sie ihn nicht mit einer Umwandlung oder einem Target in einem Mapping.

Das Developer Tool löscht die Standardgruppe, wenn Sie die letzte benutzerdefinierte Gruppe aus der Liste entfernen.

Das Developer Tool kopiert Eigenschaftsinformationen aus den Eingabeports der Eingabegruppe, um mehrere Ausgabeports für jede Ausgabegruppe zu erstellen. Sie können Ausgabeports oder deren Eigenschaften nicht ändern oder löschen.

Verwenden von Gruppenfilterbedingungen

Sie können Daten basierend auf einer oder mehreren Gruppenfilterbedingungen testen. Sie erstellen Gruppenfilterbedingungen auf der Registerkarte **Gruppen** mithilfe des Ausdruckseditors.

Sie können einen beliebigen Ausdruck eingeben, der einen einzelnen Wert zurückgibt. Sie können auch eine Konstante für die Bedingung eingeben. Eine Gruppenfilterbedingung gibt TRUE oder FALSE für jede Zeile zurück, die die Umwandlung basierend auf einer Zeile durchläuft, die die angegebene Bedingung erfüllt. Null (0) ist das Äquivalent von FALSE. Alle Nicht-Nullwerte sind das Äquivalent zu TRUE. Sie können einen einzelnen numerischen Port als eine Filterbedingung verwenden. Der Datenintegrationsdienst übergibt die Datenzeilen, die TRUE ergeben, an jede Umwandlung oder jedes Ziel, das mit allen benutzerdefinierten Gruppen verknüpft ist.

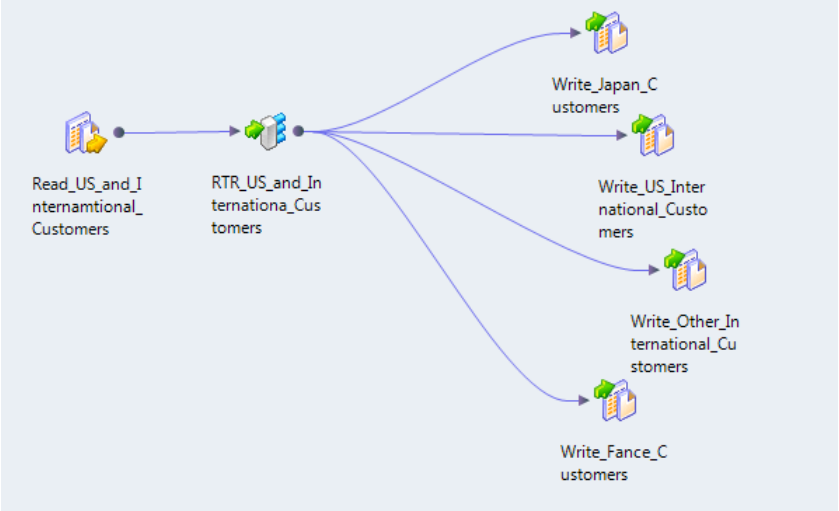
Hinweis: Sie können keinen einzelnen dynamischen Port zur Ausgabe eines booleschen Werts verwenden.

Sie haben beispielsweise Kunden aus neun Ländern und möchten verschiedene Berechnungen für die Daten aus drei Ländern durchführen. Sie können eine Router-Umwandlung in einem Mapping durchführen, um diese Daten in drei verschiedenen Ausdrucksumwandlungen zu filtern.

Sie können Parameter als Elemente in der Gruppenfilterbedingung verwenden. Sie können Systemparameter oder benutzerdefinierte Parameter verwenden. Sie können Parameter aus dem Ausdruckseditor erstellen und zu dem Ausdruck hinzufügen.

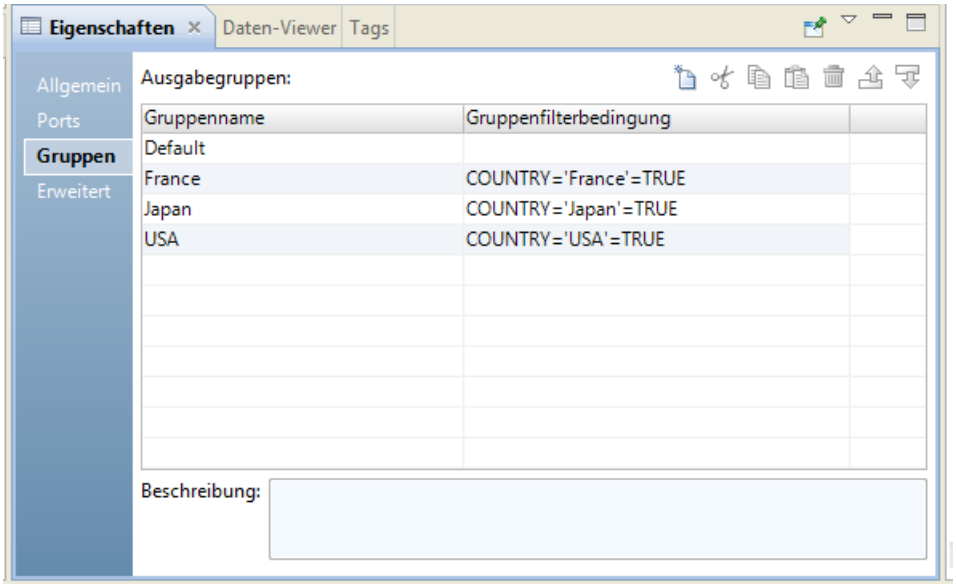
Die Standardgruppe verfügt nicht über eine Gruppenfilterbedingung. Sie können eine Ausdrucksumwandlung erstellen, um eine Berechnung basierend auf den Daten aus den anderen sechs Ländern durchzuführen.

Die folgende Abbildung zeigt ein Mapping mit einer Routerumwandlung, die Daten basierend auf mehreren Bedingungen filtert:



Erstellen Sie zum Durchführen mehrerer Berechnungen basierend auf den Daten aus drei verschiedenen Ländern drei benutzerdefinierte Gruppen und geben Sie drei Gruppenfilterbedingungen auf der Registerkarte **Gruppen** an.

Die folgende Abbildung zeigt Gruppenfilterbedingungen, mit denen Kundendaten gefiltert werden:



Die folgende Tabelle zeigt die Gruppenfilterbedingungen an, mit denen Kundendaten gefiltert werden:

Gruppenname	Gruppenfilterbedingung
Frankreich	customer_name='France'=TRUE
Japan	customer_name='Japan'=TRUE
USA	customer_name='USA'=TRUE

Im Mapping übergibt der Datenintegrationsdienst die Datenzeilen, die TRUE ergeben, an jede Umwandlung oder jedes Ziel, das mit jeder benutzerdefinierten Gruppe verknüpft ist, z. B. Japan, Frankreich und USA. Der Datenintegrationsdienst übergibt die Zeile an die Standardgruppe, wenn alle Bedingungen FALSE ergeben. Der Datenintegrationsdienst übergibt die Daten der anderen sechs Länder anschließend an die Umwandlung oder das Ziel, das mit der Standardgruppe verknüpft ist. Wenn der Datenintegrationsdienst alle Zeilen in die Standardgruppe löschen soll, verbinden Sie ihn nicht mit einer Umwandlung oder einem Ziel in einem Mapping.

Die Router-Umwandlung übergibt Daten über jede Gruppe, die die Bedingung erfüllt. Wenn die Daten drei Ausgabegruppenbedingungen erfüllen, übergibt die Router-Umwandlung die Daten über drei Ausgabegruppen.

Sie konfigurieren beispielsweise die folgenden Gruppenbedingungen in einer Router-Umwandlung:

Gruppenname	Gruppenfilterbedingung
Ausgabegruppe 1	employee_salary > 1000
Ausgabegruppe 2	employee_salary > 2000

Wenn die Router-Umwandlung eine Eingabezeile mit Gehalt_Mitarbeiter=3000 verarbeitet, werden die Daten über die Ausgabegruppen 1 und 2 geroutet.

Dynamische Ports in Gruppenfilterbedingungen

Sie können einen dynamischen Port in einer Gruppenfilterbedingung verwenden. Der Datenintegrationsdienst wendet die Filterbedingung auf jeden generierten Port in dem dynamischen Port an.

Beispiel: Ein dynamischer Port „MyDynamicPort“ enthält drei Dezimal-Ports:

```
Salary
Bonus
Stock
```

Sie können die folgende Gruppenfilterbedingung konfigurieren:

```
MyDynamicPort > 100
```

Die Gruppenfilterbedingung wird um den folgenden Ausdruck erweitert:

```
Salary > 100 AND Bonus > 100 AND Stock > 100
```

Jeder generierte Port muss einen gültigen Typ für den Ausdruck aufweisen.

Den Gruppenfilter parametrisieren

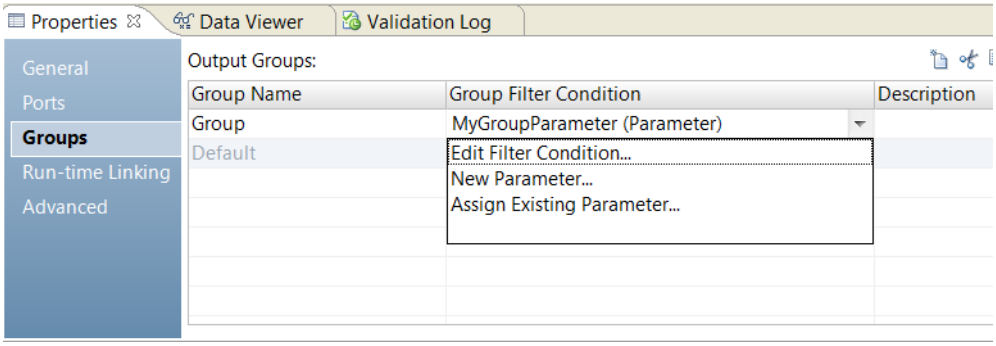
Sie können einen Ausdrucksparameter zum Definieren eines Gruppenfilters verwenden. Ein Ausdrucksparameter ist ein Parameter, der einen vollständigen Ausdruck enthält.

Beispiel: Ein Ausdrucksparameter kann den Ausdruck

`Employee_Salary > 1000` enthalten.

Um den Gruppenfilter zu parametrisieren, wählen Sie im Feld **Gruppenfilterbedingung** der Registerkarte **Gruppen** die Option **Neuer Parameter** aus. Definieren Sie den Parameter und geben Sie den Ausdruck im Ausdrucksseditor ein. Ein Ausdrucksparameter darf keine anderen Parameter enthalten.

Die folgende Abbildung zeigt die Registerkarte **Gruppen** in den Umwandlungseigenschaften:



Wenn Sie einen Ausdrucksparameter verwenden, kann das Developer Tool den Ausdruck nicht validieren. Wenn der Ausdruck zur Laufzeit nicht gültig ist, schlägt das Mapping möglicherweise fehl.

Hinzufügen von Gruppen

Beim Hinzufügen einer Gruppe kopiert das Developer Tool Eigenschaftsinformationen aus den Eingabeports in die Ausgabeports.

1. Klicken Sie auf die Registerkarte **Gruppen**.
2. Klicken Sie auf die Schaltfläche **Neu**.
3. Geben Sie im Abschnitt **Gruppenname** einen Namen für die Gruppe ein.

4. Klicken Sie auf das Feld **Gruppenfilterbedingung**, um den **Ausdruckseditor** zu öffnen.
5. Geben Sie die Gruppenfilterbedingung ein.
6. Klicken Sie auf **Überprüfen**, um die Syntax der Bedingung zu prüfen.
7. Klicken Sie auf **OK**.

Arbeiten mit Ports

Eine Router-Umwandlung weist Eingabeports und Ausgabeports auf. Eingabeports befinden sich in der Eingabegruppe und Ausgabeports in den Ausgabegruppen.

Zum Erstellen von Eingabeports können Sie diese aus einer anderen Umwandlung kopieren oder manuell auf der Registerkarte **Ports** anlegen.

Das Developer-Tool erstellt Ausgabeports, indem es die folgenden Eigenschaften aus den Eingabeports kopiert:

- Portname
- Datentyp
- Genauigkeit
- Größenordnung
- Standardwert

Wenn Sie die Eingabeports ändern, aktualisiert das Developer-Tool die Ausgabeports entsprechend. Sie können Ausgabeports nicht bearbeiten oder löschen.

Das Developer-Tool erstellt Namen für Ausgabeports basierend auf den Namen der Eingabeports. Für jeden Eingabeport erstellt das Developer-Tool einen entsprechenden Ausgabeport in jeder Ausgabegruppe.

Verbinden von Router-Umwandlungen in einem Mapping

Beim Verbinden von Umwandlungen mit einer Router-Umwandlung in einem Mapping müssen Sie die folgenden Regeln beachten:

- Sie können eine Gruppe mit einer Umwandlung oder einem Target verbinden.
- Sie können einen Ausgabeport in einer Gruppe mit mehreren Umwandlungen oder Targets verbinden.
- Sie können mehrere Ausgabeports in einer Gruppe mit mehreren Umwandlungen oder Targets verbinden.
- Sie können nicht mehr als eine Gruppe mit einem Target oder einer einzelnen Eingabegruppen-Umwandlung verbinden.
- Außer für Joiner-Umwandlungen können Sie mehr als eine Gruppe mit einer Umwandlung für mehrere Eingabegruppen verbinden, wenn Sie jede Ausgabegruppe mit einer anderen Eingabegruppe verbinden.

Router-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie die erweiterten Eigenschaften, mit denen Sie festlegen können, wie der Data Integration Service Daten für die Router-Umwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 37

Sequenzgeneratorumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Sequenzgeneratorumwandlung – Übersicht, 542
- Erstellen von Schlüsseln, 542
- Ersetzen fehlender Werte, 543
- Durchlaufen eines Wertebereichs, 544
- Sequenzgeneratorports, 544
- Sequenzgeneratorumwandlung-Eigenschaften, 545
- Sequenzdatenobjekt, 546
- Häufig gestellte Fragen (FAQ), 547

Sequenzgeneratorumwandlung – Übersicht

Die Sequenzgeneratorumwandlung ist eine passive Umwandlung, die numerische Werte generiert. Verwenden Sie die Sequenzgeneratorumwandlung zum Erzeugen eindeutiger Primärschlüsselwerte, zum Ersetzen fehlender Primärschlüssel oder zum Durchlaufen eines sequenziellen Nummernbereichs.

Der Sequenzgenerator enthält Pass-Through-Ports und einen Ausgabeport. Sie können den NEXTVAL-Port mit den Eingabeports anderer Umwandlungen verbinden. Der Integrationsdienst erhöht die Sequenz, wenn das Mapping ausgeführt wird.

Alle Sequenzgeneratorumwandlungen enthalten ein Sequenzdatenobjekt. Ein Sequenzdatenobjekt ist ein Objekt, das eine Sequenz von Werten erstellt und verwaltet.

Erstellen von Schlüsseln

Sie können Primär- oder Fremdschlüsselwerte mit der Sequenzgeneratorumwandlung erzeugen, indem Sie den NEXTVAL-Port mit einem Ziel oder einer nachgelagerten Umwandlung verbinden. Sie können Werte im Bereich von 1 bis 9.223.372.036.854.775.807 und mit dem kleinsten Intervall von 1 verwenden. Sie können außerdem zusammengesetzte Schlüssel erstellen, um jede Zeile in der Tabelle zu identifizieren.

Um einen zusammengesetzten Schlüssel zu erstellen, können Sie den Integrationsdienst so konfigurieren, dass er eine kleinere Gruppe von Werten durchläuft. Wenn Sie beispielsweise drei Filialen haben, die Bestellnummern generieren, können Sie eine Sequenzgeneratorumwandlung konfigurieren, die Werte von 1 bis 3 durchläuft und jeweils um 1 erhöht. Wenn Sie den ORDER_NO-Port mit der

Sequenzgeneratorumwandlung verbinden, erstellen die generierten Werte eindeutige zusammengesetzte Schlüssel.

Das folgende Beispiel zeigt zusammengesetzte Schlüssel und Auftragsnummern:

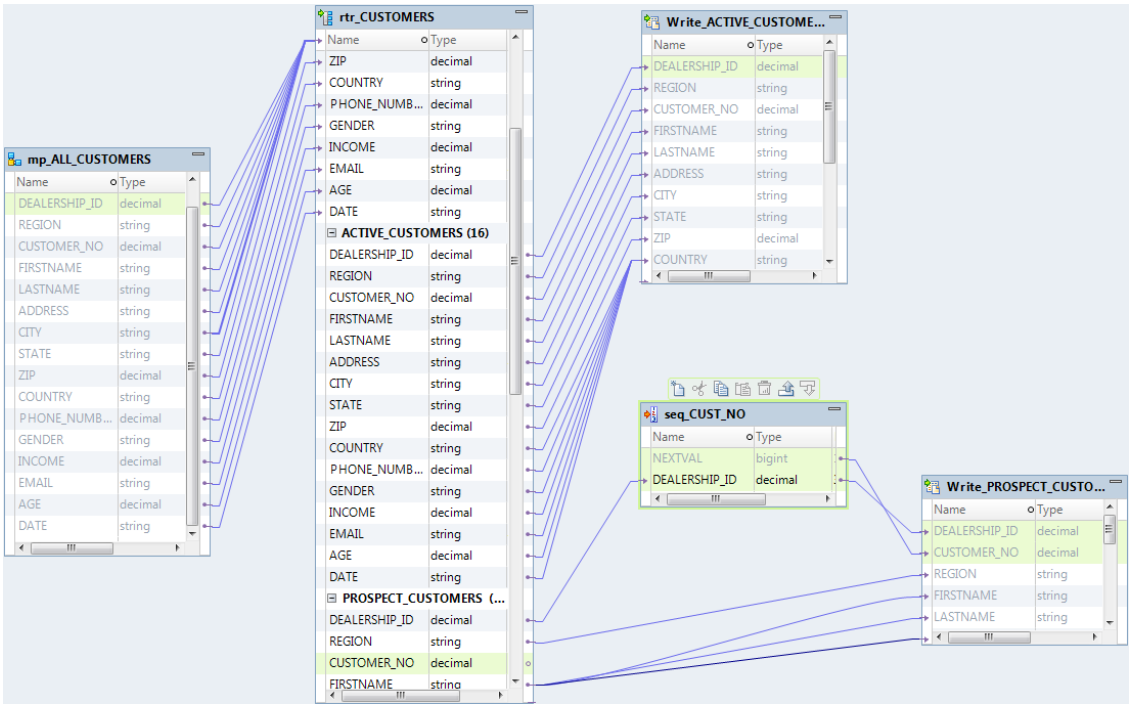
COMPOSITE_KEY	ORDER_NO
1	12345
2	12345
3	12345
1	12346
2	12346
3	12346

Ersetzen fehlender Werte

Wenn Sie eine Sequenzgeneratorumwandlung verwenden, um fehlende Schlüssel zu ersetzen, können Sie auch eine Routerumwandlung verwenden, um NULL-Werte aus Spalten mit zugewiesenen Werten zu filtern. Verbinden Sie die Router-Umwandlung mit der Sequenzgeneratorumwandlung und verwenden Sie NEXTVAL, um eine Sequenz numerischer Werte zu erzeugen und die NULL-Werte zu füllen.

Um beispielsweise NULL-Werte in einer Spalte namens CUSTOMER_NO zu ersetzen, erzeugen Sie ein Mapping mit einer Quelle, die Kundendaten enthält. Dann fügen Sie eine Router-Umwandlung hinzu, um Kunden mit zugewiesenen Kundennummern von jenen Kunden mit NULL-Werten zu trennen. Um eindeutige CUSTOMER_NO-Werte zu erzeugen, ergänzen Sie eine Sequenzgeneratorumwandlung. Fügen Sie Kundenziele hinzu, in welche die Daten geschrieben werden.

Die folgende Abbildung zeigt ein Mapping an, das NULL-Werte in der CUSTOMER_NO-Spalte ersetzt:



Durchlaufen eines Wertebereichs

Sie können einen Wertebereich für die Sequenzgeneratorumwandlung einrichten. Wenn Sie die Zyklusoption verwenden, wiederholt die Sequenzgeneratorumwandlung den Bereich, wenn der Endwert erreicht wird.

Beispiel: Wenn Sie den Sequenzbereich für den Start auf 10 und für das Ende auf 50 sowie einen Inkrementwert von 10 festlegen, erstellt die Sequenzgeneratorumwandlung die Werte 10, 20, 30, 40 und 50. Die Sequenz startet wieder bei 10.

Sequenzgeneratorports

Die Sequenzgeneratorumwandlung enthält Pass-Through-Ports und einen NEXTVAL-Ausgabeport. Sie können den Ausgabeport weder bearbeiten noch löschen.

Pass-Through-Ports

Sie können der Sequenzgeneratorumwandlung einen Port als Pass-Through-Port hinzufügen. Pass-Through-Ports sind Eingabe- und Ausgabe-Ports, die Eingabedaten empfangen und dieselben Daten in ein zurückgeben, ohne sie zu ändern.

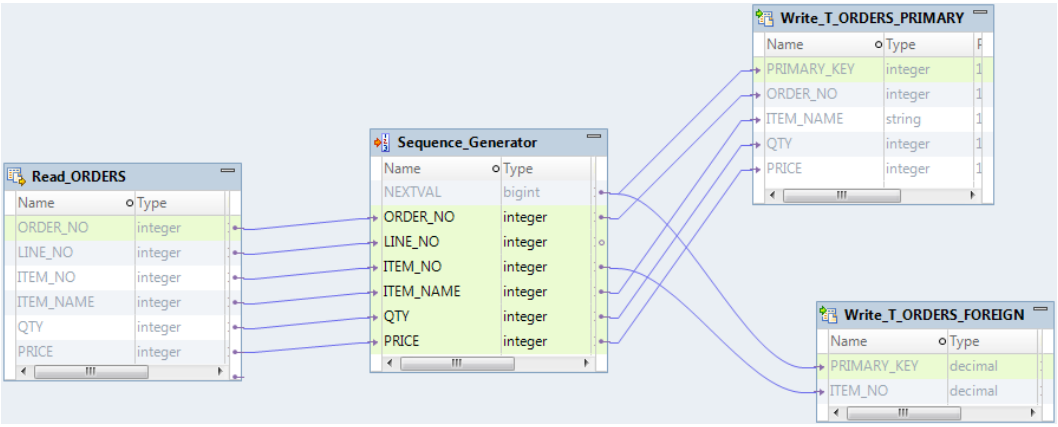
Sie müssen mindestens einen Eingabeport zur Umwandlung hinzufügen und ihn mit einer vorgelagerten Quelle oder Umwandlung verbinden, bevor Sie die NEXTVAL-Ausgabeports mit Zielen verknüpfen. Um der Umwandlung einen Übergabeport hinzuzufügen, ziehen Sie einen Port von einer vorgelagerten Quelle oder Umwandlung im Mapping in die Sequenzgeneratorumwandlung.

NEXTVAL-Port

Sie können NEXTVAL mit einer Umwandlung verbinden, um für jede Zeile in der Umwandlung eindeutige Werte zu erzeugen. Verbinden Sie den NEXTVAL-Port mit einer nachgelagerten Umwandlung oder einem Ziel, um eine Nummernsequenz zu erzeugen. Wenn Sie NEXTVAL mit mehreren Umwandlungen verbinden, erzeugt der Integrationsdienst für jede Umwandlung dieselbe Nummernsequenz.

Verbinden Sie den NEXTVAL-Port, um die Sequenz basierend auf dem Startwert und dem Inkrementwert zu erzeugen. Wenn der Sequenzgenerator nicht für den Sequenzdurchlauf konfiguriert wurde, erzeugt der NEXTVAL-Port Sequenznummern bis zum konfigurierten Endwert.

In der folgenden Abbildung wird ein Mapping angezeigt, bei dem der NEXTVAL-Port der Sequenzgeneratorumwandlung mit einer Quelle und zwei Zielen verbunden ist, um Primär- und Fremdschlüsselwerte zu erzeugen:



Wenn Sie die Sequenzgeneratorumwandlung mit einem Startwert = 1 und einem Inkrementwert = 1 konfigurieren, erzeugt der Integrationsdienst dieselben Primärschlüsselwerte für die Zieltabellen T_ORDERS_PRIMARY und T_ORDERS_FOREIGN.

Sequenzgeneratorumwandlung-Eigenschaften

Konfigurieren Sie die Umwandlungseigenschaften, die der Integrationsdienst zum Generieren von sequenziellen Werten verwendet.

In der folgenden Liste werden die Eigenschaften der Sequenzgeneratorumwandlung beschrieben, die Sie konfigurieren können:

Zurücksetzen

Bei Aktivierung setzt der Integrationsdienst das Sequenzdatenobjekt auf den Startwert zurück, wenn der Mapping-Durchlauf abgeschlossen wird. Bei Deaktivierung erhöht der Integrationsdienst den aktuellen Wert nach Abschluss des Mapping-Durchlaufs und verwendet den Wert im nächsten Mapping-Durchlauf.

Diese Eigenschaft ist für wiederverwendbare Sequenzgeneratorumwandlungen und für nicht wiederverwendbare Sequenzgeneratorumwandlungen, die ein wiederverwendbares Sequenzdatenobjekt verwenden, deaktiviert.

Tracingebene

Diese Option gibt an, wie viele Umwandlungsdetails der Integrationsdienst in das Mapping-Protokoll schreibt. Sie können „Kurz“, „Normal“, „Ausführliche Initialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Beibehalten der Zeilenreihenfolge

Beibehalten der Zeilenreihenfolge der Eingabedaten für die Umwandlung. Wählen Sie diese Option aus, wenn der Integrationsdienst keine Optimierung durchführen soll, die die Zeilenreihenfolge verändern könnte.

Wenn der Integrationsdienst Optimierungen durchführt, geht eine zu einem vorherigen Zeitpunkt im Mapping erstellte Reihenfolge möglicherweise verloren. Die Reihenfolge in einem Mapping können Sie mit einer Einfachdateiquelle, einer sortierten relationalen Quelle oder einer Sortierer-Umwandlung erstellen. Wenn Sie eine Umwandlung konfigurieren, um die Zeilenreihenfolge beizubehalten, berücksichtigt der Integrationsdienst diese Konfiguration beim Durchführen von Optimierungen für das Mapping. Der Integrationsdienst führt Optimierungen für die Umwandlung durch, wenn er die Reihenfolge beibehalten kann. Der Integrationsdienst führt keine Optimierungen für die Umwandlung durch, wenn diese Optimierungen die Zeilenreihenfolge verändern würden.

Zurücksetzen

Wenn Sie eine nicht wiederverwendbare Sequenzgeneratorumwandlung zur Verwendung der Zurücksetzungseigenschaft konfigurieren, verwendet der Integrationsdienst den ursprünglichen Startwert für jede Ausführung des Mappings. Andernfalls erhöht der Integrationsdienst den aktuellen Wert und verwendet diesen Wert in der nächsten Ausführung des Mappings.

Beispiel: Sie konfigurieren eine Sequenzgeneratorumwandlung, um Werte von 1 bis 1.000 mit einem Inkrementwert von 1 sowie einem Startwert von 1 zu konfigurieren, und Sie wählen die Option „Zurücksetzen“ aus. Während der ersten Mapping-Ausführung erzeugt der Integrationsdienst Nummern von 1 bis 234. Bei jeder folgenden Ausführung des Mappings erzeugt der Integrationsdienst erneut Nummern beginnend mit dem aktuellen Startwert 1.

Wenn Sie die Option „Zurücksetzen“ nicht auswählen, aktualisiert der Integrationsdienst den aktuellen Wert am Ende der ersten Ausführung mit dem Wert 235. Bei der nächsten Verwendung der Sequenzgeneratorumwandlung ist der erste erzeugte Wert gleich 235.

Hinweis: Die Option „Zurücksetzen“ ist für wiederverwendbare Sequenzgeneratorumwandlungen deaktiviert.

Sequenzdatenobjekt

Ein Sequenzdatenobjekt erstellt eine Sequenz mit numerischen Werten und verwaltet diese. Die Sequenzgeneratorumwandlung verwendet das Sequenzdatenobjekt zum Generieren der Werte für die Umwandlung.

Sie können ein wiederverwendbares Sequenzdatenobjekt in mehreren Sequenzgeneratorumwandlungen verwenden. Alle Sequenzgeneratorumwandlungen, die dasselbe Sequenzdatenobjekt verwenden, verwenden dieselbe Sequenz von Werten, wenn sie in demselben Integrationsdienst ausgeführt werden. Sie können ein wiederverwendbares Sequenzdatenobjekt auch in nicht wiederverwendbaren Sequenzgeneratorumwandlungen verwenden. Sie können ein nicht wiederverwendbares Sequenzdatenobjekt in einer nicht wiederverwendbaren Sequenzgeneratorumwandlung verwenden.

Beispiel: Sie erstellen mehrere Mappings, die in dasselbe Primärschlüsselfeld in einer relationalen Tabelle schreiben. Jedes Mapping verwendet dieselbe nicht wiederverwendbare Sequenzgeneratorumwandlung, die

dasselbe wiederverwendbare Sequenzdatenobjekt verwendet und in demselben Integrationsdienst ausgeführt wird. Jedes Mapping schreibt eindeutige Werte in das primäre Schlüsselfeld.

Sequenzdatenobjekt-Eigenschaften

Das Sequenzdatenobjekt verwaltet eine Sequenz von Werten in allen Sequenzgeneratorumwandlungen, in denen es verwendet wird, vorausgesetzt, alle Umwandlungen verwenden denselben Integrationsdienst.

In der folgenden Liste werden die Eigenschaften der Sequenzdatenobjekte beschrieben, die Sie im Sequenzdatenobjekt konfigurieren können:

Startwert

Anfangswert der Sequenz.

Der Maximalwert ist 9.223.372.036.854.775.806.

Standardwert ist 0.

Endwert

Dies ist der maximale Wert, den der Integrationsdienst erzeugt. Wenn der Integrationsdienst diesen Wert erreicht und die Sequenz nicht zyklisch durchlaufen werden soll, schlägt das Mapping fehl.

Wenn Sie den NEXTVAL-Port mit einem nachgelagerten Ganzzahl-Port verbinden, wählen Sie für den Endwert einen Wert, der nicht größer als der maximale Ganzzahlwert ist. Wenn der NEXTVAL-Wert den Maximalwert des Datentyps für den nachgelagerten Port überschreitet, schlägt das Mapping fehl.

Der Maximalwert ist 9,223,372,036,854,775,807.

Inkrementwert

Dies ist der Abstand zwischen zwei aufeinanderfolgenden Werten aus dem NEXTVAL-Port.

Der Maximalwert ist 2,147,483,647.

Muss ein positiver Ganzzahlwert sein.

Standardwert ist 1.

Zyklus

Wiederholen Sie die Sequenz.

Wenn diese Option aktiviert ist, durchläuft der Integrationsdienst zyklisch den Sequenzbereich. Wenn das Sequenzdatenobjekt das Ende des Bereichs erreicht, wird die Sequenz wiederholt.

Wenn diese Option deaktiviert ist, stoppt der Integrationsdienst die Sequenz beim konfigurierten Endwert. Der Integrationsdienst bricht die Mapping-Ausführung mit Überlauffehlern ab, falls er den Endwert erreicht und immer noch Zeilen zu verarbeiten hat.

Häufig gestellte Fragen (FAQ)

Kann ich eine nicht wiederverwendbare Sequenzgeneratorumwandlung ändern, um sie wiederverwendbar zu machen?

Sie können die Umwandlung nicht wiederverwendbar machen, aber Sie können die Umwandlung für die Verwendung eines Sequenzdatenobjekts ändern. Das Sequenzdatenobjekt behält die Integrität der Sequenz bei, unabhängig von der Anzahl der sie verwendenden Umwandlungen.

Kann ich eine nicht wiederverwendbare Sequenzgeneratorumwandlung im Mapplet platzieren?

Nein, das ist nicht möglich. Bei Mapplets handelt es sich um wiederverwendbare Objekte, sodass alle Objekte im Mapplet ebenfalls wiederverwendbar sein müssen. Verwenden Sie stattdessen eine wiederverwendbare Sequenzgeneratorumwandlung.

KAPITEL 38

Sortierer-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Sortierer-Umwandlung – Übersicht, 549
- Sortierer-Umwandlungen in dynamischen Mappings, 550
- Entwicklung einer Sortier-Umwandlung, 550
- Sortierer-Umwandlungsports, 551
- Registerkarte „Sortieren“, 551
- Sortierschlüssel konfigurieren, 551
- Sortierer-Umwandlung – Erweiterte Eigenschaften, 554
- Sortierer-Cache, 554
- Erstellen einer Sortier-Umwandlung, 555
- Sortierer-Umwandlung – Beispiel, 556

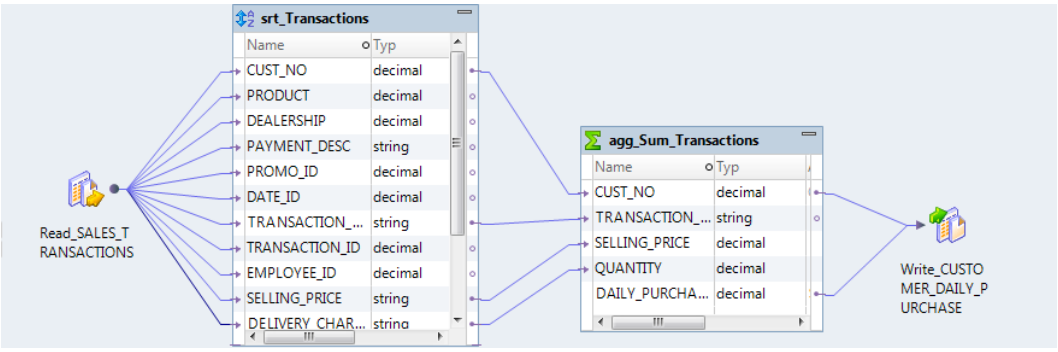
Sortierer-Umwandlung – Übersicht

Verwenden Sie eine Sortierer-Umwandlung zum Sortieren von Daten in auf- oder absteigender Reihenfolge nach einem angegebenen Sortierschlüssel. Sie können die Sortierer-Umwandlung für Sortierung nach Groß- und Kleinschreibung oder für eine unterschiedliche Ausgabe konfigurieren. Die Sortierer-Umwandlung ist eine aktive Umwandlung.

Beim Erstellen einer Sortierer-Umwandlung legen Sie Ports als Sortierschlüssel fest und konfigurieren die einzelnen Sortierschlüsselports für die Sortierung in auf- oder absteigender Reihenfolge. Der Data Integration Service sortiert die Ports nacheinander, wenn Sie mehrere Ports als Sortierschlüssel angegeben haben.

Beispielsweise müssen Sie eine Rechnung für die Gesamtverkäufe an Kunden anhand einer Kundendatenbank erstellen. Wenden Sie eine Sortierer-Umwandlung auf die Kundenverkaufstabelle an, um die Daten in absteigender Reihenfolge entsprechend der Kundennummer zu sortieren. Verwenden Sie das Ergebnis der Sortierer-Umwandlung als Eingabe in die Aggregator-Umwandlung. Sie können die Leistung der Aggregator-Umwandlung mithilfe der Option "Sortierte Eingabe" steigern.

Die folgende Abbildung zeigt das Mapping:



Sortierer-Umwandlungen in dynamischen Mappings

Sie können eine Sortierer-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports verweisen.

Sie können auf einen dynamischen Port oder einen generierten Port in der Sortierer-Umwandlung verweisen. Wenn die generierten Ports zur Laufzeit jedoch nicht vorhanden sind, schlägt das Mapping fehl.

Wenn Sie einen dynamischen Port als Sortierschlüssel verwenden, berücksichtigt der Datenintegrationsdienst alle generierten Ports im dynamischen Port und in der generierten Portreihenfolge.

Sie können die Sortierschlüssel parametrisieren. Verwenden Sie einen Sortierlistenparameter für den Sortierschlüssel.

Entwicklung einer Sortier-Umwandlung

Wenn Sie eine Sortier-Umwandlung entwickeln, müssen Sie einige Faktoren beachten, so zum Beispiel die Sortierschlüsselports, einzelne Ausgabezeilen und Sortierkriterien unter Beachtung der Groß- und Kleinschreibung.

Berücksichtigen Sie die folgenden Faktoren beim Entwickeln einer Sortier-Umwandlung:

- Die Ports, die Sie als Sortierschlüssel konfigurieren möchten und die Sortierrichtung.
- Ob Sie eine Sortierung unter Beachtung der Groß- und Kleinschreibung durchführen möchten.
- Ob Sie Nullwerte als Sortierpriorität verwenden möchten.
- Ob Sie einzelne Ausgabezeilen berücksichtigen möchten.
- Der Wert für die Sortier-Cache-Größe, den Sie festlegen möchten.

Sortierer-Umwandlungsports

Beim Erstellen von Ports in der Sortierer-Umwandlung erstellen Sie standardmäßig Eingabe- und Ausgabeports. Die Sortierer-Umwandlung gibt dieselben Ausgabeports wie Eingabeports zurück.

Sie können dynamische Ports in einer Sortierer-Umwandlung definieren. Ein dynamischer Port kann verschiedene Datenspalten aus einer vorgelagerten Umwandlung in einem Mapping empfangen. Damit kann die Sortierer-Umwandlung Zeilen sortieren, die unterschiedliche Spalten enthalten.

Definieren Sie die Sortierschlüssel auf der Registerkarte **Sortieren** in der Ansicht **Eigenschaften**.

Registerkarte „Sortieren“

Definieren Sie den Sortierschlüssel auf der Registerkarte **Sortieren** in der Ansicht **Eigenschaften** der Sortierer-Umwandlung. Wählen Sie mindestens einen Port, den Sie als Sortierkriterium verwenden möchten.

Der Datenintegrationsdienst sortiert die Daten entsprechend der Reihenfolge der Ports auf der Registerkarte „Sortieren“. Konfigurieren Sie das Sortieren der Daten in aufsteigender oder absteigender Reihenfolge. Der Standardwert ist „aufsteigend“.

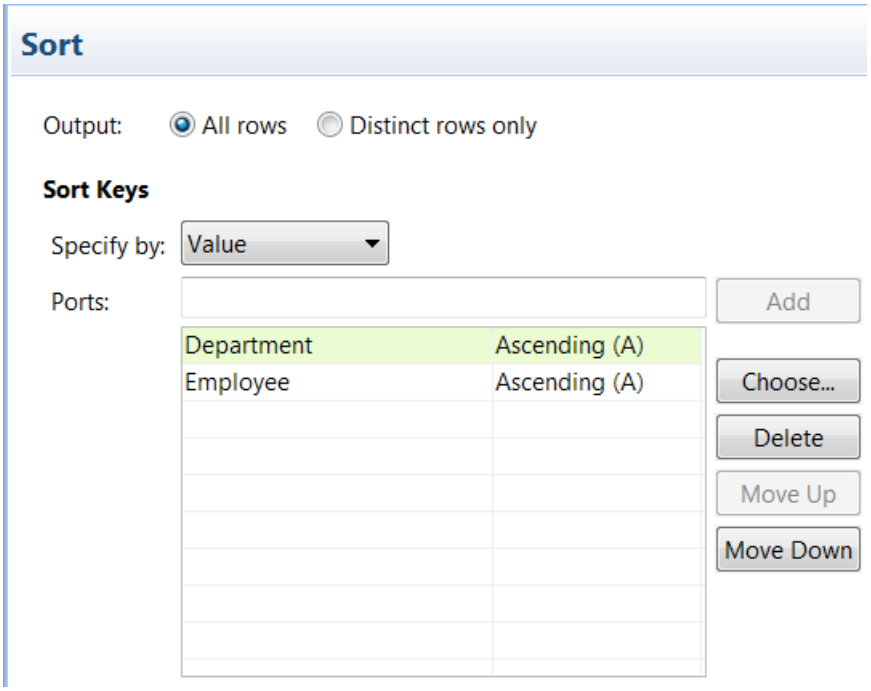
Wenn Sie die Sortierer-Umwandlung für eindeutige Ausgabezeilen konfigurieren, konfiguriert das Developer Tool alle Ports als Teil des Sortierschlüssels. Der Datenintegrationsdienst verwirft während des Sortiervorgangs doppelte Zeilen.

Sortierschlüssel konfigurieren

Definieren Sie den Sortierschlüssel auf der Registerkarte **Sortieren** in der Ansicht **Eigenschaften** der Umwandlung.

Dies ist der Beginn des Konzepts.

Die folgende Abbildung zeigt die Registerkarte **Sortieren**:



Die Registerkarte **Sortieren** enthält folgende Optionen:

Ausgabe

Wählen Sie, ob alle sortierten Zeilen zurückgegeben oder doppelte Zeilen verworfen werden sollen. Doppelte Zeilen sind Zeilen, in denen alle Spaltenwerte identisch sind.

Angeben nach

Wählen Sie **Wert** oder **Parameter** aus. Wählen Sie **Wert** für die Verwendung von Portnamen aus. Wählen Sie **Parameter** für die Verwendung eines Parameters für die Sortierliste aus.

Hinzufügen

Akzeptiert einen Portnamen, den Sie manuell eingeben. Sie müssen einen gültigen Namen eingeben, bevor Sie auf **Hinzufügen** klicken.

Auswählen

Klicken Sie auf **Auswählen**, um Ports auszuwählen, die zum Sortierschlüssel hinzugefügt werden sollen. Das Developer Tool enthält eine Liste mit Ports aus der Umwandlung, die zur Auswahl stehen.

Nach oben verschieben und nach unten verschieben

Sie können die Reihenfolge der Ports in der Gruppe ändern. Wählen Sie den Portnamen aus und klicken Sie anschließend auf eine der Schaltflächen zum Verschieben, um den Portnamen in der Sortierreihenfolge nach oben oder nach unten zu verschieben.

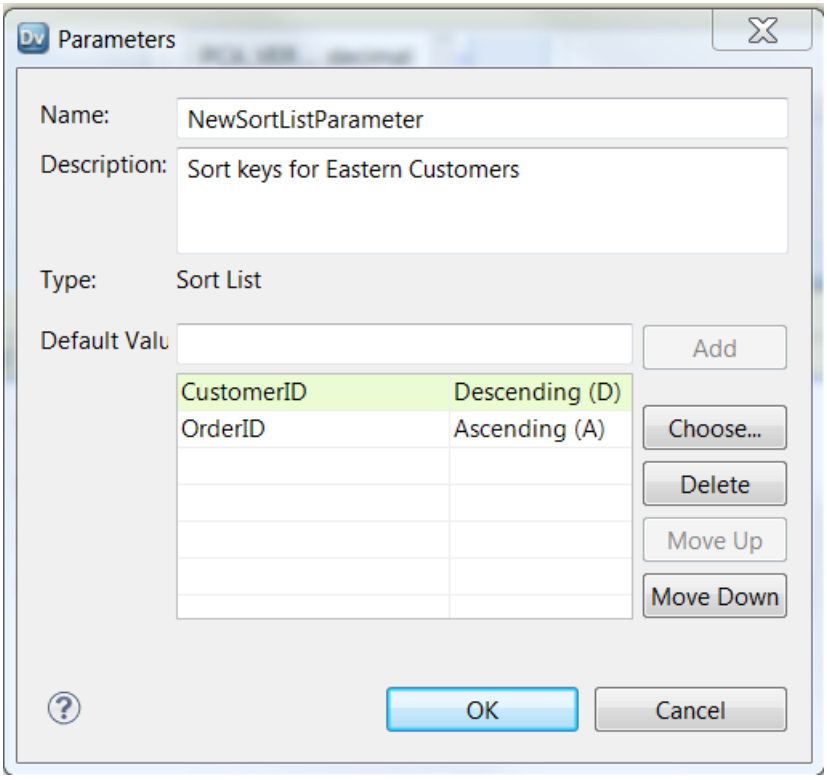
Die Sortierschlüssel parametrisieren

Sie können einen Sortierlistenparameter erstellen, der eine Liste mit Ports für die Sortierschlüssel enthält.

Wenn sich die Sortierer-Umwandlung in einem dynamischen Mapping befindet, enthält die Sortierer-Umwandlung möglicherweise generierte Ports. Sie können die Sortierschlüssel parametrisieren. Erstellen Sie einen Sortierlistenparameter, der eine Liste mit Ports enthält, nach denen sortiert werden soll.

Wählen Sie auf der Registerkarte **Sortieren** der Umwandlungseigenschaften die Option **Angeben nach Parameter** aus. Klicken Sie zum Erstellen eines Parameters auf **Neu**.

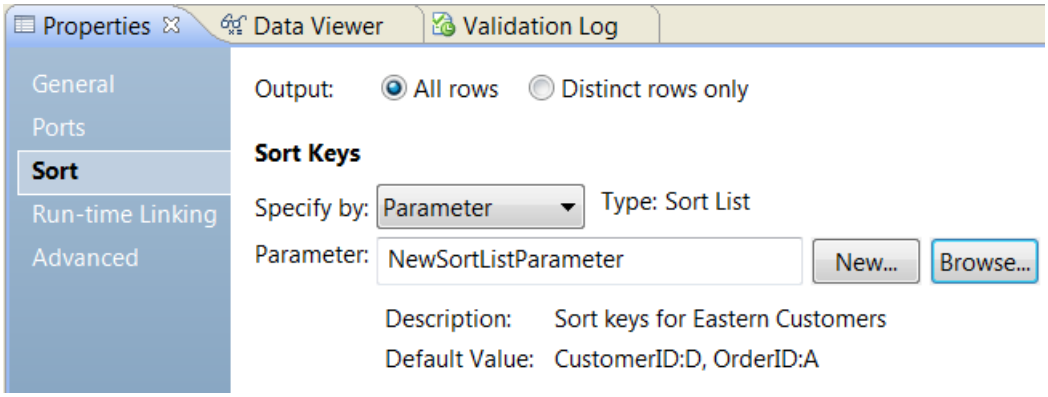
Die folgende Abbildung zeigt das Dialogfeld **Parameter**:



Wählen Sie Ports oder generierte Ports für die Sortierschlüssel aus. Sie können einen aufsteigenden oder einen absteigenden Sortiertyp auswählen.

Sie können Portnamen manuell eingeben. Geben Sie den Portnamen im Feld **Standardwert** ein und klicken Sie auf **Hinzufügen**. Das Developer Tool fügt den Portnamen zur Sortierliste hinzu.

Die folgende Abbildung zeigt die Registerkarte **Sortieren**, nachdem Sie einen Parameter für die Sortierschlüssel konfiguriert haben:



Sortierer-Umwandlung – Erweiterte Eigenschaften

In den erweiterten Eigenschaften der Sortierer-Umwandlung können Sie zusätzliche Sortierkriterien festlegen. Der Datenintegrationsdienst wendet die Eigenschaften auf alle Sortierschlüsselports an. Die Sortierer-Umwandlungseigenschaften bestimmen außerdem die Systemressourcen, die der Datenintegrationsdienst beim Sortieren von Daten zuweist.

In folgendem Abschnitt werden die erweiterten Eigenschaften für eine Sortierer-Umwandlung beschrieben:

Groß-/Kleinschreibung

Bestimmt, ob der Datenintegrationsdienst beim Sortieren von Daten die Groß-/Kleinschreibung berücksichtigt. Wenn Sie die Groß-/Kleinschreibungs-Eigenschaft aktivieren, sortiert der Datenintegrationsdienst Großbuchstaben vor Kleinbuchstaben ein. Das Developer Tool legt die Groß-/Kleinschreibung standardmäßig fest.

Null niedrig behandelt

Stuft einen Nullwert niedriger als andere Werte ein. Aktivieren Sie die Eigenschaft, wenn der Datenintegrationsdienst beim Sortieren einen Nullwert niedriger als alle anderen Werte einstufen soll.

Sortierer-Cache-Größe

Speichermenge, die der Datenintegrationsdienst zu Beginn der Mapping-Ausführung zuweist, um die Sortierung auszuführen. Der Datenintegrationsdienst übergibt vor dem Sortieren alle eingehenden Daten an die Sortierer-Umwandlung. Wenn Sie „Auto” auswählen, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Arbeitsspeicher automatisch. Geben Sie bei Anpassung der Cache-Größe den jeweiligen Wert in Byte an. Standardwert ist „Auto“.

Arbeitsverzeichnis

Verzeichnis, in dem der Datenintegrationsdienst Daten vorübergehend speichert, wenn die Menge an eingehenden Daten größer ist als die Sortierer-Cache-Größe. Nach dem Sortieren der Daten durch den Datenintegrationsdienst werden die temporären Dateien gelöscht.

Geben Sie mehrere Verzeichnisse durch Semikolon getrennt ein, um die Leistung während der Cache-Partitionierung zu erhöhen. Die Cache-Partitionierung erstellt für jede Partition, die die Umwandlung verarbeitet, einen separaten Cache.

Standardwert ist der TempDir-Systemparameter. Sie können für dieses Feld einen anderen Systemparameter oder einen benutzerdefinierten Parameter konfigurieren.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Verwandte Themen:

- “Cachegröße” auf Seite 662

Sortierer-Cache

Der Datenintegrationsdienst erstellt einen Cache im Speicher, um die Sortierer-Umwandlung auszuführen. Der Datenintegrationsdienst übergibt vor dem Sortieren alle eingehenden Daten an die Sortierer-Umwandlung.

Wenn der Datenintegrationsdienst mehr Speicherplatz benötigt als im Speicher-Cache vorhanden ist, speichert er die Daten vorübergehend im Arbeitsverzeichnis der Sortierer-Umwandlung.

Wenn Sie die Cache-Größe nicht zum Sortieren aller Daten im Speicher konfigurieren, wird im Sitzungsprotokoll eine Warnung angezeigt, dass der Datenintegrationsdienst mehrere Übergaben der Quelldaten durchgeführt hat. Der Datenintegrationsdienst führt mehrere Übergaben der Daten durch, wenn für das Abschließen der Sortierung Informationen auf eine Festplatte ausgelagert werden müssen. Die Meldung enthält Informationen über den für eine einzelne Übergabe erforderlichen Speicher. Bei der einzelnen Übergabe liest der Datenintegrationsdienst die Daten nur einmal und führt die Sortierung ohne Auslagern auf die Festplatte aus. Um die Mapping-Leistung zu optimieren, konfigurieren Sie die Cache-Größe so, dass der Datenintegrationsdienst nur eine Übergabe der Daten durchführt.

Wenn die Menge der eingehenden Daten größer ist als die Sortierer-Cache-Größe, speichert der Datenintegrationsdienst die Daten vorübergehend im Arbeitsverzeichnis der Sortierer-Umwandlung. Beim Speichern von Daten im Arbeitsverzeichnis benötigt der Datenintegrationsdienst Festplattenspeicher im Umfang von mindestens der zweifachen Menge der eingehenden Daten.

Um eine optimale Leistung zu erzielen, konfigurieren Sie die Sortierer-Cache-Größe mit einem Wert kleiner oder gleich der verfügbaren physischen Speichermenge auf dem Rechner, auf dem das Mapping ausgeführt wird. Um Daten mithilfe einer Sortierer-Umwandlung zu sortieren, stellen Sie mindestens 16 MB (16.777.216 Byte) an physischem Speicher bereit. Die Sortierer-Cache-Größe ist standardmäßig auf "Auto" eingestellt.

Erstellen einer Sortier-Umwandlung

Sie können wiederverwendbare oder nicht wiederverwendbare Sortier-Umwandlungen erstellen.

Erstellen einer wiederverwendbaren Sortierer-Umwandlung

Erstellen Sie eine wiederverwendbare Sortierer-Umwandlung zur Verwendung in mehreren Mappings oder Mapplets.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung.**

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die Sortierer-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Geben Sie einen Namen für die Umwandlung ein.
6. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.
7. Klicken Sie auf **Neu**, um der Umwandlung einen Port hinzuzufügen.
8. Bearbeiten Sie den Port, um den Namen, den Datentyp und die Präzision zu bearbeiten.
9. Wählen Sie auf der Registerkarte **Sortieren** die Ports aus, nach denen sortiert werden soll, oder wählen Sie einen Sortierlistenparameter aus.
10. Klicken Sie auf die Ansicht **Erweitert** und bearbeiten Sie die Umwandlungseigenschaften.

Erstellen einer nicht wiederverwendbaren Sortierer-Umwandlung

Erstellen Sie eine nicht wiederverwendbare Sortierer-Umwandlung in einem Mapping oder in Mapplets.

1. Ziehen Sie in einem Mapping oder Mapplet eine Sortierer-Umwandlung aus der Umwandlungspalette zum Editor.

   Die Umwandlung wird im Editor angezeigt.
2. Bearbeiten Sie in der Ansicht **Eigenschaften** den Namen und die Beschreibung der Umwandlung.
3. Klicken Sie auf der Registerkarte **Ports** auf **Neu**, um Ports zur Umwandlung hinzuzufügen.
4. Bearbeiten Sie die Ports, um den Namen, den Datentyp und die Genauigkeit festzulegen.
5. Wählen Sie **Schlüssel** aus, um den Port als Sortierschlüssel anzugeben.
6. Klicken Sie auf die Registerkarte **Erweitert** und bearbeiten Sie die Umwandlungseigenschaften.

Sortierer-Umwandlung – Beispiel

Sie haben eine Datenbanktabelle PRODUCT_ORDERS, die Informationen zu allen Aufträgen enthält, die vom Kunden erteilt wurden.

ORDER_ID	ITEM_ID	ITEM	QUANTITY	PRICE
43	123456	ItemA	3	3.04
41	456789	ItemB	2	12.02
43	000246	ItemC	6	34.55
45	000468	ItemD	5	0.56
41	123456	ItemA	4	3.04
45	123456	ItemA	5	3.04
45	456789	ItemB	3	12.02

Verwenden Sie die Sortierer-Umwandlung für PRODUCT_ORDERS, und geben Sie ORDER_ID als Sortierschlüssel mit absteigender Reihenfolge an.

Nach dem Sortieren der Daten übergibt der Data Integration Service die folgenden Zeilen aus der Sortierer-Umwandlung:

ORDER_ID	ITEM_ID	ITEM	QUANTITY	PRICE
45	000468	ItemD	5	0.56
45	123456	ItemA	5	3.04
45	456789	ItemB	3	12.02
43	123456	ItemA	3	3.04
43	000246	ItemC	6	34.55
41	456789	ItemB	2	12.02

ORDER_ID	ITEM_ID	ITEM	QUANTITY	PRICE
41	123456	ItemA	4	3.04

Sie benötigen die Gesamtmenge und Artikelmenge pro Auftrag. Sie können das Ergebnis der Sortierer-Umwandlung als Eingabe in eine Aggregator-Umwandlung verwenden. Verwenden Sie die sortierte Eingabe in der Aggregator-Umwandlung, um die Leistung zu erhöhen.

Wenn Sie die sortierte Eingabe nicht verwenden, führt der Data Integration Service Aggregatberechnungen beim Lesen durch. Der Data Integration Service speichert Daten für jede einzelne Gruppe, bis er die gesamte Quelle gelesen hat, um sicherzustellen, dass alle Aggregatberechnungen korrekt sind. Wenn Sie die sortierte Eingabe verwenden und Daten nicht ordnungsgemäß vorsortieren, erhalten Sie unerwünschte Ergebnisse.

Bei der Aggregator-Umwandlung wird die ORDER_ID nach Port gruppiert, die Option für sortierte Eingabe ist ausgewählt. Wenn Sie die Daten aus der Sortierer-Umwandlung übergeben, gruppiert die Aggregator-Umwandlung ORDER_ID so, dass der Gesamtbetrag pro Auftrag berechnet wird.

ORDER_ID	SUM
45	54.06
43	216.42
41	36.2

KAPITEL 39

SQL-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- SQL-Umwandlung – Übersicht, 558
- SQL-Umwandlungsports, 559
- SQL-Umwandlung – Erweiterte Eigenschaften, 563
- SQL-Umwandlungsabfrage, 565
- Kardinalität zwischen Eingabezeile und Ausgabezeile, 567
- Filteroptimierung mit der SQL-Umwandlung, 571
- SQL-Umwandlung - Beispiel mit einer SQL-Abfrage, 573
- Gespeicherte Prozeduren, 577
- SQL-Umwandlungsverbindung, 582
- Manuelles Erstellen einer SQL-Umwandlung, 583
- Erstellen einer SQL-Umwandlung aus einer gespeicherten Prozedur, 584

SQL-Umwandlung – Übersicht

Die SQL-Umwandlung verarbeitet SQL-Abfragen-Midstream in einem Mapping. Sie können SQL-Abfragen aus der SQL-Umwandlung ausführen oder die SQL-Umwandlung zur Ausführung der gespeicherten Prozeduren aus einer Datenbank konfigurieren.

Sie können die Eingabeportwerte an Parameter in der Abfrage oder der gespeicherten Prozedur übergeben. Die Umwandlung kann Zeilen einfügen, löschen, aktualisieren und aus einer Datenbank abrufen. Sie können SQL DDL-Anweisungen ausführen, um eine Tabelle zu erstellen oder ein Tabellen-Midstream im Mapping abzulegen. Die SQL-Umwandlung ist eine aktive Umwandlung. Die Umwandlung kann für jede Eingabezeile mehrere Zeilen zurückgeben.

Sie können eine gespeicherte Prozedur aus einer Datenbank in die SQL-Umwandlung importieren. Beim Importieren der gespeicherten Prozedur erstellt das Developer-Tool die Umwandlungsports, die den Parametern in der gespeicherten Prozedur entsprechen. Das Developer-Tool erstellt auch die gespeicherte Prozedur für Sie.

Um eine SQL-Umwandlung zur Ausführung einer gespeicherten Prozedur auszuführen, führen Sie die folgenden Aufgaben durch:

1. Definieren Sie die Umwandlungseigenschaften, einschließlich des Datenbanktyps, zu dem die Verbindung hergestellt werden soll.

2. Importieren Sie eine gespeicherte Prozedur, um die Ports zu definieren und den gespeicherten Prozeduraufruf zu erstellen.
3. Definieren Sie Ports für Ergebnissätze oder weitere gespeicherte Prozeduren, die Sie ausführen müssen.
4. Fügen Sie die zusätzlichen gespeicherten Prozeduraufrufe zum SQL-Editor hinzu.

Sie können eine SQL-Abfrage im Umwandlungs-SQL-Editor erstellen. Wenn Sie eine SQL-Umwandlung ausführen, verarbeitet die Umwandlung die Abfrage und gibt Zeilen und alle Datenbankfehler zurück.

Um eine SQL-Umwandlung zur Ausführung einer Abfrage auszuführen, führen Sie die folgenden Aufgaben durch:

1. Definieren Sie die Umwandlungseigenschaften, einschließlich des Datenbanktyps, zu dem die Verbindung hergestellt werden soll.
2. Definieren Sie die Eingabe- und Ausgabeports.
3. Erstellen Sie eine SQL-Abfrage im SQL-Editor.

Konfigurieren Sie nach dem Konfigurieren der Umwandlung die SQL-Umwandlung in einem Mapping und verbinden Sie die vorangehenden Ports. Zeigen Sie die Daten in der Vorschau an, um die Ergebnisse zu überprüfen.

SQL-Umwandlungsports

Beim Erstellen einer SQL-Umwandlung legt das Developer Tool standardmäßig den SQLError-Port an. Fügen Sie in der Ansicht **Ports** Eingabe-, Ausgabe- und Pass-Through-Ports ein.

Die SQL-Umwandlung weist die folgenden Porttypen auf:

Eingabe

Empfängt Quelldaten, die in einer SQL-Abfrage verwendet werden können.

Ausgabe

Gibt Datenbankdaten aus einer SQL SELECT-Abfrage zurück.

Pass-Through

Eingabe-/Ausgabeports, die Quelldaten ohne Änderung über die Umwandlung übergeben.

SQLError

Gibt SQL-Fehler aus der Datenbank zurück. Wenn keine Fehler auftreten, wird NULL zurückgegeben.

NumRowsAffected

Gibt die Gesamtanzahl der von den Abfrageanweisungen INSERT, DELETE und UPDATE für eine Eingabezeile betroffenen Datenbankzeilen zurück. Das Developer Tool erstellt diesen Port, wenn die Aktualisierungsstatistik in die Ausgabezeile aufgenommen werden soll.

Rückgabewert

Empfängt den Rückgabewert aus einer gespeicherten Prozedur.

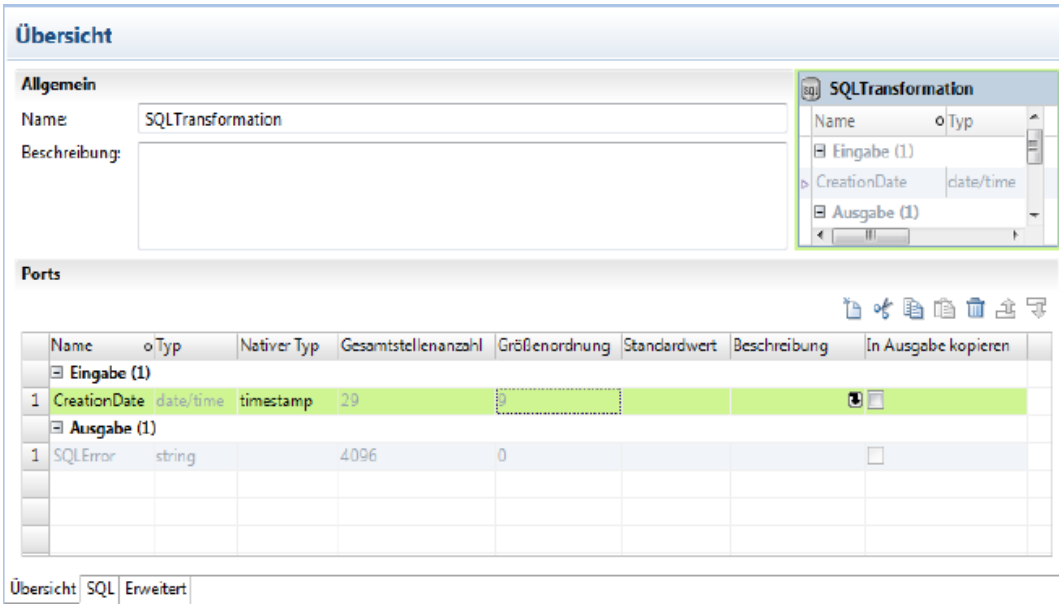
Eingabeports

Sie können Eingabeports für SQL-Umwandlungen mit Parameterbindung in SQL-Anweisungen und gespeicherten Prozeduren jedes Typs referenzieren. Sie können Eingabeports in der SQL-Umwandlung für Daten erstellen, die Sie nicht an Ausgabeports übergeben möchten.

Sie müssen Ports manuell hinzufügen, wenn Sie eine SQL-Abfrage konfigurieren, die über Eingabeparameter verfügt. Beim Importieren einer gespeicherten Prozedur in die SQL-Umwandlung erstellt die SQL-Umwandlung die Eingabeports. Sie können Pass-Through-Ports hinzufügen, um Daten über die Umwandlung zu übergeben, ohne sie zu ändern.

Sie können Ports in der Ansicht **Übersicht** hinzufügen. Wenn Sie einen Port hinzufügen, geben Sie den nativen Datentyp für den Port ein. Der native Datentyp ist ein Datentyp, der für die Datenbank, zu der Sie eine Verbindung herstellen, gültig ist. Beim Konfigurieren eines nativen Datentyps wird ein Umwandlungsdatentyp angezeigt. Wenn Sie Zeilen in die SQL-Umwandlung ziehen, legt das Developer-Tool den nativen Datentyp basierend auf Datentypen fest, die für die Datenbank, zu der Sie eine Verbindung herstellen, gültig sind. Stellen Sie sicher, dass die Datentypen für die Spalten, die Sie in der Abfrage verwenden, den Datentypen der Spalten in der Datenbank entsprechen.

Die folgende Abbildung zeigt den Eingabeport **CreationDate** in einer wiederverwendbaren SQL-Umwandlung:



Klicken Sie zum Hinzufügen von Eingabeports im Bereich **Ports** auf **Eingabe**. Klicken Sie auf **Neu**.

Hinweis: Wenn Sie die Option **In Ausgabe kopieren** für einen Port wählen, wird aus dem Eingabeport ein Pass-Through-Port. Pass-Through-Ports werden in den Abschnitten **Eingabe** und **Ausgabe** der Ansicht **Ports** aufgeführt.

Ausgabeports

SQL-Umwandlungsausgabeports geben Werte aus einer Abfrageanweisung oder aus einer gespeicherten Prozedur zurück.

Sie müssen die Ausgabeports definieren, wenn Sie eine SQL-Umwandlung manuell konfigurieren. Definieren Sie einen Ausgabeport für jeden Ausgabeparameter einer gespeicherten Prozedur oder für jeden Port, den die SELECT-Anweisung zurückgibt.

Beim Importieren einer gespeicherten Prozedur erstellt das Developer-Tool einen Ausgabeport für jeden Ausgabeparameter, den die Prozedur zurückgibt. Wenn die Prozedur einen Ergebnissatz zurückgibt, müssen Sie die Ausgabeports im Ergebnissatz manuell definieren. Eine gespeicherte Prozedur kann einen Ergebnissatz und Ausgabeparameter zurückgeben, die nicht Teil des Ergebnissatz-Caches in derselben Ausführung sind. Sie müssen die Ausgabeports für die Ergebnissatzfelder und für die Ausgabeparameter definieren.

Wählen Sie beim Konfigurieren eines Ausgabeports den nativen Datentyp für den Port aus. Der native Datentyp eines Ausgabeports muss mit dem Datentyp der entsprechenden Spalte in der Datenbank übereinstimmen. Beim Konfigurieren des nativen Datentyps definiert das Developer Tool den Umwandlungs-Datentyp für den Port.

Die SQL-Umwandlung enthält beispielsweise die folgende SQL-Abfrage für eine Oracle-Datenbank:

```
SELECT FirstName, LastName, Age FROM EMPLOYEES
```

Sie können die folgenden Ausgabeports und die nativen Datentypen in der SQL-Umwandlung konfigurieren:

Ausgabeport	Nativer Datentyp	Umwandlungsdatentyp
FirstNm	varchar2	string
LastNm	varchar2	string
Age	number	double

Die Anzahl der Ausgabeports und die Reihenfolge der Ausgabeports muss mit der Anzahl und der Reihenfolge der Spalten übereinstimmen, die die Abfrage oder die gespeicherte Prozedur zurückgibt. Wenn die Anzahl der Ausgabeports größer als die Anzahl der Spalten in der Abfrage oder gespeicherten Prozedur ist, geben die zusätzlichen Ports einen Nullwert zurück. Wenn die Anzahl der Ausgabeports kleiner als die Anzahl der Spalten in der SQL-Abfrage ist, generiert der Datenintegrationsdienst einen Zeilenfehler.

Wenn Sie den Datenbanktyp, mit dem die Umwandlung verbunden ist, ändern, ändert das Developer Tool die nativen Typen des Ausgabeports. Das Developer Tool wählt möglicherweise nicht den richtigen Datentyp für alle Ports aus. Wenn Sie den Datenbanktyp ändern, stellen Sie sicher, dass der native Datentyp für jeden Ausgabeport mit dem Datentyp der Spalte in der Datenbank übereinstimmt. Das Developer Tool kann beispielsweise nVarchar2 für eine Oracle-Datenbankspalte auswählen. Möglicherweise müssen Sie den Datentyp in varchar2 ändern.

Konfigurieren Sie Ausgabeports in der Ansicht **Übersicht** der SQL-Umwandlung.

Pass-Through-Ports

Pass-Through-Ports sind Eingabe-Ausgabeports, die Daten über die Umwandlung weiterleiten, ohne die Daten zu ändern. Die SQL-Umwandlung gibt Daten in den Pass-Through-Ports zurück, unabhängig davon, ob eine SQL-Abfrage Zeilen zurückgibt oder nicht.

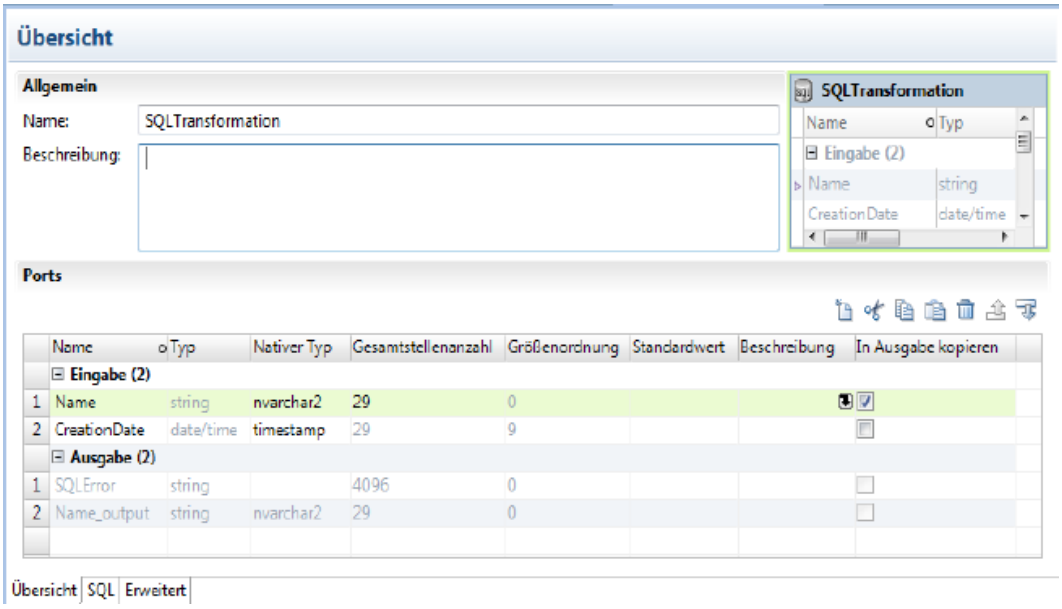
Wenn die Eingabezeile eine SELECT-Abfrageanweisung enthält, gibt die SQL-Umwandlung die Daten im Pass-Through-Port für alle Zeilen zurück, die sie von der Datenbank zurückgibt. Wenn das Abfrageergebnis mehrere Zeilen enthält, wiederholt die SQL-Umwandlung die Pass-Through-Daten in jeder Zeile.

Wenn eine Abfrage keine Zeilen zurückgibt, gibt die SQL-Umwandlung die Pass-Through-Spaltendaten mit Nullwerten in den Ausgabespalten zurück. Abfragen, die beispielsweise die Anweisungen INSERT, UPDATE und DELETE enthalten, geben keine Zeilen zurück. Wenn die Abfrage Fehler beinhaltet, gibt die SQL-Umwandlung die Pass-Through-Spaltendaten, die SQLError-Meldung und Nullwerte in den Ausgabeports zurück.

Sie können einen Pass-Through-Port konfigurieren, um Daten aus einer SELECT-Abfrage zurückzugeben.

Um einen Pass-Through-Port zu erstellen, erstellen Sie einen Eingabeport und wählen Sie **In Ausgabe kopieren** aus. Das Developer Tool erstellt einen Ausgabeport und fügt dem Portnamen einen "_output"-Suffix hinzu. Sie können den Ausgabeport nicht ändern, den das Developer Tool für einen Pass-Through-Port erstellt. Sie können keinen Ausgabeport mit dem Suffix "_output" erstellen.

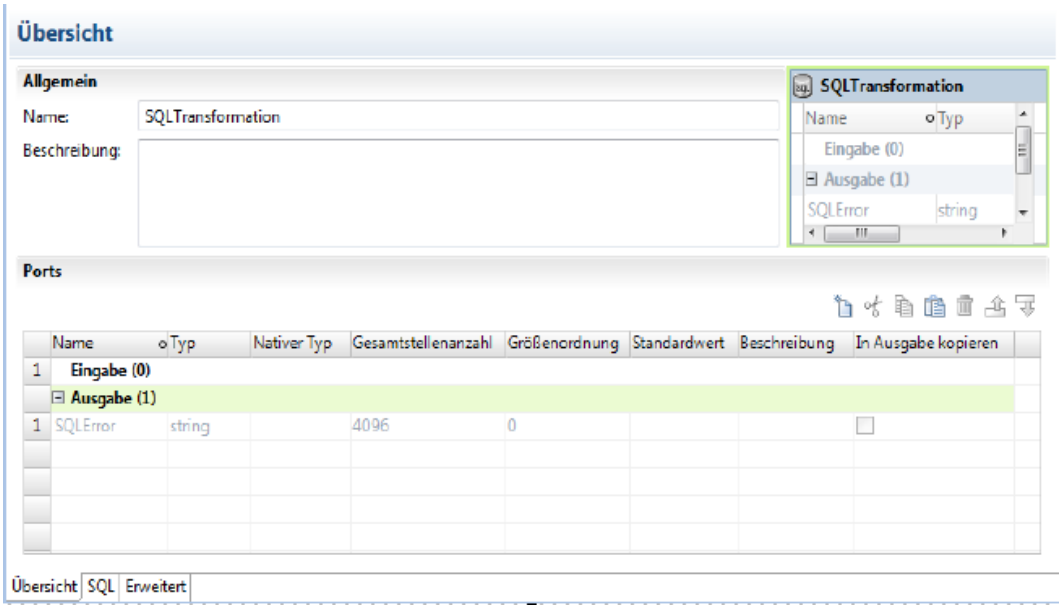
Die folgende Abbildung zeigt einen Pass-Through-Port in einer wiederverwendbaren SQL-Umwandlung:



SQLError-Port

Der SQLError-Port gibt SQL-Fehler aus der Datenbank von gespeicherten Prozeduren oder SQL-Abfragen zurück.

Die folgende Abbildung zeigt den SQLError-Port in einer wiederverwendbaren SQL-Umwandlung:



Wenn die SQL-Abfrage Syntaxfehler aufweist, enthält der SQLError-Port den Fehlertext aus der Datenbank. Die folgende SQL-Abfrage generiert beispielsweise einen Zeilenfehler aus einer Oracle-Datenbank.

```
SELECT Product_ID FROM Employees
```

Die Tabelle "Mitarbeiter" enthält keine Produkt-ID. Der Datenintegrationsdienst erzeugt eine Zeile. Der SQLError-Port enthält den Fehlertext in einer Zeile:

```
ORA-0094: “Product_ID”: invalid identifier Database driver error... Function Name:
Execute SQL Stmt: SELECT Product_ID from Employees Oracle Fatal Error
```

Sie können mehrere Abfrageanweisungen in der SQL-Abfrage konfigurieren oder Sie können mehrere gespeicherte Prozeduren aufzurufen. Beim Konfigurieren der SQL-Umwandlung zur Fortsetzung bei SQL-Fehlern gibt die SQL-Umwandlung unter Umständen Zeilen für eine Abfrageanweisung, jedoch Datenbankfehler für andere Abfrageanweisungen zurück. Die SQL-Umwandlung gibt alle Datenbankfehler in separaten Zeilen zurück.

Anzahl der betroffenen Zeilen

Aktivieren Sie den NumRowsAffected-Ausgabeport, um die Anzahl der Zeilen zurückzugeben, die die Abfrageanweisungen INSERT, UPDATE oder DELETE für jede Eingabezeile ändert. Sie können den NumRowsAffected-Ausgabeport für SQL-Abfragen konfigurieren.

Der Datenintegrationsdienst gibt die NumRowsAffected-Werte für jede Anweisung in der Abfrage zurück. NumRowsAffected ist standardmäßig deaktiviert.

Wenn Sie NumRowsAffected aktivieren und die SQL-Abfrage enthält keine INSERT-, UPDATE- oder DELETE-Anweisung, ist der NumRowsAffected-Wert in jeder Ausgabezeile null.

Wenn die SQL-Abfrage mehrere Anweisungen enthält, gibt der Datenintegrationsdienst die NumRowsAffected-Werte für jede Anweisung zurück. NumRowsAffected enthält die Summe der Zeilen, die die Anweisungen INSERT, UPDATE und DELETE für jede Eingabezeile ändern.

Eie Abfrage enthält beispielsweise die folgenden Anweisungen:

```
DELETE from Employees WHERE Employee_ID = ‘101’;
SELECT Employee_ID, LastName from Employees WHERE Employee_ID = ‘103’;
INSERT into Employees (Employee_ID, LastName, Address)VALUES (‘102’, 'Gein', '38 Beach
Rd')
```

Die DELETE-Anweisung wirkt sich auf eine Zeile aus. Der SELECT-Anweisung hat keine Auswirkungen auf irgendeine Zeile. Die INSERT-Statement wirkt sich auf eine Zeile aus.

Der Datenintegrationsdienst gibt eine Zeile aus der DELETE-Anweisung zurück. NumRowsAffected ist gleich eins. Der Datenintegrationsdienst gibt eine Zeile aus der SELECT-Anweisung zurück, NumRowsAffected ist null. Der Datenintegrationsdienst gibt eine Zeile aus der INSERT-Anweisung zurück, NumRowsAffected ist gleich null.

SQL-Umwandlung – Erweiterte Eigenschaften

Sie können die Eigenschaften der SQL-Umwandlung jederzeit ändern. Der Standarddatenbanktyp ist Oracle. Wenn es sich bei der Datenbank, zu der Sie eine Verbindung herstellen müssen, um einen anderen Datenbanktyp handelt, ändern Sie den Datenbanktyp, bevor Sie der Umwandlung Ports hinzufügen.

Konfigurieren Sie die folgenden Eigenschaften auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“. Wenn Sie die die Tracingebene der SQL-Umwandlung für Verbose-Daten konfigurieren, schreibt der Datenintegrationsdienst jede SQL-Abfrage, die er vorbereitet, in das Mapping-Protokoll.

Verbindungstyp

Beschreibt, wie der Datenintegrationsdienst eine Verbindung zur Datenbank herstellt. Der Verbindungstyp ist statisch. Der Datenintegrationsdienst stellt eine einmalige Verbindung zur Datenbank her. Wählen Sie ein Datenbankverbindungsobjekt in der SQL-Umwandlung aus. Schreibgeschützt

DB-Typ

Der Datenbanktyp, zu dem die SQL-Umwandlung eine Verbindung herstellt. Wählen Sie einen Datenbanktyp aus der Liste aus. Zur Auswahl stehen Oracle, Microsoft SQL Server, IBM DB2 oder ODBC. Der Datenbanktyp wirkt sich auf die Datentypen aus, die Sie auf der Registerkarte **Ports** zuweisen können. Wenn Sie den Datenbanktyp ändern, ändert das Developer tool die Portdatentypen für Eingabe-, Ausgabe- und Pass-Through-Ports.

Bei einem Fehler in einer Zeile fortfahren

Fährt mit den verbleibenden SQL-Anweisungen in einer Abfrage fort, nachdem ein Fehler aufgetreten ist.

Statistiken als Ausgabe einschließen

Fügt einen NumRowsAffected-Ausgabeport hinzu. Der Port gibt die Gesamtanzahl der Datenbankzeilen zurück, die von den Abfrageanweisungen INSERT, DELETE und UPDATE für eine Eingabezeile aktualisiert werden.

Max. Ausgabe des Zeilenzählers

Definiert die maximale Anzahl von Zeilen, die die SQL-Umwandlung aus einer SELECT-Abfrage ausgeben kann. Zum Konfigurieren unbeschränkter Zeilen, legen Sie den Wert für Max. Ausgabe des Zeilenzählers auf Null fest.

Abfragebeschreibung

Beschreibung der SQL-Abfrage, die Sie in der Umwandlung definieren.

SQL-Modus

Bestimmt, ob die SQL-Abfrage ein externes Skript oder die Abfrage in der Umwandlung definiert ist. Der SQL-Modus ist „Abfrage“. Die SQL-Umwandlung führt eine Abfrage aus, die Sie im SQL-Editor definieren. Schreibgeschützt

SQL-Abfrage

Zeigt die SQL-Abfrage, die Sie im SQL-Editor konfigurieren.

Hat Nebeneffekte

Gibt an, dass die SQL-Umwandlung alle Funktionen mit Ausnahme der Rückgabe von Zeilen durchführt. Die SQL-Umwandlung hat einen Nebeneffekt, wenn die SQL-Abfrage eine Datenbank aktualisiert. Aktivieren Sie **Hat Nebeneffekte**, wenn die SQL-Abfrage eine Anweisung enthält, wie beispielsweise CREATE, DROP, INSERT, UPDATE, GRANT oder REVOKE.

Die SQL-Umwandlung hat auch einen Nebeneffekt, wenn die Umwandlung Leerzeilen für SELECT-Anweisungen zurückgibt, die keine Ergebnisse liefern. Die Zeilen können unter Umständen Pass-Through-Portwerte, SQL-Fehlerinformationen oder das NUMRowsAffected-Feld enthalten.

Deaktivieren Sie die Eigenschaft **Hat Nebeneffekte**, um Push-Into- oder "Early Selection"-Optimierung zuzulassen. Sie ist standardmäßig aktiviert.

Nur Datenbankausgabe zurückgeben

Die SQL-Umwandlung erzeugt weder Zeilen für SELECT-Anweisungen, die Leerergebnisse zurückgeben, noch Zeilen für andere Anweisungen wie INSERT, UPDATE, DELETE oder COMMIT oder Nullzeilen.

Push-Into-Optimierung aktivieren

Ermöglicht dem Datenintegrationsdienst das Verschieben der Logik aus einer Filterumwandlung im Mapping in die SQL in der SQL-Umwandlung.

Beibehalten der Zeilenreihenfolge

Beibehalten der Zeilenreihenfolge der Eingabedaten für die Umwandlung. Wählen Sie diese Option aus, wenn der Datenintegrationsdienst keine Optimierung durchführen soll, die die Zeilenreihenfolge verändern könnte.

Wenn der Datenintegrationsdienst Optimierungen durchführt, geht eine zu einem vorherigen Zeitpunkt im Mapping erstellte Reihenfolge möglicherweise verloren. Die Reihenfolge in einem Mapping können Sie mit einer Einfachdateiquelle, einer sortierten relationalen Quelle oder einer Sortierer-Umwandlung erstellen. Wenn Sie eine Umwandlung konfigurieren, um die Zeilenreihenfolge beizubehalten, berücksichtigt der Datenintegrationsdienst diese Konfiguration beim Durchführen von Optimierungen für das Mapping. Der Datenintegrationsdienst führt Optimierungen für die Umwandlung durch, wenn er die Reihenfolge beibehalten kann. Der Datenintegrationsdienst führt keine Optimierungen für die Umwandlung durch, wenn die Optimierungen die Zeilenreihenfolge verändern würden.

Partitionierbar

Die Umwandlung kann mit mehreren Threads verarbeitet werden. Löschen Sie diese Option, wenn der Datenintegrationsdienst einen Thread verwenden soll, um die Umwandlung zu verarbeiten. Der Datenintegrationsdienst kann mehrere Threads verwenden, um die verbleibenden Mapping-Pipeline-Stages zu verarbeiten.

Deaktivieren Sie die Partitionierung für eine SQL-Umwandlung, wenn es für die SQL-Abfragen erforderlich ist, dass die Umwandlung mit einem Thread verarbeitet wird. Sie können die Partitionierung für eine SQL-Umwandlung auch deaktivieren, wenn nur eine Verbindung zur Datenbank hergestellt werden soll.

SQL-Umwandlungsabfrage

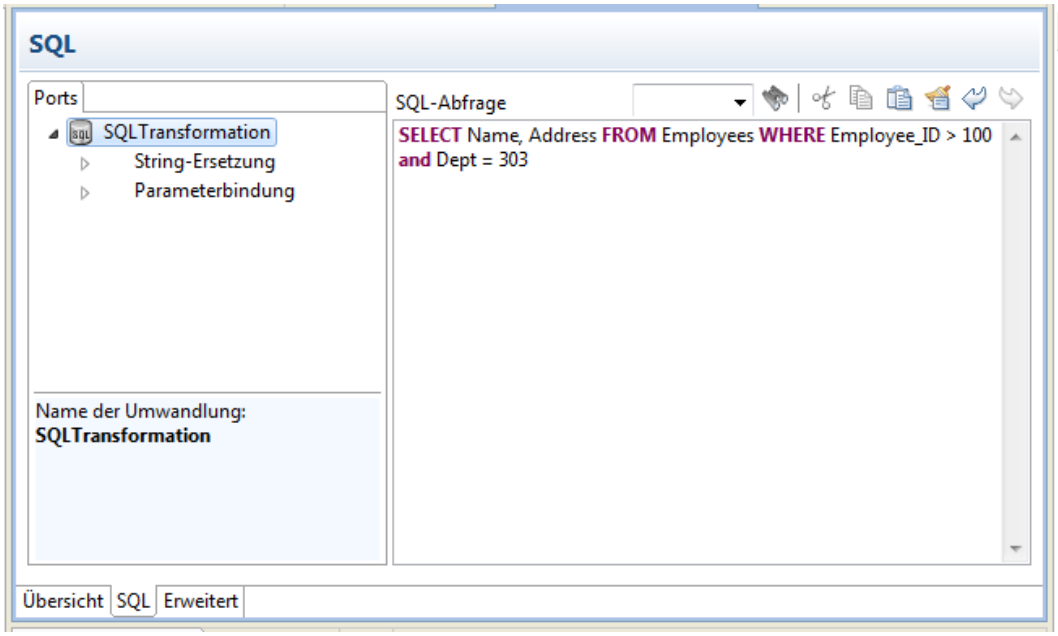
Erstellen Sie eine SQL-Abfrage im SQL-Editor, um Zeilen aus einer Datenbank abzurufen oder die Datenbank zu aktualisieren.

Geben Sie zum Erstellen einer Abfrage die Abfrageanweisung im SQL-Editor in der SQL-Ansicht ein. Der SQL-Editor stellt eine Liste der Umwandlungsports bereit, auf die Sie in der Abfrage verweisen können. Sie können auf einen Portnamen doppelklicken, um ihn als Abfrageparameter hinzuzufügen.

Wenn Sie eine Abfrage erstellen, validiert der SQL-Editor die Portnamen in der Abfrage. Außerdem wird sichergestellt, dass es sich bei den für die Stringersetzung verwendeten Ports um Stringdatentypen handelt. Der SQL-Editor validiert nicht die Syntax der SQL-Abfrage.

Sie können Konstanten in der SQL-Abfrage verwenden. Schließen Sie jede Zeichenfolge in ein einzelnes Anführungszeichen (') ein.

Die folgende Abbildung zeigt eine beispielhafte SQL-Abfrage:



Sie können eine statische SQL-Abfrage erstellen. Die Abfrageanweisung ändert sich nicht. Sie können aber Parameter zum Ändern von Werten aufnehmen. Der Datenintegrationsdienst führt die Abfrage für jede Eingabezeile aus.

Definieren der SQL-Abfrage

Definieren Sie eine SQL-Abfrage, die dieselben Abfrageanweisungen wie jede Eingabezeile ausführt. Sie können die Abfragespalten oder die Tabelle basierend auf den Eingabeportwerten in der Zeile ändern. Sie können ebenfalls die Werte in der WHERE-Klausel basierend auf den Eingabeportwerten ändern.

Um die Datenwerte in der WHERE-Klausel für jede Eingabezeile zu ändern, konfigurieren Sie die Parameterbindung.

Um die Abfragespalten oder die Tabelle basierend auf den Eingabeportwerten zu ändern, verwenden Sie die Stringersetzung.

Parameterbindung

Um die Daten in der Abfrage zu ändern, konfigurieren Sie die Abfrageparameter und binden Sie sie an Eingabeports in der Umwandlung. Wenn Sie einen Parameter auf einen Eingabeport binden, geben Sie den Port mit Namen in der Abfrage ein. Der SQL-Editor schließt den Portnamen in Fragezeichen (?) ein. Die Abfragedaten andern sich basierend auf dem Wert der Daten im Port.

Die folgenden Abfragen verwenden Parameterbindung:

```
DELETE FROM Employee WHERE Dept = ?Dept?
INSERT INTO Employee(Employee_ID, Dept) VALUES (?Employee_ID?, ?Dept?)
UPDATE Employee SET Dept = ?Dept? WHERE Employee_ID > 100
```

Die folgende SQL-Abfrage hat Abfrageparameter, die an die Eingabeports Employee_ID und Dept einer SQL-Umwandlung gebunden werden:

```
SELECT Name, Address FROM Employees WHERE Employee_Num =?Employee_ID? and Dept = ?Dept?
```

Die Quelle weißt möglicherweise die folgenden Zeilen auf:

Employee_ID	Dept
100	Products
123	HR
130	Accounting

Der Data Integration Service generiert die folgenden Abfrageanweisungen aus den Zeilen:

```
SELECT Name, Address FROM Employees WHERE Employee_ID = '100' and DEPT = 'Products'
SELECT Name, Address FROM Employees WHERE Employee_ID = '123' and DEPT = 'HR'
SELECT Name, Address FROM Employees WHERE Employee_ID = '130' and DEPT = 'Accounting'
```

Stringersetzung

Verwenden Sie Stringvariablen zum Ersetzen von Komponenten der Abfrageanweisungen. Beispielsweise können Sie mit der Stringvariablen den Tabellennamen in einer Abfrage ersetzen. Sie können auch die Spaltennamen in einer SELECT-Anweisung ersetzen.

Zum Ersetzen des Tabellennamens konfigurieren Sie einen Eingabeport so, dass er den Tabellenname von jede Eingabezeile empfängt. Wählen Sie im SQL-Editor den Port aus der Portliste **Stringersetzung** aus. Das Developer Tool kennzeichnet den Eingabeport nach Namen in der Abfrage und schließt den Namen in Tilden (~) ein.

Die folgende Abfrage enthält eine Stringvariable ~Tabelle_Port~:

```
SELECT Emp_ID, Address from ~Table_Port~ where Dept = 'HR'
```

Die Quelle kann folgenden Werte in die Spalte **Table_Port** eingeben:

Table_Port
Employees_USA
Employees_England
Employees_Australia

Der Data Integration Service ersetzt die Variable ~Table_Port~ mit dem Tabellennamenwert im Eingabeport:

```
SELECT Emp_ID, Address from Employees_USA where Dept = 'HR'
SELECT Emp_ID, Address from Employees_England where Dept = 'HR'
SELECT Emp_ID, Address from Employees_Australia where Dept = 'HR'
```

Kardinalität zwischen Eingabezeile und Ausgabezeile

Wenn der Data Integration Service eine SELECT-Abfrage ausführt, gibt die SQL-Umwandlung eine Zeile für jede Zeile zurück, die sie abruft. Wenn bei der Abfrage keine Daten abgerufen werden, gibt die SQL-Umwandlung keine oder eine Zeile für jede Eingabezeile zurück.

Verarbeitung von Abfrageanweisungen

Wenn eine SELECT-Abfrage erfolgreich ist, kann die SQL-Umwandlung mehrere Zeilen abgerufen. Wenn die Abfrage andere Anweisungen enthält, kann der Data Integration Service möglicherweise eine Zeile mit SQL-Fehlern oder der Anzahl der betroffenen Zeilen generieren.

Port-Konfiguration

Der NumRowsAffected-Ausgabeport enthält die Anzahl der Zeilen, die die Anweisung INSERT, UPDATE oder DELETE für eine Eingabezeile ändert. Die SQL-Umwandlung gibt die Anzahl der für jede Anweisung in einer Abfrage betroffenen Zeilen zurück. Wenn die SQL-Umwandlung Pass-Through-Ports enthält, gibt die Umwandlung die Spaltendaten mindestens ein Mal für jede Quellzeile zurück.

Konfiguration der max. Anzahl an Zeilen

Mit der Option "Max. Ausgabe des Zeilenzählers" wird die Anzahl der Zeilen beschränkt, die die SQL-Umwandlung aus SELECT-Abfragen zurückgibt.

Fehlerzeilen

Der Data Integration Service gibt Zeilenfehler zurück, wenn er Verbindungs- oder Syntaxfehler findet. Die SQL-Umwandlung gibt Fehler in den SQLError-Port zurück.

Fortfahren bei SQL-Fehler

Sie können die SQL-Umwandlung konfigurieren, um die Verarbeitung trotz eines Fehlers in der SQL-Anweisung fortzusetzen. Die SQL-Umwandlung generiert keinen Zeilenfehler.

Verarbeitung von Abfrageanweisungen

Der Typ der SQL-Abfrage bestimmt, wie viele Zeilen die SQL-Umwandlung zurückgibt. Die SQL-Umwandlung kann keine, eine oder mehrere Zeilen zurückgeben. Wenn die Abfrage eine SELECT-Anweisung enthält, gibt die SQL-Umwandlung jede Spalte aus der Datenbank an einen Ausgabeport zurück. Die Umwandlung gibt alle qualifizierenden Zeilen zurück.

In der folgenden Tabelle werden die Ausgabezeilen aufgelistet, die die SQL-Umwandlung für verschiedene Arten von Abfrageanweisungen erstellt, wenn im Abfragemodus keine Fehler auftreten:

Abfrageanweisung	Ausgabezeilen
Nur UPDATE, INSERT, DELETE	Eine Zeile für jede Anweisung in der Abfrage.
Eine oder mehrere SELECT-Anweisungen	Gesamtzahl der abgerufenen Datenbankzeilen.
DDL-Abfragen wie CREATE, DROP, TRUNCATE	Eine Zeile für jede Anweisung in der Abfrage.

Port-Konfiguration

Wenn Sie "Statistiken als Ausgabe einschließen" aktivieren, erstellt das Developer Tool den NumRowsAffected-Port. Der Data Integration Service gibt mindestens eine Zeile mit den NumRowsAffected basierend auf den Anweisungen in der SQL-Abfrage zurück.

In der folgenden Tabelle sind die Ausgabezeilen aufgelistet, die die SQL-Umwandlung generiert, wenn Sie NumRowsAffected aktivieren:

Abfrageanweisung	Ausgabezeilen
Nur UPDATE, INSERT, DELETE	Eine Zeile für jede Anweisung mit den NumRowsAffected für die Anweisung.
Eine oder mehrere SELECT-Anweisungen	Gesamtzahl der abgerufenen Datenbankzeilen. NumRowsAffected ist Null in jeder Zeile.
DDL-Abfragen wie CREATE, DROP, TRUNCATE	Eine Zeile mit null NumRowsAffected.

Maximale Ausgabe des Zeilenzählers

Sie können die Anzahl der Zeilen begrenzen, die die SQL-Umwandlung für SELECT-Abfragen zurückgibt. Konfigurieren Sie die Eigenschaft **Max. Ausgabe des Zeilenzählers**, um die Anzahl der Zeilen zu begrenzen. Wenn eine Abfrage mehrere SELECT-Anweisungen enthält, begrenzt die SQL-Umwandlung Gesamtzeilen aus allen SELECT-Anweisungen.

Sie legen beispielsweise **Max. Ausgabe des Zeilenzählers** auf 100 fest. Die Abfrage enthält zwei SELECT-Anweisungen:

```
SELECT * FROM table1; SELECT * FROM table2;
```

Wenn die erste SELECT-Anweisung 200 Zeilen und die zweite SELECT-Anweisung 50 Zeilen zurückgibt, gibt die SQL-Umwandlung 100 Zeilen aus der ersten SELECT-Anweisung zurück. Die SQL-Umwandlung gibt keine Zeilen aus der zweiten Anweisung zurück.

Zum Konfigurieren unbeschränkter Zeilen, legen Sie den Wert für **Max. Ausgabe des Zeilenzählers** auf Null fest.

Fehlerzeilen

Der Data Integration Service gibt Fehlerzeilen zurück, wenn er einen Verbindungs- oder Syntaxfehler entdeckt. Die SQL-Umwandlung gibt SQL-Fehler an den SQLError-Port zurück.

Wenn Sie einen Pass-Through-Port oder den NumRowsAffected-Port konfigurieren, gibt die SQL-Umwandlung mindestens eine Zeile für jede Quellzeile zurück. Wenn eine Abfrage keine Daten zurückgibt, gibt die SQL-Umwandlung die Pass-Through-Daten und die NumRowsAffected-Werte zurück. Die Nullwerte werden jedoch in den Ausgabeports zurückgegeben. Sie können Zeilen mit Nullwerten entfernen, indem Sie die Ausgabezeilen über eine Filterumwandlung weitergeben.

In der folgenden Tabelle werden die Zeilen beschrieben, die die SQL-Umwandlung für die Abfrageanweisungen UPDATE, INSERT oder DELETE generiert.

NumRowsAffected-Port oder Pass-Through-Port konfiguriert	SQLError	Zeilenausgabe
Kein Port konfiguriert	Nein	Eine Zeile mit NULL im SQLError-Port.
Kein Port konfiguriert	Ja	Eine Zeile mit dem Fehler im SQLError-Port.
Beide Ports konfiguriert	Nein	Eine Zeile für jede Abfrageanweisung mit den NumRowsAffected- oder den Pass-Through-Spaltendaten.
Beide Ports konfiguriert	Ja	Eine Zeile mit dem Fehler im SQLError-Port, im NumRowsAffected-Port oder in den Pass-Through-Daten.

In der folgenden Tabelle wird die Anzahl der Ausgabezeilen beschrieben, die die SQL-Umwandlung für SELECT-Anweisungen generiert:

NumRowsAffected-Port oder Pass-Through-Port konfiguriert	SQLError	Zeilenausgabe
Kein Port konfiguriert	Nein	Eine oder mehrere Zeile, basierend auf der Zeilen, die von jeder SELECT-Anweisung zurückgegeben werden.
Kein Port konfiguriert	Ja	Eine Zeile größer als die Summe der Ausgabezeilen für die erfolgreichen Anweisungen. Die letzte Zeile enthält den Fehler im SQLError-Port.
Beide Ports konfiguriert	Nein	Eine oder mehrere Zeilen basierend auf den für jede SELECT-Anweisung zurückgegebenen Zeilen: - Wenn NumRowsAffected aktiviert ist, enthält jede Zeile eine NumRowsAffected-Spalte mit dem Wert NULL. - Wenn ein Pass-Through-Port konfiguriert ist, enthält jede Zeile die Pass-Through-Spaltendaten. Wenn die Abfrage mehrere Zeilen zurückgibt, werden die Pass-Through-Daten in jeder Zeile dupliziert.
Beide Ports konfiguriert	Ja	Eine oder mehrere Zeilen basierend auf den für jede SELECT-Anweisung zurückgegebenen Zeilen. Die letzte Zeile enthält die Fehler im SQLError-Port: - Wenn NumRowsAffected aktiviert ist, enthält jede Zeile eine NumRowsAffected-Spalte mit dem Wert NULL. - Wenn ein Pass-Through-Port konfiguriert ist, enthält jede Zeile die Pass-Through-Spaltendaten. Wenn die Abfrage mehrere Zeilen zurückgibt, werden die Pass-Through-Daten in jeder Zeile dupliziert.

In der folgenden Tabelle wird die Anzahl der Ausgabezeilen beschrieben, die die SQL-Umwandlung für DDL-Abfragen wie CREATE, DROP oder TRUNCATE generiert:

NumRowsAffected-Port oder Pass-Through-Port konfiguriert	SQLError	Zeilenausgabe
Kein Port konfiguriert	Nein	Eine Zeile mit NULL im SQLError-Port.
Kein Port konfiguriert	Ja	Eine Zeile, die den Fehler im SQLError-Port enthält.

NumRowsAffected-Port oder Pass-Through-Port konfiguriert	SQLError	Zeilenausgabe
Beide Ports konfiguriert	Nein	Eine Zeile, die die NumRowsAffected-Spalte mit dem Wert NULL und die Pass-Through-Spaltendaten enthält.
Beide Ports konfiguriert	Ja	Eine Zeile mit dem Fehler im SQLError-Port, der NumRowsAffected-Spalte mit dem Wert NULL und den Pass-Through-Spaltendaten.

Fortfahren bei SQL-Fehler

Sie einen SQL-Fehler, der in einer Abfrageanweisung auftritt, ignorieren. Aktivieren Sie die Option **Bei einem Fehler in einer Zeile fortfahren**. Der Data Integration Service fährt fort, den Rest der SQL-Anweisungen für die Zeile auszuführen.

Der Data Integration Service generiert keinen Zeilenfehler. Der SQLError-Port enthält jedoch die fehlgeschlagene SQL-Anweisung und die Fehlermeldungen.

Eine Abfrage enthält möglicherweise die folgenden Anweisungen:

```
DELETE FROM Persons WHERE FirstName = ‘Ed’;
INSERT INTO Persons (LastName, Address) VALUES ('Gein', '38 Beach Rd')
```

Wenn die DELETE-Anweisung fehlschlägt, gibt die SQL-Umwandlung gibt eine Fehlermeldung aus der Datenbank zurück. Der Data Integration Service führt die Verarbeitung der INSERT-Anweisung fort.

Deaktivieren Sie die Option **Bei SQL-Fehler fortfahren**, um Datenbankfehler zu beheben und die Fehler den Abfrageanweisungen zuzuordnen, auf die die Fehler zurückzuführen sind.

Filteroptimierung mit der SQL-Umwandlung

Der Datenintegrationsdienst kann die Filteroptimierung mit einer SQL-Umwandlung anwenden, wenn die Filterbedingung ausschließlich auf Übergabeports verweist und die SQL-Umwandlung keine Nebeneffekte aufweist.

Die SQL-Umwandlung weist unter folgenden Umständen Nebeneffekte auf:

- Die SQL-Abfrage aktualisiert eine Datenbank. Die SQL-Abfrage enthält eine Anweisung, wie beispielsweise CREATE, DROP, INSERT, UPDATE, GRANT oder REVOKE.
- Die Umwandlung gibt Leerzeilen für SELECT-Anweisungen zurück, die keine Ergebnisse liefern. Die Zeilen können unter Umständen Übergabeportwerte, SQL-Fehlerinformationen oder das NUMRowsAffected-Feld enthalten.

Der Datenintegrationsdienst kann die frühzeitige Auswahl und Push-into-Optimierungsmethoden mit der SQL-Umwandlung anwenden.

"Early Selection"-Optimierung mit der SQL-Umwandlung

Der Data Integration Service kann "Early Selection"-Optimierung mit einer SQL-Umwandlung durchführen, wenn die Filterbedingung ausschließlich auf Pass-Through-Ports verweist und die SQL-Umwandlung keine Nebeneffekte aufweist.

Die SQL-Umwandlung weist unter folgenden Umständen Nebeneffekte auf:

- Die SQL-Abfrage aktualisiert eine Datenbank. Die SQL-Abfrage enthält eine Anweisung, wie beispielsweise CREATE, DROP, INSERT, UPDATE, GRANT oder REVOKE.
- Die Umwandlung gibt Leerzeilen für SELECT-Anweisungen zurück, die keine Ergebnisse liefern. Die Zeilen können unter Umständen Pass-Through-Portwerte, SQL-Fehlerinformationen oder das NUMRowsAffected-Feld enthalten.

Aktivieren der frühen Auswahloptimierung mit der SQL-Umwandlung

Aktivieren Sie die frühe Auswahloptimierung in der SQL-Umwandlung, wenn die SQL-Umwandlung keine Nebeneffekte aufweist.

1. Aktivieren Sie die Option **Nur Datenbank-Ausgabe zurückgeben** unter **Erweiterte Eigenschaften** in der SQL-Umwandlung
2. Löschen Sie **Hat Nebeneffekte** in der Umwandlung **Erweiterte Eigenschaften**.
3. Weist die Umwandlung einen **NumAffectedRows**-Port auf, entfernen Sie den Port.

Push-Into-Optimierung mit der SQL-Umwandlung

Mit der Push-Into-Optimierung verschiebt der Data Integration Service die Filterlogik aus einer Filterumwandlung im Mapping in die Abfrage in der SQL-Umwandlung.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie die Push-Into-Optimierung mit der SQL Umwandlung aktivieren:

- Die SQL-Umwandlungsabfrage darf nur „AUSWAHL“-Anweisungen beinhalten.
- Die SQL-Umwandlungsabfrage muss eine gültige Unterabfrage sein.
- Die Filterbedingung kann nicht auf die Felder "SQL-Fehler" oder "NumRowsAffected" verweisen.
- Die Namen der Ausgabeports müssen mit den Namen der Spalten in der SQL-SELECT-Anweisung übereinstimmen. Wenn Sie auf einen Ausgabeport in einer Filterbedingung verweisen, verschiebt der Data Integration Service den entsprechenden Portnamen in die SQL-Abfrage. Sie können der SQL Aliase hinzufügen, wenn die Spalten in der Abfrage nicht mit den Namen der Ausgabeports übereinstimmen. Beispiel: `SELECT mycolname1 AS portname1, mycolname2 AS portname2`.
- Die Umwandlung darf keine Nebeneffekte aufweisen.

Beispiel für eine Push-Into-Optimierung mit der SQL-Umwandlung

Eine SQL-Umwandlung ruft Aufträge nach Kunden-ID ab. Eine Filterumwandlung, die nach der SQL-Umwandlung erscheint, gibt nur die Zeilen zurück, bei denen die Bestellmenge mehr als 1000 beträgt.

Der Data Integration Service fügt den folgenden Filter in eine SELECT-Anweisung in der SQL-Umwandlung ein:

```
orderAmount > 1000
```

Jede Anweisung in der SQL-Abfrage wird zu einer separaten Unterabfrage der SELECT-Anweisung, die den Filter enthält.

Die folgende Abfrageanweisung zeigt die ursprüngliche Abfrageanweisung als Unterabfrage in der SELECT-Anweisung an:

```
SELECT <customerID>, <orderAmount>, … FROM (original query statements) ALIAS WHERE
<orderAmount> > 1000
```

Wenn die SQL-Abfrage mehrere Anweisungen enthält, wird jede Anweisung in eine separate Unterabfrage eingeschlossen. Der Unterabfrage hat dieselbe Syntax, einschließlich der WHERE-Klausel.

Die Ports *customerID* und *orderAmount* sind die Namen der Ausgabeports in der SQL-Umwandlung. Die Unterabfrage enthält weder Pass-Through-Ports noch den SQL-Fehler noch die statistischen SQL-Ports. Wenn Sie mehrere Filter in die SQL-Umwandlung einfügen, enthält die WHERE-Klausel alle Filter.

Aktivieren der Push-Into-Optimierung mit der SQL-Umwandlung

Aktivieren Sie die Push-Into-Optimierung, indem Sie Eigenschaften auf der Registerkarte **Erweiterte Eigenschaften** der SQL-Umwandlung konfigurieren.

1. Löschen Sie **Hat Nebeneffekte**.
2. Aktivieren Sie **Nur Datenbank-Ausgabe zurückgeben**.
3. Setzen Sie·**Max. Ausgabe des Zeilenzählers** auf Null.
4. Aktivieren Sie die Push-Into-Optimierung.

SQL-Umwandlung - Beispiel mit einer SQL-Abfrage

Sie sind ein Entwickler in der Personalabteilung von HypoStores Corporation. HypoStores verwaltet Gehaltsabrechnungsinformationen der Mitarbeiter in einer separaten Datenbank aus den Mitarbeiterdaten der Personalabteilung. Die Personalabteilung muss eine einzelne Ansicht mit Mitarbeitern und Gehältern aus verschiedenen Regionen abfragen.

Sie möchten ein Mapping logischer Datenobjekt erstellen, die eine einzelne Ansicht der Mitarbeiter- und Gehaltsdaten in einem logischen Datenobjekt für Mitarbeiter anzeigt.

Erstellen Sie ein Mapping logischer Datenobjekte mit der Mitarbeiterdatenquelle. Verwenden Sie eine SQL-Umwandlung, um die Gehalts- und Einstellungsdaten aus der Gehaltsabrechnungsdatenbank abzurufen.

Mapping logischer Datenobjekte

Das Mapping logischer Datenobjekte enthält die folgenden Objekte:

Mitarbeitertabelle

Geben Sie Mitarbeiterdaten aus der relationalen Tabelle der Mitarbeiterdatenbank ein.

Gehaltstabelle

Eine Tabelle in der Gehaltsabrechnungsdatenbank, die das Mitarbeitergehalt und Einstellungsdatum enthält. Die Datenbank ist eine Oracle-Datenbank.

SQL-Umwandlung

Die Umwandlung, die das Einstellungsdatum und das Gehalt für jede Mitarbeiterzeile abruft. Die Umwandlung stellt eine Verbindung zur Gehaltsabrechnungsdatenbank her und führt eine SQL-Abfrage der Gehaltstabelle in der Datenbank aus..

Logisches Datenobjekt

Enthält die kombinierte Ansicht der Mitarbeiter und der Gehaltsdaten. Das logische Datenobjekt empfängt die Ausgabe aus der SQL-Umwandlung.

SQLErrors-Datei

Die SQLErrors-Datei ist eine Einfachdatei, die alle SQL-Fehler aus der Datenbank enthält. Der Data Integration Service schreibt mindestens eine Zeile in die SQLErrors-Datei für jede Eingabezeile. Wenn keine SQL-Fehler auftreten, enthält die SQLError-Spalte NULL. Überprüfen Sie die SQLErrors-Datei zur Fehlerbehebung.

Gehaltstabelle

Die Gehaltstabelle einer relationalen Tabelle in der Gehaltsabrechnungsdatenbank. Die Tabelle enthält Mitarbeiterdaten, die von der Gehaltsabteilung verwaltet werden. Die SQL-Umwandlung ruft das Einstellungsdatum und das Mitarbeitergehalt aus der Gehaltstabelle ab.

Die folgende Tabelle zeigt einige Zeilen aus der Gehaltstabelle:

Employee_Num	HireDate	Gehalt
10	3-May-97	232000
11	11-Sep-01	444000
12	17-Oct-89	656000
13	13-Aug-07	332100

Mitarbeitertabelle

Die Quelle ist die Mitarbeitertabelle aus der Mitarbeiterdatenbank.

Die folgende Tabelle zeigt Beispielzeilen aus der Mitarbeitertabelle:

EmpID	LastName	FirstName	DeptId	Telefon
10	Smith	Martha	FIN	(415) 552-1623
11	Jones	Cynthia	ENG	(415) 552-1744
12	Russell	Cissy	SLS	(415) 552-1656
13	Goyal	Girish	FIN	(415) 552-1656

SQL-Umwandlung

Die SQL-Umwandlung ruft das Einstellungsdatum und das Gehalt des Mitarbeiters aus der Gehaltstabelle der Gehaltsabrechnungsdatenbank ab. Die Gehaltstabelle befindet sich in einer Oracle-Datenbank.

Gehen Sie zum Konfigurieren der SQL-Umwandlung folgendermaßen vor:

1. Konfigurieren Sie die Eigenschaften für die SQL-Umwandlung.
2. Definieren Sie die Ports.

3. Erstellen Sie die SQL-Abfrage.
4. Konfigurieren Sie die Datenbankverbindung für die SQL-Umwandlung.

Definieren von SQL-Umwandlungseigenschaften

Konfigurieren Sie die SQL-Umwandlungseigenschaften in der Ansicht **Erweiterte Eigenschaften**.

Konfigurieren Sie die folgenden Eigenschaften:

Datenbanktyp

Der Datenbanktyp ist Oracle. Wenn Sie die Ports definieren, können Sie Portdatentypen auswählen, die für Oracle gelten.

Bei einem Fehler in einer Zeile fortfahren

Disable. Stoppen Sie die Verarbeitung, wenn ein SQL-Fehler in der Zeile auftritt.

Statistiken als Ausgabe einschließen

Disable. Erstellen Sie nicht den NumRowsAffected-Ausgabeport.

Definieren der Ports

Definieren Sie Eingabeports für jede Spalte in der Mitarbeiterquelltabelle. Wählen Sie **In Ausgabe kopieren** aus, um die Eingabeports in Pass-Through-Ports für die Spalten zu ändern. Wenn Sie **In Ausgabe kopieren** auswählen, erstellt das Developer Tool den entsprechenden Ausgabeport für jeden zu kopierenden Port.

Erstellen Sie die folgenden Eingabe-Pass-Through-Ports:

Name	Typ	Nativer Typ	Präzision	Größenordnung	In Ausgabe kopieren
EmpID	decimal	number(p,2)	4	0	x
LastName	string	varchar2	30	0	x
FirstName	string	varchar2	20	0	x
DeptID	string	varchar2	4	0	x
Telefon	string	varchar2	16	0	x

Die SQL-Umwandlung hat die folgenden Ausgabeports:

Name	Typ	Nativer Typ	Präzision	Größenordnung
EmpID	decimal	number(p,s)	4	0
LastName	string	varchar2	30	0
FirstName	string	varchar2	20	0
DeptID	string	varchar2	4	0
Telefon	string	varchar2	16	0

Name	Typ	Nativer Typ	Präzision	Größenordnung
HireDate	date/time	timestamp	29	0
Gehalt	decimal	number(p,s)	8	2

Das Developer Tool fügt den "_output"-Suffix zu jedem Ausgabeport hinzu, den es bei Auswahl der Option **In Ausgabe kopieren** erstellt.

Definieren Sie die Ausgabeports für das Einstellungsdatum und die Gehaltsspalten. Die SQL-Umwandlung ruft das Einstellungsdatum und die Gehaltsspalten aus der Gehaltstabelle in den Ports ab.

Definieren der SQL-Abfrage

Erstellen Sie eine SQL-Abfrage, um das Einstellungsdatum und das Gehalt für jeden Mitarbeiter aus der Gehaltstabelle abzurufen.

Definieren Sie die Abfrage in der SQL-Ansicht der SQL-Umwandlung.

Geben Sie die folgende Abfrage im SQL-Editor ein:

```
select HIREDATE,SALARY,from Salary where EMPLOYEE_NUM =?EmpID?
```

Hiredate, Salary und Employee_Num sind Spaltennamen in der Gehaltstabelle.

?EMPID? ist ein Parameter, der den Werte des EmpID-Ports enthält.

Definieren der Datenbankverbindung

Wählen Sie in der Ansicht **Laufzeit** ein Datenbankverbindungsobjekt für die Datenbank aus, zu der die SQL-Umwandlung eine Verbindung herstellt. Wählen Sie ein Oracle-Datenbankverbindungsobjekt aus.

Ausgabe

Verbinden Sie den SQLError-Port und den EmpID_output-Ausgabeport mit der SQLErrors-Einfachdatei. Die SQLError-Port enthält Nullwerte, es sei denn, es tritt ein SQL-Fehler auf.

Verbinden Sie EmpID- und andere Ausgabeports mit dem logischen Datenobjekt.

Die SQL-Umwandlung gibt eine Zeile zurück, die Daten aus der Mitarbeitertabelle enthält und das Einstellungsdatum und das Gehalt aus der Gehaltstabelle einbezieht.

In der folgenden Tabelle werden einige Zeilen aus dem logischen Datenobjekt angezeigt:

EmpID	LastName	FirstName	DeptId	Telefon	HireDate	Gehalt
10	Smith	Martha	FIN	(415) 552-1623	19970303 00:00:00	2320.00
11	Jones	Cynthia	ENG	(415) 552-1744	20010911 00:00:00	4440.00
12	Russell	Cissy	SLS	(415) 552-1656	19891017 00:00:00	6560.00
13	Goyal	Girish	FIN	(415) 552-1660	20070813 00:00:00	3210.00

Gespeicherte Prozeduren

Sie können eine gespeicherte Prozedur aus einer SQL-Umwandlung aufrufen. Sie können eine gespeicherte Prozedur verwenden, um Aufgaben in einer relationalen Datenbank zu automatisieren. Gespeicherte Prozeduren akzeptieren benutzerdefinierte Variablen, bedingte Anweisungen und andere Funktionen, die Standard-SQL-Anweisungen nicht unterstützen.

Die SQL-Umwandlung verbindet sich mit einer relationalen Datenbank, um eine gespeicherte Prozedur auszuführen. Die SQL-Umwandlung kann gespeicherte Prozeduren aus Oracle, IBM DB2, Microsoft SQL Server, Sybase und ODBC aufrufen. Eine gespeicherte Prozedur wird in der Datenbank beibehalten und in der Datenbank ausgeführt.

Erstellen Sie eine ODBC-Verbindung, um eine gespeicherte Prozedur aus einer Sybase-Datenbank aufzurufen. Außerdem müssen Sie eine ODBC-Verbindung erstellen, um eine gespeicherte Prozedur aus einer Microsoft SQL Server-Datenbank auf Nicht-Windows-Betriebssystemen aufzurufen.

Eine gespeicherte Prozedur ist eine Sammlung von vorkompilierten Transact-SQL-, PL-SQL- oder anderen Datenbankprozedur-Anweisungen. Die Syntax für gespeicherte Prozeduren richtet sich nach der jeweiligen Datenbank.

Verwenden Sie gespeicherte Prozeduren, um die folgenden Aufgaben abzuschließen:

- Überprüfen Sie vor dem Laden von Daten in die Zieldatenbank deren Status.
- Überprüfen Sie, ob in der Datenbank genügend Speicherplatz vorhanden ist.
- Führen Sie eine spezielle Berechnung durch.
- Rufen Sie Daten anhand eines Werts ab.
- Entfernen und erstellen Sie Indizes neu.

Sie können eine gespeicherte Prozedur verwenden, um eine Abfrage oder Berechnung durchzufügen, die Sie andernfalls in eine Umwandlung einbeziehen würden. Beispiel: Wenn Sie über eine ausreichend getestete gespeicherte Prozedur zur Berechnung der Umsatzsteuer verfügen, können Sie diese Berechnung mit der gespeicherten Prozedur durchführen, anstatt die gleiche Berechnung in einer Ausdrucksumwandlung neu zu erstellen.

Eine gespeicherte Prozedur kann Eingabe akzeptieren und dann ein Ergebnissatz von Zeilen zurückgeben. Eine gespeicherte Prozedur kann DDL-Aufgaben ausführen, die keine Eingabe erfordert und anschließend keine Ausgabe zurückgibt.

Sie können die SQL-Umwandlung zur Ausführung von mehr als einer gespeicherten Prozedur ausführen. Für jede gespeicherte Prozedur, die Sie konfigurieren, konfigurieren Sie Umwandlungsports, die mit den Parametern für die gespeicherte Prozedur übereinstimmen. Jede gespeicherte Prozedur kann Daten an Ausgabeports zurückgeben.

Die Datenbank, die die gespeicherte Prozedur enthält, hat Benutzerberechtigungen. Sie müssen über Berechtigungen zum Ausführen der gespeicherten Prozedur in der Datenbank verfügen.

Hinweis: Eine gespeicherte Funktion ähnelt einer gespeicherten Prozedur mit der Ausnahme, dass die Funktion einen einzelnen Wert zurückgibt. Die SQL-Umwandlung kann gespeicherte Funktionen zurückgeben.

SQL-Umwandlungsports für gespeicherte Prozeduren

Die Eingabe- und Ausgabeports für die SQL-Umwandlung entsprechen den Eingabe- und Ausgabeparametern in der gespeicherten Prozedur.

Beim Importieren einer gespeicherten Prozedur bestimmt das Developer-Tool den Datenbanktyp aus der Datenbankverbindung. Es generiert Eingabe- und Ausgabeports in der SQL-Umwandlung aus den Parametern

in der gespeicherten Prozedur. Das Developer-Tool bestimmt den nativen Datentyp für jeden Port aus dem Parameter der gespeicherten Prozedur.

Wenn Sie die SQL-Umwandlung manuell konfigurieren, müssen die Eingabe- und Ausgabeports in der Umwandlung konfigurieren. Wenn Sie den Datenbanktyp konfigurieren, ändert die SQL-Umwandlung den nativen Datentyp für jeden Port basierend auf dem von Ihnen eingegebenen Datenbanktyp.

Die folgenden Datentypen können Daten zwischen der SQL-Umwandlung und der gespeicherten Prozedur übergeben:

Eingabe- und Ausgabeparameter

Die SQL-Umwandlung sendet Parameter an die gespeicherte Prozedur und die SQL-Umwandlung empfängt Parameter aus der gespeicherten Prozedur in den Eingabe- und Ausgabeports.

Rückgabewert

Wenn die gespeicherte Prozedur einen Rückgabewert aufweist, erstellt das Developer-Tool einen Rückgabewert-Port.

SQL-Fehler

Die SQL-Umwandlung gibt Fehler aus der gespeicherten Prozedur im SQLError-Port zurück.

Eingabe- und Ausgabeparameter

Wenn Sie eine gespeicherte Prozedur aus einer SQL-Umwandlung aufrufen, identifiziert jedes Feld, auf das die Aufrufanweisung verweist, einen Eingabe- oder Ausgabeport. Wenn Sie eine gespeicherte Prozedur importieren, generiert das Developer-Tool die Aufrufanweisung für die gespeicherte Prozedur. Andernfalls müssen Sie die Aufrufanweisung manuell konfigurieren.

Sie können die Aufrufanweisung in der Ansicht **SQL** der Umwandlung bearbeiten.

Die Aufrufanweisung hat das folgende Format:

```
?RETURN_VALUE? = call <stored proc name>(?Field1?, ?Field2?,. . . )
```

Setzen Sie die Portnamen in Fragezeichen. Die Portnamen müssen nicht mit den Parameternamen in der gespeicherten Prozedur übereinstimmen. Die Ausgabeports müssen in derselben Reihenfolge wie die Parameter in einer SELECT-Abfrage angeordnet sein.

Sie können eine gespeicherte Prozedur verwenden, die INOUT-Parameter enthält. Die SQL-Umwandlung identifiziert INOUT-Parameter durch den Namen des Eingabeports. Der Ausgabeport hat das Präfix `output_`. Der Datenintegrationsdienst bindet die Eingabe- und Ausgabeport an den gleichen Parameter.

Sie können eine SQL-Umwandlung konfigurieren, um einen Ergebnissatz zurückzugeben. Wenn die gespeicherte Prozedur einen Ergebnissatz zurückgibt, kann das Developer-Tool die Ausgabeports für die Spalten im Ergebnissatz nicht erstellen. Wenn Sie die gespeicherte Prozedur importieren, müssen Sie die Ports manuell eingeben und den gespeicherte Prozeduraufruf konfigurieren.

Rückgabewert

Einen Rückgabewert ist ein Code oder eine Textzeichenfolge, die den Status der gespeicherten Prozedur definiert. Wenn die gespeicherte Prozedur einen Rückgabewert aufweist, hat die SQL-Umwandlung einen **Rückgabewert**-Port.

Die meisten Datenbanken können einen Rückgabewert nach Ausführung einer gespeicherten Prozedur übergeben. Der Rückgabewert kann einen ganzzahligen Wert oder einen Wert enthalten, den Sie in der gespeicherten Prozedur definieren. Eine gespeicherte Prozedur kann zum Beispiel „Erfolgreich“ zurückgeben, wenn die Prozedur erfolgreich war.

Wenn eine gespeicherte Prozedur einen Ergebnissatz anstelle eines einzelnen Rückgabewerts zurückgibt, empfängt die SQL-Umwandlung den ersten Rückgabewert aus der Prozedur.

Ergebnissätze aus gespeicherten Prozeduren

Sie können eine SQL-Umwandlung konfigurieren, um einen Ergebnissatz aus einer gespeicherten Prozedur zu erhalten. Eine gespeicherte Prozedur gibt mehrere Zeilen in einem Ergebnissatz zurück. Die SQL-Umwandlung kann jede Zeile an das Mapping zurückgeben.

Die folgende gespeicherte Prozedur gibt einen Ergebnissatz zurück:

```
CREATE OR REPLACE FUNCTION fetchEMPinfo
(p_State  IN VARCHAR2 )
return types.cursortype
AS
my_cursor types.cursortype;
BEGIN
OPEN my_cursor FOR SELECT EMP_ID, NAME, CITY FROM EMP WHERE STATE = p_State ORDER BY
EMP_ID;
RETURN my_cursor;
END;
```

Beim Importieren der gespeicherten Prozedur erstellt das Developer-Tool eine Anweisung für einen gespeicherten Prozeduraufruf ähnlich der folgenden Syntax:

```
call FETCHEMPINFO (?P_STATE?)
```

Der Eingabeparameter ist p_state. Das Developer-Tool erstellt die Ausgabeports nicht für Sie. Sie müssen die Ausgabeports mit denselben Datentypen wie die Parameter für gespeicherte Prozeduren erstellen.

Beispiel: Der Ergebnissatz enthält die Spalten EMP_ID, EMPNAME und CITY. Erstellen Sie Ausgabeports für diese Spalten.

Unter Verwendung der folgenden Syntax müssen Sie auch die SQL-Abfrage mit den Ausgabespalten manuell aktualisieren:

```
(?EMP_ID?,?EMPNAME?,?CITY?) = call FETCHEMPINFO (?P_STATE?)
```

Ergebnissatz mit verschiedenen Datenbanken

Konfigurieren Sie gespeicherte Prozeduren, um Ergebnissätze unter Verwendung unterschiedlicher Syntax basierend auf dem Datenbanktyp zurückzugeben.

Oracle

Eine unter Oracle gespeicherte Funktion gibt die Ergebnisse mit einem Cursor zurück

```
create or replace function sp_ListEmp return types.cursortype
as
    l_cursor    types.cursorType;
begin
    open l_cursor for select ename, empno from emp order by ename;
    return l_cursor;
end;
```

Oracle akzeptiert auch Cursor als Eingabeparameter. Sie können die Cursor nicht als Eingabeparameter mit der SQL-Umwandlung konfigurieren.

Microsoft SQL Server

Eine unter Microsoft SQL Server gespeicherte Prozedur gibt einen Ergebnissatz für die gespeicherte Prozedur mit einer SELECT-Anweisung im Prozedurtext oder mit dem explizit als Tabelle deklarierten Rückgabetyp zurück.

```
Create  PROCEDURE InOut(
@inout varchar(100) OUT
)
AS
BEGIN
set @inout = concat(@inout,'__')
select * from mytable;
END
```

IBM DB2

Eine unter IBM DB2 gespeicherte Prozedur einen Ergebnissatz mit einem offenen Cursor zurück. Die Anzahl der zurückgegebenen Ergebnissätze wird in der RESULT SET-Klausel deklariert. Die gespeicherte Prozedur öffnet einen Cursor und gibt ihn zurück. Die folgende Beispiel gibt 2 offene Cursor zurück.

```
CREATE PROCEDURE TESTMULTIRS
 (IN i_cmacct CHARACTER(5))
RESULT SETS 2
LANGUAGE SQL
BEGIN

DECLARE csnum INTEGER;

--Declare serial cursors to consume less resources
--You do not need a rollable cursor.

DECLARE getDeptNo CHAR(50); --Be careful with the estimated length.
DECLARE getDeptName CHAR(200);
DECLARE c1 CURSOR WITH RETURN FOR s1;
SET getDeptNo = 'SELECT DEPTNO FROM DEPT';
SET getDeptName = 'SELECT DEPTNAME FROM DEPT';

PREPARE s1 FROM getDeptNo;
OPEN c1;

END;
```

Sybase

Eine unter Sybase gespeicherte Prozedur gibt einen Ergebnissatz für die gespeicherte Prozedur mit einer SELECT-Anweisung im Prozedurtext oder mit dem explizit als Tabelle deklarierten Rückgabetyp zurück.

```
CREATE PROCEDURE FETCHEMPINFO
(
@p_State VARCHAR(5),
@e_id INT OUTPUT,
@e_name VARCHAR(50) OUTPUT
)
AS
BEGIN
SET NOCOUNT ON
SELECT EMP_ID, NAME FROM EMP WHERE STATE = @p_State ORDER BY EMP_ID
SET NOCOUNT OFF
SELECT @e_id AS EMP_ID, @e_name AS NAME
RETURN
END
GO

--Configure the following variables to execute the procedure.

DECLARE @p_State VARCHAR(5)
DECLARE @EMPID int
DECLARE @EMPNAME varchar(50)
```

```
SET @p_State = 'CA'
exec FETCHEMPINFO @p_State, @e_id = @EMPID, @e_name = @EMPNAME
GO
```

Ergebnissatzzeilen

Einige gespeicherte Prozeduren geben neben den Ergebnissatzzeilen Ausgabeparameter zurück. Die SQL-Umwandlung gibt Ausgabeparameter in der letzten Zeile zurück. In den Ergebnissatzzeilen werden jedoch keine einzeln auftretenden Ausgabeparameter berücksichtigt.

Beispiel: Sie schreiben eine gespeicherte Prozedur, die eine Mitarbeiter-ID empfängt und den Namen des Mitarbeiters in Ausgabeparameter 1 und die Abteilung in Ausgabeparameter 2 zurückgibt. Die gespeicherte Prozedur gibt auch eine Zeile für die Krankheitstage für den Mitarbeiter in diesem Jahr zurück. Die Zeile enthält das Datum, die Anzahl der Stunden und der Grund für die Abwesenheit.

Der Ergebnissatz enthält eine unterschiedliche Anzahl von Zeilen für jeden Mitarbeiter. Jede Zeile im Ergebnissatz enthält einen leeren Mitarbeiternamen und eine leere Abteilung. Die SQL-Umwandlung gibt den Namen des Mitarbeiters und der Abteilung nach dem Ergebnissatz zurück. Der Name des Mitarbeiters und die Abteilung werden in der letzten Zeile angezeigt.

Gespeicherte Prozedur - Beispiel

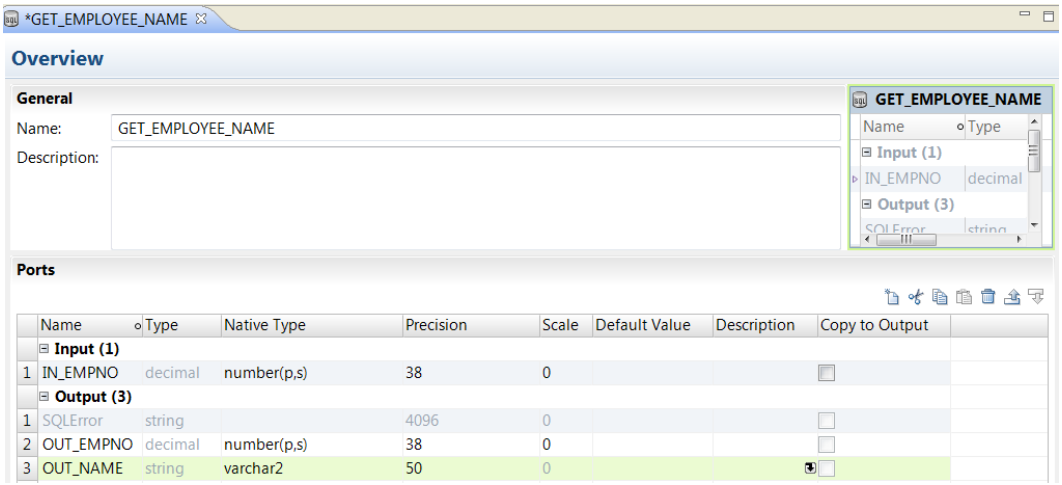
Sie können eine gespeicherte Prozedur aufrufen, die Daten an die SQL-Umwandlung zurückgibt.

Die folgende gespeicherte Prozedur empfängt eine Mitarbeiternummer und gibt eine Zeile mit der Mitarbeiternummer und den Name des Mitarbeiters zurück:

```
CREATE OR REPLACE PROCEDURE SP_GETNAME
(IN_EMPNO IN NUMBER, OUT_EMPNO NUMBER, OUT_NAME OUT STRING)
AS
BEGIN
SELECT EMP_KEY,EMP_NAME into OUT_EMPNO , OUT_NAME from EMP_TABLE where EMP_KEY=IN_EMPNO;
END;/"
```

Um die SQL-Umwandlung zu erstellen, importieren Sie die gespeicherte Prozedur. Das Developer-Tool erstellt die Eingabe- und Ausgabeports. Die Portnamen sind mit den Parameternamen in der gespeicherten Prozedur identisch.

Die folgende Abbildung zeigt die Ports für die SQL-Umwandlung:



Das Developer-Tool erstellt den folgenden gespeicherten Prozeduraufruf zum Abrufen des Mitarbeiternamens:

```
call SP_GETNAME (?IN_EMPNO?,?OUT_EMPNO?,?OUT_NAME?)
```

Sie können die gespeicherte Prozedur im SQL-Editor anzeigen. Alle SQL-Fehler werden im SQLError-Port angezeigt.

SQL-Umwandlungsverbindung

Konfigurieren Sie die SQL-Umwandlungsverbindung in den Laufzeiteigenschaften der Umwandlung. Sie müssen unter Umständen eine Laufzeitverbindung konfigurieren, wenn Sie beim Erstellen der Umwandlung keine Verbindung angegeben haben.

Sie können einen anderen Verbindungsnamen als denjenigen definieren, den Sie zum Erstellen der SQL-Umwandlung ausgewählt haben. Sie müssen eine Verbindung mit demselben Datenbanktyp wie demjenigen in den **erweiterten Eigenschaften** der SQL-Umwandlung auswählen.

Sie können einen Parameter für den Namen der SQL-Umwandlungsverbindung konfigurieren. Sie müssen den Parameter in der Ansicht **Parameter** der Zuordnung definieren, bevor Sie ihn einer Laufzeitverbindung zuweisen.

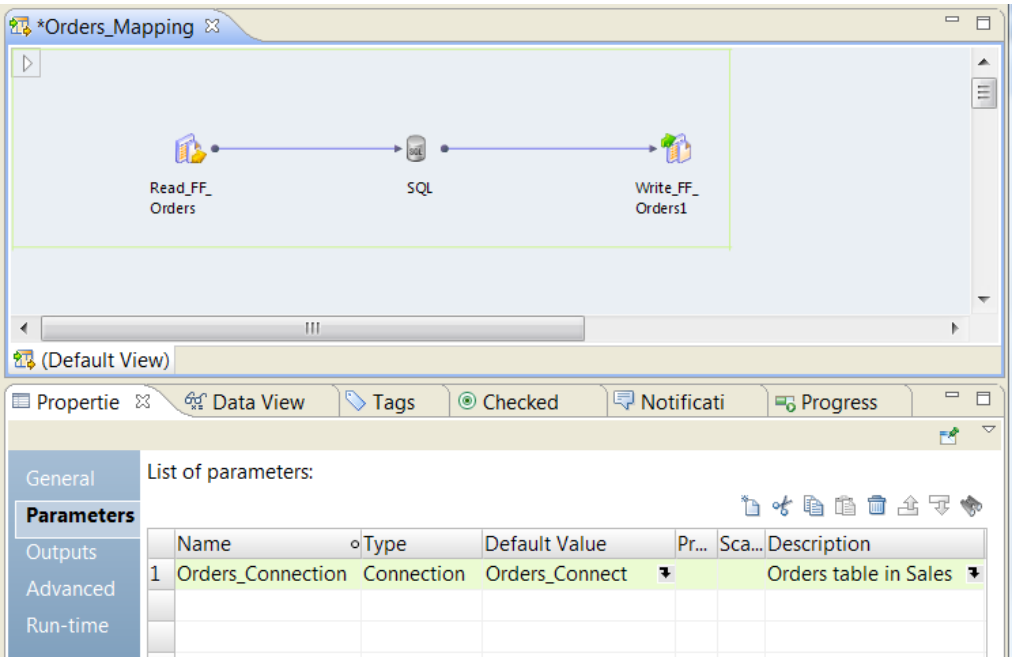
Erstellen eines Parameters für einen Verbindungsnamen

Sie können im Verbindungsnamen der Laufzeit einen benutzerdefinierten Parameter für eine SQL-Umwandlung angeben. Das Developer Tool erstellt für die Verbindung einen Mapping-Parameter anstelle eines Umwandlungsparameters.

1. Erstellen Sie ein Mapping, das die SQL-Umwandlung enthält.
2. Klicken Sie auf die Registerkarte **Laufzeit** der SQL-Umwandlung.
3. Klicken Sie bei der Option **Verbindungsname** auf den Auswahlpfeil und wählen Sie **Parameter zuweisen**.
4. Klicken Sie im Dialogfeld **Parameter zuweisen** auf **Neu**.
5. Geben Sie im Dialogfeld **Parameter** den Namen für einen Verbindungsparameter und eine Beschreibung für den Parameter ein. Der Standardwert des Parametertyps lautet Verbindung.
6. Klicken Sie im Dialogfeld „Parameter“ und im Dialogfeld **Parameter zuweisen** auf **OK**.

   Das Developer Tool erstellt einen Mapping-Parameter und weist diesen dem Verbindungsnamen zu. Der Parametername wird in den Laufzeiteigenschaften angezeigt.

7. Um eine Liste mit Mapping-Parametern anzuzeigen, klicken Sie in den Editor und klicken Sie anschließend auf die Registerkarte **Parameter**.



Manuelles Erstellen einer SQL-Umwandlung

Sie können eine SQL-Umwandlung manuell erstellen. Erstellen Sie die Umwandlung manuell, wenn Sie eine Umwandlung konfigurieren, die eine SQL-Abfrage ausführt. Sie können eine Umwandlung auch manuell erstellen, die eine gespeicherte Prozedur aufruft, wenn die Prozedur für den Import nicht verfügbar ist. Wenn Sie die Umwandlung manuell erstellen, konfigurieren Sie die Eingabe- und Ausgabeports und geben die SQL-Anweisungen im SQL-Editor ein.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die SQL-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Wählen Sie **Als leer erstellen** aus.
6. Geben Sie einen Namen für die Umwandlung den Repository-Speicherort für die Umwandlung ein.
7. Klicken Sie auf **Fertigstellen**.
8. Klicken Sie auf die Ansicht **Übersicht**, um Ports zur Umwandlung hinzuzufügen.
9. Um einen Eingabeport hinzuzufügen, klicken Sie auf **Eingabe** im Bereich **Ports**, um anzugeben, wo der Port hinzugefügt werden soll. Klicken Sie auf die Schaltfläche **Neu** und geben Sie den Portnamen, den nativen Typ und die Gesamtstellenanzahl ein.

   Der Standarddatenbanktyp ist Oracle. Das Developer-Tool zeigt native Typen für Oracle-Datenbanken an, es sei denn, Sie ändern den Datenbanktyp in der Ansicht **Erweitert**.

10. Um einen Ausgabeport hinzuzufügen, klicken Sie vor dem Hinzufügen des Ports auf **Ausgabe** im Bereich **Ports**. Klicken Sie auf die Schaltfläch **Neu** und geben Sie den Portnamen, den nativen Typ und die Gesamtstellenanzahl ein.

    Standardmäßig ist Port **SQL-Fehler** der erste Ausgabeport.

11. Wählen Sie in der Ansicht **Erweitert** den Datenbanktyp aus, mit dem sich die SQL-Umwandlung verbindet. Konfigurieren Sie andere erweiterte Eigenschaften für die Fehlerbehandlung und andere optionale Eigenschaften.

    Wenn Sie den Datenbanktyp auswählen, ändert das Developer-Tool die nativen Datentypen der Ports in der Ansicht **Übersicht**.

12. Typ der SQL-Abfrage oder des gespeicherten Prozeduraufrufs in der Ansicht **SQL**. Wählen Sie Ports für Parameterbindung oder Zeichenfolgenersetzung im **SQL-Editor** aus.

    Wenn die gespeicherte Prozedur einen Ergebnissatz zurückgibt, müssen Sie die gespeicherte Prozedur mit einer Syntax ähnlich der folgenden eingeben: `(?Field1?,?Field2?,?Field3?) = call Stored_Procedure_Name (?Input_Parm?).`

Verwandte Themen:

- "Definieren der SQL-Abfrage" auf Seite 576

Erstellen einer SQL-Umwandlung aus einer gespeicherten Prozedur

Sie können eine SQL-Umwandlung konfigurieren, indem Sie eine gespeicherte Prozedur aus einer Datenbankverbindung importieren.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.

    Das Dialogfeld **Neu** wird angezeigt.

3. Wählen Sie die SQL-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Wählen Sie **Aus einer vorhandenen gespeicherten Prozedur erstellen** aus.
6. Suchen Sie eine Datenbankverbindung und wählen Sie sie aus.
7. Suchen Sie die zu importierende gespeicherte Prozedur und wählen Sie sie aus.
8. Geben Sie einen Namen und einen Speicherort für die Umwandlung ein.
9. Klicken Sie auf **Fertigstellen**.

    Das Developer-Tool erstellt die Ports und den gespeicherten Prozeduraufruf.

10. Wenn die gespeicherte Prozedur einen Ergebnissatz zurückgibt, müssen Sie die Ausgabeports manuell hinzufügen und den gespeicherten Prozeduraufruf neu konfigurieren.

    a. Klicken Sie in der Ansicht **Übersicht** auf **Ausgabe** im Bereich **Ports**. Klicken Sie auf die Schaltfläche **Neu** und geben Sie den Namen des Ausgabeports, den nativen Typ und die Gesamtstellenanzahl ein.

b. Ändern Sie in der Ansicht **SQL** den gespeicherten Prozeduraufruf, um die folgende Syntax zu verwenden: `(?Field1?,?Field2?,?Field3?) = call Stored_Procedure_Name (?Input_Parm?)`

   Sie können die Eingabe- und die Ausgabeparameter aus der Liste **Parameterbindung** der Ports im SQL-Editor auswählen.

KAPITEL 40

Standardisierungs-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Standardisierer-Umwandlung - Übersicht, 586
- Standardisierungsstrategien, 586
- Standardisierungseigenschaften, 587
- Konfigurieren einer Standardisierungsstrategie, 588
- Standardisierer-Umwandlung – Erweiterte Eigenschaften, 588

Standardisierer-Umwandlung - Übersicht

Die Standardisierer-Umwandlung ist eine passive Umwandlung, die Eingabestrings untersucht und standardisierte Versionen dieser Strings erstellt.

Bei der Standardisierer-Umwandlung werden Spalten erstellt, die standardisierte Versionen des Eingabestrings enthalten. Die Umwandlung kann beim Erstellen dieser Spalten Strings in den Eingabedaten ersetzen oder entfernen.

Sie können die Standardisierer-Umwandlung beispielsweise verwenden, um eine Spalte mit Adressdaten zu untersuchen, die die Strings Straße,St. undSTR enthalten. Sie können alle Instanzen dieses Strings durch den String St ersetzen.

Innerhalb einer Standardisierer-Umwandlung können Sie mehrere Standardisierungsstrategien erstellen. Jeder Strategie kann mehrere Standardisierungsoperationen enthalten. Die Standardisierer-Umwandlung bietet einen Assistenten, der zum Erstellen von Strategien verwendet werden kann.

Standardisierungsstrategien

Verwenden Sie Standardisierungsstrategien zum Erstellen von Spalten mit standardisierten Versionen von Eingabestrings.

Wenn Sie eine Standardisierungsstrategie konfigurieren, fügen Sie eine oder mehrere Operationen hinzu. Jede Operation implementiert eine bestimmte Standardisierungsaufgabe.

Sie können einer Standardisierungsstrategie die folgenden Operationstypen hinzufügen:

Ersetzen von Referenztabellen-Matches durch gültige Werte

Ersetzen Sie Strings, die mit Referenztabellen-Werten-übereinstimmen, durch den Wert "Gültig" aus der Referenztabelle.

Ersetzen von Referenztabellen-Matches durch benutzerdefinierte Strings

Ersetzen Sie Strings, die mit Referenztabellen-Werten übereinstimmen, durch einen benutzerdefinierten Ersatzstring.

Entfernen von Referenztabellen-Matches

Entfernen Sie Strings, die mit Referenztabellen-Werten übereinstimmen.

Ersetzen von benutzerdefinierten Strings

Ersetzen Sie benutzerdefinierte Strings durch einen benutzerdefinierten Ersatzstring.

Entfernen benutzerdefinierter Strings

Entfernen Sie benutzerdefinierte Strings.

Wichtig: Sie können die Operationsreihenfolge ändern. Mit der Operationsreihenfolge kann die Ausgabe einer Strategie geändert werden, da jede Operation die Ergebnisse der vorangegangenen Operation liest.

Standardisierungseigenschaften

Zum Konfigurieren der Eigenschaften für Standardisierungsstrategien und -operationen wählen Sie die Ansicht **Strategien** in der Standardisierer-Umwandlung aus.

Strategieeigenschaften

Strategieeigenschaften gelten für alle Operationen in einer Strategie. Sie können die folgenden Strategieeigenschaften konfigurieren:

Mehrere Leerzeichen entfernen

Ersetzt mehrere aufeinander folgende Leerzeichen durch ein Leerzeichen.

Entfernen von voran- und nachgestellten Leerzeichen

Entfernt Leerzeichen am Anfang und Ende von Zeichenfolgen.

Delimiter

Bestimmt die Delimiter, die Such-Token definieren. Wenn Sie beispielsweise "Semikolon" auswählen, sucht die Standardisierer-Umwandlung nach der Zeichenfolge "Orangen;Äpfel" und findet die Zeichenfolgen "Orangen" und "Äpfel". Die Umwandlung verwendet standardmäßig den Leerzeichen-Delimiter.

Operationseigenschaften

Sie können Eigenschaften für die folgenden Typen von Standardisierungsoperationen konfigurieren.

Referenztabellenoperationen

Referenztabellenoperationen weisen die folgenden Eigenschaften auf:

- **Referenztabelle**. Gibt die zum Standardisieren der Daten verwendete Referenztabelle an. Klicken Sie auf **Durchsuchen**, um eine Referenztabelle auszuwählen.
- **Groß-/Kleinschreibung beachten**. Gibt an, ob Eingabezeichenfolgen mit der Groß-/Kleinschreibung der Referenztabelleneinträge übereinstimmen müssen.

- **Ersetzen durch** Ersetzt Eingabezeichenfolgen, die mit Referenztabelleneinträgen übereinstimmen, durch den angegebenen Text. Gilt nur für Ersetzungsoperationen.
- **Bereich** Gibt den Teil der Eingabezeichenfolge an, der den Referenztabellenwert enthält.

Benutzerdefinierte Zeichenfolgenoperationen

Benutzerdefinierte Zeichenfolgenoperationen enthalten die folgenden Eigenschaften:

- **Tokens matchen mit**. Definiert die Suchzeichenfolgen, nach denen in den Eingabedaten gesucht werden soll.
- **Ersetzen durch** Ersetzt Eingabezeichenfolgen, die mit der angegebenen Suchzeichenfolge übereinstimmen. Gilt nur für Ersetzungsoperationen.
- **Bereich** Gibt den Teil der zu suchenden Eingabezeichenfolge an.

Konfigurieren einer Standardisierungsstrategie

Um eine Standardisierungsstrategie zu konfigurieren, bearbeiten Sie die Einstellungen in der Ansicht **Strategien**Ansicht der Standardisierer-Umwandlung.

1. Wählen Sie die Ansicht **Strategien** aus.
2. Klicken Sie auf **Neu**.

   Der Assistent **Neue Strategie** wird geöffnet.
3. Klicken Sie auf das Feld **Eingaben**, um Ports für die Strategie auszuwählen.
4. Konfigurieren Sie die Strategieeigenschaften und klicken Sie auf **Weiter**.
5. Wählen Sie eine Operation aus und klicken Sie auf **Weiter**.
6. Konfigurieren Sie die Operationseigenschaften.
7. Optional können Sie auf **Weiter** klicken, um der Strategie mehrere Vorgänge hinzuzufügen.
8. Nachdem Sie der Strategie alle Operationen hinzugefügt haben, klicken Sie auf **Fertig stellen**.
9. Optional können Sie mehrere Strategien zur Umwandlung hinzufügen.
10. Ändern Sie optional die Reihenfolge, in der die Umwandlung die Strategien oder Operationen verarbeitet. Wählen Sie eine Strategie oder Operation aus und klicken Sie auf **Nach oben verschieben** oder **Nach unten verschieben**.

Standardisierer-Umwandlung – Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, um zu bestimmen, wie der Data Integration Service Daten für die Standardisierer-Umwandlung verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 41

Union-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Union-Umwandlung - Übersicht, 590
- Gruppen und Ports, 591
- Union-Umwandlung - Erweiterte Eigenschaften, 591
- Union-Umwandlung-Verarbeitung, 592
- Erstellen einer Union-Umwandlung, 592

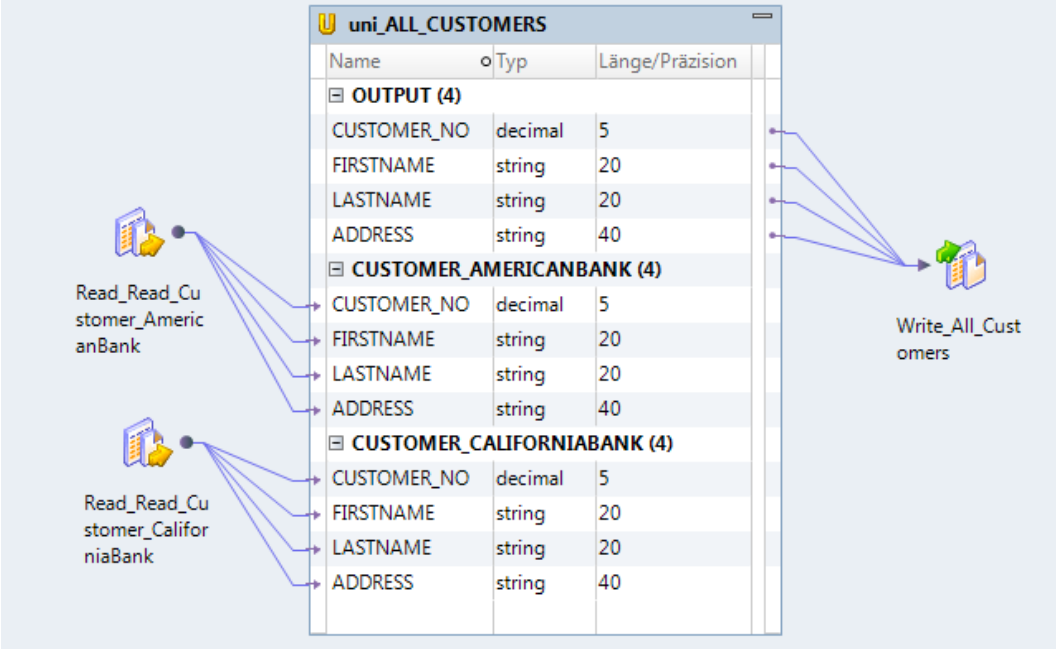
Union-Umwandlung - Übersicht

Verwenden Sie die Union-Umwandlung zum Zusammenführen von Daten aus mehreren Pipelines oder Pipeline-Zweigen in einem Pipeline-Zweig.

Die Union-Umwandlung ist eine aktive Umwandlung mit mehreren Eingabegruppen und einer Ausgabegruppe. Sie führt Quellen mit übereinstimmenden Ports zusammen und übergibt Daten über eine Ausgabegruppe, die dieselbe Portstruktur wie die Eingabegruppen aufweist. Verwenden Sie eine Union-Umwandlung im Developer-Tool, um Daten aus mehreren Quellen zusammenzuführen, ohne die doppelten Zeilen zu entfernen.

Sie möchten beispielsweise Kundenkontodaten der American Bank und der California Bank kombinieren. Sie können ein Mapping erstellen, das eine Union-Umwandlung enthält, um Daten aus den Quellobjekten zusammenzuführen und auf ein Zielobjekt zu schreiben.

Die folgende Abbildung zeigt ein Mapping mit einer Union-Umwandlung:



Gruppen und Ports

Eine Union-Umwandlung hat mehrere Eingabegruppen und eine Ausgabegruppe. Sie können eine oder mehrere Eingabegruppen erstellen. Das Developer Tool erstellt eine Ausgabegruppe. Sie können die Ausgabegruppe nicht erstellen, bearbeiten oder löschen. Jede Gruppe muss über Matching-Ports verfügen.

Um Ports zu erstellen, können Sie sie aus einer Umwandlung kopieren oder manuell erstellen. Beim Erstellen von Ports erstellt das Developer Tool Eingabeports in jeder Eingabegruppe und Ausgabeports in der Ausgabegruppe. Das Developer Tool verwendet die Ausgabeportnamen, die Sie für jeden Eingabe- und Ausgabeport angeben. Das Developer Tool verwendet ebenfalls die gleichen Metadaten für jeden Port, z. B. Datentyp, Präzision und Skala.

Sie können die Eingabegruppen aus unterschiedlichen Verzweigungen in einer einzelnen Pipeline oder aus unterschiedlichen Quell-Pipelines verbinden. Beim Hinzufügen einer Union-Umwandlung zu einem Mapping müssen Sie überprüfen, ob Sie dieselben Ports in allen Eingabegruppen verbunden haben. Wenn Sie einen Port in einer Eingabegruppe verbinden, denselben Port aber nicht in einer anderen Eingabegruppe verbinden, übergibt der Data Integration Service Nullwerte an den nicht verbundenen Port.

Union-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie Eigenschaften, um festzulegen, wie der Data Integration Service Protokolldetails für die Union-Umwandlung anzeigt.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Union-Umwandlung-Verarbeitung

Verwenden Sie die Union-Umwandlung zum Zusammenführen von Daten aus mehreren Pipelines oder Pipeline-Zweigen in einem Pipeline-Zweig. Der Data Integration Service verarbeitet alle Eingabegruppen parallel. Er liest gleichzeitig Quellen, die mit der Union-Umwandlung verbunden sind, und übergibt Datenblöcke an die Eingabegruppen der Umwandlung. Die Union-Umwandlung verarbeitet die Datenblöcke basierend auf dem Auftrag, den sie vom Data Integration Service erhält. Eingabedaten in den Eingabegruppen werden von der Union-Umwandlung nicht gesperrt.

Erstellen einer Union-Umwandlung

Sie können eine wiederverwendbare oder nicht wiederverwendbare Union-Umwandlung erstellen.

Erstellen einer wiederverwendbaren Union-Umwandlung

Erstellen Sie eine wiederverwendbare Union-Umwandlung zur Verwendung in mehreren Mappings oder Mapplets.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner.
2. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**.

   Das Dialogfeld **Neu** wird angezeigt.
3. Wählen Sie die Union-Umwandlung aus.
4. Klicken Sie auf **Weiter**.
5. Geben Sie einen Namen für die Umwandlung ein.
6. Klicken Sie auf **Fertig stellen**.

   Die Umwandlung wird im Editor angezeigt.
7. Klicken Sie auf die Schaltfläche **Neu**, um der Umwandlung einen Port hinzuzufügen.
8. Bearbeiten Sie den Port, um den Namen, den Datentyp und die Genauigkeit festzulegen.
9. Wählen Sie die Ansicht **Gruppen** aus.
10. Klicken Sie auf die Schaltfläche **Neu**, um eine Eingabegruppe hinzuzufügen.
11. Klicken Sie auf die Ansicht **Erweitert** und bearbeiten Sie die Umwandlungseigenschaften.

Erstellen einer nicht wiederverwendbaren Union-Umwandlung

Erstellen Sie eine nicht wiederverwendbare Union-Umwandlung in einem Mapping oder Mapplet.

1. Ziehen Sie in einem Mapping oder Mapplet eine Union-Umwandlung aus einer Umwandlungspalette in den Editor.

   Die Umwandlung wird im Editor angezeigt.

2. Ändern Sie auf der Registerkarte **Allgemein** den Namen und die Beschreibung der Umwandlung.
3. Wählen Sie alle Ports aus der vorangehenden Umwandlung und ziehen Sie sie in die Union-Umwandlung. Die Ports werden als Ports in einer Eingabegruppe und Ausgabegruppe der Union-Umwandlung angezeigt.
4. Klicken Sie auf der Registerkarte **Gruppen** in der Ansicht **Eigenschaften** auf **Neu**, um eine Eingabegruppe hinzuzufügen.

   Ein anderer Eingabegruppe wird angezeigt, die im Vergleich zur vorhandenen Eingabegruppe ähnlichen Ports aufweist.
5. Wählen Sie die Ports in der Ausgabegruppe der Union-Umwandlung aus und ziehen Sie sie zur nachgelagerten Umwandlung im Mapping.

KAPITEL 42

Update-Strategie-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Update-Strategie-Umwandlung - Übersicht, 594
- Aktualisierungsstrategie-Umwandlungen in dynamischen Mappings, 595
- Kennzeichnen von Zeilen in einem Mapping, 595
- Festlegen von Aktualisierungsoptionen für einzelne Targets, 597

Update-Strategie-Umwandlung - Übersicht

Die Update-Strategie-Umwandlung ist eine aktive Umwandlung, die eine Zeile zum Einfügen, Aktualisieren, Löschen oder Verwerfen kennzeichnet. Verwenden Sie eine Update-Strategie-Umwandlung, um Änderungen an vorhandenen Zeilen in einem Target basierend auf einer von Ihnen angewendeten Bedingung zu steuern.

Als aktive Umwandlung kann die Update-Strategie-Umwandlung die Anzahl der durchgeleiteten Zeilen unter Umständen ändern. Die Update-Strategie-Umwandlung testet jede Zeile, um festzustellen, ob eine bestimmte Bedingung erfüllt ist. Anschließend werden die Zeilen entsprechend gekennzeichnet. Die Umwandlung übergibt Zeilen, die zum Einfügen, Aktualisieren oder Löschen gekennzeichnet sind, an die nächste Umwandlung. Sie können die Umwandlung konfigurieren, um zum Verwerfen gekennzeichnete Zeilen an die nächste Umwandlung zu übergeben oder zu entfernen.

Sie können beispielsweise die Update-Strategie-Umwandlung verwenden, um alle Kundenzeilen für die Aktualisierung zu kennzeichnen, wenn sich die Postadresse geändert hat. Alternativ dazu können Sie alle Mitarbeiterzeilen zum Ablehnen von Personen kennzeichnen, die nicht mehr für die Organisation arbeiten.

Einstellen der Update-Strategie

Führen Sie zum Definieren einer Update-Strategie die folgenden Schritte durch:

1. Um zu steuern, wie Zeilen für das Einfügen, Aktualisieren, Löschen oder Ablehnen innerhalb eines Mappings gekennzeichnet werden, fügen Sie dem Mapping eine Update-Strategie-Umwandlung hinzu. Verwenden Sie eine Update-Strategie-Umwandlung, um Zeilen zu kennzeichnen, die für dasselbe Ziel für verschiedene Datenbankoperationen vorgesehen sind, oder, um Zeilen abzulehnen.
2. Definieren Sie für einzelne Targets Optionen zum Einfügen, Aktualisieren und Löschen, wenn Sie das Mapping konfigurieren. Auf einer Target-nach-Target-Basis können Sie Einfügungen und Löschungen für alle Zeilen zulassen, die zum Einfügen oder Löschen gekennzeichnet sind. Sie können verschiedene Methoden wählen, um Aktualisierungen für alle Zeilen zu verarbeiten, die zur Aktualisierung gekennzeichnet sind.

Aktualisierungsstrategie-Umwandlungen in dynamischen Mappings

Sie können eine Aktualisierungsstrategie-Umwandlung in einem dynamischen Mapping verwenden. Sie können dynamische Ports in der Umwandlung konfigurieren und auf die generierten Ports verweisen.

Sie können auf einen dynamischen Port oder einen generierten Port in der Aktualisierungsstrategie-Umwandlung verweisen. Wenn die generierten Ports zur Laufzeit jedoch nicht vorhanden sind, schlägt das Mapping fehl.

Wenn Sie in der Aktualisierungsstrategie-Umwandlung einen dynamischen Port verwenden, darf der dynamische Port nicht mehr als einen generierten Port aufweisen.

Sie können den Aktualisierungsstrategieausdruck parametrisieren. Verwenden Sie einen Ausdruckstypparameter und geben Sie den gesamten Ausdruck als Standardparameterwert ein.

Kennzeichnen von Zeilen in einem Mapping

Fügen Sie eine Update-Strategie-Umwandlung einem Mapping hinzu, um einzelne Zeilen für das Einfügen, Aktualisieren, Löschen oder Ablehnen zu kennzeichnen.

Definieren Sie eine Update-Strategie-Expression, die bei jeder Zeile prüft, ob eine bestimmte Bedingung erfüllt wird. Weisen Sie anschließend jeder Zeile einen numerischen Code zu, der die Zeile für eine bestimmte Datenbankoperation kennzeichnet.

In der folgenden Tabelle werden die Konstanten für jede Datenbankoperation und deren numerische Entsprechung aufgelistet:

Operation	Konstante	Numerischer Wert
Einfügen	DD_INSERT	0
Aktualisieren	DD_UPDATE	1
Delete	DD_DELETE	2
Ablehnen	DD_REJECT	3

Bei allen anderen Werten führt der Data Integration Service die Einfüge-Operation aus.

Aktualisierungsstrategieausdrücke

Geben Sie einen Aktualisierungsstrategieausdruck im Ausdrucksеditor ein.

Der Aktualisierungsstrategieausdruck verwendet die IIF- oder DECODE-Funktion der Umwandlungssprache zum Testen der einzelnen Zeilen. Die folgende IIF-Anweisung kennzeichnet beispielsweise eine Zeile zum Ablehnen, wenn das Eingabedatum nach dem Anwendungsdatum liegt. Andernfalls kennzeichnet die Anweisung die Zeile zum Aktualisieren:

```
IIF( ( ENTRY_DATE > APPLY_DATE), DD_REJECT, DD_UPDATE)
```

Sie können Parameter in einem Aktualisierungsstrategieausdruck konfigurieren. Erstellen Sie Parameter oder durchsuchen Sie Parameter im Ausdrucksseditor.

Wenn sich die Umwandlung in einem dynamischen Mapping befindet, ändern sich möglicherweise die generierten Felder in der Umwandlung. Sie können den vollständigen Aktualisierungsstrategieausdruck parametrisieren. Wenn Sie für das Definieren des Ausdrucks einen Parameter verwenden, kann das Developer Tool den Ausdruck nicht validieren. Ein Ausdrucksparameter darf keinen anderen Parameter enthalten.

Update-Strategie-Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie erweiterte Eigenschaften, um anzugeben, wie der Data Integration Service Daten für die Update-Strategie-Umwandlung verarbeitet.

Sie können die folgenden erweiterten Eigenschaften für die Update-Strategie-Umwandlung auf der Registerkarte „Erweitert“ definieren.

Weiterleiten abgelehnter Zeilen

Gibt an, ob die Update-Strategie-Umwandlung abgelehnte Zeilen an die nächste Umwandlung übergibt oder abgelehnte Zeilen entfernt. Standardmäßig leitet der Data Integration Service abgelehnte Zeilen an die nächste Umwandlung weiter. Der Data Integration Service kennzeichnet die Zeilen für die Ablehnung und schreibt sie in die Ablehnungsdatei. Wenn Sie "Abgelehnte Zeilen weiterleiten" nicht auswählen, entfernt der Data Integration Service abgelehnte Zeilen und schreibt sie in die Protokolldatei des Mappings.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

Aggregator- und Update-Strategie-Umwandlungen

Wenn Sie Aggregator- und Update-Strategie-Umwandlungen als Teil derselben Pipeline verbinden, platzieren Sie die Aggregator-Umwandlung vor der Update-Strategie-Umwandlung. In dieser Reihenfolge führt der Data Integration Service die Aggregat-Berechnung durch und markiert anschließend die Zeilen, die die Ergebnisse dieser Berechnung für das Einfügen, Aktualisieren, Löschen oder Ablehnen enthalten.

Wenn Sie die Update-Strategie-Umwandlung vor der Aggregator-Umwandlung platzieren, müssen Sie beachten, wie die Aggregator-Umwandlung die Zeilen verarbeitet, die für verschiedene Operationen gekennzeichnet sind. In dieser Reihenfolge kennzeichnet der Data Integration Service Zeilen zum Einfügen, Aktualisieren, Löschen oder Ablehnen, bevor er die Aggregat-Berechnung durchführt. Mit der Kennzeichnung einer Zeile legen Sie fest, wie die Aggregator-Umwandlung Werte in dieser Zeile verarbeitet, die in der Berechnung verwendet werden. Wenn Sie beispielsweise eine Zeile zum Löschen kennzeichnen und dann die Zeile zum Berechnen der Summe verwenden, subtrahiert der Data Integration Service den Wert in dieser Zeile. Wenn Sie eine Zeile zum Ablehnen kennzeichnen und die Zeile dann zum Berechnen der Summe verwenden, bezieht der Data Integration Service den Wert nicht in dieser Zeile ein. Wenn Sie eine Zeile zum Einfügen oder Aktualisieren kennzeichnen und die Zeile dann zum Berechnen der Summe verwenden, fügt der Data Integration Service den Wert in dieser Zeile zur Summe hinzu.

Festlegen von Aktualisierungsoptionen für einzelne Targets

Nachdem Sie eine Update-Strategie-Umwandlung zum Kennzeichnen jeder Zeile für eine Datenbankoperation verwendet haben, definieren Sie Einfüge-, Aktualisierungs- und Löschoptionen für jedes Target im Mapping. Sie können Einfüge- oder Löschvorgänge für Zeilen, die zum Einfügen oder Löschen gekennzeichnet sind, verweigern. Sie können verschiedene Methoden wählen, um Aktualisierungen für alle Zeilen zu verarbeiten, die zur Aktualisierung gekennzeichnet sind.

Definieren Sie die Update-Strategie-Optionen in den erweiterten Eigenschaften eines Target-Datenobjekts in einem Mapping. Sie können die folgenden Update-Strategie-Optionen einrichten:

Einfügen

Fügt alle zum Einfügen gekennzeichneten Zeilen in ein Target ein. Aktiviert ist der Standard.

Delete

Löscht alle zum Löschen gekennzeichneten Zeilen aus einem Target. Aktiviert ist der Standard.

Update-Strategie

Update-Strategie für vorhandene Zeilen. Wählen Sie eine der folgenden Strategien aus:

- **Update als Update.** Aktualisiert alle zum Aktualisieren gekennzeichneten Zeilen. Dies ist der Standardwert.
- **Update als Insert.** Fügt alle zum Aktualisieren gekennzeichneten Zeilen ein
- **Update sonst Insert.** Aktualisiert alle zum Aktualisieren gekennzeichneten Zeilen, wenn sie im Target vorhanden sind, und fügt dann alle verbleibenden zum Einfügen gekennzeichneten Zeilen ein.

Abschneiden der Tabelle

Schneidet das Target vor dem Laden von Daten ab. Standardmäßig deaktiviert.

KAPITEL 43

Web-Dienst-Verbraucher-Umwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Web-Dienst-Verbraucher-Umwandlung - Übersicht, 598
- WSDL-Auswahl, 601
- Web-Dienst-Verbraucher-Umwandlung - Ports, 602
- Web-Dienst-Verbraucher-Umwandlung - Eingabemapping, 604
- REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping, 607
- Erweiterte Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung, 611
- Filteroptimierungen, 615
- Erstellen einer Web-Dienst-Verbraucher-Umwandlung, 617
- Web-Dienst-Verbraucher-Umwandlung - Beispiel, 619

Web-Dienst-Verbraucher-Umwandlung - Übersicht

Die Webdienst-Verbraucher-Umwandlung verbindet sich mit einem Webdienst als Webdienst-Client für den Zugriff auf oder die Umwandlung von Daten. Die Webdienst-Verbraucher-Umwandlung ist eine Umwandlung von mehreren Gruppen.

Ein Web-Dienst verwendet offene Standards, wie z. B. SOAP, WSDL und XML. SOAP ist das Kommunikationsprotokoll für Web-Dienste. Die Web-Dienst-Clientanfrage und die Web-Dienst-Antwort sind SOAP-Meldungen. Eine WSDL ist ein XML-Schema, das die Protokolle, Formate und Signaturen der Web-Dienst-Operationen beschreibt.

Zu den Web-Dienst-Operationen gehören Informationsanfragen, Anfragen zum Aktualisieren von Daten oder Anfragen zum Durchführen von Aufgaben. Die Web-Dienst-Verbraucher-Umwandlung sendet beispielsweise eine SOAP-Anfrage zur Ausführung einer Web-Dienst-Operation mit der Bezeichnung getCustomerOrders. Die Umwandlung übergibt eine Kunden-ID in der Anfrage. Der Web-Dienst ruft die Kunden- und Auftragsinformationen ab. Der Web-Dienst gibt die Informationen in einer SOAP-Antwort an die Umwandlung zurück.

Die Web-Dienst-Verbraucher-Umwandlung verbindet sich mit einem Web-Dienst unter Verwendung einer Endpunkt-URL, die in der WSDL, in einer Web-Dienst-Verbindung oder einem Endpunkt-URL-Eingabeport definiert ist. Aktivieren Sie die Sicherheit für Web-Dienste in einer Web-Dienst-Verbindung.

SOAP-Meldungen

Die Web-Dienst-Verbraucher-Umwandlung verwendet das SOAP-Protokoll (Simple Object Access Protocol) für den Austausch von Informationen mit dem Web Services Provider sowie zum Anfordern von Web-Diensten. SOAP definiert das Format für Web-Dienst-Anfrage- und -Antwortmeldungen.

Beim Umwandeln von Daten mit einer Web-Dienst-Verbraucher-Umwandlung erzeugt die Umwandlung eine SOAP-Anfrage und stellt eine Verbindung zum Web-Dienst her. Die Umwandlung verbindet sich mit einem Web-Dienst unter Verwendung einer Endpunkt-URL, die im WSDL-Objekt, in einer Web-Dienst-Verbindung oder einem Endpunkt-URL-Eingabeport definiert ist. Die SOAP-Anfrage enthält die Informationen, die vom Web-Dienst zum Ausführen der angefragten Operation benötigt wird. Die Web-Dienst-Operation gibt Daten an die Umwandlung in einer SOAP-Antwort zurück. Die Umwandlung ordnet Daten aus der SOAP-Antwort zu und gibt die Daten in Ausgabeports zurück.

Die Webdienstbenutzer-Umwandlung verschlüsselt die SOAP-Meldungs-Header in ISO-8859-1.

Die Umwandlung kann SOAP-Meldungen mit Dokument-/Literalverschlüsselung verarbeiten. Der Dokument-/Literalstil benötigt ein XML-Schema zur Beschreibung der SOAP-Meldung. SOAP-Meldungen werden aus dem XML generiert. Wenn eine SOAP-Meldung mehrfach vorkommende Elemente enthält, bilden die Elementgruppen Ebenen in der XML-Hierarchie. Die Gruppen werden verbunden, wenn eine Ebene mit einer anderen verschachtelt wird.

Eine SOAP-Anfragemeldung kann hierarchische Daten enthalten. Die Web-Dienst-Verbraucher-Umwandlung sendet beispielsweise eine Anfrage zum Hinzufügen von Kundenaufträgen zu einer Umsatzdatendatenbank. Die Umwandlung übergibt zwei Gruppen von Daten in einer SOAP-Anfragemeldung. Eine Gruppe enthält eine Kunden-ID und einen -Namen, während die andere Gruppe Auftragsinformationen enthält. Die Auftragsinformationen kommen mehrfach vor.

Eine SOAP-Antwortmeldung kann hierarchische Daten enthalten. Die Web-Dienst-Verbraucher-Umwandlung erzeugt beispielsweise eine SOAP-Anfrage für Kundenaufträge. Der Web-Dienst gibt einen Auftrags-Header und mehrfach vorkommende Elemente mit Auftragsdetails in der SOAP-Antwort zurück.

WSDL-Dateien

Eine WSDL Datei enthält eine Beschreibung der an den Web-Dienst zu übergebenden Daten, so dass der Absender und der Empfänger die auszutauschenden Daten erkennt. Sie müssen eine WSDL-Datei vor dem Erstellen einer Web-Dienst-Verbraucher-Umwandlung in das Repository importieren.

Die WSDL beschreibt die Operationen, die auf den Daten und der Bindung zu einem Protokoll oder Transport durchgeführt werden müssen, damit der Web-Dienst-Verbraucher die Anfragemeldung im richtigen Format senden kann. Die WSDL beschreibt die Netzwerkadresse für die Verbindung mit dem Web-Dienst.

Die WSDL enthält Informationen über die Verschlüsselung von SOAP-Anfrage- und -Antwortmeldungen. Mit der SOAP-Verschlüsselung wird das Format des SOAP-Meldungstexts beschrieben. Sie beschreibt das Format für Anfrage- und Antwortmeldungen, das vom Web-Dienst für die Kommunikation mit einem Web-Dienst-Verbraucher verwendet wird. Web-Dienst-Entwickler können zahlreiche Toolkits zum Erstellen von Web-Diensten verwenden. Toolkits unterstützen verschiedene Arten der Verschlüsselung von SOAP-Meldungen.

Die Web-Dienst-Verbraucher-Umwandlung unterstützt den Dokument-/Literal-SOAP-Verschlüsselungsstil. Sie können WSDL 1.1 mit der Web-Dienst-Verbraucher-Umwandlung verwenden. Sie können keine WSDL-Anhänge, wie MIME-, DIME- und MTOM-Meldungen, verwenden.

Operationen

Ein Web-Dienst enthält eine Operation für jede vom Web-Dienst unterstützte Aktion.

Ein Web-Dienst kann beispielsweise eine Operation namens getcustomerid beinhalten, die einen Kundennamen empfängt und mit Kundendetails antwortet. Die Operation-Eingabe enthält ein Element für den Kundennamen. Die Operation-Ausgabe enthält Elemente für Kundendetails basierend auf dem Kundenname.

Beim Konfigurieren einer Web-Dienst-Verbraucher-Umwandlung definieren Sie, wie die Umwandlung Daten zur Operation-Eingabe zuordnet und wie die Umwandlung Daten aus der Operationsausgabe zuordnet. Sie können die folgenden Informationen in der Umwandlung konfigurieren:

Eingabe-Mapping

Definieren Sie, wie die Umwandlungs-Eingabeports zu den Operation-Eingabeknoten des Web-Dienstes zugeordnet werden. Die Operation-Eingabe definiert die Elemente in der SOAP-Anfrage für die Operation.

Ausgabe-Mapping

Definieren Sie, wie die Operation-Ausgabeknoten des Web-Dienstes zu den Umwandlung-Ausgabeports zugeordnet werden. Die Operation-Ausgabe definiert die Elemente in einer SOAP-Antwort für die Operation.

Web-Dienst-Sicherheit

Aktivieren Sie die Sicherheit für Web-Dienste in einer Web-Dienst-Verbindung. Sie können die folgenden Sicherheitstypen konfigurieren:

Web-Dienst-Sicherheit

Beim Senden einer SOAP-Anfrage an den Web-Dienstanbieter kann der Data Integration Service einen Web-Dienst-Security-Header einbeziehen. Die Web-Dienst-Security-Header enthält Authentifizierungsinformationen, damit der Web-Dienstanbieter den Data Integration Service authentifizieren kann.

Die Web-Dienst-Verbraucher-Umwandlung liefert den Token für den Benutzernamen. Der Data Integration Service erstellt einen separaten SOAP-Security-Header in der SOAP-Anfrage und übergibt die Anfrage an den Web-Dienstanbieter.

Sie können die folgenden Web-Dienst-Sicherheitstypen in einer Web-Dienst-Verbindung verwenden:

- PasswordText. Der Datenintegrationsdienst ändert nicht das Passwort im WS-Security SOAP-Header.
- PasswordDigest. Der Datenintegrationsdienst kombiniert das Passwort mit einem Einmalschlüssel und einem Zeitstempel. Der Data Integration Service wendet einen SHA-Hash auf das Passwort an, kodiert es mit der base64-Kodierung und verwendet das kodierte Passwort im SOAP-Header.

Transport Layer Security (TLS)

Im oberen Bereich der Transportschicht (TCP-Schicht) implementierte Sicherheit von TCP/IP unter Verwendung von SSL (Secure Sockets Layer). Web-Dienste verwenden HTTPS (Hypertext Transfer Protocol over SSL) als Webadresse für die sichere Weiterleitung von Nachrichten. Die folgende Authentifizierung kann mit TLS verwendet werden: HTTP-Authentifizierung, Proxyserver-Authentifizierung und SSL-Zertifikate.

SSL-Authentifizierung

Sie können SSL-Authentifizierung verwenden, wenn Sie eine Verbindung über das HTTPS-Protokoll herstellen.

Sie können die folgenden SSL-Authentifizierungstypen verwenden:

- Eindirektionale SSL-Authentifizierung

- Bidirektionale SSL-Authentifizierung

HTTP-Authentifizierung

Sie können HTTP-Authentifizierung verwenden, wenn Sie eine Verbindung über das HTTP-Protokoll herstellen.

Sie können die folgenden HTTP-Authentifizierungsmethoden verwenden:

- Grundlegende Authentifizierung
- Zusammengefasste Authentifizierung
- NT LAN Manager (NTLM)-Authentifizierung

WSDL-Auswahl

Bevor Sie eine Web-Dienst-Verbraucher-Umwandlung erstellen, müssen Sie eine WSDL-Datei in das Model Repository importieren. Die WSDL definiert die Operationssignatur des auszuführenden Web-Dienstes. Beim Importieren einer WSDL erstellt das Developer Tool ein physisches Datenobjekt, das für andere Umwandlungen wiederverwendet werden kann.

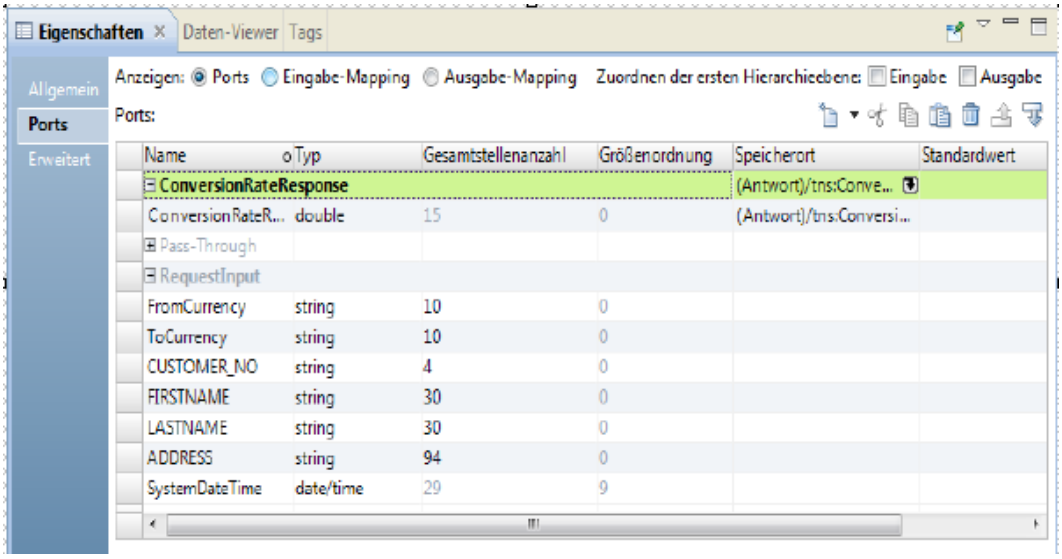
Eine WSDL kann mehrere Operationen definieren. Wählen Sie beim Erstellen einer Web-Dienst-Verbraucher-Umwandlung die Operation aus, die ausgeführt werden soll. Sie können die Operationseingabe- und Operationsausgabehierarchien in der Web-Dienst-Verbraucher-Umwandlung anzeigen. Die Hierarchien definieren die Struktur der SOAP-Anfrage- und der SOAP-Antwortmeldung.

Sie können auch ein WSDL mit einem eindirektionalen Eingabevorgang importieren. Sie müssen Dummy-Ausgabeports erstellen, wenn Sie eine WSDL mit einem eindirektionalen Eingabevorgang importieren.

Web-Dienst-Verbraucher-Umwandlung - Ports

Wenn Sie die Umwandlungsports sehen, blenden Sie die Ports ein, sofern die Operationshierarchie nicht angezeigt werden muss. Wenn Sie die Ports anzeigen, können Sie Gruppen und Ports definieren und Knoten aus der Operationsausgabe den Ausgabeports zuordnen.

Die folgende Abbildung zeigt die Ports für eine nicht wiederverwendbare Webdienstverbraucher-Umwandlung:



Eine Web-Dienst-Verbraucher-Umwandlung kann mehrere Eingabe- und mehrere Ausgabegruppen enthalten. Erstellen Sie beim Anlegen von Ports Gruppen und fügen Sie den Gruppen die Ports hinzu. Definieren Sie die Ports in einer Gruppenhierarchie basierend auf der Struktur der Operationseingabe- und Operationsausgabehierarchie. Fügen Sie einen Schlüssel hinzu, um eine untergeordnete Gruppe mit einer übergeordneten Gruppe zu verbinden. Alle Gruppen mit Ausnahme der untersten Gruppe in der Hierarchie müssen Primärschlüssel aufweisen. Alle Gruppen in der Hierarchie mit Ausnahme der Root-Gruppe müssen Fremdschlüssel aufweisen.

Die Umwandlung verfügt über eine Root-Eingabegruppe mit der Bezeichnung RequestInput. Sie müssen der Root-Eingabegruppe einen Primärschlüssel hinzufügen. Bei dem Schlüssel muss es sich um einen String, Bigint- oder Integer-Datentyp handeln.

Sie können der Root-Eingabegruppe zusätzliche Pass-Through-Ports hinzufügen. Pass-Through-Ports übergeben Daten über die Umwandlung, ohne die Daten zu ändern. Der Pass-Through-Port kann einmal in den Eingabedaten vorkommen. Sie können den Pass-Through-Port zu jeder beliebigen Ausgabegruppe hinzufügen. Ordnen Sie dem Eingabeport den Ausgabeport zu. Der Eingabewert, den Sie über eine SOAP-Anfrage übergeben haben, wiederholt sich in den Ausgabezeilen aus der SOAP-Antwort.

Sie können der Root-Eingabegruppe auch HTTP-Header, Cookie-Ports, einen dynamischen URL-Port und Ports für die Web-Dienst-Sicherheitsauthentifizierung hinzufügen. Daten in der Root-Gruppe kommen einmal vor.

Klicken Sie zum Zuweisen eines Operationsausgabeknotens zu einem Ausgabeport auf das Feld in der Spalte **Speicherort** und erweitern Sie die Hierarchie im Dialogfeld **Speicherort auswählen**. Wählen Sie anschließend einen Knoten aus der Hierarchie aus.

Eingabeports für HTTP-Kopfzeilen

Ein Web-Dienst benötigt möglicherweise zusätzliche HTTP-Kopfzeilen. Sie können Eingabeports in der Root-Eingabegruppe erstellen, um die zusätzlichen Kopfzeileninformationen an den Web-Dienstanbieter weiterzuleiten.

Wählen Sie zum Hinzufügen einer HTTP-Kopfzeile die Root-Eingabegruppe aus und klicken Sie auf den Pfeil neben der Schaltfläche **Neu**. Klicken Sie dann auf **HTTP-Kopfzeile**. Geben Sie einen Kopfzeilen- und Portnamen ein.

Sie können mehrere HTTP-Kopfzeilen erstellen.

Andere Eingabeports

Sie können vordefinierte Eingabeports zur Web-Dienst-Verbraucher-Umwandlung hinzufügen.

Sie können die folgenden vordefinierten Eingabeports hinzufügen:

Cookie-Port

Sie können die Web-Dienst-Verbraucher-Umwandlung zur Verwendung der Cookie-Authentifizierung konfigurieren. Der Remote-Web-Dienst-Server verfolgt die Web-Dienst-Verbraucher-Benutzer basierend auf den Cookies. Sie können die Leistung erhöhen, wenn ein Mapping einen Web-Dienst mehrmals aufruft.

Wenn Sie den Cookie-Port für eine Web-Dienst-Anfragemeldung projektieren, gibt der Web-Dienstanbieter einen Cookie-Wert in der Antwortmeldung zurück. Sie können den Cookie-Wert an eine im Mapping nachgelagerte Umwandlung übergeben oder den Cookie-Wert in einer Datei speichern. Beim Speichern des Cookie-Wertes in einer Datei können Sie den Cookie als Eingabe in die Web-Dienst-Verbraucher-Umwandlung konfigurieren.

Sie können den Cookie-Ausgabeport für alle Ausgabegruppen der Web-Dienst-Verbraucher-Umwandlung projektieren.

Endpunkt-URL-Port

Die Web-Dienst-Verbraucher-Umwandlung verbindet einen Web-Dienst mit einer Endpunkt-URL. Sie können die Endpunkt-URL der WSDL-Datei, in einer Web-Dienst-Verbindung oder einem Ausgabeport einer Endpunkt-URL definieren. Wenn die Umwandlung die URL in einem Port dynamisch empfängt, überschreibt der Data Integration Service die in der WSDL-Datei oder in der Web-Dienst-Verbindung definierte URL.

Die Web-Dienst-Verbraucher-Umwandlung kann einen URL-Portwert für jede Web-Dienst-Anfrage beinhalten. Hinzufügen einer Endpunkt-URL-Port zur Root-Eingabegruppe.

WS-Security-Ports

Sie aktivieren die Web-Dienst-Sicherheit in einer Web-Dienst-Verbindung. Wenn Sie Web-Dienst-Sicherheit definieren, müssen Sie den Benutzernamen und das Passwort in einer Web-Dienst-Verbindung oder in WS-Security-Eingabeports definieren.

Wenn Sie WS-Security-Ports hinzufügen, leiten Sie den Benutzernamen und das Passwort über Eingabeports in der Umwandlung weiter. Wenn die Umwandlung den Benutzernamen und das Passwort dynamisch in Ports empfängt, überschreibt der Data Integration Service die in der Web-Dienst-Verbindung definierten Werte.

Hinweis: Eine Web-Dienst-Verbindung hat einen Benutzernamen und ein Passwort für HTTP und WS-Security-Authentifizierung.

Um vordefinierte Eingabeports hinzuzufügen, klicken Sie auf die Root-Eingabegruppe im Bereich **Ports**. Klicken Sie auf den Pfeil neben der Schaltfläche **Neu** und klicken Sie auf **Andere Ports**. Wählen Sie den hinzuzufügenden Port aus.

Web-Dienst-Verbraucher-Umwandlung - Eingabemapping

Wenn Sie die Umwandlungsports sehen, blenden Sie das Eingabe-Mapping ein, um die Operationseingabehierarchie anzuzeigen. Wenn Sie das Eingabe-Mapping anzeigen, können Sie Eingabegruppen und Eingabeports definieren und die Eingabeports zu Operationseingabeknoten zuordnen.

Das Eingabe-Mapping enthält die folgenden Bereiche:

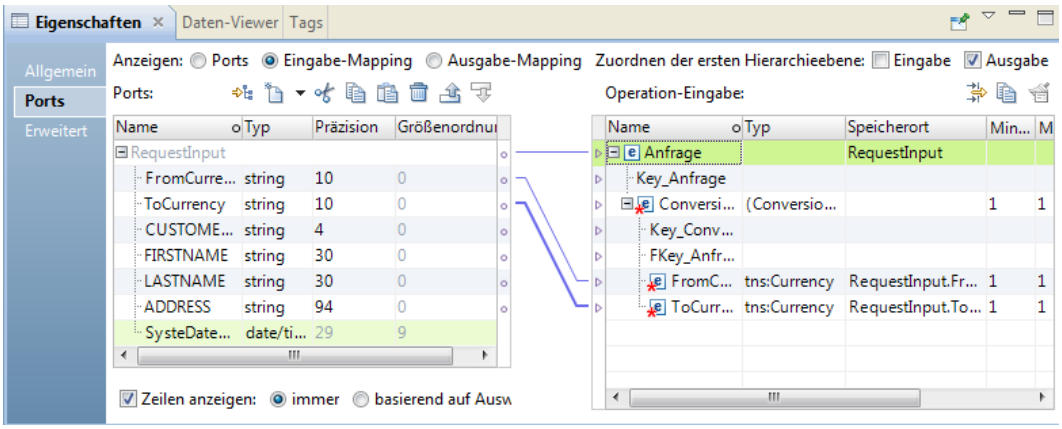
Ports

Erstellen Sie die Umwandlungseingabegruppen und -eingabeports im Bereich **Ports**.

Operationseingabe

Der Bereich **Operationseingabe** zeigt die Knoten in der SOAP-Anfragemeldung an, die von der Web-Dienst-Verbraucher-Umwandlung an den Web-Dienst gesendet werden. Das WSDL-Datenobjekt, das zum Erstellen der Umwandlung verwendet wird, definiert die Operationseingabehierarchie.

Die folgende Abbildung zeigt das Eingabe-Mapping für eine nicht wiederverwendbare Webdienstverbraucher-Umwandlung:



Ordnen Sie nach dem Erstellen der Eingabeports die Eingabeports aus dem Bereich **Ports** den Knoten im Bereich **Operationseingabe** zu. Wenn Sie einem Knoten in der Operationseingabe einen Eingabeport zuordnen, wird der Speicherort des Ports in der Spalte **Speicherort** im Bereich **Operationseingabe** angezeigt.

Das Developer Tool ordnet Eingabeports Knoten auf der ersten Ebene der Operationseingabe zu, wenn die erste Ebene der Eingabehierarchie zugeordnet werden soll. Das Developer-Tool erstellt auch die Ports zum Durchführen des Mappings. Wenn die erste Ebene der Hierarchie einen mehrfach auftretenden übergeordneten Knoten mit einem oder mehreren mehrfach auftretenden untergeordneten Knoten enthält, ordnet das Developer-Tool die erste Ebene der Hierarchie nicht zu.

Sie können der gesamten SOAP-Anfragemeldung XML-Daten aus einem String oder einem Texteingabeport zuordnen. Wenn Sie der gesamten SOAP-Anfrage XML-Daten zuordnen, können Sie Knoten in der Operationseingabe keine Ports zuordnen.

Sie können die Zeilen anzeigen, die die Eingabeports mit den Knoten in der Operationseingabe verbinden.

Verwandte Themen:

- "Generieren von Webdienst-SOAP-Meldungen – Übersicht" auf Seite 632

Regeln und Richtlinien zum Zuordnen von Eingabeports zu Knoten

Überprüfen Sie die folgenden Regeln, wenn Sie Eingabeports zu Knoten in der Operationseingabe-Hierarchie zuordnen:

- Sie können einen Eingabeport einem Knoten in der Hierarchie zuordnen. Sie können den gleichen Port einer beliebigen Anzahl von Schlüsseln in der Hierarchie zuordnen.
- Der Eingabeport und der Knoten müssen kompatiblen Datentypen aufweisen.
- Sie können Ports aus einer Eingabegruppe zu mehreren Hierarchie-Ebenen in der Operationseingabe zuordnen.
- Sie müssen Eingabeports zu den Schlüsseln in der Operationseingabe zuordnen. Alle Ports, die einem Schlüssel zugeordnet werden, müssen den String-, Integer- oder Bigint-Datentyp aufweisen. Ordnen Sie Daten den Schlüsseln auf allen Ebenen der Operationseingabe über der Hierarchie-Ebene zu, die Sie in der SOAP-Meldung einschließen. Schließen Sie die Fremdschlüssel für alle Ebenen ein, die sich über der zuzuordnenden Ebene befinden, einschließlich dieser Ebene selbst.

  Hinweis: Sie müssen Eingabeports Schlüsseln nicht zuordnen, wenn Sie nur die niedrigste Ebene der Operationseingabe-Hierarchie zuordnen.
- Sie können mehrere String-, Bigint- oder Integer-Eingabeports einem Schlüssel im Bereich **Operationseingabe** zuordnen, um einen zusammengesetzten Schlüssel zu erstellen. Wenn Sie bei einem zusammengesetzten Schlüssel auf das Feld **Speicherort** klicken, können Sie die Eingabeports neu anordnen oder einen der Ports entfernen.

Anpassen von Anzeigeoptionen

Sie können die Hierarchie für die Vorgangseingabehierarchie ändern, um Schlüssel im Bereich **Vorgangseingabe** anzuzeigen. Außerdem können Sie den Gruppierungsaufbau anzeigen, der die Reihenfolge von Knoten definiert.

Klicken Sie auf die Schaltfläche **Ansicht anpassen** ( ) im Bereich **Vorgangseingabe**. Aktivieren Sie eine der folgenden Optionen:

Sequenz, Auswahl und Alles

Zeigen Sie eine Zeile an, die angibt, ob als Elementdefinition „Sequenz", „Auswahl" oder „Alle" ausgewählt ist.

Knoten in einer Alle-Gruppe müssen alle in der SOAP-Meldung enthalten sein.

Knoten in einer Sequenz-Gruppe müssen in der in WSDL festgelegten Reihenfolge angeordnet sein.

Mindestens ein Knoten in einer Auswahlgruppe muss in der SOAP-Meldung angezeigt werden.

Schlüssel

Zeigen Sie die Schlüssel im Bereich **Vorgangseingabe** an. Der Bereich **Vorgangseingabe** umfasst Schlüssel für jede Gruppe. Im Bereich **Ports** können Sie einen Schlüssel zu jedem Eingabeport hinzufügen.

Zuordnen von Eingabeports zur Vorgangseingabe

Wenn Sie das Umwandlungseingabe-Mapping anzeigen, können Sie Eingabegruppen Eingabegruppen und Eingabeports definieren und Eingabeports zu Operationseingabeknoten zuordnen.

1. Öffnen Sie eine Webdienstverbraucherumwandlung.
2. Mithilfe einer der folgenden Methoden können Sie das Umwandlungseingabe-Mapping anzeigen:
   - Klicken Sie im Falle einer wiederverwendbaren Umwandlung auf die Ansicht **Übersicht**. Aktivieren Sie die Anzeige des Eingabe-Mappings.
   - Klicken Sie im Falle einer nicht wiederverwendbaren Umwandlung auf die Registerkarte **Ports** in der Ansicht **Eigenschaften**. Aktivieren Sie die Anzeige des Eingabe-Mappings.
3. Definieren Sie einen Primärschlüssel für die Root-Eingabegruppe.
4. Nutzen Sie zum Hinzufügen einer Eingabegruppe oder eines Eingabeports zum Bereich **Ports** eine der folgenden Methoden:

Option	Beschreibung
Knoten ziehen	Ziehen Sie einen Gruppenknoten oder einen untergeordneten Knoten im Bereich**Vorgangseingabe** in eine leere Spalte im Bereich **Ports**. Wenn es sich um einen Gruppenknoten handelt, fügt das Developer Tool eine Gruppe ohne Ports hinzu.
Gruppe oder Port manuell hinzufügen	Klicken Sie zum Hinzufügen einer Gruppe auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Gruppe**. Klicken Sie zum Hinzufügen eines Ports auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Feld**.
Port aus einer anderen Umwandlung ziehen	Ziehen Sie im Editor einen Port aus einer anderen Umwandlung in die Web-Dienst-Verbraucher-Umwandlung.
Port kopieren	Wählen Sie Ports einer anderen Umwandlung aus, und kopieren Sie sie in den Bereich **Ports**. Zum Kopieren von Ports können Sie Tastenkombinationen verwenden oder die Schaltflächen zum Kopieren und Einfügen im Developer Tool.
Zuordnen der ersten Hierarchieebene auswählen	Wählen Sie **Zuordnen der ersten Hierarchieebene** aus. Das Developer Tool weist Knoten auf der ersten Ebene der Vorgangseingabe zu Eingabeports und Gruppen zu. Das Developer Tool erstellt auch die Eingabeports und Eingabegruppen, um das Mapping durchzuführen.

5. Wenn Sie einen Port manuell erstellen oder aus einer anderen Umwandlung kopieren, klicken Sie auf die Spalte **Speicherort** im Bereich **Vorgangseingabe** und wählen Sie einen Port aus der Liste.
6. Um Eingabeports als zusammengesetzten Schlüssel zuzuordnen, nutzen Sie eine der folgenden Methoden:

Option	Beschreibung
Ziehen von Eingabeports	Wählen Sie zwei oder mehr Eingabeports und ziehen Sie sie zu einem Schlüssel in der Vorgangseingabehierarchie.
Wählen Sie Eingabeports aus dem Dialogfeld **Speicherort auswählen** aus.	Klicken Sie auf die Spalte **Speicherort** eines Schlüssels in der Vorgangseingabehierarchie und wählen Sie anschließend die Eingabeports aus.

7. Um die Knotenspeicherorte zu löschen, nutzen Sie eine der folgenden Methoden:

Option	Beschreibung
Auf **Löschen** klicken	Wählen Sie im Bereich **Vorgangseingabe** mindestens einen Knoten aus und klicken Sie auf **Löschen**.
Löschen der Zeilen, die Ports mit Knoten verbinden	Wählen mindestens eine Zeile aus, die die Eingabeports mit den Knoten in der Vorgangseingabe verbindet, und klicken Sie auf **Löschen**.

8. Wenn das zugehörige WSDL-Datenobjekt anyType-Elemente, any-Elemente, anyAttribute-Attribute, Elemente eines abgeleiteten Typs oder Substitutionsgruppen enthält, wählen Sie Objekte im Bereich **Vorgangseingabe**. Klicken Sie in der Spalte **Typ** eines Knotens auf **Auswählen** und wählen Sie dann mindestens einen Typ, ein Element oder ein Attribut aus der Liste.
9. Um XML-Daten aus einer Zeichenfolge oder einem Texteingabeport zur vollständigen SOAP-Anfrage zuzuordnen, klicken Sie mit der rechten Maustaste auf den Port und wählen Sie **Als XML zuordnen**.

REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping

Wenn Sie die Umwandlungsports sehen, blenden Sie das Ausgabe-Mapping ein, um die Operationsausgabehierarchie anzuzeigen. Wenn Sie das Ausgabe-Mapping anzeigen, können Sie Ausgabegruppen und Ausgabeports definieren und den Ausgabeports Operationsausgabeknoten zuordnen.

Das Ausgabe-Mapping enthält die folgenden Bereiche:

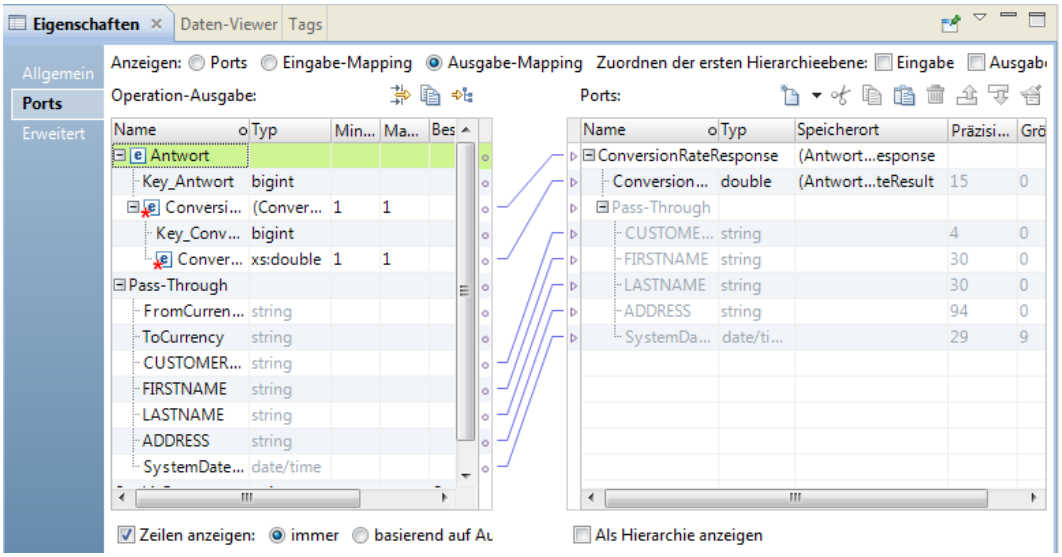
Operationsausgabe

Der Bereich **Operationsausgabe** zeigt die Knoten in der SOAP-Antwortmeldung an, die vom Web-Dienst an die Web-Dienst-Verbraucher-Umwandlung zurückgegeben werden. Das WSDL-Datenobjekt, das zum Erstellen der Umwandlung verwendet wird, definiert die Operationsausgabehierarchie.

Ports

Erstellen Sie die Umwandlungsausgabegruppen und -ports im Bereich **Ports**.

Die folgende Abbildung zeigt das Ausgabe-Mapping für eine nicht wiederverwendbare Webdienstverbraucher-Umwandlung:



Ordnen Sie nach dem Erstellen der Ausgabeports die Knoten aus dem Bereich **Operationsausgabe** den Ports im Bereich **Ports** zu. Wenn Sie einem Ausgabeport einen Knoten aus der Vorgangsausgabe zuordnen, wird der Speicherort des Knotens in der Spalte **Speicherort** im Bereich **Ports** angezeigt.

Das Developer Tool ordnet Ausgabeports Knoten auf der ersten Ebene der Operationsausgabe zu, wenn die erste Ebene der Ausgabehierarchie zugeordnet werden soll. Das Developer-Tool erstellt auch die Ports zum Durchführen des Mappings. Wenn die erste Ebene der Hierarchie einen mehrfach auftretenden übergeordneten Knoten mit einem oder mehreren mehrfach auftretenden untergeordneten Knoten enthält, ordnet das Developer-Tool die erste Ebene der Hierarchie nicht zu.

Sie können die Ausgabeports in einer Hierarchie anzeigen. Jede untergeordnete Gruppe wird unter der übergeordneten Gruppe angezeigt. Sie können die Zeilen anzeigen, die die Knoten in der Vorgangsausgabe mit den Ausgabeports verbinden.

Wenn das zugehörige WSDL-Datenobjekt aus dem Repository gelöscht wird, behält das Developer Tool den Speicherort der Operationsknoten im Ausgabe-Mapping bei. Wenn Sie das Ausgabe-Mapping anzeigen, wird im Bereich **Ports** weiterhin der Speicherort der Operationsknoten in der Spalte **Speicherort** für die Ausgabeports angezeigt. Wenn Sie eine andere WSDL mit der Umwandlung verknüpfen, überprüft das Developer Tool die Gültigkeit aller Speicherorte. Das Developer-Tool löscht den Speicherort der Operationsknoten im Bereich **Ports** des Ausgabe-Mappings, wenn der Speicherort nicht mehr gültig ist.

Verwandte Themen:

- "Parsen von Webdienst-SOAP-Meldungen – Übersicht" auf Seite 623

Regeln und Richtlinien zum Zuordnen von Knoten zu Ausgabeports

Überprüfen Sie die folgenden Regeln, wenn Sie Knoten in der Operationsausgabehierarchie zu Ausgabeports zuordnen:

- Der Operationsausgabeknoten und der Ausgabeport müssen kompatible Datentypen aufweisen.
- Sie können einen Knoten nicht mehr als einem Ausgabeport in einer Gruppe zuordnen.
- Jeder Ausgabeport muss einen gültigen Speicherort aufweisen, es sei denn, es handelt sich um einen Pass-Through-Port.

- Wenn Sie einen mehrfach vorkommenden untergeordneten Konten in einen leeren Ausgabeport ziehen, müssen Sie die Gruppe mit anderen Ausgabegruppen verknüpfen. Wenn Sie eine Gruppe auswählen, erstellt das Developer Tool Schlüssel, um die Gruppen zu verknüpfen.
- Wenn Sie ein mehrfach vorkommendes Element in einer Gruppe ziehen, das das übergeordnete Element enthält, können Sie die Anzahl der einzubeziehenden untergeordneten Elemente konfigurieren. Sie können die übergeordnete Gruppe mit der mehrfach vorkommenden untergeordnete Gruppe in der Umwandlungsausgabe ersetzen.

Mapping der SOAP-Meldung als XML

Sie können die komplette SOAP-Meldung als XML senden, anstatt die Daten an separate Ausgabeports zurückzugeben.

Wenn Sie die SOAP-Meldung als XML zuordnen, gibt der Data Integration Service die komplette SOAP-Meldung an einen Port zurück. Erstellen Sie keine Ausgabeports.

Um die komplette Nachricht zuzuordnen, klicken Sie mit der rechten Maustaste auf die Root-Gruppe im Bereich **Operationsausgabe**. Wählen Sie **Als XML zuordnen** aus.

Das Developer Tool erstellt einen Stringausgabeport. Die Präzision ist 65535 Byte.

Anpassen von Anzeigeoptionen

Sie können die Operationsausgabe-Hierarchie so ändern, dass Cookie-Ports, Übergabeports und Schlüssel im Bereich **Vorgangsausgabe** angezeigt werden. Außerdem können Sie den Gruppierungsaufbau anzeigen, der die Reihenfolge von Knoten definiert.

Klicken Sie auf die Schaltfläche **Ansicht anpassen** im Bereich **Vorgangsausgabe**. Aktivieren Sie eine der folgenden Optionen:

Sequenz, Auswahl und Alles

Zeigen Sie eine Zeile an, die angibt, ob als Elementdefinition "Sequenz", "Auswahl" oder "Alle" ausgewählt ist.

Knoten in einer Alle-Gruppe müssen alle in der SOAP-Meldung enthalten sein.

Knoten in einer Sequenz-Gruppe müssen in der in WSDL festgelegten Reihenfolge angeordnet sein.

Mindestens ein Knoten in einer Auswahlgruppe muss in der SOAP-Meldung angezeigt werden.

Schlüssel

Zeigen Sie die Schlüssel im Bereich **Vorgangsausgabe** an. Der Bereich **Vorgangsausgabe** enthält Schlüssel für jede Gruppe. Sie können einen Schlüssel einem Ausgabeport im Bereich **Ports** hinzufügen.

Übergabeports

Der Bereich **Vorgangsausgabe** zeigt die Übergabeports an. Übergabeports sind Ports, die Daten ohne Änderung an die Umwandlung übergeben. Sie können Übergabeports von der Operationsausgabe auf beliebige Ausgabegruppen der Web-Dienst-Verbraucher-Umwandlung projizieren. Ein Übergabeport empfängt Daten einmalig, damit der Port sich in den SOAP-Meldungen auf Root-Ebene befindet.

Cookie-Ports

Zeigt den Cookie-Port an. Beim Konfigurieren der Cookie-Authentifizierung verfolgt der Remote-Web-Dienst-Server die Web-Dienst-Verbraucher-Benutzer auf der Basis von Cookies. Wenn Sie ein Web-Dienst-Cookie in der Anfragemeldung projizieren, gibt der Web-Dienst ein Cookie in der Antwortmeldung zurück. Sie können das Cookie aus der Operationsausgabe auf beliebige Ausgabegruppen der Web-Dienst-Verbraucher-Umwandlung projizieren.

Zuordnen der Vorgangsausgabe zu Ausgabeports

Wenn Sie das Umwandlungs-Ausgabe-Mapping anzeigen, können Sie Ausgabegruppen oder Ausgabeports definieren und Operationsausgabe-Knoten zu Ausgabeports zuordnen.

1. Öffnen Sie eine Webdienstverbraucherumwandlung.
2. Mithilfe einer der folgenden Methoden können Sie das Umwandlungsausgabe-Mapping anzeigen:
   - Klicken Sie im Falle einer wiederverwendbaren Umwandlung auf die Ansicht **Übersicht**. Aktivieren Sie die Anzeige des Ausgabe-Mappings.
   - Klicken Sie im Falle einer nicht wiederverwendbaren Umwandlung auf die Registerkarte **Ports** in der Ansicht **Eigenschaften**. Aktivieren Sie die Anzeige des Ausgabe-Mappings.
3. Nutzen Sie zum Hinzufügen einer Ausgabegruppe oder eines Ausgabeports zum Bereich **Ports** eine der folgenden Methoden:

Option	Beschreibung
Knoten ziehen	Ziehen Sie einen Gruppenknoten oder einen untergeordneten Knoten im Bereich **Vorgangsausgabe** in eine leere Spalte im Bereich **Ports**. Wenn es sich um einen Gruppenknoten handelt, fügt das Developer Tool eine Gruppe ohne Ports hinzu.
Gruppe oder Port manuell hinzufügen	Klicken Sie zum Hinzufügen einer Gruppe auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Gruppe**. Klicken Sie zum Hinzufügen eines Ports auf den Pfeil neben der Schaltfläche **Neu** und anschließend auf **Feld**.
Port aus einer anderen Umwandlung ziehen	Ziehen Sie im Editor einen Port aus einer anderen Umwandlung in die Web-Dienst-Verbraucher-Umwandlung.
Port kopieren	Wählen Sie Ports einer anderen Umwandlung aus, und kopieren Sie sie in den Bereich **Ports**. Zum Kopieren von Ports können Sie Tastenkombinationen verwenden oder die Schaltflächen zum Kopieren und Einfügen im Developer Tool.
Zuordnen der ersten Hierarchieebene auswählen	Wählen Sie **Zuordnen der ersten Hierarchieebene** aus. Das Developer Tool ordnet Knoten in der ersten Ebene der Operationsausgabe den Ausgabeports und -gruppen zu. Das Developer Tool erstellt außerdem die Ausgabeports und -gruppen zur Ausführung des Mappings.

4. Wenn Sie einen Port manuell erstellen oder aus einer anderen Umwandlung kopieren, klicken Sie auf die Spalte **Speicherort** im Bereich **Ports** und wählen Sie einen Knoten aus der Liste.
5. Um die Portspeicherorte zu löschen, nutzen Sie eine der folgenden Methoden:

Option	Beschreibung
Auf **Löschen** klicken	Wählen Sie einen oder mehrere Ports im Bereich **Ports** aus, und klicken Sie auf **Löschen**.
Zeilen löschen, die Knoten mit Ports verbinden	Wählen Sie eine oder mehrere Zeilen aus, die die Knoten in der Operationsausgabe mit den Ausgabeports verbinden, und drücken Sie **Löschen**.

6. Wenn das zugehörige WSDL-Datenobjekt anyType-Elemente, any-Elemente, anyAttribute-Attribute, Elemente eines abgeleiteten Typs oder Substitutionsgruppen enthält, wählen Sie Objekte im Bereich **Vorgangsausgabe**. Klicken Sie in der Spalte **Typ** eines Knotens auf **Auswählen** und wählen Sie dann mindestens einen Typ, ein Element oder ein Attribut aus der Liste.
7. Um die vollständige SOAP-Antwortmeldung als XML zuzuordnen, klicken Sie mit der rechten Maustaste auf die Root-Gruppe im Bereich **Vorgangsausgabe** und wählen Sie **Als XML zuordnen**.

Erweiterte Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung

Die erweiterten Eigenschaften der Web-Dienst-Verbraucher-Umwandlung umfassen die Tracingebene, generische Fehlerports, Web-Dienst-Verbindungen und Web-Dienst-Anfragemeldungen.

Sie können die folgenden erweiterten Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung auf der Registerkarte „Erweitert“ definieren.

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

SOAP-Aktion

Überschreibt den in der WSDL definierten SOAP-Aktionswert mit einem Konstantenwert für die Web-Dienst-Verbraucher-Umwandlung.

Generische SOAP-Fehlerbehandlung aktivieren

Gibt Fehlermeldungen zurück, die nicht in der WSDL definiert sind. Erstellt Ausgabeports in einer GenericFault-Ausgabegruppe, um Fehlercodes und -meldungen zu verarbeiten.

Die folgende Tabelle beschreibt den Fehlerausgabeport für SOAP 1.1 und SOAP 1.2:

Fehlerausgabeport für SOAP 1.1	Fehlerausgabeport für SOAP 1.2	Beschreibung
Fehlercode	Code*	Gibt einen Fehleridentifikationscode zurück.
Fehlerstring	Grund*	Gibt eine Erläuterung des Fehlers in einer Fehlermeldung zurück.
Fehlerdetail	Details	Gibt die benutzerdefinierten Informationen zurück, die der Web-Dienstanbieter an die Web-Dienst-Verbraucher-Umwandlung in einer generischen Fehlermeldung übergibt.
Fehlerakteur	Rolle	Gibt die Informationen über das Objekt zurück, auf das der Fehler zurückzuführen ist.
-	Knoten	Gibt den URI des SOAP-Knoten zurück, der den Fehler generiert hat.
* Die Ausgabeports für Code und Grund sind hierarchisch angeordnet.		

Hinweis: Sie können den Fehlerausgabeport für Code erweitern, um den Fehlerausgabeport für Untercode eine Ebene nach oben zu heben.

HTTP-Fehlerbehandlung aktivieren

Gibt alle HTTP-Fehler aus dem Web-Dienst zurück. Erstellt einen HTTP-Fehlerausgabeport in der GenericFault-Ausgabegruppe.

"Fault" als Fehler behandeln

Fügt Fehlermeldungen zum Mapping-Protokoll hinzu. Wenn ein Fehler auftritt, erhöht der Data Integration Service den Fehlerzähler für das Mapping. Deaktivieren Sie diese Eigenschaft, um Early Selection- und Push-Into-Optimierung zu ermöglichen. Aktiviert ist der Standard.

Verbindung

Identifiziert das Web-Dienst-Verbindungsobjekt zum Herstellen einer Verbindung zum Web-Dienst. Erstellt die Web-Dienst-Verbindung im Developer Tool. Bearbeiten Sie die Web-Dienst-Verbindung im Developer Tool oder im Administrator Tool. Wenn Sie eine Web-Dienst-Verbindung konfigurieren, konfigurieren Sie die Endpunkt-URL, den vom Web-Dienst benötigten Sicherheitstyp sowie eine Verbindungstimeoutperiode.

Die Webdienst-Verbraucher-Umwandlung verbindet einen Web-Dienst mit einer Endpunkt-URL. Sie können die Endpunkt-URL der WSDL-Datei, in einer Web-Dienst-Verbindung oder einem Ausgabeport einer Endpunkt-URL definieren.

Mithilfe der folgenden Richtlinien können Sie bestimmen, wann Sie eine Web-Dienst-Verbindung konfigurieren:

- Konfigurieren Sie eine Verbindung, wenn Sie eine Endpunkt-URL verwenden möchten, die sich von der URL in der WSDL unterscheidet, und wenn Sie keinen Endpunkt-URL-Eingabeport verwenden.
- Konfigurieren Sie eine Verbindung, wenn für den Web-Dienst, zu dem Sie eine Verbindung herstellen, Web-Dienst-Sicherheit, HTTP-Authentifizierung oder SSL-Zertifikate erforderlich sind.
- Konfigurieren Sie eine Verbindung, wenn Sie die Standardverbindungs-Timeoutperiode ändern möchten.

Hinweis: Sie können ein WSDL-Datenobjekt im Repository zu einer Web-Dienst-Verbindung zuordnen. Die zugeordnete Verbindung wird für jede Web-Dienst-Verbraucher-Umwandlung, die Sie aus dieser WSDL erstellen, zur Standardverbindung.

Kompression aktivieren

Ermöglicht die Codierung der SOAP-Anfragen mit der GZIP-Kompressionsmethode sowie die Decodierung von SOAP-Antworten mit GZIP oder deflate.

Validierung des XML-Schemas

Validiert die SOAP-Antwortmeldung zur Laufzeit. Wählen Sie **Fehler bei ungültigem XML** oder **Keine Validierung** aus.

Sortierte Eingabe

Ermöglicht dem Data Integration Service das Generieren der Ausgabe, ohne alle Eingabedaten zu verarbeiten. Aktivieren Sie sortierte Eingabe, wenn die Eingabedaten durch die Schlüssel in der Operation-Eingabe-Hierarchie sortiert werden.

Push-Into-Optimierung

Ermöglicht Push-Into-Optimierung. Klicken Sie in der Eigenschaft **Push-Into-Optimierung** auf die Schaltfläche **Öffnen**, um Filterports auszuwählen, die Filterwerte empfangen. Wählen Sie für jeden Filterport den Ausgabeport aus, der die gefilterte Spalte in der Web-Dienst-Antwort enthält.

Hat Nebeneffekte

Kontrollkästchen, das angibt, dass vom Web-Dienst alle Funktionen mit Ausnahme der Rückgabe von Zeilen durchgeführt werden. Die Web-Dienst-Verbraucher-Umwandlung hat einen Nebeneffekt, wenn der Web-Dienst neben der Rückgabe von Zeilen ein Objekt ändert oder mit anderen Objekten und Funktionen interagiert. Der Web-Dienst kann gegebenenfalls eine Datenbank ändern, eine Gesamtsumme vergrößern, eine Ausnahmebedingung auslösen, eine E-Mail schreiben oder einen anderen Web-Dienst mit

Nebeneffekten aufrufen. Deaktivieren Sie die Eigenschaft **Hat Nebeneffekte**, um Push-Into- oder "Early Selection"-Optimierung zuzulassen. Aktiviert ist der Standard.

Parallelverarbeitung aktivieren

Ermöglicht der Web-Dienst-Verbraucher-Umwandlung die Herstellung mehrerer gleichzeitiger Verbindungen mit einem Web-Dienst, so dass mehrere Web-Dienst-Anfragen gleichzeitig gesendet werden können. Wenn Sie die Web-Dienst-Verbraucher-Umwandlung aktivieren, um mehrere gleichzeitige Verbindungen mit einem Web-Dienst herzustellen, können Sie die Obergrenze für den Gesamtspeicherbedarf und die maximale Anzahl an gleichzeitigen Verbindungen festlegen.

In der folgenden Tabelle werden die Optionen erläutert.

Optionen	Beschreibung
Parallelverarbeitung aktivieren	Erstellt mehrere gleichzeitige Verbindungen zu einem Web-Dienst.
Maximale Anzahl von Verbindungen bei Parallelverarbeitung	Die Anzahl gleichzeitiger Web-Dienst-Verbindungen. Voreingestellt ist 20.
Obergrenze des Gesamtspeicherbedarfs bei Parallelverarbeitung (in MB)	Die Obergrenze der gesamten Speicherzuordnung für alle gleichzeitigen Verbindungen. Voreingestellt ist 100 MB.

Web-Dienst-Fehlerbehandlung

Sie können die Web-Dienst-Verbraucher-Umwandlung so konfigurieren, dass SOAP- und HTTP-Fehler in einem Mapping nach unten weitergegeben werden. Sie können den Fehlerzähler erhöhen, wenn ein Fehler auftritt. Konfigurieren Sie die Web-Dienst-Fehlerbehandlung in den erweiterten Eigenschaften der Umwandlung.

Ein Web-Dienst gibt entweder eine Antwortmeldung oder einen Fehler zurück. Ein Fehler ist ein Fehler. Der Web-Dienst kann verschiedene Fehler auf Basis der auftretenden Fehler erstellen.

Die Web-Dienst-Verbraucher-Umwandlung kann die folgenden Fehlertypen zurückgeben:

SOAP-Fehler

SOAP-Fehler, der von der WSDL definiert wird. Konfigurieren Sie Fehlerausgabeports, die die Fehler in einer Web-Dienst-Antwortmeldung zurückgeben. Für eine SOAP 1.1-Bindung gibt der Data Integration Service die Fehlermeldung, den Fehlercode, Fehlerstring und Fehlerakteurelemente für den Fehler zurück. Für eine SOAP 1.2-Bindung gibt der Data Integration Service die Fehlermeldung, den Code, Grund, Knoten und Rollenelmente für den Fehler zurück.

Generische SOAP-Fehler

Der Web-Dienst erzeugt generische SOAP-Fehler zur Laufzeit. SOAP 1.1- und SOAP 1.2-Bindungen weisen unterschiedliche Fehlerelemente auf. Die WSDL definiert keine generischen SOAP-Fehler. Zu den generischen SOAP-Fehlern gehören Authentifizierungsfehler und SOAP-Anfragefehler.

HTTP-Fehler

Das Developer Tool fügt den HTTP-Fehlerausgabeport hinzu, wenn Sie HTTP-Fehlerbehandlung in der Umwandlung aktiviert haben. Der Data Integration Service gibt HTTP-Fehler aus dem Web-Dienst in einem einzelnen Stringport zurück. Ein HTTP-Fehler setzt sich aus einem Fehlercode und einer Meldung zusammen.

Wenn die SOAP-Antwort aus dem Web-Dienst ungültige XML-Daten aufweist, gibt die Web-Dienst-Verbraucher-Umwandlung einen Fehler zurück.

Sie können angeben, ob SOAP-Fehler als Fehler behandelt werden sollen. Wenn Sie "Fault als Fehler behandeln" aktiviert haben und ein SOAP-Fehler auftritt, erhöht der Data Integration Service den Fehlerzähler für das Mapping. Der Fehler wird im Meldungsprotokoll angezeigt.

Meldungskomprimierung

Wenn Sie die SOAP-Meldungskomprimierung aktivieren, komprimiert die Web-Dienst-Verbraucher-Umwandlung Web-Dienst-Anfragemeldungen und empfängt komprimierten Web-Dienst-Antwortmeldungen.

Die Web-Dienst-Verbraucher-Umwandlung kodiert die SOAP-Anfrage mit GZip-Komprimierung. Die Umwandlung akzeptiert Antwortmeldungen, die mit GZip oder deflate komprimiert sind.

Wenn der Data Integration Service die Antwort vom Web-Dienst empfängt, prüft er die Content-Encoding-HTTP-Kopfzeile in der SOAP-Meldung und dekodiert die Meldung.

Standardmäßig wird keine Komprimierungskodierung verwendet. Der Web-Dienst komprimiert keine SOAP-Antworten.

In der folgenden Tabelle werden die Kopfzeilen in den Anfrage- und Antwortmeldungen bei ein- und ausgeschalteter Komprimierung gezeigt:

Compression	Kopfzeile
Ein	Content-Encoding-Kopfzeile: GZip Accept-Encoding-Kopfzeile GZip, deflate:
Aus	Leere Content-Encoding-Kopfzeile Leere Accept-Encoding-Kopfzeile

Gelegentlich kodiert ein Web-Dienst eine Antwortmeldung mit einer Standardkomprimierung. Die Web-Dienst-Verbraucher-Umwandlung dekodiert die Meldung, wenn diese mit GZip oder deflate kodiert wurde. Die Web-Dienst-Verbraucher-Umwandlung protokolliert eine Meldung im Mapping-Protokoll, wenn der Web-Dienst die Antwortmeldung unerwartet kodiert.

Aktivieren Sie die Komprimierung in den erweiterten Eigenschaften der Umwandlung.

Parallelverarbeitung

Sie können der Web-Dienst-Verbraucher-Umwandlung ermöglichen, mehrere gleichzeitige Verbindungen mit einem Web-Dienst herzustellen, so dass mehrere Web-Dienst-Anfragen gleichzeitig gesendet werden können.

Beispiel: Beim Anfragen von Bankinformationen können Sie die Web-Dienst-Verbraucher-Umwandlung für die Parallelverarbeitung so konfigurieren, dass gleichzeitig mehrere Zeilen gesendet werden. Bei 20 Eingabezeilen können Sie zur schnelleren Verarbeitung gleichzeitig 20 Anfragen senden.

Wenn Sie die Parallelverarbeitung in der Web-Dienst-Verbraucher-Umwandlung aktivieren, können Sie die Obergrenze für den Gesamtspeicherbedarf konfigurieren.

Wenn Sie die Parallelverarbeitung in der Web-Dienst-Verbraucher-Umwandlung aktivieren, können Sie Anzahl gleichzeitiger Web-Dienst-Verbindungen konfigurieren.

Regeln und Richtlinien für die Parallelverarbeitung

Verwenden Sie die folgenden Regeln und Richtlinien bei Einsatz der Parallelverarbeitung:

- Die Parallelverarbeitung unterstützt sortierte Eingabezeilen als mehrere gleichzeitige Verbindungen mit einem Web-Dienst. Sortierte Ausgabezeilen werden nicht unterstützt.
- Verwenden Sie die Parallelverarbeitung, wenn der Datensatz mehr als 100 Zeilen aufweist.

- Es wird davon abgeraten, die Anzahl der gleichzeitigen Web-Dienst-Verbindungen zu erhöhen. Die Anzahl gleichzeitiger Web-Dienst-Verbindungen ist mit der Anzahl der Sockets verknüpft, die vom Betriebssystem verwendet werden. Eine Erhöhung der Anzahl an Sockets ist kostspielig.
- Verwenden Sie Systeme, die über Mehrkernprozessoren mit mindestens 100 MB RAM verfügen, um bei Verwendung der Parallelverarbeitung eine optimale Leistung zu erzielen.
- Die Speicherbedarfsgrenze bei Parallelverarbeitung stellt den Speicherbedarf dar, der von gleichzeitigen Arbeitsabläufen beim Abrufen von Web-Diensten benötigt wird.
- Wenn Sie die Parallelverarbeitung in der Web-Dienst-Verbraucher-Umwandlung aktivieren, können Sie die Speicherbedarfsgrenze konfigurieren. Stellen Sie sicher, dass der Speicherbedarf den physischen Arbeitsspeicher auf dem Server nicht übersteigt.

Bewährte Methoden bei der Parallelverarbeitung

Folgen Sie für eine optimale Leistung bei Verwendung der Parallelverarbeitung folgenden Empfehlungen:

- Vermeiden Sie eine Änderung der Standardwerte für die Obergrenze des Gesamtspeicherbedarfs bei Parallelverarbeitung sowie für die maximale Anzahl von Verbindungen bei Parallelverarbeitung.
- Vermeiden Sie die Verwendung der Parallelverarbeitung für Datensätze mit weniger als 100 Zeilen.
- Vermeiden Sie Pass-Through-Ports im Mapping bei Verwendung der Parallelverarbeitung.

Filteroptimierungen

Die Filteroptimierung erhöht die Leistung, indem die Anzahl der Zeilen, die durch das Mapping geleitet werden, verringert wird. Der Data Integration Service kann die "Early Selection"- oder die Push-Into-Optimierung anwenden.

Wenn der Data Integration Service eine Filteroptimierungsmethode anwendet, verschiebt er einen Filter so nahe wie möglich an die Quelle im Mapping. Wenn der Data Integration Service einen Filter nicht vor eine Umwandlung in einem Mapping verschieben kann, kann er die Filterlogik gegebenenfalls in eine Umwandlung leiten.

Aktivieren der "Early Selection"-Optimierung mit der Webdienst-Verbraucher-Umwandlung

Aktivieren Sie die "Early Selection"-Optimierung für die Webdienst-Verbraucher-Umwandlung, wenn die Umwandlung keine Nebeneffekte hat und keine "Faults" als Fehler behandelt.

1. Öffnen Sie die Ansicht **Erweiterte Eigenschaften** der Web-Dienst-Verbraucher-Umwandlung.
2. Deaktivieren Sie **"Fault" als Fehler behandeln**.
3. Deaktivieren Sie **Hat Nebeneffekte**.

Push-Into-Optimierung mit der Web-Dienst-Verbraucher-Umwandlung

Mit der Push-Into-Optimierung erhält die Web-Dienst-Verbraucher-Umwandlung den Filterwert in einem Filterport. Der Filterport ist ein nicht verbundener Eingabeport, den Sie als Filterport angeben, wenn Sie die Push-Into-Optimierung konfigurieren. Der Filterport verfügt über einen Standardwert, der sicherstellt, dass der

Web-Dienst alle Zeilen zurückgibt, wenn die Endbenutzerabfrage keine Filter enthält. Der Filterport ist kein Übergabeport.

Hinweis: Das Filterfeld muss Teil der Root-Gruppe in der Web-Dienst-Anfrage sein.

Der Filterfeld kann kein Übergabeport sein. Wenn Sie einen Filterport konfigurieren, ändert sich der Standardwert des Ports in den Wert der Filterbedingung, wodurch der Wert des Pass-Through-Ausgabeports geändert wird. Ein Filter, der auf dem Pass-Through-Ausgabeport basiert, gibt unerwartete Ergebnisse zurück.

Sie können mehrere Filterausdrücke in die Web-Dienst-Verbraucher-Umwandlung verschieben. Jede Filterbedingung muss das folgende Format aufweisen:

```
<Field> = <Constant>
```

Die Filterbedingungen müssen durch AND verbunden werden. Die Bedingungen können nicht durch OR verbunden werden.

Beispiel für eine Push-Into-Optimierung mit der Web-Dienst-Verbraucher-Umwandlung

Ein SQL-Datendienst gibt Aufträge für alle Kunden oder Aufträge für einen bestimmten Kunden basierend auf der SQL-Abfrage zurück, die er vom Benutzer erhält.

Der Datendienst enthält ein logisches Datenobjekt mit den folgenden Komponenten:

Kundentabelle

Eine Oracle-Datenbanktabelle, die Kundeninformationen enthält.

Web-Dienst-Verbraucher-Umwandlung

Eine Umwandlung, die einen Web-Dienst aufruft, um die aktuellen Kundenaufträge abzurufen. Die Web-Dienst-Verbraucher-Umwandlung verfügt über Ports für die Kunden-ID (customerID) oder die Auftragsnummer (orderNum). Die Umwandlung verfügt über Übergabeports mit Kundendaten, die aus der Kundentabelle stammen. Der Port für die Auftragsnummer (orderNum) fungiert als Filterport und ist nicht verbunden. orderNum weist den Standardwert "*" auf. Wenn der Web-Dienst diesen Wert in der Web-Dienst-Anfrage empfängt, werden alle Aufträge zurückgegeben.

Virtuelle Tabelle für Aufträge

Hierbei handelt es sich um eine virtuelle Tabelle, die Kunden- und Auftragsdaten vom Webdienst empfängt. Der Endbenutzer verwendet diese Tabelle für Anfragen. Die Tabelle für Aufträge enthält eine Spalte für Kunden, eine Spalte für die Kunden-ID sowie Kunden- und Auftragsdaten.

Der Endbenutzer übergibt die folgenden SQL-Abfrage an den SQL-Datendienst:

```
SELECT * from OrdersID where customer = 23 and orderID = 56
```

Der Data Integration Service teilt die Abfrage auf, um das Mapping zu optimieren. Der Data Integration Service verwendet "Early Selection"-Optimierung und verschiebt die Filterlogik `customer = 23` in die gelesenen Kundendaten. Der Data Integration Service verwendet Push-Into-Optimierung und verschiebt die Filterlogik `orderID = 56` in den Filterport der Web-Dienst-Verbraucher-Umwandlung. Die Web-Dienst-Verbraucher-Umwandlung ruft die Auftrags-ID 56 für den Kunden 23 ab.

Aktivieren der Push-Into-Optimierung mit der Web-Dienst-Verbraucher-Umwandlung

Aktivieren Sie die Push-Into-Optimierung für die Webdienst-Verbraucher-Umwandlung, wenn die Umwandlung keine Nebeneffekte hat und keine "Faults" als Fehler behandelt.

1. Öffnen Sie die Ansicht **Erweiterte Eigenschaften** der Web-Dienst-Verbraucher-Umwandlung.

2. Deaktivieren Sie **"Fault" als Fehler behandeln**.
3. Deaktivieren Sie **Hat Nebeneffekte**.
4. Klicken Sie auf die Schaltfläche **Öffnen** in der Eigenschaft **Push-Into-Optimierung**.
5. Wählen Sie den Namen des Filterports im Dialogfeld "Optimierte Eingabe" aus.
   Sie können mehrere Filterports auswählen.
6. Klicken Sie auf die Spalte **Ausgabe**.
7. Wählen Sie für jeden Filterport den Ausgabeport aus, der die gefilterte Spalte in der Web-Dienst-Antwort enthält.
8. Geben Sie einen Standardwert für jeden Filterport ein.

   Hinweis: Sie können einen Standardwert für einen Web-Dienst-Verbraucher-Port konfigurieren, es sei denn, es handelt sich um einen Filterport.

Erstellen einer Web-Dienst-Verbraucher-Umwandlung

Sie können eine wiederverwendbare oder nicht wiederverwendbare Web-Dienst-Verbraucher-Umwandlung erstellen. Wiederverwendbare Umwandlungen können in mehreren Mappings vorhanden sein. Nicht wiederverwendbare Umwandlungen sind in einem einzelnen Mapping vorhanden.

Sie können Web-Dienst-Verbraucher-Umwandlungen für eine SOAP 1.1- und eine SOAP 1.2-Bindung aus einem einzelnen WSDL-Objekt erstellen.

1. Mithilfe einer der folgenden Methoden können Sie eine Umwandlung erstellen:

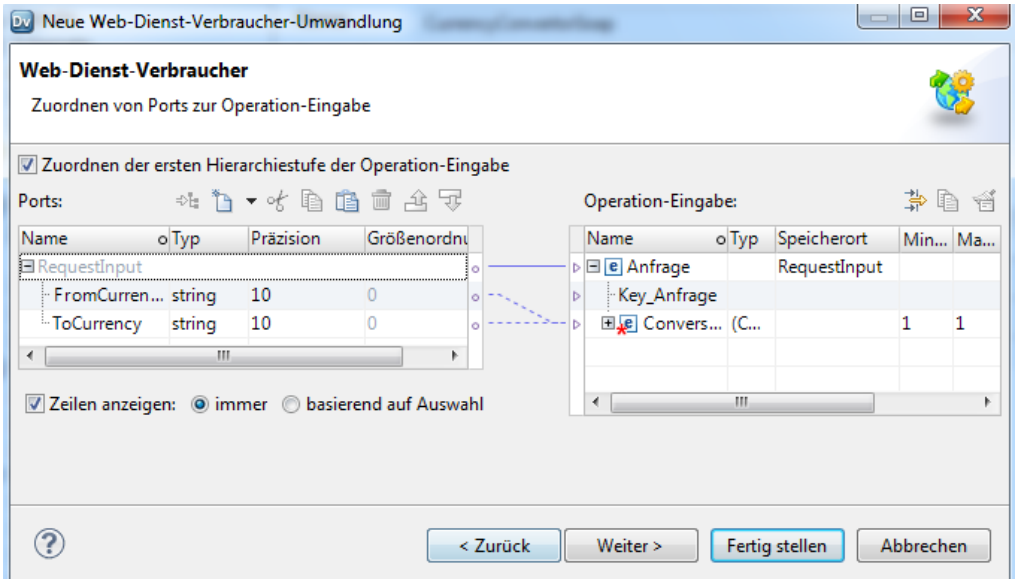
Option	Beschreibung
Wiederverwendbar	Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus. Klicken Sie auf **Datei** > **Neu** > **Umwandlung**. Wählen Sie die Web-Dienst-Verbraucher-Umwandlung aus und klicken Sie auf **Weiter**.
Nicht wiederverwendbar	Ziehen Sie in einem Mapping oder Mapplet eine Web-Dienst-Verbraucher-Umwandlung aus der Umwandlungspalette in den Editor.

   Das Dialogfeld **Neue Web-Dienst-Verbraucher-Umwandlung** wird angezeigt.
2. Suchen und wählen Sie ein WSDL-Datenobjekt, um die Anfrage- und Antwortmeldungen des Webdienstes zu definieren.

   Wenn sich die WSDL nicht im Repository befindet, können Sie sie aus dem Dialogfeld „Neue Web-Dienst-Verbraucher-Umwandlung“ importieren.
3. Suchen und wählen Sie einen Vorgang aus der WSDL.

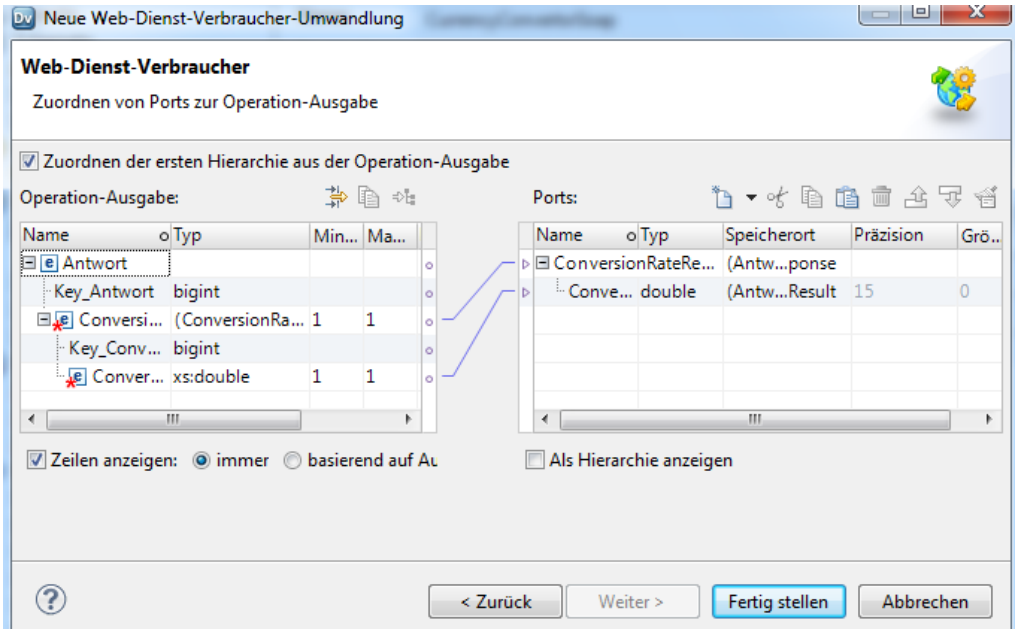
   Sie können eine Operation auswählen, die über eine SOAP 1.1- oder SOAP 1.2-Bindung verfügt.
4. Klicken Sie auf **Weiter**.

Der Bildschirm **Ports auf Vorgangseingabe abbilden** erscheint. Im Bereich **Ports** werden die Eingabegruppen und Eingabeports der Umwandlung angezeigt. Im Bereich **Vorgangseingabe** wird die Anfragemeldungshierarchie angezeigt.



5. Definieren Sie die Eingabegruppen und Eingabeports und ordnen Sie die Eingabeports den Vorgangseingabeknoten zu.
6. Klicken Sie auf **Weiter**.

   Der Bildschirm **Ports von Vorgangsausgabe abbilden** erscheint. Im Bereich **Vorgangsausgabe** wird die Antwortmeldungshierarchie angezeigt. Im Bereich **Ports** werden die Umwandlungs-Ausgabegruppen und -Ausgabeports angezeigt.



7. Definieren Sie die Ausgabegruppen und Ausgabeports und ordnen Sie die Vorgangsausgabeknoten den Ausgabeports zu.
8. Klicken Sie auf **Fertig stellen**.

9. Klicken Sie auf die Ansicht **Erweitert**, um die Umwandlungseigenschaften und die Webdienstverbindung zu konfigurieren.

Verwandte Themen:

- "Zuordnen von Eingabeports zur Vorgangseingabe" auf Seite 606
- "Zuordnen der Vorgangsausgabe zu Ausgabeports" auf Seite 610
- "Erweiterte Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung" auf Seite 611

Web-Dienst-Verbraucher-Umwandlung - Beispiel

Ihr Unternehmen muss der Vertriebsabteilung Auftragsinformationen für die RT100-Produktlinie zur Verfügung stellen. Das Verkaufsteam muss die Auftragszusammenfassung und die Auftragsdetails täglich abrufen.

Erstellen Sie ein logisches Datenobjekt, das die täglichen Auftragsinformationen in virtuellen Tabellen zusammenfasst. Das Lese-Mapping enthält eine Web-Dienst-Verbraucher-Umwandlung, die die neuesten RT100-Aufträge zurückgibt. Die Web-Dienst-Verbraucher-Umwandlung verwendet einen Web-Dienst, der Informationen zur täglichen Auftragszusammenfassung und zu Auftragsdetails für die RT100-Produktlinie zurückgibt.

Eingabedatei

Die Eingabedatei ist eine Einfachdatei, die die Produktliniennummer enthält.

Erstellen Sie ein physisches Datenobjekt zum Definieren der Eingabedatei. Die Datei hat ein Feld: Product_Line. Das Feldwert ist RT100. Definieren Sie den Speicherort des physischen Datenobjekts in der Ansicht **Laufzeiteigenschaften**.

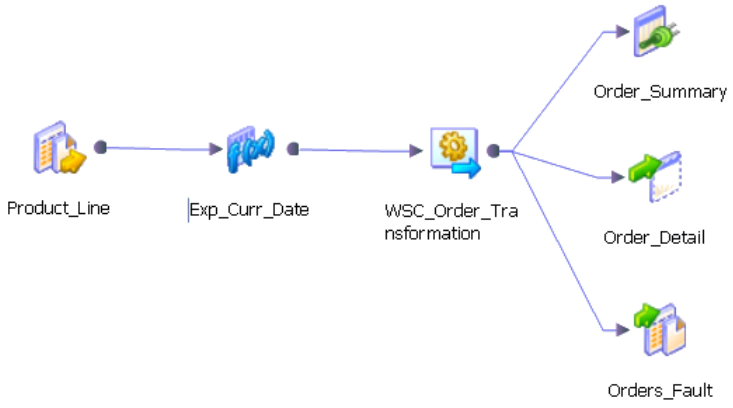
Logisches Datenobjektmodell

Ein Wirtschaftsanalytiker in Ihrem Unternehmen erstellt ein logisches Datenmodell, das die Reihenfolgenzusammenfassung und die Strukturen der Reihenfolge der Detailtabelle beschreibt. Das logische Datenmodell enthält die logischen Order_Summary- und Order_Detail-Datenobjekte.

Der Analytiker erstellt ein Schema in einem Modellierungstool, das das logische Datenmodell definiert. Importieren Sie das logische Datenmodell aus dem Schema und erstellen Sie die logischen Order_Summary- und Order_Detail-Datenobjekte.

Mapping logischer Datenobjekte

Das Mapping von logischen Datenobjekten beschreibt den Zugriff auf Daten über das logische Datenobjekt.



Das Lese-Mapping enthält die folgenden Objekte:

Product_Line

Die Eingabe-Einfachdatei, die die Produktliniennummer enthält.

Exp_Curr_Date-Umwandlung

Die Ausdrucksumwandlung, die das aktuelle Datum und einen primären Schlüssel für die Eingabegruppe auf Root-Ebene in der Web-Dienst-Verbraucherumwandlung zurückgibt.

WSC_Order-Umwandlung

Web-Dienst-Verbraucherumwandlung, die einen Web-Dienst zum Abrufen von Bestellinformationen verwendet. Die Umwandlung übergibt die Produktlinie und das aktuelle Datum an den Web-Dienst in der Anfragenachricht. Die Umwandlung empfängt Bestellinformationen vom Web-Dienst in der Antwortnachricht.

Order_Summary-Tabelle

Ein logisches Datenobjekt, das die Bestellinformationen enthält, wie zum Beispiel Order_No, Customer_Id, Qty und Order_Date.

Order_Detail-Tabelle

Ein logisches Datenobjekt, das die Bestelldetailinformationen enthält, wie zum Beispiel Order_No, Product_Id, Qty und Status.

Orders_Fault

Ausgabe-Einfachdatei, die Meldungen über generische Fehler enthält.

Web-Dienst-Verbraucher-Umwandlung

Die Web-Dienst-Verbraucher-Umwandlung erhält eine Produktlinie, Daten und eine Sequenznummer als Eingabe. Die Umwandlung verbraucht die Web-Dienst-Operation "Get_Order_Info" (Auftragsinfo_abrufen), um Auftragsinformationen abzurufen.

Wählen Sie beim Erstellen der Web-Dienst-Verbraucher-Umwandlung ein WSDL-Datenobjekt aus, das die Web-Dienst-Anfrage- und Web-Dienst-Antwortmeldungen beschreibt. Eine Web-Dienst-Meldung enthält hierarchische Gruppen von XML-Elementen. Ein Element kann andere Elemente enthalten. Bestimmte Elemente treten mehrfach auf. Erstellen Sie die Umwandlung aus dem WSDL-Objekt "Order_Info" (Auftragsinfo) im Repository.

Konfigurieren Sie die Umwandlungs-Eingabeports und weisen Sie die Ports der Operationseingabehierarchie zu. Ordnen Sie den Ausgabeports Knoten aus der Operationsausgabehierarchie zu. Definieren Sie die Web-Dienst-Verbindung und Laufzeiteigenschaften.

Umwandlungseingabe-Mapping

Wenn Sie das Eingabe-Mapping in der Ansicht **Ports** anzeigen, können Sie Eingabeports definieren und ihnen Knoten in der Operationseingabe zuordnen.

Der Bereich **Ports** weist eine Root- und eine Auftragsgruppe auf. Bei der Root-Gruppe handelt es sich um eine Anfrageeingabegruppe. Fügen Sie der Eingabegruppe "Anfrage" einen Port hinzu, um den Primärschlüssel darzustellen.

Die Gruppe "Auftrag" verfügt über die Eingabeports **Datum_Auswählen** und **Produktlinie_Auswählen**.

Ordnen Sie den Knoten **Auftragsdatum** and **Produktlinie** im Bereich **Operationseingabe** die Eingabeports zu.

Der Bereich **Operationseingabe** definiert die Anfragemeldung, die von der Web-Dienst-Verbraucher-Umwandlung an den Web-Dienst übergeben wird. Die Knoten werden standardmäßig im Bereich **Operationseingabe** angezeigt.

Umwandlungsausgabe-Mapping

Wenn Sie das Ausgabe-Mapping in der Ansicht **Ports** anzeigen, können Sie die Ausgabeports definieren, indem Sie den Umwandlungsausgabegruppen Knoten aus der Operationsausgabe zuweisen.

Der Web-Dienst gibt die folgende Hierarchie in einer Web-Dienst-Antwortmeldung zurück:

```
Response
   Orders
      Order
         Key_Order
         Order_ID
         Order_Date
         Customer_ID
         Total_Qty
         Order_Details
            Order_Detail
               Product_ID
               Description
               Qty
               Status
```

Der Web-Dienst gibt mehrere Aufträge zurück. Auftrag ist ein mehrfach vorkommender Knoten auf der Ebene "Aufträge". Für jeden Auftrag kann der Web-Dienst mehrere Auftragsdetails zurückgeben. Auftragsdetails ist ein mehrfach vorkommender Knoten auf der Ebene "Auftragsdetails".

Hinweis: Das Developer Tool fügt der Benutzeroberfläche den Knoten "Hauptauftrag" hinzu. Sie können den Ausgabegruppen den Schlüssel zuweisen, um Beziehungen zwischen Gruppen zu definieren. In diesem Beispiel fungiert die Auftrags-ID als Primärschlüssel im Auftrag und als Fremdschlüssel in den Auftragsdetails.

Erstellen Sie die folgenden Ausgabegruppen im Bereich **Ports**:

```
Order
    Order_ID
    Order_Date
    Customer_ID
    Total_Qty

Order_Detail
    Order_ID
    Product_ID
```

```
Description
Qty
Status
```

Der Data Integration Service schreibt eine Zeile aus der Gruppe "Auftrag", sobald der Wert der Auftrags-ID geändert wird.

Der Data Integration Service schreibt eine Zeile aus der Gruppe "Auftragsdetails", sobald die Werte der Auftrags-ID und Produkt-ID geändert werden.

Umwandlung - Erweiterte Eigenschaften

Konfigurieren Sie die folgenden erweiterten Eigenschaften für die Web-Dienst-Verbraucher-Umwandlung:

Generische SOAP-Fehlerbehandlung aktivieren

Fügt Ausgabeports hinzu, die SOAP-Fehlermeldungen empfangen.

Verbindung

Wählen Sie eine Web-Dienst-Verbindung aus, um auf den Web-Dienst zuzugreifen.

Kompression aktivieren

Die Web-Dienst-Verbraucher-Umwandlung komprimiert Web-Meldungen mit GZIP.

KAPITEL 44

Parsen von Webdienst-SOAP-Meldungen

Dieses Kapitel umfasst die folgenden Themen:

- Parsen von Webdienst-SOAP-Meldungen – Übersicht, 623
- Benutzeroberfläche für Umwandlungen, 624
- Konfigurieren von Mehrfachausgaben, 625
- Parsen von anyType-Elementen, 627
- Parsen von abgeleiteten Typen, 628
- Parsen von QName-Elementen, 629
- Parsen von Substitutionsgruppen, 630
- Parsen von XML-Konstrukten in SOAP-Meldungen, 630

Parsen von Webdienst-SOAP-Meldungen – Übersicht

Beim Parsen von SOAP-Meldungen in einer Web-Dienst-Umwandlung durch Data Integration Service werden Rohdaten generiert.

Die Web-Dienst-Eingabe-Umwandlung und die Web-Dienst-Verbraucher-Umwandlung sind Web-Dienst-Umwandlungen, bei denen SOAP-Meldungen geparst werden.

Um eine Umwandlung zum Parsen einer SOAP-Meldung zu konfigurieren, erstellen Sie Ausgabeports in einer der SOAP-Meldungshierarchie nachempfundenen Struktur. Ordnen Sie die Knoten in der SOAP-Meldungshierarchie den Ports zu.

Sie können normalisierte Gruppen von Ausgabeports, denormalisierte Gruppen und pivotierte Gruppen von Ports konfigurieren. Wenn die SOAP-Meldung abgeleitete Typen, anyType-Elemente oder Substitutionsgruppen enthält, können Sie – je nachdem, welche Typen in der SOAP-Meldungsinstanz auftreten – verschiedene Ausgabegruppen konfigurieren.

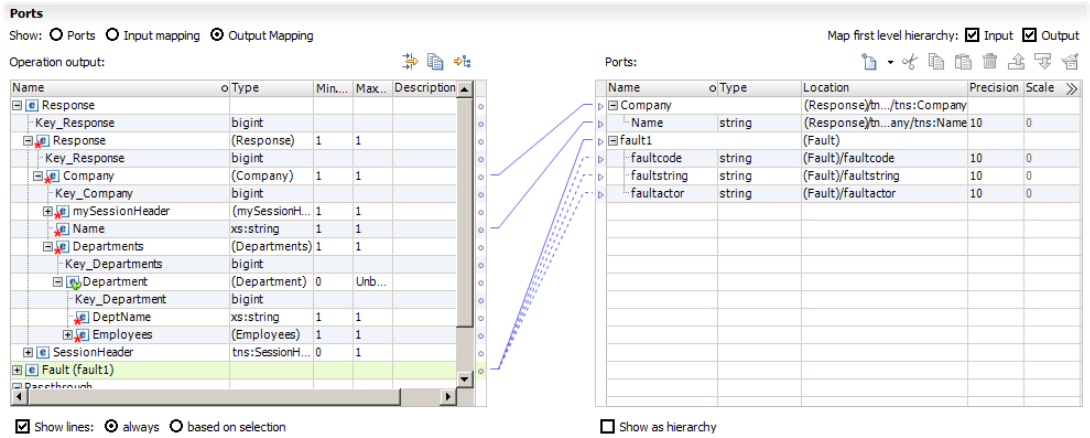
Verwandte Themen:

- "REST-Webdienst-Verbraucher-Umwandlung - Ausgabe-Mapping" auf Seite 607

Benutzeroberfläche für Umwandlungen

Die Web-Dienst-Verbraucher-Umwandlung und die Web-Dienst-Eingabe-Umwandlung bieten eine Benutzeroberfläche, in der Sie Daten aus der SOAP-Meldung den Ausgabeports für die Umwandlung zuordnen können.

Die folgende Abbildung zeigt ein Mapping zwischen SOAP 1.1-Meldungsknoten und Ausgabeports in einer Web-Dienst-Verbraucher-Umwandlung:



Operationsbereich

Der Operationsbereich enthält die SOAP-Meldungshierarchie. Die Hierarchieebenen der Struktur werden von komplexen Knoten oder mehrfach vorkommenden Knoten definiert. Das Developer Tool fügt den Ebenen Schlüssel hinzu, um die Beziehungen zwischen über- und untergeordneten Elementen festzulegen.

In der vorherigen Abbildung weist die SOAP-Meldungshierarchie folgende Ebenen auf:

Response bzw. Request

Stammebene der Antwort- oder Anfragemeldung

Firma

Die oberste Ebene der Anfragedaten

Departments

Mehrfach vorkommende Abteilungen innerhalb des Unternehmens

Employees

Komplexes Element für die Mitarbeiter innerhalb einer Abteilung

Fehlergruppe

Fehlermeldungsgruppe, die Fehlermeldungen erhält

Ports-Bereich

Sie können die Daten aus den Ebenen der SOAP-Meldung den Ausgabeports zuordnen. Alle Gruppen von Ausgabeports können über primäre Fremdschlüsselbeziehungen mit weiteren Ausgabegruppen verbunden werden.

In der vorherigen Abbildung weist die Umwandlung Gruppen von Ausgabeports auf, die den Knotengruppen in der SOAP-Meldung entsprechen.

Konfigurieren von Mehrfachausgaben

Bei Eingabe- oder Web-Dienst-Verbraucher-Umwandlungen mit mehrfach vorkommenden Daten können Ausgabeports unterschiedlich konfiguriert werden.

Sie können normalisierte Ausgabedaten, pivotierte Ausgabedaten oder denormalisierte Ausgabedaten konfigurieren.

Eine SOAP-Meldung enthält beispielsweise die komplexen Elemente „Departments“ (Abteilung) und „Employees“ (Mitarbeiter). Jede Abteilung enthält mehrere Mitarbeiter. Die Abteilungsebene ist der Mitarbeiterebene übergeordnet.

Die SOAP-Meldung weist folgende Elementhierarchie auf:

```
Departments
      Department_ID
      Department_Name
      Employees
         Employee_ID
         Employee_Name
```

Normalisierte relationale Ausgabe

Beim Erstellen normalisierter Ausgabedaten werden die Datenwerte in einer Ausgabegruppe nicht wiederholt. Sie erstellen eine 1:1-Beziehung zwischen den Hierarchieebenen in der SOAP-Meldung und den Port-Ausgabegruppen.

Wenn die SOAP-Meldung eine übergeordnete Ebene „Departments“ und eine untergeordnete Ebene „Employees“ enthält, könnten Sie beispielsweise folgende Port-Gruppen erstellen:

```
Departments
    Department_Key
    Department_ID
    Department_Name


Employees
    Department_Key
    Employee_ID
    Employee_Name
```

„Department_Key“ ist ein generierter Schlüssel, der die Ausgabegruppe „Employees“ mit der Gruppe „Departments“ verknüpft.

Generierte Schlüssel

Beim Hinzufügen einer Ausgabegruppe verbindet das Developer Tool die Ausgabegruppe mithilfe eines generierten Schlüssels mit einer anderen Ausgabegruppe. Das Developer Tool fügt der übergeordneten und

untergeordneten Gruppe einen bigint-Schlüssel hinzu. Data Integration Service erstellt die Schlüsselwerte für die generierten Schlüssel zur Laufzeit.

Beispiel

Die SOAP-Hierarchie weist folgende Knoten auf:

```
Departments
   Dept_Key
   Dept_Num
   Dept_Name

   Employees
      Dept_FK
      Employee_Num
      Employee_Name
```

Beim Erstellen einer Ausgabegruppe von Ports für „Departments“ ordnen Sie den Knoten „Departments“ einem leeren Feld im Ports-Bereich zu. Das Developer Tool erstellt die folgende Ausgabegruppe:

```
Departments
   Dept_Num
   Dept_Name
```

Beim Zuordnen des Knotens „Employees“ zu einem leeren Feld im Ports-Bereich werden Sie vom Developer Tool aufgefordert, eine Verknüpfung zwischen den Gruppen „Employees“ und „Departments“ zu definieren. Sie können die Gruppe „Employees“ mehreren Gruppen zuordnen. Das Developer Tool fügt jeder Gruppe einen Schlüssel hinzu.

Das Developer Tool erstellt folgende Gruppen und generierte Schlüssel:

```
Departments
   Key_Departments
   Dept_Num
   Dept_Name

   Employees
      Key_Departments
      Employee_Num
      Employee_Name
```

Hinweis: Die Knoten müssen den generierten Schlüsseln nicht zugeordnet werden. Data Integration Service erstellt die Schlüsselwerte zur Laufzeit.

Das Developer Tool kann in einer Ausgabegruppe generierte Schlüssel für mehrere Ebenen erstellen. Die Gruppe „Employees“ kann beispielsweise folgende Ports enthalten:

```
Employees
   Key_Employees
   Key_Departments
   Key_Managers
   Employee_Num
   Employee_Name
```

„Key_Departments“ und „Key_Managers“ sind die generierten Schlüssel, die auf übergeordnete Gruppen verweisen. „Key_Employees“ ist ein generierter Schlüssel für die Gruppe „Employees“. „Key_Employees“ wird erstellt, wenn Sie der Gruppe „Employees“ eine untergeordnete Gruppe zuweisen.

Denormalisierte relationale Ausgabe

Sie können relationale Ausgaben denormalisieren. Dabei werden die Elementwerte aus der übergeordneten Gruppe für jedes untergeordnete Element wiederholt.

Zum Denormalisieren von Ausgabedaten ordnen Sie Knoten aus der übergeordneten Hierarchieebene der untergeordneten Gruppe von Ausgabeports zu.

Das folgende Beispiel zeigt die Elemente „Department_ID“ und the „Department_Name“ in der Ausgabegruppe „Employees“:

```
Employees
    Department_ID
    Department_Name
    Employee_ID
    Employee_Name
```

„Department_ID“ und „Department_Name“ werden für jeden Mitarbeiter in der Abteilung wiederholt:

Department_ID	Department_Name	Employee_ID	Employee_Name
100	Accounting	56500	Kathy Jones
100	Accounting	56501	Tom Lyons
100	Accounting	56509	Bob Smith

Pivotierte relationale Ausgabe

Sie können eine bestimmte Anzahl mehrfach vorkommender Elemente in eine Ausgabegruppe aufnehmen.

Um mehrfach vorkommende Elemente zu pivotieren, ordnen Sie die mehrfach vorkommenden untergeordneten Elemente der übergeordneten Gruppe von Ausgabeports zu. Das Developer Tool fordert Sie auf, die Anzahl der untergeordneten Elemente festzulegen, die in die übergeordnete Gruppe aufgenommen werden sollen.

Das folgende Beispiel zeigt zwei Instanzen von „Employee_ID“ in der übergeordneten Gruppe „Departments“:

```
Departments
    Department_ID
    Department_Name
    Employee_ID1
    Employee_ID2
```

Parsen von anyType-Elementen

Das anyType-Element stellt eine Auswahl aller globalen Typen in WSDL bzw. in einem Schema dar. Beim Zuordnen von Knoten und Ports im Developer Tool wählen Sie die Typen aus, die in der SOAP-Meldung für das anyType-Element erscheinen sollen. Sie müssen ein anyType-Element in der SOAP-Meldung durch einen komplexen Typ oder durch xs:string ersetzen. Erstellen Sie Gruppen von Ports für jeden ausgewählten Typ.

Sie müssen einen Typ zum Zuweisen von Daten zu Ausgabeports auswählen. Wenn weder die WSDL noch das Schema einen globalen Typ enthält, ersetzt das Developer-Tool das anyType-Element durch xs:string.

Um einen komplexen Elementtyp im Operationsbereich auszuwählen, klicken Sie in der Spalte **Typ** des anyType-Elements auf **Auswählen**. Eine Liste der verfügbaren komplexen Typen und xs:string wird eingeblendet.

Wenn Sie ein anyType-Element durch abgeleitete Typen ersetzen, vervollständigt Data Integration Service jeweils die Elemente eines Typs. Die SOAP-Meldung enthält nicht gleichzeitig Daten für den Basistyp und für den abgeleiteten Typ.

Beispiel für abgeleitete Typen

WSDL enthält ein anyType-Element. Sie ersetzen das Element durch „AddressType“ und einen abgeleiteten Typ namens „USAddressType“. Die SOAP-Meldungshierarchie enthält folgende Gruppen:

```
Address:AddressType (base type)
  Address: AddressType
    Street
    City

  Address:USAddressType   (derived type)
     Street
     City
     State
     ZipCode
```

Die SOAP-Meldung enthält folgende Daten:

```
<address xsi: type ="AddressType">
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>
```

Data Integration Service gibt eine Zeile für „xsi: AddressType“ zurück:

Straße	Ort
1002 Mission St.	San Jose

Data Integration Service gibt eine Zeile für den abgeleiteten Typ „xsi: USAddressType“ zurück:

Straße	Ort	Bundesland	PLZ
234 Fremont Blvd.	Sunnyvale	CA	94556

Data Integration Service liefert keine Daten für „AddressType“, wenn der Typ mit „xsi: USAddressType“ festgelegt ist.

Parsen von abgeleiteten Typen

Sie können SOAP-Meldungen parsen, die abgeleitete Typen enthalten. Beim Definieren der Ports, die Daten aus der SOAP-Meldung empfangen, wählen Sie, welche Typen in einer SOAP-Meldung erscheinen können. Die Elemente in den gewählten Typen bestimmen die Ports, die Sie erstellen müssen.

WSDL kann beispielsweise den Adresstyp „AddressType“ und einen abgeleiteten Typ namens „USAddressType“ enthalten. Im Operationsbereich des Developer Tool können Sie folgende Gruppen erstellen:

```
Address
  Address: AddressType
    Street
    City

  Address:USAddressType
```

```
        Street
        City
        State
        ZipCode
```

Die SOAP-Meldung kann folgende Daten enthalten:

```
<address>
<street>1002 Mission St.</street>
<city>san jose</city>
</address>

<address xsi:type="USAddressType">
<street>234 Fremont Blvd</street>
<city>Fremont</city>
<zip>94556</zip>
<state>CA</state>
</address>

<address xsi:type="USAddressType">
<street>100 Cardinal Way</street>
<city>Redwood City</city>
<zip>94536</zip>
<state>CA</state>
</address>

<address>
<street>100 El Camino Real</street>
<city>Sunnyvale</city>
</address>
```

Data Integration Service gibt für den Basistyp (AddressType) folgende Zeilen zurück:

Straße	Ort
1002 Mission St.	San Jose
234 Fremont Blvd	Sunnyvale
100 Cardinal Way	Redwood City
100 El Camino Real	Sunnyvale

Data Integration Service gibt für den abgeleiteten Typ „USAddressType“ folgende Zeilen zurück:

Straße	Ort	Bundesland	PLZ
234 Fremont Blvd.	Sunnyvale	CA	94556
100 Cardinal Way	Redwood City	CA	94536

Data Integration Service gibt alle Adressen im Basistyp zurück. Data Integration Service gibt die US-Adressen im abgeleiteten Typ zurück. Der abgeleitete Typ enthält die Elemente für Straße und Ort (Street, City), die „USAddressType“ aus dem Basistyp erbt.

Parsen von QName-Elementen

Wenn der Data Integration Service QName-Elemente in der SOAP-Meldung parst, werden die QName-Werte aktualisiert, die zu dem Namespace des Schemas gehören, um das im Schema definierte Namespace-Präfix zu verwenden. Andernfalls wird der Wert des Elements nicht von Data Integration Service aktualisiert.

Das Schema weist beispielsweise das Namespace-Präfix `tns` auf, das für den Namespace `"http://user/test"` definiert ist. Die SOAP-Meldung enthält das Namespace-Präfix `mytns`, das für denselben Namespace definiert ist. Wenn der Data Integration Service den QName-Wert `mytns:myelement` parst, wird der Wert in `tns:myElement` geändert.

Wenn der Data Integration Service QName-Elemente in der SOAP-Meldung generiert, wird der Wert des Elements nicht aktualisiert.

Parsen von Substitutionsgruppen

Eine Substitutionsgruppe ersetzt ein Element durch ein anderes aus derselben Gruppe. Substitutionsgruppen ähneln abgeleiteten Typen, jedoch mit der Ausnahme, dass jede Elementdefinition auch einen Substitutionsgruppennamen aufweist.

Sie können eine Ausgabegruppe von Ports konfigurieren, die Elemente aus einem bestimmten Typ in einer Substitutionsgruppe enthält. Sie können eine andere Port-Ausgabegruppe erstellen, die ein Element aus einem anderen Typ in der Substitutionsgruppe übernimmt.

Parsen von XML-Konstrukten in SOAP-Meldungen

SOAP-Meldungen können XML-Konstrukte wie Auswahl-, Listen- und Union-Elemente enthalten.

Das Parsen solcher Konstrukte ist in Web-Dienst-Umwandlungen mit einigen Einschränkungen möglich.

Auswahlelement

Auswahlelemente beschränken untergeordnete Elemente auf ein Element in der Deklaration „<choice>“.

Der folgende Text zeigt das Element „Person“, das einen Mitarbeiter (Employee) oder einen Zulieferer (Contractor) bezeichnet:

```
<xs:element name="person">
  <xs:complexType>
    <xs:choice>
      <xs:element name="employee" type="employee"/>
      <xs:element name="contractor" type="contractor"/>
    </xs:choice>
  </xs:complexType>
</xs:element>
```

Sie können Auswahlelemente mithilfe der folgenden Methoden zuordnen:

- Erstellen Sie Ausgabeports für jedes Auswahlelement in einer Ausgabegruppe. Einige Elemente werden Nullwerte in der Ausgabezeile aufweisen.
- Erstellen Sie eine Ausgabegruppe für jede Auswahl. Beim Beispiel oben würden Sie eine Employee-Gruppe und eine Contractor-Gruppe erstellen. Data Integration Service generiert eine Zeile abhängig davon, welches Element in der SOAP-Meldung erscheint.

Listenelement

Eine Liste ist ein XML-Element, das mehrere Einfachtypwerte enthalten kann, z. B. "Montag Dienstag Mittwoch".

Data Integration Service kann Listen als String-Wert zurückgeben. Wenn die SOAP-Meldung eine Liste enthält, können Elemente aus der Liste nicht separaten Ausgabezeilen zugeordnet werden. Sie können eine Ausdrucksumwandlung konfigurieren, um die Elemente in der Liste zu trennen, wenn Sie für ein Mapping separate Elemente benötigen.

Union-Element

Union-Elemente sind Einfachtypen, bestehend aus einer Vereinigung mehrerer Typen.

Der folgende Text zeigt das Element „Größe“ (Size), eine Union aus den zwei Einfachtypen „size_no“ und „size_string“:

```
<xs:element name="Size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

Um die Größe einem Ausgabeport zuzuordnen, erstellen Sie einen Port für die Größe. Konfigurieren Sie den Ausgabeport als String. Sie können im Mapping eine weitere Umwandlung konfigurieren, um die Daten in einen anderen Typ zu konvertieren.

KAPITEL 45

Generieren von Webdienst-SOAP-Meldungen

Dieses Kapitel umfasst die folgenden Themen:

- Generieren von Webdienst-SOAP-Meldungen – Übersicht, 632
- Benutzeroberfläche für Umwandlungen, 633
- Beziehungen zwischen Port und Hierarchieebene , 635
- Schlüssel, 636
- Zuordnen von Ports, 637
- Pivotieren mehrfach vorkommender Ports , 639
- Zuordnen denormalisierter Daten, 640
- Abgeleitete Typen und Elementsubstitution, 642
- Generieren von XML-Konstrukten in SOAP- Meldungen, 643

Generieren von Webdienst-SOAP-Meldungen – Übersicht

Beim Generieren einer SOAP-Meldung erstellt Data Integration Service XML-Daten aus Gruppen von Eingabedaten. Wenn Sie eine Web-Dienst-Verbraucher-Umwandlung, eine Web-Dienst-Ausgabeumwandlung oder eine Fehlerumwandlung erstellen, legen Sie die Zuordnung zwischen den Eingabeports und der SOAP-Meldungshierarchie fest.

Um eine Umwandlung zum Generieren einer SOAP-Meldung zu konfigurieren , erstellen Sie Gruppen von Eingabeports und ordnen diese den Gruppen in der SOAP-Meldungshierarchie zu. Eine WSDL bzw. ein Schema definiert die Struktur der SOAP-Meldung.

Sie können Gruppen von Daten in der SOAP-Meldung aus denormalisierten Eingabedaten konfigurieren. Sie können auch mehrfach vorkommende Eingabedaten auf mehrfach vorkommende Knoten in der SOAP-Meldung pivotieren.

In einer SOAP-Meldung können Daten abgeleiteten Typen, anyType-Elementen oder Substitutionsgruppen zugeordnet werden. Sie müssen beim Definieren der Umwandlung festlegen, welche Typen in der SOAP-Meldung auftreten können. Die gewählten Typen bestimmen, welche Eingabeports Sie erstellen müssen.

Wenn Sie die SOAP-Meldungshierarchie im Developer-Tool anzeigen, enthält sie Schlüssel. Diese Schlüssel erscheinen in der SOAP-Meldung nicht. Die Schlüssel dienen Data Integration Service dazu, die Beziehungen

zwischen über- und untergeordneten Gruppen in der SOAP-Meldung festzulegen. Zum Konfigurieren der Schlüsselwerte ordnen Sie in der SOAP-Meldung die Eingabedaten den Schlüsseln zu.

Verwandte Themen:

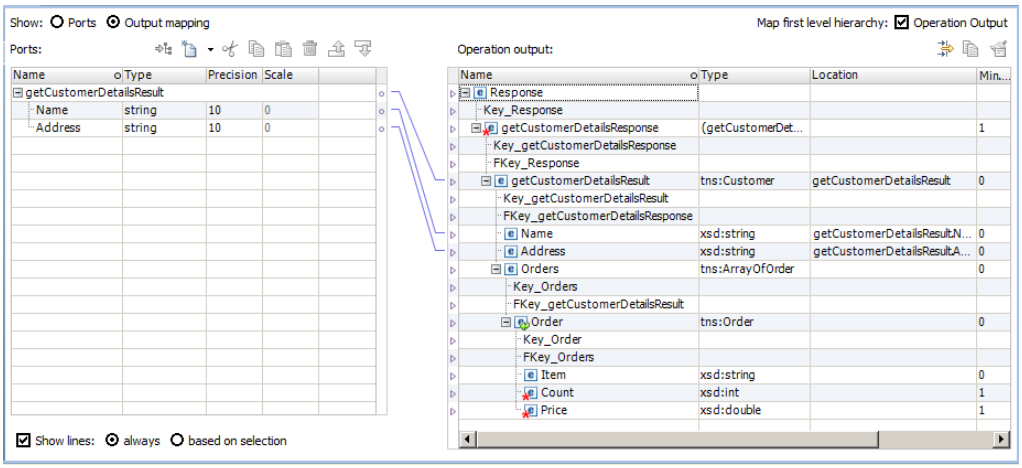
- "Web-Dienst-Verbraucher-Umwandlung - Eingabemapping" auf Seite 604

Benutzeroberfläche für Umwandlungen

Die Webdienst-Ausgabeumwandlung, Fehlerumwandlung und Webdienst-Verbraucher-Umwandlung bieten Benutzeroberflächen, die Sie zum Konfigurieren der SOAP-Meldung nutzen können.

Um eine Umwandlung zum Generieren einer SOAP-Meldung zu konfigurieren, erstellen Sie Eingabeports in einer der SOAP-Meldungshierarchie nachempfundenen Struktur. Die WSDL bzw. das Schema definiert die Struktur der Hierarchie. Ordnen Sie jeden Eingabeport einem Knoten in der SOAP-Meldung zu.

Die folgende Abbildung zeigt ein Mapping zwischen Eingabeports und SOAP-Meldungsknoten in einer Webdienst-Ausgabeumwandlung:



Eingabeports, Bereich

Im Bereich **Eingabeports** können Sie Gruppen von Eingabeports erstellen. Dabei erstellen Sie Eingabeports für jede Ebene in der SOAP-Meldungshierarchie, die zugeordnet werden soll.

Sie müssen eine Eingabegruppe „Antwort" und eine Eingabegruppe „Anfrage" sowie die untergeordneten Gruppen erstellen, die Daten empfangen.

Beim Erstellen der Eingabeportgruppen definieren Sie in jeder übergeordneten Gruppe einen Primärschlüssel. Sie definieren einen Fremdschlüssel in jeder untergeordneten Gruppe. Der Fremdschlüssel verbindet die Gruppe mit einer übergeordneten Gruppe.

Für die Antwortebene und die WSDL-Stammebene müssen Sie nur dann Schlüssel definieren, wenn Sie Daten auf der WSDL-Stammebene weitergeben. Die Stammebene könnte beispielsweise HTTP-Header enthalten.

Sie könnten etwa Gruppen von Ports erstellen, die den folgenden Gruppen für Kunden und Bestellungen ähneln:

```
Response
   Response_Key

   Customer_Details_Root
      Key_Cust_Det
      FK_Response_Key

      Customer
        Customer_ID
        FK_Cust_Det
        Name
        Address

          Orders
            Order_Num
            FK_Cust_ID

              Order_Items
                Order_Num
                Item
                Count
                Price
```

Operationsbereich

Der Bereich **Operation** zeigt die Elemente der SOAP-Meldungshierarchie so an, wie sie von der WSDL bzw. dem Schema definiert wurden. Die SOAP-Meldung muss nicht alle Elemente aus der WSDL oder dem Schema enthalten. Die Meldung enthält die Daten, die Sie aus den Eingabeports zuordnen.

Mehrfach vorkommende Knoten und komplexe Knoten definieren die Hierarchieebenen der SOAP-Meldungsstruktur. Das Developer-Tool fügt den Ebenen Schlüssel hinzu, um unter- bzw. übergeordnete Beziehungen zwischen den Ebenen definieren. Alle Hauptebenen der Hierarchie verfügen über einen Primärschlüssel. Untergeordnete Ebenen sind über einen Fremdschlüssel mit einer übergeordneten Ebene verbunden. Die Schlüssel, die in der SOAP-Meldungshierarchie erscheinen, werden in der Instanz einer SOAP-Meldung nicht angezeigt. Data Integration Service benötigt die Schlüsselwerte, um beim Generieren der SOAP-Meldung die Beziehungen zwischen den Datenebenen festzulegen.

Die Spalte **Speicherort** enthält den Gruppennamen und den Eingabeport, in denen die Daten eines Elements in der SOAP-Meldung enthalten sind. Die Spalte **Speicherort** bleibt leer, bis Sie dem Knoten einen Eingabeport zuordnen.

In der vorherigen Abbildung enthält die SOAP-Meldung eine einzige Instanz von Kundendaten und -bestellungen. Die Gruppe „Bestellungen“ (Orders) enthält ein mehrfach vorkommendes Element namens „Bestellung“ (Order). Die SOAP-Meldungshierarchie enthält die folgenden, mithilfe von Schlüsseln verbundenen Ebenen:

```
Response
   GetCustomerDetailsResponse
      GetCustomerDetailsResult
         Orders
            Order
```

Die Antwortebene (Response) ist die Stammebene der Antwortmeldung. Data Integration Service benötigt diese Ebene, um Header an die SOAP-Meldung anhängen zu können.

Die Ebene „GetCustomerDetailsResponse“ stellt die Stammebene der Meldung dar.

Beziehungen zwischen Port und Hierarchieebene

Beim Zuordnen von Eingabeports in der SOAP-Meldungshierarchie pflegen Sie eine Beziehung zwischen einer Eingabegruppe und einer Ebene in der SOAP-Meldungshierarchie. Angenommen, Sie haben zwei Eingabegruppen, Abteilung (Department) und Mitarbeiter (Employee).

Die Abteilungsgruppe erhält die folgenden Zeilen:

Dept_num	Name	Speicherort
101	HR	New York
102	Produkt	California

Die Mitarbeitergruppe erhält die folgenden Zeilen:

Dept_num	Mitarbeiter
101	Alice
101	Bob
102	Carol
102	Dave

Ordnen Sie die Abteilungsnummer in der Mitarbeitergruppe als Fremdschlüssel zu, der die Beziehung zwischen der Abteilungs- und der Mitarbeitergruppe festlegt. Die Abteilungsnummer tritt auf der Hierarchieebene der Abteilung auf, jedoch nicht auf jener der Mitarbeiter.

Die SOAP-Meldung weist folgende XML-Struktur auf:

```
<department>
   <dept_num>101</dept_num>
   <name>HR</name>
   <location>New York</location>

   <employee>
      <name>Alice</name>
   </employee>

   <employee>
      <name>Bob</name>
   </employee>
</department>

<department>
   <dept_num>102</dept_num>
   <name>Product</name>
   <location>California</location>

   <employee>
      <name>Carol</name>
   </employee>

   <employee>
      <name>Dave</name>
   </employee>

</department>
```

Schlüssel

SOAP-Meldungshierarchien enthalten Schlüssel. Data Integration Service benötigt diese Schlüsselwerte, um die XML-Hierarchie in der SOAP-Meldung erstellen zu können.

Sie müssen die Daten aus den Eingabeports den Schlüsseln in der SOAP-Meldungshierarchie zuordnen. Ordnen Sie die Daten den Schlüsseln auf jeder Ebene zu, der Sie Daten bereitstellen. Mehrfach vorkommende Knoten müssen mit einem übergeordneten Element verbunden werden.

Die Schlüssel erscheinen in der SOAP-Meldung ohne Angabe des Typs. Alle Ports, die einem Schlüssel zugeordnet werden, müssen den String-, Integer- oder Bigint-Datentyp aufweisen. Der Primärschlüssel in der übergeordneten Gruppe und der Fremdschlüssel in allen untergeordneten Gruppen müssen dieselben Werte für Datentyp, Präzision und Größenordnung aufweisen. Sie können den SOAP-Meldungsschlüsseln generierte Schlüssel zuordnen.

Ein Port kann einem Knoten und einem Schlüssel in derselben Hierarchieebene zugeordnet werden. Sie könnten etwa eine Mitarbeiter-ID einem Knoten in der SOAP-Meldung und einem Schlüssel auf der Mitarbeiterebene zuordnen.

Wenn ein Gruppenknoten in der Hierarchie einem anderen Gruppenknoten untergeordnet ist, gehen Sie folgendermaßen vor:

- Ordnen Sie einen Port dem Primärschlüssel in der übergeordneten Knotengruppe zu.
- Ordnen Sie einen Port einem Fremdschlüssel in der untergeordneten Knotengruppe zu.

Sie können Primärschlüssel auch Eingabeports zuordnen, um Zeilen mit nullwertigen oder duplizierten Primärschlüsseln zu entfernen.

Sie können in der SOAP-Meldung zusammengesetzte Schlüssel erstellen, indem Sie mehrere Ports demselben Schlüssel zuordnen. Zusammengesetzte Schlüssel dienen zur Denormalisierung von Daten und Pflege eindeutiger Schlüssel für mehrfach vorkommende Kombinationen von Werten. Zusammengesetzte Schlüssel können aus Strings, BigInt- oder ganzzahligen Werten bestehen.

Hinweis: Zum Generieren der Schlüsselwerte können Sie eine Ausdrucksumwandlung in das Operation-Mapping aufnehmen.

Beispiel für einen Zusammengesetzten Schlüssel

Konfigurieren Sie einen eindeutigen Schlüssel für Geschäftsbereich/Abteilung (Division/Department) aus den folgenden Gruppen von Ports:

```
Company
   Company_Num
   Company_Name

   Division
      Company_Num
      Divison_Num
      Division_Name

      Department
         Division_Num
         Dept_Num
         Dept_Name
         Location
```

Die Abteilungsnummer Dept_Num ist zwar innerhalb eines Geschäftsbereichs eindeutig, nicht jedoch für alle Geschäftsbereiche im Unternehmen.

Sie könnten nun eine Abteilungsgruppe konfigurieren, in der die Geschäftsbereich- und Abteilungsdaten enthalten sind. Konfigurieren Sie die Nummern für Geschäftsbereich und Abteilung als Teil der zusammengesetzten Schlüssels:

```
Department
   Division_Num + Dept_Num (key)
   Dept_Name
   Location
```

Die Reihenfolge der Zuordnung bestimmt den Schlüsselwert.

Zuordnen von Ports

Nachdem Sie die Eingabeports erstellt haben, ordnen Sie sie der SOAP-Meldungshierarchie zu. Der Speicherort des Ports wird neben dem Knoten im Bereich **Operation** angezeigt.

Sie können Ports den folgenden Typen von Knoten zuordnen:

Atomarer Knoten

Ein einfaches, unteilbares Element oder Attribut ohne untergeordnete Elemente

Mehrfach vorkommender atomarer Knoten

Ein einfaches Element oder Attribut, das am selben Ort in der Hierarchie mehrfach vorkommt

Komplexer Knoten

Ein Element, das andere Elemente enthält

Wenn der übergeordnete Knoten keinen Speicherort aufweist, übernimmt er als Speicherort den Namen der Eingabegruppe. Weist er hingegen einen auf, muss jeder Knoten in der Hierarchieebene einen Ausgabeort im selben Speicherort besitzen.

Sie können einen Eingabegruppennamen einem übergeordneten Knoten auf einer Hierarchieebene zuordnen. Das Developer-Tool aktualisiert das Speicherortfeld des übergeordneten Knotens in der Hierarchie. Das Developer-Tool aktualisiert keine untergeordneten Knoten, die zur Gruppe in der Hierarchie gehören. Beim Zuordnen von Eingabeports zu untergeordneten Knoten muss jeder Eingabeport denselben Speicherort wie der übergeordnete Knoten aufweisen.

Sie können das Mapping zwischen Eingabegruppe und Hierarchieebene später noch ändern. Sie können auf **Löschen** klicken oder die Linien zwischen den Ports- und Operation-Bereichen entfernen. Zum Entfernen der Linien ziehen Sie den Zeiger der Linien, um sie auszuwählen. Klicken Sie auf **Löschen**.

Zuordnen eines Ports

Beim Zuordnen eines Ports zu einem Knoten in der SOAP-Meldung liefert das Developer-Tool je nach Knotentyp, dem der Port zugeordnet wird, unterschiedliche Ergebnisse.

Die folgende Tabelle beschreibt die Ergebnisse der Zuordnung eines einzelnen Ports zu verschiedenen Zielknoten im Bereich **Operation**:

Zielknoten	Ergebnisse
Atomarer Knoten	Wenn Sie einen einzelnen Port einem Knoten zuordnen und der übergeordnete Knoten keinen Speicherort aufweist, übernimmt der Knoten den Speicherort des Ports. Der übergeordnete Knoten übernimmt den Speicherort der Eingabegruppe für den einzelnen Port. Wenn Sie einen einzelnen Port einem Knoten zuordnen und der übergeordnete Knoten bereits einen Speicherort besitzt, können Sie den Speicherort des übergeordneten Knoten ändern und den Speicherort der anderen untergeordneten Knoten auf derselben Ebene löschen. Der Speicherort der Hierarchieebene übernimmt den Gruppennamen des Ports.
Mehrfach vorkommende atomare Knoten bzw. der Primärschlüssel des mehrfach vorkommenden atomaren Knotens	Wenn Sie einen einzelnen Port einem mehrfach vorkommenden atomaren Knoten zuordnen, setzt das Developer-Tool den Speicherort des atomaren Knotens auf die Gruppe des ausgewählten Ports.
Komplexer Knoten	Wenn Sie einen einzelnen Port einem komplexen Knoten zuordnen, setzt das Developer-Tool den Speicherort des komplexen Knotens auf den Speicherort der Gruppe, die den Port enthält. Das Developer-Tool fordert Sie auf, den einzeln auftretenden atomaren Knoten anzugeben, dem der Port zugeordnet werden soll. Wenn alle einzeln auftretenden atomaren Knoten Speicherorte aufweisen, können Sie den komplexen Knoten nicht zuordnen.

Zuordnen einer Gruppe

Beim Zuordnen einer Eingabegruppe zu einem Knoten in der SOAP-Meldung liefert das Developer-Tool je nach Knotentyp, dem die Gruppe zugeordnet wird, unterschiedliche Ergebnisse.

Die folgende Tabelle beschreibt die Ergebnisse der Zuordnung einer Gruppe zu einem Knoten im Bereich **Operation**:

Zielknoten	Ergebnisse
Atomarer Knoten	Gruppen können keinem atomaren Knoten zugeordnet werden.
Mehrfach vorkommender atomarer Knoten	Sie werden aufgefordert, einen Port in der Eingabegruppe auszuwählen, um den Speicherort des Knotens und des Primärschlüssels zu aktualisieren.
Mehrfach vorkommender komplexer Knoten	Das Developer-Tool setzt den Speicherort des komplexen Knotens auf den Speicherort der Gruppe.

Zuordnen mehrerer Ports

Beim Zuordnen mehrerer Ports zu einem Knoten in der SOAP-Meldung liefert das Developer-Tool je nach Knotentyp, dem die Ports zugeordnet werden, unterschiedliche Ergebnisse. Sie können gleichzeitig mehrere Ports zuordnen, wenn sie aus derselben Gruppe stammen.

Die folgende Tabelle beschreibt die Ergebnisse für den Knoten bei der Zuordnung mehrerer Ports zu Knoten:

Zielknoten	Ergebnisse
Einzelner atomarer Knoten	Wenn Sie mehrere Ports einem einzigen Knoten zuordnen, aktualisieren Sie den Speicherort von mehr als einem atomaren Knoten im Bereich **Operation**. Wenn in der Hierarchieebene nicht genügend Knoten zum Aktualisieren vorhanden sind, ordnet das Developer-Tool nur den verfügbaren Knoten Ports zu.
Mehrfach vorkommender atomarer Knoten	Wenn Sie mehrere Ports einem mehrfach vorkommenden atomaren Knoten zuordnen, pivotieren Sie Ports zu Mehrfachvorkommen des Knotens. Das Developer-Tool erstellt die Instanzen des Knotens basierend darauf, wie viele Ports Sie zuordnen. Eine Meldung mit der Anzahl der projizierten Ports wird angezeigt.
Mehrfach vorkommender komplexer Knoten	Wenn Sie mehrere Ports einem komplexen Knoten zuordnen, müssen Sie bestimmen, welche einzeln auftretende atomare Knoten aktualisiert werden sollen. Sie pivotieren die Ports zu Mehrfachvorkommen des Knotens. Das Developer-Tool erstellt die Instanzen des Knotens basierend darauf, wie viele Ports Sie zuordnen.

Pivotieren mehrfach vorkommender Ports

Einem mehrfach vorkommenden Knoten in der SOAP-Meldung können mehrere Eingabeports zugeordnet werden. Das Developer-Tool pivotiert die Eingabedaten zu mehrfach vorkommenden Knoten in der SOAP-Meldung.

Um die Anzahl der zu pivotierenden Elemente zu ändern, wählen Sie im Dialogfeld **Mapping-Optionen** die Option **Bestehende Pivotierung überschreiben**.

Wenn Sie eine der pivotierten Port-Instanzen aus dem Bereich **Ports** entfernen, entfernt das Developer-Tool alle entsprechenden Instanzen aus dem Bereich **Operation**.

Beispiel für eine Pivotierung

Angenommen, eine Eingabegruppe weist folgende Zeilen auf:

Num	Name	Speicherort	emp_name1	emp_name2	emp_name3
101	HR	New York	Alice	Tom	Bob
102	Produkt	California	Carol	Tim	Dave

Jede Zeile enthält eine Abteilungsnummer und drei Mitarbeiternamen.

„Mitarbeiter“ (Employee) ist ein mehrfach vorkommender Knoten in der SOAP-Meldungshierarchie. Sie können alle Mitarbeiter-Instanzen aus der Eingabezeile der SOAP-Meldungshierarchie zuordnen. Wählen Sie alle Vorkommen von „Mitarbeiter“ aus. Klicken Sie auf **Map**. Im Dialogfeld **Mapping-Optionen** werden Sie aufgefordert, einen Knoten aus der Liste auszuwählen.

Das Developer-Tool ändert den Mitarbeiter-Knoten und nimmt die Knoten mit mehreren Namen in die SOAP-Meldungshierarchie auf:

```
Department
   num
```

```
    name
    location
    Employee     (unbounded)
        emp_name1
        emp_name2
        emp_name3
```

Die SOAP-Meldung gibt folgende Hierarchie zurück:

```
<department>
   <num>101</num>
   <name>HR</name>
   <location>New York</location>
   <employee>
      <emp_name>Alice</name>
   </employee>
   <employee>
      <emp_name>Tom</name>
   </employee>
   <employee>
      <emp_name>Bob</name>
   </employee>
</department>

<department>
   <num>102</num>
   <name>Product</name>
   <location>California</location>
   <employee>
      <emp_name>Carol</name>
   </employee>
   <employee>
      <emp_name>Tim</name>
   </employee>
   <employee>
      <emp_name>Dave</name>
   </employee>
  </department>
```

Zuordnen denormalisierter Daten

Denormalisierte Daten können zugeordnet und an normalisierte Knoten in einer SOAP-Meldung übergeben werden.

Beim Zuordnen denormalisierter Daten übergeben Sie Daten aus einer einzelnen Eingabegruppe an mehrere Knoten in der SOAP-Meldungshierarchie. Dabei können Sie in der SOAP-Meldung folgende und ähnliche Gruppenbeziehungen definieren:

Lineare Knotenbeziehung

Knoten A ist Knoten B übergeordnet, der wiederum ist Knoten C übergeordnet, welcher wiederum Knoten D übergeordnet ist.

Hierarchische Knotenbeziehung

Knoten A ist Knoten B und Knoten C übergeordnet; Knoten B und Knoten C stehen in keinerlei Beziehung zueinander.

Die folgende Tabelle zeigt die Eingabezeilen mit denormalisierten Daten für Geschäftsbereich und Abteilung:

Division	Dept_Num	Dept_Name	Telefon	Employee_Num	Employee_Name
01	100	Accounting	3580	2110	Amir

Division	Dept_Num	Dept_Name	Telefon	Employee_Num	Employee_Name
01	100	Accounting	3580	2113	Robert
01	101	Engineering	3582	2114	Stan
01	101	Engineering	3582	2115	Jim
02	102	Facilities	3583	2116	Jose

Die Eingabedaten liefern eindeutige Mitarbeiternummern und -namen. Die Geschäftsbereichs- und Abteilungsdaten werden für jeden Mitarbeiter in derselben Abteilung und dem derselben Geschäftsbereich wiederholt.

Lineare Gruppenbeziehung

Beim Konfigurieren von Ports können Sie eine separate Gruppe für Geschäftsbereich, Abteilung und Mitarbeiter festlegen. Der Geschäftsbereich ist der Abteilung übergeordnet, die Abteilung dem Mitarbeiter. Gruppen können in der folgenden linearen Struktur konfiguriert werden :

```
Division
   Division_Key
   Division_Num
   Division_Name

      Department
        Department_Key
        Division_FKey
        Dept_Num
        Dept_Name
        Phone

        Employee
           Department_Fkey
           Employee_Num
           Employee_Name
```

Die SOAP-Meldung enthält eindeutige Instanzen von Geschäftsbereich und Abteilung, wenn auch Division_Num und Dept_Num in den Eingabedaten wiederholt werden. Definieren Sie Division_Num als Primärschlüssel in der Geschäftsbereichsgruppe. Definieren Sie Dept_Num als Primärschlüssel in der Abteilungsgruppe.

Hierarchische Gruppenbeziehung

Sie können eine Gruppenhierarchie erstellen, bestehend aus einer übergeordneten Geschäftsbereichsgruppe und den untergeordneten Gruppen für Abteilung und Mitarbeiter. Abteilung und Mitarbeiter weisen keine Primärschlüssel-Fremdschlüssel-Beziehung auf. Abteilung und Mitarbeiter sind dem Geschäftsbereich untergeordnet. Die Gruppen können in folgender Struktur konfiguriert werden:

```
Division
   Division_Key
   Division_Num
   Division_Name

   Department
      Division_FKey
      Dept_Num
      Dept_Name

   Employee
      Division_FKey
      Employee_Num
      Employee_Name
```

Abgeleitete Typen und Elementsubstitution

Eingabeports können abgeleiteten komplexen Typen, anyType-Elementen und Substitutionsgruppen in einer SOAP-Meldung zugeordnet werden. Die SOAP-Meldung kann Elemente für den Basistyp und die abgeleiteten Typen enthalten.

In einer Typenbeziehung ist der Basistyp der Typ, aus dem die weiteren Typen abgeleitet werden. Ein abgeleiteter Typ erbt Elemente aus dem Basistyp. Bei einem erweiterten komplexen Typ handelt es sich um einen abgeleiteten Typ, der zwar Elemente von einem Basistyp erbt, aber auch zusätzliche Elemente enthält. Ein eingeschränkter komplexer Typ ist ein abgeleiteter Typ, der Elemente aus dem Basistyp beschränkt.

Generieren abgeleiteter Typen

Wenn eine WSDL oder ein Schema abgeleitete Typen enthält, müssen Sie die Typen bestimmen, die in die SOAP-Meldung aufgenommen werden sollen.

Die WSDL kann beispielsweise den Basistyp „AddressType“ enthalten. Als abgeleitete Adressentypen enthält sie darüber hinaus „USAddressType“ und „UKAddressType“.

Jeder Typ umfasst folgende Elemente:

- AddressType: Straße, Ort
- USAddressType (erweitert AddressType): Bundesstaat, Zip-Code
- UKAddressType (erweitert AddressType): Postcode, Land

Wenn Sie einen USAddressType im Operationsbereich auswählen, erstellt das Developer-Tool in der SOAP-Meldung eine Gruppe für das Element „USAddressType“. Die Gruppe umfasst die Straße (Street) und den Ort (City) aus dem Basistyp sowie den Bundesstaat (State) und den Zip-Code der US-Adresse. Abgeleitete Typen zur Erweiterung des Basistyps enthalten stets Elemente aus dem Basistyp.

Wenn Sie alle verfügbaren abgeleiteten Typen für die SOAP-Meldung auswählen, erstellt das Developer-Tool Gruppen ähnlich der folgenden SOAP-Hierarchie:

```
Address
  Address: Address
    Street
    City

  Address:USAddressType
     Street
     City
     State
     ZipCode

  Address: UKAddressType
     Street
     City
     PostalCode
     Country
```

Sie müssen die Eingabeportgruppen für Address, USAddress und UKAddress definieren .

Generieren von anyType-Elementen und -Attributen

Für einige Schemaelemente und Attribute sind beliebige Datentypen in der SOAP-Meldung zulässig.

Das anyType-Element stellt eine Auswahl aller global bekannten Typen dar. Bevor Sie einen Port zu einem anyType-Element in einer SOAP-Meldung zuordnen, wählen Sie einen verfügbaren komplexen Typ oder xs:string aus. Wenn weder die WSDL noch das Schema einen komplexen Typ enthält, ersetzt das Developer-Tool den anyType-Elementtyp durch xs:string.

Um einen komplexen Elementtyp im Operationsbereich auszuwählen, klicken Sie in der Spalte **Typ** des anyType-Elements auf **Auswählen**. Eine Liste der verfügbaren komplexen Typen und xs:string wird eingeblendet.

Die folgenden Elemente und Attribute lassen beliebige Datentypen zu:

anyType-Element

Lässt Elemente jedes Datentyps in der zugeordneten XML-Datei zu.

anySimpleType-Element

Lässt Elemente jedes einfachen Typs (simpleType) in der zugeordneten XML-Datei zu.

BELIEBIGES Inhaltselement

Lässt beliebige, im Schema definierte globale Elemente zu.

anyAttribute-Attribute

Lässt beliebige, im Schema definierte globale Attribute zu.

Generieren von Substitutionsgruppen

Mit Substitutionsgruppen können Sie in der SOAP-Meldung ein Element durch ein anderes ersetzen. Substitutionsgruppen funktionieren ähnlich wie abgeleitete Typen, außer dass die Elementdefinitionen einen Substitutionsgruppennamen enthalten.

Beispiel: Sie haben den Adresstyp „AddressType“ und die abgeleiteten Typen „USAddressType“ und „UKAddressType“.

```
xs:element name="Address" type="xs:string"/>
<xs:element name="USAddress" substitutionGroup="Address"/>
<xs:element name="UKAddress" substitutionGroup="Address"/>
```

Beim Konfigurieren der SOAP-Meldungshierarchie können Sie bestimmen, welches Element das Element „AdressType“ in der SOAP-Meldung ersetzen soll.

Generieren von XML-Konstrukten in SOAP-Meldungen

Eine WSDL bzw. ein Schema kann Auswahl-, Listen- und Union-Elemente enthalten. Web-Dienst-Umwandlungen können SOAP-Meldungen generieren, die diese Elemente enthalten.

Auswahlelement

Auswahlelemente beschränken untergeordnete Elemente auf ein Element in der Deklaration „<choice>“.

Um Ports einer SOAP-Meldung zuzuordnen, die Auswahlelemente enthält, erstellen Sie eine Eingabegruppe mit allen Elementen des Auswahlkonstrukts. Angenommen, ein Elementbeschreibung (Description) stellt eine Größen- (Dimension) oder Gewichtsordnung (Weight) dar:

```
item: description, choice {dimension, weight}
```

Wenn die Beschreibung eine Größenordnung ist, ist die Beschreibung ein komplexer Typ aus Länge (Length), Breite (Width) und Höhe (Height).

Wenn die Beschreibung eine Gewichtsordnung ist, ist die Beschreibung ist eine einfacher Stringtyp.

Die Eingabedaten weisen folgende Spalten und Zeilen auf:

Beschreibung	länge	width	height	weight
box	20cm	18cm	15cm	NULL
coffee	NULL	NULL	NULL	500g

Die SOAP-Meldung enthält eine Elementgruppe mit verschiedenen Beschreibungen von Größen- und Gewichtsordnungen:

```
Item
   Description
       Dimension
          Length
          Width
          Height
       Weight
```

Die NULL-Werte in den Eingabedaten werden zu fehlenden Elementen in der XML-Ausgabe.

Die SOAP-Meldung enthält folgende Daten:

```
<item>
   <desc>box</desc>
      <dimension>
        <length>20cm</length>
        <width>18cm</width>
        <height>15cm</height>
      </dimension>
</item>

<item>
   <desc>coffee</desc>
      <weight>500g</weight>
</item>
```

Listenelement

Eine Listenelement ist ein XML-Element, das mehrere Werte einfachen Typs im selben Element oder Attribut enthalten kann. Data Integration Service kann Listen in den Eingabedaten verarbeiten, wenn sie als konsolidierte Datenstrings dargestellt werden.

Wenn jedes Element in der Liste eine separates Element ist, etwa ClassDates1, ClassDates2 und ClassDates3, kann Data Integration Service die Einträge nicht als Liste verarbeiten. Wenn die SOAP-Meldung eine Liste zurückgeben soll, können Sie die Elemente mithilfe einer Ausdrucksumwandlung zu einem String zusammenfassen.

Die folgenden Eingabezeilen enthalten ein Listenelement namens „ClassDates“, das Wochentage enthält:

CourseID	Name	ClassDates
Math 1	Beginning Algebra	Mon Wed Fri
History 1	World History	Tue Thu

Data Integration Service kann eine SOAP-Meldung mit der folgenden XML-Struktur zurückgeben:

```
<class>
   <courseId>Math 1</courseId>
   <name>Beginning Algebra</name>
   <classDates>Mon Wed Fri</classDates>
</class>
<class>
   <courseId>History 1</courseId>
   <name>World History</name>
   <classDates>Tue Thu</classDates>
</class>
```

Union-Element

Union-Elemente sind Einfachtypen, bestehend aus einer Vereinigung mehrerer Typen. Bei SOAP-Meldungen mit Union-Elementen müssen Sie einen einzelnen Eingabeport zuordnen, der die Daten in einem String enthält.

Angenommen, die SOAP-Meldung enthält ein Element namens „Größe“ (Size). Größe ist eine Union aus Ganzzahl und String:

```
<xs:element name="size">
  <xs:simpleType>
    <xs:union memberTypes="size_no size_string" />
  </xs:simpleType>
</xs:element>
```

Die Eingabezeilen enthalten Elemente mit einer Beschreibung und einer Größe. Ein Element kann eine numerische Größe aufweisen, z. B. 42, oder eine Größe mit einem Stringwert angeben, etwa „Large“, „Medium“ oder „Small“.

Die folgende Tabelle zeigt Eingabezeilen mit einer numerischen Größe und einer String-Größe:

Desc	Size
shoes	42
shirt	large

Erstellen Sie einen Port für das Element „Größe“. Ordnen Sie den Port als String zu. Die SOAP-Meldung enthält die folgenden Elemente:

```
<item>
   <desc>shoes</desc>
   <size>42</size>
</item>

<item>
   <desc>shirt</desc>
   <size>large</size>
</item>
```

KAPITEL 46

Umwandlung für gewichteten Durchschnitt

Dieses Kapitel umfasst die folgenden Themen:

- Umwandlung für gewichteten Durchschnitt - Übersicht, 646
- Konfigurieren einer gewichteten Durchschnittsumwandlung, 647
- Beispiel für gewichtete Übereinstimmungswerte, 647
- Erweiterte Eigenschaften für gewichteten Durchschnitt, 648

Umwandlung für gewichteten Durchschnitt - Übersicht

Die Umwandlung für den gewichteten Durchschnitt ist eine passive Umwandlung, die Übereinstimmungswerte aus mehreren Matching-Operationen ausliest und einen einzigen Übereinstimmungswert ausgibt.

Sie können eine numerische Gewichtung für jeden Score anwenden, der in die Umwandlung für den gewichteten Durchschnitt eintritt. Eine Gewichtung ist ein Wert zwischen 0 und 1. Sie können die Gewichtung für jeden Eingabe-Score bearbeiten und diesen erhöhen oder dessen Gewichtung für den Ausgabe-Score verringern. Verwenden Sie Gewichtungen, die die relative Bedeutung jeder Datenspalte in der Duplikatsanalyse widerspiegeln.

Verwenden Sie die Umwandlung für den gewichteten Durchschnitt beim Hinzufügen von Vergleichsumwandlungen zu einem Mapping oder Mapplet.

Hinweis: Sie können auch Gewichtungen in einer Match-Umwandlung zuweisen. Verwenden Sie die Match-Umwandlung zum Konfigurieren von Matching-Strategien und weisen Sie Gewichtungen in einer einzelnen Umwandlung zu. Sie können ein passendes Mapplet in eine Match-Umwandlung einbetten.

Konfigurieren einer gewichteten Durchschnittsumwandlung

Mithilfe der gewichteten Durchschnittsumwandlung passen Sie den allgemeinen Übereinstimmungswert an, den ein Mapping für eine Reihe von Übereinstimmungsanalysen erzeugt. Bearbeiten Sie die relativen Gewichtungen jedes Eingabeports, um so die Prioritäten der von Ihnen für die Quelldaten definierten Datenvergleiche hervorzuheben. Jeder Eingabeport bei einer gewichteten Durchschnittsumwandlung repräsentiert einen ausgegebenen Übereinstimmungswert einer Vergleichsumwandlungsstrategie.

Die folgenden Schritte beschreiben das Verfahren für die Konfiguration einer nicht wiederverwendbaren gewichteten Durchschnittsumwandlung in einem Mapplet oder Mapping, das Vergleichsumwandlungen verwendet.

1. Öffnen Sie ein Mapplet oder Mapping zur Übereinstimmungsanalyse und ergänzen Sie nach den Vergleichsumwandlungen eine gewichtete Durchschnittsumwandlung.
2. Verbinden Sie den ausgegebenen Wert einer Vergleichsumwandlung mit einem Eingabeport der gewichteten Durchschnittsumwandlung.

   Wiederholen Sie diesen Schritt für weitere Vergleichsumwandlungen im Mapplet oder Mapping.
3. Wählen Sie die Registerkarte **Ports** der gewichteten Durchschnittsumwandlung.
4. Doppelklicken Sie für jede Eingabe auf das Feld **Gewichtung** und geben Sie einen Gewichtungswert zwischen `0,001` und `1` ein. Der Gewichtungswert sollte die relative Wichtigkeit des Eingabewerts im Vergleich zu anderen Eingaben der Umwandlung widerspiegeln.
5. Speichern Sie das Mapplet oder Mapping.

Beispiel für gewichtete Übereinstimmungswerte

Erstellen Sie ein Übereinstimmungsanalyse-Mapping, um die Anzahl doppelter Kundennamen in einer Kundendatenbank zu bestimmen. Fügen Sie zwei Vergleichsumwandlungen hinzu, um Übereinstimmungswerte für die Spalten `Postleitzahl` und `Nachname` in den Daten zu erzeugen.

Viele Datensätze haben übereinstimmende Postleitzahlen, aber deutlich weniger Datensätze haben übereinstimmende Nachnamen. Beim Ermitteln des Durchschnitts dieser Übereinstimmungswerte müssen Sie die Wichtigkeit der eindeutigen Übereinstimmungen hervorheben.

Um die Wichtigkeit der Übereinstimmungswerte bei Nachnamen hervorzuheben, wenden Sie eine höhere Gewichtung auf den Übereinstimmungswert bei `Nachnamen` an.

Beispielsweise setzen Sie den **Gewichtungswert** für die eingehenden Nachnamenswerte auf `0,8` und den **Gewichtungswert** für die eingehenden Postleitzahlenwerte auf `0,4`.

Erweiterte Eigenschaften für gewichteten Durchschnitt

Konfigurieren Sie Eigenschaften, um zu bestimmen, wie der Data Integration Service Daten für die Umwandlung für gewichteten Durchschnitt verarbeitet.

Sie können Tracingebenen für Protokolle konfigurieren.

Konfigurieren Sie die folgende Eigenschaft auf der Registerkarte **Erweitert**:

Tracingebene

Die Menge an Detail, die im Protokoll für diese Umwandlung erscheint. Sie können „Kurz“, „Normal“, „Ausführliche Intialisierung“ oder „Ausführliche Daten“ wählen. Standardwert ist „Normal“.

KAPITEL 47

Schreibumwandlung

Dieses Kapitel umfasst die folgenden Themen:

- Schreibumwandlung – Übersicht, 649
- Eigenschaften von Schreibumwandlungen , 649
- Eine Schreibumwandlung erstellen, 655

Schreibumwandlung – Übersicht

Bei einer Schreibumwandlung handelt es sich um eine passive Umwandlung. Sie stellt ein Ziel dar, in welches das Mapping Daten schreibt. Sie können eine Schreibumwandlung aus einem physischen oder logischen Datenobjekt oder aus einem Parameter erstellen. Die Schreibumwandlung ist nicht wiederverwendbar.

Die Eigenschaften, die Sie für eine Schreibumwandlung konfigurieren können, hängen vom Typ des Datenobjekts ab, das Sie zum Erstellen der Umwandlung verwendet haben.

Schreibumwandlungen können dynamische Ziele darstellen. Sie können eine Schreibumwandlung so konfigurieren, dass deren Ports, Metadaten und andere Eigenschaften dynamisch aktualisiert werden. Weitere Informationen zum Konfigurieren von dynamischen Zielen finden Sie im *Informatica Developer-Handbuch für Mappings*.

Eigenschaften von Schreibumwandlungen

Nach dem Erstellen einer Schreibumwandlung können Sie Eigenschaften für die Umwandlung konfigurieren.

Konfigurieren Sie Eigenschaften für Schreibumwandlungen auf Registerkarten in der Ansicht **Eigenschaften** der jeweiligen Umwandlung. Die konfigurierbaren Registerkarten richten sich nach dem Typ des Ziels, das die Schreibumwandlung darstellt.

In der folgenden Tabelle werden die Eigenschaftenregisterkarten beschrieben und der Zieltyp angegeben, für den die Registerkarte verwendet wird:

Eigenschaften-Registerkarte	Beschreibung	Zieltyp
Allgemein	Geben Sie Umwandlungseigenschaften und -verhalten an. Synchronisieren Sie für relationale und benutzerdefinierte Datenobjektquellen die Eingabeports der Umwandlung mit der Quelle.	Alle
Datenobjekt	Geben Sie die Datenquelle der Umwandlung an. Rufen Sie für relationale und benutzerdefinierte Datenobjektquellen Datenobjektspalten zur Laufzeit aus Datenquellen ab.	Einfachdatei Relational Benutzerdefiniertes Datenobjekt
Format	Eingabeeinstellungen für eine Einfachdatei-Datenquelle	Einfachdatei
Ports	Richten Sie die Portdefinition gemäß dem zugeordneten Datenobjekt oder Zuordnungsfluss ein.	Alle
Laufzeit	Eigenschaften, die vom Datenintegrationsdienst beim Schreiben von Daten in das Ziel zur Laufzeit verwendet werden, z. B. Empfänger von Ablehnungsdateien. Lehnen Sie für ein Einfachdateiziel Dateinamen und -verzeichnisse ab.	Einfachdatei Relational
Datenobjektparameter	Zeigen Sie die Parameter des Datenobjekts an. Konfigurieren Sie Parameterwerte für die Zuordnung oder binden Sie die Parameter an Zuordnungsparameter.	Einfachdatei Benutzerdefiniertes Datenobjekt Logische Datenobjekte
Verknüpfung zur Laufzeit	Erstellen Sie neue Laufzeitverknüpfungen und zeigen Sie Verknüpfungseigenschaften an.	Alle
Erweitert	Legen Sie die Tracingebene und Zeilenreihenfolge fest. Erstellen oder ersetzen Sie die Zieltabelle für ein relationales Ziel zur Laufzeit.	Einfachdatei Relational Benutzerdefinierte Datenobjekte Logische Datenobjekte

Allgemeine Eigenschaften

Sie können den Namen und die Beschreibung der Schreibumwandlung konfigurieren. Sie können zudem die folgenden Eigenschaften konfigurieren:

Bei Änderung der Spaltenmetadaten

Verfügbar für relationale und benutzerdefinierte Ziele. Wählen Sie eine der folgenden Optionen aus:

- Eingabeports synchronisieren. Das Developer Tool aktualisiert die Eingabeports der Schreibumwandlung mit den Änderungen der Metadaten, die das Modellrepository für das Datenobjekt speichert.
- Nicht synchronisieren. Das Developer Tool zeigt keine Änderungen der Metadaten im Datenobjekt an.

Standardwert ist die Option „Eingabeports synchronisieren“.

Physisches Datenobjekt

Verfügbar für Einfachdateiziele und benutzerdefinierte Ziele. Das zum Erstellen der Umwandlung verwendete Objekt.

Sie können den Dateiobjektnamen auswählen und dessen Eigenschaften konfigurieren.

Datenobjekteigenschaften

Auf der Registerkarte „Datenobjekt“ können Sie das Ziel der Schreibumwandlung angeben bzw. ändern sowie relationale und benutzerdefinierte Datenobjektziele in dynamische Datenobjektziele ändern.

Sie können die folgenden Eigenschaften konfigurieren:

Angeben nach

Wählen Sie für die Angabe von Zielspalten und Metadaten für die Schreibumwandlung eine der folgenden Optionen aus:

- Wert. Die Schreibumwandlung verwendet das zugehörige Datenobjekt, um Zielspalten und Metadaten anzugeben.
- Parameter. Die Schreibumwandlung verwendet einen Parameter, um Zielspalten und Metadaten anzugeben.

Standardwert ist die Option „Wert“.

Datenobjekt

Wenn Sie die Schreibumwandlung aus einem vorhandenen Datenobjekt erstellt haben, wird im Feld der Name des Objekts angezeigt. Klicken Sie auf **Durchsuchen**, um das Datenobjekt zu ändern, das der Schreibumwandlung zugeordnet werden soll.

Parameter

Wählen Sie einen Parameter aus oder erstellen Sie einen Parameter, der der Schreibumwandlung zugeordnet werden soll.

Datenobjektspalten zur Laufzeit aus Datenquelle abrufen

Wenn Sie diese Option aktivieren, ruft der Datenintegrationsdienst Metadaten und Änderungen bei der Datendefinition von den Zieltabellen zu der Schreibumwandlung ab.

Porteigenschaften

Auf der Registerkarte „Ports“ können Sie die folgenden Eigenschaften konfigurieren:

Spalten definiert nach

Wählen Sie eine der folgenden Optionen zum Definieren der Spalten von Schreibumwandlungen aus:

- Zugeordnetes Datenobjekt. Verwenden Sie Spaltennamen, Metadaten und andere Eigenschaften aus dem Datenobjekt auf der Registerkarte „Datenobjekt“.
- Zuordnungsfluss. Das Mapping ruft Spaltennamen, Metadaten und andere Eigenschaften aus den vorgelagerten Objekten im Mapping ab.

Standardwert ist die Option „Zugeordnetes Datenobjekt“.

Eigenschaften der Spaltenressourcen

Verfügbar für Einfachdateiziele und benutzerdefinierte Ziele. Die Ressource für jede Spalte ist das Datenobjekt, aus dem die Spalte deren Namen, Metadaten und andere Eigenschaften abruft. Wählen Sie den Ressourcennamen zur Änderung der Ressourceneigenschaften aus.

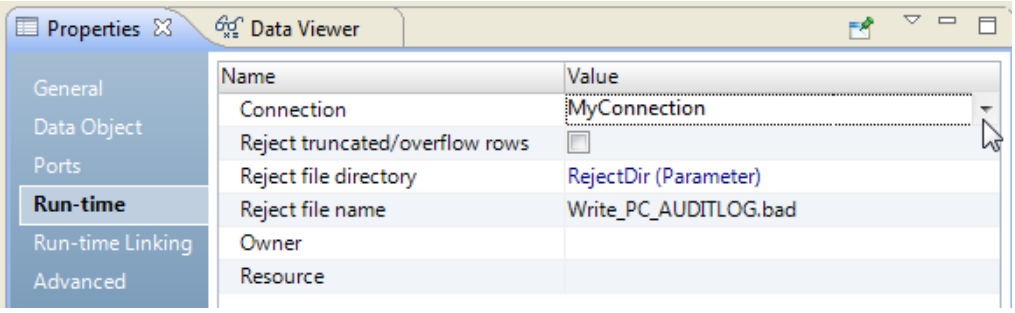
Laufzeiteigenschaften

Sie können die folgenden Eigenschaften für Schreibtransaktionen auf der Registerkarte „Laufzeit“ konfigurieren:

Verbindung

Verfügbar für relationale Ziele. Die von der Umwandlung verwendete Verbindung. Klicken Sie in die rechte Seite des Felds, um die Verbindung zu ändern.

Die folgende Abbildung zeigt die Position der Dropdown-Schaltfläche, auf die Sie klicken können:



Abgeschnittene/überlaufende Zeilen ablehnen

Verfügbar für relationale und benutzerdefinierte Ziele.

Mit dem Developer Tool können Sie Daten konvertieren, indem Sie diese von Port zu Port übergeben. Gelegentlich verursacht eine Umwandlung einen Überlauf numerischer Daten oder das Abschneiden von Zeichenfolgen in Spalten, die Zeichen enthalten. So verursacht z. B. das Übergeben von Daten aus einem Dezimal-Port (28, 2) in einen Dezimal-Port (19, 2) einen numerischen Überlauf. Ebenso schneidet der Datenintegrationsdienst die Zeichenfolge auf 10 Zeichen ab, wenn Sie Daten aus einem Zeichenfolgenport (28) in einen Zeichenfolgenport (10) übergeben.

Wenn ein Konvertierungsvorgang zu einem Überlauf führt, überspringt der Datenintegrationsdienst standardmäßig die Zeile. Der Datenintegrationsdienst schreibt keine Daten in die Ablehnungsdatei. Bei Zeichenfolgen schneidet der Datenintegrationsdienst die entsprechende Zeichenfolge ab und übergibt sie an die nächste Umwandlung.

Wählen Sie diese Option aus, um alle abgeschnittenen und überlaufenden Daten zwischen der letzten Umwandlung und dem Ziel in die Ablehnungsdatei der Sitzung einzuschließen. Der Datenintegrationsdienst sendet alle abgeschnittenen Zeilen und überlaufenden Zeilen je nach Konfiguration der Sitzung an die Ablehnungsdatei der Sitzung oder an die Zeilenfehlerprotokolle.

Verzeichnis für abgelehnte Dateien

Verzeichnis, in dem die Ablehnungsdatei gespeichert ist. Als Standardwert wird der Systemparameter RejectDir verwendet.

Name der Ablehnungsdatei

Dateiname der Ablehnungsdatei. Standardwert ist <output_file_name>.bad.

Wenn mehrere Partitionen in das Einfachdateiziel schreiben, schreibt jede Partition in eine separate Ablehnungsdatei namens <output_file_name><partition_number>.bad.

Eigenschaften von Laufzeitverknüpfungen

Erstellen und konfigurieren Sie eine Laufzeitverknüpfung auf der Registerkarte **Verknüpfung zur Laufzeit**. Eine Laufzeitverknüpfung ist eine Verknüpfung von Gruppen zwischen Umwandlungen, bei der ein Parameter,

eine Verknüpfungsrichtlinie oder beides verwendet wird, um zu bestimmen, welche Ports zur Laufzeit verknüpft werden sollen. Laufzeitverknüpfungen werden im Zuordnungseditor als starke Linien angezeigt.

Erstellen und konfigurieren Sie Laufzeitverknüpfungen mit einer Schreibumwandlung in den folgenden Fällen:

- Das Zieldatenobjekt in der Schreibumwandlung verwendet einen Parameter.
- Ports aus der vorgelagerten Umwandlung können sich zur Laufzeit ändern.

Hinweis: Erstellen Sie keine Laufzeitverknüpfungen mit einer Schreibumwandlung, wenn Sie die Zielspalten basierend auf dem Zuordnungsfluss definieren.

Sie können die folgenden Aufgaben auf der Registerkarte **Verknüpfung zur Laufzeit** durchführen:

Eine Laufzeitverknüpfung erstellen.

Klicken Sie im Bereich **Verknüpfungen** auf die Schaltfläche **Neu** und wählen Sie im Dialogfeld „Neue Verknüpfung“ die Umwandlung aus, von der Sie die Ports mit der Schreibumwandlung zur Laufzeit verknüpfen möchten.

Die Eigenschaften von Laufzeitverknüpfung konfigurieren.

Konfigurieren Sie im Bereich **Verknüpfungseigenschaften** die folgenden Eigenschaften von Laufzeitverknüpfungen:

Parameter

Wählen Sie diese Option aus, wenn sich die Portnamen zwischen den Ausführungen der Mappings ändern können und Sie die Portnamenwerte kennen. Verwenden Sie einen Parameter des Typs „Eingabeverknüpfungssatz“, um Ports zwischen den Ausführungen von Mappings nach Namenwerten zu verknüpfen. Die Syntax für den Mapping-Parameter „Eingabeverknüpfungssatz“ besteht aus kommagetrennten Portpaaren. `Afield1->Bfield2, Afield3->Bfield4 .`

Verknüpfungsrichtlinie

Wählen Sie diese Option aus, wenn Sie Ports mit übereinstimmenden Namen automatisch verknüpfen möchten. Wenn beispielsweise beide Mapping-Objekte einen Port namens SALARY enthalten, werden Sie vom Datenintegrationsdienst verknüpft. Sie können Präfixe und Suffixe in den Portnamen ignorieren.

Zur Laufzeit führt der Datenintegrationsdienst das Herstellen und Aufheben von Verknüpfungen zwischen den Ports in der folgenden Reihenfolge durch:

- Verknüpfungen, die Sie manuell im Zuordnungseditor erstellen.
- Verknüpfungen, die auf einem für eine Laufzeitverknüpfung konfigurierten Parameter basieren.
- Verknüpfungen, die auf einer für eine Laufzeitverknüpfung konfigurierten Verknüpfungsrichtlinie basieren.

Weitere Informationen zu Laufzeitverknüpfungen finden Sie im *Informatica Developer-Handbuch für Mappings*.

Erweiterte Eigenschaften

Der Datenintegrationsdienst kann SQL-Befehle ausführen.

Der Datenintegrationsdienst führt Pre-Mapping-SQL-Befehle für die Quelldatenbank aus, bevor er die Quelle liest. Er führt Post-Mapping-SQL-Befehle für die Quelldatenbank aus, nachdem er ins Ziel geschrieben hat. Sie können die folgenden Eigenschaften für Schreibtransaktionen auf der Registerkarte „Erweitert“ konfigurieren:

Tracingebene

Steuert die Detailmenge in der Mapping-Protokolldatei.

Zielladetyp

Typ des Zielladens. Wählen Sie „Normal“ oder „Masse“ aus. Sie können den Zielladetyp für relationale Ressourcen oder benutzerdefinierte Datenobjekte festlegen.

Wenn Sie „Normal“ auswählen, lädt der Datenintegrationsdienst die Ziele normal. Sie können „Masse“ auswählen, wenn Sie in DB2, Sybase, Oracle oder Microsoft SQL Server laden. Wenn Sie „Masse“ für andere Datenbanktypen angeben, wird der Datenintegrationsdienst auf einen normalen Ladevorgang zurückgesetzt. Das Massenladen kann die Zuordnungsleistung erhöhen, es beschränkt jedoch die Fähigkeit zur Wiederherstellung, da keine Datenbankprotokollierung stattfindet.

Wählen Sie den normalen Modus aus, wenn das Mapping eine Aktualisierungsstrategieumwandlung enthält. Wenn Sie „Normal“ auswählen und der Microsoft SQL Server-Zielname Leerzeichen enthält, konfigurieren Sie die folgenden Umgebungs-SQL-Befehle im Verbindungsobjekt:

```
SET QUOTED_IDENTIFIER ON
```

Überschreiben aktualisieren

Überschreibt die UPDATE-Standardanweisung für das Ziel.

Löschen

Löscht alle Zeilen, die zum Löschen gekennzeichnet sind.

Standardwert ist „Aktiviert“.

Einfügen

Fügt alle Zeilen ein, die zum Einfügen gekennzeichnet sind.

Standardwert ist „Aktiviert“.

Tabelle zur Laufzeit erstellen oder ersetzen

Der Datenintegrationsdienst entfernt die Zieltabelle zur Laufzeit und ersetzt sie durch eine Tabelle basierend auf einer von Ihnen angegebenen Zieltabelle.

Zieltabellen kürzen

Kürzt das Ziel, bevor es Daten lädt.

Standardwert ist „Aktiviert“.

Aktualisierungsstrategie

Aktualisierungsstrategie für vorhandene Zeilen. Sie können eine der folgenden Strategien auswählen:

- Update als Update. Der Datenintegrationsdienst aktualisiert alle Zeilen, die zur Aktualisierung gekennzeichnet sind.
- Update als Insert. Der Datenintegrationsdienst fügt alle Zeilen ein, die zur Aktualisierung gekennzeichnet sind. Sie müssen außerdem die Zieloption **Einfügen** auswählen.
- Update sonst Insert. Der Datenintegrationsdienst aktualisiert alle zum Aktualisieren gekennzeichneten Zeilen, wenn sie im Ziel vorhanden sind, und fügt dann alle verbleibenden zum Einfügen gekennzeichneten Zeilen ein. Sie müssen außerdem die Zieloption **Einfügen** auswählen.

PreSQL

SQL-Befehl, den der Datenintegrationsdienst für die Quelldatenbank ausführt, bevor er die Quelle liest.

Das Developer Tool validiert die SQL nicht.

PostSQL

SQL-Befehl, den der Datenintegrationsdienst für die Quelldatenbank ausführt, nachdem er in das Ziel geschrieben hat.

Das Developer Tool validiert die SQL nicht.

Zeilenreihenfolge beibehalten

Die Zeilenreihenfolge der Eingabedaten wird für das Ziel beibehalten. Wählen Sie diese Option aus, wenn der Datenintegrationsdienst keine Optimierung durchführen soll, die die Zeilenreihenfolge verändern könnte.

Wenn der Datenintegrationsdienst Optimierungen durchführt, geht möglicherweise die zu einem vorherigen Zeitpunkt im Mapping erstellte Zeilenreihenfolge verloren. Die Zeilenreihenfolge in einem Mapping können Sie mit einer sortierten Einfachdateiquelle, einer sortierten relationalen Quelle oder einer Sortierer-Umwandlung erstellen. Beim Konfigurieren eines Ziels zum Beibehalten der Zeilenreihenfolge führt der Datenintegrationsdienst keine Optimierungen für das Ziel durch.

Einschränkungen

SQL-Anweisungen für Einschränkungen der referenziellen Integrität auf Tabellenebene. Gilt nur für relationale Ziele.

Eine Schreibumwandlung erstellen

Wenn Sie eine Schreibumwandlung erstellen, wählen Sie eine der folgenden Methoden basierend auf der Ressource aus, aus der die Umwandlung erstellt wird:

Die Umwandlung aus einem Datenobjekt im Modellrepository erstellen.

Verwenden Sie diese Methode, wenn die Metadaten der Schreibumwandlung auf einem Datenobjekt basieren sollen.

Verwenden Sie zum Erstellen einer Schreibumwandlung im Zuordnungseditor den Assistenten **Neue Schreibumwandlung** oder ziehen Sie ein Datenobjekt aus der Ansicht **Objekt-Explorer** und wählen Sie „Schreiben“ als Transformationstyp aus.

Die Umwandlung aus dem Fluss der Mapping-Objekte erstellen.

Verwenden Sie diese Methode, um den Ports in der Schreibumwandlung den Metadatenfluss aus vorgelagerten Mapping-Umwandlungen zugrunde zu legen.

Die Umwandlung mithilfe eines Parameters erstellen.

Verwenden Sie diese Methode, wenn die Schreibumwandlung Ports aus dem Objekt übernehmen soll, das der Parameter darstellt.

Die Umwandlung mithilfe einer anderen Umwandlung im Mapping erstellen.

Verwenden Sie diese Methode, wenn die Schreibumwandlung über dieselben Ports wie die Quellumwandlung verfügen soll.

Erstellen einer Schreibumwandlung aus einem Datenobjekt

Sie können eine Schreibumwandlung im Zuordnungseditor erstellen und konfigurieren.

Sie sollten diese Methode verwenden, wenn Sie bestimmte Eigenschaften für die Umwandlung konfigurieren möchten. Beispielsweise können Sie die Umwandlung erstellen und anschließend so konfigurieren, dass Spaltenmetadaten aus der Datenquelle zur Laufzeit abgerufen werden.

1. Klicken Sie mit der rechten Maustaste in den Zuordnungseditor und wählen Sie **Umwandlung hinzufügen** aus.

   Das Dialogfeld **Umwandlung hinzufügen** wird geöffnet.

2. Wählen Sie **Schreiben** aus und klicken Sie auf **OK**.

   Der Assistent **Neue Schreibumwandlung** wird geöffnet.

3. Wählen Sie **Physisches Datenobjekt** oder **Logisches Datenobjekt** aus und klicken Sie anschließend auf **Durchsuchen**.

   Das Dialogfeld **Datenobjekt auswählen** wird geöffnet.

4. Wählen Sie eine Datenquelle aus und klicken Sie anschließend auf **OK**.
5. Um Spalten von Schreibumwandlungen durch das zugeordnete Datenobjekt zu definieren, wählen Sie **Zugeordnetes Datenobjekt** aus.

   Die Schreibumwandlungsports entsprechen den Spalten des zugeordneten Datenobjekts.

6. Um Zielobjektspalten mit Änderungen aus der Zieldatei zur Laufzeit dynamisch zu aktualisieren, wählen Sie **Datenobjektspalten zur Laufzeit aus Datenquelle abrufen** aus.

   Der Datenintegrationsdienst aktualisiert die Spaltenmetadaten für die Schreibumwandlung, wenn das Mapping ausgeführt wird.

7. Klicken Sie auf **Fertig stellen**.

Erstellen einer Schreibumwandlung aus einem Zuordnungsfluss

Sie können eine Schreibumwandlung im Zuordnungseditor erstellen und konfigurieren.

Sie sollten diese Methode verwenden, wenn Sie bestimmte Eigenschaften für die Umwandlung konfigurieren möchten. Beispielsweise können Sie die Umwandlung erstellen und anschließend so konfigurieren, dass Spalten basierend auf dem Zuordnungsfluss definiert und Spaltenmetadaten aus der Datenquelle zur Laufzeit abgerufen werden.

1. Klicken Sie mit der rechten Maustaste in den Zuordnungseditor und wählen Sie **Umwandlung hinzufügen** aus.

   Das Dialogfeld **Umwandlung hinzufügen** wird geöffnet.

2. Wählen Sie **Schreiben** aus und klicken Sie auf **OK**.

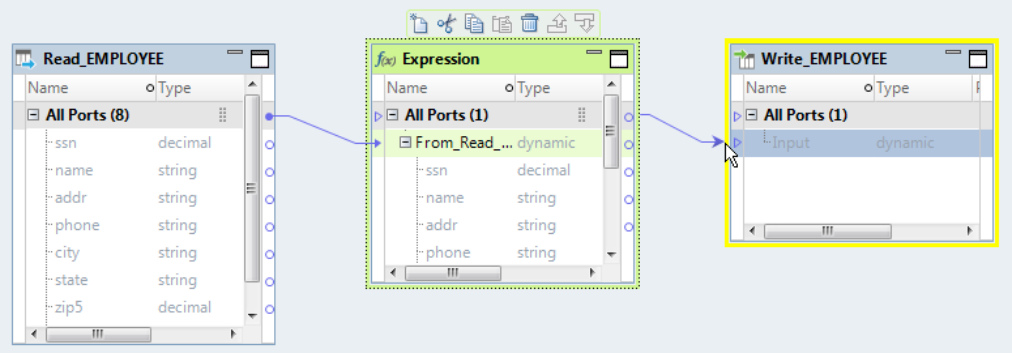
   Der Assistent **Neue Schreibumwandlung** wird geöffnet.

3. Wählen Sie **Physisches Datenobjekt** oder **Logisches Datenobjekt** aus und klicken Sie anschließend auf **Durchsuchen**.

   Das Dialogfeld **Datenobjekt auswählen** wird geöffnet.

4. Wählen Sie eine Datenquelle aus klicken Sie auf **OK**.
5. Wählen Sie **Zuordnungsfluss** aus und klicken Sie anschließend auf **Fertig stellen**.
6. Ziehen Sie Ports aus dem Port „Alle Ports“ des Upstreams in den **Eingabe**port der Schreibumwandlung.

Die folgende Abbildung zeigt, wie vorgelagerte Ports mit dem **Eingabe**port der Schreibumwandlung verbunden werden können:



Die Schreibumwandlung empfängt Spaltendefinitionen aus vorgelagerten Mapping-Objekten.

7. Um Zielobjektspalten mit Änderungen aus der Zieldatei zur Laufzeit dynamisch zu aktualisieren, wählen Sie **Datenobjektspalten zur Laufzeit aus Datenquelle abrufen** aus.

   Der Datenintegrationsdienst aktualisiert die Spaltenmetadaten für die Schreibumwandlung, wenn das Mapping ausgeführt wird.

Erstellen einer Schreibumwandlung aus einem Parameter

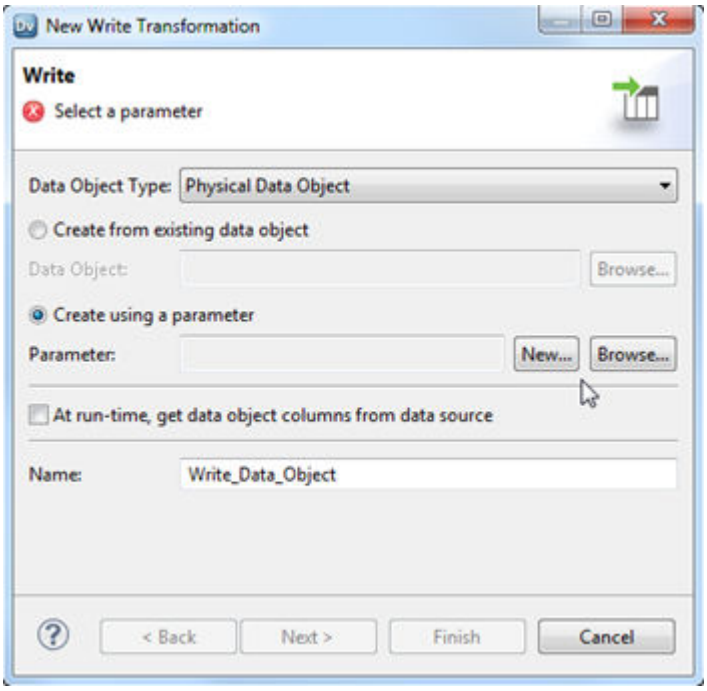
Bei einem Parameter handelt es sich um einen konstanten Wert, den Sie zwischen den Ausführungen der Mappings ändern können.

In einem dynamischen Mapping können Sie Parameter zum Ändern von Quellen und Zielen verwenden. Sie können Parameter auch für Eingaberegeln, Auswahlregeln, Laufzeitverknüpfungen und Umwandlungseigenschaften verwenden. Wenn Sie den Wert des Parameters ändern, erstellt der Datenintegrationsdienst das Ziel bzw. erstellt es neu, entsprechend dem im Parameter angegebenen Wert.

1. Klicken Sie im Editor mit der rechten Maustaste und wählen Sie **Umwandlung hinzufügen** aus.
2. Wählen Sie „Schreiben“ aus der Liste mit Umwandlungen aus und klicken Sie anschließend auf **OK**.

   Tipp: Geben Sie den ersten Buchstaben der Umwandlung zum Filtern der Liste ein.

   Das Dialogfeld „Neue Schreibumwandlung“ wird geöffnet.



3. Wählen Sie **Mithilfe eines Parameters erstellen** aus.
4. Suchen und wählen Sie den Parameter aus oder klicken Sie auf **Neu**, um einen neuen Parameter zu erstellen. Suchen und wählen Sie anschließend das Datenobjekt aus, aus dem der Parameter erstellt werden soll.
5. Optional können Sie die Option **Datenobjektspalten zur Laufzeit aus Datenquelle abrufen** auswählen, um die Umwandlungsspalten zur Laufzeit zu aktualisieren.
6. Optional können Sie auf **Weiter** klicken, um auszuwählen, wie Umwandlungsspalten definiert werden sollen.
7. Klicken Sie auf **Fertig stellen**.

Erstellen einer Schreibumwandlung aus einer vorhandenen Umwandlung

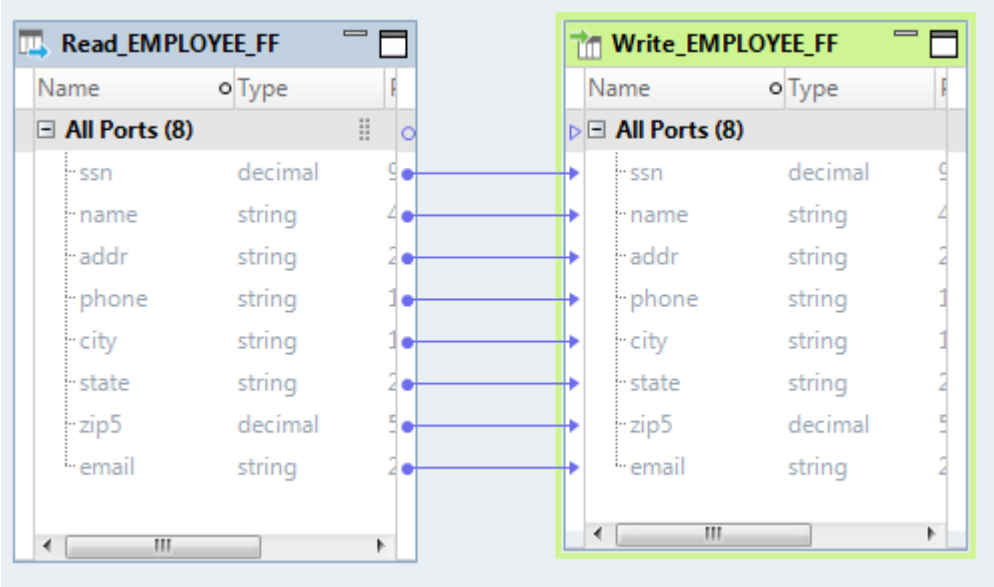
Wenn Sie eine Schreibumwandlung aus einer Umwandlung erstellen, die bereits im Mapping vorhanden ist, wird die Schreibumwandlung mit denselben Ports wie die Leseumwandlung erstellt.

1. Öffnen Sie ein Mapping im Editor.
2. Klicken Sie mit der rechten Maustaste auf eine Umwandlung im Zuordnungseditor und wählen Sie **Ziel hinzufügen** aus.

   Das Fenster **Ziel erstellen** wird geöffnet.
3. Wählen Sie als Objekttyp eine Einfachdatei oder ein relationales Datenobjekt aus.

   Hinweis: Ein Ziel kann nur auf Einfachdateien oder relationalen Ressourcen basieren.
4. Benennen Sie das neue Datenobjekt.
5. Optional können Sie auf **Durchsuchen** klicken, um einen Speicherort für das Datenobjekt auszuwählen.
6. Klicken Sie auf **Fertig stellen**.

Die folgende Abbildung zeigt eine Schreibumwandlung, die aus einer Leseumwandlung erstellt wurde, mit den jeweils gleichen Spalten:



KAPITEL 48

Umwandlungs-Caches

Dieses Kapitel umfasst die folgenden Themen:

- Umwandlungs-Caches – Übersicht, 660
- Cache-Typen, 661
- Cachedateien, 661
- Cachegröße, 662
- Erhöhen der Cachegröße durch den Datenintegrationsdienst, 664
- Cachegröße für partitionierte Caches, 665
- Optimierung der Cachegröße, 665

Umwandlungs-Caches – Übersicht

Der Datenintegrationsdienst weist Cache-Arbeitsspeicher für Aggregator-, Joiner-, Lookup, Rang- und Sortiererumwandlungen in einem Mapping zu. Der Datenintegrationsdienst erstellt Index- und Daten-Caches für Aggregator-, Joiner-, Lookup- und Rangumwandlungen. Der Datenintegrationsdienst erstellt einen Cache für die Sortiererumwandlung.

Sie können die Cachegrößen für diese Umwandlungen konfigurieren. Die Cachegröße bestimmt, wie viel Arbeitsspeicher der Datenintegrationsdienst jedem Umwandlungs-Cache zu Beginn der Mapping-Ausführung zuweist.

Wenn die Cachegröße größer ist als der auf dem Rechner verfügbare Speicher, kann der Datenintegrationsdienst nicht genügend Arbeitsspeicher zuweisen und die Mapping-Ausführung schlägt fehl.

Wenn die Cachegröße kleiner ist als die für die Ausführung der Umwandlung erforderliche Speichermenge, verarbeitet der Datenintegrationsdienst einen Teil der Umwandlung im Speicher und speichert die Überlaufdaten in Cachedateien. Wenn der Datenintegrationsdienst Cachedateien auf die Festplatte auslagert, erhöht sich die Verarbeitungszeit. Konfigurieren Sie für eine optimale Mapping-Leistung die Cachegröße so, dass der Datenintegrationsdienst die Umwandlung vollständig im Speicher ausführen kann.

Basierend auf der maximalen Speichermenge, die der Dienst zuweisen kann, berechnet der Datenintegrationsdienst den zur Laufzeit erforderlichen Speicher automatisch. Nachdem Sie ein Mapping im automatischen Cache-Modus ausgeführt haben, können Sie die Cachegrößen für die Umwandlungen anpassen. Analysieren Sie die Umwandlungsstatistiken im Mapping-Protokoll, um die für optimale Leistung erforderliche Cachegröße zu bestimmen, und konfigurieren Sie anschließend benutzerdefinierte Cachegrößen für die Umwandlungen.

Cache-Typen

Aggregator-, Joiner-, Lookup- und Rangumwandlungen erfordern einen Index-Cache und einen Daten-Cache. Der Datenintegrationsdienst speichert Schlüsselwerte im Index-Cache und Ausgabewerte im Daten-Cache. Sortiererumwandlungen erfordern einen einzelnen Cache. Der Datenintegrationsdienst speichert die Sortierschlüssel und die Daten, die im Sortierer-Cache sortiert werden sollen.

In der folgenden Tabelle werden die Informationstypen beschrieben, die der Datenintegrationsdienst in jedem Cache speichert:

Mapping-Objekt	Cache-Typen und Beschreibungen
Aggregator	- Index. Speichert die Gruppenwerte so, wie sie in „Nach Ports gruppieren" konfiguriert wurden. - Daten. Speichert Berechnungen basierend auf „Nach Ports gruppieren".
Joiner	- Index. Speichert alle Masterzeilen in der Join-Bedingung, die eindeutige Schlüssel aufweisen. - Daten. Speichert die Master-Quellzeilen.
Lookup	- Index. Speichert Informationen der Lookup-Bedingung. - Daten. Speichert Lookup-Daten, die nicht im Index-Cache gespeichert werden.
Rang	- Index. Speichert die Gruppenwerte so, wie sie in „Nach Ports gruppieren" konfiguriert wurden. - Daten. Speichert Ranking-Informationen basierend auf „Nach Ports gruppieren".
Sortierer	- Sortierer. Speichert Sortierschlüssel und -daten.

Cachedateien

Wenn Sie ein Mapping ausführen, erstellt der Datenintegrationsdienst für jede Aggregator-, Joiner-, Lookup-, Rang- und Sortiererumwandlung mindestens eine Cachedatei. Wenn der Datenintegrationsdienst eine Umwandlung nicht im Speicher ausführen kann, werden die Überlaufdaten in die Cachedateien geschrieben.

In der folgenden Tabelle werden die Typen von Cachedateien beschrieben, die der Datenintegrationsdienst für unterschiedliche Mapping-Objekte erstellt:

Mapping-Objekt	Cachedatei
Aggregator-, Joiner-, Lookup- und Rangumwandlungen	Der Datenintegrationsdienst erstellt die folgenden Typen von Cachedateien: - Eine Header-Datei für jeden Index- und Daten-Cache - Eine Datendatei für jeden Index- und Daten-Cache
Sortiererumwandlung	Der Datenintegrationsdienst erstellt eine Sortierer-Cachedatei.

Wenn Sie ein Mapping ausführen, trägt der Datenintegrationsdienst im Mapping-Protokoll eine Meldung mit dem Cachedatei-Namen und dem Umwandlungsnamen ein. Wenn ein Mapping abgeschlossen ist, gibt der Datenintegrationsdienst den Cache-Arbeitsspeicher frei und löscht im Normalfall die Cachedateien. Sie können unter folgenden Umständen Index- und Daten-Cachedateien im Cache-Verzeichnis finden:

- Sie konfigurieren die Lookup-Umwandlung für die Verwendung eines persistenten Cache.

- Das Mapping wurde nicht erfolgreich abgeschlossen. Bei der nächsten Ausführung des Mappings löscht der Datenintegrationsdienst die vorhandenen Cachedateien und erstellt neue.

Da das Schreiben in Cachedateien die Mapping-Leistung verringern kann, konfigurieren Sie die Cachegrößen für die Ausführung der Umwandlung im Speicher.

Cache-Dateiverzeichnis

Für Aggregator-, Joiner-, Lookup- und Rangumwandlungen erstellt der Datenintegrationsdienst die Cachedateien in dem Verzeichnis, das in der Cacheverzeichnis-Eigenschaft festgelegt wurde. Für Sortiererumwandlungen erstellt der Dienst die Cachedateien in dem Verzeichnis, das in der Arbeitsverzeichnis-Eigenschaft festgelegt wurde.

Wenn der Datenintegrationsdienst das Verzeichnis nicht finden kann, schlägt das Mapping fehl und im Mapping-Protokoll wird die Meldung eingetragen, dass die Cachedatei nicht erstellt oder geöffnet werden konnte. Der Datenintegrationsdienst erstellt möglicherweise mehrere Cachedateien. Die Anzahl der Cachedateien ist durch den im Cache-Verzeichnis maximal verfügbaren Festplattenspeicher begrenzt.

Cachegröße

Die Cachegröße bestimmt, wie viel Arbeitsspeicher der Datenintegrationsdienst jedem Umwandlungs-Cache zu Beginn der Mapping-Ausführung zuweist. Sie können eine Umwandlungs-Cachegröße für die Verwendung des automatischen Cache-Modus oder eines benutzerdefinierten Werts konfigurieren.

Automatische Cache-Größe

Eine Umwandlungs-Cache-Größe ist standardmäßig auf „Auto” eingestellt. Der Datenintegrationsdienst berechnet den zur Laufzeit erforderlichen Cache-Arbeitsspeicher automatisch. Sie legen den maximalen Arbeitsspeicher fest, den der Dienst zuweisen darf.

Der Datenintegrationsdienst verwendet die folgenden Richtlinien für das automatische Zuweisen von Arbeitsspeicher:

Umwandlungen mit höheren Verarbeitungszeiten wird mehr Arbeitsspeicher zugewiesen.

Der Datenintegrationsdienst weist Umwandlungen, die üblicherweise höhere Verarbeitungszeiten aufweisen, mehr Arbeitsspeicher zu. Der Datenintegrationsdienst weist z. B. der Sortierer-Umwandlung mehr Arbeitsspeicher zu, da die Sortierer-Umwandlung für die Ausführung normalerweise länger benötigt.

Dem Daten-Cache wird mehr Arbeitsspeicher zugewiesen als dem Index-Cache.

Aggregator-, Joiner-, Lookup- und Rang-Umwandlungen erfordern einen Index-Cache und einen Daten-Cache. Wenn der Datenintegrationsdienst den für die Umwandlung bestimmten Arbeitsspeicher zwischen dem Index- und dem Daten-Cache aufteilt, wird dem Daten-Cache mehr Arbeitsspeicher zugewiesen.

Sortierer-Umwandlungen erfordern einen einzelnen Cache. Der Dienst weist den für die Umwandlung bestimmten Arbeitsspeicher vollständig dem Sortierer-Cache zu.

Maximaler Speicher für automatische Cache-Größe

Sie können die maximale Speichermenge, die der Datenintegrationsdienst den Umwandlungs-Caches zuweisen kann, in der Eigenschaft „Maximale Speichergröße pro Abfrage" für die Datenintegrationsdienst-Module im Administrator tool festlegen.

Jedes Modul führt unterschiedliche Anfragetypen mit jeweils anderen Speicheranforderungen aus. Beispiel: Für Mapping- und Profilanfragen ist üblicherweise mehr Cache-Arbeitsspeicher erforderlich als für SQL-Dienst- oder Webdienst-Anfragen. Die Eigenschaft „Maximale Speichergröße pro Abfrage" kann für die folgenden Datenintegrationsdienst-Module konfiguriert werden:

- Zuordnungsdienstmodul
- Profilerstellungsdienst-Modul
- SQL-Dienstmodul
- Webdienstmodul

Hinweis: Zuordnungsdienstmodul-Anfragen beinhalten Mappings. Mappings werden von Mapping-Aufgaben aus innerhalb eines Arbeitsablaufs ausgeführt.

Für das Profilerstellungsdienst-Modul bestimmt die Eigenschaft „Maximale Speichergröße pro Abfrage" den maximalen Speicher, den der Datenintegrationsdienst für jedes Mapping, das für eine einzelne Profil-Anfrage ausgeführt wird, bereitstellen kann.

Bei den übrigen Modulen ist die Eigenschaft „Maximale Speichergröße pro Abfrage" von der Konfiguration des Datenintegrationsdiensts abhängig. Das Verhalten hängt von den Eigenschaften „Joboptionen starten" und „Maximale Speichergröße" auf dem Datenintegrationsdienst ab.

In der folgenden Tabelle wird das Verhalten der Eigenschaft „Maximale Speichergröße pro Abfrage" für Mapping-, SQL-Dienst- und Webdienstmodule basierend auf der Konfiguration des Datenintegrationsdiensts beschrieben:

Konfiguration des Datenintegrationsdiensts	Verhalten von „Maximale Speichergröße pro Abfrage"
Jobs werden in separaten lokalen oder ausgelagerten Systemprozessen ausgeführt oder „Maximale Speichergröße" beträgt 0 (Standard)	Maximale Speichermenge in Byte, die der Datenintegrationsdienst allen Umwandlungen zuweisen kann, die den automatischen Cache-Modus in einer einzelnen Anfrage verwenden. Der Wert, den Sie für „Maximale Speichergröße pro Abfrage" festlegen, bezieht sich nur auf Umwandlungen, die den automatischen Cache-Modus verwenden. Der Datenintegrationsdienst weist den Umwandlungen, für die Sie eine benutzerdefinierte Cache-Größe konfiguriert haben, den Speicher separat zu. Der von der Anfrage verwendete Gesamtspeicher kann den Wert für „Maximale Speichergröße pro Anfrage" überschreiten. Beispiel: Die Eigenschaft „Maximale Speichergröße pro Abfrage" ist auf 800 MB festgelegt. Ein Mapping besteht aus drei Umwandlungen, die Zwischenspeicherung in Caches erfordern. Sie konfigurieren zwei Umwandlungen für die Verwendung des automatischen Cache-Modus und die dritte Umwandlung so, dass insgesamt 500 MB für die Cache-Größen verwendet wird. Der Datenintegrationsdienst weist allen Umwandlungs-Caches einen Speicher von insgesamt 1.300 MB zu.
Jobs werden im Prozess des Datenintegrationsdiensts ausgeführt und „Maximale Speichergröße" ist größer als 0	Maximale Speichermenge in Byte, die der Datenintegrationsdienst einer einzelnen Anfrage zuordnen kann. Der Wert, den Sie für die Eigenschaft „Maximale Speichergröße pro Abfrage" festlegen, bezieht sich auf alle Umwandlungen. Der von der Anfrage verwendete Gesamtspeicher darf den Wert für „Maximale Speichergröße pro Anfrage" nicht überschreiten.

Wenn Sie die maximale Speichermenge, die für den automatischen Cache-Modus verwendet wird, erhöhen, erhöht sich gleichzeitig die maximale Cache-Größe, die für alle Anfragen an das Modul verwendet werden kann. Durch das Erhöhen der maximalen Speichermenge können Sie sicherstellen, dass keine Cachedateien auf die Festplatte ausgelagert werden. Allerdings weist der Datenintegrationsdienst einigen Anfragen mehr Speicher zu als erforderlich, da sich dieser Wert auf alle Anfragen bezieht.

Benutzerdefinierte Cachegröße

Sie können eine benutzerdefinierte Cachegröße für eine Umwandlung konfigurieren. Der Datenintegrationsdienst weist die benutzerdefinierte Speichermenge dem Umwandlungs-Cache zu Beginn der Mapping-Ausführung zu. Konfigurieren Sie bei Anpassung der Cachegröße den jeweiligen Wert in Byte.

Verwenden Sie bei der ersten Konfiguration der Cachegröße den automatischen Cache-Modus. Analysieren Sie nach der Ausführung des Mappings die Umwandlungsstatistiken im Mapping-Protokoll, um die für die Ausführung der Umwandlungen im Speicher erforderlichen Cachegrößen zu ermitteln. Wenn Sie die Cachegröße mit dem im Mapping-Protokoll angegebenen Wert konfigurieren, können Sie sicherstellen, dass kein zugewiesener Speicher verschwendet wird. Allerdings variiert die optimale Cachegröße je nach Größe der Quelldaten. Überprüfen Sie die Mapping-Protokolle nach weiteren Mapping-Ausführungen, um Änderungen an der Cachegröße zu überwachen. Wenn Sie eine benutzerdefinierte Cachegröße für eine wiederverwendbare Umwandlung konfigurieren, stellen Sie sicher, dass die Cachegröße für jede Verwendung der Umwandlung in einem Mapping optimal ist.

Um benutzerdefinierte Cachegrößen festzulegen, konfigurieren Sie die Werte der Cachegröße in den Umwandlungseigenschaften im Developer tool.

Erhöhen der Cachegröße durch den Datenintegrationsdienst

Der Datenintegrationsdienst erstellt jeden Speicher-Cache basierend auf der konfigurierten Cachegröße. In einigen Fällen wird die konfigurierte Cachegröße möglicherweise durch den Datenintegrationsdienst erhöht, da mehr Cache-Arbeitsspeicher erforderlich ist.

Die konfigurierte Cachegröße wird durch den Datenintegrationsdienst möglicherweise aus einem der folgenden Gründe erhöht:

Die konfigurierte Cachegröße ist kleiner als die Mindest-Cachegröße, die für die Ausführung des Vorgangs erforderlich ist.

Der Datenintegrationsdienst benötigt einen Mindestspeicher, um jedes Mapping zu initialisieren. Wenn die konfigurierte Cachegröße kleiner ist als die erforderliche Mindest-Cachegröße, erhöht der Datenintegrationsdienst die konfigurierte Cachegröße, um die Mindestanforderung zu erfüllen. Wenn der Datenintegrationsdienst den erforderlichen Mindestspeicher nicht zuweisen kann, schlägt das Mapping fehl.

Die konfigurierte Cachegröße ist kein Vielfaches der Cache-Seitengröße.

Der Datenintegrationsdienst speichert zwischengespeicherte Daten auf Cache-Seiten. Die zwischengespeicherten Seiten müssen genau in den Cache passen. Wenn Sie beispielsweise 10 MB (1.048.576 Byte) als Cachegröße konfigurieren und die Cache-Seitengröße 10.000 Byte beträgt, erhöht der Datenintegrationsdienst die konfigurierte Cachegröße auf 1.050.000 Byte, damit der Wert ein Vielfaches des Seitengrößenwerts von 10.000 Byte ist.

Wenn der Datenintegrationsdienst die konfigurierte Cachegröße erhöht, wird das Mapping weiterhin ausgeführt und die folgenden Meldungen werden im Mapping-Protokoll eingetragen:

```
INFO: MAPPING,    TE_7212,    Increasing [Index Cache] size for transformation
<transformation name> from <configured cache size> to <new cache size>.
INFO: MAPPING,    TE_7212,    Increasing [Data Cache] size for transformation
<transformation name> from <configured cache size> to <new cache size>.
```

Cachegröße für partitionierte Caches

Wenn Sie über die Partitionierungsoption verfügen, erstellt die Cache-Partitionierung einen separaten Cache für jede Partition, die eine Aggregator-, Joiner-, Rang-, Lookup- oder Sortiererumwandlung ausführt. Während der Cache-Partitionierung speichert jede Partition andere Daten in einem separaten Cache. Wenn der Datenintegrationsdienst Cache-Partitionierung für diese Umwandlungen verwendet, teilt der Dienst die zugewiesene Cachegröße zwischen den Partitionen auf.

Sie legen die Cachegröße für die Umwandlung beispielsweise auf 100 MB fest. Der Datenintegrationsdienst verwendet vier Partitionen zum Ausführen der Umwandlung. Der Dienst teilt den Wert der Cachegröße so auf, dass jede Partition maximal 25 MB als Cachegröße verwendet.

Optimierung der Cachegröße

Konfigurieren Sie für eine optimale Mapping-Leistung die Cachegrößen so, dass der Datenintegrationsdienst die Umwandlung vollständig im Speicher ausführen kann.

Um optimale Cachegrößen zu konfigurieren, führen Sie die folgenden Aufgaben durch:

1. Legen Sie die Tracingebene auf die ausführliche Initialisierung fest.
2. Führen Sie das Mapping im automatischen Cache-Modus aus.
3. Analysieren Sie die Caching-Leistung im Mapping-Protokoll.
4. Konfigurieren Sie benutzerdefinierte Werte für die Cachegrößen.

Schritt 1. Festlegen der Tracingebene auf die ausführliche Initialisierung

Wenn Sie die Tracingebene im Developer tool auf die ausführliche Initialisierung festlegen, schreibt der Datenintegrationsdienst die Umwandlungsstatistiken in das Mapping-Protokoll. In den Umwandlungsstatistiken werden die für optimale Leistung erforderlichen Cachegrößen aufgelistet. Standardmäßig ist die Tracingebene auf „Normal” eingestellt.

Legen Sie die Tracingebene mithilfe einer der folgenden Methoden auf die ausführliche Initialisierung fest:

- Ändern Sie die erweiterten Eigenschaften für jede Umwandlung, die einen Cache verwendet.
- Ändern Sie die Standardkonfigurationseigenschaften für Mappings, wenn Sie das Mapping das erste Mal über das Developer tool ausführen möchten. Weitere Informationen finden Sie im *Informatica Developer-Tool-Handbuch*.
- Ändern Sie die erweiterten Eigenschaften für eine Anwendung, die das Mapping enthält, wenn Sie das bereitgestellte Mapping das erste Mal über die Befehlszeile ausführen möchten. Weitere Informationen finden Sie im *Informatica Developer-Tool-Handbuch*.

Schritt 2. Ausführen des Mappings im automatischen Cache-Modus

Verwenden Sie bei der ersten Mapping-Ausführung den automatischen Cache-Modus für die Umwandlungs-Cachegrößen.

Sie können das Mapping über das Developer tool ausführen. Wahlweise können Sie das Mapping in eine Anwendung hinzufügen und die Anwendung anschließend dem Datenintegrationsdienst bereitstellen, sodass Sie das Mapping über die Befehlszeile ausführen können.

Schritt 3. Analysieren der Caching-Leistung

Analysieren Sie nach der Mapping-Ausführung im automatischen Cache-Modus die Umwandlungsstatistiken im Mapping-Protokoll, um die für eine optimale Mapping-Leistung erforderlichen Cache-Größen zu ermitteln.

Wenn eine Aggregator-, Joiner-, Lookup- oder Rang-Umwandlung etwas auf die Festplatte auslagert, werden im Mapping-Protokoll die Index- und Daten-Cache-Größen angegeben, die für die Ausführung der Umwandlung im Speicher erforderlich sind. Beispiel: Sie führen die Aggregator-Umwandlung „AGG_TRANS" aus. Das Mapping-Protokoll enthält folgenden Text:

```
CMN_1791, The index cache size that would hold [1098] aggregate groups of input rows for
[AGG_TRANS], in memory, is [286720] bytes
CMN_1790, The data cache size that would hold [1098] aggregate groups of input rows for
[AGG_TRANS], in memory, is [1774368] bytes
```

Das Protokoll gibt an, dass der Index-Cache 286.720 Byte und der Daten-Cache 1.774.368 Byte benötigt, um die Umwandlung im Speicher ohne Auslagern auf die Festplatte auszuführen.

Wenn eine Sortierer-Umwandlung etwas auf die Festplatte auslagert, gibt das Mapping-Protokoll an, dass der Datenintegrationsdienst mehrere Übergaben der Quelldaten durchgeführt hat. Der Datenintegrationsdienst führt mehrere Übergaben der Daten durch, wenn für das Abschließen der Sortierung eine Auslagerung auf die Festplatte erforderlich ist. Die Meldung enthält Informationen über die für eine einzelne Übergabe erforderliche Anzahl an Byte. Bei einer einzelnen Übergabe liest der Datenintegrationsdienst die Daten nur einmal und führt die Sortierung ohne Auslagern auf die Festplatte aus.

Beispiel: Sie führen die Sortierer-Umwandlung „SRT_TRANS" aus. Das Mapping-Protokoll enthält folgenden Text:

```
SORT_40427,    Sorter Transformation [SRT_TRANS] required 2-pass sort (1-pass temp I/O:
13126221824 bytes). You may try to set the cache size to 14128 MB or higher for 1-pass
in-memory sort.
```

Das Protokoll gibt an, dass der Sortierer-Cache 14.128 MB benötigt, sodass der Datenintegrationsdienst nur eine Übergabe der Daten durchführt.

Schritt 4. Konfigurieren von benutzerdefinierten Cachegrößen

Um eine optimale Leistung zu erzielen, konfigurieren Sie die Umwandlungs-Cachegrößen mit den im Mapping-Protokoll angegebenen Werten. Aktualisieren Sie die Umwandlungseigenschaften der Index- und Daten-Cachegrößen im Developer tool.

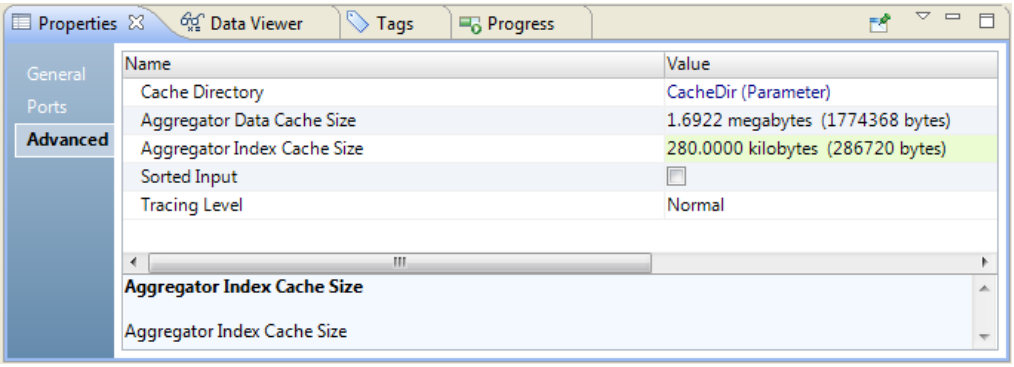
1. Öffnen Sie im Developer tool die wiederverwendbare oder nicht wiederverwendbare Umwandlung.

2. Öffnen Sie die Cachegrößeneigenschaften abhängig von den folgenden Umwandlungstypen:

Option	Beschreibung
Wiederverwendbare Aggregator-, Joiner-, Rang- oder Sortiererumwandlung	Klicken Sie auf die Ansicht **Erweitert**.
Nicht wiederverwendbare Aggregator-, Joiner-, Rang- oder Sortiererumwandlung	Klicken Sie auf die Registerkarte **Erweitert** in der Ansicht **Eigenschaften**.
Wiederverwendbare Lookup-Umwandlung	Klicken Sie auf die Ansicht **Laufzeit**.
Nicht wiederverwendbare Lookup-Umwandlung	Klicken Sie auf die Registerkarte **Laufzeit** in der Ansicht **Eigenschaften**.

3. Geben Sie die Werte aus der Empfehlung des Mapping-Protokolls für die Index- und Daten-Cachegrößen in Byte an.

   Die folgende Abbildung zeigt eine nicht wiederverwendbare Aggregatorumwandlung mit benutzerdefinierten Werten für die Index- und Daten-Cachegrößen:



4. Klicken Sie auf **Datei** > **Speichern**.

ANHANG A

Umwandlungs-Delimiter

- Umwandlungs-Delimiter - Übersicht, 668

Umwandlungs-Delimiter - Übersicht

Mit Umwandlungs-Delimitern werden Trennungen zwischen Strings angegeben.

In der folgenden Tabelle werden die Delimiter aufgelistet, die von Umwandlungen zum Analysieren und Schreiben von Strings verwendet werden:

Delimitername	Delimitersymbol
Bei Symbol	@
Komma	,
Gedankenstrich	-
Doppelte Anführungszeichen	"
Schrägstrich	/
Punkt	.
Hash	#
Pipe	|
Semikolon	;
Einfache Anführungszeichen	'
Leerzeichen	[**Leerzeichen**]
Tabulator	[**Tab**-Taste]
Unterstrich	_

INDEX

"Early selection"-Optimierung
SQL-Umwandlung 572
Web-Dienst-Verbraucher-Umwandlung 615

A

Abfragen
Lookup-Umwandlung 355
Überschreiben des Lookups 356
Abgelehnte Datensätze
Ausnahmeumwandlung bei fehlerhaftem Datensatz 126
Abgeleitete Typen
Parsen von SOAP-Meldungen 628
Web-Dienste 642
Abhängige Maskierung
Beschreibung 179
Abhängige Spalte
Datenmaskierung 179
Ablehnungsdatei
Update-Strategien 596
ABORT-Funktion
Verwenden 49
Aggregator-Umwandlung
Aggregat-Ausdrücke 107
Aggregatfunktionen 108
Eingebettete Aggregatfunktionen 108
Entwickeln 106
Erstellen einer nicht wiederverwendbaren 115
Erstellen einer wiederverwendbaren 114
Fehlerbehebung 116
Nach Ports gruppieren 109
Nicht-Aggregat-Ausdrücke 111
Ports 106
Sortieren von Daten 113
Sortierte Eingabe 112
Übersicht 105
Update-Strategie-Kombination 596
Aggregatorumwandlung
Cachegröße 660
Caches 112
Dynamische Mappings 106
Erweiterte Eigenschaften 114
Tipps 115
Verwenden von Variablen 44
Aktive Umwandlungen
Beschreibung 30
Java 266, 267
Rang 494
Aktualisieren, andernfalls einfügen (Eigenschaft)
Beschreibung 397
Aktualisierungsstrategie-Umwandlung
Ausdrücke 595
dynamische Mappings 595
Aktueller Wert (Eigenschaft)
Sequenzgeneratorumwandlung 545
Alle Elemente
Web-Dienst-Verbraucher-Umwandlung 604, 607
Alle-Gruppe
Anzeigen in der REST-Webdienst-Verbraucher-Umwandlung 530
Anzeigen in Web-Dienst-Verbraucher-Umwandlung 605, 609
Am Datenende (Registerkarte)
Java-Umwandlungen 279
Ansicht „Verweise“
Relational-in-hierarchisch-Umwandlung 515
Umwandlung von hierarchisch in relational 263
Ansicht der Datenprozessor-Ereignisse
Datenprozessorumwandlung 209
Ansicht der Einstellungen
Datenprozessorumwandlung 202
Ansicht für Ereignisse, Datenprozessor
Ereignisprotokoll anzeigen 211
Ansicht Problemzuordnung
Ausnahmeumwandlung bei fehlerhaftem Datensatz 128
Antwort-Code
REST-Webdienst-Verbraucher-Umwandlung 526
anyAttribute-Attribute
Web-Dienst-Verbraucher-Umwandlung 604, 607
anyType
Zuordnen von Ports 642
anyType-Elemente
Parsen 627
Web-Dienst-Verbraucher-Umwandlung 604, 607
Anzahl der betroffenen Zeilen
SQL-Umwandlung 563
Anzahl zwischengespeicherter Werte
Sequenzgenerator-Eigenschaftswert 545
API-Methode für Java-Ausdrücke
getResultDataType 308
API-Methoden
Java-Umwandlung 287
Java-Umwandlungen 287
API-Methoden für Java-Ausdrücke
defineJExpression 305
getBytes 308
getDouble 308
getInt 308
getLong 308
getResultMetadata 309
getStringBuffer 309
invoke 309
isResultNull 310
Ausdrücke
Eingeben 41
Hinzufügen von Kommentaren 42
Hinzufügen zu einem Port 41
In Umwandlungen 39
Java-Umwandlungen 298
Validieren 42
Vereinfachen 44
Ausdruckseditor
Beschreibung 41

Ausdruckseditor *(Fortsetzung)*
testen, Ausdrücke 243
Validieren von Ausdrücken 42
Ausdrucksmaskierung
Beispiel für eine wiederholbare Maskierung 173
Beschreibung 172
Regeln und Richtlinien 174
Wiederholbare Maskierung 172
Ausdrucksparameter
für eine Join-Bedingung 321
Lookup-Bedingung 371
Ausdrucksumwandlung
dynamischer Ausdruck 248
Einstellungen für Ausgabeport 249
Erweiterte Eigenschaften 252
Porttypen 242
testen 243
Übersicht 241
Verwenden von Variablen 44
Ausgabe-Mapping
REST-Webdienst-Verbraucher-Umwandlung 529
Web-Dienst-Verbraucher-Umwandlung 607
Ausgabegruppe bearbeiten (Dialogfeld)
Normalizer-Umwandlung 475
Ausgabegruppen
Ausnahmeumwandlung bei fehlerhaftem Datensatz 121
Normalizer-Umwandlung 473
Ausgabeports
Fehlerbehandlung 47
Java-Umwandlung 269
Java-Umwandlungen 269
Standardwerte 47
Ausgabesteuerungseinstellungen
Datenprozessorumwandlung 205
Ausnahmeumwandlung
Ansicht Problemzuordnung 128
Ausnahmeumwandlung bei dupliziertem Datensatz
Ausgabegruppen 232
automatische Konsolidierung 229
Beispiel 234
Beispiel für die Cluster-Ausgabe 238
Beispiel für Konfigurationseinstellungen 236
Beispiel für Mapping 234
Cluster 229
Datensätze sortieren 233
Erstellen einer duplizierten Datensatztabelle 230
erweiterte Eigenschaften 233
Konfigurationsansicht 229
konfigurieren 240
Ports 231
Standardausgabegruppe 232
Testausgabe 237
Tracingebenen 233
Übersicht 227
Ausnahmeumwandlung bei fehlerhaftem Datensatz
Ausgabebeispiel intakter Datensätze 135
Ausgabegruppen 121
Ausgabeports 126
Beispiel 131
Beispiel - Ausgabe 134
Beispiel für Ausgabeprobleme 134
Beispiel für die Ausgabe der fehlerhaften Datensätze 134
Eingabeports 125
erweiterte Eigenschaften 129
Konfigurationsansicht 126
konfigurieren 130
Leere Qualitätsproblemports 121
Mapping 123
Ausnahmeumwandlung bei fehlerhaftem Datensatz *(Fortsetzung)*
Mapplet - Beispiel 131
Portgruppen 125
Prozessablauf 122
Qualitätsprobleme 124
Übersicht 120
Auswahlelemente
Anzeigen in der REST-Webdienst-Verbraucher-Umwandlung 530
Anzeigen in Web-Dienst-Verbraucher-Umwandlung 605, 609
Beschreibung 643
Parsen von SOAP-Meldungen 630
Auswahlkriterien
Portauswahl 245, 369
Auswahlregeln
dynamische Mappings 246, 370
Joiner-Umwandlung 316
Portauswahlen 245, 369
Automatische Cache-Größe
Beschreibung 662
automatische Konsolidierung
Ausnahmeumwandlung bei dupliziertem Datensatz 229

B

Bedingungen
Joiner-Umwandlung 318
Router-Umwandlung 537
Bedingungstyp
Einfach für Joiner-Umwandlung 319
Erweitert für Joiner-Umwandlung 320
Bei Eingabe (Registerkarte)
Java-Umwandlungen 278
Beispiel
dynamischer Ausdruck 248
Beispielquelldatei
Definieren in der Datenprozessorumwandlung 214
Beispielquelle
In der Datenprozessorumwandlung erstellen 216
Beispiels-Eingabedatei
Datenprozessorumwandlung 199
Benachrichtigung - Ereignis
Datenprozessorumwandlung 209
Benutzerdefinierte Gruppe
Router-Umwandlung 536
Benutzerdefinierte Methoden
Java-Umwandlungen 277
Benutzerprotokoll
Datenprozessorumwandlung 211
Berechnungen
Verwenden von Variablen mit 45
Bereich
Maskieren von numerischen Werten 184
Master und Detail 316
Portauswahl 245, 369
Bereich "Vorgangseingabe"
Anpassen der Web-Dienst-Verbraucher-Umwandlung 605
Berücksichtigungen
Java-Umwandlungen 268
Bezeichnungsumwandlung
Übersicht 338
Bibliothek
In der Datenprozessorumwandlung erstellen 215
Bindung
WSDL-Dateielement 599
Blockierende Umwandlungen
Beschreibung 38

C

Cache
  Lookup-Umwandlung 362
Cache-Dateigröße
  Ausnahmeumwandlung bei dupliziertem Datensatz 233
  Konsolidierungsumwandlung 154
  Schlüsselgeneratorumwandlung 337
  Zuordnungsumwandlung 119
Cache-Dateiverzeichnis
  Ausnahmeumwandlung bei dupliziertem Datensatz 233
  Konsolidierungsumwandlung 154
  Schlüsselgeneratorumwandlung 337
  Zuordnungsumwandlung 119
Cache-Größe
  automatisch 662
Cache-Verzeichnis
  Datenmaskierungsumwandlung 193
Cachedateien
  Übersicht 661
  Verzeichnis 662
Cachegröße
  Datenmaskierungsumwandlung 193
caches
  statischer Lookup-Cache 384
Caches
  Cachedateien 660
  Daten 661
  dynamischer Lookup-Cache 389
  Erhöhen der Größe durch den Datenintegrationsdienst 664
  Größe 662
  Index 661
  Lookup-Umwandlung 382
  Optimierung der Größe 665
  Partitionierung 665
  Typen 661
  Übersicht 660
  Umwandlungen 660
  Verzeichnis für Dateien 662
Cluster
  Ausnahmeumwandlung bei dupliziertem Datensatz 229
Cluster-Daten
  Ausgabegruppe 232
Codeausschnitte
  Erstellen für Java-Umwandlungen 275
Codierungsrichtlinien
  Datenprozessorumwandlung 205
Cookie-Authentifizierung
  REST-Webdienst-Verbraucher-Umwandlung 526
  Web-Dienst-Verbraucher-Umwandlung 603

D

Data Integration Service)
  Neustartmodus 295
Dateiausgabeport
  Datenprozessorumwandlung 200
Dateieingabeport
  Datenprozessorumwandlung 199
Daten
  Temporär speichern 44
Daten-Caches
  Umwandlungen 661
Datenmaskierungsumwandlung
  Abhängige Datenmaskierung 179
  Ausdrucksmaskierung 172
  Ausdrucksmaskierung für Wörterbuchnamen 172
Datenmaskierungsumwandlung *(Fortsetzung)*
  Bereich 184
  Beschreibung 169
  Cache-Verzeichnis 193
  Cachegröße 193
  Datenmaskierungsumwandlung 193
  Eigenschaften der Substitutions-Maskierung 177, 178
  Eindeutige Ausgabe 193
  Gemeinsam genutzte Speichertabelle 193
  Laufzeiteigenschaften 193
  Maskieren von Datumswerten 185
  Maskieren von E-Mail Adressen 187
  Maskieren von IP-Adressen 188
  Maskieren von Kreditkarten 186
  Maskieren von Sozialversicherungsnummern 189
  Maskieren von Sozialversicherungsnummern (Social Security Numbers, SSN) 188
  Maskieren von Sozialversicherungsnummern (SSN) 191
  Maskieren von Telefonnummern 188
  Maskieren von URLs 189
  Maskierungsformat 182
  Maskierungstechniken 170
  Quellstringzeichen 183
  Richtlinien für die Ausdrucksmaskierung 174
  Speicher-Commit-Intervall 193
  Speichertabelle 173
  Speichertabellen 177
  Spezielle Maskierungsformate 186
  Standardwertdatei 190
  Substitutions-Maskierung 176
  Unschärfe 185
  Wiederholbare Ausdrucksmaskierung 172
  Wiederholbare SSN 189
  Wiederholbaren SIN-Nummern 190
  Wörterbuch für Substitutions-Maskierung 176
  Zufällige Maskierung 170
Datenobjekt
  doppelt parametrisiert 366
  parametrisieren 506
Datenprozessor-Ereignisse-Ansicht
  Ereignisprotokoll anzeigen 211
Datenprozessor-Hex-Quelle-Ansicht
  Beschreibung 198
Datenprozessorumwandlung
  Einstellungen für die Codierung 203
  Ansichten 198
  Ausgabeports 200
  Ausgabesteuerungseinstellungen 205
  Benutzerprotokolle 211
  Beschreibung 197, 198
  Eingabeports 199
  Einstellungen 202
  erstellen 212
  exportieren als Dienst 217
  Ports 199
  Ports für Dienstparameter 200
  Startkomponente 201
  testen im Daten-Viewer 216
  Verarbeitungseinstellungen 206
  XMap-Einstellungen 207
  XML-Einstellungen 207
Datensätze
  Normalizer-Umwandlung 468
Datentypen
  Java-Umwandlungen 267
Datetime-Werte
  Datenmaskierung 175

Datumswerte
Zufällige Datenmaskierung 171
defineJExpression
API-Methode für Java-Ausdrücke 305
defineJExpression-Methode
Java-Umwandlungen 288
Denormalisierte Ausgabe
Parsen von SOAP-Meldungen 626
Denormalisierte Eingabe
Web-Dienst-Ports 640
Dienstparameterport
Datenprozessorumwandlung 199
dynamische Ausdrücke
Beispiel 248
Einstellungen für Ausgabeport 249
Übersicht 248
dynamische Mappings
Portauswahl 246, 370
Aktualisierungsstrategie-Umwandlung 595
Auswahlregeln 246, 370
Filterumwandlung 255
Joiner-Umwandlungen 315
Lookup-Umwandlungen 363
Router-Umwandlung 535
Sortierer-Umwandlung 550
Dynamische Mappings
Aggregatorumwandlung 106
Lookup-Bedingung 368
Rangumwandlung 495
dynamische Ports
Join-Bedingung 321
Lookup-Bedingung 364
dynamische relationale Datenobjekte
erstellen, Leseumwandlungen 509
Dynamische URL
Web-Dienst-Verbraucher-Umwandlung 603
dynamischer Ausdruck
erstellen 250
dynamischer Lookup-Cache
Anwendungsfälle 390
Beschreibung 389
Lookup-SQL-Überschreibung 395
Verwenden von Einfachdateiquellen 389

E

EDataType-Klasse
Java-Ausdrücke 304
Eigenschaft sortieren
Ausnahmeumwandlung bei dupliziertem Datensatz 233
Konsolidierungsumwandlung 154
Eigenschaften
Leseumwandlung
Eigenschaften 501
Schreibumwandlung
Eigenschaften 649
eindeutige Ausgabe
Sortierer-Umwandlung 551
Eindeutige Ausgabe
Datenmaskierungsumwandlung 178
eindeutige Datensatztabelle
Erstellen 240
Einfache Schnittstelle
Beispiel 302
Java-Ausdrücke 301
Java-Umwandlungs-API-Methoden 301
Einfacher Bedingungstyp
Joiner-Umwandlung 319
Einfügen, andernfalls aktualisieren (Eigenschaft)
Beschreibung 396
Eingabe-Mapping
REST-Webdienst-Verbraucher-Umwandlung 527
Web-Dienst-Verbraucher-Umwandlung 604
Eingabehierarchie
Normalizer-Umwandlung 469
Eingabeports
Java-Umwandlungen 269
Standardwerte 47
Eingabeports, Bereich
Generieren von SOAP-Meldungen 633
Eingabezeilen
Abrufen des Zeilentyps für 290
Einschränkungen
Quelloptimierung 508
Einstellungen für Ausgabeport
Beschreibung 249
Einstellungen für die Codierung
Datenprozessorumwandlung 203
Elemente
Union 645
Elemente eines abgeleiteten Typs
Web-Dienst-Verbraucher-Umwandlung 604, 607
Endpunkt-URL
REST-Webdienst-Verbraucher-Umwandlung 525
Web-Dienst-Verbraucher-Umwandlung 603
Endwert (Eigenschaft)
Sequenzgeneratorumwandlung 545
Entscheidungsumwandlung
Beispiel für eine Strategie bei Qualitätsproblemen 124
Erweiterte Eigenschaften 226
Übersicht 220
Entwickeln
Java-Umwandlungen 268
Entwicklungszeit, Ereignisprotokoll
Beschreibung und Ort 210
Ereignisansicht
Datenprozessorumwandlung 209
Ereignisprotokoll
anzeigen 211
Ereignisprotokoll, Entwicklungszeit
Beschreibung und Ort 210
Ereignisse
Datenprozessorumwandlung 209
Ereignistypen
Datenprozessorumwandlung 209
Ergebnissätze
Ausgabeparameterplatzierung 581
Beispiel für gespeicherte Prozeduren 579
SQL-Umwandlung 579
ERROR-Funktion
Verwenden 49
Ersatzzeichen für Ergebniszeichenfolge
Datenmaskierungsumwandlung 184
erstellen
Ports 37
Erstellen
Java-Umwandlungen 283, 284
Erstellen einer Tabelle mit fehlerhaften Datensätzen
Ausnahmeumwandlung bei fehlerhaftem Datensatz 128
erweiterte Eigenschaften
Ausnahmeumwandlung bei dupliziertem Datensatz 233
Ausnahmeumwandlung bei fehlerhaftem Datensatz 129
Konsolidierungsumwandlung 154
Schlüsselgeneratorumwandlung 337

erweiterte Eigenschaften *(Fortsetzung)*
Zuordnungsumwandlung 119
Erweiterte Eigenschaften
Java-Umwandlungen 270
REST-Webdienst-Verbraucher-Umwandlung 531
SQL-Umwandlung 563
Web-Dienst-Verbraucher-Umwandlung 611
Erweiterte Schnittstelle
Aufrufen von Java-Ausdrücken 303
Beispiel 306
EDataType-Klasse 304
Java-Ausdrücke 303
JExpression-Klasse 306, 307
JExprParaMetadata-Klasse 305
Erweiterter Bedingungstyp
Joiner-Umwandlung 320

F

failSession-Methode
Java-Umwandlungen 289
Fehlende Werte
Ersetzen mit Sequenzgenerator 543
Fehler
Behandlung 51
Zunehmender Schwellenwert in Java-Umwandlungen 291
Fehler im Benutzercode
Java-Umwandlungen 286
Fehlerbehebung
Java-Umwandlungen 285
Fehlerzähler
Erhöhen für Java-Umwandlungen 291
Fehlgeschlagene Mappings
Java-Umwandlungen 289
Fehlschlag - Ereignis
Datenprozessorumwandlung 209
Feld-Match-Analyse
definierte Feldanalyse 406
Prozessablauf 429
Filtern von Quellzeilen
Lookup-Umwandlung 358
Filterumwandlung
dynamische Mappings 255
Erweiterte Eigenschaften 258
Filterbedingung 255
Filterbedingungsparameter 256
Tipps für die Leistung 258
Übersicht 254
Zeilen mit Nullwerten 257
Fortfahren bei SQL-Fehler
SQL-Umwandlung 567, 571
Fremdschlüssel
Erstellen mit einer Sequenzgeneratorumwandlung 542
Full-Code (Registerkarte)
Java-Kompilierungsfehler 285
Java-Umwandlungen 280
Funktionen (Registerkarte)
Java-Umwandlungen 280

G

GCID
Beschreibung 467
Normalizer-Umwandlung, Beispiel 473
Gemeinsam genutzte Speichertabelle
Datenmaskierungsumwandlung 193
Gemeinsam genutzter Lookup-Cache
Partitionierung, Richtlinien 387
Richtlinien 387
generateRow-Methode
Java-Umwandlungen 289
Generieren einer Ausgabezeile
Java-Umwandlungen 289
Generieren von Java-Code
Java-Ausdrücke 300
Generierte Schlüssel
Web-Dienst-Ausgabegruppen 626
Generische Fehlerausgabe
Aktivieren der Web-Dienst-Verbraucher-Umwandlung 611
Generische SOAP-Fehler
Web-Dienst-Verbraucher-Umwandlung 613
Gespeicherte Prozeduren
Beispiel 581
Eingabe- und Ausgabeports 577
Ergebnissatz-Beispiele 579
Parameter 578
Rückgabewert 578
Rückgabewert-Port 577
Schreiben in Variablen 46
SQL-Umwandlung 577
getBytes-Methode
Java-Umwandlungen 308
getDouble-Methode
Java-Umwandlungen 308
getInRowType-Methode
Java-Umwandlungen 290
getInt-Methoden
Java-Umwandlungen 308
getLong-Methode
Java-Umwandlungen 308
getMetada-Methode
Java-Umwandlungen 291
getResultDataType-Methode
Java-Umwandlungen 308
getResultMetadata-Methode
Java-Umwandlungen 309
getStringBuffer-Methode
Java-Umwandlungen 309
Gleichzeitige Web-Dienst-Anfragemeldungen
Aktivieren der Web-Dienst-Verbraucher-Umwandlung 611
Groß-/Kleinschreibungsumwandlung
Übersicht 136
Gruppen
Benutzerdefiniert 536
Hinzufügen zur Router-Umwandlung 539
Router-Umwandlung 536
Gruppen in der Match-Analyse 407
Gruppenfilterbedingung
Router-Umwandlung 537
GZip
Komprimieren von SOAP-Meldungen 614

H

Helfer (Registerkarte)
Java-Umwandlungen 277, 278
Hex-Quelle-Ansicht, Datenprozessor
Beschreibung 198
Hohe Genauigkeit der Verarbeitung
Java-Umwandlungen 270
HTTP-Antwortcode
REST-Webdienst-Verbraucher-Umwandlung 526

HTTP-Fehlerausgabe
Aktivieren der Web-Dienst-Verbraucher-Umwandlung 611
HTTP-Kopfzeile
Hinzufügen zur REST-Webdienst-Verbraucher-Umwandlung 525
Hinzufügen zur Web-Dienst-Verbraucher-Umwandlung 603
HTTP-Verbindung
REST-Webdienste 531
Human-Aufgaben
Ausnahmen bei fehlerhaftem Datensatz 124

I

Identitäts-Match-Analyse
Ausgabeeigenschaften für Indexdatenanalyse 452
Prozessablauf 443
Identitätsvergleichsanalyse
Ausgabeeigenschaften für Indexdatenanalyse 452
definierte Identitätsanalyse 406
Masterdatensätze 414
Persistenter Speicher von Indexdaten 414
Persistenzstatusbeschreibungen 423, 424
Persistenzstatuscodes 423, 424
Regeln und Richtlinien für persistente Indexdaten 414
IIF-Funktion
Ersetzen fehlender Schlüssel mit einer
Sequenzgeneratorumwandlung 543
Importe (Registerkarte)
Java-Umwandlungen 276, 278
incrementErrorCount-Methode
Java-Umwandlungen 291
Index-Caches
Umwandlungen 661
Initialisieren
Variablen 47
Inkrementschritt (Eigenschaft)
Die Eigenschaft „Sequenzgeneratorumwandlung“ 545
Instanzvariablen
Java-Umwandlungen 277
invoke
API-Methode für Java-Ausdrücke 309
invokeJExpression
API-Methode 302
invokeJExpression-Methode
Java-Umwandlungen 292
isNull-Methode
Java-Umwandlungen 293
isResultNull-Methode
Java-Umwandlungen 310

J

Java-Ausdrücke
Aufrufen 292
Aufrufen mit einfacher Schnittstelle 302
Aufrufen mit erweiterter Schnittstelle 303
EDataType-Klasse 304
Einfache Schnittstelle 301
Einfache Schnittstelle - Beispiel 302
Erstellen 300
Erstellen im Dialogfeld "Funktion definieren" 300
Erweiterte Schnittstelle 303
Erweiterte Schnittstelle - Beispiel 306
Generieren 299
Generieren von Java-Code 300
invokeJExpression-API-Methode 302
Java-Umwandlungen 298
Java-Ausdrücke *(Fortsetzung)*
JExpression-Klasse 306, 307
JExprParaMetadata-Klasse 305
Konfigurieren 299
Konfigurieren von Funktionen 300
Regeln und Richtlinien 302, 304
Regeln und Richtlinien zum Aufrufen 292
Java-Code
In Java-Umwandlungen 273
Suchen von Fehlern 285
Java-Codeausschnitte
Erstellen für Java-Umwandlungen 275
Java-Pakete
Importieren 276
Java-Umwandlung
API-Methoden 287
Übersicht 266
Java-Umwandlungen
Abrufen des Eingabezeilentyps 290
Abrufen von Metadaten 291
active 267
Am Datenende (Registerkarte) 279
API-Methoden 287
Ausgabeports 269
Bei Eingabe (Registerkarte) 278
Datentypkonvertierung 267
defineJExpression-Methode 288
Eingabeports 269
Entwickeln 268
Erkennen der Ursache von Kompilierungsfehlern 286
Erstellen 283, 284
Erstellen von Java-Codeausschnitten 275
Erstellen von Ports 269
Erweiterte Eigenschaften 270
failSession-Methode 289
Fehler im Benutzercode 286
Fehlerbehebung 285
Fehlgeschlagene Mappings in 289
Festlegen von Nullwerten in 295
Full-Code (Registerkarte) 280
Funktion (Registerkarte) 280
generateRow-Methode 289
getInRowType-Methode 290
getMetadata-Methode 291
Helfer (Registerkarte) 277, 278
Hohe Genauigkeit der Verarbeitung 270
Importe (Registerkarte) 276, 278
incrementErrorCount-Methode 291
invokeJExpression-Methode 292
isNull-Methode 293
Java-Code 273
Kompilieren 284
Kompilierungsfehler 285
logError-Methode 294
logInfo-Methode 294
Mapping-Ebene-Klassenpfad 273
Nanosekunden-Verarbeitung 270
Nicht-Benutzercode-Fehler 286
Ohne Status 270
Passiv 267
Primitive Java-Datentypen 267
Protokolle 294
Prüfen von Nullwerten in 293
resetNotification-Methode 295
setNull-Methode 295
Speichern von Metadaten 296
Standardportwerte 269
storeMetadata-Methode 296

Java-Umwandlungen *(Fortsetzung)*
Umwandlungsbereich 270
Zurücksetzen von Variablen in 295
JDK
Java-Umwandlung 266
JExpression-Klasse
Java-Ausdrücke 306, 307
JExprParaMetadata-Klasse
Java-Ausdrücke 305
Join-Bedingung
Übersicht 318
Joiner-Umwandlung
Ausdrucksparameter 321
Auswahlregeln 316
Bedingungen 318
Bedingungstyp 319, 320
Blockieren von Quell-Pipelines 330
Cachegröße 660
Caches 313
Caching von Master-Zeilen 331
Detail-Outer-Join 324
dynamische Mappings 315
dynamische Ports 321
Einfacher Bedingungstyp 319
Erweiterte Eigenschaften 312
Erweiterter Bedingungstyp 320
Full Outer Join 324
Join-Typen 322
Konfigurieren der Sortierreihenfolge 325
Leistung 331
Master Outer Join 323
mithilfe von Portauswahlen 320
Normales Join 322
Portauswahlen 315
Ports 314
Regeln und Richtlinien 332
Sortiert 331
Sortierte Eingabe 325
Übersicht 311
Unsortiert 331
Zusammenfügen von Daten aus derselben Quelle 328
JRE
Java-Umwandlung 266

K

Klassenpfad
Mapping-Eigenschaft 273
Klassifizierumwandlung
Klassifizieralgorithmus 140
Klassifiziermodell 140
Übersicht 139
Kompilieren
Java-Umwandlungen 284
Kompilieren von Java-Code
Full-Code (Registerkarte) 280
Kompilierungsfehler
Erkennen der Ursache in Java-Umwandlungen 286
Komponentenansicht
Datenprozessorumwandlung 198
Konfigurationsansicht
Ausnahmeumwandlung bei dupliziertem Datensatz 229
Ausnahmeumwandlung bei fehlerhaftem Datensatz 126
Ausnahmeumwandlung bei fehlerhaftem Datensatz - Beispiel 133
konfigurieren
Ports 37
Konsolidierungsumwandlung
Datensätze sortieren 154
erweiterte Eigenschaften 154
Tracingebenen 154
Übersicht 152
Konstanten
Ersetzen von Nullwerten mit 49

L

Laufzeit-Ereignisprotokoll
Datenprozessorumwandlung 211
Laufzeitansicht
SQL-Umwandlung 576
Leistung
Verwenden von Variablen zum Verbessern von 44
Lesedatenobjekt
parametrisieren 507
Leseumwandlung
Übersicht 501
Leseumwandlungen
Erstellen von relationalen Datenobjekten 509
Listenelemente
Beschreibung 644
Parsen von SOAP-Meldungen 630
logError-Methode
Java-Umwandlungen 294
logInfo-Methode
Java-Umwandlungen 294
Lokale Variablen
Übersicht 44
lookup
nicht zwischengespeichert 384
Lookup-Abfrage
Beschreibung 355
ORDER BY 355
Reservierte Wörter 357
Standardabfrage 355
Überschreiben 356, 358
Überschreiben von Richtlinien 357
Lookup-Bedingung
angeben nach Parameter 371
Beschreibung 360
Datenmaskierungsumwandlung 178
verwenden, generierte Ports 368
Lookup-Cache
Übersicht 362
Lookup-Caches
definition 382
dynamische 389
persistente 385
statische 384
Übersicht 382
Lookup-Quelle
parametrisieren 366
Lookup-Quellfilter
Begrenzen von Lookups 358
Lookup-SQL-Überschreibung
dynamische Caches 395
Lookup-Umwandlung
Abfrage-Eigenschaften 362
Cachegröße 660
Cachegröße, reduzieren 356
Cachen 362
Caches 382
dynamische Mappings 363
dynamische Ports 364

Lookup-Umwandlung *(Fortsetzung)*
dynamischer Cache 389
Entwickeln 355
Erstellen einer nicht wiederverwendbaren 377
Erstellen einer wiederverwendbaren 376
Erstellen von nicht verbundenen Lookups 378
Erweiterte Eigenschaften 375
Laufzeiteigenschaften 374
Lookup-Abfrage-Überschreibung 356
Lookup-Bedingung 359
Mapping-Parameter und -Variablen 356
Nicht verbunden 352, 354
nicht zwischengespeichertes Lookup 384
persistenter Cache 385
Regeln und Richtlinien für die Lookup-Bedingung 361
relationales parametrisiertes Datenobjekt 366
Standardabfrage 355
Überschreiben des Lookups 358
Übersicht 351
Verbunden 352, 354

M

manuelle Konsolidierung
Ausnahmeumwandlung bei dupliziertem Datensatz 229
Mapper
Beschreibung 197
Mapping-Parameter
In der Lookup-SQL-Überschreibung 356
Mapping-Variablen
In der Lookup-SQL-Überschreibung 356
Mappings
Kennzeichnen von Zeilen für Update 595
Mit Router-Umwandlungen 540
Mapplet
Referenz 202
Maskierung für besondere Formate
E-Mail-Adressen 187
IP-Adressen 188
Kreditkartennummern 186
Sozialversicherungsnummern 188, 189
Telefonnummern 188
URLs 189
Wiederholbaren SIN-Nummern 190
Maskierungsformat
Maskieren von Stringwerten 182
Spezielle Maskierungsformate 186
Maskierungsregeln
Bereich 184
Ersatzzeichen für Ergebniszeichenfolge 184
Maskierungsformat 182
Quellstringzeichen 183
Spezielle Maskierungsformate 186
Unschärfe 185
Maskierungstechniken
Datenmaskierung 170
Match-Analyse mit Doppelquelle 406
Match-Analyse mit Einzelquelle 406
Match-Umwandlung
Analyse mit Doppelquelle 406
Analyse mit Einzelquelle 406
Anzeigen von Cluster-Analysedaten 416
Anzeigen von Leistungsdaten 417
Ausgabeeigenschaften in der Ansicht „Match-Ausgabe“ 452
Ausgabeformate 450
Eigenschaften der Ansicht „Match-Ausgabe“ 434
Erstellen eines Mapplets für die Match-Analyse 427
Match-Umwandlung *(Fortsetzung)*
Konfigurieren eines Match-Analyse-Vorgangs 428
Konzepte in der Match-Analyse 405
Null-Match-Scores 410
Nutzungsfälle 404
Optionen für die Ansicht „Match-Ausgabe“ 434
Probe der Vergleichsstrategie 235
Prozessablauf für Feldanalyse 429
Prozessablauf für Identitätsanalyse 443
Treiber-Scores 411
Verknüpfungs-Scores 411
Vordefinierte Eingabeports 422
Max. Ausgabe des Zeilenzählers
SQL-Umwandlung 567
Maximale Ausgabe des Zeilenzählers
SQL-Umwandlung 569
Mehrfach vorkommende Felder
Normalizer-Umwandlung 467
Merge-Umwandlung
Übersicht 464
Methoden
Java-Umwandlungs-API 287
mit Editor synchronisieren
Datenprozessorumwandlung 216
Multi-Gruppe
Umwandlungen 38

N

Nanosekunden-Verarbeitung
Java-Umwandlungen 270
Nebeneffekte
SQL-Umwandlung 572
Web-Dienst-Verbraucher-Umwandlung 615
NEXTVAL-Port
Sequenzgenerator 545
Nicht verbundene Lookups
Beschreibung 352
Übersicht 354
Nicht verbundene Umwandlungen
Lookup-Umwandlung 354
Nicht wiederverwendbare Umwandlungen
Beschreibung 56
Nicht-Benutzercode-Fehler
In Java-Umwandlungen 286
Normalizer-Ansicht
Beschreibung 469
Normalizer-Umwandlung
Ausgabegruppen 472, 473
Bearbeiten einer Ausgabegruppe 475
Beispiel für Eingabe- und Ausgabegruppen 480
Beispiel für Mapping 479
Beispiel für Mapping-Ausgabe 481
Beispielnutzungsfall 478
Definitionsbeispiel 480
Eingabehierarchie 469
Eingabeports 470
erstellen 477
Erweiterte Eigenschaften 476
GCID 467
Mehrfach vorkommende Datensätze 468
Mehrfach vorkommende Felder 467
Schlüsselgenerierung 476
Teildatensätze 468
Übersicht 466
Ziehen von Ports aus anderen Objekten 478
Zusammenführen von Feldern 471, 472

Null-Match-Scores
Match-Umwandlung 410
Nullwerte
Einchecken in Java-Umwandlungen 293
Einstellung für die Java-Umwandlung 295
Ersetzen mit einer Konstante 49
Überspringen 50
Numerische Werte
Schlüsselmaskierung 175
Zufällige Maskierung 171
NumRowsAffected
Zeilenausgabe 569

O

Oberer Schwellenwert
konfigurieren 229
wird konfiguriert 126
Ohne Status
Java-Umwandlungen 270
operation
WSDL-Dateielement 599
Operation, Bereich
Web-Dienst-Umwandlungen 634
Operationsausgabe-Bereich
Anpassen der Web-Dienst-Verbraucher-Umwandlung 609
Option „Parameter"
Joiner-Umwandlung 321
Optionaler Fehlschlag - Ereignis
Datenprozessorumwandlung 209
ORDER BY
Lookup-Abfrage 355
Überschreiben 355

P

Parameter
Filterbedingung 256
Router-Umwandlung 539
Parameterbindung
SQL-Umwandlung 566
Parsen von SOAP-Meldungen
Abgeleitete Typen 628
Beschreibung 623
Denormalisierte Ausgabe 626
Normalisierte Ausgabe 625
Pivotierte Ausgabe 627
QName-Elemente 629
Union-Element 631
Parser
Beschreibung 197
Parserumwandlung
Übersicht 482
Pass-Through-Ports
Ausdrucksumwandlung 242
SQL-Umwandlung 561
Standardwerte 47
Passive Umwandlungen
Java 266, 267
Sequenzgenerator 542
Übersicht 31
Persistenter Lookup-Cache
Übersicht 385
Persistenter Speicher von Identitäts-Indexdaten 414
physische Datenobjekte
Synchronisierung 505
Pivotierte Ausgabe
Parsen von SOAP-Meldungen 627
Pivotierte Daten
SOAP-Meldungen 639
Port filtern
Web-Dienst-Verbraucher-Umwandlung 615
Portauswahl
Auswahlregeln 245, 246, 369, 370
Auswahlregeln 245, 246, 369, 370
erstellen 246, 317, 372
in dynamischen Ausdrücken 248
Joiner-Umwandlung 315, 316
Portauswahlen
In Joiner-Umwandlungen 320
Portlistenparameter
Registerkarte „Gruppieren nach" 111, 498
Ports
Ausnahmeumwandlung bei dupliziertem Datensatz 231
Ausnahmeumwandlung bei fehlerhaftem Datensatz 125
Bewertungsreihenfolge 46
Denormalisierte Web-Dienst-Eingabe 640
erstellen 37, 38
Java-Umwandlungen 269
Router-Umwandlung 540
Sequenzgeneratorumwandlung 544
Übersicht über Standardwerte 47
Variablenports 44
Zuordnen zu SOAP-Meldungen 637
Ports für Dienstparameter
Datenprozessorumwandlung 200
Ports zuweisen
Qualitätsprobleme 129
Ports, Ansicht
SQL-Umwandlungsausgabe 560
Portwerte
Java-Umwandlungen 269
Primärschlüssel
Erstellen mit einer Sequenzgeneratorumwandlung 542
Primitive Java-Datentypen
Java-Umwandlungen 267
Problemtabelle
wird erstellt 128
Protokoll
Definition 210
Protokoll, Entwicklungszeit-Ereignis
Beschreibung und Ort 210
Protokolle
Java-Umwandlungen 294
Pufferausgabeport
Datenprozessorumwandlung 200
Puffereingabeport
Datenprozessorumwandlung 199
Push-Into-Optimierung
In der SQL-Umwandlung aktivieren 573
SQL-Umwandlung 572
Web-Dienst-Verbraucher-Umwandlung 615

Q

QName-Elemente
Parsen von SOAP-Meldungen 629
Qualitätsprobleme
Zuweisen von Ports zu 129
Qualitätsproblemports
Beschreibung 125
leer statt NULL 121

Quelloptimierung
Einschränkungen 508
Quellstringzeichen
Datenmaskierungsumwandlung 183

R

Rang-Port
Rang-Umwandlung 497
Rang-Umwandlung
Definieren von Gruppen für 497
optionen 495
Ports 496
Rang-Port 497
RANKINDEX-Port 496
Rangumwandlung
Cachegröße 660
Caches 499
Dynamische Mappings 495
Erweiterte Eigenschaften 499
Übersicht 494
Verwenden von Variablen 44
Ranking
Gruppen von Daten 497
Stringwerte 495
Referenzen-Ansicht
Datenprozessorumwandlung 202
Regeln
Standardwerte 53
Registerkarte „Datenobjekt“
Beschreibung 506
Feldbeschreibungen 364
parametrisieren mithilfe eines neuen Datenobjekts 366
Registerkarte „Gruppieren nach”
Beschreibung 110
Portlistenparameter 111, 498
Registerkarte „Sortieren“
Sortierer-Umwandlung 551
Relational-in-hierarchisch-Umwandlung
Beispiel 512
Beschreibung 511
Entwicklung 515
erstellen 515
erstellen, Ports 516
Ports 515
relationale Datenobjekte
erstellen, eine dynamische Leseumwandlung 509
Repository-Ansicht
Datenprozessorumwandlung 198
Reservierte Wörter
Lookup-Abfrage 357
resetNotification-Methode
Java-Umwandlungen 295
REST-Webdienst-Verbraucher-Umwandlung
Anpassen der Ansicht „Ausgabe-Mapping“ 530
Antwortcode-Ports 526
Argumentports 525
Ausgabe-Mapping 529
Ausgabe-Mapping Regeln 530
Ausgabe-XML-Ports 526
Ausgabeports 524
Cookie-Ports 526
Delete-Methode 523
Eingabe-Mapping 527
Eingabe-Mapping-Regeln 527
Eingabeports 524
Einstellen der Basis-URL 531
REST-Webdienst-Verbraucher-Umwandlung *(Fortsetzung)*
Erstellen 533
Erweiterte Eigenschaften 531
Get-Methode 521
HTTP-Header-Ports 525
HTTP-Methoden 521
Internet-Medientyp 531
Konfiguration 519
Mapping-Ausgabe 517
Mapping-Ausgabeports 531
Mapping-Eingabe 517
Mapping-Eingabeports 528
Meldungskonfiguration 519
Nicht wiederverwendbar 532
Pass-Through-Ports 525
Ports 524
Post-Methode 522
Proxy-Server-Unterstützung 517
Prozess 519
Put-Methode 523
RequestInput-Port 524
Ressourcenkennung 520
Sicherheit 600
Sortierte Eingabe 531
Tracingebene 531
Transport Layer Security (TLS) 600
Übersicht 517
URL-Ports 525
Verbindungseigenschaft 531
Wiederverwendbar 532
XML-Schemavalidierung 531
Zuordnen von Elementen zu Ports 524
Router-Umwandlung
Beispiel 537
dynamische Mappings 535
dynamische Ports 539
Erweiterte Eigenschaften 541
Gruppen 536
Gruppenfilterbedingung 537
Parameter 539
Ports 540
Übersicht 534
Verbinden in Mappings 540

S

Schemata
Datenprozessorumwandlung 202
Relational-in-hierarchisch-Umwandlung 515
Umwandlung von hierarchisch in relational 263
Schlüssel
Erstellen mit einer Sequenzgeneratorumwandlung 542
SOAP-Meldungshierarchie 636
Schlüsselgeneratorumwandlung
erweiterte Eigenschaften 337
Tracingebenen 337
Übersicht 333
Schlüsselmaskierung
Beschreibung 174
Maskieren von Datetime-Werten 175
Maskieren von numerischen Werten 175
Maskieren von Stringwerten 174
Numerische Werte 174
Schreibumwandlung
Übersicht 649
Schwerwiegender Fehler - Ereignis
Datenprozessorumwandlung 209

Sequenz-Gruppe
Anzeigen in der REST-Webdienst-Verbraucher-Umwandlung 530
Sequenzdatenobjekt
Eigenschaften 546, 547
Sequenzgeneratorumwandlung
Eigenschaften 545, 546
Erstellen von zusammengesetzten Schlüsseln 542
NEXTVAL-Port 545
Nutzen der IIF-Funktion zum Ersetzen fehlender Schlüssel 543
Ports 544
Übersicht 542
Wertebereich 544
Zurücksetzen 546
Zyklus 544
service
WSDL-Dateielement 599
setNull-Methode
Java-Umwandlungen 295
SIN-Nummern
Maskieren von Sozialversicherungsnummern 189
Wiederholbare Datenmaskierung 190
Skript
In der Datenprozessorumwandlung erstellen 214
Skript-Hilfe-Ansicht
Datenprozessorumwandlung 198
SOAP-Aktion
Überschreiben in Web-Dienst-Verbraucher-Umwandlung 611
SOAP-Hierarchie
Beziehungen zu Eingabeports 635
SOAP-Komprimierung
Web-Dienst-Verbraucher-Umwandlung 614
SOAP-Meldung
Schlüssel 636
SOAP-Meldungen
Parsen von anyType-Elementen 627
Parsen von Auswahlelementen 630
Parsen von Listenelementen 630
Parsen von Substitutionsgruppen 630
Pivotieren von Daten 639
Übersicht 599
Zuordnen mehrerer Eingabeports 639
Zuordnen mehrfach vorkommender Knoten 625
Zuordnen von Auswahlelementen 643
Zuordnen von Listenelementen 644
Zuordnen von Ports 637
Zuordnen von Ports zu Union-Elementen 645
Sortierer-Umwandlung
Caches 555
dynamische Mappings 550
Registerkarte „Sortieren“ 551
Sortiererumwandlung
Cachegröße 660
Übersicht 549
Sortierschlüssel
konfigurieren 551
Sozialversicherungsnummern
Masking der Ortsvorwahl 189
Wiederholbare Datenmaskierung 189
Speicher verschlüsseln
Datenmaskierungsumwandlung 193
Speicher-Commit-Intervall
Datenmaskierungsumwandlung 193
Speicher-Verschlüsselungsschlüssel
Datenmaskierungsumwandlung 193
Speicherort, Spalte
Web-Dienst-Umwandlung 634
Speichertabelle
Ausdrucksmaskierung 173
Speichertabelle *(Fortsetzung)*
Substitutions-Datenmaskierung 177
SQL-Abfrage
SQL-Umwandlung 566
SQL-Eingabeports
SQL-Umwandlung 560
SQL-Umwandlung
INOUT-Parameter 578
"Early Selection"-Optimierung 572
Abfrage-Stringersetzung 567
Abfrageanweisung 576
Als leer erstellen 583
Anzahl der betroffenen Zeilen 563
Anzahl der Zeilenausgabe 569
Aufrufen einer gespeicherten Prozedur 578
Beispiel 573
Definieren der Abfrage 566
Definieren der Datenbankverbindung 576
Definieren von Ausgabeports 560
Eigenschaften der Push-Into-Optimierung 573
Eingabeports - Beschreibung 560
Einschränken von Ausgabezeilen 569
Ergebnissätze 579
Erstellen aus einer gespeicherten Prozedur 584
Erweiterte Eigenschaften (Ansicht) 563
Fortfahren bei SQL-Fehler 571
Gespeicherte Prozeduren 577
Kardinalität zwischen Eingabezeile und Ausgabezeile 567
Laufzeitverbindung 582
Parameter für gespeicherte Prozeduren 578
Parameterbindung 566
Pass-Through-Ports 561
Ports 559
Push-Into-Optimierung 572
Rückgabewert-Port 577
SQLError-Port 562
Übersicht 558
SQLError-Port
SQL-Umwandlung 562
Standardausgabegruppe
Ausnahmeumwandlung bei dupliziertem Datensatz 232
Standardausgabeport
Beschreibung 126
Standardgruppen
Router-Umwandlung 536
Standardisierungs-Umwandlung
Übersicht 586
Standardwerte
Ausgabeports 47, 48
Benutzerdefiniert 48
Datenmaskierung 190
Eingabeports 47, 48
Eingeben 51, 54
Pass-Through-Ports 47, 48
Regeln für 53
Übersicht 47
Validieren 51, 54
Starten der Ziffer
Sozialversicherungsnummern 190
Startkomponente
Datenprozessorumwandlung 201
Startwert (Eigenschaft)
Sequenzgeneratorumwandlung 545
Statische Variablen
Java-Umwandlungen 277
Statischer Code
Java-Umwandlungen 277

statischer Lookup-Cache
Übersicht 384
storeMetada-Methode
Java-Umwandlungen 296
Streamer
Beschreibung 197
Stringersetzung
SQL-Umwandlung 567
Strings
Ranking 495
Stringwerte
Benutzerdefinierte Datenmaskierung 170
Schlüsseldatenmaskierung 174
Substitutions-Maskierung
Beschreibung 176
Maskierungseigenschaften 177
Substitutionsgruppen
Parsen von SOAP-Meldungen 630
Web-Dienst-Verbraucher-Umwandlung 604, 607
Web-Dienste 643
Suchen
Fehler im Java-Code 285
Synchronisierung
benutzerdefinierte Datenobjekte 505
physische Datenobjekte 505

T

Teildatensätze
Normalizer-Umwandlung 468
Tracingebenen
Ausnahmeumwandlung bei dupliziertem Datensatz 233
Die Eigenschaft „Sequenzgeneratorumwandlung“ 545
Konsolidierungsumwandlung 154
Schlüsselgeneratorumwandlung 337
Zuordnungsumwandlung 119
Transformer
Beschreibung 197
Transport Layer Security (TLS)
REST-Webdienst-Verbraucher-Umwandlung 600
Web-Dienst-Verbraucher-Umwandlung 600
Treiber-Scores
Match-Umwandlung 411

U

Überschreiben der SOAP-Aktion
Web-Dienst-Verbraucher-Umwandlung 611
Überschreiben des Lookups
Richtlinien 357
Umwandlung von hierarchisch in relational
Beispiel 259
Beschreibung 259
Entwicklung 264
erstellen 264
erstellen, Ports 264
ports 262
Ports konfigurieren 263
testen im Daten-Viewer 265
Umwandlungen
active 30
Ausdrücke 39
Ausdrucksvalidierung 42
Bearbeiten von wiederverwendbaren 56
Behandlung von Fehlern 51
Cache-Partitionierung 665
Umwandlungen *(Fortsetzung)*
Cachedateien 661
Cachegröße 662, 664
Caches 660
Entwickeln 37
Erstellen 57
Java 266
Multi-Gruppe 38
Nicht verbunden 31
Nicht wiederverwendbar 56
Optimierung der Cachegröße 665
Sequenzgenerator 542
Übersicht 30
Verbunden 31
Wiederverwendbar 55, 56
Umwandlungsbereich
Java-Umwandlungen 270
Union-Element
Parsen von SOAP-Meldungen 631
Union-Elemente
Beschreibung 645
Union-Umwandlung
Übersicht 590
Unschärfe
Datumswerte 185
Numerische Werte 185
Unterer Schwellenwert
konfigurieren 229
wird konfiguriert 126
Update-Strategie-Umwandlung
Aggregator-Kombination 596
Erstellen 595
Erweiterte Eigenschaften 596
Konfigurationsschritte 594
Übersicht 594
Weiterleiten abgelehnter Zeilen 596

V

Validieren
Standardwerte 51, 54
Validierungsregelobjekt
In der Datenprozessorumwandlung erstellen 215
variable ports
Ausdrucksumwandlung 242
Variablen
Ergebnisse aus gespeicherten Prozeduren, erfassen 46
Initialisierungen 47
Java-Umwandlungen 277
Portbewertungsreihenfolge 46
Übersicht 44
Variablenports
Übersicht 44
Verbindung
REST-Webdienste 531
Web-Dienste 611
Verbundene Lookups
Beschreibung 352
Übersicht 354
Verbundene Umwandlungen
Java 266
Rang 494
Sequenzgenerator 542
Vergleichsstrategie
Ausnahmeumwandlung bei dupliziertem Datensatz 235
Vergleichsumwandlung
Cluster-Score-Optionen 410

Vergleichsumwandlung *(Fortsetzung)*
definierte Feldanalyse 406
definierte Identitätsanalyse 406
gruppierte Daten 407
Identitäts-Indextabellen 414
Leistungsfaktoren 415
Match-Eigenschaften in der Ansicht „Match-Ausgabe“ 452
Persistenzstatusbeschreibungen 423, 424
Persistenzstatuscodes 423, 424
Übersicht 147
Vordefinierte Ausgabeports 422
Verknüpfungs-Scores
Match-Umwandlung 411

W

Warnung - Ereignis
Datenprozessorumwandlung 209
Web-Dienst
Abgeleitete Typen 642
Substitutionsgruppen 643
Zuordnen von Ports zu anyTypes 642
Web-Dienst-Umwandlungen
Speicherort, Spalte 634
Web-Dienst-Verbindungen
Übersicht 611
Web-Dienst-Verbraucher-Umwandlung
"Early selection"-Optimierung 615
Aktivieren der generischen Fehlerausgabe 611
Aktivieren der HTTP-Fehlerausgabe 611
Anzeigen von Schlüsseln 605, 609
Ausgabe-Mapping 607
Cookie-Authentifizierung 603
Dynamische Web-Dienst-URL 603
Dynamischer WS-Security-Name 603
Eingabe-Mapping 604
Endpunkt-URL 603
Erstellen 617
Erweiterte Eigenschaften 611
Fehlerbehandlung 613
Filteroptimierung 615
Generische SOAP-Fehler 613
Gleichzeitige Web-Dienst-Anfragemeldungen 611
Hinzufügen von HTTP-Kopfzeilen 603
Mapping-Ausgabeknoten 607
Mapping-Eingabeports 604
Operationen 600
Push-Into-Optimierung 615
Push-Into-Optimierung aktivieren 616
Sicherheit 600
SOAP-Komprimierung 614
Transport Layer Security (TLS) 600
Übersicht 598
Webdienstbenutzer-Umwandlung
SOAP-Meldungen 599
Weiterleiten abgelehnter Zeilen
Konfigurieren 596
Option 596
Wiederverwendbar
Sequenzgeneratorumwandlung 546
Wiederverwendbare Umwandlungen
Bearbeiten 56
Beschreibung 55
Wörterbuch
Substitutions-Datenmaskierung 176
Wiederholbare Ausdrucksmaskierung 172
Wörterbuchinformationen
Datenmaskierungsumwandlung 178
WS-Security-Benutzername
Dynamischer Port 603
WSDL-Datei
Bindungselement 599
Dienstelement 599
Operationselement 599
Portelement 599

X

XMap-Objekt
In der Datenprozessorumwandlung erstellen 214

Z

Zeilen
Kennzeichnen für Update 595
Zufällige Maskierung
Maskieren von Datumswerten 171
Maskieren von Stringwerten 170
Numerische Werte 170
Zuordnungsumwandlung
erweiterte Eigenschaften 119
Tracingebenen 119
Übersicht 117
Zurücksetzen
Sequenzgeneratorumwandlung 546
Zurücksetzen (Eigenschaft)
Sequenzgeneratorumwandlung 545
Zusammengesetzte Schlüssel
Erstellen mit einer Sequenzgeneratorumwandlung 542
REST-Webdienst-Verbraucher-Umwandlung 527
Web-Dienst-Verbraucher-Umwandlung 604
Zyklus (Eigenschaft)
Die Eigenschaft „Sequenzgeneratorumwandlung“ 545