



Informatica®

9.6.1 HotFix 4

Informatica® PowerCenter Express

9.6.1 HotFix 2

Handbuch für Mapping

© Copyright Informatica LLC 1998, 2018

Diese Software und die zugehörige Dokumentation enthalten proprietäre Informationen der Informatica Corporation, 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 Corporation 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 und Informatica Master Data Management sind Marken oder eingetragene Marken der Informatica Corporation 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.

Dieses Produkt enthält Software, die von der Apache Software Foundation (<http://www.apache.org/>) entwickelt wurde, und/oder 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 GARANTIE ODER KONDITIONEN IRGEND EINER 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/hsqldbLicense.html>, <http://httpunit.sourceforge.net/doc/license.html>, <http://doc/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/license.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/license.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-snpmp.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>, und <http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>.

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 vom 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.

Die Software ist durch die amerikanischen Patentnummern 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,640,226; 6,789,096; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; 7,774,791; 8,065,266; 8,150,803; 8,166,048; 8,166,071; 8,200,622; 8,224,873; 8,271,477; 8,327,419; 8,386,435; 8,392,460; 8,453,159; 8,458,230; 8,707,336; 8,886,617 und RE44,478 geschützt. Internationale Patente und andere Patente sind angemeldet.

HAFTUNGSAUSSCHLUSS: Informatica Corporation 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 Corporation gewährleistet 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: 2018-06-25

Inhalt

Einleitung	8
Informatica-Ressourcen.	8
Informatica-Netzwerk.	8
Informatica-Wissensdatenbank.	9
Informatica-Dokumentation.	9
Informatica-Produktverfügbarkeitsmatrizen.	9
Informatica Velocity.	9
Informatica Marketplace.	9
Globaler Kundensupport von Informatica.	10
 Kapitel 1: Mappings.....	 11
Mappings - Übersicht.	11
Entwickeln eines Mappings.	12
Erstellen eines Mappings.	12
Mapping-Objekte.	13
Objektabhängigkeit in einem Mapping.	13
Hinzufügen von Objekten zu einem Mapping.	14
Verknüpfen von Ports.	14
Eins-zu-eins-Links.	15
Eins-zu-viele-Links.	15
Manuelles Verknüpfen von Ports.	16
Automatisches Verknüpfen von Ports.	16
Regeln und Richtlinien für das Verknüpfen von Ports.	18
Propagieren der Portattribute.	18
Abhängigkeitstypen.	19
Linkpfadabhängigkeiten.	19
Implizite Abhängigkeiten.	19
Durch Umwandlung propagierte Portattribute.	20
Erweiterte Eigenschaften des Mappings.	22
Laufzeiteigenschaften des Mappings.	22
Mapping-Validierung.	23
Verbindungsvalidierung.	24
Ausdrucksvalidierung.	24
Objektvalidierung.	24
Validieren eines Mappings.	24
Ausführen eines Mappings.	25
Segmente.	25
Kopieren eines Segments.	25

Kapitel 2: Mapplets.....	26
Mapplets - Übersicht.	26
Mapplet-Typen.	27
Mapplets und Regeln.	28
Mapplet-Eingabe und -Ausgabe.	28
Mapplet-Eingabe.	29
Mapplet-Ausgabe.	29
Erstellen eines Mapplets.	29
Validieren eines Mapplet.	30
 Kapitel 3: Mapping-Parameter und Parameterdateien.....	 31
Mapping-Parameter und Parameterdateien - Übersicht.	31
Systemparameter.	32
Benutzerdefinierte Parameter.	32
Prozess zum Ausführen von Mappings mit benutzerdefinierten Parametern.	33
Speicherort zum Erstellen von benutzerdefinierten Parametern.	33
Erstellen eines benutzerdefinierten Parameters.	34
Speicherort zum Zuweisen von Parametern.	35
Zuweisen eines Parameters.	36
Parameterdateien.	36
Parameterdateistruktur.	37
Projektelement.	37
Anwendungselement.	39
Regeln und Richtlinien für Parameterdateien.	40
Beispielparameterdatei.	41
Erstellen einer Parameterdatei.	44
Ausführen eines Mapping mit einer Parameterdatei.	45
 Kapitel 4: Export an PowerCenter.....	 46
Export an PowerCenter - Übersicht.	46
PowerCenter Release-Kompatibilität.	47
Einstellen des Kompatibilitätslevels.	47
Mapplet-Export.	47
Export an PowerCenter - Optionen.	48
Exportieren eines Objekts an PowerCenter.	49
Exporteinschränkungen.	50
Regeln und Richtlinien für das Exportieren an PowerCenter.	52
Fehlerbehebung - Exportieren an PowerCenter.	52
 Kapitel 5: Importieren aus PowerCenter.....	 54
Importieren aus PowerCenter - Übersicht.	54
Überschreibungseigenschaften.	55

Konfliktlösung.	55
Importübersicht.	55
Datentypkonvertierung.	56
Umwandlungskonvertierung.	56
Einschränkungen für Umwandlungseigenschaften.	57
Verbindungsparameter für das PowerCenter-Repository.	62
Importieren eines Objekts aus PowerCenter.	62
Importieren von Einschränkungen.	64
Importieren von Leistung.	64
Kapitel 6: Leistungsoptimierung.	65
Leistungsoptimierung - Übersicht.	65
Optimierungsebenen.	66
Optimierungsmethoden - Übersicht.	66
Frühe Projektionsoptimierungsmethode.	67
Frühe Auswahloptimierungsmethode.	67
Methode zur Optimierung der Verzweigungsbereinigung.	67
Vorhersageoptimierungsmethode.	68
Kostenbasierte Optimierungsmethode.	68
Semi-Join Optimierungsmethode.	69
Vollständige Optimierung und Speicherzuordnung.	69
Einstellen der Optimierungsebene für ein Developer Tool-Mapping.	70
Einstellen der Optimierungsebene für ein bereitgestelltes Mapping.	70
Kapitel 7: Pushdown-Optimierung.	71
Die Pushdown-Optimierung - Übersicht.	71
Umwandlungslogik.	72
Pushdown-Optimierung zu Quellen.	72
Pushdown-Optimierung für relationale Quellen.	73
Pushdown-Optimierung für native Quellen.	75
Pushdown-Optimierung für nichtrelationale PowerExchange-Quellen.	75
Pushdown-Optimierung für ODBC-Quellen.	75
Pushdown-Optimierung für SAP-Quellen.	76
Pushdown-Optimierungsausdrücke.	77
Funktionen.	77
Operatoren.	90
Vergleichen der Ausgabe vom Data Integration Service und von Quellen.	91
Kapitel 8: Partitionierte Mappings.	93
Partitionierte Mappings – Übersicht.	93
Ein Thread für jede Pipeline-Stage.	94
Mehrere Threads für jede Pipeline-Stage.	95
Berechnungen für den Parallelismuswert.	96

„Auto“ als maximaler Parallelismus für das Mapping.	97
Ganzzahlwert als maximaler Parallelismus für das Mapping.	98
Partitionierte Einfachdateiquellen.	99
Partitionierung für gleichzeitiges Lesen.	99
Partitionierte relationale Quellen.	100
Relationale Verbindungstypen für Partitionierung.	101
SQL-Abfragen für partitionierte relationale Quellen.	101
Regeln und Richtlinien für Partitionen der relationalen Quelle.	102
Partitionierte Einfachdateiziele.	103
Zusammenführungsoptionen für partitionierte Dateiziele.	104
Befehle für partitionierte Dateiziele.	104
Partitionierte relationale Ziele.	105
Relationale Verbindungstypen für Partitionierung.	106
Regeln und Richtlinien für Partitionen des relationalen Ziels.	107
Partitionierte Umwandlungen.	107
Einschränkungen für partitionierte Umwandlungen.	108
Cache-Partitionierung für Umwandlungen.	108
Umwandlungen, die numerische Funktionen verwenden.	109
Umwandlungen, die lokale Variablen verwenden.	109
Adress-Validierer- und Match-Umwandlungen.	110
Mappings, die Quellzeilen begrenzen.	110
Deaktivieren der Partitionierung für ein Mapping.	111
Deaktivieren der Partitionierung.	111
Überschreiben des maximalen Parallelismus für ein Mapping.	112
Überschreiben des maximalen Parallelismuswerts.	113
Index.	114

Einleitung

Das *Informatica-Handbuch für Mapping* enthält Informationen zum Erstellen und Verwenden von Mappings, zum Exportieren und Importieren von Objekten und zum Aktivieren der Leistungsoptimierung und Pushdown-Optimierung. Das *Informatica-Handbuch für Mapping* richtet sich an Entwickler von Mappings und Mapplets, wie zum Beispiel Datendienste, sowie an Datenumwandlungsentwickler. Das Handbuch setzt voraus, dass Ihnen die Konzepte von Einfachdateien und relationalen Datenbanken sowie die Datenbank-Engines in Ihrer Umgebung bekannt sind.

Das *Informatica PowerCenter Express-Handbuch für Mapping* enthält Informationen zum Erstellen und Verwenden von Mappings und Mapplets. Das *Informatica PowerCenter Express-Handbuch für Mapping* enthält auch Informationen zum Aktivieren der Leistungsoptimierung und Pushdown-Optimierung. Das *Informatica PowerCenter Express-Handbuch für Mapping* richtet sich an Entwickler von Datenintegration. Dieses Handbuch setzt Kenntnisse über Einfachdatei- und relationale Datenbankkonzepte, die Datenbankmodule in Ihrer Umgebung und Datenintegrationskonzepte voraus.

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.

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.

- 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 <https://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

Mappings

Dieses Kapitel umfasst die folgenden Themen:

- [Mappings - Übersicht, 11](#)
- [Entwickeln eines Mappings, 12](#)
- [Erstellen eines Mappings, 12](#)
- [Mapping-Objekte, 13](#)
- [Verknüpfen von Ports, 14](#)
- [Propagieren der Portattribute, 18](#)
- [Erweiterte Eigenschaften des Mappings, 22](#)
- [Laufzeiteigenschaften des Mappings, 22](#)
- [Mapping-Validierung, 23](#)
- [Ausführen eines Mappings, 25](#)
- [Segmente, 25](#)

Mappings - Übersicht

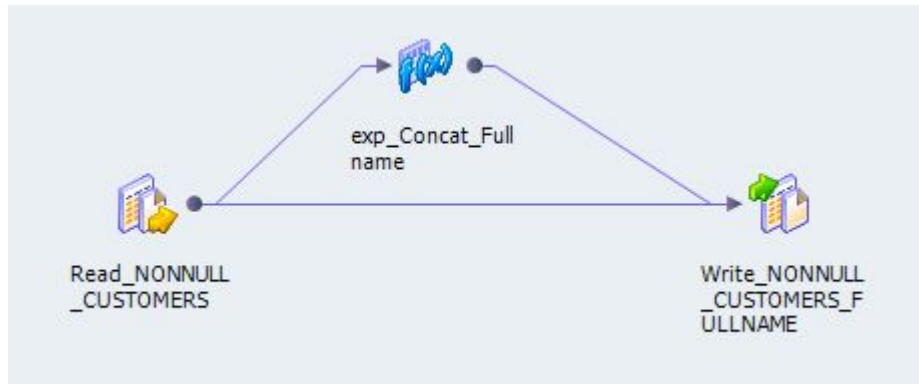
Ein Mapping ist ein Satz von Eingaben und Ausgaben, die den Datenfluss zwischen Quellen und Zielen darstellen. Diese können durch Umwandlungsobjekte verknüpft sein, die die Regeln für die Datenumwandlung definieren. Der Data Integration Service verwendet die im Mapping konfigurierten Anweisungen, um Daten zu lesen, umzuwandeln und zu schreiben.

Der in ein Mapping einbezogene Eingabe- und Ausgabebetyp legt den Mapping-Typ fest. Sie können im Developer Tool die folgenden Mapping-Typen erstellen:

- Mapping mit physischen Datenobjekten als Ein- und Ausgabe
- logisches Datenobjekt-Mapping mit einem logischen Datenobjekt als Mapping-Eingabe oder -Ausgabe
- Operations-Mapping mit einem Vorgang als Mapping-Eingabe, Ausgabe oder beides
- virtuelles Tabellen-Mapping mit virtueller Tabelle als Mapping-Ausgabe

Sie können ein Mapping mit physischen Datenobjekten als Ein- und Ausgabe in eine Mapping-Aufgabe in einem Arbeitsablauf einfügen. Möglicherweise möchten Sie ein Mapping von einem Arbeitsablauf ausführen, sodass Sie mehrere Mappings sequenziell ausführen können. Oder Sie können einen Arbeitsablauf entwickeln, der Befehle ausführt, um Schritte vor und nach der Ausführung eines Mappings durchzuführen.

Die folgende Abbildung zeigt ein Beispiel für ein Mapping:



Entwickeln eines Mappings

Entwickeln Sie ein Mapping, um Daten entsprechend der Geschäftsanforderungen zu lesen, umzuwandeln und zu schreiben.

1. Bestimmen Sie den Mappingtyp, den Sie erstellen möchten.
2. Erstellen Sie die Eingabe, die Ausgabe und wiederverwendbare Objekte, die Sie im Mapping verwenden möchten. Erstellen Sie physische Datenobjekte, logische Datenobjekte oder virtuelle Tabellen, um sie als Mapping-Eingabe oder -Ausgabe zu verwenden. Erstellen Sie physische Datenobjekte oder logische Datenobjekte, um sie als Mapping-Eingabe oder -Ausgabe zu verwenden. Erstellen Sie wiederverwendbare Umwandlungen, die Sie verwenden möchten. Wenn Sie Mäpplets verwenden möchten, müssen Sie auch diese erstellen.
3. Erstellen Sie das Mapping.
4. Fügen Sie Objekte zum Mapping hinzu. Sie müssen Eingabe- und Ausgabe-Objekte zum Mapping hinzufügen. Fügen Sie optional Umwandlungen und Mäpplets hinzu.
5. Verknüpfen Sie die Ports zwischen Mapping-Objekten, um einen Datenfluss von Quellen zu Zielen mittels Mäpplets und Umwandlungen für das Hinzufügen, Entfernen oder Ändern von Daten entlang dieses Flusses zu erstellen.
6. Validieren Sie das Mapping, um Fehler zu ermitteln.
7. Speichern Sie das Mapping im Modellrepository.

Wenn Sie ein Mapping entwickelt haben, führen Sie es aus, um die Mapping-Ausgabe anzuzeigen.

Erstellen eines Mappings

Erstellen Sie ein Mapping zum Verschieben von Daten zwischen Quellen und Zielen und zum Umwandeln der Daten.

1. Wählen Sie in der Ansicht **Objekt-Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei > Neu > Mapping**.

3. Geben Sie einen Mapping-Namen ein.
4. Klicken Sie auf **Fertig stellen**.
Ein leeres Mapping wird im Editor angezeigt.

Mapping-Objekte

Mapping-Objekte legen den Datenfluss zwischen Quellen und Zielen fest.

Jedes Mapping muss die folgenden Objekte enthalten:

- Eingabe. Beschreibt die Merkmale der Mapping-Quelle.
- Ausgabe. Beschreibt die Merkmale des Mapping-Ziels.

Ein Mapping kann auch die folgenden Komponenten enthalten:

- Umwandlung. Ändert Daten, bevor sie in Ziele geschrieben werden. Verwenden Sie verschiedene Umwandlungsobjekte zur Ausführung unterschiedlicher Funktionen.
- Mapplet. Ein wiederverwendbares Objekt, das einen Satz von Umwandlungen enthält, die Sie in mehreren Mappings verwenden können.

Wenn Sie ein Objekt zu einem Mapping hinzufügen, konfigurieren Sie die Eigenschaften so, dass der Data Integration Service die gewünschten Änderungen an den Daten vornimmt. Sie verbinden auch die Mapping-Objekte so, dass der Data Integration Service die Daten wie gewünscht verschiebt. Sie verbinden Objekte über Ports.

Der Editor zeigt Objekte wie folgt an:

- Symbolisiert. Zeigt ein Symbol des Objekts mit dem Objektnamen.
- Normal. Zeigt die Spalten und die Eingabe- und Ausgabeport-Indikatoren. Sie können Objekte in der Normalansicht miteinander verbinden.

Objektabhängigkeit in einem Mapping

Ein Mapping ist von einigen Objekten abhängig, die als unabhängige Objekte in einem Repository gespeichert sind.

Wenn sich Objekt-Metadaten ändern, verfolgt das Developer Tool die Auswirkungen dieser Änderungen in Mappings. Mappings können auch ungültig werden, wenn Sie sie nicht bearbeitet haben. Der Data Integration Service kann ungültig gewordene Mappings nicht ausführen.

Die folgenden Objekte sind als unabhängige Objekte in einem Repository gespeichert:

- logische Datenobjekte
- physische Datenobjekte
- wiederverwendbare Umwandlungen
- Mapplets

Ein Mapping ist von diesen Objekten abhängig.

Die folgenden Objekte in einem Mapping sind als abhängige Repository-Objekte gespeichert:

- Virtuelle Tabellen. Virtuelle Tabellen sind als Teil eines SQL-Datendienstes gespeichert.
- Nicht wiederverwendbare Umwandlungen, die Sie im Mapping erstellen. Nicht wiederverwendbare Umwandlungen sind nur im Mapping gespeichert.

Nicht wiederverwendbare Umwandlungen, die Sie innerhalb des Mappings erstellt haben, werden als abhängige Repositoryobjekte innerhalb des Mapping gespeichert.

Hinzufügen von Objekten zu einem Mapping

Fügen Sie Objekte zu einem Mapping hinzu, um den Datenfluss zwischen Quellen und Zielen festzulegen.

1. Öffnen Sie das Mapping.
2. Ziehen Sie ein physisches Datenobjekt in den Editor und wählen Sie „Lesen“, um das Datenobjekt als Quelle hinzuzufügen.
3. Ziehen Sie ein physisches Datenobjekt in den Editor und wählen Sie „Schreiben“, um das Datenobjekt als Ziel hinzuzufügen.
4. Um eine Lookup-Umwandlung hinzuzufügen, ziehen Sie ein Einfachdatei-Datenobjekt, logisches Datenobjekt, eine Referenztabelle oder ein relationales Datenobjekt in den Editor und wählen Sie „Lookup“ aus. Um eine Lookup-Umwandlung hinzuzufügen, ziehen Sie ein Einfachdatei-Datenobjekt, logisches Datenobjekt, eine Referenztabelle oder ein relationales Datenobjekt in den Editor und wählen Sie „Lookup“ aus.
5. Um eine wiederverwendbare Umwandlung hinzuzufügen, ziehen Sie die Umwandlung aus dem Umwandlungsordner in die Ansicht **Objekt-Explorer** des Editors.
Wiederholen Sie diesen Schritt für jede wiederverwendbare Umwandlung, die Sie hinzufügen möchten.
6. Wählen Sie zum Hinzufügen einer nicht wiederverwendbaren Umwandlung die Umwandlung aus der **Umwandlungspalette** aus und ziehen Sie sie in den Editor.
Wiederholen Sie diesen Schritt für jede nicht wiederverwendbare Umwandlung, die Sie hinzufügen möchten.
7. Konfigurieren Sie die Ports und Eigenschaften für jede nicht wiederverwendbare Umwandlung.
8. Optional können Sie ein Mapplet in den Editor ziehen.

Verknüpfen von Ports

Wenn Sie Eingabe-, Ausgabe-, Umwandlungs- und Mapplet-Objekte in einem Mapping hinzugefügt haben, stellen Sie das Mapping durch das Verknüpfen von Ports zwischen Mapping-Objekten fertig.

Die Daten kommen über die folgenden Ports in eine Umwandlung und aus einer Umwandlung:

- Eingabeports. Daten empfangen.
- Ausgabeports. Daten weitergeben.
- Eingabe-/Ausgabeports. Daten empfangen und unverändert weitergeben.

Jedes Eingabeobjekt, Ausgabeobjekt, Mapplet und jede Umwandlung enthält eine Sammlung von Ports. Jeder Port repräsentiert eine Datenspalte:

- Eingabeobjekte stellen Daten bereit, sie enthalten daher nur Ausgabeports.
- Ausgabeobjekte empfangen Daten, sie enthalten daher nur Eingabeports.
- Mapplets enthalten nur Eingabe- und Ausgabeports.
- Umwandlungen enthalten eine Mischung aus Eingabe-, Ausgabe- und Eingabe-/Ausgabeports, je nach der Umwandlung und ihrer Anwendung.

Um Ports miteinander zu verbinden, erstellen Sie eine Verknüpfung zwischen Ports in verschiedenen Mapping-Objekten. Das Developer Tool stellt die Verbindung nur her, wenn die Verbindung die Anforderungen für die Validierung und Verkettung von Verknüpfungen erfüllt.

Sie können Ports unverbunden lassen. Der Data Integration Service ignoriert unverbundene Ports.

Wenn Sie Ports zwischen Eingabeobjekten, Umwandlungen, Mapplets und Ausgabeobjekten verknüpfen, können Sie die folgenden Typen von Verknüpfungen erstellen:

- Ein zu eins
- Eins zu viele

Sie können Ports manuell oder automatisch miteinander verknüpfen.

Eins-zu-eins-Links

Verknüpfen Sie einen Port in einem Eingabeobjekt oder einer Umwandlung in einen Port in einem Ausgabeobjekt oder einer Umwandlung.

Eins-zu-viele-Links

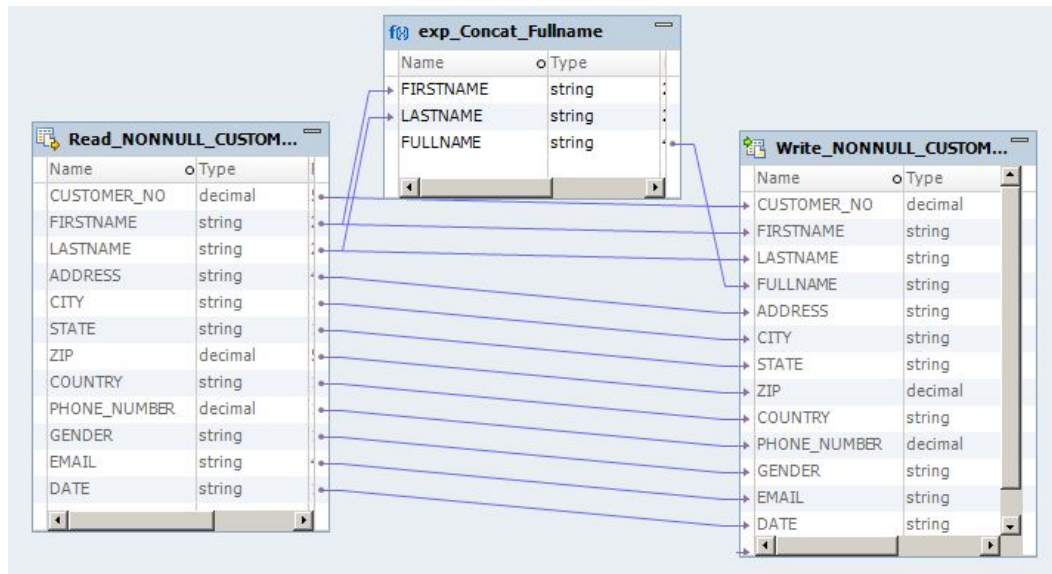
Wenn Sie dieselben Daten für unterschiedliche Zwecke verwenden möchten, können Sie den Port, der diese Daten bereitstellt, mit mehreren Ports im Mapping verknüpfen.

Sie können eine Eins-zu-viele-Verknüpfung wie folgt erstellen:

- Verknüpfen Sie einen Port mit mehreren Umwandlungen oder Ausgabeobjekten.
- Verknüpfen Sie mehrere Ports in einer Umwandlung mit mehreren Umwandlungen oder Ausgabeobjekten.

Beispiel: Sie möchten zur Berechnung des Durchschnittsgehalts in einer Bank-Zweigstelle Gehaltsinformationen über die Aggregator-Umwandlung verwenden. Sie können dieselben Informationen in einer Ausdrucksumwandlung verwenden, die zur Berechnung des Monatsgehalts jedes Mitarbeiters konfiguriert ist.

Die folgende Abbildung zeigt ein Beispiel für ein Mapping mit Eins-zu-viele-Verknüpfungen:



Manuelles Verknüpfen von Ports

Sie können manuell einen Port oder mehrere Ports verknüpfen.

Ziehen Sie einen Port aus einem Eingabeobjekt oder einer Umwandlung in einen Port in einem Ausgabeobjekt oder einer Umwandlung.

Verwenden Sie die Strg- oder Umschalt-Taste, um mehrere Ports auszuwählen, die mit einer Umwandlung oder einem Ausgabeobjekt verknüpft werden sollen. Das Developer Tool verknüpft die Ports, angefangen vom obersten Paar. Es verknüpft alle Ports, die die Validierungsanforderungen erfüllen.

Wenn Sie einen Port in einen leeren Port ziehen, kopiert das Developer-Tool den Port und erstellt eine Verknüpfung.

Automatisches Verknüpfen von Ports

Wenn Sie Ports automatisch verknüpfen, können Sie nach Position oder nach Namen verknüpfen.

Wenn Sie Ports automatisch nach Namen verknüpfen, können Sie ein Präfix oder Suffix angeben, über das die Ports verknüpft werden sollen. Verwenden Sie Präfixe und Suffixe, um die Position der Ports in einem Mapping anzuzeigen.

Verknüpfen von Ports nach Namen

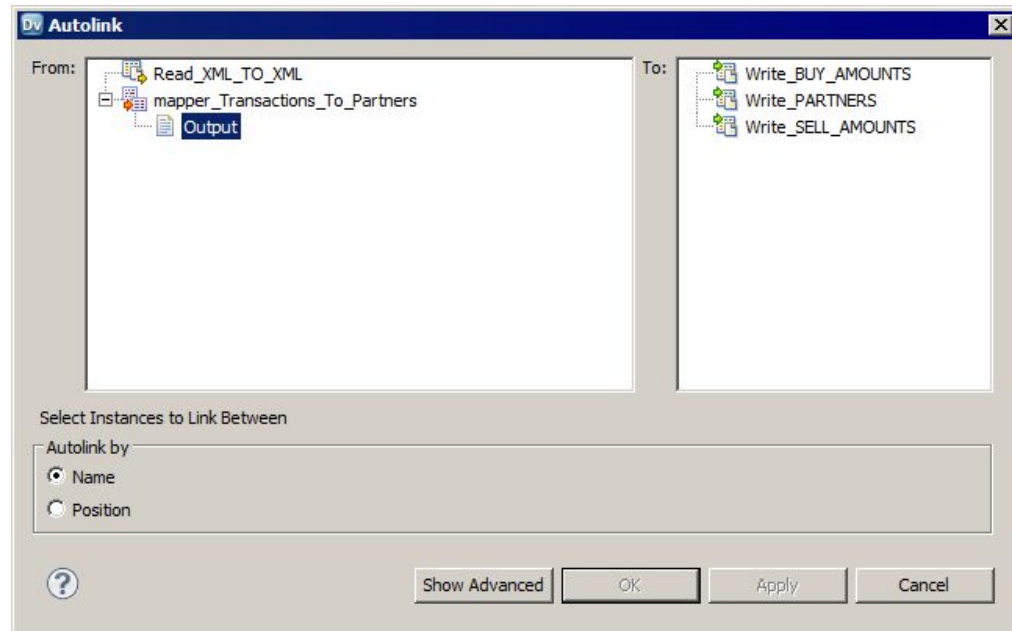
Wenn Sie Ports nach Namen verknüpfen, fügt das Developer Tool Verknüpfungen zwischen Eingabe- und Ausgabeports mit demselben Namen ein. Verknüpfen Sie nach Namen, wenn Sie in Umwandlungen dieselben Portnamen verwenden.

Sie können Ports auf Basis von Präfixen und Suffixen verknüpfen, die Sie definieren. Verwenden Sie Präfixe und Suffixe, um die Position der Ports in einem Mapping anzuzeigen. Verknüpfen Sie nach Namen und Präfix oder Suffix, wenn Sie Präfixe oder Suffixe in Portnamen verwenden, um deren Stelle im Mapping oder Mapplet zu unterscheiden.

Bei der Verknüpfung nach Namen wird nicht zwischen Groß- und Kleinschreibung unterschieden.

1. Klicken Sie auf **Mapping > Auto-Link**.

Das **Auto-Link** Dialogfeld wird geöffnet.



2. Wählen Sie ein Objekt im Fenster **from** aus, von dem aus die Verknüpfung erstellt werden soll.
3. Wählen Sie ein Objekt aus dem Fenster **to** aus, zu dem die Verknüpfung erstellt werden soll.
4. Wählen Sie **Name**.
5. Klicken Sie optional auf **Erweiterte Einstellungen anzeigen**, um Ports basierend auf Präfixen und Suffixen zu verknüpfen.
6. Klicken Sie auf **OK**.

Verknüpfen von Ports nach Position

Wenn Sie nach Position verknüpfen, verknüpft das Developer-Tool jeden Ausgabeport mit dem entsprechenden Eingabeport. Beispielsweise ist der erste Ausgabeport mit dem erste Eingabeport verknüpft, der zweite Ausgabeport mit dem zweiten Eingabeport. Verknüpfen Sie nach Position, wenn Sie Umwandlungen mit zugehörigen Ports in derselben Reihenfolge erstellen.

1. Klicken Sie auf **Mapping > Auto-Link**.
Das **Auto-Link** Dialogfeld wird geöffnet.
2. Wählen Sie ein Objekt im Fenster **from** aus, von dem aus die Verknüpfung erstellt werden soll.
3. Wählen Sie ein Objekt im Fenster **to** aus, zu dem die Verknüpfung erstellt werden soll.
4. Wählen Sie **Position**, und klicken Sie auf **OK**.

Das Developer-Tool verlinkt jeden Ausgabeport mit dem entsprechenden Eingabeport. Beispielsweise ist der erste Ausgabeport mit dem erste Eingabeport verknüpft, der zweite Ausgabeport mit dem zweiten Eingabeport.

Regeln und Richtlinien für das Verknüpfen von Ports

Beim Verknüpfen von Ports werden bestimmte Regeln und Richtlinien angewandt.

Halten Sie sich an die folgenden Regeln und Richtlinien, wenn Sie Mapping-Objekte verbinden:

- Entdeckt das Developer Tool beim Versuch, Ports zwischen zwei Mapping-Objekten zu verknüpfen, einen Fehler, zeigt es ein Symbol an, das darauf hinweist, dass Sie die Ports nicht miteinander verknüpfen können.
- Folgen Sie im Mapping der Logik des Datenflusses. Sie können die folgenden Porttypen verknüpfen:
 - Der empfangende Port muss ein Eingabe- oder Eingabe-/Ausgabeport sein.
 - Der Ursprungsport muss ein Ausgabe- oder Eingabe-/Ausgabeport sein.
 - Sie können nicht Eingabeports mit Eingabeports oder Ausgabeports mit Ausgabeports verknüpfen.
- Sie müssen mindestens einen Port einer Eingabegruppe mit einer vorangehenden Umwandlung verknüpfen.
- Sie müssen mindestens einen Port einer Ausgabegruppe mit einer nachgeschalteten Umwandlung verknüpfen.
- Sie können Ports von einer aktiven Umwandlung oder einer Ausgabegruppe einer aktiven Umwandlung mit einer Eingabegruppe einer anderen Umwandlung verknüpfen.
- Sie können nicht eine aktive Umwandlung und eine passive Umwandlung in derselben nachgeschalteten Umwandlung oder Umwandlungseingabegruppe verbinden.
- Sie können nicht mehr als eine aktive Umwandlung mit derselben nachgeschalteten Umwandlung oder Umwandlungseingabegruppe verbinden.
- Sie können eine beliebige Anzahl passiver Umwandlungen mit derselben nachgeschalteten Umwandlung, Umwandlungseingabegruppe oder demselben Ziel verbinden.
- Sie können Ports aus zwei Ausgabegruppen in derselben Umwandlung mit einer für sortierte Daten konfigurierten Joiner-Umwandlung verknüpfen, wenn die Daten beider Ausgabegruppen sortiert sind.
- Sie können nur Ports mit kompatiblen Datentypen miteinander verknüpfen. Das Developer Tool überprüft, bevor es die Verknüpfung herstellt, ob es die beiden Datentypen einander zuordnen kann. Der Data Integration Service kann Daten nicht zwischen Ports mit inkompatiblen Datentypen umwandeln.
- Das Developer-Tool markiert einige Mappings als ungültig, wenn das Mapping gegen die Datenflussvalidierung verstößt.

Propagieren der Portattribute

Propagieren Sie Portattribute, um geänderte Attribute während der Ausführungsdauer eines Mappings an einen Port zu übergeben.

1. Wählen Sie im Editor einen Port in einer Umwandlung.
2. Klicken Sie auf **Mapping > Attribute propagieren**.
Es wird das Dialogfeld **Attribute propagieren** angezeigt.
3. Wählen Sie eine Richtung, in die die Attribute propagiert werden sollen.
4. Wählen Sie die Attribute aus, die Sie propagieren möchten.
5. Zeigen Sie optional die Ergebnisse in der Vorschau an.

6. Klicken Sie auf **Anwenden**.

Das Developer Tool propagiert die Portattribute.

Abhängigkeitstypen

Wenn Sie Portattribute propagieren, aktualisiert das Developer Tool die Abhängigkeiten.

Das Developer Tool kann die folgenden Eigenschaften aktualisieren:

- Linkpfadabhängigkeiten
- Implizite Abhängigkeiten

Linkpfadabhängigkeiten

Eine Linkpfadabhängigkeit ist eine Abhängigkeit zwischen einem propagierten Port und den Ports in dessen Linkpfad.

Wenn Sie Abhängigkeiten in einem Linkpfad propagieren, aktualisiert das Developer Tool alle Eingabe- und Eingabe-/Ausgabeports in dessen Vorwärts-Linkpfad und alle Ausgabe- und Eingabe-/Ausgabeports in dessen Rückwärts-Linkpfad. Das Developer Tool führt die folgenden Aktualisierungen durch:

- Es aktualisiert den Portnamen, den Datentyp, die Genauigkeit, die Dezimalstellenanzahl und die Beschreibung für alle Ports im Linkpfad des propagierten Ports.
- Es aktualisiert alle Ausdrücke oder Bedingungen, die den propagierten Port mit dem geänderten Portnamen referenzieren.
- Es aktualisiert die zugehörige Port-Eigenschaft in einer dynamischen Lookup-Umwandlung, wenn sich der zugehörige Portname ändert.

Implizite Abhängigkeiten

Eine implizite Abhängigkeit ist eine Abhängigkeit innerhalb einer Umwandlung zwischen zwei Ports, die auf einem Ausdruck oder einer Bedingung basieren.

Sie können Datentyp, Genauigkeit, Dezimalstellenanzahl und Beschreibung zu Ports mit impliziten Abhängigkeiten propagieren. Sie können auch Bedingungen und Ausdrücke parsen, um die impliziten Abhängigkeiten des propagierten Ports zu identifizieren. Alle Ports mit impliziten Abhängigkeiten sind Ausgabe- oder Eingabe-/Ausgabeports.

Wenn Sie Bedingungen einschließen, aktualisiert das Developer Tool die folgenden Eigenschaften:

- Linkpfadabhängigkeiten
- Ausgabeports, die in derselben Lookup-Bedingung wie der propagierte Port sind
- Verbundene Ports in dynamischen Lookup-Umwandlungen, die mit dem propagierten Port verbunden sind
- Masterports, die in derselben Join-Bedingung wie der Detail-Port sind

Wenn Sie Ausdrücke einschließen, aktualisiert das Developer Tool die folgenden Eigenschaften:

- Linkpfadabhängigkeiten
- Ausgabeports, die einen Ausdruck mit dem propagierten Port enthalten

Das Developer Tool propagiert nicht an implizite Abhängigkeiten innerhalb derselben Umwandlung. Sie müssen die geänderten Attribute aus einer anderen Umwandlung propagieren. Beispiel: Wenn Sie den Datentyp des Ports ändern, der in einer Lookup-Bedingung verwendet wird, und diese Veränderung aus der Lookup-Bedingung propagieren, propagiert das Developer Tool die Änderung nicht an den anderen von der Bedingung abhängigen Port in derselben Lookup-Umwandlung.

Durch Umwandlung propagierte Portattribute

Das Developer Tool propagiert Abhängigkeiten und Attribute für jede Umwandlung.

Die folgende Tabelle beschreibt die Abhängigkeiten und Attribute, die das Developer Tool für jede Umwandlung propagiert.

Umwandlung	Abhängigkeit	Propagierte Attribute
Adressvalidator	Keine.	Keine. Diese Umwandlung hat vordefinierte Portnamen und Datentypen.
Aggregator	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad - Ausdruck - Implizite Abhängigkeiten 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
Zuordnung	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Fallkonverter	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Klassifizierer	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Vergleich	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Konsolidator	Keine.	Keine. Diese Umwandlung hat vordefinierte Portnamen und Datentypen.
Datenmaskierung	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Datenprozessor	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Entscheidung	<ul style="list-style-type: none"> - Nachfolgende Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Ausdruck	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad - Ausdruck - Implizite Abhängigkeiten 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
Filter	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad - Bedingung 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname
Joiner	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad - Bedingung - Implizite Abhängigkeiten 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
Schlüssel-Generator	<ul style="list-style-type: none"> - Ports im Verknüpfungspfad 	<ul style="list-style-type: none"> - Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung

Umwandlung	Abhängigkeit	Propagierte Attribute
Labeler	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Suche	- Ports im Verknüpfungspfad - Bedingung - Zugehörige Ports (dynamisches Lookup) - Implizite Abhängigkeiten	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
Übereinstimmung	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Zusammenführung	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Normalisierer	- Ports im Verknüpfungspfad	- Portname
Parser	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Rang	- Ports im Verknüpfungspfad - Ausdruck - Implizite Abhängigkeiten	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
REST-Webdienst-Verbraucher	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Router	- Ports im Verknüpfungspfad - Bedingung	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname
Sequenzgenerator	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Sortierer	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
SQL	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Standardisierer	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung
Union	- Ports im Verknüpfungspfad - Implizite Abhängigkeiten	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Datentyp, Genauigkeit, Dezimalstellenanzahl
Update-Strategie	- Ports im Verknüpfungspfad - Ausdruck - Implizite Abhängigkeiten	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung - Portname - Datentyp, Genauigkeit, Dezimalstellenanzahl
Gewichteter Durchschnitt	- Ports im Verknüpfungspfad	- Portname, Datentyp, Genauigkeit, Dezimalstellenanzahl, Beschreibung

Erweiterte Eigenschaften des Mappings

Die erweiterten Mapping-Eigenschaften definieren die Validierungsumgebung für das Mapping und können Eingabe- und Ausgabeeinstellungen für Umwandlungen definieren.

Sie können die native Validierungsumgebung in PowerCenter Express verwenden.

Konfigurieren Sie die folgende erweiterte Mapping-Eigenschaft:

Validierungsumgebung

Gibt an, ob die Mapping-Definition für die native Laufzeitumgebung, die Hive-Laufzeitumgebung oder für beide vom Developer-Tool validiert wird. Beim Ausführen eines Mappings in der nativen Umgebung, verarbeitet der Datenintegrationsdienst das Mapping. Wenn Sie PowerCenter Big Data Edition installiert haben, können Sie ein Mapping in der Hive-Umgebung ausführen. Beim Ausführen eines Mappings in der Hive-Umgebung verschiebt der Datenintegrationsdienst die Mapping-Ausführung über eine Hive-Verbindung in den Hadoop-Cluster. Der Hadoop-Cluster verarbeitet das Mapping.

Wählen Sie beide Validierungsumgebungen aus, wenn Sie das Mapping in der nativen Umgebung testen möchten, bevor Sie das Mappings in der Hive-Umgebung ausführen. Sie können alternativ beide Validierungsumgebungen auswählen, wenn Sie während der Mapping-Ausführung den Wert der Laufzeitumgebung in einer Parameterdatei definieren möchten.

Wenn Sie beide Umgebungen auswählen, müssen Sie die Laufzeitumgebung für das Mapping in den Laufzeiteigenschaften angeben.

Standardwert ist "Nativ".

Laufzeiteigenschaften des Mappings

Die Laufzeiteigenschaften eines Mappings richten sich nach der für das Mapping ausgewählten Laufzeitumgebung.

Konfigurieren Sie die folgenden Laufzeiteigenschaften für ein Mapping:

Laufzeitumgebung

Wählen Sie die bei der Ausführung des Mappings zu verwendende Laufzeitumgebung aus. Beim Ausführen eines Mappings in der nativen Umgebung, verarbeitet der Datenintegrationsdienst das Mapping. Wenn Sie PowerCenter Big Data Edition installiert haben, können Sie ein Mapping in der Hive-Umgebung ausführen. Beim Ausführen eines Mappings in der Hive-Umgebung verschiebt der Datenintegrationsdienst die Mapping-Ausführung über eine Hive-Verbindung in den Hadoop-Cluster. Der Hadoop-Cluster verarbeitet das Mapping.

Wenn Sie die Laufzeitumgebung auswählen, speichert das Developer-Tool eine der zugeordneten Validierungsumgebungen für die Mapping-Ausführung.

Sie können die native Laufzeitumgebung in PowerCenter Express verwenden.

Maximaler Parallelismus

Gültig für die native Laufzeitumgebung. Maximale Anzahl paralleler Threads, die eine einzelne Mapping-Pipeline-Stage verarbeiten. Zum Aktivieren der Partitionierung von Mappings können Administratoren das Administrator-Tool verwenden, um maximalen Parallelismus für den Datenintegrationsdienst-Prozess auf einen Wert größer als 1 zu setzen.

Standardmäßig ist der maximale Parallelismus für ein Mapping auf „Auto“ gesetzt. Der Datenintegrationsdienst berechnet den tatsächlichen Parallelismuswert basierend auf dem für den Datenintegrationsdienst-Prozess festgelegten maximalen Parallelismus und auf der maximalen Anzahl der Partitionen für alle Quellen im Mapping.

Sie können Partitionierung für das Mapping deaktivieren, indem Sie den Wert auf 1 setzen. Sie können den Standardwert auch überschreiben, um die Anzahl der Threads zu definieren, die der Datenintegrationsdienst erstellt. Wenn der maximale Parallelismus für den Datenintegrationsdienst-Prozess und das Mapping auf andere Ganzzahlwerte festgelegt ist, verwendet der Datenintegrationsdienst den minimalen Wert für den tatsächlichen Parallelismuswert.

PowerCenter Express unterstützt diese Option nicht. Wenn Sie einen Wert für maximalen Parallelismus eingeben, wird dieser vom Datenintegrationsdienst ignoriert.

Standardwert ist "Auto".

Weitere Informationen darüber, wie der Datenintegrationsdienst den Parallelismuswert berechnet, finden Sie unter ["Berechnungen für den Parallelismuswert" auf Seite 96](#).

Benutzername für den Identitätswechsel bei Zuordnungen

Ermöglicht einem anderen Benutzer, den Datenintegrationsdienst-Benutzer zu imitieren, der eine Verbindung zu Hive-, HBase- oder HDFS-Quellen und -Zielen herstellt, die Kerberos-Authentifizierung verwenden.

Geben Sie einen Benutzernamen in folgendem Format ein: <Hadoop-Dienstname>/<Hostname>@<YOUR-REALM>

Wobei:

- Hadoop-Dienstname der Name des Hadoop-Diensts ist, in dem sich die Hive-, HBase- oder HDFS-Quellen oder -Ziele befinden.
- Hostname der Name oder die IP-Adresse des Hadoop-Diensts ist.
- YOUR-REALM der Kerberos-Bereich ist.

Die folgenden Sonderzeichen können nur als Trennzeichen verwendet werden: '/' und '@'

Hive-Verbindung

Gültig für die Hive-Laufzeitumgebung. Eine Hive-Verbindung definiert die Verbindungsinformationen, die vom Datenintegrationsdienst zum Verschieben der Mapping-Ausführung in den Hadoop-Cluster benötigt werden. Wählen Sie die Hive-Verbindung aus, um das Mapping im Hadoop-Cluster zu validieren oder auszuführen.

Mapping-Validierung

Wenn Sie ein Mapping entwickeln, müssen Sie es so konfigurieren, dass der Data Integration Service das gesamte Mapping lesen und verarbeiten kann. Das Developer-Tool markiert einen Arbeitsablauf als nicht gültig, wenn es Fehler entdeckt, die den Data Integration Service daran hindern, das Mapping auszuführen.

Das Developer Tool betrachtet folgende Validierungs-Typen:

- Verbindung
- Ausdruck
- Objekt

- Datenfluss

Verbindungsvalidierung

Das Developer Tool führt jedes Mal eine Verbindungsvalidierung durch, wenn Sie Ports in einem Mapping verbinden, sowie jedes Mal, wenn Sie ein Mapping validieren.

Wenn Sie Ports verbinden, überprüft das Developer Tool, ob Sie gültige Verbindungen herstellen. Wenn Sie ein Mapping validieren, überprüft das Developer Tool, ob die Verbindungen gültig sind und alle erforderlichen Ports verbunden sind. Das Developer Tool führt die folgenden Verbindungsvalidierungen durch:

- Mindestens ein Eingabeobjekt und ein Ausgabeobjekt sind miteinander verbunden.
- Mindestens ein Mapplet-Eingabeport- und -Ausgabeport ist mit dem Mapping verbunden.
- Die Datentypen zwischen den Ports sind kompatibel. Wenn Sie einen Datentyp in einen anderen ändern, der mit dem Port, mit dem er verbunden ist, inkompatibel ist, generiert das Developer Tool einen Fehler und wird das Mapping ungültig. Sie können jedoch den Datentyp ändern, wenn er mit den verbundenen Ports kompatibel bleibt, etwa Char und Varchar.

Ausdrucksvalidierung

Sie können einen Ausdruck in einer Umwandlung während des Entwickelns eines Mappings validieren. Wenn Sie die Fehler nicht korrigiert haben, werden beim Validieren des Mappings in der Ansicht

Validationsprotokoll Fehlermeldungen angezeigt.

Wenn Sie in einem Ausdruck verwendete Eingabeports löschen, markiert das Developer-Tool das Mapping als ungültig.

Objektvalidierung

Wenn Sie in Mapping validieren, überprüft das Developer Tool, ob die Definitionen der unabhängigen Objekte, z. B. Eingabe-Umwandlungen oder Mapplets, mit der Instanz im Mapping übereinstimmen.

Änderungen an einem Objekt während des Konfigurierens eines Mappings können zu Fehlern im Mapping führen. Bei Änderungen an einem Objekt während des Konfigurierens eines Mappings verfolgt das Developer Tool die Auswirkungen dieser Änderungen in Mappings.

Validieren eines Mappings

Validieren Sie ein Mapping, um sicherzustellen, dass der Data Integration Service das gesamte Mapping lesen und verarbeiten kann.

1. Klicken Sie auf **Bearbeiten > Validieren**.
In der Ansicht **Validierungsprotokoll** werden Fehler angezeigt.
2. Beheben Sie die Fehler und validieren Sie das Mapping erneut.

Ausführen eines Mappings

Führen Sie ein Mapping aus, um eine Ausgabe von Quellen zu Zielen zu verschieben und Daten umzuwandeln.

Wenn Sie keinen Standard-Data Integration Service ausgewählt haben, werden Sie vom Developer Tool aufgefordert, eines auszuwählen.

- Klicken Sie mit der rechten Maustaste in einen leeren Bereich im Editor, und klicken Sie auf **Mapping ausführen**.

Der Data Integration Service führt das Mapping aus und das Mapping schreibt die Ausgabe in das Ziel.

Segmente

Ein Segment besteht aus einem oder mehreren Objekten in einem Mapping, Mapplet, einer Regel oder einer virtuellen gespeicherten Prozedur. Ein Segment kann eine Quelle, ein Ziel, eine Umwandlung oder ein Mapplet enthalten. Ein Segment besteht aus einem oder mehreren Objekten in einem Mapping, Mapplet oder einer Regel. Ein Segment kann eine Quelle, ein Ziel, eine Umwandlung oder ein Mapplet enthalten.

Sie können Segmente kopieren. Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie ein Segment kopieren:

- Sie können Segmente ordner- oder projektübergreifend kopieren.
- Das Developer Tool verwendet wo möglich Abhängigkeiten erneut. Andernfalls kopiert es Abhängigkeiten.
- Wenn ein Mapping, Mapplet, eine Regel oder virtuelle gespeicherte Prozedur Parameter enthält und Sie eine Umwandlung kopieren, die sich auf den Parameter bezieht, verwendet die Umwandlung im Zielobjekt einen Standardwert für den Parameter.
- Wenn ein Mapping, Mapplet oder eine Regel Parameter enthalten und Sie eine Umwandlung kopieren, die sich auf den Parameter bezieht, verwendet die Umwandlung im Zielobjekt einen Standardwert für den Parameter.
- Sie können keine Eingabeumwandlungen oder Ausgabeumwandlungen kopieren.
- Wenn Sie ein Segment einfügen, können Sie vorhergegangene Aktionen nicht mehr rückgängig machen.

Kopieren eines Segments

Sie können ein Segment kopieren, wenn Sie einen Teil der Mapping-Logik in einem anderen Mapping, Mapplet, einer anderen Regel oder einer virtuell gespeicherten Prozedur wiederverwenden möchten. Sie können ein Segment kopieren, wenn Sie einen Teil der Mapping-Logik in einem anderen Mapping, Mapplet oder einer anderen Regel wiederverwenden möchten.

1. Öffnen Sie das Objekt, das die Segment enthält, die Sie kopieren möchten.
2. Wählen Sie ein Segment aus, indem Sie jedes Objekt, das Sie kopieren möchten, markieren.
Halten Sie die Strg-Taste gedrückt, um mehrere Objekte auszuwählen. Sie können Segmente auch auswählen, indem Sie mit dem Pointer ein Rechteck um die Objekte im Editor ziehen.
3. Klicken Sie zum Kopieren des Segments in die Zwischenablage auf **Bearbeiten > Kopieren**.
4. Öffnen Sie ein Ziel-Mapping, -Mapplet, eine Zielregel oder eine virtuell gespeicherte Zielprozedur.
5. Öffnen Sie ein Ziel-Mapping, -Mapplet oder eine Zielregel.
6. Klicken Sie auf **Bearbeiten > Einfügen**.

KAPITEL 2

Mapplets

Dieses Kapitel umfasst die folgenden Themen:

- [Mapplets - Übersicht, 26](#)
- [Mapplet-Typen, 27](#)
- [Mapplets und Regeln, 28](#)
- [Mapplet-Eingabe und -Ausgabe, 28](#)
- [Erstellen eines Mapplets, 29](#)
- [Validieren eines Mapplet, 30](#)

Mapplets - Übersicht

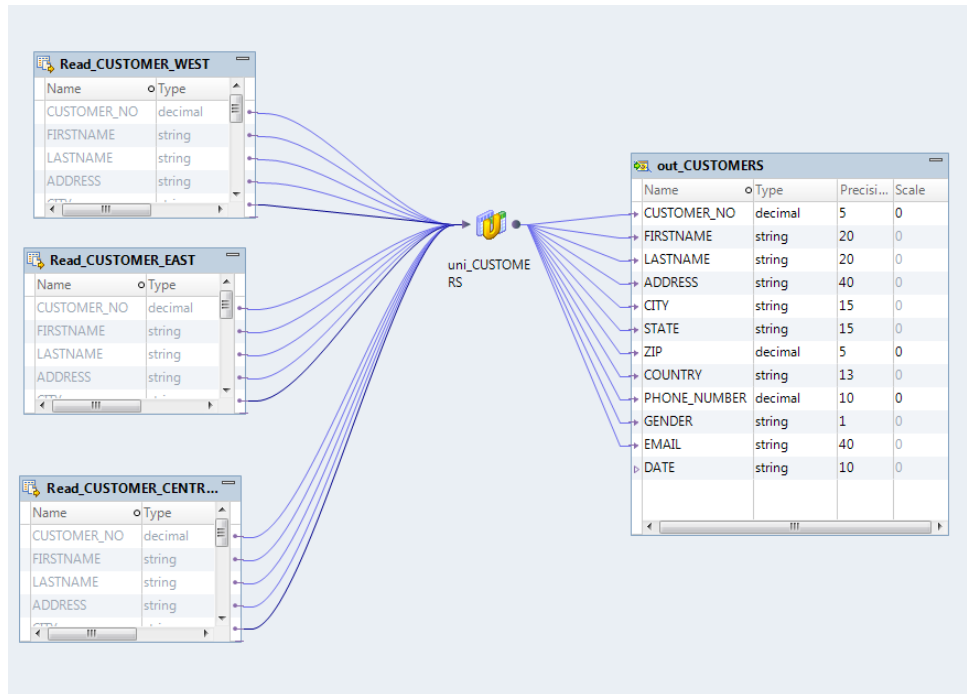
Ein Mapplet ist ein wiederverwendbares Objekt, das einen Satz von Umwandlungen enthält, die Sie in mehreren Mappings verwenden können. Verwenden Sie ein Mapplet in einem Mapping. Oder validieren Sie das Mapplet als Regel.

Umwandlungen in einem Mapplet können wiederverwendbar oder nicht wiederverwendbar sein. Wenn Sie einem Mapplet eine Sequenzgeneratorumwandlung hinzufügen, muss das Mapplet wiederverwendbar sein.

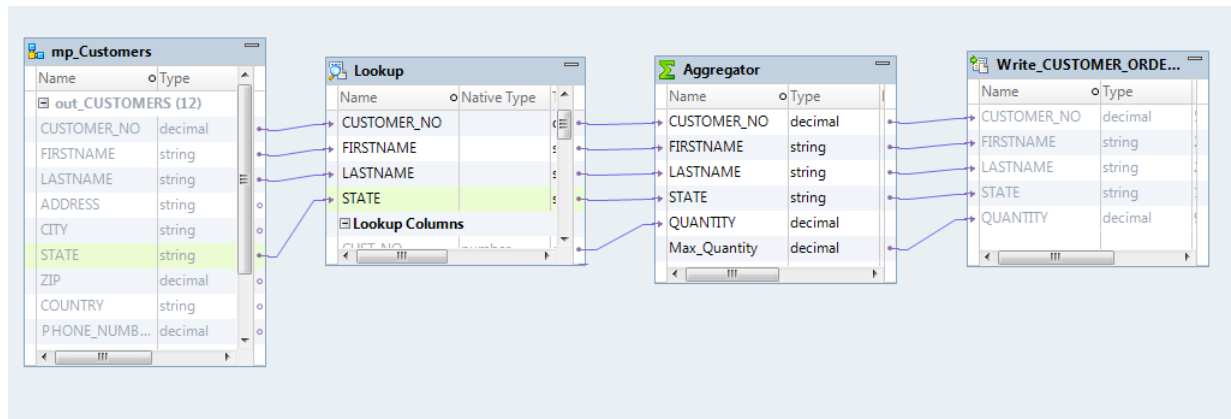
Wenn Sie ein Mapplet in einem Mapping verwenden, verwenden Sie eine Instanz des Mapplets. Jede Änderung am Mapplet wird durch alle Instanzen des Mapplets geerbt.

Ein Mapplet kann andere Mapplets enthalten. Sie können ein Mapplet in einem Mapping oder Mapplet mehrmals verwenden. Zirkuläre Verschachtelungen von Mapplets sind nicht möglich. Beispiel: Wenn Mapplet A Mapplet B enthält, kann Mapplet B nicht Mapplet A enthalten.

Beispielsweise zeigt die folgende Abbildung ein Mapplet, das Kundeninformationen aus östlichen, westlichen und zentralen Regionen mit Hilfe von einer Union-Umwandlung kombiniert:



Die folgende Abbildung zeigt ein Mapping, wo Sie das Benutzer-Mapplet erneut verwenden und die Kunden finden können, die in den einzelnen Bundesstaaten die meisten Artikel gekauft haben:



Mapplet-Typen

Der Mapplet-Typ wird durch die Mapplet-Eingabe und -Ausgabe festgelegt.

Sie können die folgenden Mapplet-Typen erstellen:

- Quelle. Das Mapplet enthält eine Datenquelle als Eingabe und eine Ausgabeumwandlung als Ausgabe.
- Ziel. Das Mapplet enthält eine Eingabeumwandlung als Eingabe und eine Datenquelle als Ausgabe.

- Midstream. Das Mapplet enthält eine Eingabeumwandlung und eine Ausgabeumwandlung. Es enthält keine Datenquelle für Eingabe oder Ausgabe.

Mapplets und Regeln

Eine Regel ist eine Geschäftslogik, die Bedingungen definiert, welche auf Quelldaten angewendet werden, wenn ein Profil ausgeführt wird. Es handelt sich um ein Midstream-Mapplet, das in einem Profil verwendet wird.

Eine Regel muss folgende Anforderungen erfüllen:

- Sie muss eine Eingabe- und eine Ausgabe-Umwandlung enthalten. In einer Regel können Sie keine Datenquellen verwenden.
- Sie kann Expressions-Umwandlungen, Lookup-Umwandlungen und passive Datenqualitäts-Umwandlungen enthalten. Andere Umwandlungstypen darf sie nicht enthalten. So darf eine Regel beispielsweise keine Match-Umwandlung enthalten, da es sich hierbei um eine aktive Umwandlung handelt.

Es kann Ausdrucks- und Lookup-Umwandlungen enthalten. Andere Umwandlungstypen darf sie nicht enthalten.

- Sie gibt keine Kardinalität zwischen Eingabegruppen an.

Hinweis: Die Regelfunktionalität ist nicht auf Profiling begrenzt. Sie können alle Mapplets, die Sie validieren, als Regel zu einem Profil im Analyst-Tool hinzufügen. Zum Beispiel können Sie die Datenqualität einer Postadresse untersuchen, indem Sie eine Regel auswählen, die für die Validierung von Postadressen und deren Hinzufügung zu einem Profil konfiguriert wurde.

Mapplet-Eingabe und -Ausgabe

Um ein Mapplet in einem Mapping verwenden zu können, müssen Sie es für Eingabe und Ausgabe konfigurieren.

Ein Mapplet hat die folgenden Eingabe- und Ausgabekomponenten:

- Mapplet-Eingabe. Sie können Daten von einer Datenquelle oder Eingabeumwandlung oder beidem in ein Mapplet weitergeben. Wenn Sie ein Mapplet als Regel validieren, müssen Sie Daten durch eine Eingabeumwandlung in das Mapplet weitergeben. Wenn Sie eine Eingabeumwandlung verwenden, verbinden Sie sie mit einer Quelle oder vorangehenden Umwandlung im Mapping.
- Mapplet-Ausgabe. Sie können Daten aus einem Mapplet von Datenquellen oder Ausgabeumwandlungen oder beides weitergeben. Wenn Sie ein Mapplet als Regel validieren, müssen Sie Daten durch eine Ausgabeumwandlung aus dem Mapplet weitergeben. Wenn Sie eine Ausgabeumwandlung verwenden, verbinden Sie sie mit einem Ziel oder einer nachgeschalteten Umwandlung im Mapping.
- Mapplet-Ports. Sie können Mapplet-Ports im Mapping-Editor anzeigen. Mapplet-Eingabeports und -Ausgabeports stammen aus Eingabeumwandlungen und Ausgabeumwandlungen. Sie stammen nicht aus Datenquellen.

Mapplet-Eingabe

Die Mapplet-Eingabe kann aus einer Datenquelle oder einer Eingabeumwandlung stammen.

Sie können mehrere Pipelines in einem Mapplet erstellen. Verwenden Sie mehrere Datenquellen oder Eingabeumwandlungen. Sie können auch eine Kombination aus Datenquellen und Eingabeumwandlungen verwenden.

Verwenden Sie eine oder mehrere Datenquellen, um Quelldaten im Mapplet bereitzustellen. Wenn Sie das Mapplet in einem Mapping verwenden, so ist es das erste Objekt in der Mapping-Pipeline und enthält keine Eingabeports.

Verwenden Sie eine Eingabeumwandlung, um Eingaben vom Mapping zu empfangen. Die Eingabeumwandlung stellt Eingabeports bereit, sodass Sie Daten durch das Mapplet weitergeben können. Jeder Port in der Eingabeumwandlung, der mit einer anderen Umwandlung im Mapplet verbunden ist, wird ein Mapplet-Eingabeport. Eingabeumwandlungen können Daten von einer einzelnen aktiven Quelle empfangen. Unverbundene Ports erscheinen nicht im Mapping-Editor.

Sie können eine Eingabeumwandlung mit mehreren Umwandlungen in einem Mapplet verbinden. Sie können auch einen Port in einer Eingabeumwandlung mit mehreren Umwandlungen in einem Mapplet verbinden.

Mapplet-Ausgabe

Verwenden Sie eine Datenquelle, wenn Sie ein Ziel-Mapplet erstellen möchten. Verwenden Sie eine Ausgabeumwandlung in einem Mapplet, um Daten über ein Mapplet in ein Mapping weiterzugeben.

Verwenden Sie eine oder mehrere Datenquellen, um Zieldaten im Mapplet bereitzustellen. Wenn Sie das Mapplet in einem Mapping verwenden, so ist es das letzte Objekt in der Mapping-Pipeline und enthält keine Ausgabeports.

Verwenden Sie eine Ausgabeumwandlung, um Ausgaben an eine nachgeschaltete Umwandlung oder ein nachgeschaltetes Ziel in einem Mapping weiterzugeben. Verbundene Ports in einer Ausgabeumwandlung werden als Mapplet-Ausgabeports in einem Mapping angezeigt. Ausgabeumwandlungen in einem Mapplet werden als Ausgabegruppen angezeigt. Eine Ausgabegruppe kann Daten an mehrere Pipelines in einem Mapping weitergeben.

Erstellen eines Mapplets

Erstellen Sie ein Mapplet, um ein wiederverwendbares Objekt zu definieren, das einen Satz von Umwandlungen enthält, die Sie in mehreren Mappings verwenden können.

1. Wählen Sie in der Ansicht **Object Explorer** ein Projekt oder einen Ordner aus.
2. Klicken Sie auf **Datei > Neu > Mapplet**.
3. Geben Sie einen Mapplet-Namen ein.
4. Klicken Sie auf **Fertigstellen**.
Es wird ein leeres Mapplet im Editor angezeigt.
5. Fügen Sie Mapplet-Eingaben, -Ausgaben und Umwandlungen ein.

Validieren eines Mapplet

Bevor Sie ein Mapplet zu einem Mapping hinzufügen, müssen Sie es validieren. Sie können ein Mapplet auch als Regel validieren, um es in ein Profil aufzunehmen.

1. Klicken Sie mit der rechten Maustaste auf den Mapplet-Editor.
2. Wählen Sie **Validieren als > Mapplet** oder **Validieren als > Regel**.
Im Validierungs-Log werden Mapplet-Fehlermeldungen angezeigt.

KAPITEL 3

Mapping-Parameter und Parameterdateien

Dieses Kapitel umfasst die folgenden Themen:

- [Mapping-Parameter und Parameterdateien - Übersicht, 31](#)
- [Systemparameter, 32](#)
- [Benutzerdefinierte Parameter, 32](#)
- [Speicherort zum Zuweisen von Parametern, 35](#)
- [Parameterdateien, 36](#)

Mapping-Parameter und Parameterdateien - Übersicht

Ein Mapping-Parameter stellt einen konstanten Wert dar, der sich während Mapping-Ausführungen ändern kann, wie zum Beispiel Verbindungen und Verzeichnisse für Quelldateien und Cachedateien.

Sie können System- oder benutzerdefinierte Parameter beim Ausführen eines Mapping verwenden. Systemparameter definieren die Verzeichnisse, in denen der Data Integration Service Cachedateien, Ablehnungsdateien, Quelldateien, Zieldateien und temporäre Dateien speichert. Sie definieren die Werte der Systemparameter für einen Data Integration Service-Prozess im Administrator Tool.

Mit benutzerdefinierten Parametern können Sie Mapping-Werte in einer Parameterdatei definieren und diese Werte bei jeder Mapping-Ausführung aktualisieren. Erstellen Sie benutzerdefinierte Mapping-Parameter, damit Sie ein Mapping mit unterschiedlichen Werten für die Verbindung, Einfachdatei, Cachedatei, temporäre Datei oder Referenztabelle erneut ausführen können. Erstellen Sie benutzerdefinierte Parameter, damit Sie ein Mapping mit unterschiedlichen Werten für die Verbindung, Einfachdatei, Cache-Datei oder temporäre Datei erneut ausführen können. Sie definieren die Parameterwerte in einer Parameterdatei. Wenn Sie das Mapping über die Befehlszeile ausführen und eine Parameterdatei angeben, verwendet der Data Integration Service die in der Parameterdatei definierten Parameterwerte.

Sie können benutzerdefinierte Arbeitsablaufparameter erstellen, wenn Sie einen Arbeitsablauf entwickeln. Ein Arbeitsablaufparameter ist ein konstanter Wert, der sich zwischen Arbeitsablaufausführungen ändern kann.

Hinweis: Wenn Sie ein Mapping als einen SQL-Datendienst bereitstellen und das Mapping system- oder benutzerdefinierte Parameter verwendet, verwendet der Datenintegrationsdienst die Parameterstandardwerte für den SQL-Datendienst.

Systemparameter

Systemparameter sind konstante Werte für die Definition der Verzeichnisse, in denen Data Integration Service Cachedateien, Ablehnungsdateien, Quelldateien, Zieldateien und temporäre Dateien speichert.

Sie definieren die Werte der Systemparameter für einen Data Integration Service-Prozess im Administrator-Tool. Sie können System-Parameter-Werte nicht in einer Parameterdatei definieren oder überschreiben.

Sie können keine Systemparameter erstellen. Das Developer Tool bietet eine vordefinierte Liste mit System-Mapping-Parametern, die Sie einem Datenobjekt oder einer Umwandlung in einem zuweisen können. Standardmäßig werden die Systemparameter Verzeichnisfeldern für Einfachdateien, Cachedateien und temporären Dateien zugewiesen. Wenn Sie zum Beispiel eine Aggregator-Umwandlung erstellen, stellt der Systemparameter für das Cache-Verzeichnis den Standardwert dar, der dem Cache-Verzeichnisfeld zugewiesen ist.

Die folgende Tabelle beschreibt die Systemparameter:

Systemparameter	Typ	Beschreibung
CacheDir	String	Standardverzeichnis für Index- und Datencachedateien.
RejectDir	String	Standardverzeichnis für Ablehnungsdateien.
SourceDir	String	Standardverzeichnis für Quelldateien.
TargetDir	String	Standardverzeichnis für Zieldateien.
TempDir	String	Standardverzeichnis für temporäre Dateien.

Benutzerdefinierte Parameter

Benutzerdefinierte Parameter stellen Werte dar, die sich zwischen Mapping-Ausführungen ändern. Sie können benutzerdefinierte Parameter erstellen, die Verbindungen, bigint-Werte, Ganzzahlenwerte oder Zeichenfolgenwerte darstellen.

Erstellen Sie Parameter, damit Sie ein Mapping mit unterschiedlichen Werten erneut ausführen können. Sie erstellen beispielsweise eine Zuordnung, die Kundenbestellungen verarbeitet. Die Zuordnung liest Kundeninformationen aus einer relationalen Tabelle, die Kundendaten für ein Land enthält. Sie möchten die Zuordnung für Kunden in den Vereinigten Staaten, Kanada und Mexiko verwenden. Erstellen Sie einen benutzerdefinierten Parameter, der die Verbindung zur Kundentabelle darstellt. Erstellen Sie drei Parameterdateien, die den Verbindungsnamen auf die jeweilige Kundentabelle für die Vereinigten Staaten, Kanada und Mexiko festlegen. Führen Sie das Mapping über die Befehlszeile aus und verwenden Sie für jede Mapping-Ausführung eine andere Parameterdatei.

Sie können die folgenden Typen von benutzerdefinierten Parametern erstellen:

- **Verbindung.** Stellt eine Datenbankverbindung dar. Sie können keine Verbindungsparameter für Unternehmensanwendungen oder Verbindungen zu sozialen Medien erstellen.
- **Bigint.** Stellt einen langen Wert dar.

- String. Stellt Folgendes dar: Einfachdateinamen, Einfachdatei-Verzeichnis, Cache-Dateiverzeichnis, temporäres Dateiverzeichnis, Referenztabellenname, Referenztabellen-Verzeichnis oder Typ der Mapping-Laufzeitumgebung. Stellt Folgendes dar: Einfachdateinamen, Einfachdatei-Verzeichnis, die Cache-Dateiverzeichnis, temporäres Dateiverzeichnis oder Typ der Mapping-Laufzeitumgebung.

Beim Definieren eines Parameters dürfen Sie keine Dollarzeichen im Parameternamen verwenden. Eine aus einer Informatica-Version vor 9.6.0 stammende Parameterdatei kann Parameternamen mit Dollarzeichen enthalten. Diese Parameter werden im Modellrepository ohne das Dollarzeichen gespeichert.

Parameter können in SQL-Anweisungen, Filter- oder Join-Bedingungen nicht verwendet werden. Relationale Datenobjekte können jedoch einen Feld- oder Eigenschaftsnamen aufweisen, der das Dollarzeichen (\$) enthält. Sie müssen das Dollarzeichen mit einem Escape-Zeichen versehen, wenn Sie in einer SQL-Umwandlungsanweisung, einer Filter- oder Join-Bedingung auf den Namen verweisen möchten.

Hinweis: Eine virtuelle Tabellenzuordnung schlägt fehl, wenn sie einen benutzerdefinierten Parameter enthält.

Prozess zum Ausführen von Mappings mit benutzerdefinierten Parametern

Ein benutzerdefinierter Parameter stellt einen konstanten Wert dar, den Sie in einer Parameterdatei vor dem Ausführen eines Mappings definieren.

Um Mappings mit verschiedenen Parameterwerten auszuführen, führen Sie die folgenden Aufgaben aus:

1. Erstellen Sie einen benutzerdefinierten Parameter und weisen Sie ihm einen Standardwert zu.
2. Wenden Sie den Parameter am Mapping oder an einem Datenobjekt bzw. einer Umwandlung im Mapping an.
3. Fügen Sie das Mapping zu einer Anwendung hinzu und stellen Sie die Anwendung bereit.
4. Erstellen Sie eine Parameterdatei, die den benutzerdefinierten Parameterwert enthält.
5. Führen Sie das Mapping über die Befehlszeile mit der Parameterdatei aus.

Speicherort zum Erstellen von benutzerdefinierten Parametern

Sie können benutzerdefinierte Parameter in physischen Datenobjekten, in einigen wieder verwendbaren Umwandlungen, in Mappings und in Mapplets erstellen.

Beim Erstellen eines Parameters in einem physischen Datenobjekt oder einer wieder verwendbaren Umwandlung können Sie den Parameter im Datenobjekt oder in der Umwandlung verwenden. Beim Erstellen eines Parameters in einem Mapping oder Mapplet können Sie den Parameter in einem nicht wieder verwendbaren Datenobjekt, einer nicht wieder verwendbaren Umwandlung oder einer wieder verwendbaren Lookup-Umwandlung im Mapping oder Mapplet verwenden, das Parameter akzeptiert. Beim Erstellen eines Parameters in einem Mapping können Sie auch den Parameter im Mapping verwenden.

Die folgende Tabelle listet die Objekte auf, in denen Sie benutzerdefinierte Parameter erstellen können:

Objekt	Parametertyp
Aggregatormwandlung	String
Fallumwandlerumwandlung	String
Angepasstes Datenobjekt (wieder verwendbar)	Verbindung
Einfachdatei-Datenobjekt	Verbindung, String

8. Klicken Sie auf **OK**.

Das Developer Tool fügt die Parameter zur Liste der Parameter hinzu.

Speicherort zum Zuweisen von Parametern

Weisen Sie einem Feld einen Systemparameter zu, wenn der Data Integration Service den Parameter mit dem für den Data Integration Service-Prozess definierten Wert ersetzen soll. Weisen Sie einem Feld einen benutzerdefinierten Parameter zu, wenn der Data Integration Service den Parameter mit dem in der Parameterdatei definierten Wert ersetzen soll.

Die folgende Tabelle listet die Objekte und Felder auf, denen Sie System- oder benutzerdefinierte Parameter zuweisen können:

Objekt	Feld
Zuordnung	Cache-Verzeichnis
Aggregatoremwandlung	Cache-Verzeichnis
Fallumwandlerumwandlung	Referenztable
Konsolidierungsumwandlung	Cache-Verzeichnis
Benutzerdefiniertes Datenobjekt	Verbindung
Ausnahmeumwandlung	
Einfachdatei-Datenobjekt	Quelldateiname Ausgabedateiname Quelldateiverzeichnis Ausgabedateiverzeichnis Verbindungsname Ablehnungsdateiverzeichnis
Joiner-Umwandlung	Cache-Verzeichnis
Beschreibungsumwandlung	Referenztable
Lookup-Umwandlung (Einfachdatei- oder Referenztabellen-Lookups) Lookup-Umwandlung (Einfachdatei-Lookups)	Lookup-Cache-Verzeichnisname
Lookup-Umwandlung (relationale Lookups)	Verbindung Lookup-Cache-Verzeichnisname
Mapping	Laufzeitumgebung Maximaler Parallelismus
Match-Umwandlung	Cache-Verzeichnis
Nichtrelationales Datenobjekt	Verbindung

Objekt	Feld
Parserumwandlung	Referenztable
Rangumwandlung	Cache-Verzeichnis
Aus zugehörigen relationalen Datenobjekten erstellte Lesen-Umwandlung	Verbindung
Sortiererumwandlung	Arbeitsverzeichnis
Standardisierungsumwandlung	Referenztable

Zuweisen eines Parameters

Weisen Sie einem Feld einen System-Mapping-Parameter zu, damit der Data Integration Service den Parameter mit dem für den Data Integration Service-Prozess definierten Wert ersetzt. Weisen Sie einem Feld einen benutzerdefinierten Mapping-Parameter zu, damit der Data Integration Service den Parameter mit dem in der Parameterdatei definierten Wert ersetzt, wenn Sie ein über die Befehlszeile ausführen.

1. Öffnen Sie das Feld, in dem Sie einen Parameter zuweisen möchten.
2. Klicken Sie auf **Parameter zuweisen**.
Das Dialogfeld **Parameter zuweisen** wird angezeigt.
3. Wählen Sie den System- oder benutzerdefinierten Parameter aus.
4. Klicken Sie auf **OK**.

Parameterdateien

Eine Parameterdatei ist eine XML-Datei, die benutzerdefinierte –Parameter und deren zugewiesenen Werte auflistet. Parameterdateien bieten die Flexibilität, bei jeder Ausführung eines Mapping die Parameterwerte zu ändern.

Die Parameterwerte definieren Eigenschaften für einen Arbeitsablauf oder für ein Mapping, Mapplet, physisches Datenobjekt oder eine Umwandlung in einer Mapping-Aufgabe, die der Arbeitsablauf ausführt. Der Data Integration Service wendet diese Werte an, wenn Sie einen Arbeitsablauf über die Befehlszeile ausführen und eine Parameterdatei angeben.

Die Parameterwerte definieren die Eigenschaften für ein Mapping, ein Mapplet, ein physisches Datenobjekt oder eine Umwandlung. Der Data Integration Service wendet diese Werte an, wenn Sie ein Mapping über die Befehlszeile ausführen und eine Parameterdatei angeben.

Sie können keine Systemparameterwerte in einer Parameterdatei definieren.

Sie können Parameter für mehrere Mappings in einer einzelnen Parameterdatei definieren. Sie können auch mehrere Parameterdateien erstellen und bei jeder Ausführung eines Mapping eine andere Datei verwenden. Der Data Integration Service liest die Parameterdatei am Anfang der Ausführung eines Mapping, um die Parameter aufzulösen.

Verwenden Sie den infacmd-Befehl `ms ListMappingParams`, um die in einem Mapping verwendeten Parameter mit den Standardwerten aufzulisten. Sie können die Ausgabe dieses Befehls als eine Parameterdateivorlage verwenden.

Verwenden Sie den infacmd-Befehl `ms RunMapping`, um ein Mapping mit einer Parameterdatei auszuführen.

Hinweis: Parameterdateien für Mappings und Arbeitsabläufe verwenden dieselbe Struktur. Sie können Parameter für bereitgestellte Mappings und für bereitgestellte Arbeitsabläufe in einer einzigen Parameterdatei definieren.

Parameterdateistruktur

Eine Mapping-Datei ist eine XML-Datei, die mindestens einen Parameter und ihren zugewiesenen Wert enthält.

Der Data Integration Service verwendet die in der Parameterdatei definierte Hierarchie, um Parameter und deren definierte Werte anzugeben. Die Hierarchie gibt den Arbeitsablauf, das Mapping, das Mapplet, das physische Datenobjekt oder die Umwandlung an, die den Parameter verwendet.

Der Data Integration Service verwendet die in der Parameterdatei definierte Hierarchie, um Parameter und deren definierte Werte anzugeben. Die Hierarchie gibt das Mapping, das Mapplet, das physische Datenobjekt oder die Umwandlung an, die der Parameter verwendet.

Sie definieren Parameterwerte innerhalb eines Elements auf der obersten Ebene eines Projekts oder einer Anwendung. Ein Projektelement definiert Parameterwerte, die bei der Ausführung eines bestimmten Mapping im Projekt einer beliebigen bereitgestellten Anwendung verwendet werden. Ein Projektelement definiert ebenfalls die Parameterwerte, die Sie bei der Ausführung eines beliebigen Mapping verwenden, das die Objekte im Mapping verwendet. Ein Anwendungselement definiert Parameterwerte, die bei der Ausführung eines bestimmten Mapping in einer bestimmten bereitgestellten Anwendung verwendet werden. Wenn Sie in einer Parameterdatei denselben Parameter in einem Element auf der obersten Ebene eines Projekts und ein Element auf der obersten Ebene einer Anwendung definieren, hat der für das Element auf der obersten Ebene einer Anwendung definierte Parameterwert Vorrang.

Der Data Integration Service sucht in der folgenden Reihenfolge nach Parameterwerten:

1. Der innerhalb eines Anwendungselements angegebene Wert.
2. Der innerhalb eines Projektelements angegebene Wert.
3. Der Parameterstandardwert.

Eine Parameterdatei muss mit der Struktur der XSD (XML Schema Definition) in der Parameterdatei übereinstimmen. Wenn die Parameterdatei nicht mit der Schema-Definition übereinstimmt, kann der Data Integration Service den Mapping nicht ausführen.

Auf dem Computer, der als Host für das Developer Tool eingesetzt wird, wird die XML-Schema-Definition der Parameterdatei in folgendem Verzeichnis angezeigt:

```
<Informatica Installation Directory>\clients\DeveloperClient\infacmd\plugins\ms  
\parameter_file_schema_1_0.xsd
```

Auf dem Computer, der als Host für Informatica Services eingesetzt wird, wird die XML-Schema-Definition der Parameterdatei in folgendem Verzeichnis angezeigt:

```
<Informatica Installation Directory>\isp\bin\plugins\ms\parameter_file_schema_1_0.xsd
```

Projektelement

Ein Projektelement definiert die Parameterwerte, die bei der Ausführung eines bestimmten Mapping in dem Projekt in einer bereitgestellten Anwendung verwendet werden. Ein Projektelement definiert ebenfalls die

Parameterwerte, die Sie bei der Ausführung eines beliebigen Mapping verwenden, das die Objekte im Mapping verwendet.

Das Projektelement definiert das Projekt in dem Model Repository, das Objekte enthält, die Parameter verwenden. Das Projektelement enthält zusätzliche Elemente, die spezielle Objekte innerhalb des Projekts definieren.

In der folgenden Tabelle werden die Elemente beschrieben, die ein Projektelement enthalten kann:

Elementname	Beschreibung
Ordner	Definiert einen Ordner innerhalb des Projekts. Verwenden Sie ein Ordnererelement, wenn Objekte in mehreren Ordnern innerhalb des Projekts organisiert sind. Ein Ordnererelement kann eine Datenquelle, ein Mapping, ein Mapplet, eine Umwandlung oder ein Arbeitsablaufelement enthalten. Ein Ordnererelement kann eine Datenquelle, ein Mapping, ein Mapplet oder ein Umwandlungselement enthalten.
Datenquelle	Definiert ein physisches Datenobjekt innerhalb des Projekts, das Parameter verwendet. Ein dataSource-Element enthält eine oder mehrere Parameterelemente, die Parameterwerte für das Datenobjekt definieren.
Mapping	Definiert ein Mapping in dem Projekt, das Parameter verwendet. Ein Mapping-Element enthält ein oder mehrere Parameterelemente, die Parameterwerte für das Mapping oder für ein nicht wieder verwendbares Datenobjekt, eine nicht wieder verwendbare Umwandlung oder eine wieder verwendbare Lookup-Umwandlung im Mapping definieren, das Parameter akzeptiert.
Mapplet	Definiert ein Mapplet innerhalb des Projekts, das Parameter verwendet. Ein Mapplet-Element enthält ein oder mehrere Parameterelemente, die Parameterwerte für ein nicht wieder verwendbares Datenobjekt, eine nicht wieder verwendbare Umwandlung oder eine wiederverwendbare Lookup-Umwandlung im Mapplet definieren, das Parameter akzeptiert.
Umwandlung	Definiert eine wiederverwendbare Umwandlung innerhalb des Projekts, das Parameter verwendet. Eine Umwandlungselement enthält ein oder mehrere Parameterelemente, die Parameterwerte für die Umwandlung definieren.
Arbeitsablauf	Definiert einen Arbeitsablauf in dem Projekt, das Parameter verwendet. Ein Arbeitsablauf-Element enthält ein oder mehrere Parameterelemente, die Parameterwerte für den Arbeitsablauf definieren.

Beim Ausführen eines Arbeitsablaufs mit einer Parameterdatei, die Parameterwerte in einem Element auf der obersten Ebene eines Projekts definieren, wendet der Data Integration Service die Parameterwerte auf den angegebenen Arbeitsablauf an. Der Dienst wendet Parameterwerte ebenfalls auf jedes der angegebenen Objekte an, die bei einer Mapping-Aufgabe im Arbeitsablauf in eine Mapping-Ausführung einbezogen werden.

Beispiel: Der Data Integration Service soll Parameterwerte anwenden, wenn Sie den Arbeitsablauf "MyWorkflow" ausführen. Der Arbeitsablauf enthält eine Mapping-Aufgabe, die das Mapping "MyMappings" ausführt. Das Mapping enthält das Datenobjekt "MyDataObject" und die wiederverwendbare Umwandlung "MyTransformation". Sie möchten die Parameterwerte verwenden, wenn Sie "MyWorkflow" in einer beliebigen bereitgestellten Anwendung ausführen. Sie möchten die Parameterwerte ebenfalls anwenden, wenn Sie einen beliebigen anderen Arbeitsablauf mit einer Mapping-Aufgabe ausführen, die ein Mapping mit diesen Objekten im Projekt "MyProject" ausführt. Definieren Sie die Parameter innerhalb der folgenden Elemente:

```
<project name="MyProject">

  <!-- Apply this parameter value to workflow "MyWorkflow" in project "MyProject". -->
  <workflow name="MyWorkflow">
    <parameter name="MyWorkflow_Param">Param_value</parameter>
  </workflow>

  <!-- Apply this parameter value when you run any workflow that runs mapping
```

```

"MyMapping"
    in project "MyProject". -->
    <mapping name="MyMapping">
        <parameter name="MyMapping_Param">Param_value</parameter>
    </mapping>

    <!-- Apply this parameter value when you run any workflow that runs any mapping that
        uses data object "MyDataObject" in project "MyProject". -->
    <dataSource name="MyDataObject">
        <parameter name="MyDataObject_Param">Param_value</parameter>
    </dataSource>

    <!-- Apply this parameter value when you run any workflow that runs any mapping that
        uses reusable transformation "MyTransformation" in project "MyProject". -->
    <transformation name="MyTransformation">
        <parameter name="MyTransformation_Param">Param_value</parameter>
    </transformation>
</project>

```

Beim Ausführen eines Mappings mit einer Parameterdatei, die Parameterwerte in einem Element auf der obersten Ebene eines Projekts definiert, wendet der Data Integration Service die Parameterwerte auf das angegebene Mapping an. Der Dienst wendet Parameterwerte ebenfalls auf jedes der angegebenen Objekte an, die im Mapping enthalten sind.

Beispiel: Bei der Ausführung des Mappings "MyMapping" soll der Data Integration Service Parameterwerte anwenden. Das Mapping enthält das Datenobjekt "MyDataObject" und die wiederverwendbare Umwandlung "MyTransformation". Sie möchten die Parameterwerte bei der Ausführung von "MyMapping" in einer beliebigen bereitgestellten Anwendung verwenden. Sie möchten die Parameterwerte ebenfalls verwenden, wenn Sie ein beliebiges anderes Mapping ausführen, das "MyDataObject" und "MyTransformation" im Projekt "MyProject" verwendet. Definieren Sie die Parameter innerhalb der folgenden Elemente:

```

<project name="MyProject">

    <!-- Apply this parameter value to mapping "MyMapping" in project "MyProject". -->
    <mapping name="MyMapping">
        <parameter name="MyMapping_Param">Param_value</parameter>
    </mapping>

    <!-- Apply this parameter value when you run any mapping that uses data object
        "MyDataObject" in project "MyProject". -->
    <dataSource name="MyDataObject">
        <parameter name="MyDataObject_Param">Param_value</parameter>
    </dataSource>

    <!-- Apply this parameter value when you run any mapping that uses reusable
transformation
        "MyTransformation" in project "MyProject". -->
    <transformation name="MyTransformation">
        <parameter name="MyTransformation_Param">Param_value</parameter>
    </transformation>
</project>

```

Anwendungselement

Ein Anwendungselement bietet einen Laufzeitbereich für ein Projektelement. Ein Anwendungselement definiert die bei der Ausführung eines bestimmten Mapping in einer bestimmten bereitgestellten Anwendung zu verwendenden Parameterwerte.

Ein Anwendungselement definiert die bereitgestellte Anwendung mit den Objekten, die Parameter verwenden. Ein Anwendungselement kann ein Mappingelement enthalten, das einen Mapping in der bereitgestellten Anwendung definiert, die Parameter verwendet. Ein Mappingelement enthält ein Projektelement.

Beispiel: Der Data Integration Service soll Parameterwerte anwenden, wenn Sie den Arbeitsablauf "MyWorkflow" in der bereitgestellten Anwendung "MyApp" ausführen. Sie möchten die Parameterwerte nicht verwenden, wenn Sie den Arbeitsablauf in einer anderen Anwendung oder bei der Ausführung eines anderen

Arbeitsablaufs im Projekt "MyProject" ausführen. Definieren Sie die Parameter innerhalb der folgenden Elemente:

```
<application name="MyApp">
  <workflow name="MyWorkflow">
    <project name="MyProject">
      <workflow name="MyWorkflow">
        <parameter name="MyWorkflow_Param">Param_value</parameter>
      </workflow>

      <mapping name="MyMapping">
        <parameter name="MyMapping_Param">Param_value</parameter>
      </mapping>

      <dataSource name="MyDataObject">
        <parameter name="MyDataObject_Param">Param_value</parameter>
      </dataSource>

      <transformation name="MyTransformation">
        <parameter name="MyTransformation_Param">Param_value</parameter>
      </transformation>
    </project>
  </workflow>
</application>
```

Beispiel: Der Data Integration Service soll bei der Ausführung des Mappings "MyMapping" in der bereitgestellten Anwendung "MyApp" Parameterwerte anwenden. Sie möchten die Parameterwerte nicht verwenden, wenn Sie das Mapping in einer anderen Anwendung oder bei der Ausführung eines anderen Mappings im Projekt "MyProject" ausführen. Definieren Sie die Parameter innerhalb der folgenden Elemente:

```
<application name="MyApp">
  <mapping name="MyMapping">
    <project name="MyProject">
      <mapping name="MyMapping">
        <parameter name="MyMapping_Param">Param_value</parameter>
      </mapping>

      <dataSource name="MyDataObject">
        <parameter name="MyDataObject_Param">Param_value</parameter>
      </dataSource>

      <transformation name="MyTransformation">
        <parameter name="MyTransformation_Param">Param_value</parameter>
      </transformation>
    </project>
  </mapping>
</application>
```

Regeln und Richtlinien für Parameterdateien

Beim Erstellen von Parameterdateien werden bestimmte Regeln und Richtlinien angewendet.

Verwenden Sie die folgenden Regeln beim Erstellen eines Parameterdatei:

- Parameterwerte dürfen nicht leer sein. Beispiel: Der Datenintegrationsdienst kann das Mapping nicht ausführen, wenn die Parameterdatei den folgenden Eintrag enthält:

```
<parameter name="Param1"> </parameter>
```
- Innerhalb eines Elements muss die Groß-/Kleinschreibung für Artefaktnamen nicht berücksichtigt werden. Der Data Integration Service interpretiert `<Anwendungsname="App1">` und `<Anwendungsname="APP1">` als dieselbe Anwendung.
- Ein Parameter, der eine Referenztable angibt, muss einen Schrägstrich verwenden, um Ordnernamen in einem Repository-Ordnerpfad zu trennen.

Beispielparameterdatei

Das folgende Beispiel zeigt eine Beispielparameterdatei, die zur Ausführung eines Mapping verwendet wird.

```
<?xml version="1.0"?>
<root description="Sample Parameter File"
  xmlns="http://www.informatica.com/Parameterization/1.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <!--
    The Data Integration Service uses this section only when you run workflow
    "Workflow1" or
    "Workflow2" in project "Project1" in deployed application "App1."

    This section assigns values to parameters created in workflows "Workflow1" and
    "Workflow2."
  -->
  <application name="App1">
    <workflow name="Workflow1">
      <project name="Project1">
        <workflow name="Workflow1">
          <parameter name="WORKFLOW1_PARAM1">WORKFLOW1_PARAM1_VAL</parameter>
          <parameter name="WORKFLOW1_PARAM2">WORKFLOW1_PARAM2_VAL</parameter>
        </workflow>
      </project>
    </workflow>
    <workflow name="Workflow2">
      <project name="Project1">
        <workflow name="Workflow2">
          <parameter name="WORKFLOW2_PARAM1">WORKFLOW2_PARAM1_VAL</parameter>
          <parameter name="WORKFLOW2_PARAM2">WORKFLOW2_PARAM2_VAL</parameter>
        </workflow>
      </project>
    </workflow>
  </application>

  <!--
    The Data Integration Service uses this section only when you run workflow
    "Workflow1"
    in project "Project1" in deployed application "App2." "Workflow1" includes a
    Mapping
    task that runs mapping "Map1".

    This section assigns values to parameters created in the following objects:
    * Workflow "Workflow1"
    * Data source "DS1" in mapping "Map1"
    * Mapping "Map1"
  -->
  <application name="App2">
    <workflow name="Workflow1">
      <project name="Project1">
        <workflow name="Workflow1">
          <parameter name="WORKFLOW1_PARAM1">WORKFLOW1_PARAM1_VAL</parameter>
          <parameter name="WORKFLOW1_PARAM2">WORKFLOW1_PARAM2_VAL</parameter>
        </workflow>
        <dataSource name="DS1">
          <parameter name="PROJ1_DS1">PROJ1_DS1_APP2_MAP1_VAL</parameter>
          <parameter name="PROJ1_DS1">PROJ1_DS1_APP2_MAP1_VAL</parameter>
        </dataSource>
        <mapping name="Map1">
          <parameter name="MAP1_PARAM2">MAP1_PARAM2_VAL</parameter>
        </mapping>
      </project>
    </workflow>
  </application>

  <!--
    The Data Integration Service uses this section when you run any workflow that
    includes a Mapping task that runs a mapping that includes data source "DS1" or
    mapplet "DS1" in project "Project1".
```

```

        This section assigns values to parameters created in the following
objects:
    * Data source "DS1"
    * Mapplet "DS1"
-->
<project name="Project1">
    <dataSource name="DS1">
        <parameter name="PROJ1_DS1">PROJ1_DS1_VAL</parameter>
        <parameter name="PROJ1_DS1_PARAM1">PROJ1_DS1_PARAM1_VAL</parameter>
    </dataSource>
    <mapplet name="DS1">
        <parameter name="PROJ1_DS1">PROJ1_DS1_VAL</parameter>
        <parameter name="PROJ1_DS1_PARAM1">PROJ1_DS1_PARAM1_VAL</parameter>
    </mapplet>
</project>

<!--
    The Data Integration Service uses this section when you run any workflow that
    includes a Mapping task that runs a mapping that includes reusable
transformation
    "TX2", mapplet "MPLT1" in folder "Folder2", or Mapplet "RULE1" in nested folder
    "Folder2_1_1" in project "Project2".

    This section assigns values to parameters created in the following
objects:
    * Reusable transformation "TX2"
    * Mapplet "MPLT1" in folder "Folder2"
    * Mapplet "RULE1" in nested folder "Folder2_1_1"
-->
<project name="Project2">
    <transformation name="TX2">
        <parameter name="RTM_PATH">Project1\Folder1\RTM1</parameter>
    </transformation>
    <folder name="Folder2">
        <mapplet name="MPLT1">
            <parameter name="PROJ2_FOLD2_MPLT1">PROJ2_FOLD2_MPLT1_VAL</parameter>
        </mapplet>
        <folder name="Folder2_1">
            <folder name="Folder2_1_1">
                <mapplet name="RULE1">
                    <parameter name="PROJ2_RULE1">PROJ2_RULE1_VAL</parameter>
                </mapplet>
            </folder>
        </folder>
    </folder>
</project>
</root>

<?xml version="1.0"?>
<root description="Sample Parameter File"
    xmlns="http://www.informatica.com/Parameterization/1.0"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <!--
        The Data Integration Service uses this section only when you run mapping "Map1"
or "Map2"
        in project "Project1" in deployed application "App1."

        This section assigns values to parameters created in mappings "Map1" and "Map2."
-->
    <application name="App1">
        <mapping name="Map1">
            <project name="Project1">
                <mapping name="Map1">
                    <parameter name="MAP1_PARAM1">MAP1_PARAM1_VAL</parameter>
                    <parameter name="MAP1_PARAM2">MAP1_PARAM2_VAL</parameter>
                </mapping>
            </project>
        </mapping>
        <mapping name="Map2">
            <project name="Project1">
                <mapping name="Map2">

```

```

        <parameter name="MAP2_PARAM1">MAP2_PARAM1_VAL</parameter>
        <parameter name="MAP2_PARAM2">MAP2_PARAM2_VAL</parameter>
    </mapping>
</project>
</mapping>
</application>

<!--
The Data Integration Service uses this section only when you run mapping "Map1"
in
    project "Project1" in deployed application "App2."

This section assigns values to parameters created in the following
objects:
    * Reusable data source "DS1" in mapping "Map1"
    * Mapping "Map1"
-->
<application name="App2">
    <mapping name="Map1">
        <project name="Project1">
            <dataSource name="DS1">
                <parameter name="PROJ1_DS1">PROJ1_DS1_APP2_MAP1_VAL</parameter>
                <parameter name="PROJ1_DS1">PROJ1_DS1_APP2_MAP1_VAL</parameter>
            </dataSource>
            <mapping name="Map1">
                <parameter name="MAP1_PARAM2">MAP1_PARAM2_VAL</parameter>
            </mapping>
        </project>
    </mapping>
</application>

<!--
The Data Integration Service uses this section when you run any mapping that
includes data source "DS1" or mapplet "DS1" in project "Project1."

This section assigns values to parameters created in the following
objects:
    * Data source "DS1"
    * Mapplet "DS1"
-->
<project name="Project1">
    <dataSource name="DS1">
        <parameter name="PROJ1_DS1">PROJ1_DS1_VAL</parameter>
        <parameter name="PROJ1_DS1_PARAM1">PROJ1_DS1_PARAM1_VAL</parameter>
    </dataSource>
    <mapplet name="DS1">
        <parameter name="PROJ1_DS1">PROJ1_DS1_VAL</parameter>
        <parameter name="PROJ1_DS1_PARAM1">PROJ1_DS1_PARAM1_VAL</parameter>
    </mapplet>
</project>

<!--
The Data Integration Service uses this section when you run any mapping that
includes reusable transformation "TX2", mapplet "MPLT1" in folder "Folder2",
or Mapplet "RULE1" in nested folder "Folder2_1_1" in project "Project2".

This section assigns values to parameters created in the following
objects:
    * Reusable transformation "TX2"
    * Mapplet "MPLT1" in folder "Folder2"
    * Mapplet "RULE1" in nested folder "Folder2_1_1"
-->
<project name="Project2">
    <transformation name="TX2">
        <parameter name="RTM_PATH">Project1\Folder1\RTM1</parameter>
    </transformation>
    <folder name="Folder2">
        <mapplet name="MPLT1">
            <parameter name="PROJ2_FOLD2_MPLT1">PROJ2_FOLD2_MPLT1_VAL</parameter>
        </mapplet>
    </folder name="Folder2_1_1">

```

```

        <folder name="Folder2_1_1">
            <mapplet name="RULE1">
                <parameter name="PROJ2_RULE1">PROJ2_RULE1_VAL</parameter>
            </mapplet>
        </folder>
    </folder>
</folder>
</project>
</root>

```

Erstellen einer Parameterdatei

Der Befehl `infacmd ms ListMappingParams` listet die Parameter, die in einem Mapping in einer bereitgestellten Anwendung verwendet werden, sowie die Standardwerte für jeden Parameter auf. Verwenden Sie die Ausgabe dieses Befehls, um eine Parameterdatei zu erstellen.

Der Befehl listet alle Parameter innerhalb eines Elements auf der obersten Ebene eines Projekts auf. Sie können die Standardparameterwerte im Projektelement bearbeiten, um die Werte für ein Mapping in dem Projekt festzulegen, das für eine beliebige Anwendung bereitgestellt wird. Wahlweise können Sie das Projektelement in ein Anwendungselement kopieren, um die Werte für ein bestimmtes Mapping in einer bestimmten bereitgestellten Anwendung festzulegen.

Wenn das Mapping Objekte desselben Typs verwendet, die in demselben Projekt oder Ordner vorhanden sind, denselben Namen aufweisen und Parameter verwenden, schlägt der Befehl `ms ListMappingParams` fehl. Ein Ordner enthält beispielsweise die Bezeichnungsumwandlung „T1“ und die Standardisierer-Umwandlung „T1.“ Wenn beide Umwandlungen Parameter verwenden, schlägt der Befehl `ms ListMappingParams` fehl. Wenn die Objekte sich in anderen Ordnern befinden, oder wenn ein Objekt keine Parameter verwendet, dann listet der Befehl `ms ListMappingParams` erfolgreich die Parameter auf, die in dem Mapping verwendet werden.

1. Führen Sie den Befehl `infacmd ms ListMappingParams` aus, um alle im Mapping verwendeten Parameter und den Standardwert für jeden Parameter aufzulisten.

Das Argument `-o` sendet die Befehlsausgabe an eine XML-Datei.

Der folgende Befehl listet beispielsweise die Parameter im Arbeitsablauf `MyWorkflow` in der Datei `"MyOutputFile.xml"` auf:

```
infacmd wfs ListWorkflowParams -dn MyDomain -sn MyDataIntSvs -un MyUser -pd
MyPassword -a MyApplication -wf MyWorkflow -o MyOutputFile.xml
```

Der folgende Befehl listet beispielsweise die Parameter im Mapping `MyMapping` in der Datei `"MyOutputFile.xml"` auf:

```
infacmd ms ListMappingParams -dn MyDomain -sn MyDataIntSvs -un MyUser -pd MyPassword
-a MyApplication -m MyMapping -o MyOutputFile.xml
```

Der Data Integration Service listet alle Parameter im Mapping mit den jeweiligen Standardwerten innerhalb eines Elements auf der obersten Ebene eines Projekts auf.

2. Wenn Sie das Argument `-o` nicht angegeben haben, kopieren Sie die Befehlsausgabe in eine XML-Datei und speichern Sie die Datei.
3. Bearbeiten Sie die XML-Datei und ersetzen Sie die Parameterstandardwerte mit den Werten, die Sie beim Ausführen des Mapping verwenden möchten.

Wenn Sie die Werte für das Mapping in einer bestimmten Anwendung festlegen möchten, kopieren Sie das Element auf der obersten Ebene des Projekts in das Element auf der obersten Ebene der Anwendung.

4. Speichern Sie die XML-Datei.

Ausführen eines Mapping mit einer Parameterdatei

Verwenden Sie den Befehl `infacmd ms RunMapping`, um ein Mapping mit einer Parameterdatei auszuführen. Das Argument `-pf` gibt den Namen der Parameterdatei an.

Beispiel: Der folgende Befehl führt den Arbeitsablauf `MyWorkflow` unter Verwendung der Parameterdatei `"MyParamFile.xml"` aus:

```
infacmd wfs StartWorkflow -dn MyDomain -sn MyDataIntSvs -un MyUser -pd MyPassword -a  
MyApplication -wf MyWorkflow -pf MyParamFile.xml
```

Beispiel: Der folgende Befehl führt den Arbeitsablauf `MyMapping` unter Verwendung der Parameterdatei `"MyParamFile.xml"` aus:

```
infacmd ms RunMapping -dn MyDomain -sn MyDataIntSvs -un MyUser -pd MyPassword -a  
MyApplication -m MyMapping -pf MyParamFile.xml
```

Der Data Integration Service kann das Mapping nicht ausführen, wenn Sie es mit einer Parameterdatei ausführen und eine der folgenden Bedingungen zutrifft:

- Der Computer, auf dem Sie den Befehl `infacmd ms RunMapping` ausführen, kann nicht auf die Parameterdatei zugreifen.
- Der Parameterdatei ist nicht gültig oder nicht vorhanden.
- Objekte desselben Typs existieren in demselben Projekt oder Ordner, haben denselben Namen, und verwenden Parameter. Beispielsweise enthält ein Ordner die Bezeichnungsumwandlung „T1“ und die Standardisierungsumwandlung „T1.“ Wenn beide Umwandlungen Parameter verwenden, kann der Data Integration Service das Mapping nicht ausführen, wenn Sie es mit einer Parameterdatei ausführen. Wenn sich die Objekte in verschiedenen Ordnern befinden, oder wenn ein Objekt keine Parameter verwendet, kann der Data Integration Service das Mapping ausführen.

KAPITEL 4

Export an PowerCenter

Dieses Kapitel umfasst die folgenden Themen:

- [Export an PowerCenter - Übersicht, 46](#)
- [PowerCenter Release-Kompatibilität, 47](#)
- [Mapplet-Export, 47](#)
- [Export an PowerCenter - Optionen, 48](#)
- [Exportieren eines Objekts an PowerCenter, 49](#)
- [Exporteinschränkungen, 50](#)
- [Regeln und Richtlinien für das Exportieren an PowerCenter, 52](#)
- [Fehlerbehebung - Exportieren an PowerCenter, 52](#)

Export an PowerCenter - Übersicht

Sie können vom Developer Tool aus Objekte zur Verwendung in PowerCenter exportieren.

Sie können die folgenden Objekte exportieren:

- Mappings. Exportieren Sie Mappings an PowerCenter-Mappings oder -Mapplets.
- Mapplets. Exportieren Sie Mapplets an PowerCenter-Mapplets.
- Logische Datenobjektmodelle. Exportieren Sie die logischen Datenobjektmodelle in PowerCenter-Mapplets.

Sie können Objekte in ein PowerCenter-Repository oder in eine XML-Datei exportieren. Falls Sie Objekte in eine XML-Datei exportieren, können PowerCenter-Benutzer die Datei in das PowerCenter-Repository importieren.

Wenn Sie Objekte exportieren, geben Sie Exportoptionen, wie z. B. die PowerCenter-Version, die Vorgehensweise zum Konvertieren von Mappings und Mapplets sowie den Export von Referenztabellen.

Sie müssen Metadata Converter auf dem Computer installieren, auf dem der PowerCenter Express-Client gehostet wird, um Objekte aus PowerCenter Express nach PowerCenter zu exportieren.

PowerCenter Release-Kompatibilität

Um zu überprüfen, ob Objekte mit einer bestimmten PowerCenter-Version kompatibel sind, stellen Sie das PowerCenter Release-Kompatibilitätslevel ein. Das Kompatibilitätslevel trifft auf alle Mappings, Mapplets und logischen Datenobjektmodelle zu, die Sie im Developer Tool anzeigen können.

Sie können das Developer Tool konfigurieren, um anhand einer bestimmten Version von PowerCenter zu validieren, oder Sie können es so einstellen, dass die Validierung der Versionskompatibilität übersprungen wird. Standardmäßig führt das Developer Tool keine Validierung von Objekten anhand einer PowerCenter-Version durch.

Stellen Sie das Kompatibilitätslevel auf eine PowerCenter-Version ein, bevor Sie Objekte in PowerCenter exportieren. Wenn Sie das Kompatibilitätslevel eingestellt haben, führt das Developer Tool zwei Validierungsprüfungen durch, wenn Sie ein Mapping, Mapplet oder ein logisches Datenobjektmodell validieren. Das Developer Tool überprüft zunächst, ob das Objekt im Developer Tool gültig ist. Falls das Objekt gültig ist, überprüft das Developer Tool weiter, ob das Objekt für den Export in die gewählte Version von PowerCenter gültig ist. Sie können Kompatibilitätsfehler in der Ansicht **Validierungsprotokoll** anzeigen.

Einstellen des Kompatibilitätslevels

Stellen Sie das Kompatibilitätslevel ein, um Mappings, Mapplets und logische Datenobjektmodelle anhand einer PowerCenter-Version zu validieren. Wenn Sie keines auswählen, überspringt das Developer Tool die Versionskompatibilitätsvalidierung beim Validieren eines Objekts.

1. Klicken Sie auf **Bearbeiten > Kompatibilitätslevel**.
2. Wählen Sie das Kompatibilitätslevel aus.

Das Developer Tool setzt neben dem ausgewählten Kompatibilitätslevel im Menü einen Punkt. Das Kompatibilitätslevel trifft auf alle Mappings, Mapplets und logischen Datenobjektmodelle zu, die Sie im Developer Tool anzeigen können.

Mapplet-Export

Wenn Sie ein Mapplet exportieren oder ein Mapping als Mapplet exportieren, erstellt der Exportprozess Objekte im Mapplet. Der Exportprozess benennt also einige Mapplet-Objekte um.

Der Exportprozess kann die folgenden Mapplet-Objekte in der Export-XML-Datei erstellen:

Ausdrucksumwandlungen

Der Exportprozess erstellt unmittelbar nach jeder Eingabe-Umwandlung und unmittelbar vor jeder Ausgabe-Umwandlung in einem Mapplet eine Ausdrucksumwandlung. Der Exportprozess benennt die Ausdrucksumwandlungen wie folgt:

Expr_<InputOrOutputTransformationName>

Die Ausdrucksumwandlungen enthalten Pass-Through-Ports.

Ausgabeumwandlungen

Falls Sie ein Mapplet exportieren und Ziele in Ausgabeumwandlungen konvertieren, erstellt der Exportprozess eine Ausgabeumwandlung für jedes Ziel. Der Exportprozess benennt die Ausgabeumwandlungen wie folgt:

<MappletInstanceName>_<TargetName>

Der Exportprozess benennt die folgenden Mapplet-Objekte in der Export-XML-Datei um:

Mapplet-Eingabe- und -Ausgabeumwandlungen

Der Exportprozess benennt Mapplet-Ein- und -Ausgabeumwandlungen wie folgt:

<TransformationName>_<InputOrOutputGroupName>

Mapplet-Ports

Der Exportprozess benennt die Mapplet-Ports wie folgt um:

<PortName>_<GroupName>

Export an PowerCenter - Optionen

Wenn Sie ein Objekt zur Verwendung in PowerCenter exportieren, müssen Sie die Exportoptionen festlegen.

In der folgenden Tabelle werden die Exportoptionen erläutert.

Option	Beschreibung
Projekt	Projekt im Modellrepository, aus dem Objekte exportiert werden sollen.
Ziel-Release:	PowerCenter-Release-Version.
Ausgewählte Objekte in eine Datei exportieren	Exportiert Objekte in eine PowerCenter XML-Datei. Wenn Sie diese Option auswählen, geben Sie den Namen und Speicherort der XML-Exportdatei an.
Ausgewählte Objekte in ein PowerCenter-Repository exportieren	<p>Exportiert Objekte in ein PowerCenter-Repository. Wenn Sie diese Option auswählen, geben Sie die folgenden Verbindungsdetails für das PowerCenter-Repository an:</p> <ul style="list-style-type: none">- Hostname. Hostname des PowerCenter Domänengateways.- Portnummer. HTTP-Portnummer des PowerCenter-Domänengateways.- Authentifizierungstyp. Wählen Sie einen der folgenden Werte aus: Kerberos Single Sign On, Nativ oder LDAP.- Authentifizierungstyp. Wählen Sie einen der folgenden Werte aus: Nativ oder LDAP.- Benutzername. Benutzername des Repositories- Passwort. Passwort für den Repository-Benutzernamen. <p>Hinweis: Geben Sie den Benutzernamen und das Passwort ein, wenn als Authentifizierungstyp „Nativ“ oder „LDAP“ ausgewählt wurde.</p> <ul style="list-style-type: none">- Sicherheitsdomäne. Handelt es sich beim Authentifizierungstyp um LDAP, geben Sie den Namen der LDAP-Sicherheitsdomäne ein. Ansonsten geben Sie "Native" ein.- Name des Repositories. PowerCenter-Repository-Name.
Senden an Repository-Ordner	Exportiert Objekte in den angegebenen Ordner im PowerCenter-Repository.
Steuerdatei verwenden	Exportiert Objekte in das PowerCenter-Repository mithilfe der angegebenen <i>pmrep</i> -Steuerdatei.

Option	Beschreibung
Konvertieren von exportierten Mappings zu PowerCenter-Mapplets	Wandelt Developer-Tool-Mappings in PowerCenter-Mapplets um. Das Developer-Tool wandelt Datenobjekte, die als Quellen und Ziele in den Mappings verwendet werden, in Eingabe- und Ausgabeumwandlungen in einem PowerCenter-Mapplet um.
Ziel-Mapplets umwandeln	Wandelt in Mapplets als Ziele verwendete Datenobjekte in Ausgabeumwandlungen im PowerCenter-Mapplet um. PowerCenter-Mapplets dürfen keine Ziele enthalten. Der Export schlägt fehl, wenn das Exportobjekt ein Mapplet mit einem Ziel enthält und diese Option von Ihnen nicht ausgewählt wurde.
Referenzdaten exportieren	Exportiert alle, von einer Umwandlung in einem von Ihnen exportierten Objekt verwendeten Referenztabellendaten.
Referenzdaten-Speicherort	Speicherort für die Referenztabellendaten, die das Developer-Tool exportiert. Das Developer-Tool exportiert die Referenztabellendaten als eine oder mehrere Wörterbuchdateien. Geben Sie einen Pfad zum Verzeichnis auf der Maschine ein, die das Developer-Tool hostet.
Codepage	Codepage des PowerCenter Repositorys.

Exportieren eines Objekts an PowerCenter

Wenn Sie Mappings, Mapplets oder logische Datenobjektmodelle in PowerCenter exportieren, können Sie die Objekte in eine Datei oder in ein PowerCenter-Repository exportieren.

Bevor Sie ein Objekt exportieren, stellen Sie das Kompatibilitätslevel auf die geeignete PowerCenter-Version. Validieren Sie das Objekt, um zu überprüfen, ob es mit der PowerCenter-Version kompatibel ist.

1. Klicken Sie auf **Datei > Exportieren**.
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 im Modellrepository aus, aus dem Objekte exportiert werden sollen.
5. Wählen Sie die PowerCenter-Version aus, in die die Objekte exportiert werden sollen.
6. Wählen Sie den Speicherort aus, in den die Objekte exportiert werden sollen. Sie können die Objekte in eine XML-Datei oder ein PowerCenter-Repository exportieren.
 - Geben Sie zum Exportieren von Objekten in eine Datei den Namen und Speicherort einer XML-Datei ein.
 - Klicken Sie zum Exportieren von Objekten in ein PowerCenter-Repository auf **Durchsuchen**, um die Verbindungsdetails für das Repository festzulegen.
7. Wenn Sie in ein PowerCenter-Repository exportieren, wählen Sie einen Ordner im PowerCenter-Repository oder die Steuerdatei *pmrep* aus, in der die Vorgehensweise zum Importieren von Objekten in PowerCenter festgelegt ist.

8. Wählen Sie **Exportierte Mappings in PowerCenter-Mapplets umwandeln**, um die Developer-Tool-Mappings in Mappings in PowerCenter umzuwandeln.
9. Wählen Sie **Ziel-Mapplets umwandeln** aus, um als Ziele in einem Mapplet verwendete Datenobjekte in Ausgabeumwandlungen im PowerCenter-Mapplet umzuwandeln.
10. Wählen Sie **Referenzdaten exportieren** aus, um alle Referenztabellendaten zu exportieren, die von einer Umwandlung in einem zu exportierenden Objekt verwendet werden.
11. Wenn Sie die Referenzdaten exportieren möchten, geben Sie den Speicherort für die Referenztabellendaten ein, die vom Developer-Tool exportiert werden.
12. Wählen Sie die Codepage des PowerCenter-Repositorys aus.
13. Klicken Sie auf **Weiter**.
Sie werden im Developer Tool aufgefordert, die Objekte für den Export auszuwählen.
14. Wählen Sie die zu exportierenden Objekte und klicken Sie auf **Fertig stellen**.
Das Developer Tool exportiert die Objekte an den von Ihnen gewählten Ort.

Wenn Sie Objekte in eine Datei exportieren, können Sie Objekte aus der Datei in das PowerCenter-Repository importieren.

Wenn Sie Referenztabellendaten exportieren, kopieren Sie die Referenzdaten in die PowerCenter-Verzeichnisstruktur auf dem Computer, der als Host für Informatica-Dienst eingesetzt wird. Die Speicherorte für die Referenzdaten müssen mit den Speicherorten der Referenztabellenobjekte im Modellrepository übereinstimmen.

Beispiel: Kopieren Sie die Referenzdaten in folgendem Verzeichnis:

```
<PowerCenter-Installationsverzeichnis>\Dienste\<Modellrepository-Projektname>\<Ordnername>
```

Exporteinschränkungen

Beim Exportieren eines Modellrepository-Objekts nach PowerCenter werden bestimmte Modellrepository-Objekte unter Umständen nicht in das PowerCenter-Repository exportiert. Sie können kein Mapping oder Mapplet exportieren, das ein Objekt enthält, das in PowerCenter nicht gültig ist.

Die folgenden Objekte können nicht nach PowerCenter exportiert werden:

Objekte mit langen Namen

PowerCenter können kein Mapping, Mapplet oder Objekt innerhalb eines Mappings oder Mapplets importieren, falls der Objektname 80 Zeichen übertrifft.

Mappings oder Mapplets, die eine benutzerdefinierte Datenumwandlung enthalten.

Sie können keine Mappings oder Mapplets exportieren, die benutzerdefinierte Datenumwandlungen enthalten.

Mappings oder Mapplets, die eine Joiner-Umwandlung mit bestimmten Join-Bedingungen enthalten.

Das Developer Tool erlaubt Ihnen nicht, Mappings und Mapplets zu exportieren, die eine Joiner-Umwandlung mit einer Join-Bedingung enthalten, die in PowerCenter nicht gültig ist. In PowerCenter definiert ein Benutzer Join-Bedingungen auf Basis von Gleichheit zwischen den angegebenen Master- und Detail-Quellen. Im Developer Tool können Sie andere Join-Bedingungen definieren. Zum Beispiel können Sie eine Join-Bedingung auf Basis von Gleichheit oder Ungleichheit zwischen Master- und Detail-Quellen definieren. Sie können eine Join-Bedingung definieren, die Umwandlungsausdrücke enthält. Sie

können auch eine Join-Bedingung wie etwa $1 = 1$ definieren, sodass die Joiner-Umwandlung einen Cross-Join ausführt.

Diese Typen von Join-Bedingungen sind in PowerCenter ungültig. Deshalb können Sie keine Mappings oder Mapplets in PowerCenter exportieren, die Joiner-Umwandlungen mit diesen Arten von Join-Bedingungen enthalten

Mappings oder Mapplets, die eine Lookup-Transformation mit umbenannten Ports enthalten.

Der PowerCenter Integration Service führt auf Basis der Lookup-Ports in der Umwandlungs- und Lookup-Bedingung Abfragen in den Lookup-Quellen aus. Deshalb müssen die Portnamen in den Lookup-Transformationen mit den Spaltennamen in der Lookup-Quelle übereinstimmen.

Mappings oder Mapplets, die eine bestimmte Lookup-Transformation mit bestimmten benutzerdefinierten SQL-Abfragen enthalten

Das Developer Tool verwendet andere Regeln als PowerCenter, um in einer Lookup-Transformation SQL-Abfrage-Syntax zu validieren. Eine im Developer Tool geschriebene benutzerdefinierte SQL-Abfrage, die das AS-Schlüsselwort oder berechnete Felder verwendet, ist in PowerCenter nicht gültig. Deshalb können Sie keine Mappings oder Mapplets in PowerCenter exportieren, die eine Lookup-Transformation mit einer benutzerdefinierten SQL-Abfrage enthalten, die das AS-Schlüsselwort oder berechnete Felder verwendet.

Mappings oder Mapplets, die Quellen enthalten, welche in PowerCenter nicht verfügbar sind

Wenn Sie ein Mapping oder Mapplet zu exportieren versuchen, das Quellen enthält, welche in PowerCenter nicht verfügbar sind, schlägt der Export des Mapping oder Mapplet fehl.

Sie können kein Mapping oder Mapplet mit den folgenden Quellen exportieren:

- Komplexes Datei-Datenobjekt
- DataSift
- Webinhalt - Kapow Katalyst

Mapplets, die Ports verketteten

Der Exportprozess scheitert, falls Sie ein Mapplet exportieren, das eine Multigroup-Eingabenumwandlung enthält, und die Ports in verschiedenen Eingabegruppen mit derselben nachfolgenden Umwandlung oder Transformationsausgabengruppe verbunden sind.

Verschachtelte Mapplets mit nicht verbundenen Lookup-Transformationen

Der Exportprozess scheitert, falls Sie einen Typ von Mapping oder Mapplet exportieren, der ein anderes Mapplet mit einer nicht verbundenen Lookup-Transformation enthält.

Mappings mit einer SAP-Quelle

Wenn Sie ein Mapping mit einer SAP-Quelle exportieren, exportiert das Developer Tool das Mapping ohne die SAP-Quelle. Wenn Sie das Mapping in das PowerCenter-Repository exportieren, importiert der PowerCenter Client das Mapping ohne die Quelle. Das Ausgabefenster zeigt eine Meldung an, die angibt, dass das Mapping nicht gültig ist. Sie müssen per Hand die SAP-Quelle in PowerCenter erstellen und sie zum Mapping hinzufügen.

Regeln und Richtlinien für das Exportieren an PowerCenter

Aufgrund der Unterschiede zwischen dem Developer Tool und PowerCenter sind manche Developer Tool-Objekte womöglich nicht mit PowerCenter kompatibel.

Verwenden Sie die folgenden Regeln und Richtlinien, wenn Sie Objekte in PowerCenter exportieren.

Überprüfen Sie die PowerCenter-Version.

Stellen Sie sicher, dass die aus dem Developer-Tool zu exportierenden Objekte mit der PowerCenter-Zielfersion kompatibel sind.

Überprüfen Sie, ob die Objektnamen einmalig sind.

Falls Sie ein Objekt in ein PowerCenter-Repository exportieren, ersetzt der Exportprozess das PowerCenter-Objekt, falls es denselben Namen wie ein exportiertes Objekt hat.

Überprüfen Sie, ob die Codepages kompatibel sind.

Der Exportprozess scheitert, falls das Developer Tool und PowerCenter Codepages verwenden, die nicht miteinander kompatibel sind.

Überprüfen Sie den Präzisionsmodus.

Standardmäßig ist im Developer Tool beim Ausführen von Mappings und Mapplets hohe Präzision aktiviert, während in PowerCenter hohe Präzision beim Ausführen von Sitzungen deaktiviert ist. Falls Sie Developer Tool-Mappings und PowerCenter-Sitzungen in verschiedenen Präzisionsmodi ausführen, können Sie verschiedene Ergebnisse erzeugen. Um verschiedene Ergebnisse zu vermeiden, führen Sie die Objekte in demselben Präzisionsmodus aus.

Kopieren Sie Referenzdaten.

Wenn Sie Mappings oder Mapplets mit Umwandlungen exportieren, die Referenztabelle verwenden, müssen Sie die Referenztabelle in ein Verzeichnis kopieren, in dem der PowerCenter Integration Service auf sie zugreifen kann. Kopieren Sie die Referenztabelle in das in der Umgebungsvariablen INFA_CONTENT bestimmte Verzeichnis. Falls INFA_CONTENT nicht eingestellt ist, kopieren Sie die Referenztabelle in das folgende PowerCenter-Dienst-Verzeichnis:

```
$INFA_HOME\services\<Developer Tool-Projektnamen>\<Developer Tool-Ordnername>
```

Fehlerbehebung - Exportieren an PowerCenter

Der Exportprozess scheitert, wenn ich ein Mapplet exportiere, das Objekte mit langen Namen enthält.

Wenn Sie ein Mapplet oder ein Mapping als Mapplet exportieren, erstellt der Exportprozess Objekte im Mapplet oder benennt sie um. Der Exportprozess kann Ausdrücke oder Ausgabeumwandlungen in der Export-XML-Datei erstellen: Der Exportprozess benennt also Ein- und -Ausgabeumwandlungen sowie Mapplet-Ports um.

Um Namen für Ausdrucksumwandlungen zu erzeugen, hängt der Exportprozess Zeichen an den Ein- und Ausgabeumwandlungsnamen an. Falls Sie ein Mapplet exportieren und Ziele in Ausgabeumwandlungen konvertieren, kombiniert der Exportprozess den Mapplet-Instanz-Namen und den Zielnamen, um den

Ausgabeumwandlungsnamen zu erzeugen. Wenn der Exportprozess Eingabeumwandlungen, Ausgabeumwandlungen und Maplet-Ports umbenennt, hängt er Gruppennamen an die Objektnamen an.

Falls ein bestehendes Objekt einen langen Namen hat, kann das exportierte Objekt in der Export-XML-Datei oder im PowerCenter-Repository das Objektnamen-Limit von 80 Zeichen überschreiten. Wenn ein Objektname 80 Zeichen überschreitet, scheitert der Exportprozess mit einem internen Fehler.

Falls Sie ein Maplet exportieren und der Exportprozess einen internen Fehler zurückgibt, prüfen Sie die Namen der Eingabeumwandlungen, Ausgabeumwandlungen, Ziele und Ports. Falls die Namen lang sind, verkürzen Sie sie.

KAPITEL 5

Importieren aus PowerCenter

Dieses Kapitel umfasst die folgenden Themen:

- [Importieren aus PowerCenter - Übersicht, 54](#)
- [Überschreibungseigenschaften, 55](#)
- [Konfliktlösung, 55](#)
- [Importübersicht, 55](#)
- [Datentypkonvertierung, 56](#)
- [Umwandlungskonvertierung, 56](#)
- [Verbindungsparameter für das PowerCenter-Repository, 62](#)
- [Importieren eines Objekts aus PowerCenter, 62](#)
- [Importieren von Einschränkungen, 64](#)
- [Importieren von Leistung, 64](#)

Importieren aus PowerCenter - Übersicht

Sie können Objekte aus einem PowerCenter Repository in ein Modell-Repository importieren. Beim Importprozess werden PowerCenter Repository-Objekte validiert und in Modell-Repository-Objekte konvertiert und anschließend importiert.

Beim Importieren von Objekten aus PowerCenter wählen Sie die Objekte, die Sie importieren möchten, und den Ziel-Speicherort im Modellrepository aus. Der Importprozess stellt Optionen bereit, um Objektnamen-Konflikte während des Imports aufzulösen. Sie können auch Verbindungen im Modell-Repository den PowerCenter-Objekten zuweisen. Sie können eine einzelne Verbindung mehreren PowerCenter-Objekten gleichzeitig zuweisen. Nachdem der Importprozess abgeschlossen ist, können Sie die Importzusammenfassung anzeigen.

Sie müssen Metadata Converter auf dem Computer installieren, auf dem der PowerCenter Express-Client gehostet wird, um Objekte aus PowerCenter in PowerCenter Express zu importieren.

Überschreibungseigenschaften

Sie haben die Wahl, die Überschreibungseigenschaften von PowerCenter-Objekten während des Importprozesses zu bewahren oder zu ignorieren. Standardmäßig bewahrt der Importprozess die Überschreibungseigenschaften von PowerCenter-Objekten.

Wenn Sie Überschreibungseigenschaften bewahren, erstellt der Importprozess nicht wieder verwendbare Umwandlungen oder wieder verwendbare Datenobjekte für die PowerCenter-Objekte. Wenn ein PowerCenter-Mapping Quellen- und Zieleigenschaften überschreibt, erstellt der Importprozess ein Datenobjekt mit denselben Überschreibungseigenschaftswerten wie das PowerCenter-Mapping. Der Importprozess fügt eine Zahl an den Namen des PowerCenter-Objekts an und erstellt das Datenobjekt.

Konfliktlösung

Sie können Objektnamenskonflikte lösen, wenn Sie ein Objekt aus PowerCenter importieren und ein Objekt mit demselben Namen im Modellrepository bereits vorhanden ist.

Folgende Konfliktlösungsoptionen stehen zur Auswahl:

Objekt in Ziel umbenennen

Benennt das PowerCenter Repository-Objekt mit der Standardnamenskonvention um und importiert es anschließend. Die Standardkonfliktauflösung besteht darin, das Objekt umzubenennen.

Objekt in Ziel ersetzen

Ersetzt das Model Repository-Objekt mit dem PowerCenter Repository-Objekt.

Objekt in Ziel erneut verwenden

Verwendet das Objekt im Model Repository im Mapping erneut.

Importübersicht

Der Importprozess erstellt eine Importzusammenfassung, nachdem Sie die PowerCenter-Objekte in das Modellrepository importiert haben.

Sie können die Importzusammenfassung in einer Datei speichern, wenn es Konvertierungsfehler gibt. Die Importzusammenfassung umfasst den Status des Imports, die Anzahl der Objekte, die nicht konvertiert wurden, die Anzahl der Objekte, die nach dem Import ungültig sind sowie die Konvertierungsfehler. Sie können auch die Objekte nach dem Importieren im Developer-Tool validieren, um die Validierungsfehler anzuzeigen.

Datentypkonvertierung

Einige PowerCenter-Datentypen sind im Modell-Repository nicht gültig. Wenn Sie PowerCenter-Objekte mit ungültigen Datentypen importieren, werden diese während des Imports in gültige, vergleichbare Datentypen im Modell-Repository konvertiert.

Die folgende Tabelle listet die PowerCenter Repository-Datentypen auf, die während des Imports in den entsprechenden Modell-Repository-Datentyp konvertiert werden.

PowerCenter Repository-Datentyp	Modell-Repository-Datentyp
Real	Double
Small Int	Ganzzahl
Nstring	Zeichenfolge
Ntext	Text

Umwandlungskonvertierung

Der Importprozess wandelt PowerCenter-Umwandlungen basierend auf die Kompatibilität um. Einige Umwandlungen sind nicht mit dem Modellrepository kompatibel. Import von anderen mit Einschränkungen.

Die folgende Tabelle beschreibt die PowerCenter-Umwandlungen, die mit Einschränkungen importieren oder bei denen der Import fehlschlägt:

PowerCenter-Umwandlung	Importaktion
Aggregator	Importiert mit Einschränkungen.
Datenmaskierung	Der Import schlägt fehl.
Externes Verfahren	Der Import schlägt fehl.
HTTP	Der Import schlägt fehl.
Identitätsauflösung	Der Import schlägt fehl.
Java	Importiert mit Einschränkungen.
Joiner	Importiert mit Einschränkungen.
Lookup	Importiert mit Einschränkungen.
Normalisierer	Der Import schlägt fehl.
Rang	Importiert mit Einschränkungen.
Sequenzgenerator	Der Import schlägt fehl.

PowerCenter-Umwandlung	Importaktion
Sortierer	Importiert mit Einschränkungen.
Quellqualifikator	Importiert mit Einschränkungen. Eine Quell- und Quellqualifikator-Umwandlung importiert vollständig als ein Datenobjekt aus.
Gespeicherte Prozedur	Der Import schlägt fehl.
Transaktionssteuerung	Der Import schlägt fehl.
SQL	Importiert mit Einschränkungen.
Union	Importiert mit Einschränkungen.
Unstrukturierte Daten	Der Import schlägt fehl.
Update-Strategie	Der Import schlägt fehl.
XML-Parser	Der Import schlägt fehl.
XML-Generator	Der Import schlägt fehl.

Einschränkungen für Umwandlungseigenschaften

Einige PowerCenter-Umwandlungen importieren mit Einschränkungen basierend auf den Umwandlungseigenschaften.

Der Importprozess kann eine der folgenden Aktionen basierend auf der Kompatibilität bestimmter Umwandlungseigenschaften ausführen:

- Ignore. Ignoriert die Umwandlungseigenschaft und importiert das Objekt.
- Intern konvertieren. Importiert das Objekt mit der Umwandlungseigenschaft, das Developer-Tool zeigt die Eigenschaft jedoch nicht an.
- Import fehlschlagen. Der Import des Objekts schlägt fehl und das Mapping ist nicht gültig.

Aggregatorumwandlung

Die folgende Tabelle beschreibt die Importaktion für Aggregator-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Umwandlungsbereich	Ignore.

Java-Umwandlung

In einer Java-Umwandlung müssen die Ports Eingabeports oder Ausgabeports sein. Der Import fehlt, wenn die Java-Umwandlung sowohl Eingabeports als auch Ausgabeports aufweist.

Die folgende Tabelle beschreibt die Importaktion für Java-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Klassenname	Ignore.
Funktionsbezeichner	Ignore.
Transaktion generieren	Ignore.
Eingaben müssen blockieren	Ignore.
Ist partitionierbar	Ignore.
Sprache	Ignore.
Modulbezeichner	Ignore.
Ausgabe ist deterministisch	Ignore.
Ausgabe ist wiederholbar	Ignore.
Benötigt einen einzelnen Thread pro Partition	Ignore.
Laufzeit-Speicherort	Ignore.
Update-Strategie-Umwandlung	Ignore.

Joiner-Umwandlung

Die folgende Tabelle beschreibt die Importaktion für Joiner-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Null-Ordnungsfolge in Master	Intern konvertieren.
Null-Ordnungsfolge im Detail	Intern konvertieren.
Umwandlungsbereich	Intern konvertieren.

Lookup-Umwandlung

Die folgende Tabelle beschreibt die Importaktion für Lookup-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Cache-Dateinamenpräfix	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Dynamischer Lookup-Cache	Import fehlschlagen, wenn auf "Ja" eingestellt. Ignorieren, wenn auf "Nein" eingestellt.
Einfügen, andernfalls aktualisieren	Ignore.

Umwandlungseigenschaft	Importaktion
Lookup-Cache-Initialisierung	Ignore.
Lookup-Cache-Verzeichnisname	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Lookup-Caching ist aktiviert	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Lookup-Daten-Cachegröße	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Lookup-Index-Cachegröße	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Lookup-Quelle ist statisch	Ignore.
Lookup-SQL-Override	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings in eine benutzerdefinierte SQL-Abfrage importiert.
Lookup-Quellfilter	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Bei Aktualisierung alten Wert ausgeben	Ignore.
Zuvor erstellter Lookup-Cache	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Erneut von Lookup-Quelle zwischenspeichern	Wird bei einer Konvertierung als eigenständige Umwandlung ignoriert und bei einer Konvertierung innerhalb eines Mappings importiert.
Erneut zwischenspeichern, wenn veraltet	Ignore.
Subsekundenpräzision	Ignore.
Dynamischen Cache synchronisieren	Ignore.
Dynamische Cache-Bedingung aktualisieren	Ignore.
Update sonst Insert	Ignore.

Rangumwandlung

Die folgende Tabelle beschreibt die Importaktion für Rangumwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Umwandlungsbereich	Ignore.

Sortiererumwandlung

Die folgende Tabelle beschreibt die Importaktion für Sortierer-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Umwandlungsbereich	Ignore.

Die Quellqualifikator-Umwandlung

Die folgende Tabelle beschreibt die importieren Aktion für die Quellqualifikator-Umwandlung Eigenschaften:

Umwandlungseigenschaft	Importaktion
Anzahl der sortierten Ports	Ignore.

SQL-Umwandlung

Die folgende Tabelle beschreibt die Importaktion für SQL-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Auto Commit	Ignore.
Klassenname	Ignore.
Verbindungstyp	Import fehlschlagen, wenn das dynamische Verbindungsobjekt oder vollständige dynamische Verbindungsinformationen festgelegt sind.
Datenbanktyp	Import für Sybase, Informix oder Teradata fehlgeschlagen.
Funktionsbezeichner	Ignore.
Transaktion generieren	Ignore.
Eingaben müssen blockieren	Ignore.
Ist partitionierbar	Ignore.
Sprache	Ignore.
Maximaler Verbindungspool	Ignore.
Modulbezeichner	Ignore.
Ausgabe ist deterministisch	Ignore.
Ausgabe ist wiederholbar	Ignore.
Benötigt einen einzelnen Thread pro Partition	Ignore.
Laufzeit-Speicherort	Ignore.

Umwandlungseigenschaft	Importaktion
SQL-Modus	Import fehlschlagen für Skriptmodus.
Umwandlungsbereich	Ignore.
DB-Verbindungsfehler als Schwerwiegend behandeln	Intern konvertieren.
Update-Strategie-Umwandlung	Ignore.
Verbindungs-Pool verwenden	Ignore.

Union-Umwandlung

Die folgende Tabelle beschreibt die Importaktion für Union-Umwandlungseigenschaften:

Umwandlungseigenschaft	Importaktion
Klassenname	Ignore.
Funktionsbezeichner	Ignore.
Transaktion generieren	Ignore.
Eingaben müssen blockieren	Ignore.
Ist partitionierbar	Ignore.
Sprache	Ignore.
Modulbezeichner	Ignore.
Ausgabe ist deterministisch	Ignore.
Ausgabe ist wiederholbar	Ignore.
Benötigt einen einzelnen Thread pro Partition	Ignore.
Laufzeit-Speicherort	Ignore.
Umwandlungsbereich	Ignore.
Update-Strategie-Umwandlung	Ignore.

Verbindungsparameter für das PowerCenter-Repository

Beim Importieren von Objekten aus einem PowerCenter-Repository müssen Sie die Verbindungsparameter für das Repository angeben. Das Developer Tool verwendet die Importparameter, um eine Verbindung zum PowerCenter Repository herzustellen.

In der folgenden Tabelle werden die Importparameter beschrieben:

Parameter	Beschreibung
Hostname	Hostname des PowerCenter Domänengateways.
Portnummer	HTTP-Portnummer des PowerCenter Domänengateways.
Release-Nummer	PowerCenter-Release-Version.
Authentifizierungstyp	<p>Der Typ der Benutzerauthentifizierung, der zum Herstellen einer Verbindung zum PowerCenter-Repository benötigt wird. Wählen Sie einen der folgenden Werte aus: Kerberos Single Sign On, Nativ oder LDAP.</p> <p>Der Typ der Benutzerauthentifizierung, der zum Herstellen einer Verbindung zum PowerCenter-Repository benötigt wird. Wählen Sie einen der folgenden Werte aus: Nativ oder LDAP.</p> <p>Hinweis: Geben Sie den Benutzernamen und das Passwort ein, wenn als Authentifizierungstyp „Nativ“ oder „LDAP“ ausgewählt wurde.</p>
Benutzername	PowerCenter Repository-Benutzername.
Passwort	Passwort für den PowerCenter-Repository-Benutzernamen.
Sicherheitsdomäne	Handelt es sich beim Authentifizierungstyp um LDAP, geben Sie den Namen der LDAP-Sicherheitsdomäne ein. Geben Sie andernfalls „Nativ“ ein.
Repository-Name	Name des PowerCenter Repositories.
Codepage	Codepage des PowerCenter Repositories.

Importieren eines Objekts aus PowerCenter

Sie können Objekte aus einem PowerCenter Repository in ein Modell-Repository importieren.

Stellen Sie eine Verbindung zum Zielmodellrepository her, bevor Sie Objekte aus PowerCenter importieren.

1. Wählen Sie **Datei > Importieren** aus.
Das Dialogfeld **Import** wird angezeigt.
2. Wählen Sie **Informatica > PowerCenter** aus.
3. Klicken Sie auf **Weiter**.
Das Dialogfeld **Aus PowerCenter importieren** wird angezeigt.
4. Geben Sie die Verbindungsparameter für das PowerCenter-Repository ein.

5. Klicken Sie auf **Verbindung testen**.
Das Developer Tool testet die Verbindung zum PowerCenter Repository.
6. Wenn die Verbindung zum PowerCenter-Repository erfolgreich ist, klicken Sie auf **OK**. Klicken Sie auf **Weiter**.
Das Developer Tool zeigt die Ordner im PowerCenter-Repository an und fordert Sie auf, die zu importierenden Objekte auszuwählen.
7. Wählen Sie mindestens ein zu importierendes Objekt aus.
8. Klicken Sie auf **Weiter**.
9. Wählen Sie einen Zielspeicherort für die Importobjekte im Modellrepository aus.
10. Wählen Sie eine Konfliktlösungsoption für Objektnamenskonflikte aus. Treffen Sie eine Auswahl, um die Objekte im Zielmodellrepository umzubenennen, zu ersetzen oder erneut zu verwenden.
 - Wählen Sie zum Umbenennen des PowerCenter-Repository-Objekts mithilfe der Standardnamenskonvention und anschließend Importieren in das Modellrepository die Option **Objekt in Ziel umbenennen** aus. Die standardmäßige Konfliktlösung besteht darin, das Objekt umzubenennen.
 - Wählen Sie zum Ersetzen des Modellrepository-Objekts durch das PowerCenter-Repository-Objekt die Option **Objekt in Ziel ersetzen** aus.
 - Um das Modellrepository-Objekt im Mapping erneut zu verwenden, statt es in ein PowerCenter-Objekt zu importieren, wählen Sie die Option **Objekt in Ziel erneut verwenden** aus.
11. Klicken Sie auf **Weiter**.
Das Developer Tool zeigt die PowerCenter-Objekte und die abhängigen Objekte an.
12. Klicken Sie auf **Überschriebene Eigenschaften ignorieren**, um Überschreibungseigenschaften für wiederverwendbare Quellen, Ziele und Umwandlungen in PowerCenter zu ignorieren. Standardmäßig behält der Prozess Überschreibungseigenschaften bei.
13. Wenn Sie ein IBM DB2-Objekt importieren, wählen Sie den DB2-Objektyp aus. Sie können einen der folgenden Objektypen auswählen: LOW, z/OS, i5/OS.
14. Klicken Sie auf **Weiter**.
15. Geben Sie die Verbindungsdetails des Modellrepositorys für die PowerCenter-Repository-Objekte an. Verwenden Sie eine der folgenden Methoden, um PowerCenter-Objekten Verbindungen im Modellrepository zuzuweisen.
 - Weisen Sie eine einzelne Verbindung mehreren PowerCenter-Objekten gleichzeitig zu. Eine einzelne Verbindung können Sie allen Quellen, allen Zielen, allen Lookup-Umwandlungen oder allen Objekten zuweisen, die nicht über eine zugewiesene Verbindung verfügen. Wahlweise können Sie eine einzelne Verbindung zu allen Objekten mit Namen zuweisen, die einem angegebenen Namensmuster entsprechen. Wählen Sie eine Option in der Liste **Auswählen** aus und klicken Sie auf **Verbindung zuweisen**.
 - Weisen Sie mehreren PowerCenter-Objekten mit verschiedenen Objektypen eine einzelne Verbindung zu. Wählen Sie die Option **Benutzerdefiniert** in der Liste **Auswählen** aus, markieren Sie mehrere PowerCenter-Objekte und klicken Sie dann auf **Verbindung zuweisen**.
 - Weisen Sie einem PowerCenter-Objekt eine Verbindung zu. Wählen Sie ein PowerCenter-Objekt aus und klicken Sie in der Spalte **Verbindungsname** auf die Schaltfläche **Öffnen**.
16. Das Dialogfeld **Verbindung auswählen** wird geöffnet. Wählen Sie eine Verbindung aus und klicken Sie auf **OK**.
17. Klicken Sie auf **Weiter**.

Das Developer-Tool erzeugt eine Importzusammenfassung und listet die PowerCenter-Objekte sowie die abhängigen zu importierenden Objekte auf.

18. Klicken Sie auf **Umwandlungsprüfung**, um sicherzustellen, dass Objekte als gültige Modellrepository-Objekte importiert werden können.

Das Developer Tool zeigt einen Zusammenfassungsbericht über die Umwandlungsprüfung mit den Ergebnissen der Umwandlungsprüfung an.

19. Klicken Sie auf **OK**. Klicken Sie auf **Fertigstellen**.

Das Developer Tool zeigt die Fortschrittsinformationen während des Imports an. Das Developer Tool importiert die PowerCenter-Objekte und die abhängigen Objekte in das Modell-Repository und generiert einen abschließenden Zusammenfassungsbericht über den Import.

20. Klicken Sie auf **Speichern** und geben Sie einen Dateinamen zum Speichern der Importzusammenfassung an, wenn Konvertierungsfehler vorliegen.

Importieren von Einschränkungen

Die folgenden Einschränkungen gelten beim Importieren von PowerCenter-Objekten:

Quelle und Ziel

- Beim Importieren einer Quelle oder eines Ziels aus PowerCenter Version 9.1.0 oder früher kann der Importprozess nicht überprüfen, ob ein dem Objekt zugeordneter Verbindungstyp gültig ist.
- Wenn die Version des PowerCenter-Repository älter als 9.5.0 ist, muss der Name einer IBM DB2-Quelldatenbank oder der Name eines IBM DB2-Ziels mit "DB2" beginnen, um den DB2-Typ festzulegen.
- Wenn der Zeilen-Delimiter für eine Einfachdateiquelle nicht gültig ist, ändert ihn der Importprozess auf den Standardwert.

Umwandlung

- Ein Ausdruck in einer Umwandlung muss 4.000 oder weniger Zeichen enthalten.
- Der Datenbanktyp für eine SQL-Umwandlung oder eine Lookup-Umwandlung konvertiert zu ODBC während des Importprozesses.
- Wenn Sie die Datencachegröße oder die Indexcachegröße für eine Umwandlung auf einen Wert festlegen, der nicht gültig ist, ändert der Importprozess den Wert auf Auto.

Mapping

- Ein Mapping darf nur eine Pipeline enthalten.

Importieren von Leistung

Wenn Sie Mappings importieren möchten, die größer als 68 MB sind, importieren Sie das Mapping über die Befehlszeile für optimale Leistung.

Tipp: Sie können die folgenden Befehlszeilenoption verwenden: `ImportFromPC`

KAPITEL 6

Leistungsoptimierung

Dieses Kapitel umfasst die folgenden Themen:

- [Leistungsoptimierung - Übersicht, 65](#)
- [Optimierungsebenen, 66](#)
- [Optimierungsmethoden - Übersicht, 66](#)
- [Vollständige Optimierung und Speicherzuordnung, 69](#)
- [Einstellen der Optimierungsebene für ein Developer Tool-Mapping, 70](#)
- [Einstellen der Optimierungsebene für ein bereitgestelltes Mapping, 70](#)

Leistungsoptimierung - Übersicht

Der Datenintegrationsdienst wendet Optimierungsmethoden zum Verbessern der Leistung eines Mappings an. Wenn Sie ein Mapping ausführen, können Sie eine Optimierungsebene zur Angabe der Optimierungsmethoden auswählen, die vom Datenintegrationsdienst auf das Mapping angewendet werden können.

Wenn der Datenintegrationsdienst ein Mapping optimiert, versucht er, die Menge der zu verarbeitenden Daten zu verringern. Beispielsweise kann der Datenintegrationsdienst mithilfe der frühen Auswahloptimierung einen Filter näher an die Quelle verschieben. Mit der Pushdown-Optimierung kann Umwandlungslogik in eine Datenbank übertragen werden. Mit der kostenbasierten Optimierungsmethode kann die Join-Verarbeitungsreihenfolge geändert werden.

Der Datenintegrationsdienst kann gleichzeitig mehrere Optimierungsmethoden auf ein Mapping anwenden. Der Datenintegrationsdienst wendet beispielsweise folgende Optimierungsmethoden an, wenn Sie die Optimierungsebene „Normal“ auswählen: frühe Projektion, Vorhersage, frühe Auswahl, Verzweigungsoptimierung oder Push-Into.

Sie können Einschränkungen für relationale Quellen, logische Datenobjekte physische Datenobjekte und virtuelle Tabellen in einem Mapping einrichten, um nicht benötigte Zeilen zu filtern. Der Datenintegrationsdienst kann Einschränkungen verarbeiten, um die Mapping-Leistung verbessern.

Optimierungsebenen

Der Data Integration Service versucht, verschiedene Optimierer-Methoden anzuwenden, basierend auf der Optimierer-Ebene, die Sie für das Objekt konfigurieren.

Sie können eine der folgenden Optimierungsebenen auswählen:

Keiner

Der Data Integration Service wendet keine Optimierung an.

Minimal

Der Data Integration Service wendet die frühe Projektionsoptimierungsmethode an.

Normal

Der Datenintegrationsdienst wendet folgende Optimierungsmethoden an: frühe Projektion, frühe Auswahl, Verzweigungsreinigung, Push-Into, Pushdown und Prädikat. Normal ist die Standard-Optimierungsebene.

Komplett

Der Datenintegrationsdienst wendet folgende Optimierungsmethoden an: kostenbasiert, frühe Projektion, frühe Auswahl, Verzweigungsreinigung, Prädikat, Push-Into, Pushdown und Semi-Join.

VERWANDTE THEMEN:

- [“Die Pushdown-Optimierung - Übersicht” auf Seite 71](#)

Optimierungsmethoden - Übersicht

Der Data Integration Service wendet Optimierungsmethoden an, um die Anzahl der im Mapping zu verarbeitenden Zeilen zu verringern. Sie können die Optimierungsebene für das Mapping konfigurieren, um zu beschränken, welche Optimierungsmethoden der Data Integration Service anwendet.

Der Data Integration Service kann die folgenden Optimierungsmethoden anwenden:

- Pushdown-Optimierung
- Frühe Projektion
- Frühe Auswahl
- Verzweigungsreinigung
- Push-Into-Optimierung
- Vorhersageoptimierung
- Kostenbasiert
- Semi-Join

Der Data Integration Service kann gleichzeitig mehrere Optimierungsmethoden auf ein Mapping anwenden. Der Datenintegrationsdienst wendet beispielsweise die Methoden zur frühen Projektionsoptimierung, zur Vorhersageoptimierung, zur frühen Auswahloptimierung oder zur Push-Into-Optimierung an, wenn Sie die Optimierungsebene „Normal“ auswählen.

Frühe Projektionsoptimierungsmethode

Wenn der Data Integration Service die frühe Projektionsoptimierungsmethode anwendet, identifiziert er ungenutzte Ports und entfernt die Links zwischen diesen Ports.

Die frühe Projektion verbessert die Leistung, indem die Menge der Daten, die vom Data Integration Service zwischen Umwandlungen verschoben werden, verringert wird. Wenn der Data Integration Service ein Mapping verarbeitet, verschiebt er die Daten aus allen verbundenen Ports in einem Mapping von einer Umwandlung zu einer anderen Umwandlung. In großen komplexen Mappings oder in Mappings mit geschachtelten Mapplets stellen einige Ports möglicherweise keine Daten für das Target bereit. Der Data Integration Service identifiziert die Ports, die keine Daten für das Ziel bereitstellen. Nachdem der Data Integration Service ungenutzte Ports identifiziert hat, entfernt er die Links zwischen allen ungenutzten Ports aus dem Mapping.

Der Data Integration Service entfernt nicht alle Links. Beispielsweise entfernt er nicht die folgenden Links:

- Mit einer Umwandlung verbundene Verknüpfungen mit Nebeneffekten.
- Links, die mit Umwandlungen verbunden sind, die eine ABORT()- oder ERROR()-Funktion aufrufen, E-Mails senden oder eine gespeicherte Prozedur aufrufen.

Wenn der Data Integration Service bestimmt, dass alle Ports in einer Umwandlung nicht genutzt werden, entfernt er alle Umwandlungslinks außer dem Link zu dem Port mit den wenigsten Daten. Der Data Integration Service entfernt nicht die ungenutzte Umwandlung aus dem Mapping.

Diese Optimierungsmethode wird vom Developer Tool standardmäßig aktiviert.

Frühe Auswahloptimierungsmethode

Wenn der Data Integration Service die frühe Auswahloptimierungsmethode anwendet, werden die Filter-Umwandlungen in einem Mapping aufgeteilt, verschoben oder entfernt. Filter werden im Mapping aufwärts näher zur Quelle verschoben.

Der Data Integration Service kann eine Filter-Umwandlung aufteilen, wenn die Filterbedingung eine Konjunktion ist. Der Data Integration Service kann beispielsweise die Filterbedingung „A>100 AND B<50“ in die beiden einfacheren Bedingungen „A>100“ und „B<50“ aufteilen. Wenn der Data Integration Service einen Filter aufteilt, verschiebt er die vereinfachten Filter in der Mapping-Pipeline aufwärts näher zur Quelle. Der Data Integration Service verschiebt die Filter getrennt in der Pipeline aufwärts, wenn er den Filter aufteilt.

Das Developer-Tool aktiviert standardmäßig die frühe Auswahloptimierungsmethode, wenn Sie eine normale oder vollständige Optimierungsebene auswählen. Der Data Integration Service aktiviert nicht die frühe Auswahl, wenn eine Umwandlung, die vor der Filterumwandlung erscheint, Nebenwirkungen aufweist. Sie können die SQL-Umwandlung, die Webdienstkunden-Umwandlung und die Java-Umwandlung für die frühe Auswahloptimierung konfigurieren, jedoch kann das Developer-Tool nicht erkennen, ob die Umwandlungen Nebeneffekte aufweisen.

Sie können die frühe Auswahloptimierung deaktivieren, wenn sie die Leistung nicht erhöht.

Methode zur Optimierung der Verzweigungsbereinigung

Der Datenintegrationsdienst kann die Methode zur Optimierung der Verzweigungsbereinigung auf Umwandlungen anwenden, die keine Zeilen zum Ziel in einem Mapping beitragen.

Der Datenintegrationsdienst kann eine Filterumwandlung unter Umständen entfernen, wenn die Filterbedingung für die Datenzeilen mit FALSE bewertet wird. Beispiel: Ein Mapping besteht aus zwei Filterumwandlungen, die Daten aus zwei relationalen Quellen filtern. Eine Filterumwandlung weist die Filterbedingung Country=US und die andere Filterumwandlung die Filterbedingung Country=Canada auf. Eine Unionumwandlung verknüpft die beiden relationalen Quellen und weist die Filterbedingung Country=US auf.

Der Datenintegrationsdienst entfernt die Filterumwandlung mit der Filterbedingung `Country=Canada` unter Umständen aus dem Mapping.

Das Developer-Tool aktiviert standardmäßig die Methode zur Optimierung der Verzweigungsreinigung, wenn Sie eine normale oder vollständige Optimierungsebene auswählen. Sie können die Verzweigungsreinigung deaktivieren, wenn die Leistung durch die Optimierung nicht erhöht wird, indem Sie die Optimierungsebene auf „Minimal“ oder „Keine“ setzen.

Vorhersageoptimierungsmethode

Wenn der Data Integration Service die Vorhersageoptimierungsmethode anwendet, überprüft er die von einem Mapping generierten Vorhersage-Expressionen. Er bestimmt, ob er die Expressionen vereinfachen oder umschreiben kann, um die Mapping-Leistung zu erhöhen.

Wenn der Data Integration Service ein Mapping ausführt, generiert er Abfragen der Mapping-Quellen und führt basierend auf der Mapping-Logik und den Umwandlungen im Mapping Operationen für die Abfrageergebnisse aus. Die Abfragen und Operationen enthalten häufig Vorhersage-Expressionen. Vorhersage-Expressionen stellen die Bedingungen dar, die von den Daten erfüllt werden müssen. Die Filter- und die Join-Bedingung in der Filter- bzw. Joiner-Umwandlung sind Beispiele für Vorhersage-Expressionen.

Der Data Integration Service versucht mit der Vorhersageoptimierungsmethode außerdem, zu einem möglichst frühen Zeitpunkt im Mapping Vorhersage-Expressionen anzuwenden, um die Mapping-Leistung zu erhöhen.

Der Data Integration Service leitet aus vorhandenen Vorhersage-Expressionen Beziehungen ab und erstellt neue Vorhersage-Expressionen. Angenommen, ein Mapping enthält eine Joiner-Umwandlung mit der Join-Bedingung „`A=B`“ und eine Filter-Umwandlung mit der Filterbedingung „`A>B`“. Der Data Integration Service kann der Join-Bedingung „`B>5`“ hinzufügen.

Der Data Integration Service wendet die Vorhersageoptimierungsmethode mit der frühen Auswahloptimierungsmethode an, wenn er beide Methoden auf ein Mapping anwenden kann. Wenn der Data Integration Service beispielsweise neue Filterbedingungen über die Vorhersageoptimierungsmethode erstellt, versucht er außerdem, sie mit der frühen Auswahloptimierungsmethode im Mapping aufwärts zu verschieben. Die Anwendung beider Optimierungsmethoden bietet eine bessere Mapping-Leistung als die Anwendung nur einer dieser Methoden.

Der Data Integration Service wendet die Vorhersageoptimierungsmethode an, wenn dies die Leistung erhöht. Er wendet diese Methode nicht an, wenn die Anwendung die Mapping-Ergebnisse ändert oder die Mapping-Leistung verringert.

Kostenbasierte Optimierungsmethode

Mit kostenbasierter Optimierung evaluiert der Data Integration Service ein Mapping, generiert semantisch äquivalente Mappings und führt das Mapping mit der besten Leistung aus. Kostenbasierte Optimierung verringert die Laufzeit von Mappings, die benachbarte unsortierte Inner-Join-Vorgänge ausführen.

Semantisch äquivalente Mappings führen identische Funktionen aus und erzeugen die gleichen Ergebnisse. Zum Generieren von semantisch äquivalenten Mappings unterteilt der Data Integration Service das ursprüngliche Mapping in Fragmente. Anschließend bestimmt der Data Integration Service, welche Mapping-Fragmente optimiert werden können.

Der Data Integration Service optimiert jedes Fragment, das optimiert werden kann. Der Data Integration Service kann während der Optimierung Umwandlungen in einem Fragment hinzufügen, entfernen oder neu anordnen. Der Data Integration Service überprüft, ob die optimierten Fragmente die gleichen Ergebnisse wie die ursprünglichen Fragmente erzeugen, und bildet alternative Mappings, die die optimierten Fragmente nutzen.

Der Data Integration Service generiert alle oder fast alle Mappings, die mit dem ursprünglichen Mapping semantisch äquivalent sind. Er verwendet Profiling oder Datenbankstatistik, um die Kosten für das Original-Mapping und jedes alternative Mapping zu berechnen. Anschließend ermittelt er das Mapping, das am schnellsten ausgeführt wird. Der Data Integration Service führt eine Validierungsprüfung zum Ermitteln des besten alternativen Mappings aus, um sicherzustellen, dass es gültig ist und die gleichen Ergebnisse wie das ursprüngliche Mapping erzeugt.

Der Data Integration Service zwischenspeichert das beste alternative Mapping. Wenn Sie ein Mapping ausführen, ruft der Data Integration Service das alternative Mapping ab und führt es anstelle des ursprünglichen Mappings aus.

Semi-Join Optimierungsmethode

Die Semi-Join Optimierungsmethode ändert die Join-Operationen im Mapping, um die Menge der aus der Quelle extrahierten Daten zu verringern.

Der Data Integration Service wendet diese Methode auf eine Joiner-Umwandlung an, wenn eine Eingabegruppe über mehr Zeilen als die andere Gruppe verfügt und wenn die größere Gruppen viele Zeilen enthält, für die in der kleineren Gruppen keine Übereinstimmung gemäß der Join-Bedingung vorhanden ist. Der Data Integration Service 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 Data Integration Service nicht mehr unnötige Zeilen aus der größeren Quellgruppe liest. Der Data Integration Service verschiebt die Join-Bedingung in die größere Quellgruppe und liest nur die Zeilen, die mit der kleineren Gruppe übereinstimmen.

Bevor der Data Integration Service diese Optimierungsmethoden anwendet, führt er Analysen aus, um zu bestimmen, ob eine Semi-Join-Optimierung möglich ist und den Aufwand lohnt. Wenn die Analysen ergeben, dass die Leistung durch diese Methode wahrscheinlich erhöht wird, wird sie vom Data Integration Service auf das Mapping angewendet. Anschließend analysiert der Data Integration Service das Mapping erneut, um zu bestimmen, ob weitere Möglichkeiten für Semi-Join Optimierung bestehen. Ggf. führt er weitere Optimierungen aus. Der Data Integration Service führt nur dann eine Semi-Join Optimierung aus, wenn die Analysen ergeben, dass eine hohe Wahrscheinlichkeit für eine Leistungsverbesserung besteht.

Im Developer Tool wird diese Methode nicht standardmäßig aktiviert.

Vollständige Optimierung und Speicherzuordnung

Beim Konfigurieren einer vollständigen Optimierung für ein Mapping müssen Sie eventuell den freien Arbeitsspeicher erhöhen, um Mapping-Fehler zu vermeiden.

Enthält ein Mapping Joiner-Umwandlungen und andere Umwandlungen, die Caching verwenden, kann das Mapping womöglich mit dem Standardoptimierungslevel fehlerfrei ablaufen. Falls Sie das Optimierungslevel auf vollständige Optimierung stellen und der Data Integration Service Semi-Join Optimierung durchführt, benötigt der Data Integration Service mehr Speicher, um die Daten zu sortieren. Das Mapping kann scheitern, falls Sie die maximale Sitzungsgröße nicht erhöhen.

Ändern Sie die **Maximale Sitzungsgröße** in den **Ausführungsoptionen** für den Data Integration Service-Prozess. Erhöhen Sie die **Maximale Sitzungsgröße** um 50 MB bis 100 MB.

Einstellen der Optimierungsebene für ein Developer Tool-Mapping

Wenn Sie ein Mapping über das Ausführen-Menü oder vom Mapping-Editor aus starten, führt das Developer Tool das Mapping mit der normalen Optimierungsebene aus. Um das Mapping mit einer anderen Optimierungsebene auszuführen, starten Sie das Mapping im Dialogfeld **Konfigurationen ausführen**.

1. Öffnen Sie das Mapping.
2. Wählen Sie **Ausführen > Ausführungsdialog öffnen**.
Das Dialogfeld **Konfigurationen ausführen** wird eingeblendet.
3. Wählen Sie eine Mapping-Konfiguration mit der gewünschten Optimierungsebene oder erstellen Sie eine neue Mapping-Konfiguration.
4. Klicken Sie auf die Registerkarte **Erweitert**.
5. Ändern Sie, wenn nötig, die Optimierungsebene.
6. Klicken Sie auf **Anwenden**.
7. Klicken Sie auf **Ausführen**, um das Mapping zu starten.

Das Developer Tool führt das Mapping mit der Optimierungsebene in der ausgewählten Mapping-Konfiguration aus.

Einstellen der Optimierungsebene für ein bereitgestelltes Mapping

Stellen Sie die Optimierungsebene für ein von der Befehlszeile aus gestartetes Mapping ein, indem Sie die Eigenschaften für die Mapping-Bereitstellung in der Anwendung ändern.

Das Mapping muss in einer Anwendung sein.

1. Öffnen Sie die Anwendung, die das Mapping enthält.
2. Klicken Sie auf die Registerkarte **Erweitert**.
3. Wählen Sie die Optimierungsebene.
4. Speichern Sie die Anwendung.

Nachdem Sie die Optimierungsebene geändert haben, müssen Sie die Anwendung erneut bereitstellen.

KAPITEL 7

Pushdown-Optimierung

Dieses Kapitel umfasst die folgenden Themen:

- [Die Pushdown-Optimierung - Übersicht, 71](#)
- [Umwandlungslogik, 72](#)
- [Pushdown-Optimierung zu Quellen, 72](#)
- [Pushdown-Optimierungsausdrücke, 77](#)
- [Vergleichen der Ausgabe vom Data Integration Service und von Quellen, 91](#)

Die Pushdown-Optimierung - Übersicht

Die Pushdown-Optimierung veranlasst den Data Integration Service, die Umwandlungslogik in die Quelldatenbank zu verschieben. Der Data Integration Service übersetzt die Umwandlungslogik in SQL-Abfragen und sendet die SQL-Abfragen an die Datenbank. Die Quelldatenbank führt die SQL-Abfragen durch, um die Umwandlungen zu verarbeiten.

Die Pushdown-Optimierung verbessert die Leistung von Mappings, wenn die Quelldatenbank die Umwandlungslogik schneller als der Data Integration Service verarbeiten kann. Der Data Integration Service liest auch weniger Daten aus der Quelle.

Der Data Integration Service wendet die Pushdown-Optimierung auf ein Mapping an, wenn Sie die normale oder vollständige Optimierungsebene auswählen. Wenn Sie die normale Optimierungsebene auswählen, wendet der Data Integration Service die Pushdown-Optimierung an, nachdem er alle anderen Optimierungsmethoden angewendet hat. Wenn Sie die vollständige Optimierungsebene auswählen, wendet der Data Integration Service die Pushdown-Optimierung vor der Semi-Join-Optimierung aber nach allen anderen Optimierungsmethoden an.

Wenn Sie die Pushdown-Optimierung anwenden, analysiert der Datenintegrationsdienst das optimierte Mapping von der Quelle zum Ziel oder so lange, bis er eine nachgelagerte Umwandlung erreicht, die er nicht in die Quelldatenbank verschieben kann.

Der Data Integration Service generiert und führt eine SELECT-Anweisung basierend auf der Umwandlungslogik für jede Umwandlung auf, die er in die Datenbank verschieben kann. Wenn die SQL-Abfrage Zeilenfehler erzeugt, schlägt das Mapping fehl. Der Datenintegrationsdienst liest die Ergebnisse der SQL-Abfrage und verarbeitet die übrigen Umwandlungen im Mapping.

VERWANDTE THEMEN:

- [“Optimierungsebenen” auf Seite 66](#)

Umwandlungslogik

Der Data Integration Service verwendet die Pushdown-Optimierung zum Verschieben der Umwandlungslogik in die Quelldatenbank. Die Anzahl der Umwandlungslogik, die der Data Integration Service in die Quelldatenbank verschiebt, richtet sich nach der Konfiguration der Datenbank, der Umwandlungslogik und des Mappings. Der Data Integration Service verarbeitet die gesamte Umwandlungslogik, die er nicht in die Datenbank verschieben kann.

Der Data Integration Service kann die folgende Umwandlungslogik in die Quelldatenbank verschieben:

- Aggregator
- Ausdruck
- Filter
- Joiner
- Sortierer
- Union

Unter folgenden Bedingungen kann der Datenintegrationsdienst die Umwandlungslogik nicht in eine Quelle verschieben:

- Die Quelle enthält eine Spalte mit einem binären Datentyp.
- Die Quelle ist ein konfiguriertes Datenobjekt, das eine Filterbedingung oder ein benutzerdefiniertes Join für Ausdrucks- oder Joiner-Umwandlungslogik enthält.
- Die Quellen befinden sich auf verschiedenen Datenbankverwaltungssystemen oder verwenden unterschiedliche Verbindungen für die Joiner- oder Union-Umwandlungslogik.

Pushdown-Optimierung zu Quellen

Der Data Integration Service kann Umwandlungslogik in verschiedene Quellen verschieben, wie zum Beispiel relationale Quellen und Quellen, die ODBC-Treiber verwenden. Der Typ der Umwandlungslogik, die der Data Integration Service verschiebt, hängt vom Quelltyp ab.

Der Data Integration Service kann die Umwandlungslogik in die folgenden Quelltypen verschieben:

- Relationale Quellen
- Quellen, die native Datenbanktreiber verwenden
- Nichtrelationale PowerExchange-Quellen
- Quellen, die ODBC-Treiber verwenden
- SAP-Quellen

Pushdown-Optimierung für relationale Quellen

Der Data Integration Service kann mit nativen Treibern oder ODBC-Treibern Umwandlungslogik in relationale Quellen verschieben.

Der Datenintegrationsdienst kann Aggregator-, Ausdrucks-, Filter-, Joiner-, Sortierer- und Union-Umwandlungslogik in die folgenden relationalen Quellen verschieben:

- Greenplum
- Hive
- IBM DB2
- Microsoft SQL Server
- Oracle
- SAP HANA
- Sybase
- Teradata

Wenn Sie die Aggregator-Umwandlungslogik in eine relationale Quelle verschieben, sind Pass-Through-Ports gültig, wenn es Group-By-Ports sind. Die Umwandlungssprache umfasst Aggregatfunktionen, die Sie in einer Aggregat-Umwandlung verwenden können.

In der folgenden Tabelle werden die Aggregatfunktionen angezeigt, die in einer relationalen IBM DB2-Quelle gültig sind:

Aggregatfunktionen	DB2-LUW	DB2i	DB2z/os
AVG	Ja	Ja	Ja
COUNT	Ja	Ja	Ja
FIRST	Nein	Nein	Nein
LAST	Nein	Nein	Nein
MAX	Ja	Ja	Ja
MEDIAN	Nein	Nein	Nein
MIN	Ja	Ja	Ja
PERCENTILE	Nein	Nein	Nein
STDDEV	Ja	Ja	Ja
SUM	Ja	Ja	Ja
VARIANCE	Ja	Ja	Ja

In der folgenden Tabelle werden die Aggregatfunktionen angezeigt, die in relationalen Greenplum-, MSSQL-, Oracle-, Sybase- und Hive-Quellen gültig sind:

Aggregatfunktionen	Greenplum	Hive	MSSQL	Oracle	Sybase	Teradata
AVG	Ja	Ja	Ja	Ja	Ja	Ja
COUNT	Ja	Ja	Ja	Ja	Ja	Ja
FIRST	Nein	Nein	Nein	Nein	Nein	Nein
LAST	Nein	Nein	Nein	Nein	Nein	Nein
MAX	Ja	Ja	Ja	Ja	Ja	Ja
MEDIAN	Nein	Nein	Nein	Ja	Nein	Nein
MIN	Ja	Ja	Ja	Ja	Ja	Ja
PERCENTILE	Nein	Nein	Nein	Nein	Nein	Nein
STDDEV	Nein	Ja	Ja	Ja	Nein	Ja
SUM	Ja	Ja	Ja	Ja	Ja	Ja
VARIANCE	Nein	Ja	Ja	Ja	Nein	Ja

Die folgende Tabelle enthält die Aggregatfunktionen, die in relationalen MSSQL-, Oracle- und Sybase-Quellen gültig sind:

Aggregatfunktionen	MSSQL	Oracle	Sybase
AVG	Ja	Ja	Ja
COUNT	Ja	Ja	Ja
FIRST	Nein	Nein	Nein
LAST	Nein	Nein	Nein
MAX	Ja	Ja	Ja
MEDIAN	Nein	Ja	Nein
MIN	Ja	Ja	Ja
PERCENTILE	Nein	Nein	Nein
STDDEV	Ja	Ja	Nein
SUM	Ja	Ja	Ja
VARIANCE	Ja	Ja	Nein

Eine relationale Quelle hat eine Standardkonfiguration zu Behandlung von Nullwerten. Standardmäßig behandeln einige Datenbanken Nullwerte niedriger als einen anderen Wert und einige Datenbanken behandeln Nullwerte höher als einen anderen Wert. Sie können die Sortierer-Umwandlungslogik in die relationale Quelle verschieben und genaue Ergebnisse erhalten, wenn die Quelle die Standard-Null-Ordnungsfolge enthält.

Wenn Sie eine Sortierumwandlung für unterschiedliche Ausgabezeilen konfigurieren, müssen Sie eine Sortierung mit Groß-/Kleinschreibung aktivieren, um die Umwandlungslogik in die Quelle für DB2, Sybase und Oracle zu verschieben.

Der Datenintegrationsdienst kann keine Funktion verschieben, die den Datentyp „Decimal“ für eine Hive-Quelle enthält.

Pushdown-Optimierung für native Quellen

Wenn der Data Integration Service Umwandlungslogik mit nativen Treibern in relationale Quellen verschiebt, generiert der Data Integration Service SQL-Anweisungen, die die native Datenbank-SQL verwenden.

Der Datenintegrationsdienst kann Aggregat-, Ausdrucks-, Filter-, Joiner-, Sortierer- und Union-Umwandlungslogik in die folgenden nativen Quellen verschieben:

- IBM DB2 für Linux, UNIX und Windows ("DB2 für LUW")
- Microsoft SQL Server. Der Data Integration Service kann eine native Verbindung zu Microsoft SQL Server verwenden, wenn der Data Integration Service unter Windows ausgeführt wird.
- Oracle

Der Data Integration Service kann Filterumwandlungslogik in die folgenden nativen Quellen verschieben:

- IBM DB2 für i5/OS
- IBM DB2 für z/OS

Pushdown-Optimierung für nichtrelationale PowerExchange-Quellen

Für nichtrelationale PowerExchange-Datenquellen auf z/OS-Systemen verschiebt der Data Integration Service Filterumwandlungslogik nach PowerExchange. PowerExchange übersetzt die Logik in einer Abfrage, die die Quelle verarbeiten kann.

Der Data Integration Service kann die Filterumwandlungslogik für die folgenden nichtrelationalen Quelltypen verschieben:

- IBM IMS
- Sequentielle Datensätze
- VSAM

Pushdown-Optimierung für ODBC-Quellen

Der Data Integration Service kann Umwandlungslogik in Datenbanken verschieben, die ODBC-Treiber verwenden.

Wenn Sie ODBC für die Verbindung zu einer Quelle verwenden, kann der Data Integration Service kann SQL-Anweisungen mit ANSI SQL oder mit nativem Datenbank-SQL generieren. Der Data Integration Service kann mehr Umwandlungslogik in die Quelle verschieben, wenn er SQL-Anweisungen mit der nativen Datenbank-SQL generiert. Die Quelle kann natives Datenbank-SQL schneller als ANSI SQL verarbeiten.

Sie können den ODBC-Provider im ODBC-Verbindungsobjekt angeben. Wenn es sich um einen datenbankspezifischen ODBC-Provider handelt, kann der Data Integration Service SQL-Anweisungen mit nativer Datenbank-SQL generieren. Wenn es sich um einen ODBC-Provider vom Typ **Sonstige** handelt, kann der Data Integration Service SQL-Anweisungen mit ANSI SQL generieren.

Sie können einen bestimmten ODBC-Provider für die folgenden ODBC-Verbindungstypen konfigurieren:

- Microsoft SQL Server
Verwenden Sie eine ODBC-Verbindung, um eine Verbindung zu Microsoft SQL Server herzustellen, wenn der Data Integration Service unter UNIX oder Linux ausgeführt wird. Verwenden Sie eine native Verbindung zu Microsoft SQL Server, wenn der Data Integration Service unter Windows ausgeführt wird.
- Netezza
- Sybase ASE
- Teradata

Hinweis: Wenn Sie **Greenplum** für den ODBC-Provider angeben, behandelt der Datenintegrationsdienst den ODBC-Provider als **Andere** und verwendet ANSI SQL, um SQL-Anweisungen zu erzeugen.

Pushdown-Optimierung für SAP-Quellen

Der Data Integration Service kann die Filterumwandlungslogik in SAP-Quellen für die Ausdrücke verschieben, die einen Spaltennamen, einen Operator und ein Literal-String beinhalten. Wenn der Data Integration Service die Umwandlungslogik nach SAP verschiebt, konvertiert der Data Integration Service den Literal-String in den Ausdrücken in einen SAP-Datentyp.

Der Data Integration Service kann die Filterumwandlungslogik verschieben, die die TO_DATE-Funktion enthält, wenn TO_DATE ein DATS-, TIMS- oder ACCP-Datentypzeichen in eines der folgenden Datumsformate konvertiert:

- 'MM/DD/YYYY'
- 'YYYY/MM/DD'
- 'YYYY-MM-DD HH24:MI:SS'
- 'YYYY/MM/DD HH24:MI:SS'
- 'MM/DD/YYYY HH24:MI:SS'

Der Data Integration Service verarbeitet die Umwandlungslogik, wenn Sie die TO_DATE-Funktion auf einen Datentyp anwenden, bei dem es sich nicht um DATS, TIMS oder ACCP handelt, oder wenn TO_DATE einen Zeichenstring in ein Format konvertiert, das der Data Integration Services nicht nach SAP verschieben kann. Der Data Integration Service verarbeitet Umwandlungslogik, die andere Informatica-Funktionen enthält. Der Data Integration Service verarbeitet Umwandlungslogik, die andere Informatica-Funktionen enthält.

Die Filterumwandlungsausdrücke können mehrere Bedingungen getrennt durch AND oder OR beinhalten. Wenn Bedingungen auf mehrere SAP-Tabellen angewendet werden können, kann der Data Integration Service Umwandlungslogik nach SAP verschieben, wenn das SAP-Datenobjekt die Open SQL ABAP-Join-Syntax verwendet. Konfigurieren Sie den Select-Syntaxmodus in der Leseoperation des SAP-Datenobjekts.

Ausnahmen bei SAP-Datentypen

Der Datenintegrationsdienst verarbeitet Filterumwandlungslogik, wenn die Quelle die Umwandlungslogik nicht verarbeiten kann. Der Datenintegrationsdienst verarbeitet Filterumwandlungslogik für eine SAP-Quelle, wenn der Umwandlungsausdruck die folgenden Datentypen enthält:

- RAW
- LRAW

- LCHR

Pushdown-Optimierungsausdrücke

Der Data Integration Service kann die Umwandlungslogik in die Quelldatenbank verschieben, wenn die Umwandlung Operatoren und Funktionen enthält, die die Quelle unterstützt. Der Data Integration Service übersetzt den Umwandlungsausdruck in eine Abfrage, indem er äquivalente Operatoren und Funktionen in der Datenbank ermittelt. Wenn kein entsprechender Operator oder keine passende Funktion gefunden wird, verarbeitet der Data Integration Service die Umwandlungslogik.

Wenn die Quelle eine ODBC-Verbindung verwendet und Sie einen datenbankspezifischen ODBC-Provider im ODBC-Verbindungsobjekt konfigurieren, betrachtet der Data Integration Service die Quelle als nativen Quelltyp.

Funktionen

Informatica-Funktionen sind für nichtrelationale Quellen unter z/OS nicht verfügbar. Die folgende Tabelle zeigt die Informatica-Funktionen an, die für die Pushdown-Optimierung für IBM DB2-Quellen verfügbar sind:

Funktion	DB2 für i5/OS ¹	DB2 für LUW	DB2 für z/OS ¹
ABORT()	Nein	Nein	Nein
ABS()	Nein	Ja	Nein
ADD_TO_DATE()	Ja	Ja	Ja
AES_DECRYPT()	Nein	Nein	Nein
AES_ENCRYPT()	Nein	Nein	Nein
ASCII()	Ja	Ja	Ja
AVG()	Ja	Ja	Ja
CEIL()	Ja	Ja	Ja
CHOOSE()	Nein	Nein	Nein
CHR()	Nein	Ja	Nein
CHRCODE()	Nein	Ja	Ja
COMPRESS()	Nein	Nein	Nein
CONCAT()	Ja	Ja	Ja
COS()	Ja	Ja	Ja
COSH()	Ja	Ja	Ja

Funktion	DB2 für i5/OS¹	DB2 für LUW	DB2 für z/OS¹
COUNT()	Ja	Ja	Ja
CRC32()	Nein	Nein	Nein
CUME()	Nein	Nein	Nein
DATE_COMPARE()	Ja	Ja	Ja
DATE_DIFF()	Nein	Nein	Nein
DECODE()	Nein	Ja	Nein
DECODE_BASE64()	Nein	Nein	Nein
DECOMPRESS()	Nein	Nein	Nein
ENCODE_BASE64()	Nein	Nein	Nein
ERROR()	Nein	Nein	Nein
EXP()	Nein	Ja	Nein
FIRST()	Nein	Nein	Nein
FLOOR()	Nein	Ja	Nein
FV()	Nein	Nein	Nein
GET_DATE_PART()	Ja	Ja	Ja
GREATEST()	Nein	Nein	Nein
IIF()	Nein	Ja	Nein
IN()	Nein	Ja	Nein
INDEXOF()	Nein	Nein	Nein
INITCAP()	Nein	Nein	Nein
INSTR()	Ja	Ja	Ja
IS_DATE()	Nein	Nein	Nein
IS_NUMBER()	Nein	Nein	Nein
IS_SPACES()	Nein	Nein	Nein
ISNULL()	Ja	Ja	Ja
LAST()	Nein	Nein	Nein
LAST_DAY()	Nein	Nein	Nein

Funktion	DB2 für i5/OS ¹	DB2 für LUW	DB2 für z/OS ¹
LEAST()	Nein	Nein	Nein
LENGTH()	Ja	Ja	Ja
LN()	Ja	Ja	Ja
LOG()	Ja	Ja	Ja
LOWER()	Ja	Ja	Ja
LPAD()	Nein	Nein	Nein
LTRIM()	Ja	Ja	Ja
MAKE_DATE_TIME()	Nein	Nein	Nein
MAX()	Ja	Ja	Ja
MD5()	Nein	Nein	Nein
MEDIAN()	Nein	Nein	Nein
METAPHONE()	Nein	Nein	Nein
MIN()	Ja	Ja	Ja
MOD()	Ja	Ja	Ja
MOVINGAVG()	Nein	Nein	Nein
MOVINGSUM()	Nein	Nein	Nein
NPER()	Nein	Nein	Nein
PERCENTILE()	Nein	Nein	Nein
PMT()	Nein	Nein	Nein
POWER()	Ja	Ja	Ja
PV()	Nein	Nein	Nein
RAND()	Nein	Nein	Nein
RATE()	Nein	Nein	Nein
REG_EXTRACT()	Nein	Nein	Nein
REG_MATCH()	Nein	Nein	Nein
REG_REPLACE	Nein	Nein	Nein
REPLACECHR()	Nein	Nein	Nein

Funktion	DB2 für i5/OS ¹	DB2 für LUW	DB2 für z/OS ¹
REPLACESTR()	Nein	Nein	Nein
REVERSE()	Nein	Nein	Nein
ROUND(DATE)	Nein	Nein	Ja
ROUND(NUMBER)	Ja	Ja	Ja
RPAD()	Nein	Nein	Nein
RTRIM()	Ja	Ja	Ja
SET_DATE_PART()	Nein	Nein	Nein
SIGN()	Ja	Ja	Ja
SIN()	Ja	Ja	Ja
SINH()	Ja	Ja	Ja
SOUNDEX()	Nein	Ja ¹	Nein
SQRT()	Nein	Ja	Nein
STDDEV()	Ja	Ja	Ja
SUBSTR()	Ja	Ja	Ja
SUM()	Ja	Ja	Ja
SYSTIMESTAMP()	Ja	Ja	Ja
TAN()	Ja	Ja	Ja
TANH()	Ja	Ja	Ja
TO_BIGINT	Ja	Ja	Ja
TO_CHAR(DATE)	Ja	Ja	Ja
TO_CHAR(NUMBER)	Ja	Ja ²	Ja
TO_DATE()	Ja	Ja	Ja
TO_DECIMAL()	Ja	Ja ³	Ja
TO_FLOAT()	Ja	Ja	Ja
TO_INTEGER()	Ja	Ja	Ja
TRUNC(DATE)	Nein	Nein	Nein
TRUNC(NUMBER)	Ja	Ja	Ja

Funktion	DB2 für i5/OS ¹	DB2 für LUW	DB2 für z/OS ¹
UPPER()	Ja	Ja	Ja
VARIANCE()	Ja	Ja	Ja

. ¹Der Datenintegrationsdienst kann diese Funktionen nur zur Quelle verschieben, wenn sie in der Filterumwandlungslogik enthalten sind.

. ²Wenn diese Funktion ein Decimal- oder Float-Argument übernimmt, kann der Datenintegrationsdienst die Funktion nur verschieben, wenn sie in der Filterumwandlungslogik enthalten ist.

. ³Wenn diese Funktion ein String-Argument übernimmt, kann der Datenintegrationsdienst die Funktion nur verschieben, wenn sie in der Filterumwandlungslogik enthalten ist.

In der folgenden Tabelle werden die Informatica-Funktionen angezeigt, die für die Pushdown-Optimierung für Greenplum-, Hive-, Microsoft SQL Server-, Netezza-, ODBC-, Oracle-, SAP, SAP HANA-, Sybase ASE- und Teradata-Quellen verfügbar sind:

Funktion	Greenplum	Hive	Microsoft SQL Server	Netezza	ODBC	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
ABORT()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ABS()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
ADD_TO_DATE()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
AES_DECRYPT()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
AES_ENCRYPT()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ASCII()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Nein
AVG()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
CEIL()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
CHOOSE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
CHR()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Nein
CHRCODE()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Nein
COMPRESS()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
CONCAT()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
COS()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
COSH()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
COUNT()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
CRC32()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein

Funktion	Greenplum	Hive	Microsoft SQL Server	Netezza	ODBC	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
CUME()	Nein	Nein	Ja	Nein	Nein	Nein	Nein	Nein	Nein	Nein
DATE_COMPARE()	Nein	Nein	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
DATE_DIFF()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
DECODE()	Ja	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
DECODE_BASE64()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
DECOMPRESS()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ENCODE_BASE64()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ERROR()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
EXP()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
FIRST()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
FLOOR()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
FV()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
GET_DATE_PART()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
GREATEST()	Nein	Nein	Nein	Nein	Nein	Ja	Nein	Nein	Nein	Nein
IIF()	Ja	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
IN()	Ja	Nein	Ja	Nein	Ja	Ja	Nein	Ja	Ja	Ja
INDEXOF()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
INITCAP()	Nein	Nein	Nein	Ja	Nein	Ja	Nein	Nein	Nein	Nein
INSTR()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
IS_DATE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
IS_NUMBER()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
IS_SPACES()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ISNULL()	Ja	Ja	Ja	Nein	Ja	Ja	Nein	Ja	Ja	Ja
LAST()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
LAST_DAY()	Nein	Nein	Nein	Ja	Nein	Ja	Nein	Nein	Nein	Nein
LEAST()	Nein	Nein	Nein	Nein	Nein	Ja	Nein	Nein	Nein	Nein

Funktion	Greenplum	Hive	Microsoft SQL Server	Netezza	ODBC	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
LENGTH()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
LN()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
LOG()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
LOWER()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
LPAD()	Nein	Ja	Nein	Ja	Nein	Ja	Nein	Nein	Nein	Nein
LTRIM()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Nein	Ja
MAKE_DATE_TIME()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
MAX()	Ja	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
MD5()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
MEDIAN()	Nein	Nein	Nein	Nein	Nein	Ja	Nein	Nein	Nein	Nein
METAPHONE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
MIN()	Ja	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
MOD()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
MOVINGAVG()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
MOVINGSUM()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
NPER()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
PERCENTILE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
PMT()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
POWER()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
PV()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
RAND()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
RATE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
REG_EXTRACT()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
REG_MATCH()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
REG_REPLACE	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
REPLACECHR()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein

Funktion	Greenplum	Hive	Microsoft SQL Server	Netezza	ODBC	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
REPLACESTR()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
REVERSE()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
ROUND(DATE)	Nein	Nein	Nein	Nein	Nein	Ja	Nein	Nein	Nein	Nein
ROUND(NUMBER)	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
RPAD()	Nein	Ja	Nein	Ja	Nein	Ja	Nein	Nein	Nein	Nein
RTRIM()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
SET_DATE_PART()	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein	Nein
SIGN()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
SIN()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
SINH()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
SOUNDEX()	Nein	Nein	Ja	Nein	Nein	Ja	Nein	Nein	Ja	Nein
SQRT()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
STDDEV()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
SUBSTR()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
SUM()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
SYSTIMESTAMP()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Nein
TAN()	Nein	Nein	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
TANH()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_BIGINT	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_CHAR(DATE)	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_CHAR(NUMBER)	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_DATE()	Nein	Nein	Ja	Ja	Nein	Ja	Ja	Nein	Ja	Ja
TO_DECIMAL()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_FLOAT()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TO_INTEGER()	Nein	Ja	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
TRUNC(DATE)	Nein	Nein	Nein	Ja	Nein	Ja	Nein	Nein	Nein	Nein

Funktion	Greenplum	Hive	Microsoft SQL Server	Netezza	ODBC	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
TRUNC(NUMBER)	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja
UPPER()	Nein	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja
VARIANCE()	Nein	Nein	Ja	Ja	Nein	Ja	Nein	Nein	Ja	Ja

. ¹Der Datenintegrationsdienst kann diese Funktionen nur zur Quelle verschieben, wenn sie in der Filterumwandlungslogik enthalten sind.

In der folgenden Tabelle werden die Informatica-Funktionen zusammengefasst, die für die Pushdown-Optimierung zur Verfügung stehen.

Funktion	DB2 für LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
ABS()	Ja	Ja	Ja	Ja	Ja
ADD_TO_DATE()	Ja	Ja	Nein	Ja	Ja
ASCII()	Ja	Ja	Nein	Ja	Ja
CEIL()	Ja	Ja	Nein	Ja	Ja
CHR()	Ja	Ja	Nein	Ja	Ja
CONCAT()	Ja	Ja	Nein	Ja	Ja
COS()	Ja	Ja	Ja	Ja	Ja
COSH()	Ja	Ja	Nein	Ja	Ja
DATE_COMPARE()	Ja	Ja	Ja	Ja	Ja
DECODE()	Ja	Ja	Ja	Ja	Ja
EXP()	Ja	Ja	Ja	n/v	Ja
FLOOR()	Nein	Ja	Nein	Ja	Ja
GET_DATE_PART()	Ja	Ja	Nein	Ja	Ja
IIF()	Ja	Ja	Ja	Nein	Ja
IN()	Nein	Ja	Ja	Nein	Ja
INITCAP()	Nein	Nein	Nein	Ja	Nein
INSTR()	Ja	Ja	Nein	Ja	Ja
ISNULL()	Ja	Ja	Ja	Ja	Ja

Funktion	DB2 für LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
LAST_DAY()	Nein	Nein	Nein	Ja	n/v
LENGTH()	Ja	Ja	Nein	Ja	Ja
LN()	Ja	Nein	Nein	Ja	Ja
LOG()	Ja	Ja	Nein	Ja	Ja
LOOKUP()	Nein	Nein	Ja	Nein	Nein
LOWER()	Ja	Ja	Ja	Ja	Ja
LPAD()	Nein	Nein	Nein	Ja	n/v
LTRIM()	X	Ja	Nein	Ja	Ja
MOD()	X	Ja	Nein	Ja	Ja
POWER()	X	Ja	Nein	Ja	Ja
ROUND(DATE)	Nein	Nein	Nein	Ja	n/v
ROUND(NUMBER)	Ja	Ja	Nein	Ja	Ja
RPAD()	Nein	Nein	Nein	Ja	Nein
RTRIM()	Ja	Ja	Nein	Ja	Ja
SIGN()	Ja	Ja	Nein	Ja	Ja
SIN()	Ja	Ja	Ja	Ja	Ja
SINH()	Ja	Ja	Nein	Ja	Ja
SOUNDEX()	Ja ¹	Ja	Nein	Ja	Ja
SQRT()	Ja	Ja	Ja	Ja	Ja
SUBSTR()	Ja	Ja	Nein	Ja	Ja
SYSDATE()	Ja	Ja	Nein	Ja	Ja
SYSTIMESTAMP()	Ja	Ja	Nein	Ja	Ja
TAN()	Ja	Ja	Ja	Ja	Ja
TANH()	Ja	Ja	Nein	Ja	Ja
TO_BIGINT	Ja	Ja	Nein	Ja	Ja
TO_CHAR(DATE)	Ja	Ja	Nein	Ja	Ja
TO_CHAR(NUMBER)	Ja ²	Ja	Nein	Ja	Ja

Funktion	DB2 für LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
TO_DATE()	Ja	Ja	Nein	Ja	Ja
TO_DECIMAL()	Ja ³	Ja	Nein	Ja	Ja
TO_FLOAT()	Ja	Ja	Nein	Ja	Ja
TO_INTEGER()	Ja	Ja	Nein	Ja	Ja
TRUNC(DATE)	Nein	Nein	Nein	Ja	Nein
TRUNC(NUMBER)	Ja	Ja	Nein	Ja	Ja
UPPER()	Ja	Ja	Ja	Ja	Ja

. ¹Der Datenintegrationsdienst kann diese Funktionen nur zur Quelle verschieben, wenn sie in der Filterumwandlungslogik enthalten sind.

. ²Wenn diese Funktion ein Decimal- oder Float-Argument übernimmt, kann der Datenintegrationsdienst die Funktion nur verschieben, wenn sie in der Filterumwandlungslogik enthalten ist.

. ³Wenn diese Funktion ein String-Argument übernimmt, kann der Datenintegrationsdienst die Funktion nur verschieben, wenn sie in der Filterumwandlungslogik enthalten ist.

Ausnahmen für Hive-Funktionen

Unter bestimmten Bedingungen kann der Datenintegrationsdienst unterstützte Funktionen nicht in Oracle-Quellen verschieben.

Der Datenintegrationsdienst verarbeitet die Umwandlungslogik für Hive-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- LTRIM enthält Leerzeichen als zweites Argument.
- RTRIM enthält Leerzeichen als zweites Argument.

Der Datenintegrationsdienst kann die Umwandlungslogik für Hive-Quellen nicht verarbeiten, wenn Sie die folgenden Funktionen mit dem Datum-Datentyp verwenden:

- CONCAT
- MAX
- MIN
- ROUND
- TO_BIGINIT
- TO_INTEGER

Ausnahmen für IBM DB2-Funktionen

Unter bestimmten Bedingungen kann der Data Integration Service keine unterstützten Funktionen in IBM DB2 für i5/OS-, DB2 für LUW- und DB2 für z/OS- Quellen verschieben. Unter bestimmten Bedingungen kann der Data Integration Service keine unterstützten Funktionen in IBM DB2 für LUW-Quellen verschieben.

Der Data Integration Service verarbeitet Umwandlungslogik für IBM DB2-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- ADD_TO_DATE oder GET_DATE_PART gibt Ergebnisse mit einer Präzision von Millisekunden oder Nanosekunden zurück.
- LTRIM enthält mehr als ein Argument.
- RTRIM enthält mehr als ein Argument.
- TO_BIGINT konvertiert einen String in einen bigint-Wert in einer DB2 für LUW-Quelle.
- TO_CHAR konvertiert ein Datum in einen Zeichenstring und gibt ein Format an, das nicht von DB2 unterstützt wird.
- TO_DATE konvertiert einen Zeichenstring in ein Datum und gibt ein Format an, das nicht von DB2 unterstützt wird.
- TO_DECIMAL konvertiert einen String in einen Dezimalwert ohne das Scale-Argument.
- TO_FLOAT konvertiert einen String in eine Gleitkommazahl mit doppelter Präzision.
- TO_INTEGER konvertiert einen String in einen ganzzahligen Wert in einer DB2 für LUW-Quelle.

Ausnahmen für Microsoft SQL Server-Funktionen

Unter bestimmten Bedingungen kann der Data Integration Service unterstützt Funktionen nicht nach Microsoft SQL Server-Quellen verschieben.

Der Data Integration Service verarbeitet Umwandlungslogik für Microsoft SQL Server-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- IN enthält das CaseFlag-Argument.
- INSTR enthält mehr als drei Argumente.
- LTRIM enthält mehr als ein Argument.
- RTRIM enthält mehr als ein Argument.
- TO_BIGINT enthält mehr als ein Argument.
- TO_INTEGER enthält mehr als ein Argument.

Ausnahmen für Netezza-Funktion

Unter bestimmten Bedingungen kann der Datenintegrationsdienst unterstützte Funktionen nicht in Netezza-Quellen verschieben.

Der Datenintegrationsdienst verarbeitet Umwandlungslogik für Netezza-Quellen, wenn Ausdrücke unterstützte Funktionen mit der folgenden Logik enthalten:

- SYSTIMESTAMP enthält Daten im Format JJJJ-MM-TT HH24: MI: SS.US.
- TO_CHAR(DATE) und TO_DATE() enthalten Datumsangaben im Format JJJJ-MM-TT HH24:MI:SS.US mit Mikrosekundengenauigkeit.

Ausnahmen für Oracle-Funktionen

Unter bestimmten Bedingungen kann der Data Integration Service unterstützt Funktionen nicht nach Oracle-Quellen verschieben.

Der Data Integration Service verarbeitet Umwandlungslogik für Oracle-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- ADD_TO_DATE oder GET_DATE_PART gibt Ergebnisse mit einer Subsekunden-Präzision zurück.
- ROUND rundet Werte in Sekunden oder Subsekunden.
- SYSTIMESTAMP gibt das Datum und die Uhrzeit mit Mikrosekunden-Präzision zurück.
- TRUNC trunziert Sekunden oder Subsekunden.

Ausnahmen für ODBC-Funktionen

Der Data Integration Service verarbeitet Umwandlungslogik für ODBC, wenn es sich bei dem CaseFlag-Argument für die IN-Funktion um eine Zahl handelt (nicht Null).

Hinweis: Wenn die ODBC-Verbindungsobjekteigenschaften einen datenbankspezifischen ODBC-Provider enthalten, betrachtet der Data Integration Service die Quelle als den nativen Quelltyp.

Der Datenintegrationsdienst kann die Funktion EXP() nicht in Teradata-Quellen verschieben, wenn Sie den ODBC-Provider im Verbindungsobjekt als **Andere** angeben. Stellen Sie den ODBC-Provider auf **Teradata** ein, um die Funktion EXP() zu verschieben.

Ausnahmen für Sybase ASE-Funktionen

Unter bestimmten Bedingungen kann der Data Integration Service unterstützt Funktionen nicht nach Sybase ASE-Quellen verschieben.

Der Data Integration Service verarbeitet Umwandlungslogik für Sybase ASE-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- IN enthält das CaseFlag-Argument.
- INSTR enthält mehr als zwei Argumente.
- LTRIM enthält mehr als ein Argument.
- RTRIM enthält mehr als ein Argument.
- TO_BIGINT enthält mehr als ein Argument.
- TO_INTEGER enthält mehr als ein Argument.
- TRUNC(Numbers) enthält mehr als ein Argument.

Teradata-Funktionen - Ausnahmen

Unter bestimmten Bedingungen kann der Datenintegrationsdienst unterstützte Funktionen nicht in Teradata-Quellen verschieben.

Der Datenintegrationsdienst verarbeitet die Umwandlungslogik für Teradata-Quellen, wenn Ausdrücke unterstützte Funktion mit der folgenden Logik enthalten:

- ADD_TO_DATE enthält Attribute, bei denen es sich nicht um YEAR und MONTH handelt.
- IN enthält das CaseFlag-Argument.
- INSTR enthält mehr als zwei Argumente.
- LTRIM enthält mehr als ein Argument.

- ROUND enthält mehr als ein Argument.
- RTRIM enthält mehr als ein Argument.

Operatoren

Die folgende Tabelle enthält eine Zusammenfassung der Verfügbarkeit von Informatica-Operatoren nach Quelltyp. In jeder Spalte wird angezeigt, ob der Datenintegrationsdienst den Operator in die Quelle verschieben kann.

Hinweis: Nichtrelationale Quellen sind IMS-, VSAM- und sequentielle Datensätze unter z/OS.

Operator	DB2 für LUW	DB2 für i5/OS oder z/OS [*]	Microsoft SQL Server	Nichtrelational*	ODBC	Oracle	SAP*	Sybase ASE	Hive	Teradata	Greenplum
+ - *	Ja	Ja	Ja	Ja	Ja	Ja	Nein	Ja	Ja	Ja	Ja
/	Ja	Ja	Ja	Nein	Ja	Ja	Nein	Ja	Ja	Ja	Ja
%	Ja	Ja	Ja	Nein	Nein	Ja	Nein	Ja	Ja	Ja	Nein
	Ja	Ja	Ja	Nein	Nein	Ja	Nein	Ja	Ja	Ja	Nein
= > < >= <=	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
<>	Ja	Ja	Ja	Nein	Ja	Ja	Ja	Ja	Ja	Ja	Ja
!=	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
^=	Ja	Ja	Ja	Nein	Ja	Ja	Ja	Ja	Ja	Ja	Ja
AND oder	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja	Ja
NOT	Ja	Ja	Ja	Nein	Ja	Ja	Nein	Ja	Ja	Ja	Ja

^{*}Der Data Integration Service kann diese Operatoren nur zur Quelle verschieben, wenn sie in der Filterumwandlungslogik enthalten sind.

Operator	DB2 für LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
+ - *	Ja	Ja	Ja	Ja	Ja
/	Ja	Ja	Ja	Ja	Ja

Operator	DB2 für LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
%	Ja	Ja	Nein	Ja	Ja
	Ja	Ja	Nein	Ja	Ja
=	Ja	Ja	Ja	Ja	Ja
>					
<					
>=					
<=					
<>	Ja	Ja	Ja	Ja	Ja
!=	Ja	Ja	Ja	Ja	Ja
^=	Ja	Ja	Ja	Ja	Ja
AND oder	Ja	Ja	Ja	Ja	Ja
NOT	Ja	Ja	Ja	Ja	Ja

Vergleichen der Ausgabe vom Data Integration Service und von Quellen

Der Data Integration Service und Quellen können bei der Verarbeitung derselben Umwandlungslogik unterschiedliche Ergebnisse produzieren. Wenn der Data Integration Service Umwandlungslogik in die Quelle verschiebt, kann die Ausgabe der Umwandlungslogik hiervon abweichen.

Unterscheidung nach Groß-/Kleinschreibung

Der Data Integration Service und eine Datenbank können die Unterscheidung nach Groß-/Kleinschreibung unterschiedlich behandeln. Im Gegensatz zur Datenbank verwendet beispielsweise der Data Integration Service Abfragen unter Beachtung der Groß-/Kleinschreibung. Eine Filterumwandlung verwendet die folgende Filterbedingung: IIF(col_varchar2 = 'CA', TRUE, FALSE). Sie benötigen die Datenbank, um Zeilen zurückzugeben, die mit "CA" übereinstimmen. Wenn Sie die Umwandlungslogik jedoch in eine Datenbank verschieben, bei der nicht nach Groß-/Kleinschreibung unterschieden wird, gibt er Zeilen zurück, die den Werten "CA", "ca", "cA" und "CA" entsprechen.

In Zeichenwerte konvertierte numerische Werte

Der Data Integration Service und eine Datenbank können den gleichen numerischen Wert in einen Zeichenwert in verschiedenen Formaten konvertieren. Die Datenbank konvertiert numerische Werte möglicherweise in ein inakzeptables Zeichenformat. Eine Tabelle enthält beispielsweise die Zahl 1234567890. Wenn der Data Integration Service die Zahl in einen Zeichenwert konvertiert, fügt er die Zeichen "1234567890" ein. Eine Datenbank konvertiert die Zahl möglicherweise in "1.2E9". Beide Zeichensätze stellen denselben Wert dar.

Datumsformate für TO_CHAR- und TO_DATE-Funktionen

Der Data Integration Service verwendet das Datumsformat in der TO_CHAR- oder TO_DATE-Funktion, wenn der Data Integration Service die Funktion in die Datenbank verschiebt. Verwenden Sie die TO_DATE-Funktionen, um Datums- oder Uhrzeitwerte zu vergleichen. Wenn Sie TO_CHAR zum Vergleichen von Datums- oder Uhrzeitwerten verwenden, kann die Datenbank ein Leerzeichen oder eine vorangestellte Null als Werte hinzufügen, zum Beispiel einen Monat, einen Tag oder eine Stunde mit einer einzelnen Ziffer. Die Ergebnisse des Datenbankvergleichs können sich von den Ergebnissen des Data Integration Service unterscheiden, wenn die Datenbank ein Leerzeichen oder eine vorangestellte Null hinzufügen.

Präzision

Der Data Integration Service und eine Datenbank können eine andere Präzision für bestimmte Datentypen aufweisen. Umwandlungsdattentypen verwenden eine numerische Standardpräzision, die von nativen Datentypen abweichen können. Die Ergebnisse können variieren, wenn die Datenbank eine andere Präzision als der Data Integration Service verwendet.

Funktion SYSTIMESTAMP

Bei Verwendung der Funktion SYSTIMESTAMP gibt der Datenintegrationsdienst das aktuelle Datum und die Uhrzeit für den Knoten zurück, der den Dienstprozess ausführt. Wenn Sie jedoch die Umwandlungslogik in die Datenbank verschieben, gibt die Datenbank das aktuelle Datum und die Uhrzeit für den Computer zurück, der als Host für die Datenbank eingesetzt wird. Wenn die Zeitzone des Computers, der als Host für die Datenbank eingesetzt wird, nicht der Zeitzone des Computers entspricht, auf dem der Data Integration Service-Prozess ausgeführt wird, können die Ergebnisse variieren.

Wenn Sie SYSTIMESTAMP in eine IBM DB2- oder eine Sybase ASE-Datenbank verschieben und Sie das Format für SYSTIMESTAMP angeben, ignoriert die Datenbank das Format und gibt den kompletten Zeitstempel zurück.

LTRIM-, RTRIM- oder SOUNDEX-Funktion

Wenn Sie LTRIM, RTRIM oder SOUNDEX in eine Datenbank verschieben, behandelt die Datenbank das Argument (' ') als NULL, wohingegen der Data Integration Service das Argument (' ') als Leerzeichen behandelt.

LAST_DAY-Funktion in Oracle-Quelle

Wenn Sie LAST_DAY nach Oracle verschieben, gibt Oracle das Datum bis auf die Sekunde genau zurück. Wenn das Eingabedatum Subsekunden beinhaltet, passt Oracle das Datum sekundengenaue an.

KAPITEL 8

Partitionierte Mappings

Dieses Kapitel umfasst die folgenden Themen:

- [Partitionierte Mappings – Übersicht, 93](#)
- [Ein Thread für jede Pipeline-Stage, 94](#)
- [Mehrere Threads für jede Pipeline-Stage, 95](#)
- [Berechnungen für den Parallelismuswert, 96](#)
- [Partitionierte Einfachdateiquellen, 99](#)
- [Partitionierte relationale Quellen, 100](#)
- [Partitionierte Einfachdateiziele, 103](#)
- [Partitionierte relationale Ziele, 105](#)
- [Partitionierte Umwandlungen, 107](#)
- [Mappings, die Quellzeilen begrenzen, 110](#)
- [Deaktivieren der Partitionierung für ein Mapping, 111](#)
- [Überschreiben des maximalen Parallelismus für ein Mapping, 112](#)

Partitionierte Mappings – Übersicht

Bei vorhandener Partitionierungsoption können Administratoren den Datenintegrationsdienst aktivieren, um den Parallelismus beim Ausführen von Mappings zu maximieren. Beim Maximieren des Parallelismus durch Administratoren unterteilt der Datenintegrationsdienst die zugrunde liegenden Daten dynamisch in Partitionen und verarbeitet alle Partitionen gleichzeitig.

Wenn Mappings umfangreiche Datasets verarbeiten oder Umwandlungen enthalten, die komplizierte Berechnungen durchführen, kann die Mapping-Verarbeitung viel Zeit in Anspruch nehmen und einen geringen Datendurchsatz zur Folge haben. Wenn Sie Partitionierung für diese Mappings aktivieren, verwendet der Datenintegrationsdienst zusätzliche Threads zum Verarbeiten des Mappings, wodurch die Leistung gesteigert werden kann.

Zum Aktivieren der Partitionierung können Administratoren und Entwickler die folgenden Aufgaben durchführen:

Administratoren setzen den maximalen Parallelismus für den Datenintegrationsdienst-Prozess im Administrator-Tool auf einen Wert größer als 1.

Maximaler Parallelismus bestimmt die maximale Anzahl paralleler Threads, die eine einzelne Pipeline-Stage verarbeiten. Administratoren erhöhen den maximalen Parallelismuswert basierend auf der Anzahl

der CPUs, die auf dem Knoten verfügbar sind, auf dem der Datenintegrationsdienst-Prozess ausgeführt wird.

Entwickler können den maximalen Parallelismuswert für ein Mapping optional im Developer-Tool festlegen.

Standardmäßig ist der maximale Parallelismus für jedes Mapping auf „Auto“ gesetzt. Der Datenintegrationsdienst berechnet den tatsächlichen Parallelismuswert basierend auf dem maximalen Parallelismus für den Datenintegrationsdienst-Prozess und basierend auf der maximalen Anzahl von Partitionen für alle Quellen im Mapping.

Entwickler können den maximalen Parallelismus für ein Mapping auf 1 festlegen, um die Partitionierung für das Mapping zu deaktivieren. Alternativ dazu können Entwickler den Standardwert überschreiben, um die Anzahl der Threads festzulegen, die der Datenintegrationsdienst erstellt. Wenn der maximale Parallelismus für den Datenintegrationsdienst-Prozess und das Mapping auf andere Ganzzahlwerte festgelegt ist, verwendet der Datenintegrationsdienst den minimalen Wert für den tatsächlichen Parallelismuswert.

Wenn Partitionierung für ein Mapping deaktiviert ist, teilt der Datenintegrationsdienst das Mapping in Pipeline-Stages auf und erstellt einen Thread zur Verarbeitung aller Stages.

Wenn Partitionierung für ein Mapping aktiviert ist, erstellt der Datenintegrationsdienst mehrere Threads für die Verarbeitung jeder Mapping-Pipeline-Stage.

Der Datenintegrationsdienst kann Partitionen für Mappings mit physischen Daten als Eingabe und Ausgabe erstellen. Der Datenintegrationsdienst kann Partitionen für Einfachdatei-, IBM DB2 für LUW- und relationale Oracle-Datenobjekte sowie für Umwandlungen in einer Mapping-Pipeline erstellen.

Ein Thread für jede Pipeline-Stage

Wenn der maximale Parallelismus auf eins gesetzt ist, ist die Partitionierung deaktiviert. Der Datenintegrationsdienst unterteilt ein Mapping in Pipeline-Stages und erstellt einen Thread zur Verarbeitung jeder Stage.

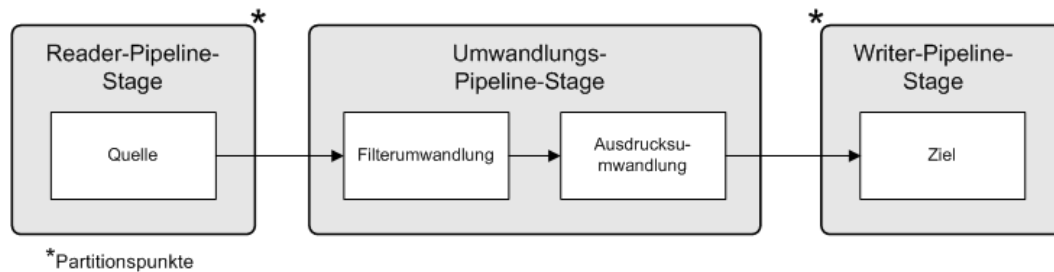
Jedes Mapping enthält eine oder mehrere Pipelines. Eine Pipeline besteht aus einer Quelle und sämtlichen Umwandlungen und Zielen, die Daten aus dieser Quelle empfangen. Der Datenintegrationsdienst teilt eine Mapping-Pipeline in Pipeline-Stages auf und führt dann Extrahierung, Umwandlung und Laden für jede Pipeline-Stage parallel durch.

Partitionspunkte markieren die Grenzen in einer Pipeline und teilen die Pipeline in Stages ein. Der Datenintegrationsdienst fügt für jede Mapping-Pipeline hinter der Quellinstanz und vor der Zielinstanz einen Partitionspunkt ein, um mehrere Pipeline-Stages zu erstellen.

Jede Pipeline-Stage wird in einem der folgenden zugehörigen Threads ausgeführt:

- Reader-Thread, das die vom Datenintegrationsdienst durchgeführte Extrahierung von Daten aus der Quelle steuert.
- Umwandlungs-Thread, das die vom Datenintegrationsdienst durchgeführte Verarbeitung von Daten in der Pipeline steuert.
- Writer-Thread, das den vom Datenintegrationsdienst durchgeführten Vorgang zum Laden von Daten in das Ziel steuert.

Die folgende Abbildung zeigt ein Mapping, das in eine Reader-, eine Umwandlungs- und eine Writer-Pipeline-Stage aufgeteilt wurde:



Da die Pipeline drei Stages enthält, kann der Datenintegrationsdienst gleichzeitig drei Zeilensätze verarbeiten und die Mapping-Leistung steigern. Während der Reader-Thread beispielsweise den dritten Zeilensatz verarbeitet, verarbeitet der Umwandlungs-Thread den zweiten Zeilensatz und der Writer-Thread den ersten Zeilensatz.

Die folgende Tabelle zeigt, wie mehrere Threads gleichzeitig drei Zeilensätze verarbeiten können:

Reader-Thread	Umwandlungs-Thread	Writer-Thread
Zeilensatz 1	-	-
Zeilensatz 2	Zeilensatz 1	-
Zeilensatz 3	Zeilensatz 2	Zeilensatz 1
Zeilensatz 4	Zeilensatz 3	Zeilensatz 2
Zeilensatz n	Zeilensatz (n-1)	Zeilensatz (n-2)

Enthält die Mapping-Pipeline Umwandlungen, die komplizierte Berechnungen durchführen, kann die Verarbeitung der Umwandlungs-Pipeline-Stage sehr viel Zeit in Anspruch nehmen. Zur Leistungssteigerung fügt der Datenintegrationsdienst Partitionspunkte vor bestimmten Umwandlungen ein, um eine weitere Umwandlungs-Pipeline-Stage zu erstellen.

Mehrere Threads für jede Pipeline-Stage

Wenn der maximale Parallelismus auf einen Wert größer als eins gesetzt ist, ist die Partitionierung aktiviert. Der Datenintegrationsdienst unterteilt ein Mapping in Pipeline-Stages und erstellt mehrere Threads zur Verarbeitung jeder Stage. Die Anzahl der Threads in einer beliebigen Pipeline-Stage stimmt mit der Anzahl der Partitionen in der Stage überein.

Wenn Sie Parallelismus maximieren, führt der Datenintegrationsdienst dynamisch folgende Aufgaben zur Laufzeit aus:

Aufteilen der Daten in Partitionen.

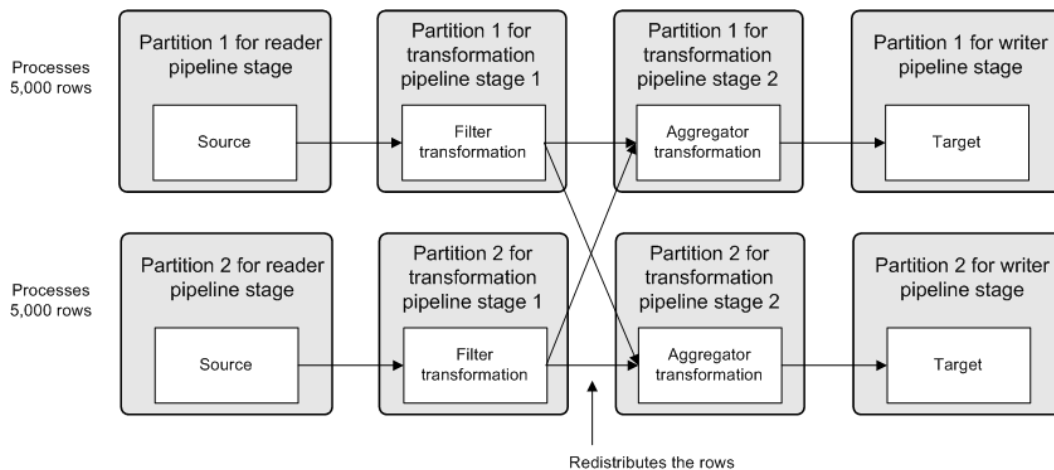
Der Datenintegrationsdienst teilt die zugrunde liegenden Daten dynamisch in Partitionen auf und verarbeitet alle Partitionen gleichzeitig. Der Datenintegrationsdienst bestimmt die Anzahl der Partitionen basierend auf den maximalen Parallelismuswerten. Sie setzen den maximalen Parallelismus im Datenintegrationsdienst-Prozess beispielsweise auf zwei. Sie führen ein Mapping aus, das aus einer

einzelnen Einfachdateiquelle mit 10.000 Zeilen liest. Der Datenintegrationsdienst teilt die Quelldaten zur Laufzeit dynamisch auf, um sie gleichmäßig auf die Zeilen in den beiden Partitionen für jede Pipeline-Stage zu verteilen.

Verteilt Daten auf Partitionsunkten neu.

Der Datenintegrationsdienst legt dynamisch die beste Methode zur Neuverteilung der Daten auf einem Partitionspunkt fest. Beispiel: Der Datenintegrationsdienst verwendet unter Umständen Hash-Partitionierung, um Datenzeilen zwischen Partitionen zu gruppieren. Der Datenintegrationsdienst kann alternativ auch Round-Robin-Partitionierung verwenden, um Daten gleichmäßig auf Partitionen zu verteilen. In bestimmten Fällen kann der Datenintegrationsdienst alle Zeilen an einem Partitionspunkt an den nächsten Partitionspunkt übergeben, ohne sie neu zu verteilen.

Die folgende Abbildung zeigt ein Mapping, das 10.000 Zeilen mit Quelldaten auf zwei Partitionen für jede Pipeline-Stage verteilt:



In der vorherigen Abbildung hat der Datenintegrationsdienst die Quelldaten aufgeteilt, um die Zeilen gleichmäßig auf zwei Partitionen für die Reader- und die erste Umwandlungs-Pipeline-Stage zu verteilen. Am Aggregat-Partitionsunkt verteilt der Datenintegrationsdienst die Daten neu, um Zeilen für den Aggregat-Ausdruck zu gruppieren. Der Datenintegrationsdienst muss die Zeilen auf dem Zielpartitionsunkt nicht neu verteilen. Alle Zeilen in einer einzelnen Partition verbleiben in dieser Partition, nachdem der Zielpartitionsunkt überschritten wurde.

Berechnungen für den Parallelismuswert

Der Datenintegrationsdienst berechnet den tatsächlichen Parallelismuswert basierend auf den maximalen Parallelismuswerten und der maximalen Anzahl der Partitionen für alle Einfachdatei-, IBM DB2 für LUW- oder Oracle-Quellen in einem Mapping. Der Datenintegrationsdienst verwendet den Parallelismuswert zum Erstellen der Anzahl der Threads, die jede Mapping Pipeline-Stage verarbeiten.

Hinweis: Der Datenintegrationsdienst kann Partitionen für Einfachdatei-, DB2 für LUW- und Oracle-Datenobjekte erstellen. Der Datenintegrationsdienst kann keine Partitionen für einen anderen Datenobjekttyp erstellen. Wenn ein Mapping ein Datenobjekt enthält, bei dem es sich nicht um ein Einfachdatei-, DB2 für LUW- oder Oracle-Datenobjekt als eine Quelle handelt, berechnet der Datenintegrationsdienst den Parallelismuswert immer als 1. Die maximalen Parallelismuswerte und die Anzahl der Quellpartitionen werden ignoriert.

Die Datenintegrationsdienst-Berechnungen variieren je nach maximalem Parallelismuswert für das Mapping.

„Auto“ als maximaler Parallelismus für das Mapping

Wenn der maximale Parallelismus für das Mapping auf „Auto“ gesetzt ist, berücksichtigt der Datenintegrationsdienst die Anzahl der Quellpartitionen in den Berechnungen.

Der Datenintegrationsdienst verwendet die folgende Gleichung zur Berechnung des Parallelismuswerts:

$$\text{MIN}(\text{Data Integration Service maximum parallelism}, \text{MAX}(\text{source partitions}))$$

Der Parallelismuswert entspricht dem Minimum der folgenden Werte:

- Für den Datenintegrationsdienst-Prozess festgelegter maximaler Parallelismuswert.
- Maximale Anzahl der Quellpartitionen für alle Quellen im Mapping.

Der Datenintegrationsdienst bestimmt die Anzahl der Quellpartitionen basierend auf den folgenden Quelltypen:

- Einfachdatei. Die Anzahl von Partitionen für eine Einfachdateiquelle entspricht dem maximalen Parallelismuswert, der für den Datenintegrationsdienst-Prozess festgelegt ist.
- Relational. Die Anzahl von Partitionen für eine relationale DB2 für LUW- oder Oracle-Quelle entspricht der Anzahl der Datenbankpartitionen in der relationalen Quelle.

Wenn ein Mapping mehrere Einfachdatei-, DB2 für LUW- oder relationale Oracle-Quellen enthält, verwendet der Datenintegrationsdienst die maximale Anzahl der Partitionen für alle Quellen. Beispiel: Ein Mapping enthält eine DB2 für LUW-Quelle mit vier Datenbankpartitionen und eine andere DB2 für LUW-Quelle, die drei Datenbankpartitionen enthält. Der Datenintegrationsdienst verwendet vier Partitionen als maximale Anzahl der Quellpartitionen.

Der Datenintegrationsdienst führt die folgenden Vergleiche zur Berechnung des tatsächlichen Parallelismuswerts durch:

1. Vergleicht die Anzahl der Partitionen für alle Einfachdateien und relationalen Quellen im Mapping.
2. Gibt den höheren Wert des ersten Vergleichs an.
3. Vergleicht den maximalen für den Datenintegrationsdienst-Prozess festgelegten Parallelismuswert und den höheren der Quellwerte.
4. Gibt den niedrigeren Wert des zweiten Vergleichs an. Verwendet diesen niedrigeren Wert als tatsächlichen Parallelismuswert.
5. Verwendet den Parallelismuswert zum Erstellen der Anzahl der Threads, die jede Mapping-Pipeline-Stage verarbeiten.

Beispielwerte für ein Mapping mit einer einzelnen Einfachdateiquelle

Der maximale Parallelismus für das Mapping ist „Auto“. Das Mapping enthält eine einzelne Einfachdateiquelle.

Die folgende Tabelle enthält eine Auflistung der berechneten Parallelismuswerte für das Mapping:

Maximaler Parallelismus für den Datenintegrationsdienst	Maximaler Parallelismus für das Mapping	Parallelismuswert
5	Auto	5
6	Auto	6

Beispielwerte für ein Mapping mit einer einzelnen relationalen Quelle

Der maximale Parallelismus für das Mapping ist „Auto“. Das Mapping enthält eine einzelne DB2 für LUW- oder relationale Oracle-Quelle.

Die folgende Tabelle enthält eine Auflistung der berechneten Parallelismuswerte für das Mapping:

Maximaler Parallelismus für den Datenintegrationsdienst	Maximaler Parallelismus für das Mapping	Anzahl der Quelldatenbankpartitionen	Parallelismuswert
5	Auto	none	1
5	Auto	3	3
5	Auto	6	5

Beispielwerte für ein Mapping mit einer Einfachdateiquelle und einer relationalen Quelle

Der maximale Parallelismus für das Mapping ist „Auto“. Das Mapping enthält eine Einfachdateiquelle und eine einzelne DB2 für LUW- oder relationale Oracle-Quelle.

Die folgende Tabelle enthält eine Auflistung der berechneten Parallelismuswerte für das Mapping:

Maximaler Parallelismus für den Datenintegrationsdienst	Maximaler Parallelismus für das Mapping	Einfachdatei-Partitionswert	Anzahl der Quelldatenbankpartitionen	Parallelismuswert
5	Auto	5	none	5
5	Auto	5	3	5
5	Auto	5	6	5

Ganzzahlwert als maximaler Parallelismus für das Mapping

Wenn der maximale Parallelismus für das Mapping ein Ganzzahlwert ist, berücksichtigt der Datenintegrationsdienst nicht die Anzahl der Quellpartitionen in den Berechnungen.

Der Datenintegrationsdienst verwendet die folgende Gleichung zur Berechnung des Parallelismuswerts:

`MIN(Data Integration Service maximum parallelism, mapping maximum parallelism)`

Der Parallelismuswert entspricht dem Minimum der folgenden Werte:

- Für den Datenintegrationsdienst-Prozess festgelegter maximaler Parallelismuswert.
- Maximaler Parallelismuswert für das Mapping.

In der folgenden Tabelle finden Sie eine Auflistung der berechneten Parallelismuswerte für ein Mapping, das eine beliebige Anzahl von Einfachdatei-, DB2 für LUW- oder Oracle-Quellen enthält, wenn der maximale Parallelismus für das Mapping ein Ganzzahlwert ist:

Maximaler Parallelismus für den Datenintegrationsdienst	Maximaler Parallelismus für das Mapping	Parallelismuswert
5	3	3
5	6	5

Partitionierte Einfachdateiquellen

Wenn ein für die Partitionierung aktiviertes Mapping eine Einfachdateiquelle enthält, verwendet der Datenintegrationsdienst mehrere Partitionen zum Lesen der Dateiquelle.

Der Datenintegrationsdienst verwendet den Parallelismuswert, um die Anzahl der Partitionen festzulegen, die die Einfachdateiquelle lesen. Wenn der Parallelismuswert beispielsweise auf vier gesetzt ist, erstellt der Datenintegrationsdienst vier Partitionen zum Lesen der Einfachdateiquelle.

Der Datenintegrationsdienst kann Partitionen für die folgenden Einfachdatei-Quellentypen erstellen:

- Direkte Datei
- Indirekte Datei
- Dateiverzeichnis
- Befehl
- Datei oder Dateiverzeichnis im verteilten Hadoop-Dateisystem (Hadoop Distributed File System, HDFS)

Wenn der Datenintegrationsdienst mehrere Partitionen zum Lesen der Dateiquelle verwendet, werden mehrere gleichzeitige Verbindungen zur Quelle hergestellt. Der Datenintegrationsdienst behält die Reihenfolge der Zeilen nicht bei, da die Zeilen in der Datei oder Dateiliste nicht der Reihe nach gelesen werden. Um die Reihenfolge der Zeilen beizubehalten, wenn mehrere Partitionen aus einer Dateiquelle lesen, konfigurieren Sie eine Partitionierung für gleichzeitiges Lesen.

Wenn der Datenintegrationsdienst mehrere Partitionen zum Lesen einer direkten Datei verwendet, werden mehrere Reader-Threads zum gleichzeitigen Lesen der Quelle erstellt.

Wenn der Datenintegrationsdienst mehrere Partitionen zum Lesen einer direkten Datei oder eines Dateiverzeichnisses verwendet, werden mehrere Reader-Threads zum gleichzeitigen Lesen der Dateien in der Liste oder dem Verzeichnis erstellt. Der Datenintegrationsdienst verwendet unter Umständen mehrere Threads zum Lesen einer einzelnen Datei. Alternativ kann der Datenintegrationsdienst auch einen einzelnen Thread zum Lesen mehrerer Dateien in der Liste oder im Verzeichnis verwenden.

Partitionierung für gleichzeitiges Lesen

Um die Reihenfolge der Zeilen beizubehalten, wenn mehrere Partitionen aus einer Dateiquelle lesen, konfigurieren Sie eine Partitionierung für gleichzeitiges Lesen. Standardmäßig behält der Datenintegrationsdienst die Zeilenreihenfolge nicht bei.

Konfigurieren Sie Partitionierung für gleichzeitiges Lesen in den Leseigenschaften des Einfachdatei-Datenobjekts. Wählen Sie in der Ansicht **Lesen** die Ausgabeumwandlung zum Bearbeiten von Laufzeiteigenschaften aus, die vom Datenintegrationsdienst beim Lesen von Daten aus der Datei verwendet werden.

Wählen Sie eine der folgenden Optionen für die Eigenschaft **Partitionierung für gleichzeitiges Lesen** aus:

Durchsatz optimieren

Der Datenintegrationsdienst behält die Reihenfolge der Zeilen nicht bei, wenn mehrere Partitionen aus einer einzelnen Dateiquelle lesen. Verwenden Sie diese Option, wenn die Reihenfolge, in der mehrere Partitionen aus einer Dateiquelle lesen, keine Rolle spielt.

Relative Reihenfolge beibehalten

Behält die Sortierreihenfolge der Eingabezeilen bei, die von jeder Partition gelesen werden.

In der folgenden Tabelle wird eine Beispielsortierreihenfolge einer Dateiquelle mit 10 Zeilen dargestellt, die von zwei Partitionen gelesen wird:

Partition	Gelesene Zeilen
Partition Nr. 1	1,3,5,8,9
Partition Nr. 2	2,4,6,7,10

Absolute Reihenfolge beibehalten

Behält die Sortierreihenfolge aller Eingabezeilen bei, die von allen Partitionen gelesen werden. In einem Pass-Through-Mapping mit passiven Umwandlungen stimmt die Reihenfolge der Zeilen, die in das Ziel geschrieben werden, mit der Reihenfolge der Eingabezeilen überein.

In der folgenden Tabelle wird eine Beispielsortierreihenfolge einer Dateiquelle mit 10 Zeilen dargestellt, die von zwei Partitionen gelesen wird:

Partition	Gelesene Zeilen
Partition Nr. 1	1,2,3,4,5
Partition Nr. 2	6,7,8,9,10

Partitionierte relationale Quellen

Wenn ein Mapping mit aktivierter Partitionierung als Quelle ein IBM DB2 für LUW- oder ein relationales Oracle-Datenobjekt enthält, kann der Datenintegrationsdienst mehrere Partitionen zum Lesen der relationalen Quelle verwenden.

Hinweis: Wenn ein Mapping ein relationales Datenobjekt enthält, bei dem es sich nicht um DB2 für LUW oder Oracle als Quelle handelt, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Der Datenintegrationsdienst ruft beim DB2 für LUW- oder Oracle-Datenbanksystem Partitionsinformationen ab. Wenn die Quelltabellen Datenbankpartitionierung unterstützen, kann der Datenintegrationsdienst mehrere Threads zum Lesen der partitionierten Daten aus den entsprechenden Knoten in der Datenbank verwenden. Der Datenintegrationsdienst erstellt eine SQL-Abfrage für jeden Reader-Thread.

Der Datenintegrationsdienst verwendet den Parallelismuswert, um die Anzahl der Threads zu erstellen, die aus einer DB2 für LUW- oder relationalen Oracle-Quelle lesen. Sie können die Leistung verbessern, wenn der Parallelismuswert größer oder gleich der Anzahl der Datenbankpartitionen ist. Für Oracle-Quellen, die gemischte Partitionierung verwenden, kann die Leistung verbessert werden, wenn die Anzahl der Threads mit der Anzahl der Datenbankunterpartitionen übereinstimmt. Wenn eine Oracle-Quelle beispielsweise drei Partitionen und zwei Unterpartitionen für jede Partition enthält, können Sie die Leistung verbessern, wenn der Parallelismuswert auf sechs festgelegt ist.

Die Anzahl der Reader-Threads, die vom Datenintegrationsdienst verwendet werden, richtet sich nach den folgenden Situationen:

Die Anzahl der Datenbankpartitionen stimmt mit dem Parallelismuswert überein.

Der Datenintegrationsdienst verwendet alle vom Parallelismuswert definierten Reader-Threads. Der Datenintegrationsdienst verteilt eine Datenbankpartition an jeden Reader-Thread.

Die Anzahl der Datenbankpartitionen ist kleiner als der Parallelismuswert.

Wenn die Anzahl der Datenbankpartitionen den Parallelismuswert unterschreitet, verwendet der Datenintegrationsdienst diese Thread-Anzahl, um aus der Quelle zu lesen. Die verbleibenden Mapping-Pipeline-Stages können die Anzahl der Partitionen nutzen, die vom Parallelismuswert definiert werden.

Beispiel: Eine Oracle-Quelle hat zwei Datenbankpartitionen und der Parallelismuswert ist auf sechs festgelegt. Der Datenintegrationsdienst erstellt sechs Reader-Threads, verwendet aber zwei der Threads, um aus der Quelle zu lesen. Der Datenintegrationsdienst kann sechs Threads in jeder der verbleibenden Mapping-Pipeline-Stages verwenden.

Die Anzahl der Datenbankpartitionen überschreitet den Parallelismuswert.

Wenn die Anzahl der Datenbankpartitionen den Parallelismuswert überschreitet, verwendet der Datenintegrationsdienst alle vom Parallelismuswert definierten Reader-Threads. Der Datenintegrationsdienst verteilt mehrere Datenbankpartitionen auf bestimmte Reader-Threads.

Beispiel: Eine DB2 für LUW-Quelle hat fünf Datenbankpartitionen und der Parallelismuswert ist auf drei festgelegt. Der Datenintegrationsdienst erstellt drei Reader-Threads. Der Datenintegrationsdienst verteilt zwei Datenbankpartitionen auf den ersten und den zweiten Reader-Thread. Der Dienst verteilt eine Datenbankpartition auf den dritten Reader-Thread.

Keine Datenbankpartitionen.

Wenn die Quelle keine Datenbankpartitionen aufweist, verwendet der Datenintegrationsdienst einen Thread zum Lesen aus der Quelle. Die verbleibenden Mapping-Pipeline-Stages können die Anzahl der Partitionen nutzen, die vom Parallelismuswert definiert werden.

Relationale Verbindungstypen für Partitionierung

Der Datenintegrationsdienst kann Partitionen für ein Mapping erstellen, das ein DB2 für LUW- oder relationales Oracle-Datenobjekt als Quelle enthält, wenn das Datenobjekt eine DB2 für LUW- oder Oracle-Verbindung verwendet.

Sie können die folgenden Verbindungstypen zum Herstellen einer Verbindung zu einer DB2 für LUW- oder Oracle-Datenbank verwenden:

- DB2 für LUW-Verbindung oder Oracle-Verbindung
- JDBC-Verbindung
- ODBC-Verbindung

Um mehrere DB2 für LUW- oder relationale Oracle-Quellen zu verwenden, muss das relationale Datenobjekt eine DB2 für LUW- oder Oracle-Verbindung verwenden.

Wenn das DB2 für LUW- oder relationale Datenobjekt eine JDBC- oder ODBC-Verbindung verwendet, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

SQL-Abfragen für partitionierte relationale Quellen

Wenn der Datenintegrationsdienst zum Lesen einer relationalen Quelle mehrere Partitionen verwendet, erzeugt er eine SQL-Abfrage für jeden Reader-Thread.

Wenn die Datenbankquelle mehr Datenbankpartitionen als der Parallelismuswert aufweist, verteilt der Datenintegrationsdienst die Daten auf die Reader-Threads. Der Datenintegrationsdienst kann SQL-Abfragen

erzeugen, die aus mehreren Datenbankpartitionen lesen. Wenn eine Oracle-Quelle Unterpartitionen enthält, kann der Datenintegrationsdienst SQL-Abfragen erzeugen, die aus mehreren Datenbankunterpartitionen lesen.

Beispiel für DB2 für LUW- oder Oracle-Quellen

Der Parallelismuswert ist drei und die relationale Quelle hat fünf Datenbankpartitionen. Wenn der Datenintegrationsdienst SQL-Abfragen in den Datenbankpartitionen ausführt, empfangen der erste und zweite Reader-Thread Daten aus zwei Datenbankpartitionen. Der dritte Reader-Thread empfängt Daten aus einer Datenbankpartition. In diesem Beispiel wurde für das relationale Datenobjekt nicht die Option „Select Distinct“ aktiviert.

Wenn Sie eine DB2 für LUW-Quelle verwenden, erzeugt der Datenintegrationsdienst SQL-Anweisungen, die mit den folgenden Anweisungen für den ersten Reader-Thread vergleichbar sind:

```
SELECT <column list> FROM <table name>
WHERE (nodenumber(<column 1>)=0 OR nodenumber(<column 1>) = 3)
```

Wenn Sie eine Oracle-Quelle verwenden, erzeugt der Datenintegrationsdienst SQL-Anweisungen, die mit den folgenden Anweisungen für den ersten Reader-Thread vergleichbar sind:

```
SELECT <column list> FROM <table name> PARTITION <database_partition1 name> UNION ALL
SELECT <column list> FROM <table name> PARTITION <database_partition4 name> UNION ALL
```

Beispiel für eine Oracle-Quelle mit Unterpartitionen

Eine Oracle-Quelle hat fünf Partitionen (1 bis 5) und zwei Unterpartitionen (a und b) je Partition. Der Parallelismuswert ist drei. Der erste Reader-Thread empfängt Daten aus vier Datenbankunterpartitionen. Der zweite und dritte Reader-Thread empfangen Daten aus drei Datenbankunterpartitionen. In diesem Beispiel wurde für das relationale Datenobjekt nicht die Option „Select Distinct“ aktiviert.

Der Datenintegrationsdienst erzeugt SQL-Anweisungen, die mit den folgenden Anweisungen für den ersten Reader-Thread vergleichbar sind:

```
SELECT <column list> FROM <table name> SUBPARTITION <database_subpartition1_a name>
UNION ALL
SELECT <column list> FROM <table name> SUBPARTITION <database_subpartition1_b name>
UNION ALL
SELECT <column list> FROM <table name> SUBPARTITION <database_subpartition4_a name>
UNION ALL
SELECT <column list> FROM <table name> SUBPARTITION <database_subpartition4_b name>
UNION ALL
```

Regeln und Richtlinien für Partitionen der relationalen Quelle

Bestimmte Regeln und Richtlinien gelten, wenn Sie Partitionierung für ein Mapping aktivieren, das ein relationales Datenobjekt als Quelle enthält.

Beachten Sie die folgenden Regeln und Richtlinien, wenn Sie Partitionierung für ein Mapping mit einer relationalen Quelle aktivieren:

- Der Datenintegrationsdienst erstellt keine Partitionen für das gesamte Mapping und verwendet in folgenden Situationen einen Thread für jede Mapping-Pipeline-Stage:
 - Das Mapping enthält ein relationales Datenobjekt als Quelle, das sich von der DB2 für LUW- oder Oracle-Quelle unterscheidet.
 - Das Mapping enthält ein DB2 für LUW- oder relationales Oracle-Datenobjekt, das eine JDBC- oder ODBC-Verbindung verwendet.

- Sie sortieren Quellzeilen im relationalen Datenobjekt. Um Quellzeilen zu sortieren, können Sie die einfache Abfrage verwenden, um die Ports auszuwählen, nach denen sortiert werden soll. Alternativ dazu können Sie die erweiterte Abfrage verwenden, um eine benutzerdefinierte SQL-Abfrage mit einer ORDER BY-Klausel zu erstellen.
- Der Datenintegrationsdienst verwendet einen Thread zum Lesen der Quelle, kann aber in folgenden Situationen mehrere Partitionen für die verbleibenden Mapping-Pipeline-Stages erstellen:
 - Das relationale Datenobjekt fügt mehrere Tabellen in derselben Quelldatenbank zusammen.
 - Das Mapping verschiebt die Umwandlungslogik in die Quelldatenbank.
 - Sie verwenden die erweiterte Abfrage im relationalen Datenobjekt, um eine benutzerdefinierte SQL-Abfrage zu erstellen, die keine ORDER BY-Klausel enthält.

Wenn jedoch der maximale Parallelismus für das Mapping auf „Auto“ eingestellt ist und das Mapping eine relationale Quelle enthält, berechnet der Datenintegrationsdienst den Parallelismuswert in diesen Situationen immer als 1. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten. Um mehrere Partitionen für die verbleibenden Mapping-Pipeline-Stages zu verwenden, überschreiben Sie den maximalen Parallelismus für das Mapping mit einem Ganzzahlwert größer als 1.

- Wenn Sie die einfache Abfrage im relationalen Datenobjekt verwenden, um Hinweise zu erstellen, eindeutige Werte auszuwählen oder einen Quellfilter einzugeben, kann der Datenintegrationsdienst mehrere Partitionen zum Lesen der Quelle verwenden. Der Datenintegrationsdienst fügt der für jede Partition erzeugten SQL-Abfrage Hinweise, eindeutige Werte oder Quellfilter hinzu.

Partitionierte Einfachdateiziele

Wenn ein für die Partitionierung aktiviertes Mapping ein Einfachdateiziel enthält, verwendet der Datenintegrationsdienst mehrere Partitionen zum Schreiben in das Dateiziel.

Der Datenintegrationsdienst verwendet den Parallelismuswert, um die Anzahl der Partitionen festzulegen, die in das Einfachdateiziel schreiben. Wenn der Parallelismuswert beispielsweise auf vier gesetzt ist, erstellt der Datenintegrationsdienst vier Partitionen zum Schreiben in das Einfachdateiziel.

Der Datenintegrationsdienst kann Partitionen für die folgenden Einfachdatei-Zieltypen erstellen:

- Direkte Datei
- Befehl
- Datei im verteilten Hadoop-Dateisystem

Wenn mehrere Threads in das Einfachdateiziel schreiben, gehört es zu den Standardaufgaben des Datenintegrationsdiensts, die Zielausgabe gleichzeitig in eine separate Datei für jede Partition zu schreiben. Der Datenintegrationsdienst verwendet folgendes Format, um jede von einer Partition geschriebene Datei zu benennen:

```
<output_file_name><partition_number>.out
```

Drei Partitionen schreiben unter Umständen in Dateien mit der Bezeichnung MyOutput1.out, MyOutput2.out und MyOutput3.out.

Hinweis: Der Datenintegrationsdienst verwendet eine Partition, um die Sortiererumwandlung und alle nachgelagerten Mapping-Pipeline-Stages zu verarbeiten. Wenn ein Mapping mit aktivierter Partitionierung eine Sortiererumwandlung und ein Einfachdateiziel enthält, verwendet der Datenintegrationsdienst einen Thread zum Schreiben in die Einfachdatei. Der Dienst hängt die Partitionsnummer 1 an den Dateinamen an. Der Datenintegrationsdienst schreibt beispielsweise in eine Einfachdatei mit der Bezeichnung MyOutput1.out.

Sie können den Datenintegrationsdienst zum Schreiben in eine Zusammenführungsdatei konfigurieren, die die Zielausgabe für alle Partitionen enthält. Sie können auch einen Befehl verwenden, um Zusammenführungsdaten für alle Zielpartitionen zu verarbeiten.

Zusammenführungsoptionen für partitionierte Dateiziele

Sie können Zieldaten für die Partitionen in einem Mapping zusammenführen. Wenn Sie Zieldaten zusammenführen, erstellt der Datenintegrationsdienst eine Zusammenführungsdatei für alle Zielpartitionen.

Konfigurieren Sie die Zusammenführungsoptionen in den Schreibeigenschaften des Einfachdatei-Datenobjekts. Wählen Sie in der Ansicht **Schreiben** die Eingabeumwandlung zur Bearbeitung der Laufzeiteigenschaften aus, die vom Datenintegrationsdienst beim Schreiben von Daten in die Datei verwendet werden.

Wählen Sie eine der folgenden Optionen für den Zusammenführungstyp aus:

Keine Zusammenführung

Der Datenintegrationsdienst schreibt die Zielausgabe gleichzeitig in eine separate Datei für jede Partition.

Sequenziell

Der Datenintegrationsdienst erstellt eine Ausgabedatei für jede Partition und führt diese Dateien dann am Ende des Mappings in einer einzelnen Zusammenführungsdatei zusammen. Der Datenintegrationsdienst fügt der Zusammenführungsdatei nacheinander die Ausgabedaten für jede Partition hinzu. Der Datenintegrationsdienst erstellt die einzelnen Zieldateien mithilfe des Namens und der Verzeichniswerte der Ausgabedatei.

Dateiliste

Der Datenintegrationsdienst erstellt eine Zieldatei für jede Partition sowie eine Dateiliste mit den Pfaden der einzelnen Dateien. Der Datenintegrationsdienst erstellt die einzelnen Zieldateien mithilfe des Namens und der Verzeichniswerte der Ausgabedatei. Wenn Sie die Zieldateien in das Zusammenführungsverzeichnis oder ein Verzeichnis unterhalb des Zusammenführungsverzeichnisses schreiben, enthält die Dateiliste relative Pfade. Andernfalls enthält die Dateiliste absolute Pfade. Verwenden Sie diese Datei als Quelldatei, wenn Sie die Zieldateien als Quelldateien in einem anderen Mapping verwenden.

Gleichzeitig

Der Datenintegrationsdienst schreibt die Daten für alle Zielpartitionen gleichzeitig in die Zusammenführungsdatei. Es werden keine temporären Dateien für jede Partition erstellt. Da der Datenintegrationsdienst gleichzeitig in die Zusammenführungsdatei für alle Partitionen schreibt, wird die Sortierreihenfolge der Daten in der Zusammenführungsdatei unter Umständen nicht eingehalten.

Wenn Sie die Einfachdatei-Datenobjekte für die Zusammenführung von Zieldaten konfigurieren, können Sie die Standardwerte für die Eigenschaften **Verzeichnis der Zusammenführungsdatei** und **Name der Zusammenführungsdatei** optional bearbeiten.

Wenn Sie die Einfachdatei-Datenobjekte für die Zusammenführung von Zieldaten konfigurieren und der Datenintegrationsdienst keine Partitionen für das Ziel erstellt, werden die Zusammenführungsoptionen vom Datenintegrationsdienst ignoriert. Der Dienst schreibt in die in den Eigenschaften **Name der Ausgabedatei** und **Verzeichnis der Ausgabedatei** definierte Datei.

Befehle für partitionierte Dateiziele

Sie können einen Befehl verwenden, um Zieldaten für eine einzelne Partition oder Zusammenführungsdaten für alle Zielpartitionen in einem Mapping zu verarbeiten. Der Datenintegrationsdienst sendet die Daten an

einen Befehl oder einen Zusammenführungsbefehl statt an eine Einfachdatei oder eine Zusammenführungsdatei.

Konfigurieren Sie den Befehl, der Daten für Partitionen in den Schreibeigenschaften des Einfachdatei-Datenobjekts verarbeitet. Wählen Sie in der Ansicht **Schreiben** die Eingabeumwandlung zur Bearbeitung der Laufzeiteigenschaften aus, die vom Datenintegrationsdienst beim Schreiben von Daten in die Datei verwendet werden.

Verwenden Sie unter UNIX einen beliebigen gültigen UNIX-Befehl oder ein beliebiges Shell-Skript. Verwenden Sie unter Windows einen beliebigen gültigen DOS-Befehl oder eine Batchdatei.

Sie können einen Befehl zum Verarbeiten der folgenden Typen von Zieldaten verwenden:

Daten für eine einzelne Partition

Geben Sie einen Befehl ein, der von jedem Writer-Thread einzeln ausgeführt wird. Jeder Thread sendet die Zieldaten an den Befehl, wenn das Mapping ausgeführt wird. Jeder Thread führt denselben Befehl mit einem anderen Datensatz aus.

Wenn Sie den Befehl eingeben, müssen Sie das Betriebssystem beachten, auf dem der Datenintegrationsdienst ausgeführt wird. Wenn Sie beispielsweise den Befehl `cat > C:\data\MyOutput.txt` eingeben, schreiben mehrere Threads in dieselbe Datei, wodurch ein Betriebssystemfehler ausgelöst werden kann. Wenn Sie den Befehl `cat >> C:\data\MyOutput.txt` eingeben, hängen mehrere Threads Daten an dieselbe Datei an, wodurch ein Betriebssystemfehler weniger wahrscheinlich wird.

Um die Zieldaten für eine einzelne Partition an einen Befehl zu senden, wählen Sie „Befehl“ für den **Ausgabety**p und „Keine Zusammenführung“ für den **Zusammenführungstyp** aus. Geben Sie einen Befehl für die Eigenschaft **Befehl** ein.

Zusammenführen von Daten für alle Partitionen

Geben Sie einen Zusammenführungsbefehl ein, der die Zusammenführungsdaten für alle Writer-Threads verarbeitet. Der Datenintegrationsdienst muss einen gleichzeitigen Zusammenführungstyp für einen Befehl verwenden, um Zusammenführungsdaten zu verarbeiten. Jeder Thread sendet gleichzeitig Zieldaten an den Zusammenführungsbefehl, wenn das Mapping ausgeführt wird. Der Zusammenführungsbefehl wird einmal mit allen Daten ausgeführt. Der Befehl behält die Reihenfolge der Zieldaten unter Umständen nicht bei.

Um Zusammenführungsdaten für alle Partitionen an einen Zusammenführungsbefehl zu senden, wählen Sie „Befehl“ für den **Ausgabety**p und „Gleichzeitig“ für den **Zusammenführungstyp** aus. Geben Sie einen Befehl für die Eigenschaft **Zusammenführungsbefehl** ein.

Partitionierte relationale Ziele

Wenn ein Mapping mit aktivierter Partitionierung als Ziel ein relationales IBM DB2 für LUW- oder Oracle-Datenobjekt enthält, kann der Datenintegrationsdienst mehrere Partitionen zum Schreiben in das relationale Ziel verwenden.

Hinweis: Wenn ein Mapping ein relationales Datenobjekt enthält, bei dem es sich nicht um DB2 für LUW oder Oracle als Ziel handelt, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Wenn das Ziel Oracle ist, verwendet der Datenintegrationsdienst den Parallelismuswert zum Erstellen der Anzahl der Threads, die in das Oracle-Ziel schreiben. Der Datenintegrationsdienst verwendet alle Writer-Threads. Beispiel: Wenn der Parallelismuswert vier ist, erstellt und verwendet der Datenintegrationsdienst

vier Partitionen zum Schreiben in ein Oracle-Ziel. Wenn die relationale Tabelle Partitionen aufweist, leitet die Datenbank die Daten an die richtigen Partitionen weiter.

Wenn DB2 für LUW als Ziel fungiert, ruft der Datenintegrationsdienst im DB2 für LUW-System Partitionsinformationen ab. Der Datenintegrationsdienst lädt die partitionierten Daten in die entsprechenden Knoten in der Zieldatenbank. Der Datenintegrationsdienst verwendet den Parallelismuswert, um die Anzahl der Threads zu erstellen, die in das DB2 für LUW-Ziel schreiben.

Die Anzahl der Writer-Threads, die vom Datenintegrationsdienst für ein DB2 für LUW-Ziel verwendet werden, richtet sich nach den folgenden Situationen:

Die Anzahl der Datenbankpartitionen stimmt mit dem Parallelismuswert überein.

Der Datenintegrationsdienst verwendet alle durch den Parallelismuswert definierten Writer-Threads. Jeder Writer-Thread schreibt in eine Datenbankpartition.

Die Anzahl der Datenbankpartitionen ist kleiner als der Parallelismuswert.

Wenn die Anzahl der Datenbankpartitionen im DB2 für LUW-Ziel den Parallelismuswert unterschreitet, verwendet der Datenintegrationsdienst diese Thread-Anzahl, um in das Ziel zu schreiben. Beispiel: Ein DB2 für LUW-Ziel hat zwei Datenbankpartitionen und der Parallelismuswert ist auf sechs festgelegt. Der Datenintegrationsdienst erstellt sechs Writer-Threads, verwendet aber zwei der Threads, um in die beiden Datenbankpartitionen zu schreiben.

Die Anzahl der Datenbankpartitionen überschreitet den Parallelismuswert.

Wenn die Anzahl der Datenbankpartitionen im DB2 für LUW-Ziel den Parallelismuswert überschreitet, verwendet der Datenintegrationsdienst alle vom Parallelismuswert definierten Writer-Threads. Der Datenintegrationsdienst verteilt mehrere Datenbankpartitionen auf bestimmte Writer-Threads.

Beispiel: Ein DB2 für LUW-Ziel hat fünf Datenbankpartitionen und der Parallelismuswert ist auf drei festgelegt. Der Datenintegrationsdienst erstellt drei Writer-Threads. Der Datenintegrationsdienst verteilt zwei Datenbankpartitionen auf den ersten und den zweiten Writer-Thread. Der Dienst verteilt eine Datenbankpartition auf den dritten Writer-Thread.

Keine Datenbankpartitionen.

Wenn das DB2 für LUW-Ziel keine Datenbankpartitionen aufweist, verwendet der Datenintegrationsdienst alle durch den Parallelismuswert definierten Writer-Threads. Sie können die Ladeleistung jedoch verbessern, wenn das Ziel über Datenbankpartitionen verfügt. In diesem Fall stellt jeder Writer-Thread eine Verbindung zu dem DB2 für LUW-Knoten her, der die Datenbankpartition enthält. Da sich die Writer-Threads nicht alle mit einem einzelnen Master-Knoten verbinden, sondern eine Verbindung zu unterschiedlichen DB2 für LUW-Knoten herstellen, wird die Leistung erhöht.

Relationale Verbindungstypen für Partitionierung

Der Datenintegrationsdienst kann Partitionen für ein Mapping erstellen, das ein DB2 für LUW- oder ein relationales Oracle-Datenobjekt als Ziel enthält, wenn das Datenobjekt eine DB2 für LUW- oder Oracle-Verbindung verwendet.

Sie können die folgenden Verbindungstypen zum Herstellen einer Verbindung zu einer DB2 für LUW- oder Oracle-Datenbank verwenden:

- DB2 für LUW-Verbindung oder Oracle-Verbindung
- JDBC-Verbindung
- ODBC-Verbindung

Um mehrere Partitionen zum Schreiben in ein DB2 für LUW- oder relationales Oracle-Ziel zu verwenden, muss das relationale Datenobjekt eine DB2 für LUW- oder Oracle-Verbindung verwenden.

Wenn das DB2 für LUW- oder relationale Datenobjekt eine JDBC- oder ODBC-Verbindung verwendet, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Regeln und Richtlinien für Partitionen des relationalen Ziels

Bestimmte Regeln und Richtlinien gelten, wenn Sie Partitionierung für ein Mapping aktivieren, das ein relationales Datenobjekt als Ziel enthält.

Der Datenintegrationsdienst verwendet die folgenden Regeln und Richtlinien, wenn Sie Partitionierung für ein Mapping mit einem relationalen Ziel aktivieren:

- Wenn ein Mapping ein relationales Datenobjekt enthält, bei dem es sich nicht um DB2 für LUW oder Oracle als Ziel handelt, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.
- Wenn das Mapping ein DB2 für LUW- oder relationales Oracle-Datenobjekt enthält, das eine JDBC- oder ODBC-Verbindung verwendet, erstellt der Datenintegrationsdienst keine Partitionen für das gesamte Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.
- Aktivieren Sie hohe Genauigkeit für das Mapping, wenn es sich bei dem Partitionsschlüssel einer DB2 für LUW-Zieltabelle um eine Dezimalspalte handelt. Das Mapping kann im Datenintegrationsdienst unter Umständen fehlschlagen, wenn ein Partitionsschlüssel eine Dezimalspalte darstellt und „Hohe Genauigkeit“ für das Mapping nicht aktiviert wurde.

Partitionierte Umwandlungen

Wenn ein Mapping mit aktivierter Partitionierung eine Umwandlung enthält, die Partitionierung unterstützt, verwendet der Datenintegrationsdienst mehrere Partitionen zum Umwandeln der Daten.

Der Datenintegrationsdienst verwendet den Parallelismuswert, um die Anzahl der Partitionen festzulegen, die die Umwandlungs-Pipeline-Stage verarbeiten. Wenn der Parallelismuswert beispielsweise auf vier gesetzt ist, erstellt der Datenintegrationsdienst vier Partitionen zum Verarbeiten der Umwandlungs-Pipeline-Stage.

Der Datenintegrationsdienst gibt an, ob der Umwandlung ein zusätzlicher Partitionsunkt hinzugefügt werden muss. Der Datenintegrationsdienst legt weiterhin fest, ob Daten am Partitionsunkt neu verteilt werden müssen. Eine Aggregat-Umwandlung benötigt beispielsweise eine Hash-Partitionierung, um Gruppen von Zeilen zwischen Partitionen neu zu verteilen.

Die folgenden Umwandlungen unterstützen Partitionierung:

- Aggregator
- Ausdruck
- Filter
- Java
- Joiner
- Lookup (verbunden)
- Rang
- Router
- SQL

- Union
- Update-Strategie

Wenn ein Mapping mit aktivierter Partitionierung eine nicht verbundene Lookup-Umwandlung enthält, erstellt der Datenintegrationsdienst keine Partitionen des Lookup-Caches. Der Datenintegrationsdienst kann mehrere Partitionen verwenden, um die verbleibenden Mapping-Pipeline-Stages zu verarbeiten.

Die Sortiererumwandlung unterstützt Partitionierung teilweise. Wenn ein Mapping mit aktivierter Partitionierung eine Sortiererumwandlung enthält, verwendet der Datenintegrationsdienst eine Partition zum Verarbeiten der Sortiererumwandlung und aller nachgelagerten Mapping-Pipeline-Stages. Der Datenintegrationsdienst kann mehrere Partitionen verwenden, um vorangehende Mapping-Pipeline-Stages zu verarbeiten. Wenn ein Mapping mit aktivierter Partitionierung eine Sortiererumwandlung und ein Einfachdateiziel enthält, hängt der Datenintegrationsdienst die Partitionsnummer 1 an den Dateinamen an. Der Datenintegrationsdienst schreibt beispielsweise in eine Einfachdatei mit der Bezeichnung MyOutput1.out.

Partitionierung wird von allen übrigen Umwandlungen nicht unterstützt. Wenn ein Mapping mit aktivierter Partitionierung eine Umwandlung enthält, die Partitionierung nicht unterstützt, erstellt der Datenintegrationsdienst keine Partitionen für das Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Einschränkungen für partitionierte Umwandlungen

Bestimmte Umwandlungen, die Partitionierung unterstützen, erfordern gesonderte Konfigurationen. Wenn ein Mapping mit aktivierter Partitionierung eine Umwandlung mit einer nicht unterstützten Konfiguration enthält, erstellt der Datenintegrationsdienst keine Partitionen für das Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Die folgenden Umwandlungen benötigen bestimmte Konfigurationen zur Unterstützung der Partitionierung:

- Aggregat-Umwandlungen müssen unsortierte Eingaben verwenden und ein Group-by-Port enthalten.
- Für Java-Umwandlungen muss die Eigenschaft „Ohne Status“ aktiviert sein.
- Joiner-Umwandlungen müssen unsortierte Eingaben verwenden und eine Join-Bedingung enthalten, die einen Gleichheitsoperator verwendet. Wenn die Join-Bedingung mehrere Bedingungen enthält, müssen die Bedingungen mithilfe des AND-Operators verbunden werden.
- Rang-Umwandlungen müssen ein Group-by-Port enthalten.

Cache-Partitionierung für Umwandlungen

Bei der Cache-Partitionierung wird ein separater Cache für jede Partition erstellt, die eine Aggregat-, Joiner-, Rang- oder Lookup-Umwandlung verarbeitet. Während der Cache-Partitionierung speichert jede Partition andere Daten in einem separaten Cache. Jeder Cache enthält die Zeilen, die von der jeweiligen Partition benötigt werden.

Die Cache-Partitionierung erhöht die Mapping-Leistung, da jede Partition gleichzeitig einen separaten Cache abfragt. Wenn der Datenintegrationsdienst Partitionen für ein Mapping erstellt, verwendet er immer Cache-Partitionierung für partitionierte Aggregat-, Joiner- und Rangumwandlungen. Der Datenintegrationsdienst verwendet unter Umständen Cache-Partitionierung für partitionierte Lookup-Umwandlungen.

Der Datenintegrationsdienst verwendet Cache-Partitionierung für verbundene Lookup-Umwandlungen unter folgenden Bedingungen:

- Die Lookup-Bedingung enthält nur Gleichheitsoperatoren.

- Wenn die verbundene Lookup-Umwandlung nach Daten in einer relationalen Tabelle sucht, ist die Datenbank für Vergleiche unter Berücksichtigung der Groß-/Kleinschreibung konfiguriert.

Wenn die Lookup-Bedingung beispielsweise einen Zeichenfolge-Port enthält und die Datenbank nicht für Vergleiche unter Berücksichtigung der Groß-/Kleinschreibung konfiguriert ist, verwendet der Datenintegrationsdienst keine Cache-Partitionierung.

Der Datenintegrationsdienst verwendet keine Cache-Partitionierung für nicht verbundene Lookup-Umwandlungen.

Wenn der Datenintegrationsdienst keine Cache-Partitionierung für eine Lookup-Umwandlung verwendet, nutzen alle Partitionen der Lookup-Umwandlung denselben Cache. Jede Partition fragt denselben Cache fortlaufend ab.

Cache-Größe für partitionierte Caches

Wenn der Datenintegrationsdienst Cache-Partitionierung für Aggregat-, Joiner-, Rang- und Lookup-Umwandlungen verwendet, verteilt der Dienst die Cache-Größe auf die Partitionen.

Sie konfigurieren die Cache-Größe in den erweiterten Eigenschaften der Umwandlung. Sie können einen numerischen Wert in Byte eingeben oder „Automatisch“ auswählen, damit der Datenintegrationsdienst die Cache-Größe zur Laufzeit festlegt.

Wenn Sie einen numerischen Wert eingeben, teilt der Datenintegrationsdienst die Cache-Größe zur Laufzeit auf die Anzahl der Umwandlungspartitionen auf. Sie legen die Cache-Größe für die Umwandlung beispielsweise auf 2.000.000 Byte fest. Der Datenintegrationsdienst verwendet vier Partitionen zum Verarbeiten der Umwandlung. Der Dienst teilt den Wert der Cache-Größe so auf, dass jede Partition maximal 500.000 Byte als Cache-Größe verwendet.

Bei Auswahl von „Automatisch“ legt der Datenintegrationsdienst die Cache-Größe für die Umwandlung zur Laufzeit fest. Der Dienst teilt die Cache-Größe dann auf die Anzahl der Umwandlungspartitionen auf.

Umwandlungen, die numerische Funktionen verwenden

Eine partitionierte Umwandlung, die die Funktion CUME, MOVINGSUM oder MOVINGAVG verwendet, gibt unter Umständen nicht dasselbe berechnete Ergebnis mit jeder Mapping-Ausführung zurück. Wenn eine Umwandlung eine dieser Funktionen verwendet, möchten Sie Partitionierung für das Mapping unter Umständen deaktivieren.

Die numerischen Funktionen CUME, MOVINGSUM und MOVINGAVG berechnen laufende Kontostände und Durchschnittswerte auf Zeilenbasis. Wenn eine partitionierte Umwandlung eine dieser Funktionen enthält, verarbeitet jede Partition die Funktion getrennt. Jede Funktion berechnet das Ergebnis, indem eine Teilmenge der Daten und nicht alle Daten herangezogen werden. Deshalb gibt eine partitionierte Umwandlung, die die Funktionen CUME, MOVINGSUM oder MOVINGAVG verwendet, unter Umständen nicht dasselbe berechnete Ergebnis mit jeder Mapping-Ausführung zurück.

Zum Deaktivieren der Partitionierung für das Mapping setzen Sie den maximalen Parallelismus in den Laufzeiteigenschaften des Mappings auf 1.

Umwandlungen, die lokale Variablen verwenden

Eine partitionierte Umwandlung, die eine vom vorherigen Wert der Zeile abhängige lokale Variable verwendet, gibt unter Umständen nicht dasselbe berechnete Ergebnis mit jeder Mapping-Ausführung zurück. Wenn eine

partitionierte Umwandlung eine dieser Variablen verwendet, möchten Sie die Partitionierung für das Mapping unter Umständen deaktivieren.

Lokale Variablen speichern Werte oder Berechnungen zur Verwendung in einem Ausdruck in der Umwandlung. Ist eine lokale Variable vom Wert der vorherigen Zeile abhängig, unterscheiden sich die berechneten Ergebnisse unter Umständen, wenn die Umwandlung von mehreren Partitionen verarbeitet wird.

Eine Umwandlung enthält beispielsweise eine lokale Variable, die sich mit jeder Zeile um eins erhöht. Die Quelldaten enthalten 100 Zeilen. Wenn die Umwandlung nicht partitioniert ist, liegt das berechnete Endergebnis der lokalen Variable bei 100. Wenn zwei Partitionen die Umwandlung verarbeiten, erhöht jede Partition separat den Wert der lokalen Variable. Das berechnete Endergebnis der lokalen Variable lautet 50 für jede Partition.

Zum Deaktivieren der Partitionierung für das Mapping setzen Sie den maximalen Parallelismus in den Laufzeiteigenschaften des Mappings auf 1.

Adress-Validierer- und Match-Umwandlungen

Wenn ein Mapping mit aktivierter Partitionierung eine Adress-Validierer- oder Match-Umwandlung enthält, erstellt der Datenintegrationsdienst keine Partitionen für das Mapping. Der Datenintegrationsdienst kann jedoch mehrere Threads zum Verarbeiten der Umwandlung verwenden, wenn die Anzahl der Ausführungsinstanzen für die Umwandlung konfiguriert wurde.

Die Adress-Validierer- und Match-Umwandlung bieten keine Unterstützung für die Partitionierung. Wenn ein Mapping eine dieser Umwandlungen enthält, ignoriert der Datenintegrationsdienst den maximalen Parallelismuswert. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Sie können jedoch die Adress-Validierer- oder die Match-Umwandlung konfigurieren, um mehrere Ausführungsinstanzen zu erhalten. Die Anzahl der Ausführungsinstanzen bestimmt die Anzahl der Threads, die vom Datenintegrationsdienst zum Verarbeiten der Umwandlung zur Laufzeit verwendet werden. In diesem Fall verwendet der Datenintegrationsdienst mehrere Threads zum Verarbeiten der Umwandlung. Der Datenintegrationsdienst verwendet einen Thread zum Verarbeiten aller verbleibenden Mapping-Pipeline-Stages.

Mappings, die Quellzeilen begrenzen

Wenn ein Mapping mit aktivierter Partitionierung die Quellzeilen begrenzt, kann der Datenintegrationsdienst Partitionen für das Mapping basierend auf Pushdown-Konfigurationen erstellen.

Wenn ein Mapping mit aktivierter Partitionierung die Anzahl der aus der Quelle gelesenen Zeilen begrenzt, erstellt der Datenintegrationsdienst Partitionen basierend auf den folgenden Pushdown-Konfigurationen:

Das Mapping verschiebt die Umwandlungslogik in die Quelle.

Wenn das Mapping die Umwandlungslogik in die Quelldatenbank verschiebt, erstellt der Datenintegrationsdienst keine Partitionen für die Quelle. Der Datenintegrationsdienst verwendet einen Thread zum Lesen der Quelle und der Umwandlungen, die in die Quelldatenbank verschoben werden. Der Datenintegrationsdienst kann mehrere Partitionen für die verbleibenden Mapping-Pipeline-Stages erstellen.

Das Mapping verschiebt keine Umwandlungslogik in die Quelle.

Wenn das Mapping keine Umwandlungslogik in die Quelldatenbank verschiebt, erstellt der Datenintegrationsdienst keine Partitionen für das Mapping. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

Deaktivieren der Partitionierung für ein Mapping

Standardmäßig ist der maximale Parallelismus für jedes Mapping auf „Auto“ gesetzt. Sie können den maximalen Parallelismuswert für ein Mapping ändern, um die Partitionierung für das Mapping zu deaktivieren.

Sie möchten unter Umständen die Partitionierung für ein Mapping aus folgenden Gründen deaktivieren:

Das Mapping verarbeitet ein kleines Dataset.

Wenn das Mapping ein kleines Dataset verarbeitet, übersteigt der Aufwand zur Erstellung mehrerer Threads unter Umständen die Vorteile der parallelen Verarbeitung. Ein Mapping liest beispielsweise eine Einfachdatei mit 50 Zeilen. Maximaler Parallelismus ist für den Datenintegrationsdienst-Prozess auf 10 und für das Mapping auf „Auto“ gesetzt. Der Datenintegrationsdienst berechnet den Parallelismuswert als 10 und verwendet 10 Reader-Threads, wobei jeder Thread fünf Zeilen liest. In diesem Fall haben die zusätzlichen Reader-Threads keine Leistungssteigerung zur Folge.

Eine Umwandlung verwendet eine numerische Funktion.

Eine partitionierte Umwandlung, die die numerische Funktion CUME, MOVINGSUM oder MOVINGAVG verwendet, gibt unter Umständen nicht dieselben berechneten Ergebnisse mit jeder Mapping-Ausführung zurück.

Eine Umwandlung verwendet lokale Variablen, die vom Wert einer vorherigen Zeile abhängen.

Eine partitionierte Umwandlung, die eine dieser lokalen Variablen verwendet, gibt unter Umständen nicht dasselbe berechnete Ergebnis mit jeder Mapping-Ausführung zurück.

Deaktivieren der Partitionierung

Zum Deaktivieren der Partitionierung für ein Mapping setzen Sie den maximalen Parallelismus in den Laufzeiteigenschaften des Mappings auf 1. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten.

1. Öffnen Sie das Mapping.
2. Klicken Sie in der Ansicht **Eigenschaften** auf die Registerkarte **Laufzeit**.
3. Geben Sie 1 für die Eigenschaft **Maximaler Parallelismus** ein.

Überschreiben des maximalen Parallelismus für ein Mapping

Standardmäßig ist der maximale Parallelismus für jedes Mapping auf „Auto“ gesetzt. Sie können den maximalen Parallelismuswert überschreiben, um die Anzahl der Threads zu definieren, die der Datenintegrationsdienst zum Verarbeiten jeder Mapping-Pipeline-Stage erstellt.

Wenn der maximale Parallelismus für den Datenintegrationsdienst-Prozess und das Mapping auf unterschiedliche Ganzzahlwerte festgelegt wurde, verwendet der Datenintegrationsdienst den Mindestwert.

Möglicherweise möchten Sie den maximalen Parallelismus für ein Mapping aus folgenden Gründen überschreiben:

Sie führen ein komplexes Mapping aus, das mehr Threads erzeugt als von der CPU verarbeitet werden können.

Die Gesamtzahl der parallelen Threads, die für die vollständige Mapping-Pipeline ausgeführt werden können, ergibt sich aus dem maximalen Parallelismuswert multipliziert mit der Anzahl der Pipeline-Stages. Jeder Partitionspunkt fügt eine weitere Pipeline-Stage hinzu. Ein komplexes Mapping mit mehreren Aggregat- oder Joiner-Umwandlungen kann unter Umständen viele Pipeline-Stages aufweisen. Eine große Anzahl an Pipeline-Stages kann den Datenintegrationsdienst dazu veranlassen, mehr Threads zu erstellen, als von der CPU verarbeitet werden können.

Die Mapping-Leistung kann mithilfe einer geringeren Anzahl an parallelen Threads für jede Pipeline-Stage optimiert werden.

Wenn ein einzelnes Mapping mit einer geringeren Anzahl an parallelen Threads ausgeführt wird, stehen dem Datenintegrationsdienst für die Ausführung zusätzlicher Jobs mehr Threads zur Verfügung.

Die Anzahl der relationalen Quelldatenbankpartitionen soll nicht die Anzahl der Threads festlegen, die für jede Mapping Pipeline-Stage erstellt wurde.

Ein Mapping enthält eine IBM DB2 für LUW- oder relationale Oracle-Quelle und der maximale Parallelismus für das Mapping ist auf „Auto“ gesetzt. Der Datenintegrationsdienst berechnet den tatsächlichen Parallelismuswert basierend auf der Anzahl der Quelldatenbankpartitionen. Der Datenintegrationsdienst berücksichtigt nicht die Anzahl der Datenbankpartitionen in den Zielen. Um die Leistung bei der Verarbeitung der Umwandlungen und Ziele zu erhöhen, können Sie den maximalen Parallelismus für das Mapping auf einen Wert festlegen, der höher als die Anzahl der Quelldatenbankpartitionen ist.

Beispiel: Ein Mapping enthält eine DB2 für LUW- oder relationale Oracle-Quelle ohne Datenbankpartitionen. Das Mapping enthält ein DB2 für LUW-Ziel mit vier Datenbankpartitionen. Der maximale Parallelismus für den Datenintegrationsdienst-Prozess beträgt fünf und der maximale Parallelismus für das Mapping ist auf „Auto“ eingestellt. Der Datenintegrationsdienst berechnet den Parallelismuswert als eins. Der Datenintegrationsdienst verwendet einen Thread, um jede Mapping-Pipeline-Stage zu verarbeiten. Wenn Sie den maximalen Parallelismus für das Mapping mit vier überschreiben, berechnet der Datenintegrationsdienst den Parallelismuswert als vier. Der Datenintegrationsdienst erstellt vier Threads für jede Mapping-Pipeline-Stage. Der Datenintegrationsdienst verwendet ein Thread zum Lesen der Quelle. Der Datenintegrationsdienst verwendet vier Threads zum Verarbeiten der Umwandlungen und zum Schreiben in das partitionierte DB2 für LUW-Ziel.

Ein Mapping verschiebt die Umwandlungslogik in die Quelldatenbank und die verbleibenden Mapping-Pipeline-Stages sollen mit mehreren Partitionen verarbeitet werden.

Wenn der maximale Parallelismus auf „Auto“ gesetzt ist und das Mapping eine einzelne relationale Quelle enthält, berechnet der Datenintegrationsdienst den Parallelismuswert in den folgenden Situationen immer mit 1:

- Das relationale Datenobjekt fügt mehrere Tabellen in derselben Quelldatenbank zusammen.
- Das Mapping verschiebt die Umwandlungslogik in die Quelldatenbank.
- Sie verwenden die erweiterte Abfrage im relationalen Datenobjekt, um eine benutzerdefinierte SQL-Abfrage zu erstellen, die keine ORDER BY-Klausel enthält.

In all diesen Situationen verwendet der Datenintegrationsdienst einen Thread für jede Mapping-Pipeline-Stage. Um einen Thread zum Lesen der Quelle und mehrerer Threads für die verbleibenden Mapping-Pipeline-Stages zu verwenden, überschreiben Sie den maximalen Parallelismus für das Mapping mit einem Ganzzahlwert größer als 1.

Überschreiben des maximalen Parallelismuswerts

Um den maximalen Parallelismuswert zu überschreiben, stellen Sie den maximalen Parallelismus in den Laufzeiteigenschaften des Mappings auf einen Ganzzahlwert größer als 1 und kleiner als der Wert ein, der für den Datenintegrationsdienst-Prozess festgelegt ist.

1. Öffnen Sie das Mapping.
2. Klicken Sie in der Ansicht **Eigenschaften** auf die Registerkarte **Laufzeit**.
3. Geben Sie für die Eigenschaft **Maximaler Parallelismus** einen Ganzzahlwert größer als 1 und kleiner als der Wert ein, der für den Datenintegrationsdienst-Prozess festgelegt ist.

INDEX

A

- Abhängigkeiten
 - Implizite [19](#)
 - Linkpfad [19](#)
- Adress-Validiererumwandlung
 - Partitionierungseinschränkungen [110](#)
- Aggregatorumwandlung
 - Cache-Partitionierung [108](#)
 - Partitioniert [108](#)
- Anwendungselement
 - Parameterdateien [39](#)
- Ausdrücke
 - Pushdown-Optimierung [77](#)

B

- Benutzerdefinierte Parameter
 - Mapping [31](#)

C

- Cache-Partitionierung
 - Beschreibung [108](#)
 - Cachegröße [109](#)
- Cachegröße
 - Partitionen [109](#)
- CUME, Funktion
 - Partitionierungseinschränkungen [109](#)

E

- Einfachdatei-Quellen
 - Partitioniert [99](#)
- Einfachdatei-Ziele
 - Partitioniert [103](#)
 - Zusammenführen von Partitionen [104](#)
- Export an PowerCenter
 - Release-Kompatibilität [47](#)
 - Einstellen des Kompatibilitätslevels [47](#)
 - Exporteinschränkungen [50](#)
 - Exportieren von Objekten [49](#)
 - Optionen [48](#)
 - Regeln und Richtlinien [52](#)
 - Übersicht [46](#)
- Export an PowerCenter.
 - Fehlersuche [52](#)
- exportieren
 - Nach PowerCenter [46](#)

F

- Fehlersuche
 - Exportieren von Objekten nach PowerCenter [52](#)
- frühe Auswahloptimierung
 - Beschreibung [67](#)
- frühe Projektionsoptimierung
 - Beschreibung [67](#)
- Funktionen
 - Pushdown-Optimierung [77](#)
 - Verfügbar in Quellen [77](#)

I

- IBM DB2 für LUW-Quellen
 - Partitioniert [100](#)
- IBM DB2 für LUW-Ziele
 - Partitioniert [105](#)
- IBM DB2-Quellen
 - Pushdown-Optimierung [75](#)
- Importieren aus PowerCenter
 - Importieren von Einschränkungen [64](#)
 - Importieren von Leistung [64](#)
 - Importieren von Objekten [62](#)
 - Konfliktlösung [55](#)
 - Optionen [62](#)
 - Übersicht [54](#)
 - Umwandlungstypkonvertierung [56](#)

J

- Java-Umwandlung
 - Partitioniert [108](#)
- Joiner-Umwandlung
 - Cache-Partitionierung [108](#)
 - Partitioniert [108](#)

K

- kostenbasierte Optimierung
 - Beschreibung [68](#)

L

- Leistungsoptimierung
 - frühe Auswahloptimierungsmethode [67](#)
 - frühe Projektionsoptimierungsmethode [67](#)
 - kostenbasierte Optimierungsmethode [68](#)
 - Methode zur Optimierung der Verzweigungsbereinigung [67](#)
 - Optimierungsebenen [66](#)
 - Optimierungsmethoden [66](#)
 - Pushdown-Optimierung [71](#)

- Leistungsoptimierung (*Fortsetzung*)
 - Semi-Join Optimierungsmethode [69](#)
 - Vorhersageoptimierungsmethode [68](#)
- Lokale Variablen
 - Partitionierungseinschränkungen [110](#)
- Lookup-Umwandlung
 - Cache-Partitionierung [108](#)

M

- Mapping-Parameter
 - Anwendungsbeispiele [35](#)
 - Benutzerdefiniert [31](#)
 - Erstellungsbeispiele [33](#)
 - nicht mit Data Services unterstützt [31](#)
 - System [31](#), [32](#)
 - Übersicht [31](#)
- Mapping-Pipelines
 - Beschreibung [94](#)
- mappings
 - Verringern des Parallelismus [112](#)
- Mappings
 - Objekte [13](#)
 - Ausdrucksvalidierung [24](#)
 - Deaktivieren der Partitionierung [111](#)
 - Entwickeln [12](#)
 - erstellen [12](#)
 - Erweiterte Eigenschaften [22](#)
 - Hinzufügen von Objekten [14](#)
 - Laufzeiteigenschaften [22](#)
 - Maximaler Parallelismus [95](#)
 - Objektabhängigkeit [13](#)
 - Objektvalidierung [24](#)
 - Optimierungsmethoden [66](#)
 - Partitioniert [95](#)
 - Partitionierungseinschränkungen [110](#)
 - Partitionspunkte [94](#)
 - pipelines [94](#)
 - Übersicht [11](#)
 - Umgebung [22](#)
 - validieren [24](#)
 - Validierung [23](#)
 - Validierungsumgebung [22](#)
 - Verarbeitungs-Threads [94](#)
 - Verbindungsvalidierung [24](#)
 - Vorhersageoptimierungsmethode [68](#)
 - wird ausgeführt [25](#)
- Mapplets
 - Ausgabe [29](#)
 - Eingabe [29](#)
 - erstellen [29](#)
 - Exportieren an PowerCenter [47](#)
 - Regeln [28](#)
 - Typen [27](#)
 - Übersicht [26](#)
 - Wird validiert... [30](#)
- Match-Umwandlung
 - Partitionierungseinschränkungen [110](#)
- Maximaler Parallelismus
 - Beschreibung [95](#)
- Microsoft SQL Server-Quellen
 - Pushdown-Optimierung [75](#)
 - Pushdown-Optimierung [75](#)
- Minimale Optimierungsebene
 - Beschreibung [66](#)
- MOVINGAVG, Funktion
 - Partitionierungseinschränkungen [109](#)

- MOVINGSUM, Funktion
 - Partitionierungseinschränkungen [109](#)

N

- Nichtrelationale Quellen
 - Pushdown-Optimierung [75](#)
- Normale Optimierungsebene
 - Beschreibung [66](#)

O

- Operatoren
 - Pushdown-Optimierung [90](#)
 - Verfügbar in Quellen [90](#)
- Optimierung
 - frühe Auswahloptimierungsmethode [67](#)
 - frühe Projektionsoptimierungsmethode [67](#)
 - kostenbasierte Optimierungsmethode [68](#)
 - Methode zur Optimierung der Verzweigungsreinigung [67](#)
 - Methoden zur Optimierung der Mapping-Leistung [66](#)
 - Semi-Join Optimierungsmethode [69](#)
- Optimierung der Verzweigungsreinigung
 - Beschreibung [67](#)
- Optimierungsebenen
 - Beschreibung [66](#)
- Oracle-Quellen
 - Pushdown-Optimierung [75](#)
 - Partitioniert [100](#)

P

- Parallelismus
 - Berechnen des Werts [96](#)
 - Verringern für Mapping [112](#)
- Parameterdateien
 - Anwendungselement [39](#)
 - Arbeitsablauf ausführen mit [36](#)
 - Beispiel [41](#)
 - erstellen [44](#)
 - Mapping [31](#)
 - Mappings ausführen mit [36](#)
 - Projektelement [38](#)
 - Struktur [37](#)
 - XML-Schemadefinition [37](#)
 - Zweck [36](#)
- parameters
 - Mapping [31](#)
- Partitionierung
 - Adress-Validierungsumwandlung [110](#)
 - Aggregatormwandlung [108](#)
 - cache [108](#)
 - Cachegröße [109](#)
 - Deaktivieren für Mapping [111](#)
 - Einfachdatei-Quellen [99](#)
 - Einfachdatei-Ziele [103](#)
 - Gleichzeitiges Lesen [99](#)
 - IBM DB2 für LUW-Quellen [100](#)
 - IBM DB2 für LUW-Ziele [105](#)
 - Java-Umwandlung [108](#)
 - Joiner-Umwandlung [108](#)
 - Mappings [95](#)
 - Match-Umwandlung [110](#)
 - Maximaler Parallelismus [95](#)
 - Oracle-Quellen [100](#)

Partitionierung (Fortsetzung)

- Parallelismuswert [96](#)
- Rangumwandlung [108](#)
- Relationale Verbindungstypen [101](#), [106](#)
- Sortierumwandlung [107](#)
- Umwandlungen [107](#)
- Verringern für Mapping [112](#)
- Zusammengeführte Dateiziele [104](#)
- Partitionierung für gleichzeitiges Lesen
 - Beschreibung [99](#)
- Partitionierungseinschränkungen
 - Adress-Validierumwandlung [110](#)
 - Aggregatorumwandlung [108](#)
 - Begrenzen von Quellzeilen [110](#)
 - Java-Umwandlung [108](#)
 - Joiner-Umwandlung [108](#)
 - Lokale Variablen [110](#)
 - Match-Umwandlung [110](#)
 - Numerische Funktionen [109](#)
 - Rangumwandlung [108](#)
 - Relationale Quellen [101](#), [102](#)
 - Relationale Ziele [106](#), [107](#)
 - Sortierumwandlung [107](#)
- Partitionspunkte
 - Beschreibung [94](#)
- Pipeline-Stages
 - Beschreibung [94](#)
- Portattribute
 - Verteilen [18](#)
- Ports
 - Automatisch verlinken [16](#)
 - Manuelle Zusammenführung [16](#)
 - Verbindungsvalidierung [24](#)
 - verknüpfen [14](#)
 - Verknüpfen nach Name [16](#)
 - Verknüpfen nach Position [17](#)
 - Verknüpfen von Regeln und Richtlinien [18](#)
 - Verteilte Attribute nach Umwandlung [20](#)
- Projektelement
 - Parameterdateien [38](#)
- Pushdown-Optimierung
 - Ausdrücke [77](#)
 - Relationale Quellen [73](#), [75](#)
 - SAP-Quellen [76](#)
 - Funktionen [77](#)
 - IBM DB2-Quellen [75](#)
 - Microsoft SQL Server-Quellen [75](#)
 - Nichtrelationale Quellen unter z/OS [75](#)
 - ODBC-Quellen [75](#)
 - Operatoren [90](#)
 - Oracle-Quellen [75](#)
 - Quellen [72](#)
 - Relationale Quellen [73](#), [75](#)
 - Sybase ASE-Quellen [75](#)
 - Übersicht [71](#)

Q

Quellen

- Partitioniert, relational [100](#)
- Partitionierte Einfachdatei [99](#)

R

- Rangumwandlung
 - Cache-Partitionierung [108](#)
 - Partitioniert [108](#)
- Relationale Datenobjekte
 - Partitioniert [100](#), [105](#)
- Relationale Quellen
 - Partitioniert [100](#)
 - Partitionierungseinschränkungen [101](#), [102](#)
 - Pushdown-Optimierung [75](#)
- Relationale Ziele
 - Partitioniert [105](#)
 - Partitionierungseinschränkungen [106](#), [107](#)

S

- SAP-Quellen
 - Pushdown-Optimierung [76](#)
- Segmente
 - kopieren [25](#)
- Semi-Join-Optimierung
 - Beschreibung [69](#)
- Sortierumwandlung
 - Partitioniert [107](#)
- Sybase ASE-Quellen
 - Pushdown-Optimierung [75](#)
- Systemparameter
 - Mapping [31](#)

T

- threads
 - Verarbeiten von Mappings [94](#)

U

- Umwandlungen
 - Partitioniert [107](#)

V

- Verarbeitungs-Threads
 - Mappings [94](#)
- Vollständige Optimierungsebene
 - Beschreibung [66](#)

Z

- Ziele
 - Partitioniert, relational [105](#)
 - Partitionierte Einfachdatei [103](#)
 - Zusammenführungsdatei [104](#)
- Zuordnungsparameter
 - Benutzerdefiniert [32](#)
 - Typen [32](#)