



Informatica®
10.1.1

Developer 映射指南

Informatica Developer 映射指南

10.1.1

2016 年 12 月

© 版权所有 Informatica LLC 2014, 2018

本软件和文档仅根据包含使用与披露限制的单独许可协议提供。未事先征得 Informatica LLC 同意，不得以任何形式、通过任何手段（电子、影印、录制或其他手段）复制或传播本文档的任何部分。

Informatica、Informatica 标志、PowerCenter 和 PowerExchange 是 Informatica LLC 在美国和世界其他许多司法管辖区的商标或注册商标。欲获得 Informatica 商标的最新列表，请访问 <https://www.informatica.com/trademarks.html>。其他公司和产品名称可能是其各自所有者的商业名称或商标。

本软件和/或文档的某些部分受第三方版权制约，包括但不限于：版权所有 DataDirect Technologies。保留所有权利。版权所有 (C) Sun Microsystems。保留所有权利。版权所有 (C) RSA Security Inc. 保留所有权利。版权所有 (C) Ordinal Technology Corp. 保留所有权利。版权所有 (C) Aandacht c.v. 保留所有权利。版权所有 Genivia, Inc. 保留所有权利。版权所有 Isomorphic Software。保留所有权利。版权所有 (C) Meta Integration Technology, Inc. 保留所有权利。版权所有 (C) Intalio。保留所有权利。版权所有 (C) Oracle。保留所有权利。版权所有 (C) Adobe Systems Incorporated。保留所有权利。版权所有 (C) DataArt, Inc. 保留所有权利。版权所有 (C) ComponentSource。保留所有权利。版权所有 (C) Microsoft Corporation。保留所有权利。版权所有 (C) Rogue Wave Software, Inc. 保留所有权利。版权所有 (C) Teradata Corporation。保留所有权利。版权所有 (C) Yahoo! Inc. 保留所有权利。版权所有 (C) Glyph & Cog, LLC。保留所有权利。版权所有 (C) Thinkmap, Inc. 保留所有权利。版权所有 (C) Clearpace Software Limited。保留所有权利。版权所有 (C) Information Builders, Inc. 保留所有权利。版权所有 (C) OSS Nokalva, Inc. 保留所有权利。版权所有 Edifecs, Inc. 保留所有权利。版权所有 Cleo Communications, Inc. 保留所有权利。版权所有 (C) International Organization for Standardization 1986。保留所有权利。版权所有 (C) ej-technologies GmbH。保留所有权利。版权所有 (C) Jaspersoft Corporation。保留所有权利。版权所有 (C) International Business Machines Corporation。保留所有权利。版权所有 (C) yWorks GmbH。保留所有权利。版权所有 (C) Lucent Technologies。保留所有权利。版权所有 (C) University of Toronto。保留所有权利。版权所有 (C) Daniel Veillard。保留所有权利。版权所有 (C) Unicode, Inc. 保留所有权利。版权所有 IBM Corp. 保留所有权利。版权所有 (C) MicroQuill Software Publishing, Inc. 保留所有权利。版权所有 (C) PassMark Software Pty Ltd. 保留所有权利。版权所有 (C) LogiXML, Inc. 保留所有权利。版权所有 (C) 2003-2010 Lorenzi Davide。保留所有权利。版权所有 (C) Red Hat, Inc. 保留所有权利。版权所有 (C) The Board of Trustees of the Leland Stanford Junior University。保留所有权利。版权所有 (C) EMC Corporation。保留所有权利。版权所有 (C) Flexera Software。保留所有权利。版权所有 (C) Jinfonet Software。保留所有权利。版权所有 (C) Apple Inc. 保留所有权利。版权所有 (C) Telerik Inc. 保留所有权利。版权所有 (C) BEA Systems。保留所有权利。版权所有 (C) PDFlib GmbH。保留所有权利。版权所有 (C) Orientation in Objects GmbH。保留所有权利。版权所有 (C) Tanuki Software, Ltd. 保留所有权利。版权所有 (C) Ricebridge。保留所有权利。版权所有 (C) Sencha, Inc. 保留所有权利。版权所有 (C) Scalable Systems, Inc. 保留所有权利。版权所有 (C) jQWidgets。保留所有权利。版权所有 (C) Tableau Software, Inc. 保留所有权利。版权所有 (C) MaxMind, Inc. 保留所有权利。版权所有 (C) TMate Software s.r.o. 保留所有权利。版权所有 (C) MapR Technologies Inc. 保留所有权利。版权所有 (C) Amazon Corporate LLC。保留所有权利。版权所有 (C) Highsoft。保留所有权利。版权所有 (C) Python Software Foundation。保留所有权利。版权所有 (C) BeOpen.com。保留所有权利。版权所有 (C) CNRI。保留所有权利。

本产品包括由 Apache Software Foundation (<http://www.apache.org/>) 开发的软件 and/或在不同 Apache 许可证版本（以下简称“许可证”）下许可的其他软件。您可从 <http://www.apache.org/licenses/> 获取这些许可证的副本。除非适用法律要求或者有相应书面协议，否则依据这些“许可证”分发的软件以“原样”提供，不附带任何明示或暗示的担保或条件。请参阅“许可证”中规定的具体语言管理权限和限制。

本产品包括由 Mozilla (<http://www.mozilla.org/>) 开发的软件、由 JBoss Group, LLC 开发的软件（版权所有 JBoss Group, LLC 保留所有权利）、由 Bruno Lowagie 和 Paulo Soares 开发的软件（版权所有 (C) 1999-2006 Bruno Lowagie 和 Paulo Soares）以及在 <http://www.gnu.org/licenses/lgpl.html> 网站上的不同版本 GNU Lesser General 公共许可协议下许可的软件。这些材料由 Informatica 按“原样”免费提供，不附带任何明示或暗示的担保，包括但不限于适销性和特定用途适用性的暗示担保。

本产品包括 ACE(TM) 和 TAO(TM) 软件，这些软件版权归 Douglas C. Schmidt 及其在华盛顿大学、加利福尼亚大学欧文分校以及范德堡大学的研发团队所有（版权所有 (C) 1993-2006，保留所有权利）。

本产品包括由 OpenSSL Project 开发并在 OpenSSL Toolkit（版权所有 OpenSSL Project。保留所有权利）中使用的软件，该软件的再分发受 <http://www.openssl.org> 和 <http://www.openssl.org/source/license.html> 上规定条款之制约。

本产品包括 Curl 软件，版权所有 1996-2013, Daniel Stenberg <daniel@haxx.se>。保留所有权利。有关该软件的权限和限制受 <http://curl.haxx.se/docs/copyright.html> 上规定条款之制约。允许出于任何目的以免费或收费形式使用、复制、修改和分发该软件，但前提是所有副本均应注明上述版权声明以及本许可声明。

本产品包括由 MetaStuff, Ltd. 开发的软件，版权所有 2001-2005 ((C)) MetaStuff, Ltd. 保留所有权利。有关该软件的权限和限制受 <http://www.dom4j.org/license.html> 上规定条款之制约。

本产品包括由 Dojo Foundation 开发的软件，版权所有 (C) 2004-2007, Dojo Foundation。保留所有权利。有关该软件的权限和限制受 <http://dojotoolkit.org/license> 上规定条款之制约。

本产品包括 ICU 软件，版权所有 International Business Machines Corporation 和其他方。保留所有权利。有关该软件的权限和限制受 <http://source.icu-project.org/repos/icu/icu/trunk/license.html> 上规定条款之制约。

本产品包括由 Per Bothner 开发的软件，版权所有 (C) 1996-2006 Per Bothner。保留所有权利。<http://www.gnu.org/software/kawa/Software-License.html> 上的许可证中规定了您使用这些材料的权利。

本产品包括 OSSP UUID 软件，版权所有 (C) 2002 Ralf S. Engelschall，版权所有 (C) 2002 OSSP Project，版权所有 (C) 2002 Cable & Wireless Deutschland。有关该软件的权限和限制受 <http://www.opensource.org/licenses/mit-license.php> 上规定条款之制约。

本产品包括由 Boost (<http://www.boost.org/>) 开发的软件或在 Boost 软件许可证下许可的软件。有关该软件的权限和限制受 http://www.boost.org/LICENSE_1_0.txt 上规定条款之制约。

本产品包括由 University of Cambridge 开发的软件，版权所有 (C) 1997-2007 University of Cambridge。有关该软件的权限和限制受 <http://www.pcre.org/license.txt> 上规定条款之制约。

本产品包括由 The Eclipse Foundation 开发的软件，版权所有 (C) 2007 The Eclipse Foundation。保留所有权利。有关该软件的权限和限制受 <http://www.eclipse.org/org/documents/epl-v10.php> 和 <http://www.eclipse.org/org/documents/edl-v10.php> 上规定条款之制约。

本产品包括在 <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://htpunittest.sourceforge.net/doc/license.html>、<http://jung.sourceforge.net/license.txt>、http://www.zip.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/licenses-agreements/fuse-message-broker-v-5-3-license-agreement>、<http://antlr.org/license.html>、<http://aopalliance.sourceforge.net/>、<http://www.bouncycastle.org/licence.html>、<http://www.jgraph.com/jgraphdownload.html>、<http://www.jcraft.com/jsch/LICENSE.txt>、http://jotm.objectweb.org/bsd_license.html、<http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>、<http://www.slf4j.org/license.html>、<http://nanoxml.sourceforge.net/orig/copyright.html>、<http://www.json.org/en/jcalendar/license.html>、<http://forge.ow2.org/projects/jaservice/>、<http://www.postgresql.org/about/licence.html>、<http://www.sqlite.org/copyright.html>、<http://www.tcl.tk/software/tcltk/license.html>、<http://www.jaxen.org/faq.html>、<http://www.jdom.org/docs/faq.html>、<http://www.slf4j.org/license.html>、<http://www.iodbc.org/dataspace/iodbc/wiki/ODBC/License>、<http://www.keplerproject.org/md5/license.html>、<http://www.toedter.com/en/jcalendar/license.html>、<http://www.edankert.com/bounce/index.html>、<http://www.net-snmp.org/about/license.html>、<http://www.openmdx.org/#FAQ>、http://www.php.net/license/3_01.txt、<http://srp.stanford.edu/license.txt>、<http://www.schneier.com/blowfish.html>、<http://www.jmock.org/license.html>、<http://xsom.java.net>、<http://benalman.com/about/license/>、<http://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>、<http://www.h2database.com/html/license.html#summary>、<http://jsoncpp.sourceforge.net/LICENSE>、<http://jdbc.postgresql.org/license.html>、<http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto>、<https://github.com/>

rantav/hector/blob/master/LICENSE; <http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html>、<http://jibx.sourceforge.net/jibx-license.html>、<https://github.com/lyokato/libgeohash/blob/master/LICENSE>、<https://github.com/hjiang/jsonxx/blob/master/LICENSE>、<https://code.google.com/p/lz4/>、<https://github.com/jedisct1/libsodium/blob/master/LICENSE>、<http://one-jar.sourceforge.net/index.php?page=documents&file=license>、<https://github.com/EsotericSoftware/kryo/blob/master/license.txt>、<http://www.scala-lang.org/license.html>、<https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt>、<http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html>、<https://aws.amazon.com/asl/>、<https://github.com/twbs/bootstrap/blob/master/LICENSE> 和 <https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt> 下许可的软件。

本产品包括在 Academic 免费许可证 (<http://www.opensource.org/licenses/afl-3.0.php>)、通用开发和分发许可证 (<http://www.opensource.org/licenses/cddl1.php>)、通用公共许可证 (<http://www.opensource.org/licenses/cpl1.0.php>)、Sun Binary Code 许可协议补充许可条款、BSD 许可证 (<http://www.opensource.org/licenses/bsd-license.php>)、新 BSD 许可证 (<http://opensource.org/licenses/BSD-3-Clause>)、MIT 许可证 (<http://www.opensource.org/licenses/mit-license.php>)、Artistic 许可证 (<http://www.opensource.org/licenses/artistic-license-1.0>) 以及原始开发者公共许可证版本 1.0 (<http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/>) 下许可的软件。

本产品包括由 Joe Walnes 和 XStream Committers 开发的软件，版权所有 (C) 2003-2006 Joe Walnes，2006-2007 XStream Committers。保留所有权利。有关该软件的权限和限制受 <http://xstream.codehaus.org/license.html> 上规定条款之制约。本产品包括由 Indiana University Extreme! Lab 开发的软件。有关详细信息，请访问 <http://www.extreme.indiana.edu/>。

本产品包括软件版权所有 (c) 2013 Frank Balluffi 和 Markus Moeller。保留所有权利。有关此软件的权限和限制受 MIT 许可证上规定条款之制约。

请参阅位于以下位置的专利：<https://www.informatica.com/legal/patents.html>。

免责声明：Informatica LLC 以“原样”提供本文档，不附带任何明示或暗示的担保，包括但不限于非侵权、适销性或特定用途适用性的暗示担保。Informatica LLC 不保证本软件和文档中没有错误。本软件或文档中提供的信息可能包括技术上的不准确性或排字错误。本软件和文档中包含的信息随时可能更改，恕不另行通知。

声明

本 Informatica 产品（以下称“软件”）包括由 Progress Software Corporation 的运营公司 DataDirect Technologies（以下称“DataDirect”）提供的某些驱动程序（以下称“DataDirect 驱动程序”），受以下条款和条件制约：

1. DataDirect 驱动程序以“原样”提供，不附带任何明示或暗示的担保，包括但不限于适销性、特定用途适用性以及非侵权的暗示担保。
2. 在任何情况下，DataDirect 或其第三方供应商均不对最终用户客户承担因使用 ODBC 驱动程序而引起的任何直接、间接、偶发、特殊、继发或其他损害赔偿的责任，无论是否已提前告知该种损害的可能性。这些限制适用于所有诉因，包括但不限于违反合同、违反担保、过失、严格责任、虚假陈述以及其他侵权行为。

本文档中的信息如有更改，恕不另行通知。如果您发现本文档中存在任何问题，请以书面形式将问题报告给我们，邮寄地址是 Informatica LLC 2100 Seaport Blvd.Redwood City, CA 94063。

INFORMATICA LLC 按“原样”提供本文档中的信息，无任何明示或暗示的担保，包括但不限于任何适销性和特定用途适用性担保，也没有任何非侵权担保或条件。

发布日期: 2018-06-25

目录

前言	12
Informatica 资源	12
Informatica Network	12
Informatica 知识库	12
Informatica 文档	12
Informatica 产品可用性矩阵	13
Informatica Velocity	13
Informatica Marketplace	13
Informatica 全球客户支持部门	13
第 1 章：映射简介	14
映射概览	14
映射开发	15
映射组件	15
转换	16
Mapplet	16
段	16
复制段	16
视图	17
映射验证	17
连接验证	18
表达式验证	18
对象验证	18
映射运行时属性	18
验证环境	21
执行环境	21
最大并行数	22
目标提交间隔	22
出错时停止	22
映射模拟用户名	22
建议的并行值	23
Hive 连接	23
目标加载顺序约束	23
与插入行和删除行相关的约束	23
目标加载顺序的规则和准则	24
目标加载顺序示例	24
如何开发映射	26
创建映射	26
将对象添加到映射	26
连接映射对象	27

创建目标加载顺序约束.	27
验证映射.	28
运行映射.	29
第 2 章：Mapplet.	30
Mapplet 概览.	30
Mapplet 类型.	31
Mapplet 输入和输出.	32
Mapplet 输入.	32
Mapplet 输出.	32
生成的 Mapplet.	32
生成的 Mapplet 规则和准则.	33
生成 Mapplet.	33
创建 Mapplet.	34
Mapplet 验证.	34
验证 Mapplet.	34
将 Mapplet 验证为规则.	35
第 3 章：映射参数.	36
映射参数概览.	36
系统参数.	37
用户定义参数.	38
日期/时间参数.	39
创建用户定义参数的位置.	39
分配参数的位置.	39
表达式中的参数.	42
SQL 语句中的参数.	44
Hive 源的自定义查询中的参数.	45
关系表资源的参数.	46
字段和属性值的参数.	47
端口列表的参数.	47
映射中的参数.	48
参数实例值.	48
Mapplet 中的参数.	49
Mapplet 中的参数实例值.	49
映射中的 Mapplet 参数.	50
Mapplet 示例中的参数.	50
逻辑数据对象中的参数.	51
虚拟表映射中的参数.	51
参数集.	52
从 Infacmd 中使用参数集运行映射.	52
参数文件.	53
参数文件结构.	53

项目元素.	54
应用程序元素.	55
参数文件的规则和准则.	55
参数文件示例.	55
导出参数文件.	58
从 infacmd ms ListMappingParams 创建参数文件.	59
使用参数文件运行映射.	59
如何配置参数.	60
为转换属性创建参数.	60
在表达式中创建参数.	62
将转换参数公开为映射参数.	65
设置参数实例值.	66
创建参数集.	67
第 4 章：映射输出.	69
映射输出概览.	69
用户定义的映射输出.	70
输出视图.	70
映射输出表达式.	71
系统定义的映射输出.	72
保留映射输出.	73
保留值维护.	74
保留映射输出和部署.	75
将映射输出绑定到工作流变量.	75
Mapplet 中的映射输出.	76
将 Mapplet 输出绑定到映射输出.	77
逻辑数据对象中的映射输出.	79
如何配置映射输出.	79
创建映射.	80
定义映射输出.	82
配置映射输出表达式.	83
保留映射输出.	85
将保留输出分配给映射任务输入.	86
将映射输出绑定到工作流变量.	87
如何将 Mapplet 输出绑定到映射输出.	88
定义 Mapplet 输出.	89
在 Mapplet 中配置映射输出表达式.	90
将 Mapplet 输出绑定到映射输出.	91
第 5 章：从 SQL 查询生成映射.	93
从 SQL 查询生成映射概览.	93
从 SQL 查询生成映射的示例.	93
用于生成映射的 SQL 语法.	94

相关子查询.	94
生成映射的查询中的函数支持.	94
从包含不受支持的函数的 SQL 查询生成映射.	95
INSERT、UPDATE 和 DELETE 语法.	96
INSERT、UPDATE 和 DELETE 语句的规则和准则.	96
从 SQL 查询生成映射或逻辑数据对象.	96
从 SQL 语句生成映射.	97
创建 SQL 语句.	97
将 SQL 语句粘贴或导入到 Developer tool.	97
完成映射开发.	99
第 6 章：动态映射.	100
动态映射概览.	100
动态映射配置.	101
动态数据源.	101
动态映射端口和链接.	102
动态映射规则.	102
动态映射中的参数.	103
动态源.	103
从数据源获取列.	104
为平面文件名分配参数.	105
为关系源属性分配参数.	105
为源数据对象分配参数.	106
动态目标.	106
从数据源获取列.	107
根据映射流定义目标.	108
根据数据对象定义目标.	108
在运行时创建或替换目标.	108
为关系目标属性分配参数.	109
为目标数据对象分配参数.	110
动态目标的规则和准则.	110
动态端口和生成的端口.	110
动态端口和生成的端口的配置.	111
动态端口和生成的端口的规则和准则.	112
动态表达式.	112
输入规则.	113
输入规则配置.	114
包含或排除端口.	114
包含所有剩余端口.	115
重命名生成的端口.	116
重新排序生成的端口.	119
选择规则和端口选择器.	121
端口选择器配置.	122

选择规则.	122
示例 - 选择规则和端口选择器.	123
设计时链接.	123
链接解析.	124
运行时链接.	125
运行时链接配置.	126
示例 - 运行时链接.	127
动态映射故障排除.	128
 第 7 章： 如何开发和运行动态映射.	130
开发和运行动态映射.	130
配置动态源.	131
使用参数作为动态映射源.	131
配置源以在运行时获取元数据更改.	132
创建动态端口.	132
使用输入规则配置动态端口.	134
步骤 1. 打开“输入规则”对话框.	134
步骤 2. 定义输入规则.	134
步骤 2a. 选择运算符和选择条件.	135
步骤 2b. 配置名称选择条件详细信息.	135
步骤 2c. 配置类型选择条件详细信息.	136
步骤 2d. 配置模式选择条件详细信息.	136
步骤 3. 重命名生成的端口.	137
步骤 4. 重新排序生成的端口.	137
步骤 5. 验证动态端口配置.	137
创建端口选择器.	137
创建动态表达式.	138
配置动态目标.	140
使用参数作为动态映射目标.	141
在运行时从数据源获取目标对象列.	142
定义 DDL 查询以在运行时创建或替换目标.	142
定义写入转换端口.	143
创建和配置运行时链接.	144
创建运行时链接.	147
验证和运行动态映射.	147
 第 8 章： 动态映射使用案例.	148
使用案例：用于处理关系源中元数据更改的动态映射.	148
源表.	148
目标表.	149
动态映射.	149
步骤 1. 配置读取转换.	150
步骤 2. 配置连接器转换.	151

步骤 3. 配置汇总器转换.	152
步骤 4. 配置写入转换.	155
步骤 5. 创建并配置运行时链接.	155
步骤 6. 验证并运行映射.	156
步骤 7. 在源架构发生更改后运行映射.	157
使用案例：将动态映射重用于不同的源和目标.	159
源文件.	159
目标文件.	161
动态映射.	161
步骤 1. 配置 Read_Customer_FF 读取转换.	161
步骤 2. 配置 Exp_TRIM 表达式转换.	162
步骤 3. 配置 Exp_Output 表达式转换.	166
步骤 4. 配置 Write_customerTrim_FF 写入转换.	168
步骤 5. 验证并保存映射.	169
步骤 6. 针对不同的源和目标运行动态映射.	170
 第 9 章：映射管理.	 173
映射管理概览.	173
查看映射作业的属性.	173
查看映射作业的摘要统计信息.	174
查看映射作业的详细统计信息.	174
查看映射作业的日志.	175
重新发出已部署映射作业.	175
取消映射作业.	175
拒绝文件.	176
拒绝文件的位置.	176
拒绝文件的内容.	176
 第 10 章：导出到 PowerCenter.	 178
导出到 PowerCenter：概览.	178
PowerCenter 版本兼容性.	179
设置兼容性级别.	179
Maplet 导出.	179
导出包含参数的映射.	180
导出到 PowerCenter：选项.	180
将对象导出到 PowerCenter.	181
导出限制.	182
导出到 PowerCenter：规则和准则.	183
导出到 PowerCenter：故障排除.	184
 第 11 章：从 PowerCenter 导入.	 185
从 PowerCenter 导入：概览.	185
替代属性.	186

冲突解决方案.	186
导入摘要.	186
数据类型转换.	187
对转换进行转换.	187
转换属性限制.	188
导入包含参数的映射.	193
导入包含系统定义的参数的映射.	194
PowerCenter 存储库连接参数.	194
连接分配.	195
从 PowerCenter 导入对象.	196
导入限制.	197
导入性能.	197
 第 12 章：性能优化.	 199
性能优化概览.	199
优化方法.	200
早期计划优化方法.	200
早期选择优化方法.	200
分支修剪优化方法.	201
谓词优化方法.	201
基于成本的优化方法.	201
船舶数据中心联接优化方法.	202
半联接优化方法.	202
查看优化映射.	203
优化器级别.	203
为 Developer tool 映射设置优化器级别.	204
为已部署的映射设置优化器级别.	204
 第 13 章：下推优化.	 205
下推优化概览.	205
下推类型.	205
完整下推优化.	206
源下推.	207
配置下推.	207
转换逻辑下推.	207
到源的下推优化.	208
到关系源的下推优化.	208
到本地源的下推优化.	210
到 PowerExchange 非关系源的下推优化.	210
到 ODBC 源的下推优化.	211
到 SAP 源的下推优化.	211
下推优化表达式.	212
函数.	212

运算符.....	224
比较数据集成服务的输出和源.....	226
第 14 章：已分区的映射.....	227
已分区的映射概览.....	227
每个管道阶段一个线程.....	228
每个管道阶段多个线程.....	229
已分区平面文件源.....	230
并发读取分区.....	230
已分区关系源.....	231
分区的关系连接类型.....	232
已分区关系源的 SQL 查询.....	232
关系源分区的规则和准则.....	232
已分区平面文件目标.....	233
优化分区文件目标的输出文件目录.....	233
已分区文件目标的合并选项.....	233
已分区文件目标的命令.....	235
已分区关系目标.....	236
分区的关系连接类型.....	236
关系目标分区的规则和准则.....	236
已分区转换.....	237
已分区转换的约束.....	237
转换的缓存分区.....	238
为转换禁用分区.....	239
保持已分区映射中的顺序.....	239
保持稳定排序.....	240
替代映射的最大并行数值.....	240
转换的建议并行值.....	241
地址验证器和匹配转换的执行实例.....	242
替代最大并行数值.....	242
对已分区映射进行故障排除.....	243
第 15 章：Developer tool 命名约定.....	244
转换命名约定.....	244
对象类型命名约定.....	246
工作流对象命名约定.....	246
索引.....	248

前言

《Informatica Developer 映射指南》包含有关如何创建和使用映射与 Maplet 的信息，以及如何配置动态映射、如何导出与导入对象和如何启用性能调整与下推优化的信息。《Informatica Developer 映射指南》面向负责开发映射和 Maplet 的开发人员，例如大数据、数据服务和数据转换开发人员。本指南假设您已了解平面文件和关系数据库的概念以及环境中的数据库引擎。

Informatica 《PowerCenter Express 映射指南》包含如何创建和使用映射与 Maplet 的相关信息。Informatica 《PowerCenter Express 映射指南》包含如何启用性能优化与下推优化的相关信息。Informatica 《PowerCenter Express 映射指南》面向数据集成开发人员。本指南假定您已了解平面文件和相关数据库概念、您环境中的数据库引擎以及数据集成概念。

Informatica 资源

Informatica Network

Informatica Network 囊括了 Informatica 全球客户支持部门、Informatica 知识库和其他产品资源。要访问 Informatica Network，请访问 <https://network.informatica.com>。

成员可以执行以下操作：

- 在一个位置访问您的所有 Informatica 资源。
- 在知识库中搜索文档、常见问题和最佳实践等产品资源。
- 查看产品可用性信息。
- 查看支持案例。
- 查找当地的 Informatica 用户组网络并与您的伙伴进行协作。

Informatica 知识库

使用 Informatica 知识库可在 Informatica Network 中搜索文档、入门知识文章、最佳实践和 PAM 等产品资源。

要访问知识库，请访问 <https://kb.informatica.com>。如果您对知识库有任何疑问、意见或建议，请与 Informatica 知识库团队联系，电子邮件地址为 KB_Feedback@informatica.com。

Informatica 文档

要获取有关产品的最新文档，请浏览 Informatica 知识库，网址为 https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx。

如果您对此文档有任何疑问、意见或建议，请与 Informatica 文档团队联系，电子邮件地址为 infa_documentation@informatica.com。

Informatica 产品可用性矩阵

产品可用性矩阵 (PAM) 指明了产品版本支持的操作系统版本、数据库以及其他类型的数据源和目标。如果您是 Informatica Network 成员，您可以访问 PAM，网址为 <https://network.informatica.com/community/informatica-network/product-availability-matrices>。

Informatica Velocity

Informatica Velocity 收集了 Informatica 专业服务开发的一系列提示和最佳实践。Informatica Velocity 基于数以百计的数据管理项目的实际经验而开发，汇集了我们曾在世界各地组织就职的顾问在成功规划、开发、部署和维护数据管理解决方案方面的知识。

如果您是 Informatica Network 成员，您可以访问 Informatica Velocity 资源，网址为 <http://velocity.informatica.com>。

如果您对 Informatica Velocity 有任何疑问、意见或建议，请通过 ips@informatica.com 与 Informatica 专业服务联系。

Informatica Marketplace

Informatica Marketplace 是一个论坛，该论坛中提供的解决方案可补充、扩展或增强您的 Informatica 实现。您可以利用 Informatica 开发人员和合作伙伴提供的数以百计解决方案中的任何方案，提高生产率，加快项目的实现时间。您可以访问 Informatica Marketplace，网址为 <https://marketplace.informatica.com>。

Informatica 全球客户支持部门

您可以通过电话或 Informatica Network 上的联机支持与全球支持中心联系。

要查找您当地的 Informatica 全球客户支持部门电话号码，请访问 Informatica 网站，链接为：
<http://www.informatica.com/us/services-and-training/support-services/global-support-centers>。

如果您是 Informatica Network 成员，您可以使用联机支持，网址为 <http://network.informatica.com>。

第 1 章

映射简介

本章包括以下主题：

- [映射概览, 14](#)
- [映射组件, 15](#)
- [视图, 17](#)
- [映射验证, 17](#)
- [映射运行时属性, 18](#)
- [目标加载顺序约束, 23](#)
- [如何开发映射, 26](#)

映射概览

映射是一组代表源和目标之间的数据流的输入和输出对象。它们通过定义数据转换规则的转换对象进行链接。数据集成服务使用映射中配置的说明来读取、转换和写入数据。

您可以部署映射并从命令行运行映射。您还可以从 Developer tool 直接运行映射，而无需工作流。您可以在开发期间预览映射。

映射中包含的输入和输出对象类型决定了映射类型。在 Developer tool 中可以创建以下映射类型：

通用映射

具有作为输入和输出的物理数据对象。当您需要从源中提取字段并将字段加载到目标文件时，请使用通用映射。

逻辑数据对象映射

将一个逻辑数据对象链接至一个或多个物理数据对象。逻辑数据对象映射可以帮助您集成来自多个源的数据，并调整其格式，形成一个标准化视图。

操作映射

具有作为映射输入、输出或两者的操作。操作映射可为 Web 服务客户端执行 Web 服务操作。

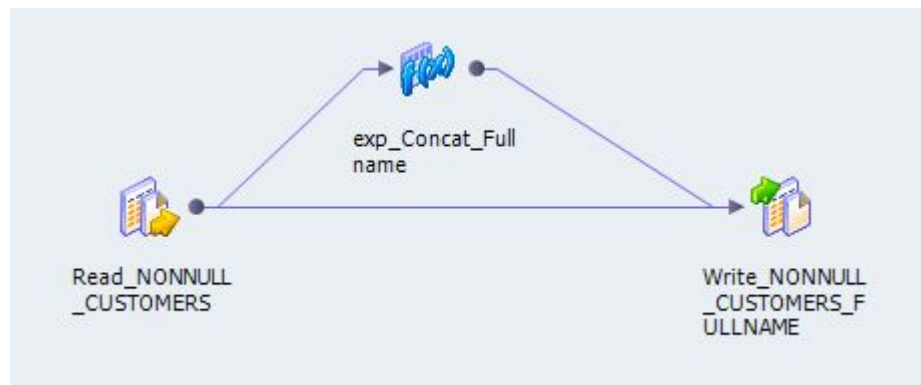
虚拟表映射

具有作为映射输出的虚拟表。虚拟表映射定义了源与 SQL 数据服务中的虚拟表之间的虚拟数据流。使用虚拟表映射可转换数据。

动态映射

可以根据您定义的参数和规则在运行时更改源、目标以及转换逻辑的映射。使用动态映射可以管理频繁发生变化的架构或元数据。

下图显示了一个映射示例：



映射开发

对于将物理数据对象用作输入和输出的映射，可以将其包含在工作流的映射任务中。

您可能需要从工作流运行映射，以便按顺序运行多个映射。也可以开发一个工作流，使其在映射运行前后运行命令以执行步骤。

还可以通过以下方式开发和运行映射：

开发和运行动态映射

当平面文件或关系源的架构或元数据会频繁更改时，需要开发一个动态映射以在运行时管理架构或元数据更改。也可以通过开发动态映射来重用映射逻辑并对不同的物理源和目标运行映射。动态映射是指可在运行时基于您定义的参数和规则更改源、目标和转换逻辑的映射。

从工作流运行映射

您可能需要从工作流运行映射，以便按顺序运行多个映射。也可以开发一个工作流，使其在映射运行前后运行命令以执行步骤。对于将物理数据对象用作输入和输出的映射，可以将其包含在工作流的映射任务中。

映射组件

映射组件决定源和目标之间的数据流。

每个映射必须包含从映射组件或文件读取数据的输入对象。每个映射还必须包含将数据写入映射组件或文件的输出对象。

映射还可以包含以下组件：

转换

在将数据写入到目标之前，对其进行修改。使用不同的转换对象可执行不同的功能。

Mapplet

是可重用对象，其中包含一组可以在多个映射中使用的转换。

段

由映射、Mapplet、规则或虚拟存储过程中的一个或多个对象组成。

转换

转换是一种用于生成、修改或传递数据的对象。

Informatica Developer 提供了一组执行特定功能的转换。例如，汇总器转换执行数据组的计算。映射中的转换代表数据集成服务对数据执行的操作。数据通过您在映射或 Mapplet 中链接的转换端口传递。

转换可以是主动转换或被动转换。转换可以连接到数据流，也可以取消转换的连接。有关转换的详细信息，请参阅《Developer 转换指南》。

Mapplet

Mapplet 是一个可重用对象，其中包含一组可在多个映射中使用的转换。

在映射中使用 Mapplet 时，使用的是 Mapplet 的一个实例。对该 Mapplet 所做的任何更改都会由该 Mapplet 的所有实例继承。Mapplet 可以包含其他 Mapplet。您也可以在一个映射或 Mapplet 中多次使用某个 Mapplet。您可以手动创建 Mapplet。也可以从映射或 Mapplet 中的段生成 Mapplet。

有关 Mapplet 的详细信息，请参阅 [第 2 章，“Mapplet” 页面上 30](#)。

段

段由映射、Mapplet、规则或虚拟存储过程中的一个或多个对象组成。段可以包含源、目标、转换或 Mapplet。

可以跨不同的文件夹或项目复制段。复制段时考虑以下规则和准则：

- 如果可能，Developer tool 会重用相关性。如果不能重用相关性，则会复制相关性。
- 如果映射、Mapplet、规则或虚拟存储过程中包含参数，并且复制了引用该参数的转换，则目标对象中的转换将使用该参数的默认值。
- 不能复制输入转换和输出转换。
- 粘贴段之后，不能撤消先前的操作。

也可以从映射或 Mapplet 中的段生成 Mapplet。当映射或 Mapplet 包含您要重用的已连接的转换流时，您可能想要生成 Mapplet。有关生成 Mapplet 的详细信息，请参阅 [“生成 Mapplet” 页面上 33](#)。

复制段

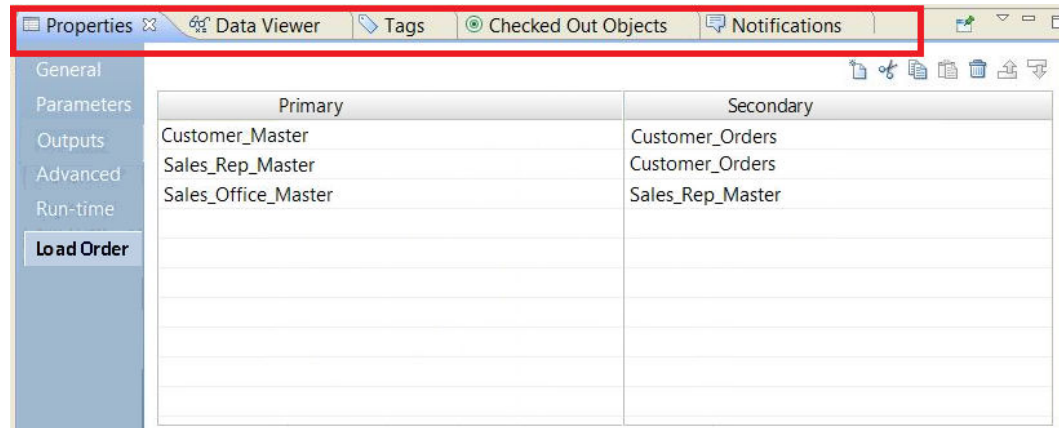
如果您想在其他映射、Mapplet、规则或虚拟存储过程中重复使用一部分映射逻辑，则可以复制段。

1. 打开包含要复制的段的对象。
2. 通过突出显示要复制的每个对象来选择段。
按 Ctrl 键以选择多个对象。还可以通过拖动矩形中的指针使其包含编辑器中的对象来选择段。
3. 单击 **编辑 > 复制** 以将段复制到剪贴板。
4. 打开目标映射、Mapplet、规则或虚拟存储过程。
5. 单击 **编辑 > 粘贴**。

视图

在编辑器中单击时会出现各种视图。视图是可以导航信息层次结构或显示对象属性的工作台部分。您可以在这些视图中切换以便在编辑器中修改属性或其他详细信息。您还可以使用这些视图选择您想让其出现在编辑器中的对象。

下图显示了 Informatica Developer 中的各种视图：



在以下视图之间进行切换以执行不同的任务：

属性

配置常用的映射属性，包括映射名称、运行时属性和加载顺序约束。

数据查看器

预览数据并查看每个转换的映射输出。您还可以在**数据查看器**视图中导出数据。

标记

创建标记来添加元数据、向对象分配标记并查看所有向对象分配的标记。

已签出对象

查看已签出的对象。

通知

为结果卡通知设置并配置全局设置。您还可以在 Informatica 域中选择收件人以便在工作流期间接收通知。

映射验证

开发映射时，必须对其进行配置，以便数据集成服务可以读取和处理整个映射。如果 Developer 工具检测到存在阻止数据集成服务运行某个映射的错误，则会将该映射标记为无效。

Developer tool 将执行以下类型的验证：

- 连接
- 表达式
- 对象

连接验证

每次连接映射中的端口和验证映射时，Developer 工具都将执行连接验证。

连接端口时，Developer 工具将验证连接是否有效。验证映射时，Developer tool 将验证连接是否有效以及是否所有必需端口均已连接。

Developer tool 将执行以下连接验证：

- 至少连接了一个输入对象和一个输出对象。
- 至少将一个 Mapplet 输入端口和输出端口连接到映射。
- 端口之间的数据类型兼容。如果将某个端口的数据类型更改为与连接到的端口不兼容的数据类型，Developer tool 将生成错误，使映射失效。但是，如果更改后的端口与连接的端口保持兼容（例如 Char 和 Varchar），则可以更改数据类型。

表达式验证

开发映射时，可以验证转换中的表达式。如果没有更正错误，则验证映射时**验证日志**视图中将出现错误消息。

如果删除表达式中使用的输入端口，则 Developer 工具会将映射标记为无效。

对象验证

验证映射时，Developer 工具将验证独立对象（例如输入转换或 Mapplet）的定义是否与映射中的实例相匹配。

如果在配置映射时任何对象发生了更改，则映射可能包含错误。如果对象更改时并未配置映射，则 Developer 工具将跟踪这些更改对映射的影响。

映射运行时属性

映射的运行时属性取决于您为映射选择的执行环境。

请配置下列映射运行时属性：

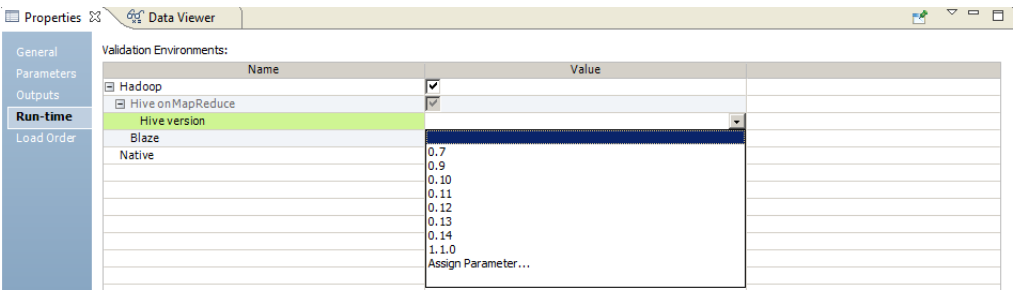
验证环境

指示 Developer tool 是否对本地执行环境和/或 Hadoop 执行环境的映射定义进行验证。在本地环境中运行映射时，数据集成服务会处理映射。

您可以根据自己的许可证在 Hadoop 环境中运行映射。在 Hadoop 环境中运行映射时，数据集成服务会通过 Hadoop 连接将映射执行推送至 Hadoop 群集。Hadoop 群集将会处理映射。

选择 Hadoop 执行环境时，可以选择基于 MapReduce 的 Hive 引擎或 Blaze 引擎来处理映射。也可以选择 Hive 版本。从列表中选择一个版本号或将参数分配给 Hive 版本。必须先在**参数**视图中定义一个参数，之后才能将该参数分配给 Hive 版本属性。参数必须是包含 Hive 版本列表中某个版本的字符串。

下图显示了验证环境：



如果要在 Hadoop 环境中运行映射之前先在本地环境中测试映射，请同时选择两个验证环境。或者，如果要在运行映射时在参数中定义执行环境值，请同时选择两个验证环境。

如果同时选择了两个环境，必须在运行时属性中为映射选择执行环境。

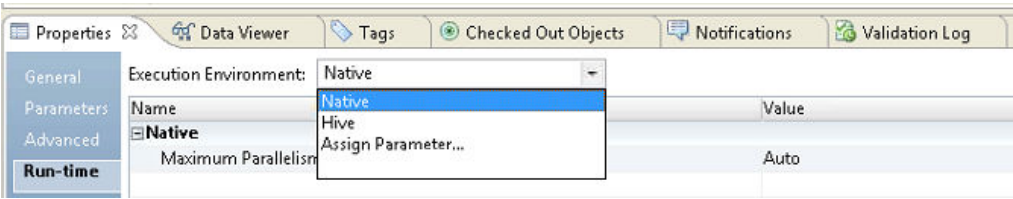
默认为“本地”。

执行环境

选择要在运行映射时使用的执行环境。在本地环境中运行映射时，数据集成服务将处理映射。如果安装了 PowerCenter Big Data Edition，则可以在 Hive 环境中运行映射。当您在 Hive 环境中运行映射时，数据集成服务会通过 Hive 连接将转换逻辑推送到 Hadoop 群集。Hadoop 群集负责处理数据。

可以使用映射参数来指示执行环境。请配置字符串参数。将默认值设置为“本地”或“Hive”。选择执行环境时，请单击“分配参数”，然后选择所配置的参数。

下图显示了选择映射执行环境的位置：



选择执行环境时，Developer tool 会为映射运行保存一个关联的验证环境。

在 PowerCenter Express 中可以使用本地执行环境。

最大并行数

对本地执行环境有效。处理单个映射管道阶段的最大并行线程数量。管理员将数据集成服务的最大并行数值设置为大于 1 的值，以启用映射分区。管理员在 Administrator 工具中设置最大并行数值。

映射的默认最大并行数值为“自动”。每个映射都使用为数据集成服务定义的最大并行数值。您可以更改默认的最大并行数值来为特定映射定义最大值。当数据集成服务和映射的最大并行数值设置为不同的整数值时，数据集成服务将使用最小的值。

PowerCenter Express 不支持此选项。如果输入某个最大并行数值，数据集成服务会忽略该值。

默认为“自动”。最大值为 64。

有关分区的详细信息，请参阅[“已分区的映射概览”](#)页面上 227。

目标提交间隔

要用作提交基础的行数。数据集成服务将根据其处理的目标行数和目标表上的约束提交数据。数据集成服务可以调整提交间隔。默认提交间隔为 10,000 行。

提交间隔是数据集成服务发出提交的近似间隔。数据集成服务可能会在提交间隔之时、之前或之后发出提交。通常，数据集成服务会在写入整个写入器缓冲区块后检查目标提交间隔。

出错时停止

如果读取器、写入器或转换线程中出现非严重错误，则停止映射。默认为禁用。

启用了“出错时停止”时，以下错误类型将导致映射停止：

读取器错误

数据集成服务在读取源数据库或源文件时遇到的错误。在 Unicode 模式下运行会话时，读取器错误可以包括对齐错误。

写入器错误

数据集成服务在向目标数据库或目标文件写入数据时遇到的错误。写入器错误可以包括违反键约束、将空值加载到非空字段以及数据库触发器响应。

转换错误

数据集成服务在转换数据期间遇到的错误。转换错误可以包括转换错误以及任何设置为错误的条件（例如空输入）。

映射模拟用户名

对本地执行环境有效。使其他用户能够模拟连接到使用 Kerberos 身份验证的 Hive、HBase 或 HDFS 源及目标的数据集成服务用户。

按以下格式输入用户名：<Hadoop service name>/<hostname>@<YOUR-REALM>

其中：

- Hadoop service name 是指 Hive、HBase 或 HDFS 源或目标所在的 Hadoop 服务的名称。
- hostname 是指 Hadoop 服务的名称或 IP 地址。
- YOUR-REALM 是指 Kerberos 域。

以下特殊字符只能用作分隔符：“/”和“@”

建议的并行值

当“最大并行数”属性分配了大于 1 的值或分配给参数时，对本地执行环境有效。建议用于处理转换管道阶段的并行线程数。为特定转换定义建议的并行值后，数据集成服务会在确定该转换管道阶段的最佳线程数量时考虑该值。当转换包含许多端口或执行复杂计算时，您可能会希望定义一个建议的并行值来优化转换的性能。

PowerCenter Express 不支持此选项。如果输入了某个建议的并行值，数据集成服务会忽略该值。

默认值为“自动”，这意味着转换使用为映射定义的最大并行数值。最大值为 64。

有关定义建议的并行值的详细信息，请参阅[“转换的建议并行值”页面上 241](#)。

Hive 连接

对 Hive 执行环境有效。Hive 连接定义了数据集成服务将映射执行推送至 Hadoop 群集所需的连接信息。选择 Hive 连接以在 Hadoop 群集中运行映射。您可以为 Hive 连接分配一个用户定义的参数。在映射的[参数视图](#)上定义参数。

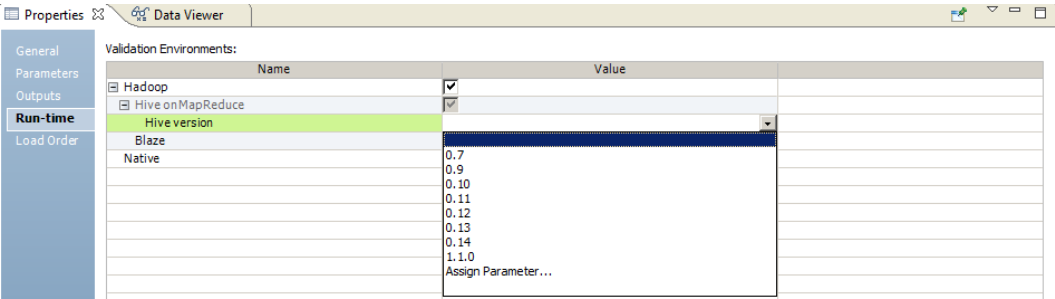
验证环境

验证环境指示 Developer tool 是否对本地执行环境和/或 Hadoop 执行环境的映射定义进行验证。在本地环境中运行映射时，数据集成服务会处理映射。

您可以根据自己的许可证在 Hadoop 环境中运行映射。在 Hadoop 环境中运行映射时，数据集成服务会通过 Hadoop 连接将映射执行推送至 Hadoop 群集。Hadoop 群集将会处理映射。

选择 Hadoop 执行环境时，可以选择基于 MapReduce 的 Hive 引擎或 Blaze 引擎来处理映射。也可以选择 Hive 版本。从列表选择一个版本号或将参数分配给 Hive 版本。必须先在参数视图中定义一个参数，之后才能将该参数分配给 Hive 版本属性。参数必须是包含 Hive 版本列表中某个版本的字符串。

下图显示了验证环境：



如果要在 Hadoop 环境中运行映射之前先在本地环境中测试映射，请同时选择两个验证环境。或者，如果要在运行映射时在参数中定义执行环境值，请同时选择两个验证环境。

如果同时选择了两个环境，必须在运行时属性中为映射选择执行环境。

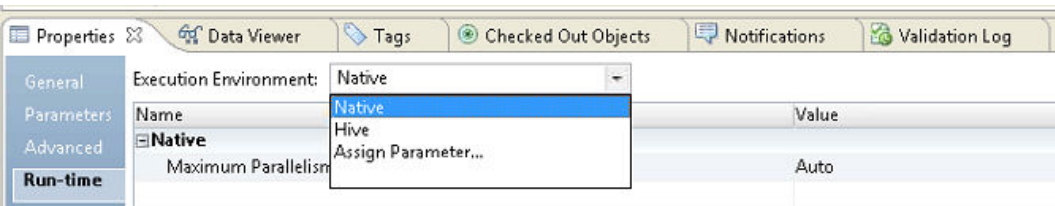
默认值为“本地”。

执行环境

选择要在运行映射时使用的执行环境。在本地环境中运行映射时，数据集成服务会处理映射。如果安装了 Big Data Management™，则可以在该环境中运行映射。数据集成服务会将处理推送到 Hadoop 群集的节点中。选择 Hadoop 环境时，还可以选择引擎将映射逻辑推送到 Hadoop 群集。

可以使用映射参数来指示执行环境。请配置字符串参数。将默认值设置为“本地”或“Hive”。选择执行环境时，请单击分配参数，然后选择所配置的参数。

下图显示了选择映射执行环境的位置：



选择执行环境时，Developer tool 会为映射运行保存一个关联的验证环境。

最大并行数

最大并行数对本地执行环境有效。最大并行数是指处理单个映射管道阶段的最大并行线程数。管理员将数据集成服务的最大并行数设置为大于 1 的值，以启用映射分区。管理员在 Administrator 工具中设置最大并行数。

映射的默认最大并行数值为“自动”。每个映射都使用为数据集成服务定义的最大并行数值。您可以更改默认的最大并行数值来为特定映射定义最大值。当数据集成服务和映射的最大并行数值设置为不同的整数值时，数据集成服务将使用最小的值。

默认为“自动”。最大值为 64。

有关分区的详细信息，请参阅 [第 14 章，“已分区的映射”页面上 227](#)。

目标提交间隔

目标提交间隔是指要用作提交基础的行数。数据集成服务根据处理的目标行数和目标表的约束提交数据。数据集成服务可以调整提交间隔。默认提交间隔为 10,000 行。

提交间隔是数据集成服务发出提交的近似间隔。数据集成服务可能会在提交间隔之时、之前或之后发出提交。通常，数据集成服务会在写入整个写入器缓冲区块后检查目标提交间隔。

出错时停止

如果读取器、写入器或转换线程中出现非严重错误，此功能则会停止映射。默认为已禁用。

启用了“出错时停止”时，以下错误类型将导致映射停止：

读取器错误

数据集成服务在读取源数据库或源文件时遇到的错误。在 Unicode 模式下运行会话时，读取器错误可以包括对齐错误。

写入器错误

数据集成服务在向目标数据库或目标文件写入数据时遇到的错误。写入器错误可以包括违反键约束、将空值加载到非空字段以及数据库触发器响应。

转换错误

数据集成服务在转换数据期间遇到的错误。转换错误可以包括转换错误以及任何设置为错误的条件（例如空输入）。

映射模拟用户名

映射模拟用户名对本地执行环境有效。使用映射模拟来模拟连接到使用 Kerberos 身份验证的 Hive、HBase 或 HDFS 源及目标的数据集成服务用户。

按以下格式输入用户名：<Hadoop service name>/<hostname>@<YOUR-REALM>

其中：

- Hadoop service name 是指 Hive、HBase 或 HDFS 源或目标所在的 Hadoop 服务的名称。
- 主机名是指 Hadoop 服务的名称或 IP 地址。
- YOUR-REALM 是指 Kerberos 域。

您只能将以下特殊字符用作分隔符：“/”和“@”

建议的并行值

当“最大并行数”属性分配了大于1的值或分配给参数时，建议的并行值对本地执行环境有效。建议用于处理转换管道阶段的并行线程数。

为转换定义建议的并行值后，数据集成服务会在确定该转换管道阶段的最佳线程数量时考虑该值。当转换包含许多端口或执行复杂计算时，您可能会希望定义一个建议的并行值来优化转换的性能。

默认值为“自动”，这意味着转换使用为映射定义的最大并行数值。最大值为 64。

Hive 连接

Hive 连接对 Hive 执行环境有效。Hive 连接定义了数据集成服务将映射执行推送至 Hadoop 群集所需的连接信息。

选择 Hive 连接以在 Hadoop 群集中运行映射。您可以为 Hive 连接分配一个用户定义的参数。在映射的**参数视图**上定义参数。

目标加载顺序约束

目标加载顺序约束可限制数据集成服务加载行并将其提交到同一映射中两个彼此相关的目标实例的方式。

在 Developer tool 中，您可以配置约束以限制数据集成服务向目标表中加载行的顺序。

可以将约束配置为强制数据集成服务在主目标实例数据加载完毕后再将数据加载到辅助目标实例。定义为主目标和辅助目标的表取决于输入行中的事务。

对于目标加载顺序约束，请考虑以下方案：

将行插入到主数据和详细信息目标。

如果要行插入到具有主键-外键关系的目标，可以配置目标加载顺序约束。将具有主键的目标配置为主目标实例。将具有外键的目标配置为辅助目标实例。数据集成服务可以暂存辅助目标的数据，直至其完成主目标的加载。

从主数据和详细信息目标中删除行。

如果要从具有主键-外键关系的目标中删除行，需要配置不同的约束。将具有外键的目标配置为主目标实例，以便先删除详细信息目标中的行。将具有主键的目标配置为辅助目标实例。

将行插入同一关系表并进行更新。

可以为映射配置目标加载顺序约束，以将插入行和更新行从两个单独的转换加载到一个关系表。配置该约束以限制数据集成服务在加载插入行之后再加载更新行。

可以在一个映射中配置多个约束。数据集成服务会确定最有效的执行计划来加载目标，而不会违反约束。

与插入行和删除行相关的约束

目标加载顺序约束在处理同一个文件中的插入行、更新行和删除行时不会进行特殊处理。

如果需要处理插入行、更新行和删除行，您可以配置路由器转换，使其将插入行和更新行返回到与删除行不同的目标实例。请配置目标加载顺序约束，以指定目标加载顺序。

例如，您可能有 Order_Header 和 Order_Detail 两个目标，而 Order_Detail 表包含指向 Order_Header 表的 OrderID 外键，那么您需要处理这两个表中的插入行、更新行和删除行。

您可以使用路由器转换来将插入行和更新行与删除行分开。可以通过路由器转换配置以下输出组：

1. Order_Header 插入行和更新行
2. Order_Header 删除行
3. Order_Detail 插入行和更新行
4. Order_Detail 删除行

针对将这些行加载到目标上这一操作，您可以创建以下约束：

```
Group #4 before group #2
Group #2 before group #1
Group #1 before group #3
```

这些约束会强制数据集成服务先处理 Order_Detail 中的删除行再处理 Order_Header 中的删除行。数据集成服务会先处理所有删除行再处理插入行和更新行，而且会先处理 Order_Header 中的插入行和更新行再处理 Order_Detail 中的插入行和更新行。

目标加载顺序的规则和准则

定义目标加载顺序约束时，请考虑以下规则和准则：

- 在 Developer tool 中，可以将某些目标列配置为主键或外键。加载顺序约束将忽略这些键。如果目标具有主键-外键约束，则必须定义加载顺序约束。
- 当您定义加载顺序约束时，Developer tool 并不会对其进行验证。Developer tool 将在验证映射时对加载顺序约束进行验证。
- 数据集成服务可以将数据暂存在目标加载顺序约束中第二个目标实例的本地磁盘中。当映射具有多个辅助目标实例时，数据集成服务会在不违反约束的情况下将暂存的数据加载到目标。
- 数据集成服务会先后加载两个目标实例，但不会确定行是插入项、删除项还是更新项。对于具有主键外键约束的目标表，孤行是外键目标中没有主键目标匹配行的行。数据集成服务不会检查是否存在孤行，而是按您在加载顺序约束中指定的顺序加载所有行。

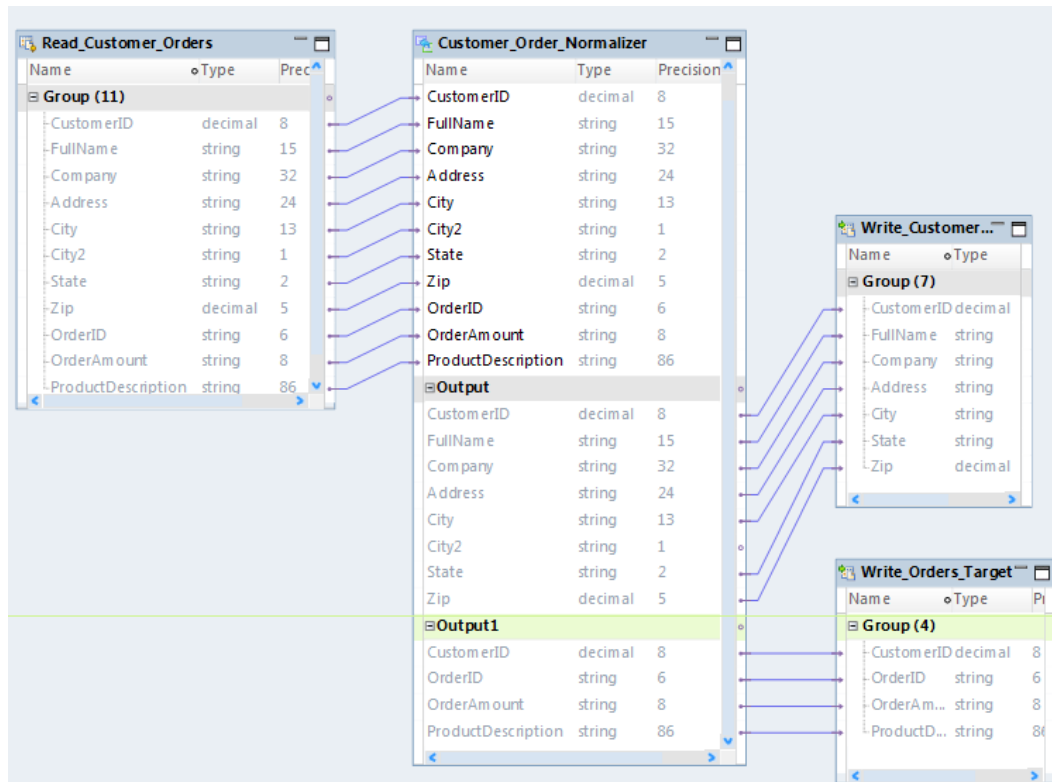
警告：Informatica 强烈建议不要使用暂存文件或其内的数据。对于因客户更改暂存文件或表中数据而导致的数据损坏，Informatica 概不负责。暂存文件的结构可能随 Informatica 版本而更改。

目标加载顺序示例

组织一天处理客户订单两次。客户信息和订单信息接收到同一事务文件中。组织需要确保处理订单文件的映射先加载客户信息，之后再加载订单。

开发人员创建一个映射，以将客户信息返回至 Customer_Target 表，并将订单返回至 Orders_Target 表。Customer_Master 的主键为 CustomerID。“订单”表中的每个订单都有一个指向 Customer_Master 中 CustomerID 的外键。开发人员创建一个目标加载顺序约束，以限制数据集成服务在将客户信息加载到目标之后再加载订单。

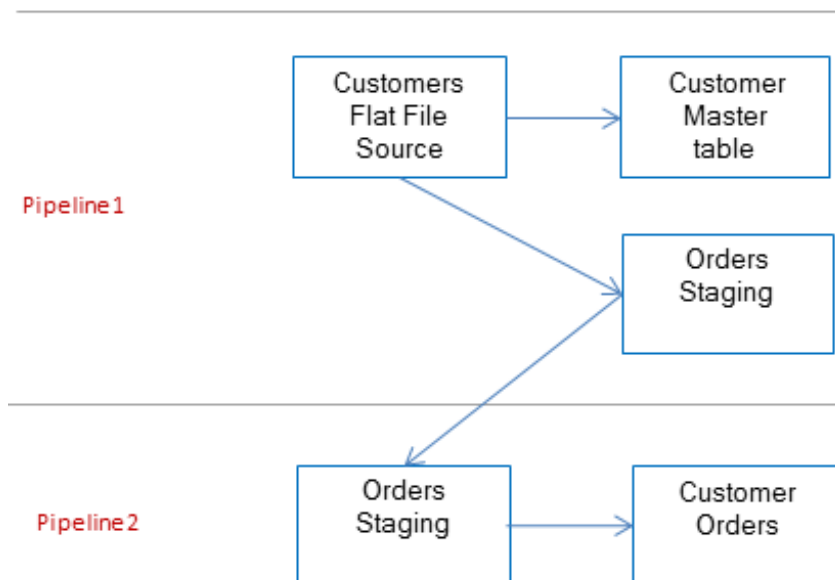
下图显示了该映射：



规范器转换会为客户和订单数据创建单独的输出组。开发人员需要确保客户数据在订单数据之前加载到目标。

数据集成服务可以使用不同的策略来实现目标加载顺序约束。在本示例中，数据集成服务创建两条管道来将客户数据和订单数据加载到目标表。

下图显示了用于将客户数据和订单数据加载到目标表的管道：



在第一条管道中，数据集成服务将客户加载到 Customer_Master 并将订单暂存至本地磁盘文件。在第二条管道中，数据集成服务将暂存订单加载到“订单”表。

如何开发映射

根据业务需要，可以开发用于读取、转换和写入数据的映射。

要开发映射并运行工作流，请执行以下任务：

1. 确定要创建的映射类型。
2. 创建要在映射中使用的输入、输出和可重用对象。创建物理数据对象、逻辑数据对象或虚拟表来用作映射输入或输出。
3. 创建要使用的可重用转换。如果要使用 Mapplet，则还必须创建 Mapplet。
4. 创建映射。
5. 将对象添加到映射。必须向映射添加输入和输出对象。可以选择添加转换和 Mapplet。
6. 链接映射对象之间的端口，并通过随着流添加、删除或修改数据的 Mapplet 和转换来创建源到目标的数据流。
7. 验证映射以发现错误。
8. 将映射保存到模型存储库。
9. 运行工作流以查看映射输出。

创建映射

创建映射，使数据在源和目标之间移动，并转换数据。

1. 在**对象浏览器**视图中选择一个项目或文件夹。
2. 单击**文件 > 新建 > 映射**。
3. 输入映射名称。
4. 单击**完成**。

此时编辑器会显示一个空的映射。

将对象添加到映射

将对象添加到映射可以确定源和目标之间的数据流。

- 将数据对象拖动到编辑器中，然后选择“读取”以将数据对象添加为源。
- 将数据对象拖动到编辑器中，然后选择“写入”以将数据对象添加为目标。
- 要添加查找转换，请将平面文件数据对象、逻辑数据对象、引用表或关系数据对象拖动到编辑器中，然后选择“查找”。
- 要添加可重用转换，请从**对象浏览器**视图的转换文件夹中将转换拖动到编辑器中。
为每个要添加的可重用转换重复此步骤。
- 要添加不可重用转换，请选择**转换**选项板上的转换，然后将其拖动到编辑器中。
为每个要添加的不可重用转换重复此步骤。
- 为每个不可重用转换配置端口和属性。
- 可以选择将 Mapplet 拖动到编辑器中。

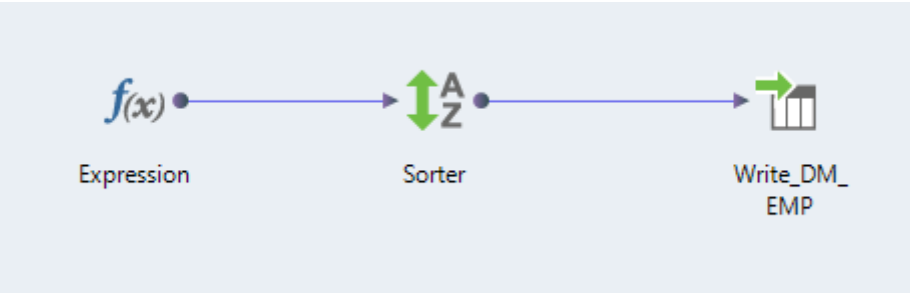
连接映射对象

通过连接映射对象完成映射。通过端口连接这些映射对象。数据通过输入端口、输出端口和输入/输出端口来传入和传出转换。

将对象添加到映射后，可以根据您希望数据集成服务转换数据的方式来连接属性。编辑器通过以下方式显示映像对象：

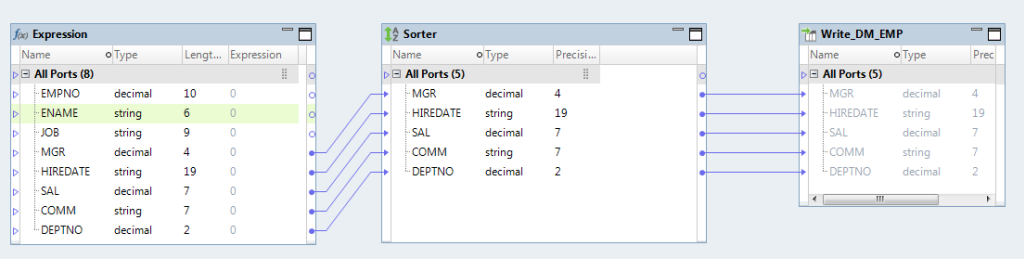
- 图标视图。显示包含对象名称的对象图标。

下图显示了具有图标视图对象的映射：



- 普通视图。显示列以及输入和输出端口指示器。您可以连接普通视图中的对象。

下图在普通视图中显示了上一个图标视图映射：



在输入对象、转换、Mapplet 和输出对象之间链接端口时，可以创建以下链接类型：

- 一对一链接。将输入对象的一个端口链接到输出对象的一个端口。
- 一对多链接。将一个端口链接到多个输出对象。还可以将一个对象的多个端口链接到多个输出对象。

可以手动或自动链接端口：

- 手动链接端口。您可以手动链接一个或多个端口。将输入对象的端口拖动到输出对象的端口。
- 自动链接端口。自动链接端口时，可按位置或名称进行链接。

有关链接端口的详细信息，请参阅《Developer 转换指南》。

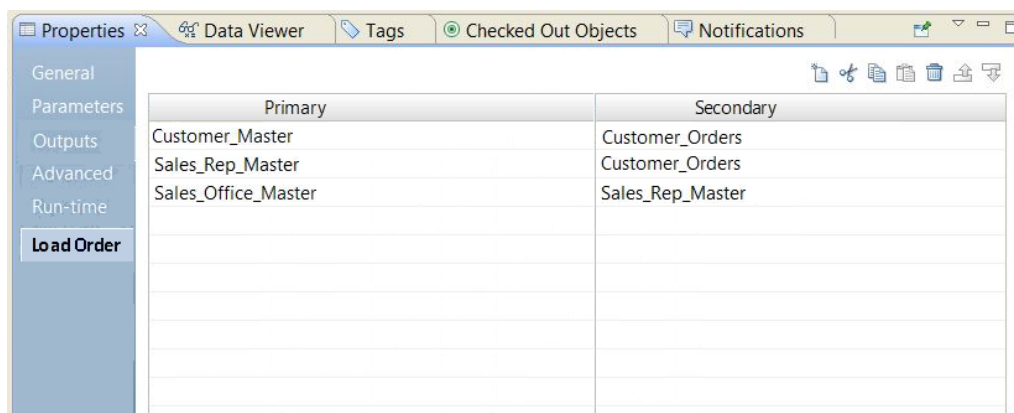
创建目标加载顺序约束

目标加载顺序约束会限制数据集成服务将行加载到同一个映射的目标表的顺序。在映射的**加载顺序**选项卡上创建目标加载顺序约束。

1. 在编辑器中单击。
底部窗口会出现映射**属性**选项卡。

2. 单击**加载顺序**选项卡。

下图显示了**加载顺序**选项卡：



该图显示了三个约束。每个约束均包含主要目标和辅助目标。这些约束指定数据集成服务必须在加载 Customer_Orders 目标之前加载 Customer_Master 目标。Sales_Rep_Master 必须在 Customer_Orders 目标之前加载。Sales_Office_Master 必须在 Sales_Rep_Master 之前加载。

3. 要输入约束，请单击**新建**按钮。

Developer tool 将为约束创建一个行。

4. 单击**主要**字段。

此时将显示映射中的目标实例的列表。

5. 选择要首先加载的目标实例。

6. 在**辅助**字段中，选择要其次加载的目标实例。

可以按任何顺序输入约束。可输入以下几对约束以指示与上图相同的限制：

主要	次要
Sales_Office_Master	Sales_Rep_Master
Sales_Rep_Master	Customer_Master
Customer_Master	Customer_Orders

7. 根据需要创建任意数量的约束。

验证映射

通过验证映射，可确保数据集成服务能够读取和处理整个映射。

1. 单击**编辑 > 验证**。

验证日志视图中将出现错误。

2. 修复错误，然后重新验证映射。

运行映射

通过运行映射，可以将输出从源移动到目标并转换数据。

如果没有选择默认数据集成服务，那么在预览数据或运行映射时，Developer tool 会提醒您进行选择。

- ▶ 右键单击编辑器中的空白区域，然后单击**运行映射**。

数据集成服务将运行映射，并将输出写入到目标。

数据集成服务配置为使用操作系统配置文件时，它通过操作系统配置文件运行映射。

第 2 章

Mapplet

本章包括以下主题：

- [Mapplet 概览, 30](#)
- [Mapplet 类型, 31](#)
- [Mapplet 输入和输出, 32](#)
- [生成的 Mapplet, 32](#)
- [创建 Mapplet, 34](#)
- [Mapplet 验证, 34](#)

Mapplet 概览

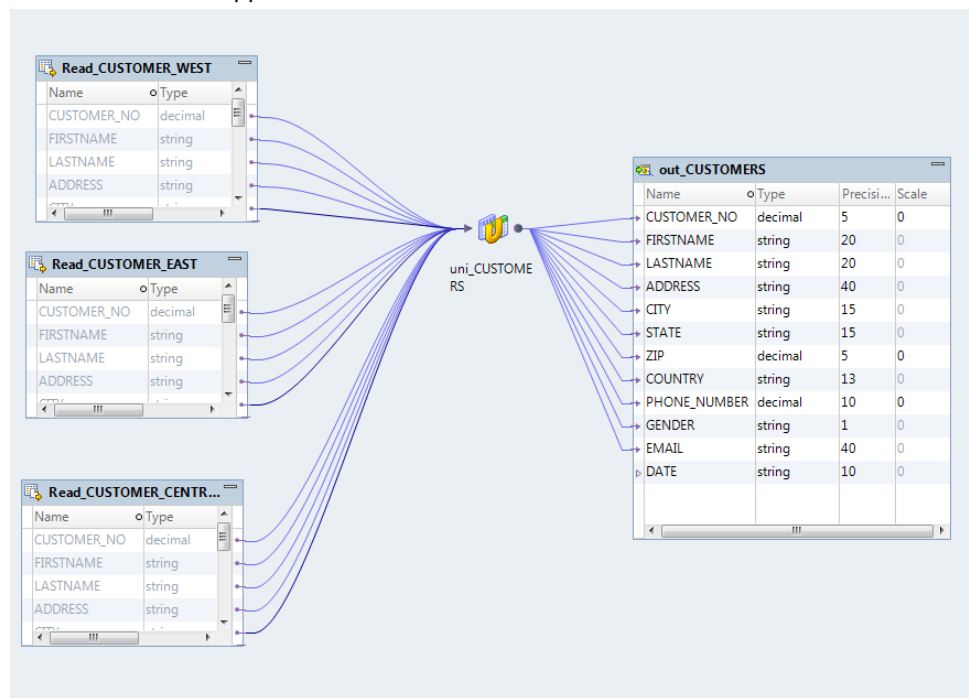
Mapplet 是一个可重用对象，其中包含一组可在多个映射中使用的转换。可在一个映射中使用一个 Mapplet。或者，也可以将该 Mapplet 验证为规则。

Mapplet 中的转换可以是可重用转换，也可以是不可重用转换。如果要将序列生成器转换添加到 Mapplet，则转换必须是可重用转换。

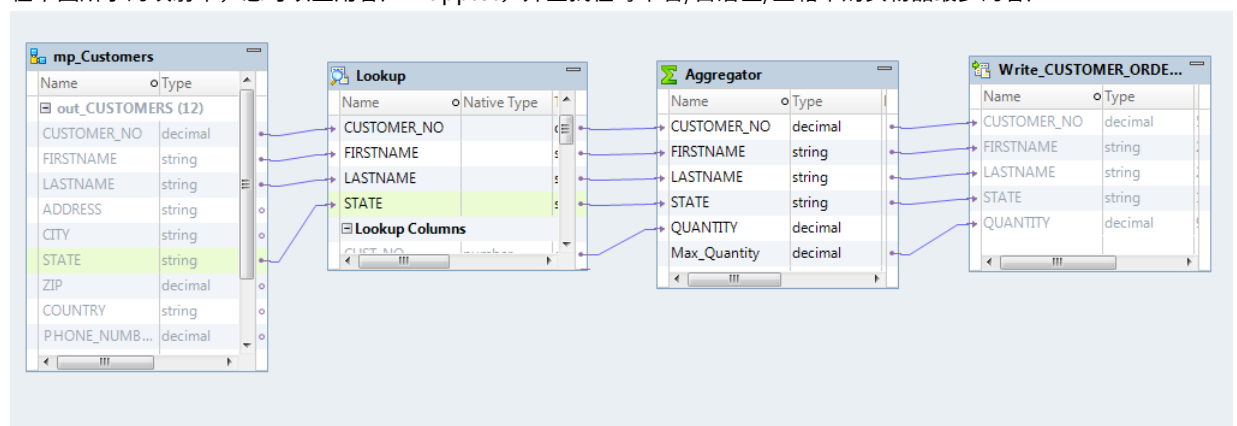
在映射中使用 Mapplet 时，使用的是 Mapplet 的一个实例。对该 Mapplet 所做的任何更改都会由该 Mapplet 的所有实例继承。

Mapplet 可以包含其他 Mapplet。您也可以在一个映射或 Mapplet 中多次使用某个 Mapplet。不能对 Mapplet 进行循环嵌套。例如，如果 Mapplet A 包含 Mapplet B，则 Mapplet B 不能包含 Mapplet A。

例如，下图所示的 Mapplet 借助于联合转换将东部、西部和中部区域的客户信息组合在一起：



在下图所示的映射中，您可以重用客户 Mapplet，并查找在每个省/自治区/直辖市购买物品最多的客户：



您可以手动创建 Mapplet。也可以从映射或 Mapplet 中的段生成 Mapplet。

Mapplet 类型

Mapplet 类型取决于 Mapplet 输入和输出。

可以创建或生成以下类型的 Mapplet：

- 源。此 Mapplet 包含一个数据源来作为输入，并包含一个输出转换来作为输出。
- 目标。此 Mapplet 包含一个输入转换来作为输入，并包含一个数据源来作为输出。
- 中游。此 Mapplet 包含一个输入转换和一个输出转换。它不包含输入或输出数据源。

Mapplet 输入和输出

要在映射中使用 Mapplet，必须为其配置输入和输出。

Mapplet 具有以下输入和输出组件：

- Mapplet 输入。可以将数据从数据源和/或输入转换传递到 Mapplet 中。如果要将该 Mapplet 验证为规则，则必须通过输入转换将数据传递到该 Mapplet 中。使用输入转换时，可将其连接到映射中的源或上游转换。
- Mapplet 输出。可以将数据源和/或输出转换中的数据从 Mapplet 传递出来。如果要将 Mapplet 验证为规则，则必须通过输出转换从 Mapplet 传递数据。使用输出转换时，可将其连接到映射中的目标或下游转换。
- Mapplet 端口。可以在映射编辑器中查看 Mapplet 端口。Mapplet 输入端口和输出端口来自输入转换和输出转换。它们并不来自数据源。

Mapplet 输入

Mapplet 输入可以来自数据源或输入转换。

可以在一个 Mapplet 中创建多个管道。可以使用多个数据源或输入转换。也可以将多个数据源和输入转换组合在一起。

可使用一个或多个数据源在 Mapplet 中提供源数据。在映射中使用 Mapplet 时，该 Mapplet 是映射管道中的第一个对象，并且不包含输入端口。

使用输入转换可从映射接收输入。输入转换提供输入端口，以便您可以通过 Mapplet 传递数据。输入转换中连接到 Mapplet 中其他转换的每个端口都将成为一个 Mapplet 输入端口。输入转换可以从单个活动源接收数据。未连接的端口不会出现在映射编辑器中。

可以将一个输入转换连接到 Mapplet 中的多个转换。也可以将输入转换中的一个端口连接到 Mapplet 中的多个转换。

Mapplet 输出

如果要创建目标 Mapplet，请使用数据源作为输出。使用 Mapplet 中的输出转换通过 Mapplet 将数据传递到映射中。

使用一个或多个数据源在 Mapplet 中提供目标数据。在映射中使用 Mapplet 时，该 Mapplet 是映射管道中的最后一个对象，并且不包含输出端口。

使用输出转换将输出传递到映射中的下游转换或目标。输出转换中连接的每个端口都会在映射中显示为一个 Mapplet 输出端口。Mapplet 中的每个输出转换都显示为一个输出组。一个输出组可以将数据传递到映射中的多个管道。

生成的 Mapplet

可以从映射或 Mapplet 中的段生成 Mapplet。当映射或 Mapplet 包含您要重用的已连接的转换流时，您可能想要生成 Mapplet。

Developer tool 会在生成过程中将段验证为 Mapplet。请查看生成的 Mapplet 的规则和准则以避免验证错误。

生成的 Mapplet 规则和准则

如果以下任意条件成立，Mapplet 生成将失败：

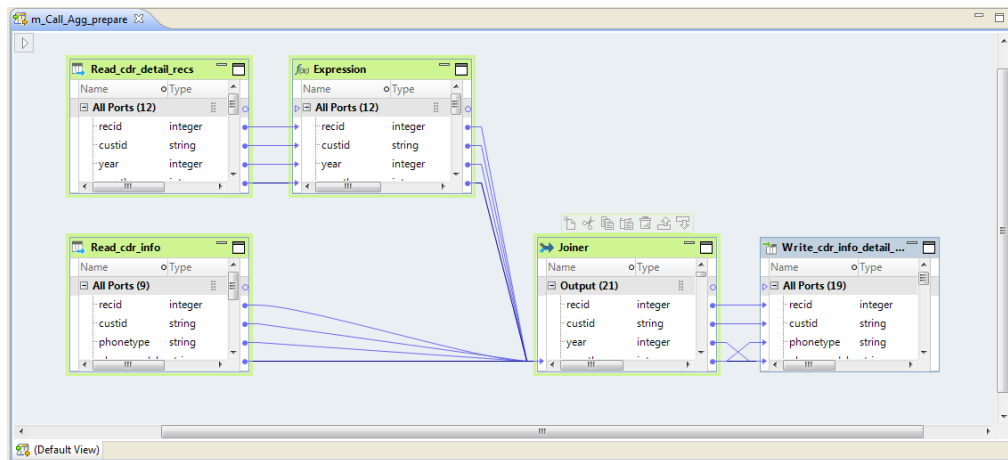
- 选定的转换不按顺序。
- 段同时包含读取转换和写入转换。但是，段可以包含多个读取转换或多个写入转换。
- 段包含不可重用的序列生成器转换、输入转换、输出转换或含有设置操作的转换。
- 选定的段未将所有转换包含在管道分支中。
- 段中的第一个转换和最后一个转换包含动态字段。
- 段包含指向第一个转换的传入运行时链接，或包含来自最后一个转换的传出运行时链接。
- 段包含单个参数化的读取转换、写入转换或查找转换。

生成 Mapplet

从包含已连接的转换的段生成 Mapplet。该段可以包含读取转换、写入转换或中游转换。

1. 打开包含要生成为 Mapplet 的段的映射或 Mapplet。
2. 选择要包含在 Mapplet 中的映射。
3. 右键单击选定的转换之一，然后选择**提取 Mapplet**。

下图显示了包含四个选定的转换的映射：



生成过程会验证该段并报告任何验证错误。

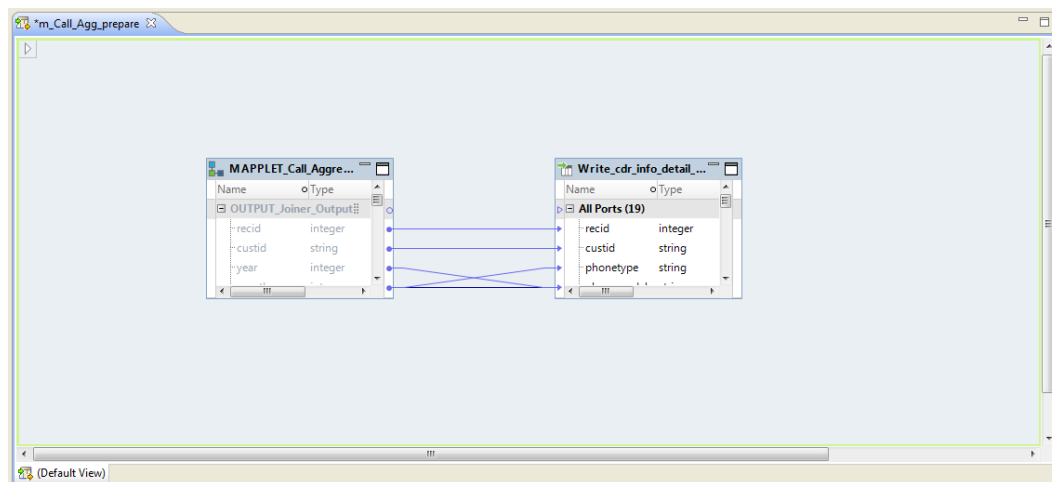
4. 浏览到要在其中生成 Mapplet 的 Mapplet 节点。

默认情况下，Developer tool 将在当前项目内的 Mapplet 节点中生成 Mapplet。

5. 单击**完成**。

Mapplet 将替换在原始映射或 Mapplet 中选择的转换。工具还会将输入转换或输出转换添加到该 Mapplet，具体取决于该段包含读取转换还是写入转换或中游转换。

下图显示了由 Mapplet 替换的所选转换：



请注意，必须显式保存已修改的映射或 Mapplet 以将选定的转换替换为 Mapplet。要使映射或 Mapplet 恢复为原始状态，请选择**文件 > 撤消**三次。

创建 Mapplet

创建 Mapplet 以定义一个可重用对象，其中包含一组可在多个映射中使用的转换。

1. 在**对象浏览器**视图中选择一个项目或文件夹。
2. 单击**文件 > 新建 > Mapplet**。
3. 输入 Mapplet 名称。
4. 单击**完成**。

此时编辑器会显示一个空的 Mapplet。

5. 添加 Mapplet 输入、输出和转换。

Mapplet 验证

您可以在将 Mapplet 添加到映射之前对其进行验证。也可以验证 Mapplet 以在配置文件中将其用作规则。

验证 Mapplet

您可以在将 Mapplet 添加到映射之前对其进行验证。也可以将 Mapplet 验证为规则以包含在配置文件中。

1. 右键单击 Mapplet 编辑器。
2. 选择**验证为 > Mapplet** 或 **验证为 > 规则**。

验证日志会显示所发生的任务错误。

将 Mapplet 验证为规则

规则是指一种业务逻辑，用于定义在运行配置文件时对源数据应用的条件。它是一个在配置文件中使用的中游 Mapplet。可以将您要在配置文件中使用的 Mapplet 验证为规则。

规则必须符合以下要求：

- 规则必须包含输入转换和输出转换。不能在规则中使用数据源。
- 规则可以包含表达式转换、查找转换和被动数据质量转换。规则不能包含其他任何类型的转换。例如，规则不能包含匹配转换，因为它是主动转换。

规则可以包含表达式转换和查找转换。规则不能包含其他任何类型的转换。

- 规则不指定输入组之间的基数。

注意：规则功能不仅限于进行剖析。您可以在 Analyst 工具中将验证为规则的任何 Mapplet 添加到配置文件中。例如，您可以通过选择一个为验证通信地址而配置的规则，然后将其添加到配置文件中，以评估通信地址数据质量。

第 3 章

映射参数

本章包括以下主题：

- [映射参数概览, 36](#)
- [系统参数, 37](#)
- [用户定义参数, 38](#)
- [创建用户定义参数的位置, 39](#)
- [分配参数的位置, 39](#)
- [映射中的参数, 48](#)
- [Mapplet 中的参数, 49](#)
- [逻辑数据对象中的参数, 51](#)
- [虚拟表映射中的参数, 51](#)
- [参数集, 52](#)
- [参数文件, 53](#)
- [如何配置参数, 60](#)

映射参数概览

映射参数表示可以在映射运行之间更改的常量值。创建参数后，可以使用不同的值重新运行映射。使用参数可更改连接、文件目录、表达式组件、端口列表、端口链接和任务属性的值。

可以配置系统参数或用户定义参数。

系统参数。

数据集成服务的内置参数。系统参数可定义数据集成服务存储日志文件、缓存文件、拒绝文件、源文件、目标文件和临时文件的目录。管理员在 Administrator 工具中定义数据集成服务的系统参数默认值。

用户定义参数。

您在转换、逻辑数据对象、映射和工作流中定义的参数。创建用户定义参数后，可以使用不同的连接、平面文件、缓存文件、临时文件、表达式、端口或引用表值重新运行映射。创建用户定义参数后，可以使用不同的表达式、连接、文件、端口或资源值重新运行映射。

可以使用参数确定运行时在动态映射中使用哪些生成的端口。可以配置参数指示要在运行时链接哪些端口。可以分配参数以更改读取、写入或查找转换中的数据对象。

通过向映射分配参数集或参数文件，可以替代参数值。参数集是包含映射参数值的存储库对象。参数文件是包含参数值的 XML 文件。当您使用某个参数集或参数文件运行映射时，数据集成服务将使用在该参数集或参数文件中定义的参数值。这些值将替代您在转换、映射、Mapplet 或工作流中配置的默认参数值。

有关工作流参数的详细信息，请参阅《*Informatica Developer 工作流指南*》。

相关主题：

- [“动态映射中的参数” 页面上 103](#)

系统参数

系统参数是常量值，可定义数据集成服务存储缓存文件、拒绝文件、源文件、目标文件、日志文件和临时文件的目录。

可在数据集成服务的执行选项中定义一些系统参数的值。管理员可以从 Administrator 工具中更新这些值。数据集成服务将在运行时确定其他系统参数的值。不能替代参数文件或参数集中的系统参数值。

您无法创建系统参数。Developer tool 提供了预定义的系统参数列表，可以将这些参数分配给映射中的数据对象或转换。例如，创建汇总器转换时，缓存目录系统参数是分配给 Informatica Administrator 中缓存目录字段的默认值。如果要使用一个不同的缓存目录位置，请创建一个用户定义参数并配置一个默认参数值。

下表介绍了系统参数：

系统参数	类型	说明
CacheDir	字符串	索引和数据缓存文件的默认目录。
LogDir	字符串	映射任务日志文件的默认目录。
RejectDir	字符串	拒绝文件的默认目录。
SourceDir	字符串	源文件的默认目录。
TargetDir	字符串	目标文件的默认目录。
TempDir	字符串	临时文件的默认目录。
ApplicationName	字符串	应用程序的名称
ExecutionEnvironment	字符串	Hadoop 环境或本地环境。
MappingName	字符串	正在运行的映射的名称。
MappingRunStartTime	日期/时间	正在运行的映射的开始时间。
ServiceName	字符串	数据集成服务名称。
UserName	字符串	正在运行映射的用户的名称。

用户定义参数

用户定义参数表示可在映射运行之间更改的常量值。

例如，创建处理客户订单的映射。该映射从包含某个国家/地区的客户数据的关系表中读取客户信息。您要将该映射用于美国、加拿大和墨西哥的客户。创建一个用户定义参数以表示与客户表的连接。创建三个参数集，分别将连接名称设置为美国客户表、加拿大客户表和墨西哥客户表。对于每次映射运行，使用一个不同的参数集运行映射。

您可以创建以下类型的参数：

连接参数

Informatica 连接名称。

日期/时间参数

日期。

表达式

用于定义联接条件、筛选器表达式或查找条件的表达式。

输入链接集

要在**运行时链接**对话框中链接的一组端口。

数值参数

整数、长整型、小数和双精度参数。

端口

单个端口的名称。您可以使用等级转换的等级端口中的端口参数。

端口列表

要包括某个组的端口列表。例如，您可以在汇总器转换或等级转换中使用端口列表参数。

资源

关系数据对象的表、视图或同义词名称。资源名称参数化后，数据集成服务会使用运行时查询中的参数值来提取对象。

排序列表

要使用排序器转换进行排序的端口列表。该列表包括用于升序或降序排序顺序的端口名称和指示器。

字符串

字符串参数表示平面文件名称、目录、表名称或运行时属性。定义精度等于或小于 32768 个字符的字符串参数。

创建参数时，无法将美元符号 (\$) 用作参数名称的前导字符。

使用参数设置属性值时，必须使用该属性的正确参数类型。例如，无法将连接类型的参数用于目标文件名。如果在数值表达式中使用参数，必须使用数值参数类型。

在关系数据对象中，不需要对 SQL 替代、筛选器条件、联接条件中的美元符号 (\$) 进行转义。数据集成服务会将 SQL 语句中以美元符号开头的字段视为参数。

参数不能包含一系列值。如果您在参数中提供一系列的值，数据集成服务会将参数值视为单个字符串值。

例如，您拥有参数 \$IndexParameter1 (value 2) 和 \$IndexParameter2 (value1, value2, value3)。将这些参数包括在表达式 INDEXOF 中，如下所示：

```
INDEXOF($IndexParameter1,'value1','value2','value3')
```

数据集成服务会返回 0 而不是值 2。

日期/时间参数

可以创建日期参数并在表达式中使用这些参数。

必须按以下格式之一定义日期参数：

MM/DD/RR
MM/DD/YYYY
MM/DD/YYYY HH24:MI
MM/DD/RR HH24:MI
MM/DD/RR HH24:MI:SS
MM/DD/YYYY HH24:MI:SS
MM/DD/RR HH24:MI:SS.NS
MM/DD/YYYY HH24:MI:SS.NS

创建用户定义参数的位置

可以在平面文件数据对象、转换、自定义数据对象、Mapplet、映射和工作流中创建用户定义参数。创建参数后，可以将参数分配给条件、表达式、连接、目录和文件名等字段。

为某个转换、逻辑数据对象、Mapplet、映射或工作流创建参数后，参数将应用于该对象。例如，您在转换中创建一个参数，然后将该转换添加到 Mapplet。您可以使用转换中的默认参数值，也可以创建一个 Mapplet 参数以替代转换参数值。

要替代转换参数默认值，需将 Mapplet 参数绑定到转换参数。更改 Mapplet 参数的默认值。可以使用映射参数替代 Mapplet 参数。

以下列表显示了可以创建参数的位置：

- Workflow parameters
- Mapping parameters
- Mapplet parameters
- Logical data objects
- Transformation/data object parameters

在运行时，可以通过配置参数集或参数文件中的参数值设置工作流参数值和映射参数值。

可以在向字段和属性分配参数的同时创建参数。向字段分配参数时，可以创建要使用的参数。也可以浏览以前创建的参数。

在转换或数据对象的**参数**选项卡中维护用户定义参数。映射、Mapplet、工作流或逻辑数据对象还具有一个**参数**选项卡。可以在**参数**选项卡上添加、更改和删除参数。

注意：在**参数**选项卡上创建参数时，请勿在参数名称中使用前置美元符号 (\$)。

分配参数的位置

可以将用户定义参数和系统参数分配给字段。必须先创建用户定义参数，然后才能将其分配给字段。

可以参数化对象和转换中的某些属性。如果可以将参数分配给属性，则当您配置属性值时将会显示该选项。

您可以参数化可重用转换或不可重用转换的属性。将可重用转换添加到映射时，可以使用映射参数替代默认值。如果参数化了不可重用转换，则参数为映射参数。

读取转换和写入转换是您基于物理数据对象创建的不可重用转换。您可以参数化用于创建转换的物理数据对象，也可以将参数分配给读取和写入转换中的某些属性。

下表列出了可分配参数的对象和字段：

对象	字段
所有转换	链接解析顺序
关联转换	缓存文件目录 缓存文件大小
地址验证器转换	大小写样式 默认国家/地区 地理编码数据类型 全局最大字段长度 行分隔符 最大结果计数 优化级别 标准化无效地址
汇总器转换	缓存目录 表达式元素。并非完整表达式。 分组依据
离散记录异常转换	阈值下界 阈值上界
大小写转换器转换	引用表。
合并转换	缓存文件目录 缓存文件大小
自定义数据对象	连接 数据对象 所有者 SQL 查询元素 表名称
判定转换	判定脚本。
重复记录异常转换	缓存文件目录 阈值下界 阈值上界
表达式转换	表达式元素。并非完整表达式。 端口选择器
筛选器转换	筛选条件元素 筛选条件。完整表达式。

对象	字段
平面文件数据对象	控制文件目录 控制文件名 连接名称 默认小数位数 平面文件分隔符 合并文件目录 源文件目录 源文件名 输出文件名 输出文件目录 拒绝文件目录 目标目录
联接器转换	缓存目录 联接条件元素 端口选择器
键生成器转换	缓存文件目录 缓存文件大小
标签创建器转换	引用表
没有查找源的物理数据对象的查找转换	数据对象。不可重用的转换。 动态端口规则。不可重用的转换。 查找条件。完整表达式、不可重用转换。 端口选择器。不可重用的转换。
映射	Hive 版本 运行时环境 最大并行数
匹配转换	“匹配输出”选项卡上的缓存目录 “匹配类型”选项卡上的缓存目录 “匹配类型”选项卡上的索引目录 持久性方法 阈值
非关系数据对象	连接
等级转换	缓存目录 表达式元素。并非完整表达式。 分组依据端口 等级端口
读取转换	连接 数据对象 所有者名称。仅限关系。 资源/表名称。仅限关系。

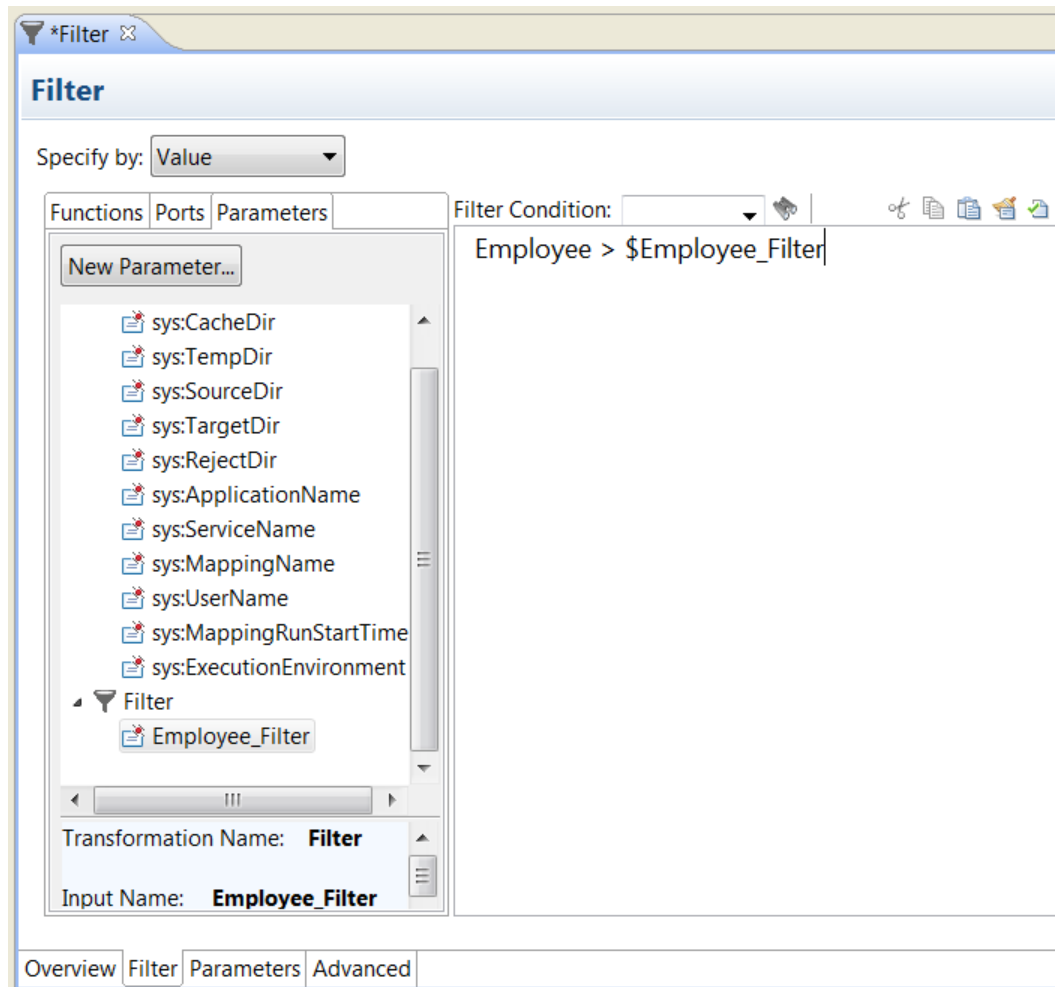
对象	字段
关系数据对象	筛选条件元素 联接条件元素 PreSQL 查询元素 PostSQL 查询元素 SQL 替代元素
路由器转换	组筛选条件元素。 组筛选条件。完整表达式。
排序器转换	排序键 分组依据 工作目录
SQL 转换	连接
标准创建器转换	引用表
标志解析器转换	引用表
更新策略转换	更新策略表达式元素。 更新策略表达式。完整表达式。
写入转换	数据对象 链接解析顺序 拒绝目录 拒绝文件名

表达式中的参数

您可以在汇总器转换、查找转换、表达式转换和筛选器转换等转换的表达式或条件中配置参数。

例如，在筛选器转换中配置筛选条件。选择要包括在条件中的端口和参数。选择要包括在筛选条件中的系统参数或用户定义参数。

下图所示的筛选条件包含员工端口和 Employee_Filter 参数：



在将端口名称作为参数接受的参数中，可以在表达式中使用参数。不能使用参数替换表达式中的常量参数。

例如，请考虑将字符串转换为小数值的 TO_DECIMAL 表达式：

```
TO_DECIMAL( value [, scale] )
```

小数位数参数必须是表达式中的一个常量值。

由于以下表达式包含一个常量小数位数参数，因此该表达式有效：

```
TO_DECIMAL( Input_Port,10 )
```

由于以下表达式包含一个用户定义的小数位数参数，因此该表达式无效：

```
TO_DECIMAL( Input_Port,$Scale_Param )
```

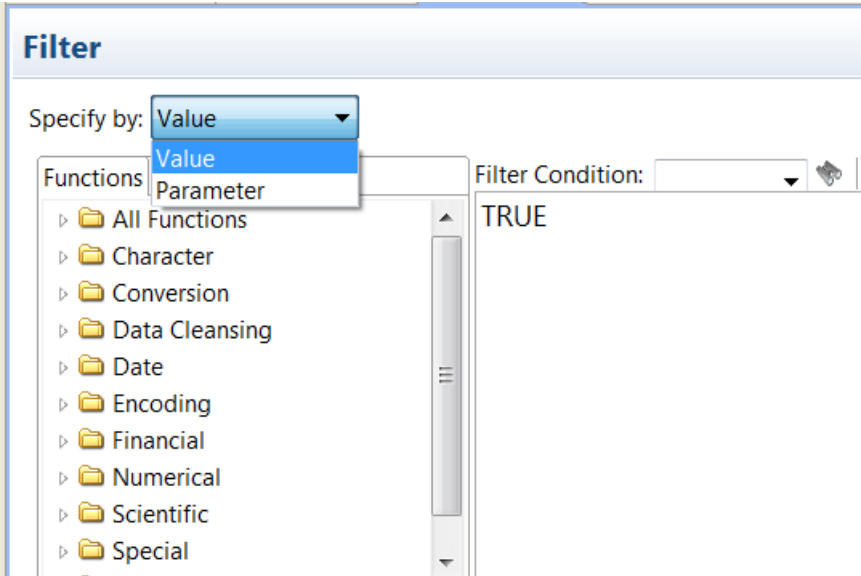
一个参数不能包含另一个参数。例如，如果在转换中配置了 Parameter1 和 Parameter2，则无法将 Parameter1 的默认值设置为 \$Parameter2。如果嵌套了参数，映射将在运行时失败并显示验证错误。

表达式参数

可以配置表达式参数类型。表达式参数是包含完整表达式的参数。可以在筛选器转换和查找转换中使用表达式参数。

在表达式编辑器中定义表达式参数。选择**指定依据: 参数**指示将整个表达式参数化。

下图显示了筛选条件的**指定依据: 参数**选项：



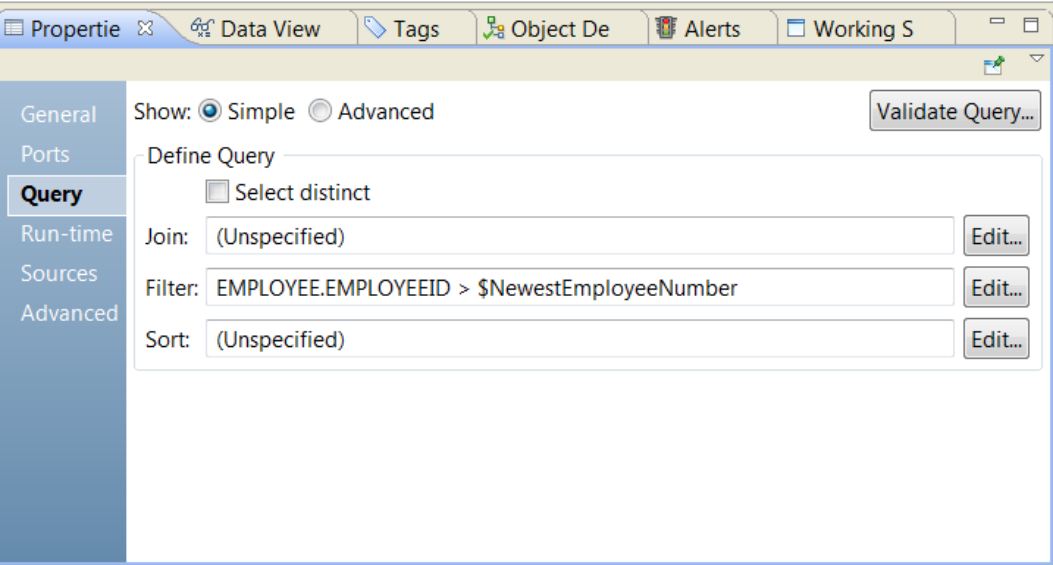
使用表达式参数时，您可以创建表达式参数，也可以选择要在转换中使用的现有表达式参数。表达式参数可以包含端口、运算符和常量，但不能包含其他参数。

例如，在筛选器转换中，您可以使用以下默认值创建一个筛选器表达式参数：EmployeeID > 100. 在映射中，您可以使用以下默认值创建一个不同的表达式参数：Dept < 2000. 如果将映射参数绑定到了转换参数，可以在运行时替代映射表达式参数。对于动态映射，可以使用不同的端口名称和运算符创建表达式参数。

SQL 语句中的参数

在添加到关系数据对象或查找转换的 SQL 语句中可以加入参数。

下图显示了如何参数化从关系源中读取的 SQL 查询：



数据集成服务在扩展查询时会每个带有单引号 (') 的参数中的数据进行转义。例如，您的 SQL 语句可能包含一个名为 \$date_parm 的日期参数：

select * from <table_name> where <date_port> >\$date_parm

该查询可以扩展为 select * from <table_name> where <date_port> > '01/31/2000 00:00:00'

请勿在参数的默认值中使用单引号。可能会产生意外结果。

参数名称不能包含句点 (.)。如果 SQL 查询中的参数包含句点，则该查询无效。例如，以下 SQL 语句中的参数名称包含句点：

SELECT \$tname.ID,"MY_SOURCE"."NAME" FROM "MY_SOURCE" where FIELDX=1

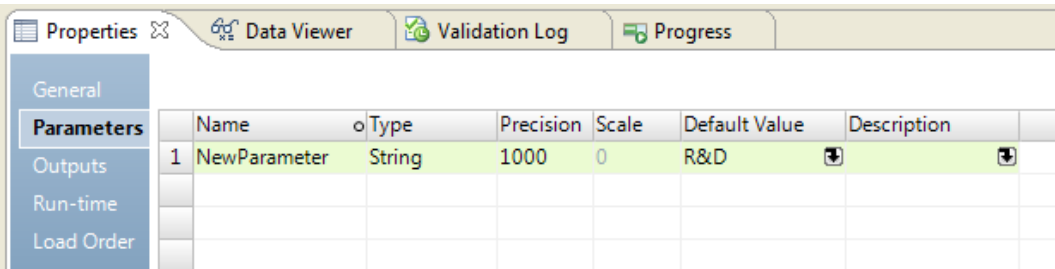
当您验证该查询时，数据集成服务会返回错误，指示找不到 tname.ID 参数。

Hive 源的自定义查询中的参数

在 Hive 源的 SQL 替代、联接表达式或筛选查询中使用字符串参数时，如果参数代表文字值，则需要在参数引用两边添加引号。可以使用单引号或双引号。此要求针对的是在本地执行环境或在 Hadoop 执行环境中运行的映射的 Hive 源。

例如，您需要创建筛选器来选择具有特定部门名称的 Hive 源行。创建代表部门名称的字符串参数。为部门名称参数分配了默认值 R&D。

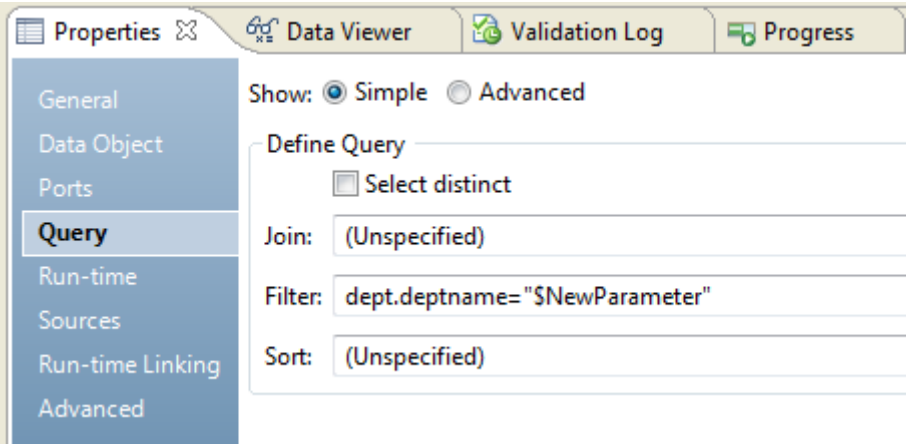
下图显示了字符串参数：



Parameters						
	Name	Type	Precision	Scale	Default Value	Description
1	NewParameter	String	1000	0	R&D	

在针对 Hive 源的筛选查询中使用该参数时，您必须在参数名称两边加上引号。否则，映射会在运行时失败，并出现 SQL 解析器错误。

下图显示了属性选项卡的查询视图针对 Hive 源的筛选查询：



Properties | Data Viewer | Validation Log | Progress

General | Parameters | Outputs | Run-time | Load Order

Show: ☒ Simple ☐ Advanced

Define Query

☐ Select distinct

Join: (Unspecified)

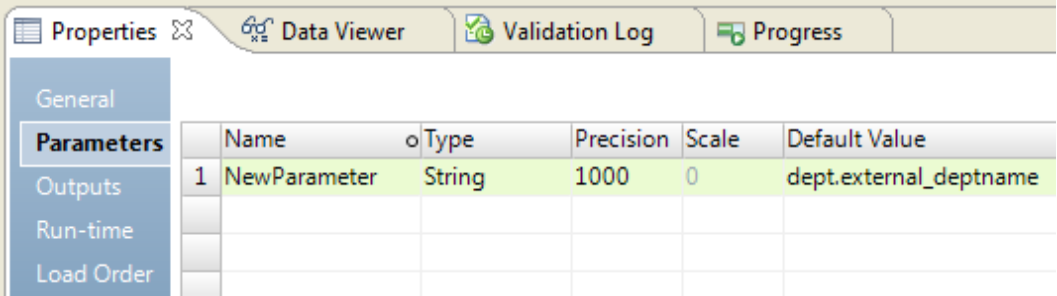
Filter: dept.deptname="\$NewParameter"

Sort: (Unspecified)

注意：默认情况下，表达式编辑器不会在参数两边添加引号。您必须手动添加引号。

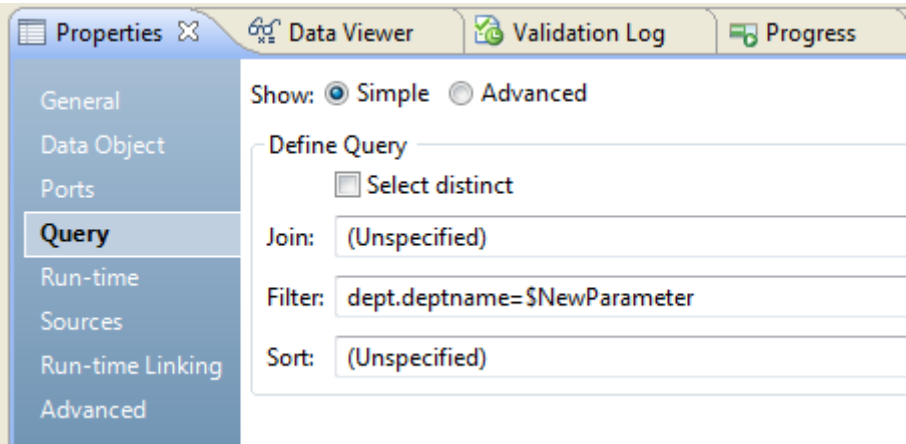
如果参数包含列名称或子查询名称，则不需要在参数名称两边添加单引号或双引号。

下图显示了默认值是列名称的字符串参数：



	Name	Type	Precision	Scale	Default Value
1	NewParameter	String	1000	0	dept.external_deptname

下图显示了使用该参数的筛选查询：



Properties Data Viewer Validation Log Progress

General
Parameters
Outputs
Run-time
Load Order

Show: ☒ Simple ☐ Advanced

Define Query

☐ Select distinct

Join: (Unspecified)

Filter: dept.deptname=\$NewParameter

Sort: (Unspecified)

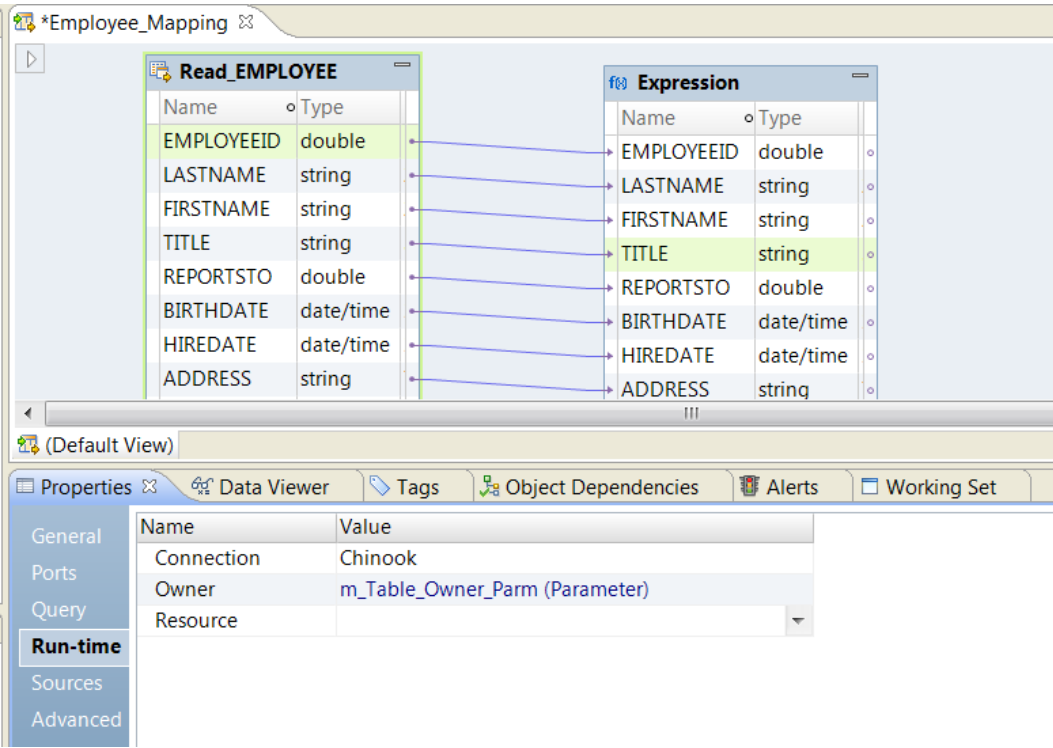
关系表资源的参数

可以参数化读取转换的资源名称、表所有者和连接。资源是指关系数据对象的表、视图或同义词名称。

如果需要处理动态映射中多个来自同一数据库的表，可以将资源名称参数化。

选择映射中的读取转换。在属性视图的运行时选项卡上，单击值列为连接、表所有者或资源分配参数。

下图显示了为读取转换的连接、资源名称和表所有者分配参数的位置。

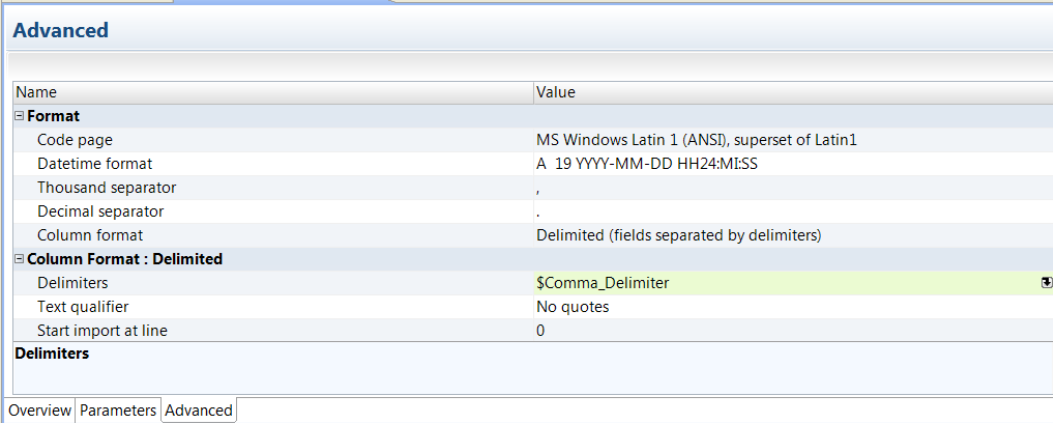


字段和属性值的参数

可以为转换和物理数据对象中的某些字段或属性值配置参数。

可以为关系数据对象、自定义数据对象和查找转换配置连接名称。在平面文件数据对象中，可以为输入和输出文件目录以及拒绝文件目录配置参数，还可以配置一个参数以更改平面文件分隔符类型。

下图显示了物理数据对象高级选项卡上的平面文件分隔符参数。



端口列表的参数

可以创建包含端口列表的参数。可以在排序器转换、等级转换、联接器转换和表达式转换等转换中引用这些参数。

可以配置以下类型的包含多个端口名称的参数：

端口列表

以逗号分隔的端口名称列表。端口列表参数使用以下语法：Port1,Port2,Port3

排序列表

每个端口的端口名称和排序类型列表。排序列表参数使用以下语法：Port1:A,Port2:A,Port3:D

输入链接集

要在运行时链接的一组端口。链接集参数包含采用以下语法的名称-值对：Port1>:=Port2, Port3>:=Port4

映射中的参数

如果在可重用转换或数据对象中定义了参数，则可以在将转换添加到映射时替代参数默认值。配置映射参数可替代特定映射中的参数值。

定义映射参数时，可以将该映射参数绑定到特定转换参数。映射参数值将替代转换中的默认参数值。

将映射参数绑定到转换参数时，两个参数的类型必须相同。映射参数名称不必与转换参数名称相同。

在运行时，可以使用参数集或参数文件设置映射参数值。不能使用参数集或参数文件设置转换参数值。如果您希望在运行时更改参数值，则必须配置映射参数。

请使用以下方法之一定义映射参数：

在映射属性视图的参数选项卡上定义映射参数。

在映射的**参数**选项卡上，可以手动输入每个参数名称、参数属性和默认值。可以在将转换添加到映射时将这些参数绑定到转换参数。可以在映射的**参数**选项卡上更新映射参数。

从转换参数中添加映射参数

将转换添加到映射后，可以直接从转换的**参数**选项卡创建映射参数。可以将转换参数公开为映射参数。Developer tool 会创建一个与转换参数具有相同属性的映射参数。

将参数添加到不可重用转换

如果在映射中创建了一个转换，则该转换为不可重用转换。如果参数化了任何转换属性，请创建映射参数而非转换参数。

参数实例值

将包含参数的可重用转换添加到映射时，可以在转换中配置每个参数的实例值。

实例值是特定映射的参数值。可以将实例值设置为默认值、特定值或映射参数值。

映射参数或 Mapplet 参数可以替代转换参数的默认值。选择映射参数或 Mapplet 参数，然后将该参数绑定到转换参数。

在转换**属性**视图上的**参数**选项卡上设置实例值。

为**实例值**选择下列选项之一：

公开为映射参数

创建与转换参数具有相同属性的映射参数，同时也在同一个步骤中将映射参数绑定到转换参数。如果第二次单击**公开为映射参数**按钮，且转换参数已经绑定到映射参数，Developer tool 则不会更改映射参数。

参数

浏览并选择要绑定到转换参数的映射参数。也可以创建一个映射参数并将其绑定到转换参数。创建映射参数并绑定该参数时，需执行的任务与**公开为映射参数**选项相同。但当手动创建映射参数时，可以配置一个不同于转换参数的名称。

使用默认值

使用转换参数中的默认值。跳过将映射参数绑定到转换参数。

值

输入要在映射中使用的默认参数值。跳过将映射参数绑定到转换参数。

Mapplet 中的参数

可以将 Mapplet 参数绑定到 Mapplet 中数据对象或转换的参数。

定义 Mapplet 参数时，可以将该 Mapplet 参数绑定到特定转换参数。Mapplet 参数值将替代转换中的默认参数值。将 Mapplet 参数绑定到转换参数时，两个参数的类型必须相同。Mapplet 参数名称不必与转换参数名称相同。可以将 Mapplet 参数绑定到多个转换参数。

请使用以下方法之一定义 Mapplet 参数：

在 Mapplet 属性视图的参数选项卡上定义 Mapplet 参数。

在 Mapplet 的“参数”选项卡上，可以手动输入每个参数名称、参数属性和默认值。

从转换参数中添加 Mapplet 参数

将转换添加到 Mapplet 后，可以直接从转换的**参数**选项卡创建 Mapplet 参数。

Mapplet 中的参数实例值

将包含转换参数的可重用转换添加到 Mapplet 时，可以设置每个参数的实例值。参数的实例值是指特定 Mapplet 中的参数值。

将转换添加到 Mapplet 后，在转换的**参数**选项卡上设置实例值。

为**实例值**选择下列选项之一：

公开为 Mapplet 参数

创建一个与转换参数具有相同属性的 Mapplet 参数。同时将映射参数绑定到转换参数。

参数

将 Mapplet 参数绑定到转换参数。可以浏览并选择要绑定到转换参数的 Mapplet 参数，也可以创建 Mapplet 参数并将该参数绑定到转换参数。创建 Mapplet 参数并绑定该参数时，需执行的任务与**公开为 Mapplet 参数**选项相同。但当手动创建 Mapplet 参数时，可以配置不同于转换参数的名称，并设置不同的默认值。

使用默认值

使用转换参数中的默认值。跳过将 Mapplet 参数绑定到转换参数。

值

输入要在 Mapplet 中使用的不同默认参数值。跳过将 Mapplet 参数绑定到转换参数。

映射中的 Mapplet 参数

将包含 Mapplet 参数的 Mapplet 添加到映射时，可以设置 Mapplet 参数的实例值。Mapplet 参数的实例值是指特定映射的参数值。

在 Mapplet 属性视图上的参数选项卡上设置实例值。

为实例值选择下列选项之一：

公开为映射参数

创建一个与 Mapplet 参数具有相同属性的映射参数。同时将映射参数绑定到 Mapplet 参数。

参数

将映射参数绑定到 Mapplet 参数。可以浏览并选择要绑定到 Mapplet 参数的映射参数，也可以创建映射参数并将其绑定到 Mapplet 参数。创建映射参数并绑定该参数时，需执行的任务与公开为映射参数选项相同。但当手动创建映射参数时，可以使用不同于 Mapplet 参数的名称和默认值配置该参数。

使用默认值

使用 Mapplet 参数中的默认值。跳过将映射参数绑定到 Mapplet 参数。

值

输入要在映射中使用的默认参数值。跳过将映射参数绑定到 Mapplet 参数。

Mapplet 示例中的参数

可以定义 Mapplet 参数并将其替代为映射参数。

可以定义一个从客户表中返回客户数据的 SQL 转换，将该 SQL 转换添加到 Mapplet 并将运行时连接参数化。

然后将 Mapplet 添加到从不同数据库中检索客户数据的映射。在每个映射中定义一个映射参数，以替代 Mapplet 参数中的默认连接。

下表列出了可以为 Mapplet 和映射创建的连接参数：

对象名称	对象类型	参数名称	参数默认值
mp_Get_Customer	Mapplet	mp_cust_connection	Oracle_Default
m_billing_mapping	映射	m_acctg_connection	Oracle_AcctDB
m_order_fill_mapping	映射	m_shipping_connection	Oracle_Warehouse
m_cust_main_mapping	映射	m_master_connection	Oracle_Cust_Mast

Mapplet (mp_Get_Customer) 具有一个名为 mp_cust_connection 的连接参数。该参数将 Oracle_Default 作为默认连接名称。例如，此连接可以引用测试数据库。

每个映射都具有一个连接参数，用于替代 mp_cust_connection 参数。每个映射均连接到会计、仓库或客户主数据库。

必须将每个映射参数绑定到 Mapplet 参数才能替代默认值。要在运行时更改映射参数值，可以配置参数集或参数文件。

逻辑数据对象中的参数

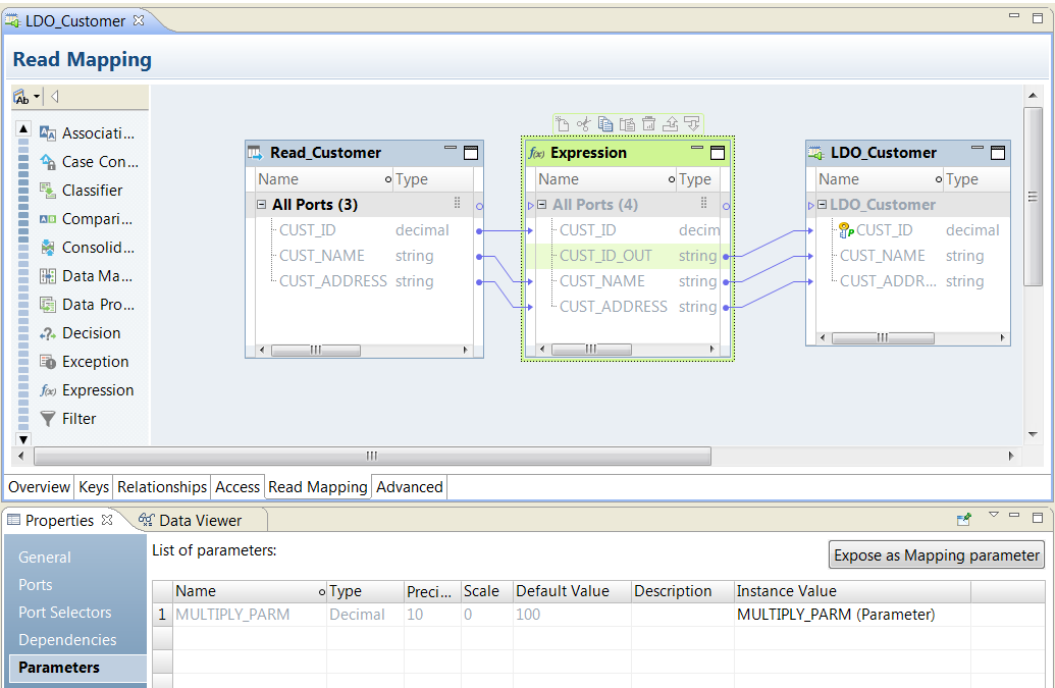
可以在逻辑数据对象中加入参数。可在转换以及读取和写入映射中使用参数。

逻辑数据对象可以包含读取映射和写入映射。读取或写入映射可以包含使用参数的转换。可以将可重用转换参数绑定到读取或写入映射中的参数。

例如，逻辑数据对象具有一个包含表达式转换的读取映射。表达式转换具有一个定义表达式中小数值的参数。默认值为 100。

将表达式转换添加到读取映射时，您可能希望使用不同的参数值。可以在读取映射级别创建用于替代转换参数的参数。单击**公开为映射参数**可在读取映射中创建重复参数。Developer tool 会将该重复参数绑定到转换参数。

下图显示了读取映射中表达式转换的**参数**选项卡：



要查看重复参数，请单击编辑器以显示映射的**属性**视图。单击**参数**选项卡。可以在读取映射级别更改参数默认值。

将逻辑数据对象添加到 Mapplet 或映射时，可以替代读取映射参数。在 Mapplet 或映射中创建重复参数。更改重复参数的默认值。

虚拟表映射中的参数

虚拟表映射定义了源与 SQL 数据服务中虚拟表之间的数据流。虚拟表映射可以包含参数，但不能使用参数文件或参数集替代参数默认值。

虚拟表映射可以包含具有参数的可重用转换或 Mapplet。可以将映射参数绑定到虚拟表映射中的转换或 Mapplet 参数。

但当虚拟表映射包含参数时，数据集成服务将应用映射级别的默认参数值。数据集成服务无法将参数文件或参数集中的值绑定到虚拟表映射中的参数。

可以使用已连接到虚拟表映射的参数化源。映射将使用默认参数值。

参数集

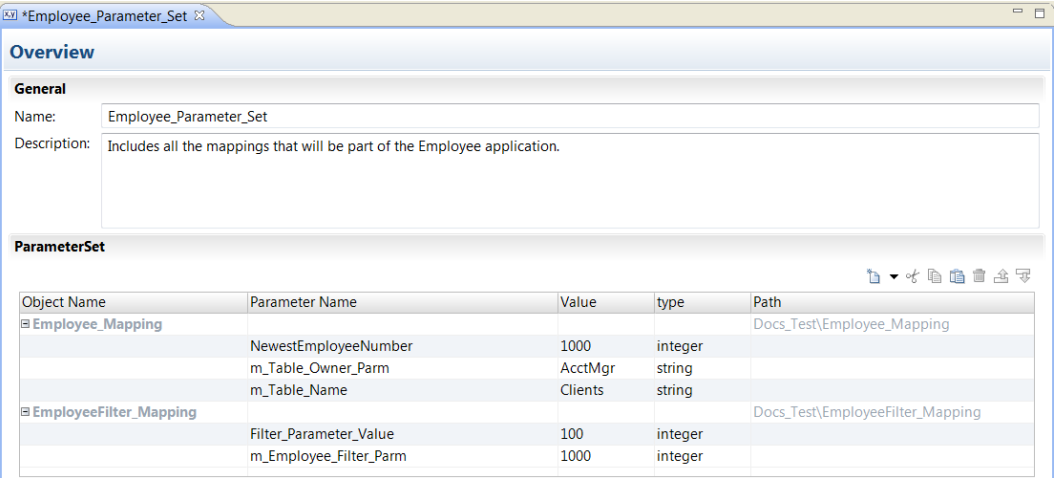
参数集是模型存储库中的对象，包含一组用于运行映射和工作流的参数和参数值。

创建参数集时，需选择使用参数的映射或工作流。选择映射或工作流之后，可以手动输入参数集中的参数，也可以选择已存在于映射或工作流存储库中的参数。

可针对不同的情况使用参数集。例如，在测试环境下运行工作流时，可以使用特定的参数集。

参数集适用于映射、映射任务或工作流。部署应用程序时，可向该应用程序添加一个或多个参数集。可将某个参数集添加到多个应用程序并部署这些应用程序。要将某个参数集用于工作流或映射，必须在部署该工作流或映射时将该参数集添加到应用程序。

下图显示了一个包含两个映射的参数的参数集：



Object Name	Parameter Name	Value	type	Path
Employee_Mapping	NewestEmployeeNumber	1000	integer	Docs_Test\Employee_Mapping
	m_Table_Owner_Parm	AcctMgr	string	
	m_Table_Name	Clients	string	
EmployeeFilter_Mapping	Filter_Parameter_Value	100	integer	Docs_Test\EmployeeFilter_Mapping
	m_Employee_Filter_Parm	1000	integer	

此参数集包含以下信息：

对象名称

包含参数定义的映射、Mapplet 或工作流的名称。

参数名称

映射、Mapplet 或工作流中参数的名称。

值

要在运行时使用的参数的值。参数集中的参数值将替代映射或工作流中的参数值。

类型

参数的类型。示例参数类型包括字符串、数字类型、连接、端口列表、排序列表和日期/时间参数。

从 Infacmd 中使用参数集运行映射

可以将映射部署为应用程序并在该应用程序中包含参数集，然后运行已部署的应用程序并使用参数集。

部署映射后，可以使用参数集从命令行运行已部署的映射。如果需要使用其他参数集，可以在应用程序中部署多个参数集。运行映射时，可以指定要使用的参数集。

部署应用程序后，可以使用 `infacmd addParameterSetEntries` 命令添加参数集条目，也可使用 `infacmd updateParameterSetEntries` 命令更新参数集条目。

有关参数集与 `infacmd` 配合使用的详细信息，请参阅《*Informatica 命令引用*》。

参数文件

参数文件是一个 XML 文件，该文件将列出用户定义参数及其分配值。每次从命令行运行映射时，都可以利用参数文件灵活地更改参数值。

参数值可定义工作流属性或者该工作流所运行的映射任务中包含的映射或 Mapplet 的属性。从命令行运行工作流并指定参数文件时，数据集成服务将应用这些值。

参数值可定义映射或 Mapplet 的属性。从命令行运行映射并指定参数文件时，数据集成服务将应用这些值。

可在参数文件中定义映射参数和工作流参数。无法在参数文件中定义系统参数值。

可以在一个参数文件中为多个映射定义参数。还可以创建多个参数文件，然后在每次运行映射时使用不同的文件。数据集成服务在映射运行的开始阶段读取参数文件以解析参数。

您可以从 Developer tool 导出参数文件。从映射或工作流 **参数** 选项卡导出文件。Developer tool 会生成包含映射或工作流参数以及默认参数值的参数文件。您可以指定参数文件名并选择将文件保存到哪个位置。

还可以使用 `infacmd ms ListMappingParams` 命令，列出在映射中使用的参数以及默认值。可以将该命令的输出用作参数文件模板。

使用 `ms RunMapping` 命令可对参数文件运行映射。

注意：映射和工作流的参数文件使用的结构相同。可以在一个参数文件中定义用于已部署映射和已部署工作流的参数。

参数文件结构

参数文件是一个 XML 文件，该文件包含至少一个参数及其分配值。

数据集成服务使用参数文件中定义的层次结构来标识参数及其定义值。层次结构可标识使用参数的工作流、映射或 Mapplet。

数据集成服务使用参数文件中定义的层次结构来标识参数及其定义值。层次结构可标识使用参数的映射、Mapplet 或工作流。

可以在项目或应用程序顶级元素内定义参数值。项目元素可定义您在任何已部署应用程序中运行项目内的特定映射时所使用的参数值。项目元素还可定义您运行使用项目中的对象的任何映射时所使用的参数值。应用程序元素可定义您在特定的已部署应用程序中运行特定映射时所使用的参数值。如果在同一个参数文件的项目顶级元素和应用程序顶级元素中定义相同的参数，则在应用程序元素中定义的参数值优先。

数据集成服务按以下顺序搜索参数值：

1. 在应用程序元素内指定的值。
2. 在项目元素内指定的值。
3. 参数默认值。

参数文件必须符合参数文件 XML 架构定义 (XSD) 的结构。如果参数文件不符合架构定义，数据集成服务将使映射运行失败。

在托管 Developer 工具的计算机上，参数文件 XML 架构定义显示在以下目录中：

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

在托管 Informatica 服务的计算机上，参数文件 XML 架构定义显示在以下目录中：

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

项目元素

项目元素可定义您在任何已部署应用程序中运行项目内的特定映射时所使用的参数值。项目元素还可定义您运行使用项目中的对象的任何映射时所使用的参数值。

项目元素可定义模型存储库中包含使用参数的对象的项目。项目元素可以包括工作流或映射。不能在项目中包括转换或数据源。

下表介绍了项目元素可以包含的元素：

元素名称	说明
文件夹	定义项目内的文件夹。如果对象整理到项目内的多个文件夹中，则使用文件夹元素。 文件夹元素可以包含数据源、映射、Maplet、转换或工作流元素。 文件夹元素可以包含数据源、映射、Maplet 或转换元素。
映射	定义项目内使用参数的映射。映射元素包含一个或多个参数元素，这些元素可为映射或映射中接受参数的任何不可重用数据对象、不可重用转换或可重用查找转换定义参数值。
工作流	定义项目内使用参数的工作流。工作流元素包含一个或多个可定义工作流参数值的参数元素。

使用在项目顶级元素中定义参数值的参数文件运行工作流时，数据集成服务会将参数值应用到指定的工作流。该服务还会将参数值应用到由工作流中的映射任务运行的映射中所包含的任何指定对象。

例如，您希望数据集成服务在您运行工作流“MyWorkflow”时应用参数值。该工作流包含运行映射“MyMapping”的映射任务。该映射包含数据对象“MyDataObject”和可重用转换“MyTransformation”。您希望在任何已部署应用程序中运行“MyWorkflow”时使用参数值。您还希望在运行包含映射任务的任何其他工作流（其使用“MyProject”项目中的这些对象运行映射）时使用参数值。在以下元素中定义参数：

```
<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>
</project>
```

使用在项目顶级元素中定义参数值的参数文件运行映射时，数据集成服务会将参数值应用到指定的映射。

例如，您希望数据集成服务在您运行映射“MyMapping”时应用参数值。

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

应用程序元素

应用程序元素提供项目元素的运行时范围。应用程序元素可定义您在特定的已部署应用程序中运行特定映射时使用的参数值。

应用程序元素可定义包含使用参数的对象的已部署应用程序。应用程序元素可以包含映射元素，该元素可在使用参数的已部署应用程序中定义映射。映射元素中包含项目元素。

例如，您希望数据集成服务当您在已部署应用程序“MyApp”中运行工作流“MyWorkflow”时应用参数值。您不希望任何其他应用程序中运行工作流时使用参数值或者在项目“MyProject”中运行其他工作流时使用参数值。在以下元素中定义参数：

```
<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>
    </project>
  </workflow>
</application>
```

例如，您希望数据集成服务当您在已部署应用程序“MyApp”中运行映射“MyMapping”时应用参数值。您不希望任何其他应用程序中运行映射时使用参数值或者在项目“MyProject”中运行其他映射时使用参数值。在以下元素中定义参数：

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

参数文件的规则和准则

创建参数文件时，应遵循特定规则和准则。

创建参数文件时请遵循以下规则：

- 可在参数文件中引用映射级别的参数。不能引用转换级别的参数。
- 参数值不能为空。例如，如果参数文件包含以下条目，数据集成服务将使映射运行失败：

```
<parameter name="Param1"> </parameter>
```

- 元素内的项目名称不区分大小写。因此，数据集成服务将<parameter name="SrcDir">和<parameter name="Srcdir">解释为同一个应用程序。
- 用于标识引用表的参数必须使用正斜杠 (/) 来分隔存储库文件夹路径中的文件夹名称。

参数文件示例

以下示例显示了用于运行映射的参数文件示例。

```
<?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"
    * 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>

      <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:
    * Mapplet "DS1"
-->
<project name="Project1">
  <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:

- * Maplet "MPLT1" in folder "Folder2"
- * Maplet "RULE1" in nested folder "Folder2_1_1"

```
-->
<project name="Project2">
  <folder name="Folder2">
    <maplet name="MPLT1">
      <parameter name="PROJ2_FOLD2_MPLT1">PROJ2_FOLD2_MPLT1_VAL</parameter>
    </maplet>
    <folder name="Folder2_1">
      <folder name="Folder2_1_1">
        <maplet name="RULE1">
          <parameter name="PROJ2_RULE1">PROJ2_RULE1_VAL</parameter>
        </maplet>
      </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:

- * Mapping "Map1"

```
-->
<application name="App2">
  <mapping name="Map1">
    <project name="Project1">
      <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 mapplet "DS1" in project "Project1."

    This section assigns values to parameters created in the following objects:

    * Mapplet "DS1"
-->
<project name="Project1">

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

```

导出参数文件

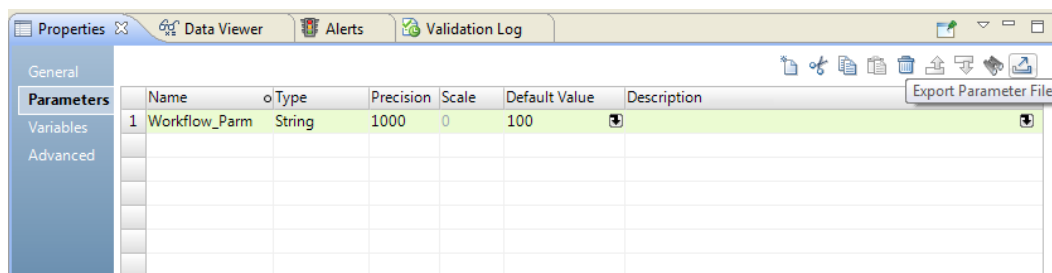
您可以从 Developer tool 导出映射参数文件或工作流参数文件。在 Developer tool 中定义参数，然后将其导出到文件。Developer tool 会创建 .XML 格式的参数文件。

您可以导出包含映射参数或工作流参数的参数文件。您可以从映射**参数**选项卡或工作流**参数**选项卡导出参数。Developer tool 会从**参数**选项卡导出所有参数。

要导出参数文件，请执行以下步骤：

1. 定义映射或工作流的参数和参数默认值。
2. 在映射的**参数**选项卡或工作流**属性**中，单击**导出参数文件**选项。
3. 输入参数文件的名称并浏览到文件放置位置。
4. 单击**保存**。

下图显示了工作流的“参数”选项卡上的导出参数文件选项：



导出参数文件时，Developer tool 会创建其中具有映射参数或工作流参数的参数文件。Developer tool 不会将映射和工作流参数导出到同一个文件。

例如，导出工作流参数 Workflow_Parm 时，Developer tool 将创建以下参数文件：

```
<?xml version="1.0" encoding="UTF-16LE"?>
<root version="2.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema" xmlns="http://www.informatica.com/Parameterization/1.0">
  <project name="Orders">
    <workflow name="Customer_Workflow">
      <parameter name="Workflow_Parm">100</parameter>
    </workflow>
  </project>
</root>
```

从 infacmd ms ListMappingParams 创建参数文件

Infacmd ms ListMappingParams 命令将列出已部署应用程序中映射的参数以及每个参数的默认值。使用此命令的输出创建参数文件。

1. 运行 infacmd ms ListMappingParams 命令以列出映射的参数以及每个参数的默认值。

参数 -o 可将命令输出发送到 XML 文件。

例如，以下命令在文件“MyOutputFile.xml”中列出工作流 MyWorkflow 中的参数：

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

例如，以下命令在文件“MyOutputFile.xml”中列出映射 MyMapping 中的参数：

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

数据集成服务将列出所有映射参数及其默认值。

2. 如果不指定参数 -o，可将命令输出复制到 XML 文件中并保存该文件。
3. 编辑 XML 文件并将参数默认值替换为您要在运行映射时使用的值。
4. 保存 XML 文件。

使用参数文件运行映射

使用 infacmd ms RunMapping 命令并通过参数文件运行映射。参数 -pf 可指定参数文件名称。

例如，以下命令使用参数文件“MyParamFile.xml”运行工作流 MyWorkflow：

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

例如，以下命令使用参数文件“MyParamFile.xml”运行映射 MyMapping：

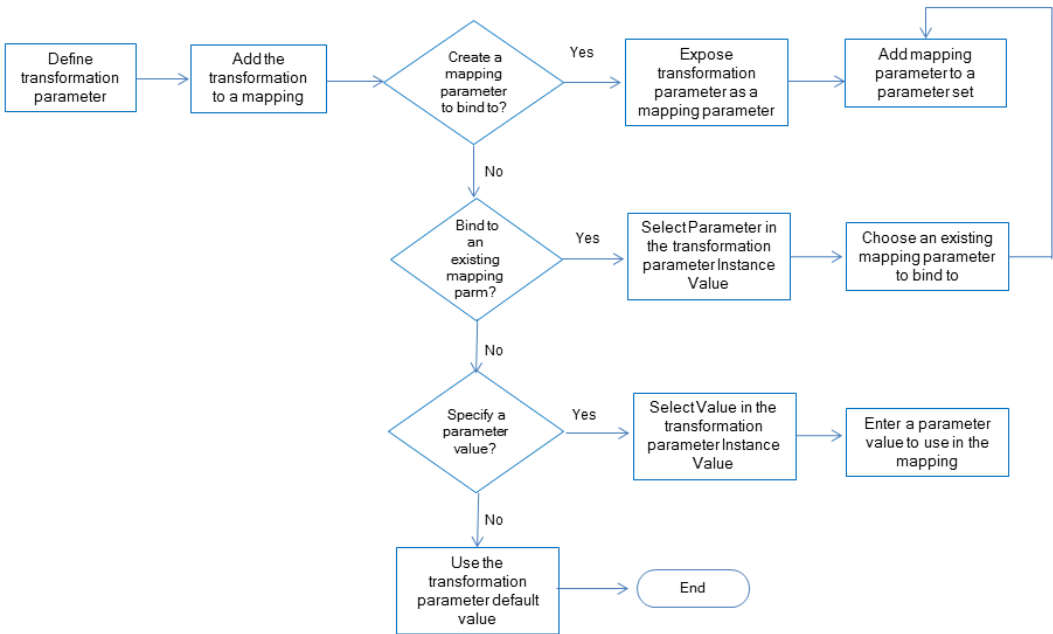
```
infacmd ms RunMapping -dn MyDomain -sn MyDataIntSvs -un MyUser -pd MyPassword -a MyApplication -m
MyMapping -pf MyParamFile.xml
```

如果使用参数文件运行映射，但参数文件无效，则数据集成服务会使该映射失败。如果数据集成服务找不到参数文件或者无法访问参数文件，则该服务会使映射失败。

如何配置参数

可在转换、映射、Mapplet 或工作流中定义参数。

下图显示了在映射的可重用转换中使用参数的过程：



1. 在可重用转换中，为转换中的属性或表达式编辑器中的变量创建参数。
2. 将转换添加到映射或 Mapplet。
3. 在转换的**参数**选项卡上，选择如何在映射或 Mapplet 中设置参数值。
 - 将转换参数公开为映射参数。在映射级别创建转换参数的重复项。
 - 将转换参数绑定到映射参数。浏览映射参数或手动创建映射参数以绑定到转换参数。
 - 输入特定参数值。输入要在映射运行中使用的默认值。
 - 使用转换参数的默认值。使用映射中的原始参数值。

将映射参数绑定到转换参数后，可创建参数集以在运行时替代映射参数值。从命令行运行映射并指定要在映射运行中使用的参数集。

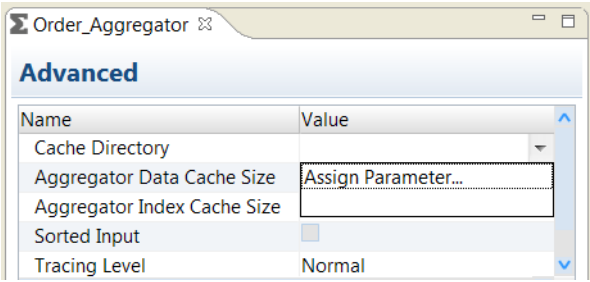
为转换属性创建参数

向字段或转换属性分配参数时，可以浏览要使用的参数，也可以专门为该字段创建一个参数。

1. 导航到要更新的字段或属性。
2. 单击**值**列中的选择箭头。

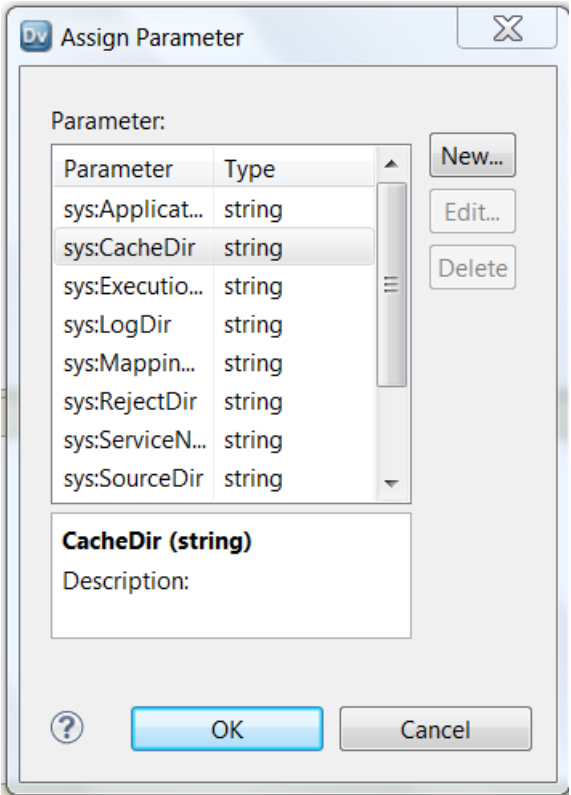
如果可以将属性参数化，则会显示**分配参数**选项。

下图显示了缓存目录的**分配参数**选项：



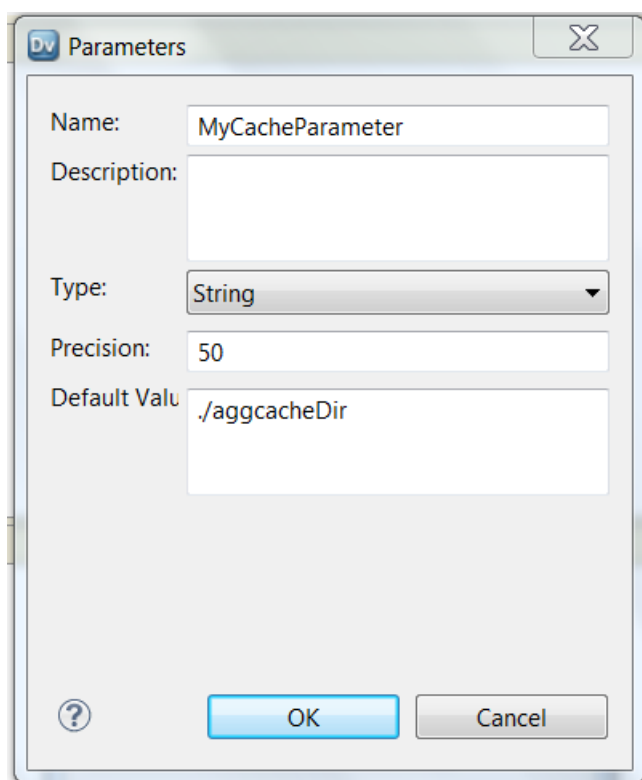
3. 单击**分配参数**。

此时将显示**分配参数**对话框。该对话框显示您在转换中创建的系统参数和用户定义参数。
下图显示了**分配参数**对话框：



- 4. 要创建参数，请单击**新建**。
- 5. 输入参数的名称、数据类型、数据精度和默认值。

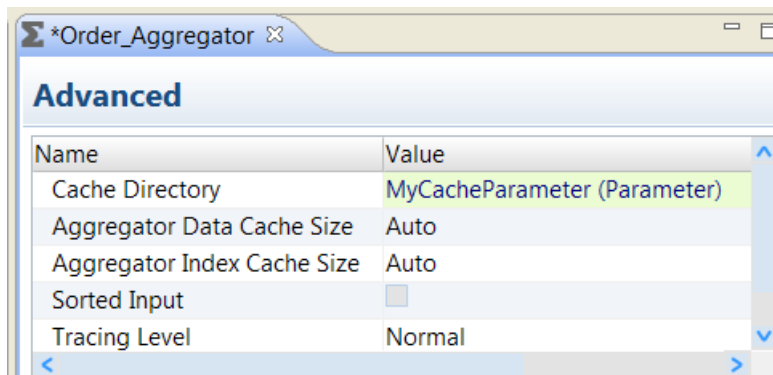
下图显示了名为 MyCacheParameter 的参数的**参数**对话框：



6. 单击**确定**。

参数名称将显示在转换属性中。

下图显示了汇总器转换缓存目录中的 MyCacheParameter：



可以在转换的**参数**选项卡上添加、更改和删除参数。

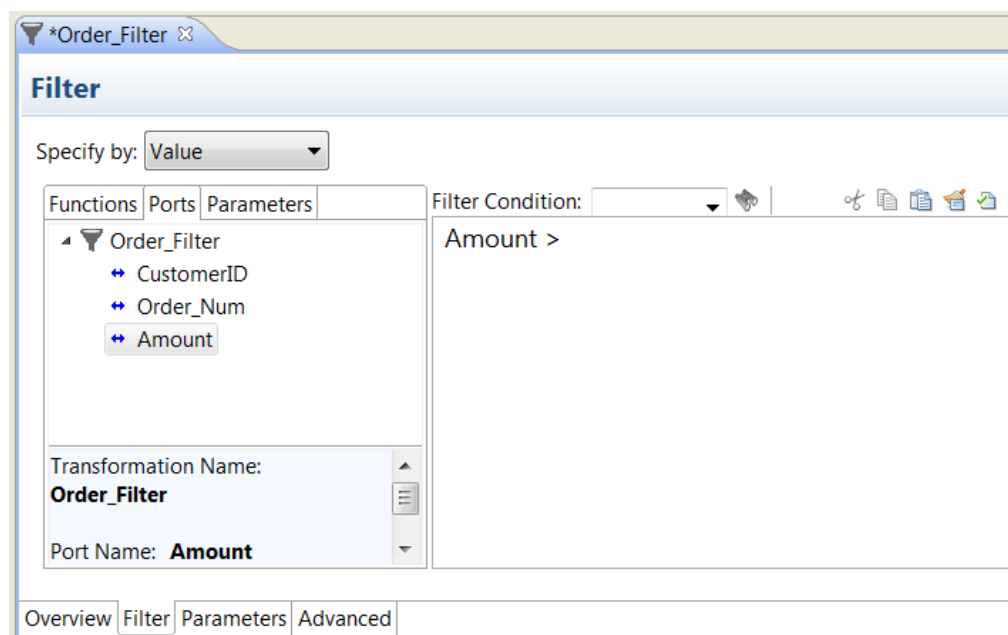
在表达式中创建参数

定义参数后，可以在表达式中引用参数。以下示例显示了如何将参数分配给筛选器表达式中的组件。

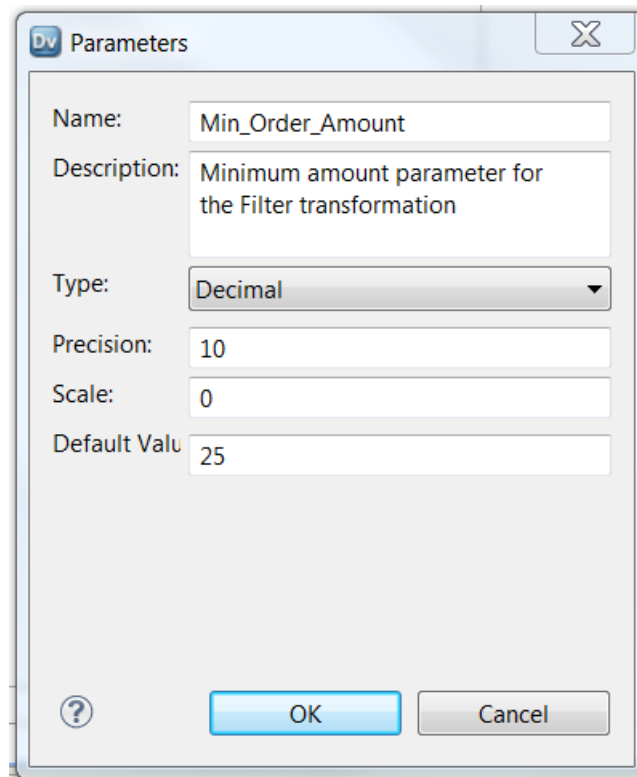
1. 在筛选器转换中，单击**筛选器**选项卡。
此时将显示表达式编辑器。可以选择函数、端口和参数以创建表达式。
2. 选择**指定依据：值**以定义表达式而不使用表达式参数。
3. 在“筛选器”选项卡上，单击**端口**选项卡。

4. 选择 Amount 端口。在函数选项卡上，选择大于 (>) 函数。

下图显示了包含 Amount 端口和 > 运算符的表达式：



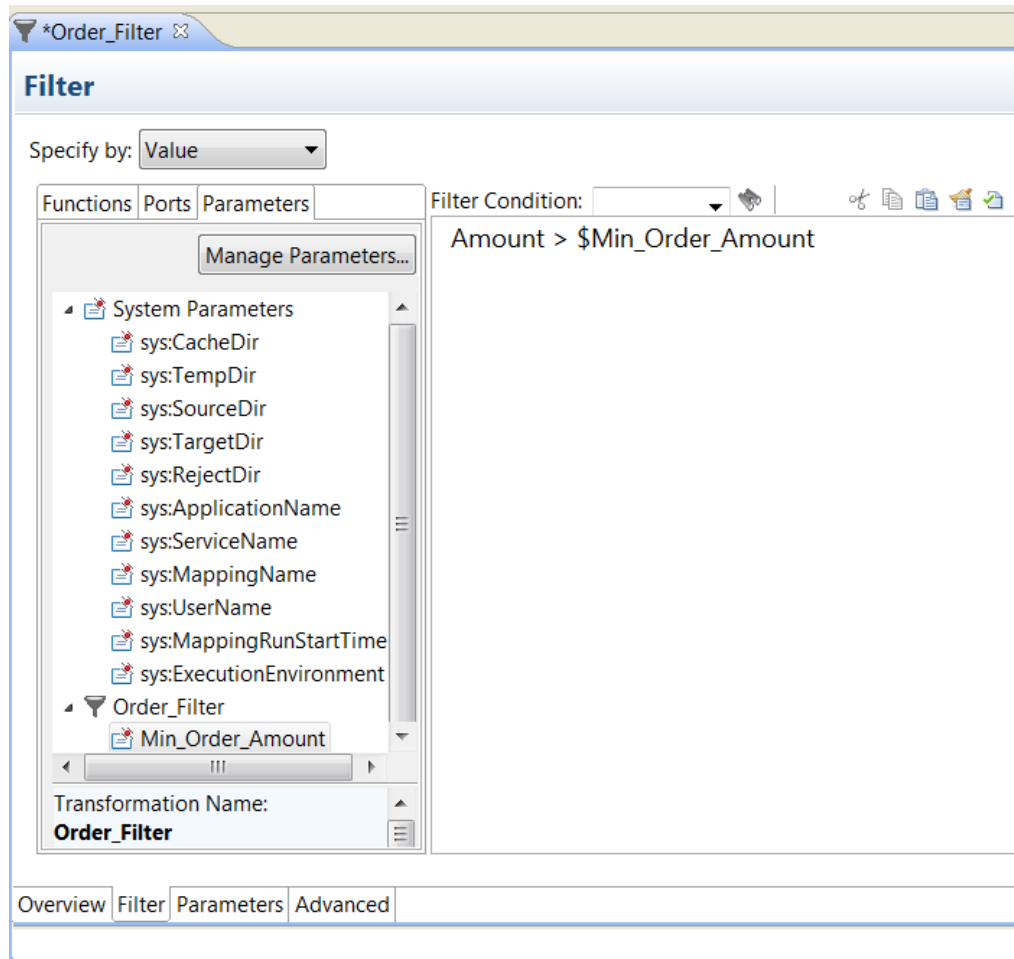
5. 单击表达式编辑器中的参数选项卡。
表达式编辑器列出了系统参数和用户定义参数。
6. 单击管理参数添加参数。
此时将显示参数对话框。
7. 单击新建。
此时将显示包含默认参数值的对话框。
8. 输入参数的名称、数据类型、数据精度和默认值。
下图显示了参数对话框：



9. 在表达式编辑器中，单击**确定**
您创建的参数将显示在参数列表中。

10. 选择 Min_Order_Amount 参数以将其添加到表达式中。

Min_Order_Amount 参数将显示在表达式中。



此参数将出现在表达式中，而且带有一个美元符号 (\$) 标识符。Min_Order_Amount 默认值为 50。如果将转换添加到了映射而且未替代 Min_Order_Parameter 的值，筛选器转换将返回 Amount 大于 50 的行。

将转换参数公开为映射参数

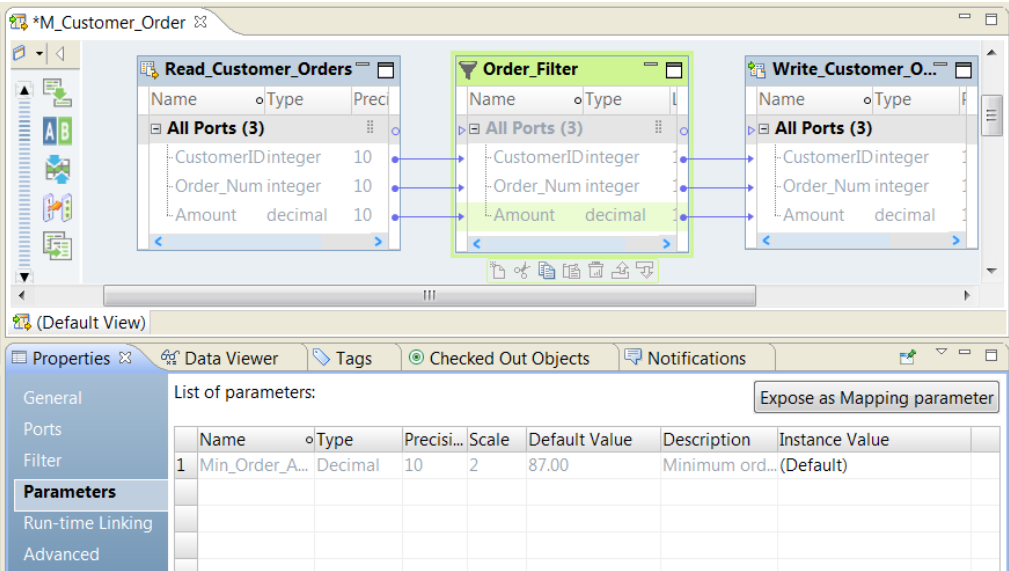
将转换添加到映射后，可以将转换参数公开为映射参数。将转换参数公开为映射参数后，就可以创建一个与转换参数重复的映射参数。

参数的实例值是指在特定映射或 Mapplet 中使用的参数值。如果将转换参数公开为了映射参数，则可以创建一个映射参数并同时将该参数绑定到转换参数。在运行时可以使用参数集或参数文件设置映射参数值。

注意: 您可以为转换参数单击一次**公开为映射参数**。如果单击**公开为映射参数**，且转换参数已经绑定到映射参数，Developer tool 则不会更改映射参数。Developer tool 不会创建其他映射参数，也不会更新映射参数默认值。创建映射参数后，多个对象可以使用映射参数。如果您需要更改映射参数默认值，请在映射中更改值或在运行时更改值。

1. 打开映射。单击包含参数的转换。
此时将显示转换的**属性**视图。
2. 单击**参数**选项卡。

下图显示了筛选器转换的参数选项卡：



3. 要为参数创建映射参数，请选择该参数并单击**公开为映射参数**。
Developer tool 会创建一个同名的映射参数并将该参数绑定到转换参数。
4. 要更新映射参数，请在映射的**属性**视图中打开**参数**选项卡。
可以更改默认映射参数值。还可以在映射**参数**选项卡上添加映射参数。

设置参数实例值

可以在转换的“参数”选项卡上设置**实例值**列中的参数实例值。如果您不希望创建重复的映射参数，可设置此列中的实例值。

可以将转换参数设置为默认值，也可以将现有映射参数绑定到转换参数。

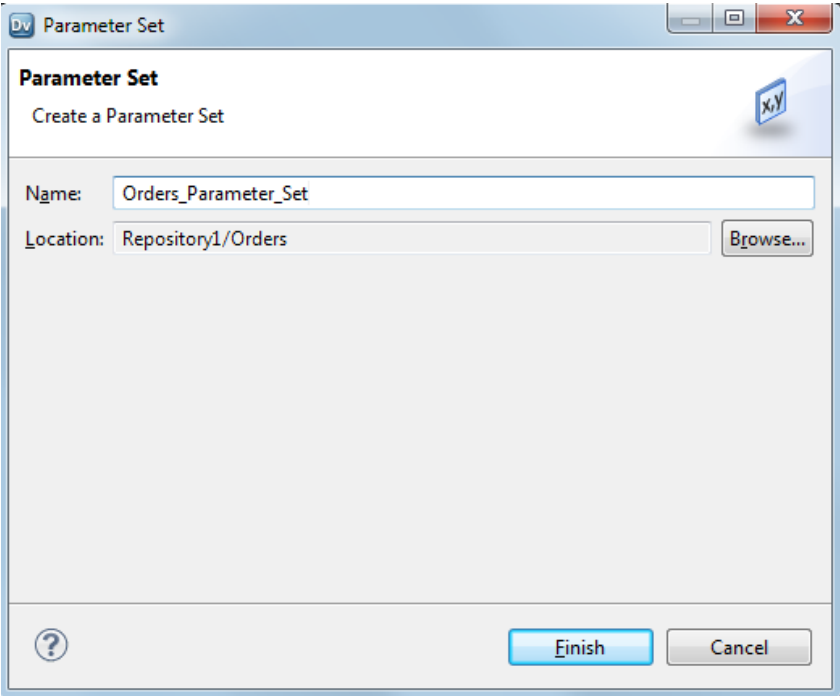
1. 将转换添加到映射后，单击转换**属性**视图上的**参数**选项卡。
2. 要将映射参数绑定到转换参数，请执行以下步骤：
 - a. 单击转换参数的**实例值**列。
此时将显示**指定依据**对话框。
 - b. 单击**指定依据: 参数**。
 - c. 在**分配参数**对话框上，浏览并选择要绑定到转换参数的映射参数或系统定义参数。
 - d. 单击**确定**。
映射参数名称将以参数值形式显示在**指定依据**对话框中。
 - e. 在**指定依据**对话框中单击**确定**。
映射参数名称将显示在**实例值**列中。
3. 要为转换参数实例设置默认值，请执行以下步骤：
 - a. 单击转换参数的**实例值**列。
此时将显示**指定依据**对话框。
 - b. 要输入默认值，请单击**指定依据: 值**并输入实例的默认值。
 - c. 要使用转换参数默认值，请单击**使用默认值**。

创建参数集

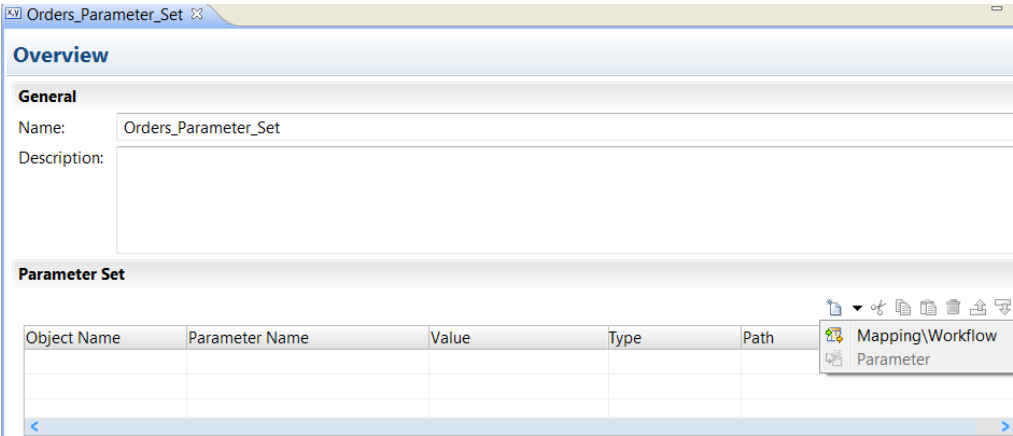
创建参数集，以便使用其来更改映射和工作流的运行时上下文。

创建参数集时，需选择用于包含参数的映射或工作流。选择映射或工作流后，可以手动在参数集中输入参数，也可以选择参数。

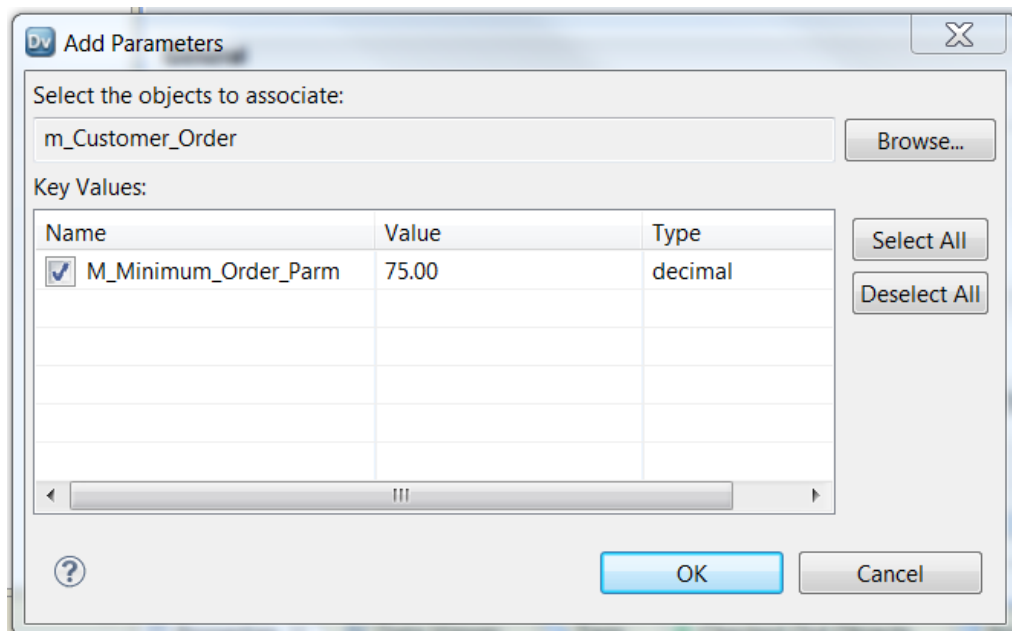
- 1. 在 Object Explorer 视图中，右键单击项目，然后单击新建 > 参数集。
- 2. 输入参数集名称，然后单击完成。



- 3. 向下拖动属性面板并查看网格，以向参数集添加参数。
- 4. 单击新建 > 映射/工作流。

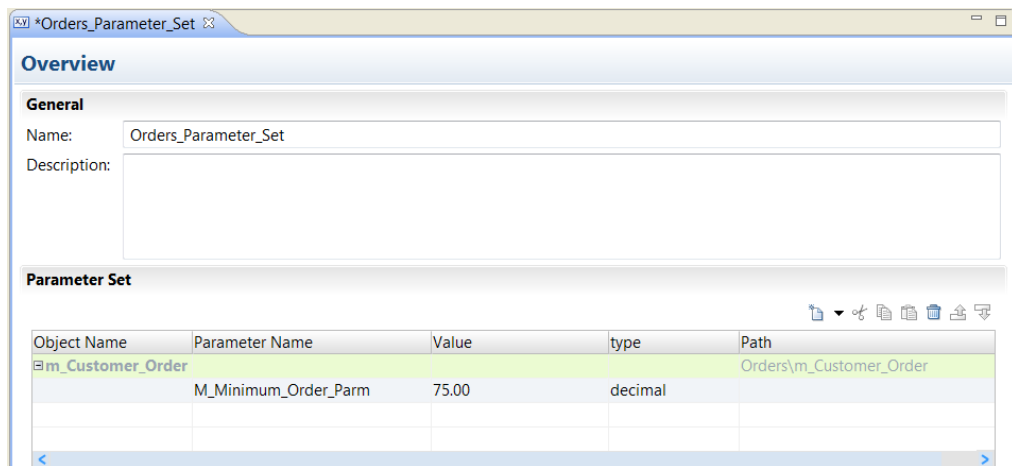


- 5. 在添加参数对话框中，单击浏览以查找需要加入参数集中的参数所属的映射或工作流。
此时会出现映射和工作流列表。
- 6. 选择映射或工作流，然后单击确定。
此时会出现映射或工作流中的参数列表。



7. 选择要包含在参数集中的参数，然后单击**确定**。

映射或工作流名称以及路径即会出现在参数集中。选择的每个参数将显示在对象下方。



8. 要添加尚不存在于工作流或映射中的参数，请右键单击映射或对象名称，然后选择**参数插入**。

Developer tool 会在工作流或映射下方创建参数。更改参数名称、值和类型。

注意：必须先将参数添加到映射或工作流中，然后才能使用参数集。

第 4 章

映射输出

本章包括以下主题：

- [映射输出概览, 69](#)
- [用户定义的映射输出, 70](#)
- [系统定义的映射输出, 72](#)
- [保留映射输出, 73](#)
- [将映射输出绑定到工作流变量, 75](#)
- [Mapplet 中的映射输出, 76](#)
- [逻辑数据对象中的映射输出, 79](#)
- [如何配置映射输出, 79](#)
- [如何将 Mapplet 输出绑定到映射输出, 88](#)

映射输出概览

映射可以返回映射输出。映射输出为单一值，它是映射所处理的每一行中的某字段或表达式的汇总结果。

映射输出的返回值提供与映射运行相关的信息。例如，映射输出可以返回映射已找到的错误行数，还可以返回映射已处理的最新订单日期和所有订单的总金额。

转换不会接收映射输出值。当映射完成后，映射会返回每个映射值。可以将映射输出传递到工作流中的其他任务，还可以保存这些值以便在下次运行映射时将其用作输入参数。您可以在同一个映射中定义多个映射输出。

映射可以返回用户定义的映射输出或系统定义的映射输出。

用户定义的映射输出

用户定义的映射输出是映射对其中每一行的某字段或表达式进行汇总后返回的数值或日期。例如，您可能需要了解订单何时达到特定阈值。这种情况下，您可以配置映射返回映射已处理的订单总额。首先定义一个名为 TotalOrderAmt 的映射输出，并配置映射汇总每一行中 Order_Amount 字段的值。然后，在表达式转换中定义一个表达式或端口名称以进行汇总。

系统定义的映射输出

系统定义的映射输出是映射在完成时始终返回的内置值。映射将返回映射处理的源行数、目标行数和错误行数。您可以将工作流变量中的这些值传递到工作流中的另一个任务，如通知任务或独占网关任务。无需定义系统定义的映射输出。

使用映射输出执行以下任务：

将映射输出保存在存储库中

可以配置一个映射任务，将映射输出值保留在存储库中。还可以将保留的映射输出值分配给映射任务输入。例如，可以配置映射以返回生成的最新序列号。将 Last_Seq_Num 映射输出保留在存储库中。当下一次运行映射时，就可以使用 Last_Seq_Num 作为起始序列号。

将输出绑定到工作流变量

可以将映射输出绑定到工作流变量，然后将其值传递给工作流中的其他任务。可以将当前映射任务运行的映射输出绑定到工作流变量。还可以将上次映射任务运行中保留的映射输出绑定到当前运行中的工作流变量。

用户定义的映射输出

用户定义的映射输出是映射对其中每一行的某一字段或表达式进行汇总后返回的数值或日期。定义要汇总的表达式和结果值的数据类型。

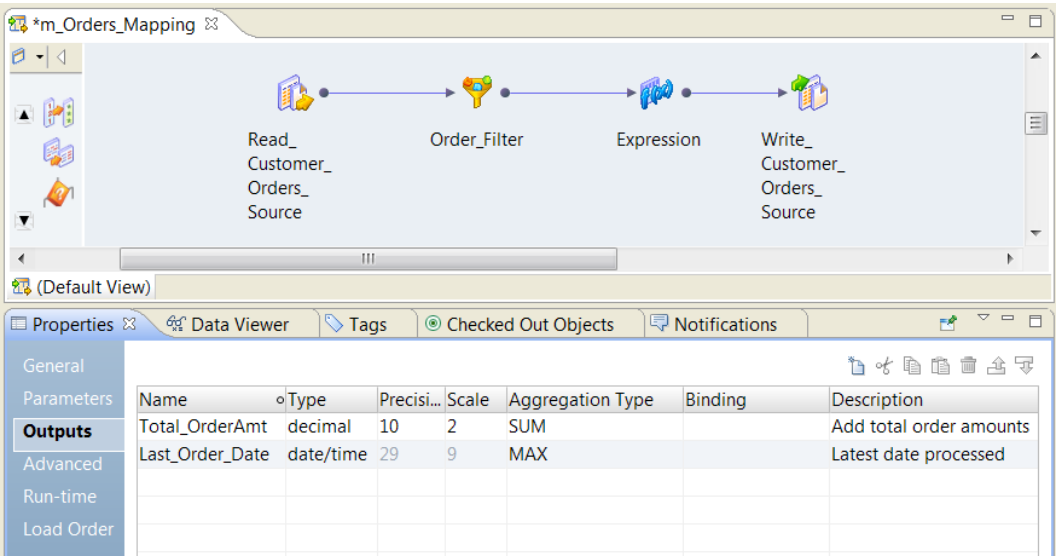
在“属性”视图的**输出**选项卡中定义映射输出。配置映射输出的名称和结果的数据类型，并指示要执行哪种类型的汇总以返回结果。

定义映射输出名称和输出类型后，在映射中配置表达式转换。在表达式转换中，定义要汇总的输出表达式。表达式可以包含端口名称，也可以包含具有端口、函数和参数的表达式。

输出视图

在映射**属性**视图的**输出**选项卡中定义映射输出。定义每个映射输出时，输入映射输出名称、映射输出类型和要执行的汇总类型。

下图显示了映射**属性**视图的**输出**选项卡中的映射输出：



输出选项卡包含以下字段：

名称

输出的名称。默认为 Output。

类型

映射输出的类型。可以选择数值类型或日期/时间类型。默认值为整数。

精度

映射输出字段的长度。默认值为 10。

小数位数

映射输出字段中小数点右侧的位数。默认值为零。

汇总类型

可以选择以下汇总类型之一：

SUM

返回数据集成服务处理的每个输入行中字段或表达式的总和。

MIN

返回数据集成服务处理的每个输入行中特定字段或表达式的最小数值或日期。

MAX

返回数据集成服务处理的每个输入行中特定字段或表达式的最大数值或日期。

绑定

要绑定到映射输出的 Maplet 或逻辑数据对象输出的名称。除非返回的映射输出来自 Maplet 而非映射中的表达式转换，否则此字段为空。

说明

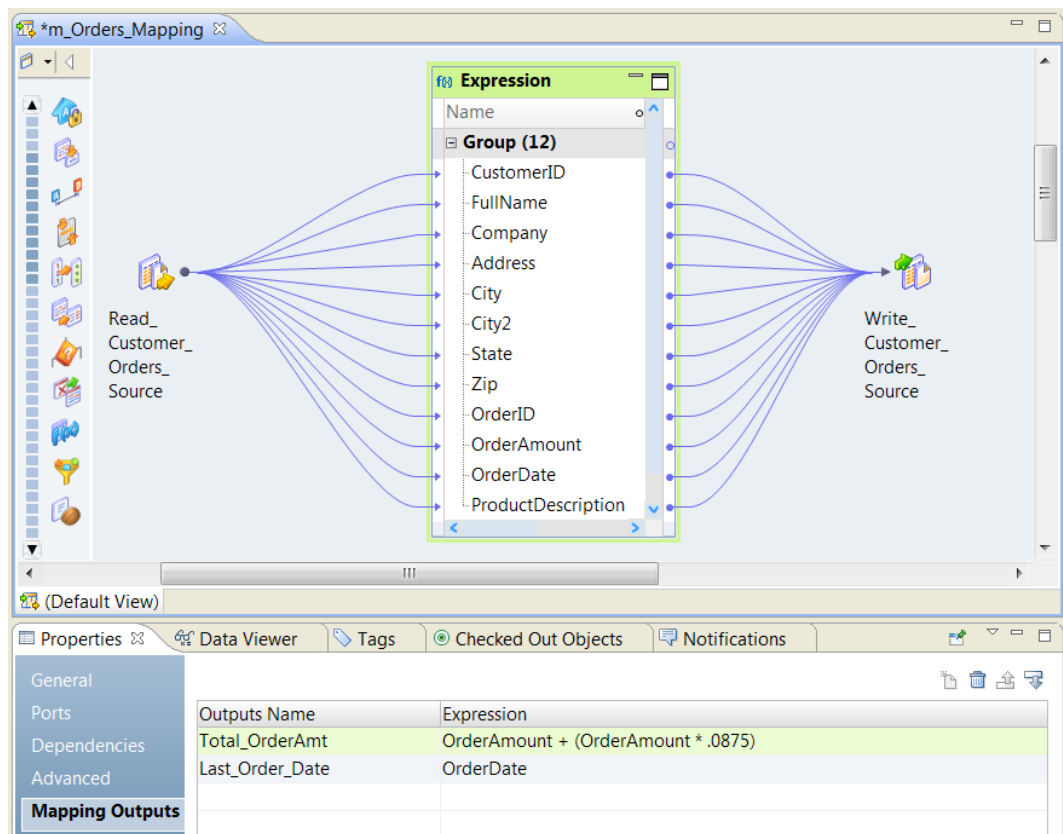
映射输出的说明。

映射输出表达式

可以在表达式转换的**映射输出**视图中配置映射输出表达式。映射输出表达式是对表达式转换所接收的行进行汇总的字段或表达式。

在映射中配置表达式转换并包含要汇总的输出表达式。根据映射是包含筛选器转换还是活动转换，表达式转换在管道中的位置可能会影响映射输出结果。如果需要汇总不同管道中的行，可以向映射添加多个表达式转换。

下图显示了表达式转换**映射输出**视图中的表达式。



映射输出视图包含以下字段：

输出名称

您在映射级别创建的映射输出的名称。必须先在映射级别创建映射输出。当在表达式转换中添加映射输出时，需在已创建的输出列表中选择输出名称。

表达式

要针对映射中的每一行进行汇总的表达式。可以在表达式编辑器中输入端口名称或表达式。表达式结果必须为数字或日期。可以在表达式中使用参数。数据集成服务将对表达式转换接收的每一行应用该表达式。当映射完成后，每个映射输出都将返回一个值。

注意：不要在表达式转换中指定要执行的汇总类型，请指明映射在处理每一行时汇总的字段或表达式。

系统定义的映射输出

系统定义的映射输出是每个映射都会生成的映射输出。无需为系统定义的映射输出配置汇总。可以将系统定义的映射输出传递给工作流变量。

映射将返回以下类型的系统定义映射输出：

numberOfTargetRows

映射已写入到目标的行数。

numberOfSourceRows

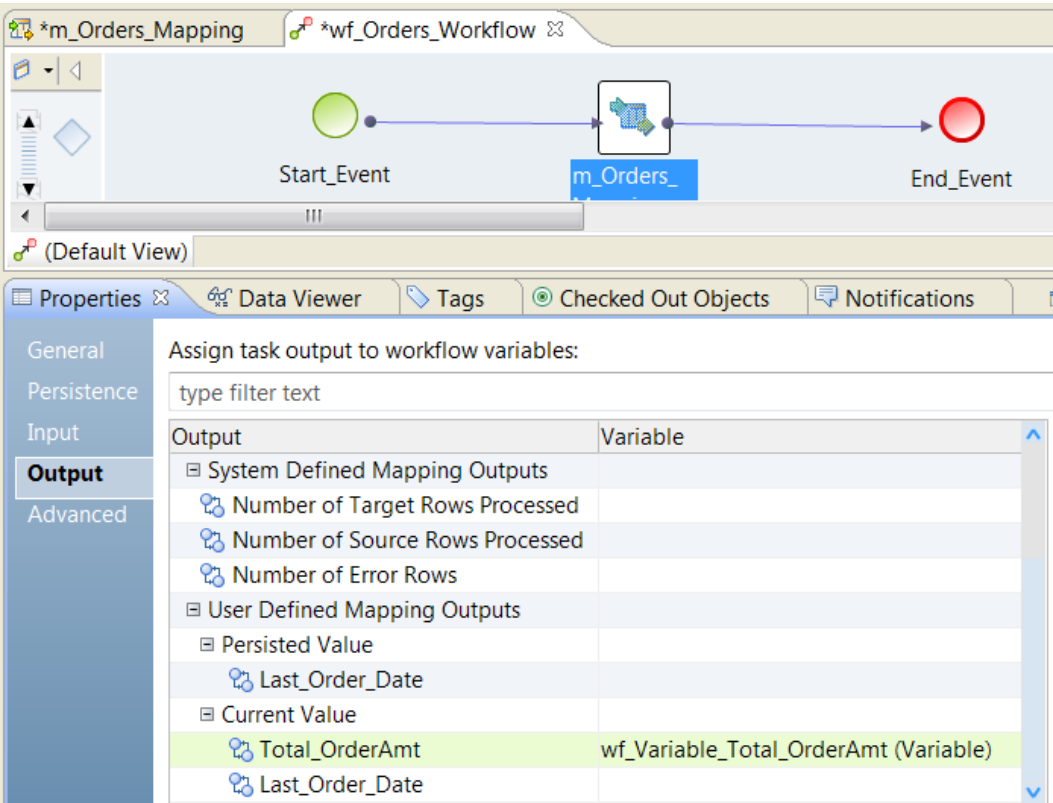
映射从源中读取的行数。

numberOfErrorRows

映射生成的错误行数。

在映射任务**属性**视图的**输出**选项卡上，配置要将系统定义的映射输出分配到的工作流变量。

下图显示了**输出**选项卡上的系统定义映射输出：



保留映射输出

如果运行工作流中的映射，可以将映射输出保存在存储库中。这样，就可以在同一映射任务的后续运行中使用前面的某个映射输出，也可以将上次映射任务运行中保留的映射输出分配给当前映射任务运行中的工作流变量。

将映射输出保留在映射任务中。映射任务是指工作流中具有映射配置和参数绑定的映射实例。有关映射任务的详细信息，请参阅《*Informatica Developer 工作流指南*》。

在映射任务的**属性**视图中单击**持久性**选项卡后，Developer tool 会显示映射的所有映射输出。要保留任何映射输出，请为该映射输出启用**保留**，然后选择要执行的汇总类型以返回保留值。

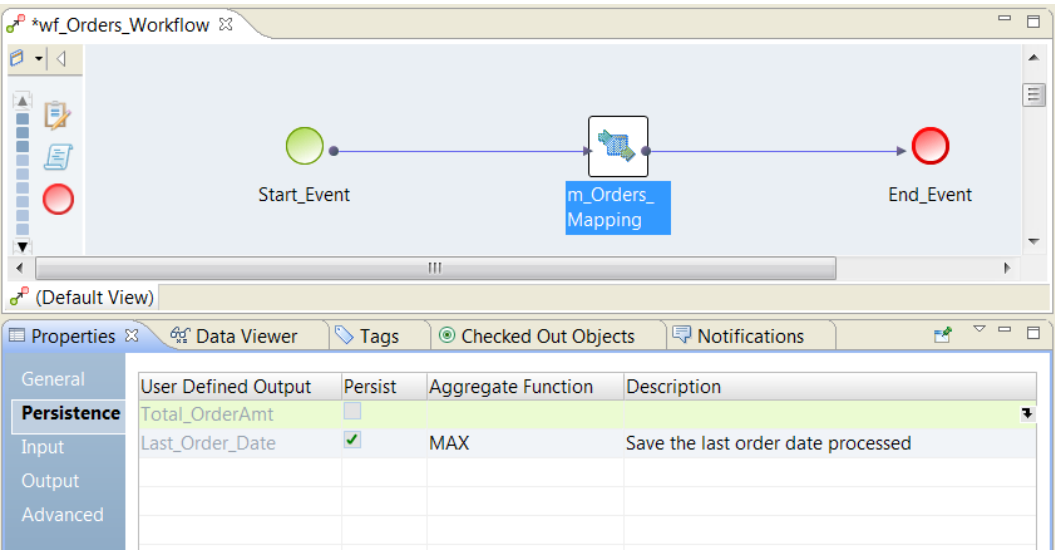
当数据集成服务将映射输出保留在模型存储库时，数据集成服务将具有映射任务名称的映射输出保存为键。例如，如果工作流包含 4 个映射任务，各任务运行同一映射，则数据集成服务将这 4 个输出保存在模型存储库中。

保留某个映射输出时，可以为保留值配置与在映射级别定义的汇总函数不同的汇总函数。数据集成服务会生成多个映射输出值。例如，OrderDate 映射输出可以包含 MIN OrderDate。保留的 OrderDate 映射输出可以包含 MAX OrderDate。

可以将映射任务的映射输出绑定到映射任务下一次运行时使用的输入参数。当您一次映射运行的结果配置为同一个映射在下一次运行时的输入时，需要进行反馈绑定。必须将映射输出保留在映射任务中，才能将其用于反馈绑定。

反馈绑定的一个示例是保留映射处理的最新订单日期。下次运行该映射任务时，映射的输入参数将为已处理的最近日期。映射可以筛选参数源行，以包含那些订单日期晚于已处理的最近订单日期的行。

下图显示了映射任务属性视图上的持久性选项卡。



持久性选项卡包含以下字段：

用户定义的输出

映射返回的映射输出的名称。

保留

使数据集成服务能够将映射输出保留在存储库中。

汇总函数

针对要保留的映射输出执行的汇总类型。选择 MIN、MAX 或 SUM。默认值是在映射属性中定义的映射输出值。可以更改已保留映射输出的汇总函数类型。保留在存储库中的值可以不同于传递给工作流变量的映射输出值。

说明

关于要保留在存储库中的映射输出的说明。

保留值维护

可以列出、更新和重置保留在存储库中的映射输出。

可以针对保留的映射任务值运行以下 infacmd 命令：

listMappingPersistedOutputs

列出为工作流中的某个映射任务实例保留的映射输出及其值。

setMappingPersistedOutputs

更新或重置为工作流中的特定映射任务实例保留的映射输出。重置值时，将从存储库中删除保留值。要设置映射输出，请在命令行中输入以空格分隔的映射输出名称-值对。要重置映射输出，请结合使用 -reset 选项和以空格分隔的映射输出列表。

有关 infacmd 的详细信息，请参阅《Informatica 命令引用》。

保留映射输出和部署

重新部署工作流或更改映射输出时，可能会影响已保留映射输出的状态。

对于保留的映射输出，请考虑以下规则和准则：

- 第一次将工作流作为应用程序进行部署时，如果映射任务具有保留的映射输出，则无需执行任何附加任务。
- 重新部署应用程序时，可以选择是保留状态信息还是丢弃该信息。如果选择了保留状态信息，则在重新部署应用程序时，存储库中的映射输出值不会更改。否则，将从持久性中删除映射输出状态。
- 当备份和还原映射或工作流时，不会备份映射输出状态。
- 如果重命名或重新创建了映射输出，则不能使用在上次工作流运行中保留的映射输出值。

将映射输出绑定到工作流变量

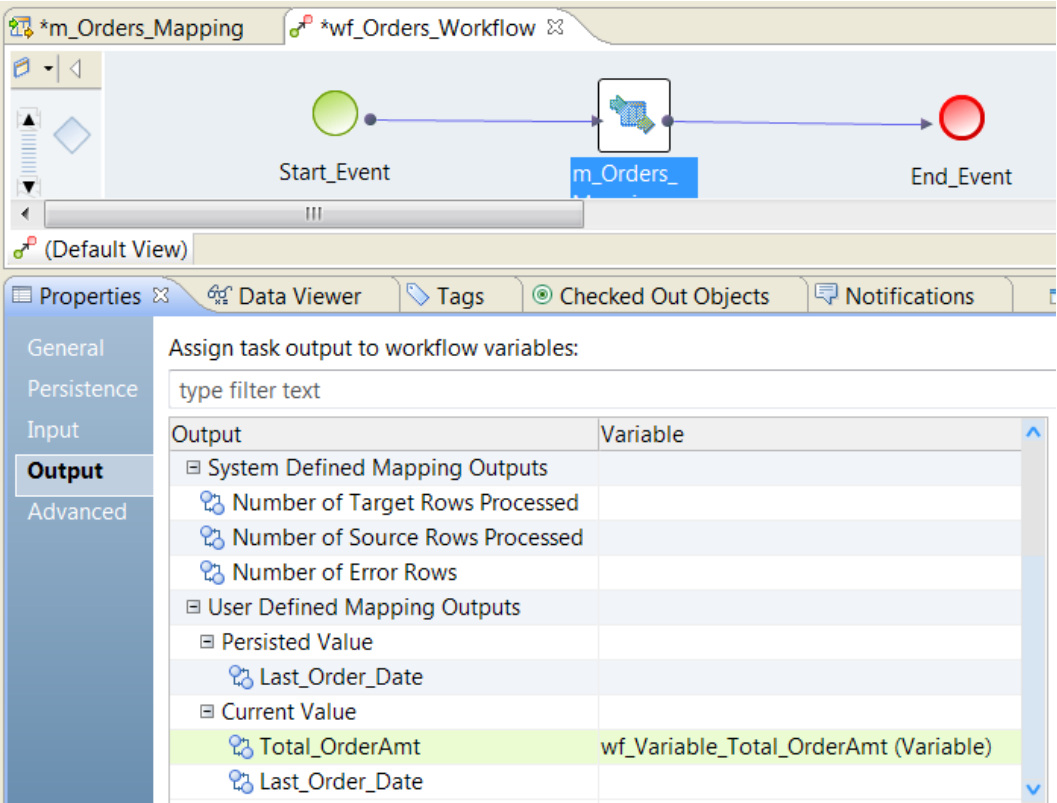
将映射添加到工作流后，可以将映射输出绑定到工作流变量。可以将值传递到工作流中的其他任务。

例如，您可能希望数据集成服务先评估映射输出值，然后再确定接下来运行哪个对象。或者，您可能希望数据集成服务在下一个任务的某个字段中使用映射输出值。

要在另一个任务中使用映射输出，请将映射输出绑定到映射任务**输出**视图中的工作流变量。

注意: 如果为工作流变量分配映射输出，并且该映射不对行进行处理，则输出为 NULL。映射任务不会更改工作流变量值。变量值与运行映射任务前的值相同。

下图显示了映射任务的**输出**视图。



输出列包含以下类型的映射输出：

系统定义的映射输出

转换返回到映射的内置映射输出。系统定义的映射输出包含映射处理的源行数、目标行数和错误行数。

用户定义的映射输出

可以将保留的映射输出值和当前映射输出值绑定到工作流变量。

保留值

上次工作流运行中的用户定义映射输出值。保留值是指在映射任务上次运行后位于存储库中的值。保留值不是当前映射汇总的值。

当前值

当前映射任务中的用户定义映射输出值。

有关工作流变量的详细信息，请参阅《*Informatica Developer 工作流指南*》。

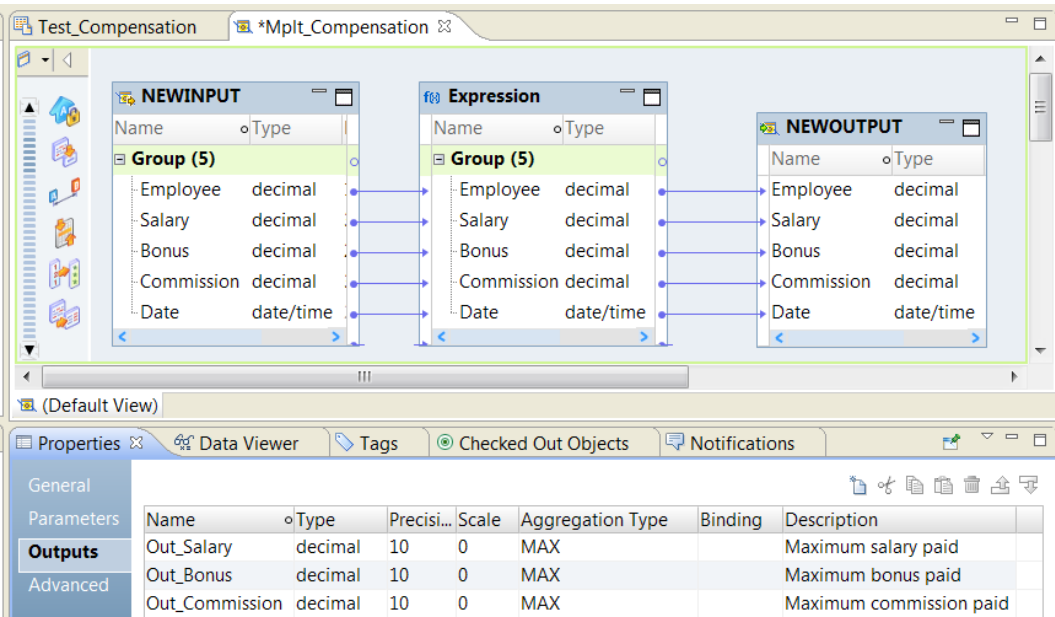
Mapplet 中的映射输出

可以将 Mapplet 配置为返回映射输出。可以将 Mapplet 中的映射输出绑定到映射级别的映射输出。

将某个 mapplet 加入映射中后，该 mapplet 会计算输出值并将输出值传递到映射。可以将 Mapplet 中的多个输出绑定到映射级别的同一个输出。还可以将 Mapplet 中的系统定义输出绑定到映射输出。Mapplet 输出和映射输出必须属于相同类型。

例如，Mapplet 可以返回三个映射输出中“薪资”端口、“奖金”端口和“佣金”端口的最大值。

下图显示了输出视图中的 Out_Salary、Out_Bonus 和 Out_Commission 映射端口。



输出选项卡包含以下字段：

名称

输出的名称。默认为 Output。

类型

映射输出的类型。可以选择数值类型或日期/时间类型。默认值为整数。

精度

映射输出字段的长度。

小数位数

映射输出字段中小数点右侧的位数。

汇总类型

要针对输出表达式执行的汇总类型。选择 SUM、MIN 或 MAX。默认值为 SUM。

绑定

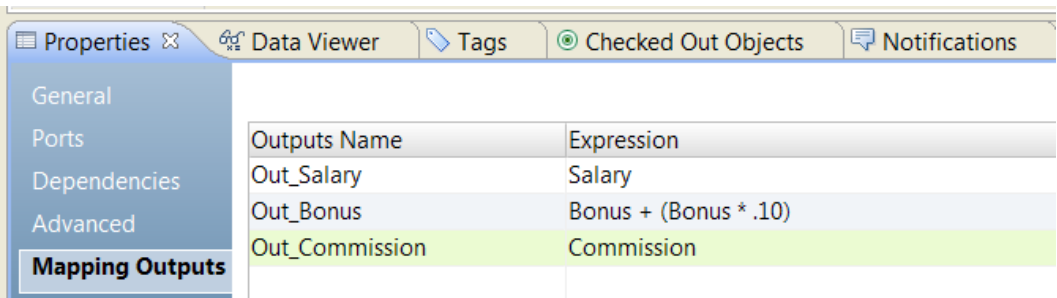
要绑定到此映射输出的另一个 Mapplet 中的输出名称。除非 Mapplet 包含另一个将返回映射输出的 Mapplet，否则绑定字段将为空。

说明

映射输出的说明。

针对 Mapplet 中的每个映射输出，在表达式转换中创建一个关联的输出表达式。每个表达式都标识要汇总的字段。

下图显示了表达式转换中的映射输出表达式：



Outputs Name	Expression
Out_Salary	Salary
Out_Bonus	Bonus + (Bonus * .10)
Out_Commission	Commission

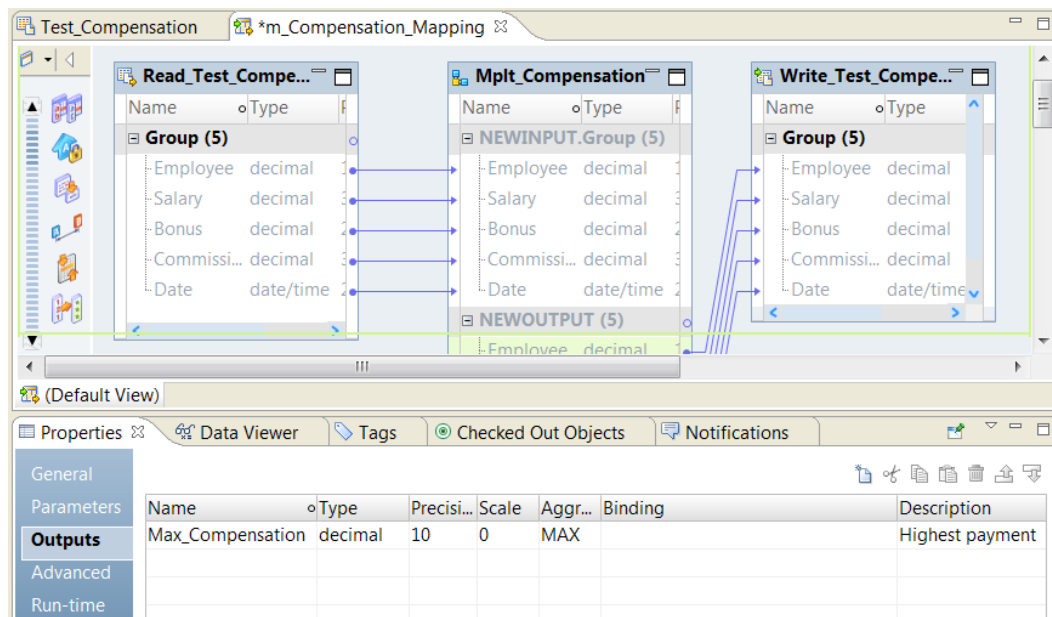
在本例中，表达式转换汇总“薪资”和“佣金”端口值。Out_Bonus 映射输出是一个表达式，该表达式为“奖金”端口值加上 10% 的奖金。

将 Mapplet 输出绑定到映射输出

如果某个 Mapplet 用于计算映射输出，则需将该 Mapplet 中的输出值传递到映射。

将 Mapplet 输出绑定到映射输出视图上的映射输出。

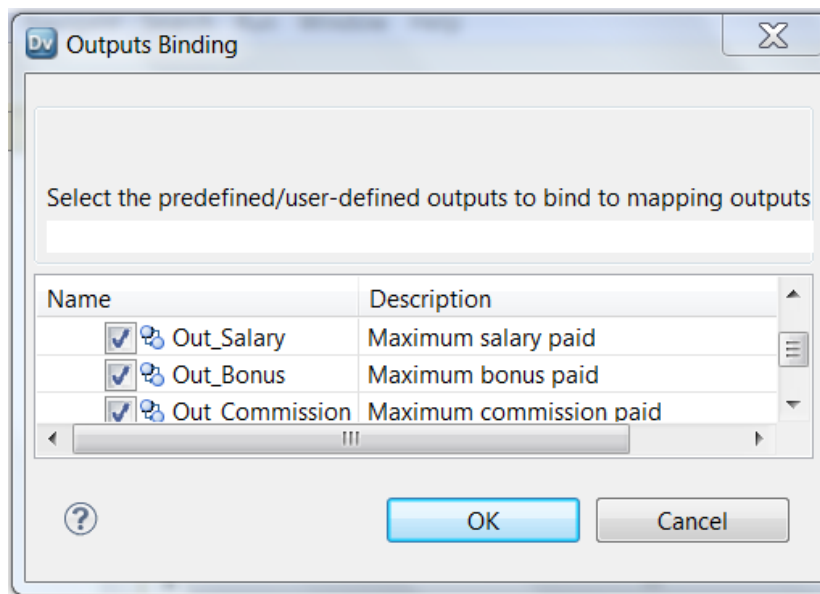
下图显示了映射级别的 Max_Compensation 映射输出：



在映射级别，可以将薪资 Mapplet 输出、奖金 Mapplet 输出和佣金 Mapplet 输出绑定到名为 Max_Compensation 的相同映射输出。

要将 Mapplet 输出绑定到映射输出，请单击映射输出的绑定列。此时将显示可用 Mapplet 输出的列表。该列表包含与映射输出类型和汇总相同的 Mapplet 输出。选择要分配给映射输出的 Mapplet 输出。

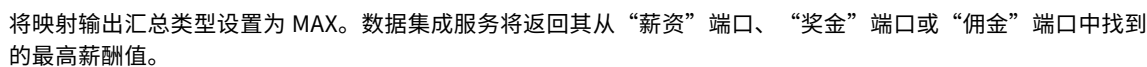
下图显示了输出绑定对话框：



选择要绑定到 Max_Compensation 的输出后，绑定字段将包含以下文本：

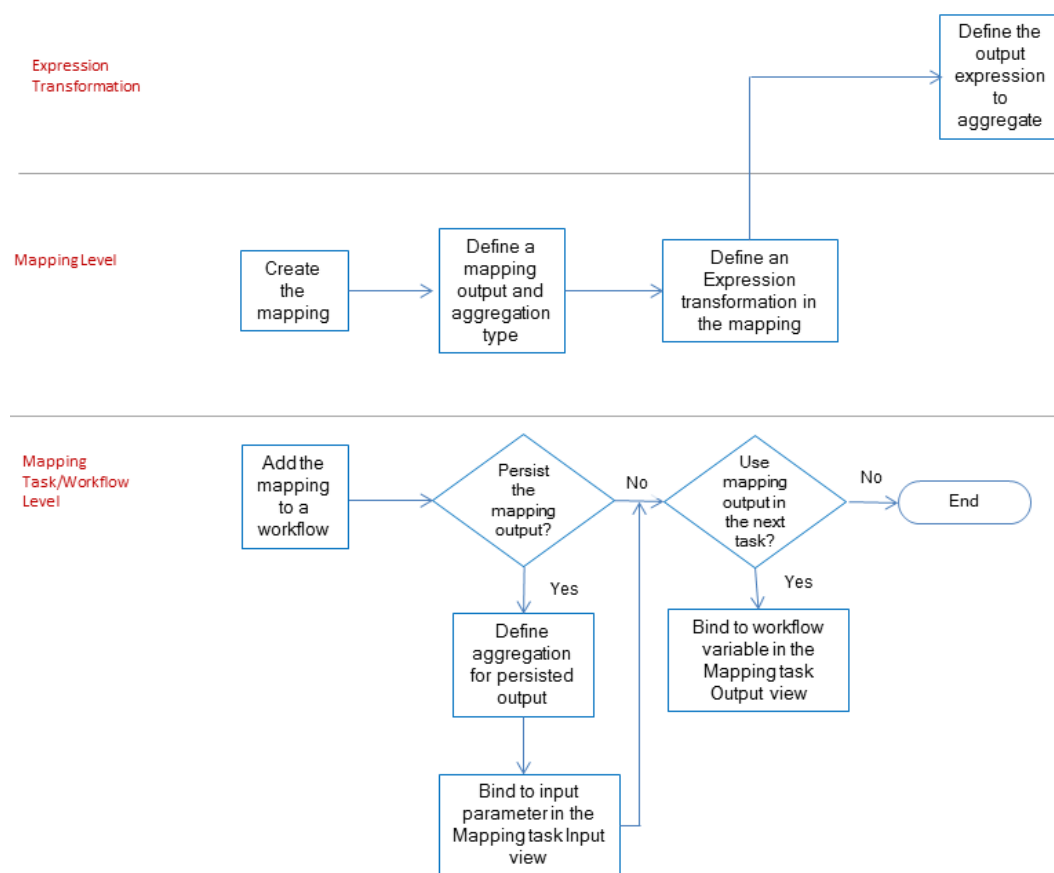
Mplt_Compensation.Out_Salary,Mplt_Compensation.Out_Bonus,Mplt_Compensation.Out_Commission

下图显示了绑定字段中的映射输出：



逻辑数据对象可以包含读取或写入映射。可以将这些映射配置为返回映射输出，还可以将逻辑数据对象中的映射输出绑定到映射中的映射输出。

如何配置映射输出



要配置映射输出，请执行以下步骤：

1. 创建映射。
2. 在映射的**输出**视图中，定义映射输出名称和汇总类型。
3. 将表达式转换添加到映射，然后在表达式的**映射输出**视图中配置映射输出表达式。
4. 要创建映射任务，请将映射添加到工作流中。
5. 将映射输出保留在映射任务的**持久性**视图中，然后配置保留值的汇总函数类型。
6. 将保留的映射输出分配给映射任务中的一个输入参数。
7. 如果希望在另一个工作流任务中使用映射输出，请将映射输出分配给工作流变量。

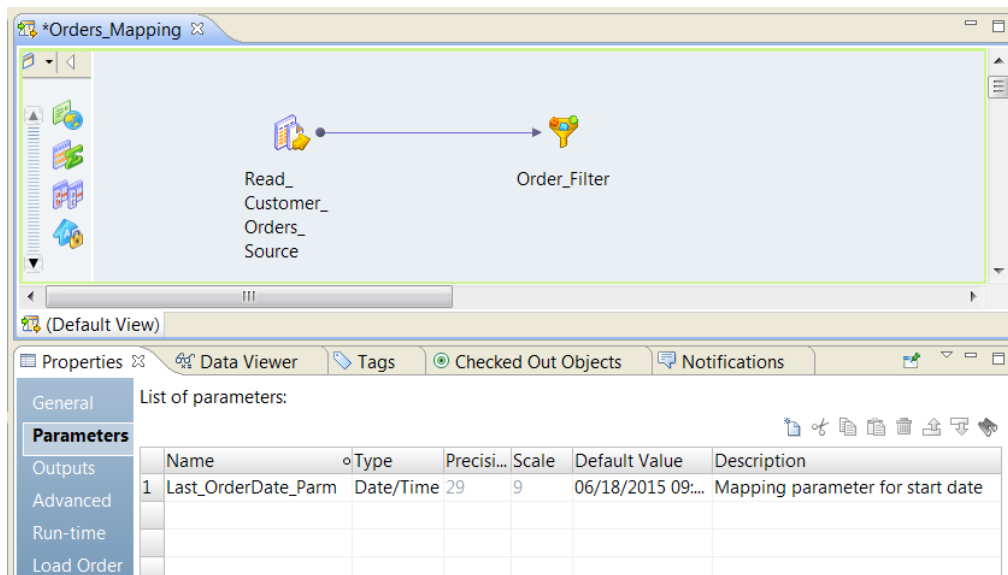
创建映射

创建一个包含可重用筛选器转换的映射。筛选器转换用于筛选订单日期早于特定日期的行。筛选器表达式包括一个名为 Last_Order_Date_Parm 的参数。

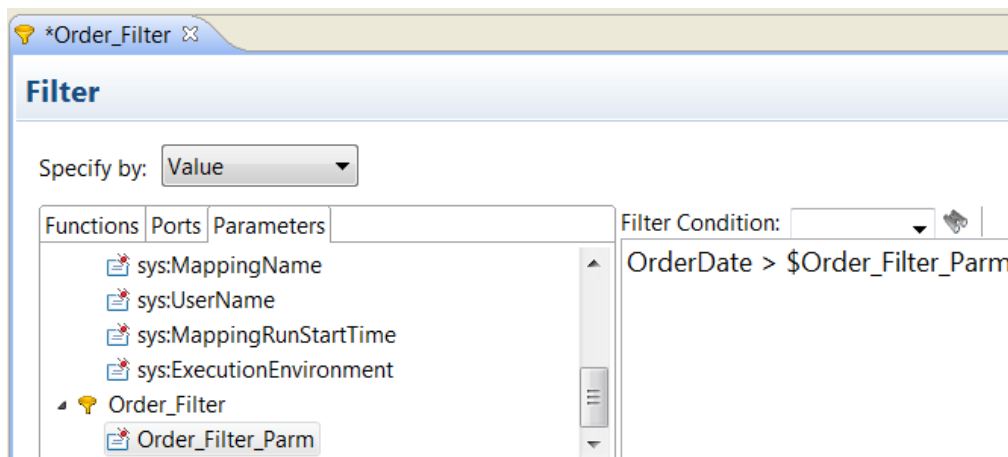
1. 创建用于处理 Customer_Order 文件中订单数据的映射。
2. 在映射的“属性”视图中，单击**参数**选项卡。
3. 添加一个名为 Last_Order_Date_Parm 的日期/时间映射参数。

输入起始参数的默认值。

下图显示了映射参数：

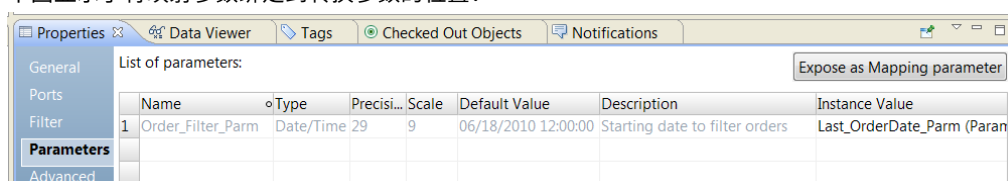


4. 创建一个用于筛选 Customer_Order 行的可重用筛选器转换。
5. 在名为 Order_Filter 的筛选器转换中定义一个参数。
输入起始参数的默认值。
6. 添加用于查找晚于该参数的订单日期的筛选器表达式：



7. 将筛选器转换添加到映射中。
8. 单击筛选器转换以显示转换的属性视图。
9. 单击参数选项卡。
10. 要将 Order_Filter_Parm 转换参数绑定到 Last_Order_Date 映射参数，请单击 Order_Filter_Parm 的实例值列。
11. 选择 Last_Order_Date。

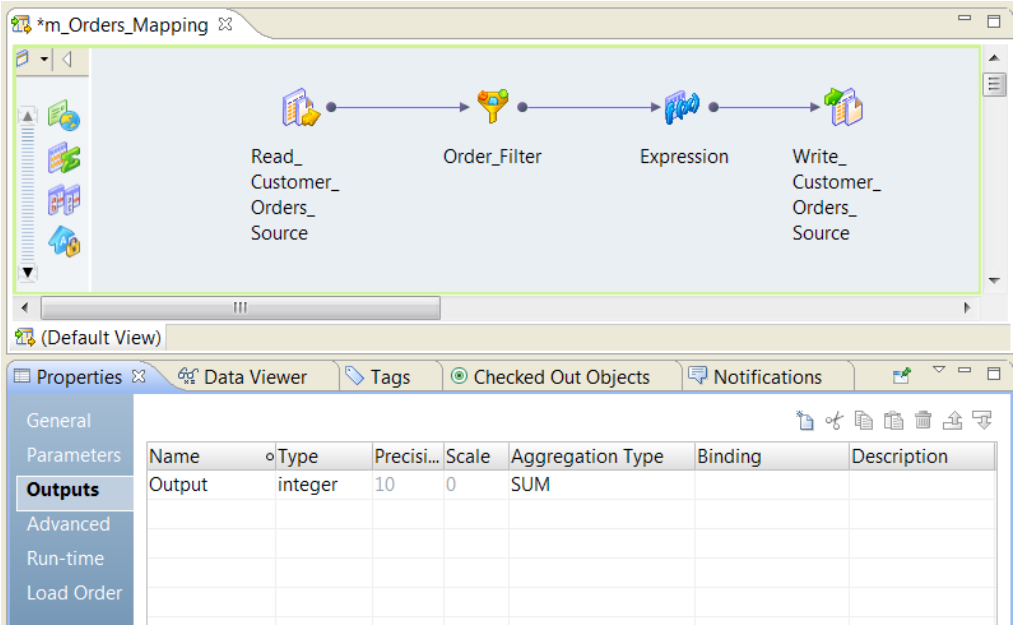
下图显示了将映射参数绑定到转换参数的位置：



定义映射输出

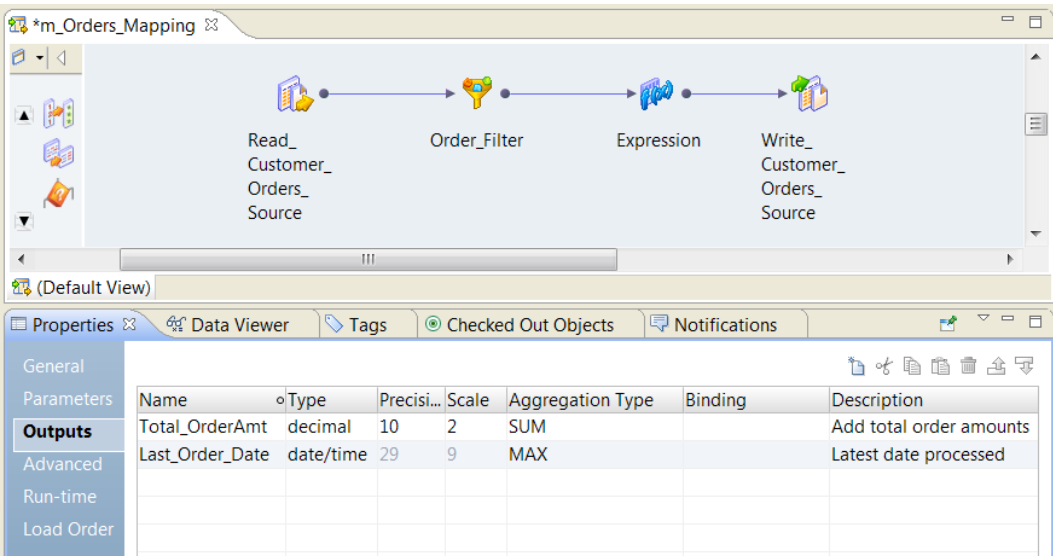
创建映射，然后在映射的**属性**中定义映射输出。每个映射输出定义都说明要执行的汇总类型和结果的数据类型。

- 1. 创建映射后，单击编辑器以访问映射的**属性**。
 - 2. 单击**输出**视图。
 - 3. 单击**新建**创建映射输出。
- Developer tool 会创建一个包含默认字段值的映射输出。
- 下图显示了**输出**视图中的映射输出默认值：



- 4. 更改用于标识映射输出的名称。
 - 5. 选择数值或日期映射输出类型。输入精度和小数位。
 - 6. 选择映射输出的汇总类型。
- 可以汇总输出表达式，也可以查找映射处理的最小或最大表达式值。默认值为 SUM。
- 7. 单击**文件 > 保存**以保存映射输出。
- 必须先保存映射输出，然后才能在表达式转换中创建映射输出表达式。

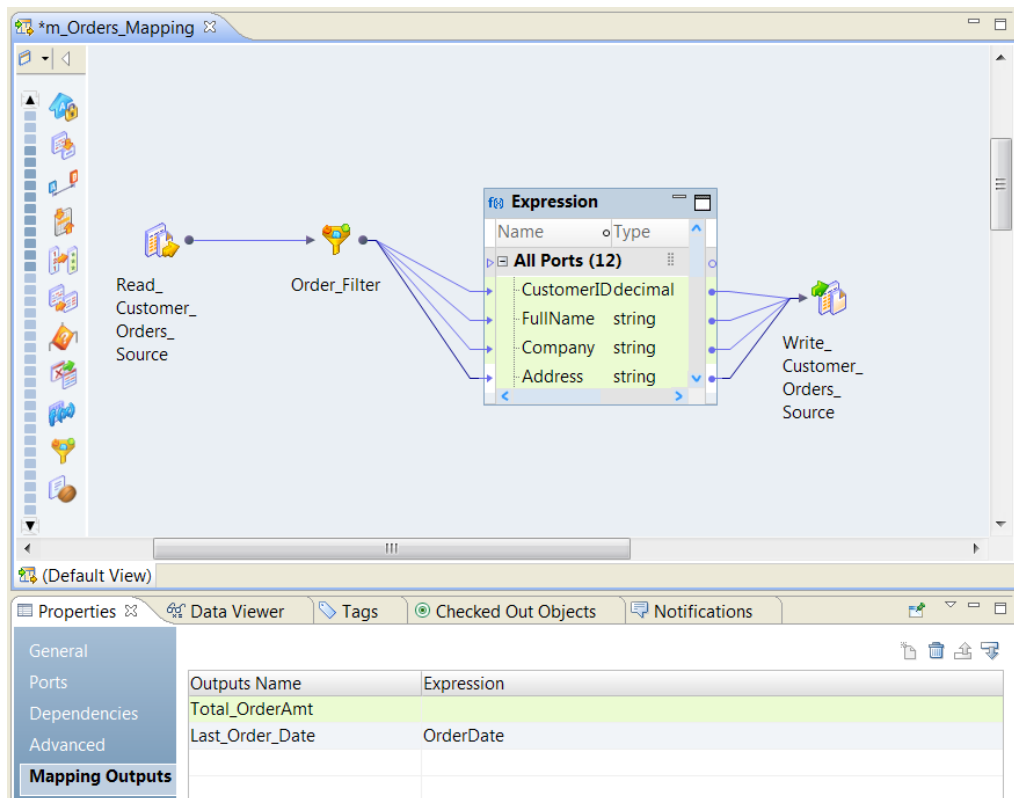
下图显示了一个包含小数字段总和的映射输出和一个包含最大日期值的映射输出：



配置映射输出表达式

在表达式转换中，配置要针对映射处理的每一行进行汇总的表达式。

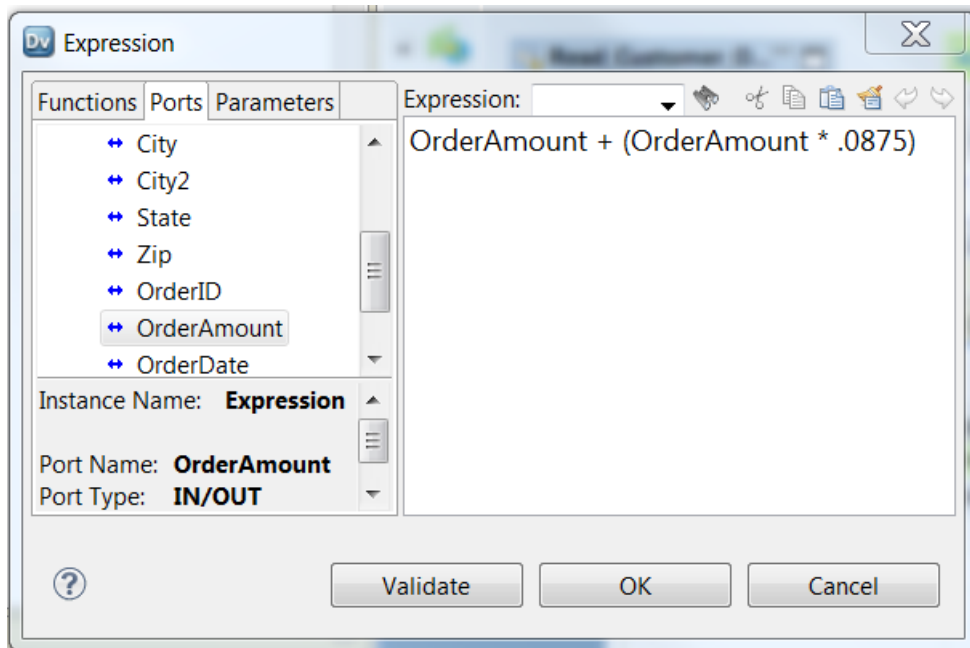
1. 将表达式转换添加到映射。
在确定将转换放置在何处之前，请考虑映射逻辑。映射输出包含表达式转换所接收行的汇总。
2. 在表达式转换中，单击**映射输出**视图。
3. 单击**新建**添加一个映射输出表达式。
Developer tool 会创建输出名称与在映射级别创建的某个映射输出相匹配的映射输出。如果映射的**属性**中包含多个映射输出，请选择适用的映射输出名称。
下图显示了表达式转换中的**映射输出**视图：



- 单击**表达式**列，在表达式编辑器中输入表达式。

表达式可以只包含端口名称，也可以包含函数、端口和参数。

下图显示了表达式编辑器中用于计算 Total_OrderAmt 的表达式：



- 单击**验证**以验证表达式是否有效。
- 单击**确定**保存表达式。

表达式将显示在映射输出的**表达式**列中。

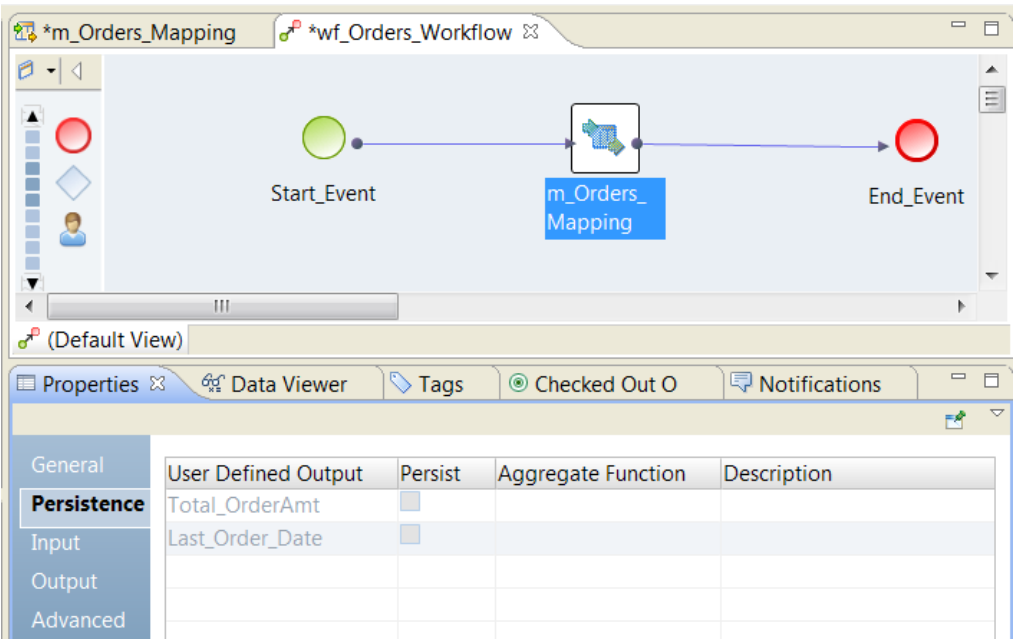
- 7. 单击**文件 > 保存**以保存表达式转换。

保留映射输出

将映射添加到工作流后，可以保留映射任务中的映射输出。可以将保留的映射输出用作映射任务在下次运行时的输入。

- 1. 将映射添加到工作流以创建映射任务。
- 2. 单击工作流中的映射任务图标，查看映射任务的**属性**。
- 3. 单击**持久性**视图。

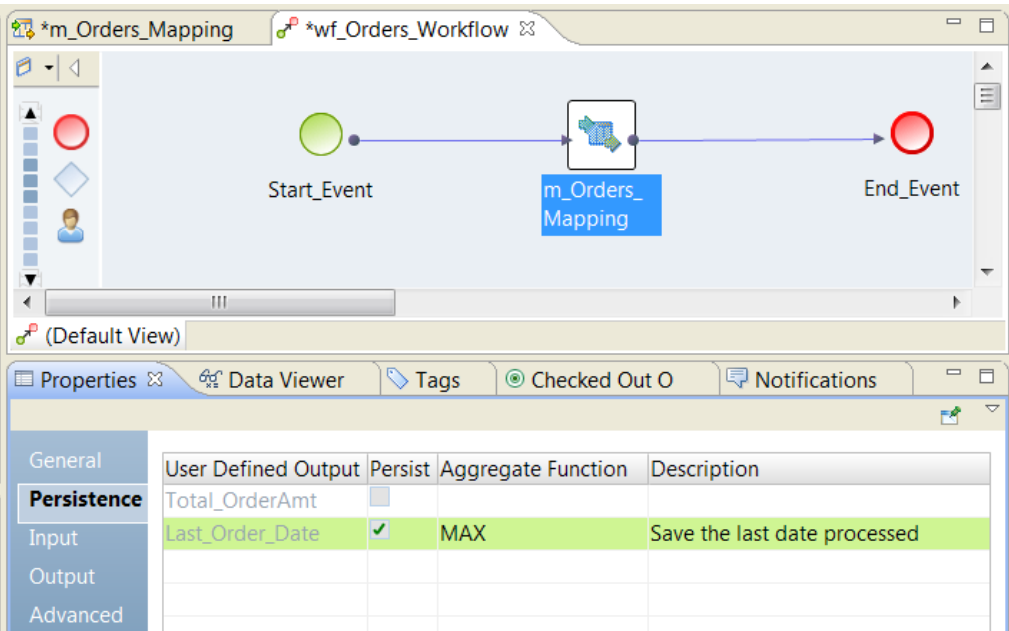
用户定义的映射输出列表将显示在**持久性**视图中。



- 4. 启用**保留**以在映射任务运行后保存映射输出。

5. (可选) 更改汇总类型并输入说明。

下图显示了映射任务的“持久性”视图：



Last_Order_Date 映射输出已保留。由于汇总函数为 MAX，因此数据集成服务将最大订单日期值保存在存储库中。

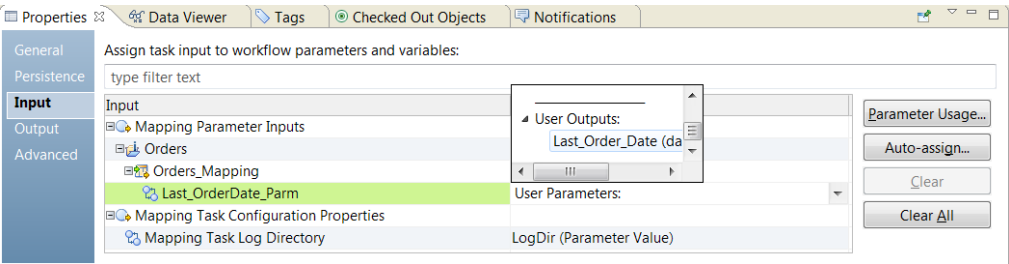
将保留输出分配给映射任务输入

当工作流下次运行时，可以将映射任务中保留的映射输出绑定到同一映射任务的输入参数。

将映射任务中保留的最新订单日期值分配为同一映射任务的输入参数。配置筛选器转换，使其使用 Last_OrderDate_Parm 参数选择要处理的订单。用于选择输入行的筛选器表达式为 Order_Date > Last_OrderDate_Parm。

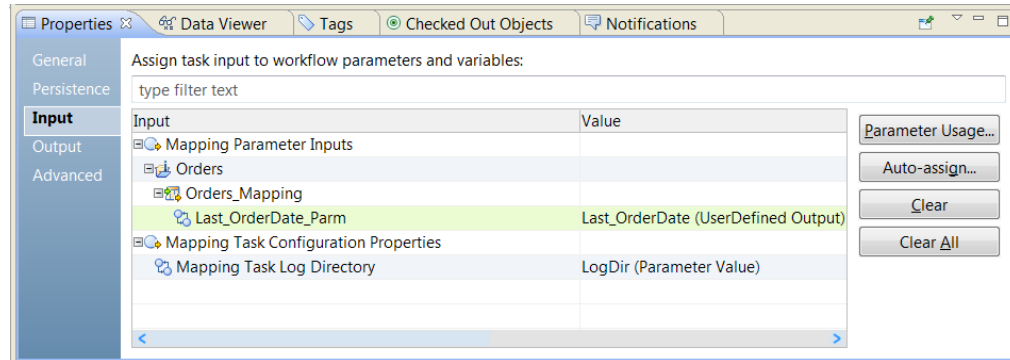
1. 单击工作流中的映射任务图标，查看映射任务的**属性**视图。
此时将显示映射任务输入参数列表和已参数化映射任务配置属性列表。映射必须具有一个要将映射输出分配到的映射参数。
2. 找到要将映射输出绑定到的映射输入参数。双击**值**列查看选择箭头。
3. 单击选择箭头，查看可分配给输入参数的参数和变量列表。
4. 滚动至列表的**用户输出**部分，然后选择要使用的已保留映射输出。

下图显示了映射任务**输入**视图上的 Last_OrderDate_Parm 映射参数：



5. 选择要分配给参数的映射输出。

映射输出名称将显示在输入参数的值列中。



6. 单击**文件 > 保存**以保存映射任务。

Last_OrderDate_Parm 将绑定到保留在存储库中的订单日期值。

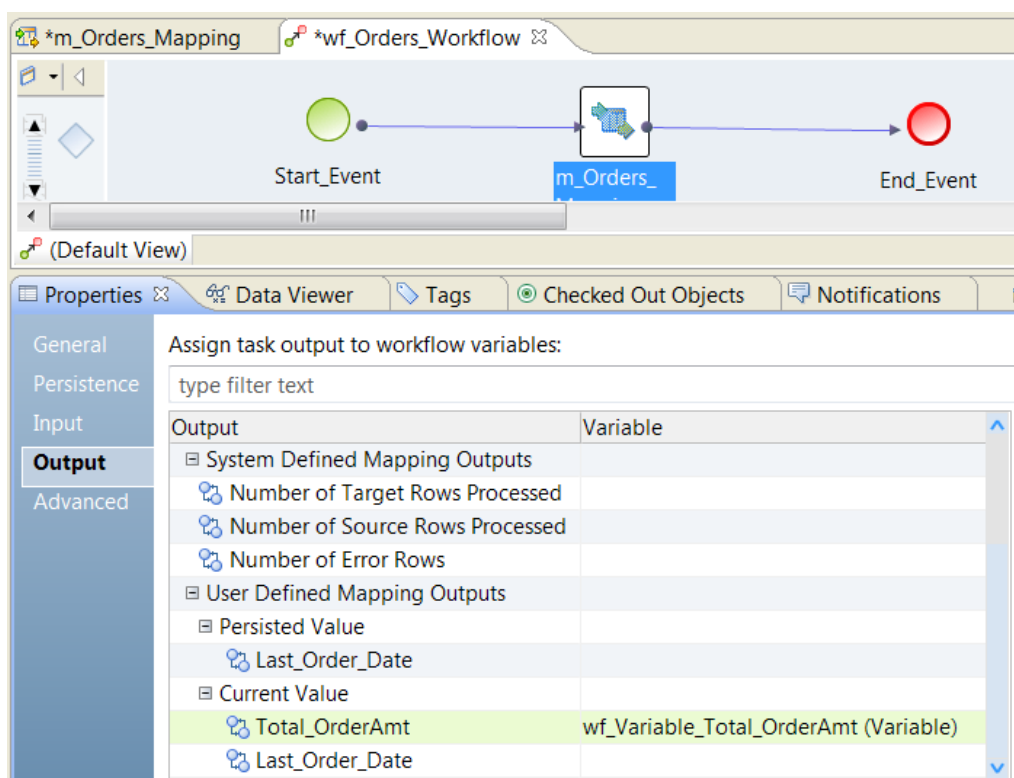
将映射输出绑定到工作流变量

可以将映射输出绑定到工作流变量，然后将值传递到工作流中的其他任务。

要将映射输出值传递到另一个任务，请将映射输出绑定到映射任务**输出**视图中的工作流变量。可以绑定当前映射任务中的映射输出，也可以绑定在上次映射任务运行中保留的映射输出。

1. 将具有映射输出的映射添加到工作流。
2. 单击工作流中的映射任务图标，查看映射任务的**属性**。
3. 在映射任务的**属性**中，单击**输出**视图。
4. 查找要绑定到某个变量的映射输出。
5. 双击**变量**列，访问选择箭头并查看工作流变量列表。

下图显示了映射任务**输出**视图中可将 Total_Order_Amt 映射输出绑定到 wf_Variable_Total_OrderAmt 工作流变量的位置。



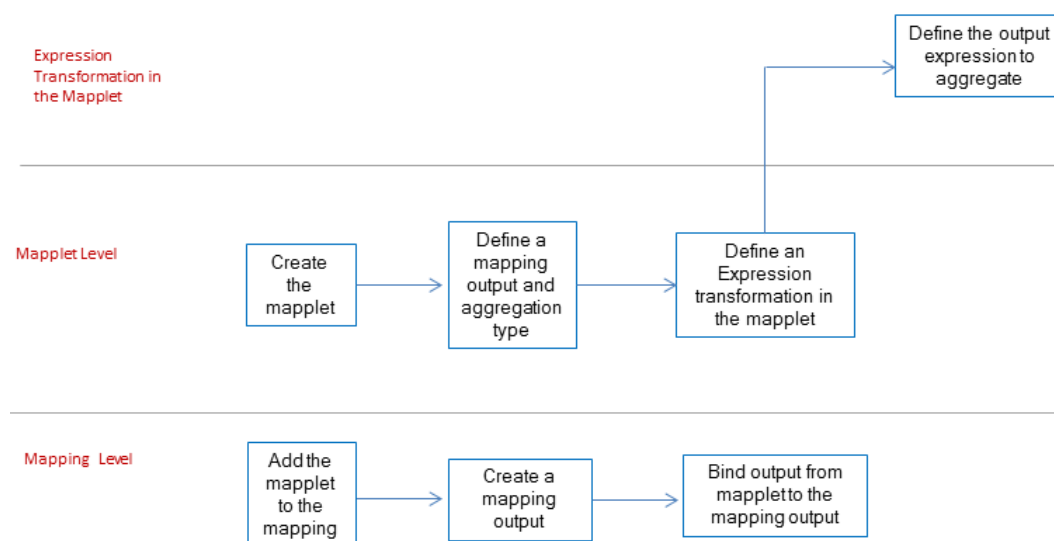
6. 要创建工作流变量，请在值列中单击工作流变量列表中的**新建变量**选项。
输入变量的名称、类型和默认值。

如何将 Mapplet 输出绑定到映射输出

可以将 Mapplet 配置为返回映射输出。可以将 Mapplet 中的映射输出绑定到映射级别的映射输出。

将某个 mapplet 加入映射中后，该 mapplet 会计算输出值并将输出值传递到映射。可以将 Mapplet 中的多个输出绑定到映射级别的同一个输出。还可以将 Mapplet 中的系统定义输出绑定到映射输出。

下图显示了配置 Mapplet 输出并将其绑定到映射输出的过程：



要将 Maplet 中的输出绑定到映射输出，请执行以下步骤：

1. 创建 Maplet。
2. 在 Maplet 的**输出**视图中定义 Maplet 输出名称和汇总类型。
3. 将表达式转换添加到 Maplet，然后在表达式的**映射输出**视图中配置映射输出表达式。
4. 将 Maplet 添加到映射中。
5. 在映射中创建映射输出。
6. 将 Maplet 中的输出绑定到映射输出。

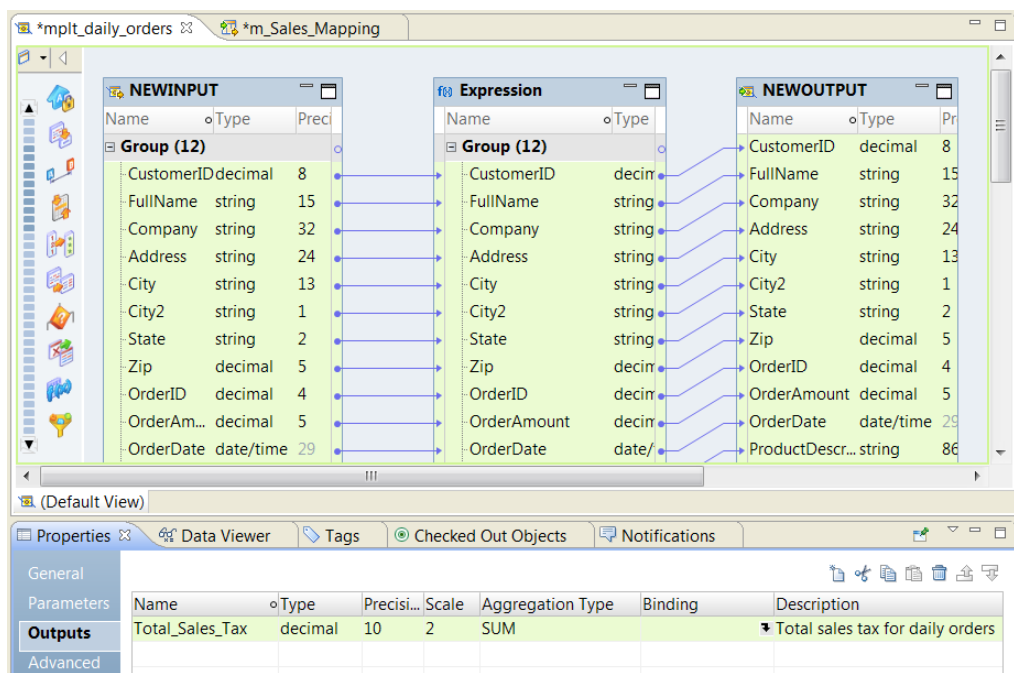
定义 Maplet 输出

创建 Maplet，然后在 Maplet **属性**视图的**输出**选项卡中定义映射输出。每个映射输出定义都说明要执行的汇总类型和结果的数据类型。

1. 创建 Maplet 后，在映射画布中单击以访问 Maplet 属性。
2. 单击**输出**视图。
3. 单击**新建**创建映射输出。
Developer tool 会创建一个包含默认字段值的映射输出。
4. 更改用于标识映射输出的名称。
5. 选择数值或日期映射输出类型。如果要创建数值类型，请输入精度和小数位数。
6. 选择映射输出的汇总类型。

可以汇总输出表达式，也可以查找映射处理的最小或最大表达式值。默认值为 SUM。

下图显示了一个汇总类型为 SUM、名为 Total_Sales_Tax 的 Maplet 输出：

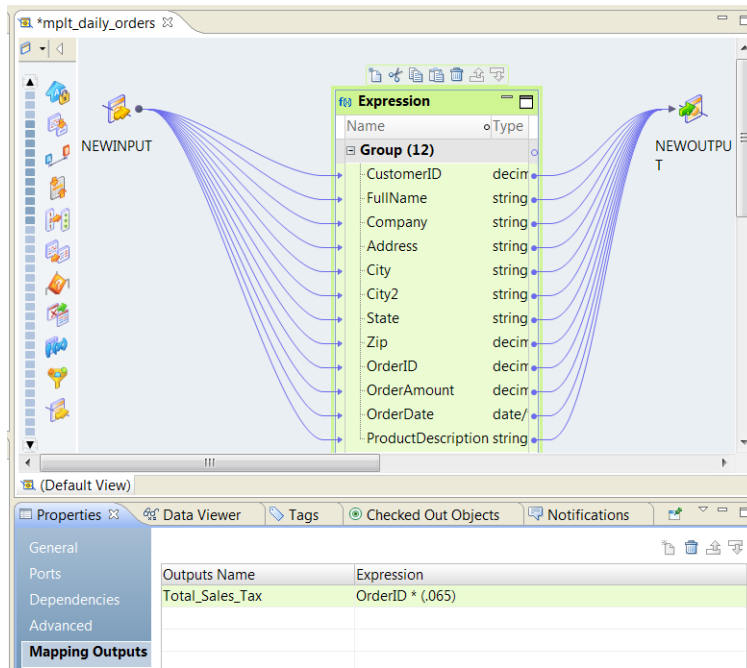


- 单击**文件 > 保存**以保存映射输出。
必须先保存映射输出，然后才能在表达式转换中创建任何映射输出表达式。

在 Maplet 中配置映射输出表达式

将表达式配置为针对 Maplet 处理的每一行进行汇总。

- 将表达式转换添加到 Maplet。
在确定将转换放置在何处之前，请考虑 Maplet 逻辑。
- 在表达式转换中，单击**映射输出**视图。
- 单击**新建**添加输出表达式。
Developer tool 会创建输出名称与您在 Maplet 级别创建的某个映射输出相匹配的映射输出。可能有多个输出供您选择。
- 使用表达式编辑器输入表达式。
表达式可以包含端口名称，也可以包含函数、端口和参数。
- 单击**验证**以验证表达式是否有效。
- 单击**确定**保存表达式。
在下图显示的**映射输出**视图中，包含一个用于计算销售税的映射输出表达式：



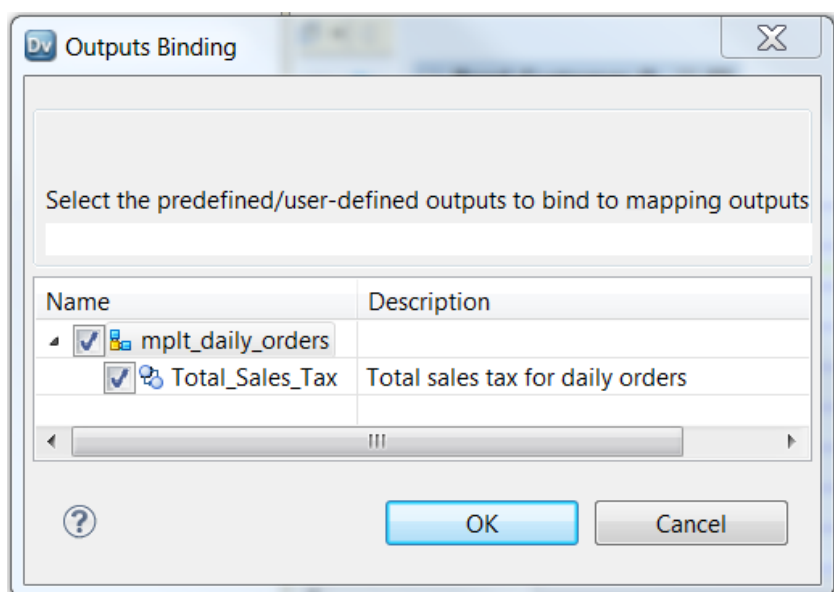
7. 单击**文件 > 保存**以保存表达式转换。

将 Maplet 输出绑定到映射输出

当您将在 Maplet 加入映射后，可以将 Maplet 中的输出绑定到您在映射级别定义的映射输出。

1. 定义映射，然后将 Maplet 添加到该映射中。
2. 单击映射画布，查看映射的**属性**。
3. 单击**输出**视图。
4. 单击**新建**创建映射输出。
Developer tool 会创建一个包含默认字段值的映射输出。
5. 更改映射输出类型、汇总类型、精度和小数位数，以便与要绑定到的 Maplet 中的字段相匹配。
6. （可选）更改名称并输入说明。
7. 单击“**绑定**”字段中的选择箭头，查看输出列表。

下图显示了“输出绑定”对话框：



8. 选择要绑定到映射输出的 Mapplet 输出。
可以选择多个 Mapplet 输出以绑定到同一个映射输出。
9. 单击“确定”。

您选择的 Mapplet 输出将显示在**绑定**字段中。
下图显示了映射输出**绑定**字段中的 Mapplet 输出名称：

Name	Type	Precisi...	Scale	Aggregation Type	Binding	Description
Total_Tax	decimal	10	2	SUM	mplt_daily_orders.Total_Sales_Tax	Sales tax

第 5 章

从 SQL 查询生成映射

本章包括以下主题：

- [从 SQL 查询生成映射概览, 93](#)
- [从 SQL 查询生成映射的示例, 93](#)
- [用于生成映射的 SQL 语法, 94](#)
- [生成映射的查询中的函数支持, 94](#)
- [从 SQL 查询生成映射或逻辑数据对象, 96](#)
- [从 SQL 语句生成映射, 97](#)

从 SQL 查询生成映射概览

在 Developer tool 中，可以从 SQL 查询生成映射。要生成映射，您可以输入 SQL 查询或加载包含查询的文本文件。（可选）可以定义查询表的源。Developer tool 会验证 SQL 查询并生成映射。

也可以从只含 SELECT 语句的 SQL 查询生成逻辑数据对象。

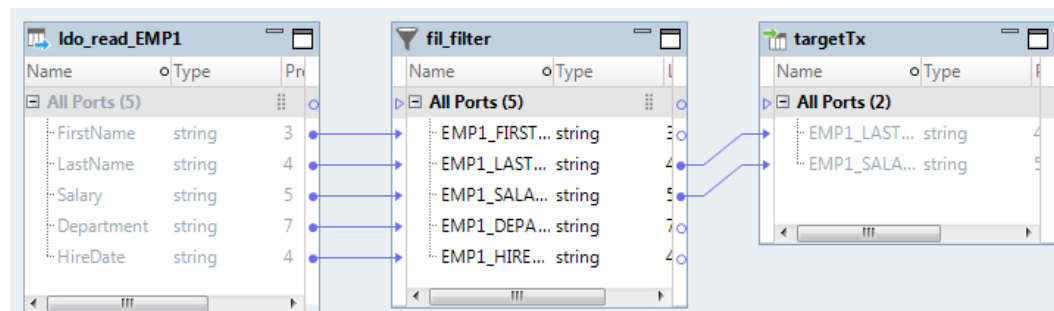
从 SQL 查询生成映射的示例

您有一个员工表，并希望获得在 2001 年 1 月 1 日以后入职的员工的薪酬列表。

为此可以创建以下 SQL 语句：

```
SELECT LastName, Salary from emp1 where HireDate > 01/01/2001
```

下图显示了从 SQL 语句创建的映射：



用于生成映射的 SQL 语法

可以在 Developer tool 中使用符合 ANSI 标准的 SQL 语句生成映射。

Developer tool 可以从标准 SELECT 查询生成映射。例如：

```
SELECT column_list FROM table-name
[WHERE clause]
[GROUP BY clause]
[HAVING clause]
[ORDER BY clause]
```

如果 SELECT SQL 语句包含相关子查询，则当可以将查询平面化或重新编写为单个标准查询时，该查询是有效的。

ANSI SQL 不支持某些数据类型。例如，如果查询从数据源中请求结果，而数据源中的某个列具有类型 timeStampTZ，则该 SQL 无效。

相关子查询

相关子查询是一种子查询，这种子查询在其 WHERE 子句中使用来自外部查询的值。数据集成服务会在运行查询之前将相关子查询平面化。

下表显示了数据集成服务展平的一个相关子查询的结果：

类型	查询
未展平	SELECT huge.* FROM huge WHERE c1 IN (SELECT c1 FROM tiny)
展平后	SELECT huge.* FROM huge, tiny WHERE huge.c1 = tiny.c1

当相关子查询满足以下要求时，数据集成服务可将其平面化：

- 类型为 IN 或定量比较。
- 不位于 OR 运算符内或者不属于 SELECT 列表的一部分。
- 不包含 LIMIT 关键字。
- 不包含 GROUP BY 子句、SELECT 列表中的汇总或者 EXIST 或 NOT IN 逻辑运算符。
- 生成具有唯一性的结果。相关子查询的其中一列是主键。例如，如果 r_regionkey 列是 vs.nation 虚拟表的主键，您可以发出以下查询：SELECT * FROM vs.nation WHERE n_regionkey IN (SELECT b.r_regionkey FROM vs.region b WHERE b.r_regionkey = n_regionkey)。
- 如果它包含 FROM 列表，则 FROM 列表中的每个表都是 SQL 数据服务中的一个虚拟表。

生成映射的查询中的函数支持

Informatica 支持符合 ANSI SQL-92 标准的函数。

此外，部分函数具有特定的语法要求。

下表列出了函数和支持的语法：

功能	语法
DATE()	指定日期格式： DATE(format '<format>') 其中 <format> 是一种标准日期格式。 示例： SELECT DATE(format 'dd-mm-yyyy') from table
POSITION()	确定子字符串在文字字符串中的位置： POSITION('<substring>', '<string>') 示例： POSITION('MA', 'James Martin') 确定子字符串在表列中的位置： POSITION('<substring>', <column_name>) 示例： POSITION('MA', FULL_NAME)

从包含不受支持的函数的 SQL 查询生成映射

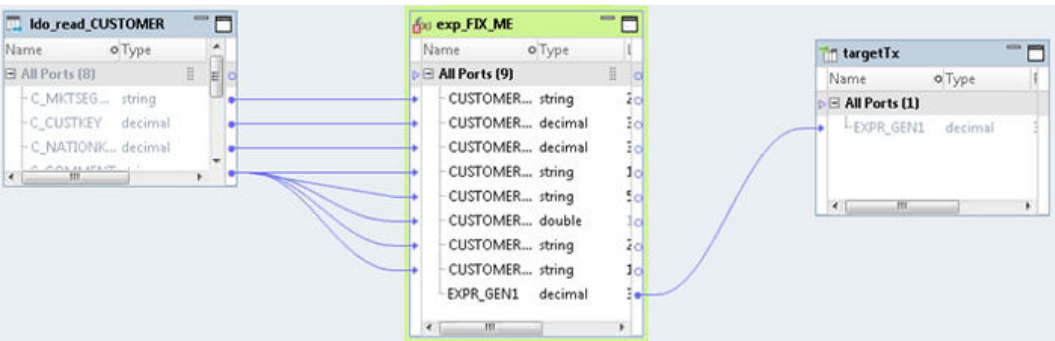
当 Developer tool 从 SQL 生成映射时，它会验证查询中的函数。请使用符合 ANSI 标准的 SQL 以确保生成的映射有效。

如果 Developer tool 在有效的 SQL 语句中遇到未知函数，它可能会生成包含带有标签 FIX_ME 的转换或带有标签 FIX_EXPR 的表达式的映射。请编辑这些对象以修复映射并获取有效结果。未知函数会在映射日志文件中显示为警告消息。

例如，您可以使用以下 SQL 语句生成映射：

```
SELECT unknownFunctionABC(c_custkey,c_comment) from customer
```

下图显示了从该 SQL 语句生成的映射如何包含需要修复的表达式转换：



请注意，该表达式以错误图标标记。请使用“端口”选项卡编辑不正确的表达式。更正错误之前，映射是无效的。

INSERT、UPDATE 和 DELETE 语法

可以使用以下语法创建有效的 INSERT、UPDATE 和 DELETE 语句：

- 对于 INSERT 语句，请使用以下语法：

```
INSERT INTO <TABLENAME> [<list>]  
<select query>
```

- 对于 UPDATE 语句，请使用以下语法：

```
UPDATE [schema .] { table | view } [ alias ]  
SET column = { expr | subquery }  
[, column = { expr | subquery } ]... [WHERE condition]
```

- 对于 DELETE 语句，请使用以下语法：

```
DELETE FROM <Table> [[<AS>] <ALIAS>] [WHERE condition]
```

INSERT、UPDATE 和 DELETE 语句的规则和准则

请注意 INSERT、UPDATE 和 DELETE 语句的以下规则和准则：

- INSERT、UPDATE 或 DELETE 语句将在映射中创建源对象和目标对象，这些对象是逻辑数据对象。
- 仅有一个 INSERT、UPDATE 或 DELETE 语句是有效的。例如，包含一个 INSERT 语句和一个嵌套 UPDATE 语句的语句是无效的。
- 当 INSERT、UPDATE 或 DELETE SQL 语句包含相关子查询时，Developer tool 无法生成映射。
- UPDATE 或 DELETE 语句将在映射中创建更新策略转换。由于更新策略转换需要主键，因此数据目标必须包含主键。生成映射之后，请验证主键。
- Developer tool 会忽略 ORDER BY 子句中的任何 INSERT 语句，因为关系数据库在插入数据时不会遵从顺序。

从 SQL 查询生成映射或逻辑数据对象

可以将 SQL 语句转换为映射或逻辑数据对象。您可能想要生成逻辑数据对象，以创建可在其他映射中重用的对象。

- 单击**文件 > 新建 > 从 SQL 查询映射**。
此时将打开**从 SQL 查询生成映射或逻辑数据对象**对话框。
- 选择要输入 SQL 查询还是选择包含 SQL 查询的文件。
 - 要输入可编辑的 SQL 查询，请选择**输入 SQL 查询**，在编辑器中键入或粘贴 SQL 查询，然后单击**验证查询**。
 - 要选择包含 SQL 查询的文件，请选择**选择一个 SQL 文件**，然后浏览到包含 SQL 查询的文件。Developer tool 会验证 SQL 语法。如果语法无效，您必须修复语法才能继续操作。
- 要生成逻辑数据对象而非映射，请选择**生成映射作为逻辑数据对象**。
- （可选）重命名要生成的映射或逻辑数据对象。
- 单击**下一步**。
对话框将显示与数据源对应的表。
- 单击**数据源**下的表的行以选择映射的数据源。
如果该表具有数据源，您可以选择单击该数据源以更改数据源。
选择数据源对话框将打开，并列出了您可以访问的模型存储库上的表。

7. 选择模型存储库中的任何数据源。

8. 单击**完成**。

生成的映射或逻辑数据对象将在编辑器中打开。

您可以选择映射中的任何对象以查看或编辑该对象。然后，可以运行映射，或者将该映射包含在部署到数据集成服务的应用程序或工作流中。

如果创建了逻辑数据对象，则您可以在其他映射中重用该逻辑数据对象。例如，您可以在映射中重用已生成的逻辑数据对象作为源。

从 SQL 语句生成映射

要从 SQL 语句生成映射，请执行以下任务：

1. 创建 SQL 语句。
2. 将 SQL 语句粘贴或导入到 Developer tool，验证 SQL 语句，并生成映射。
3. 完成映射开发。请遵循以下步骤：
 - a. 测试并迭代开发映射，直至其满足要求。
 - b. 将映射部署到数据集成服务。

创建 SQL 语句

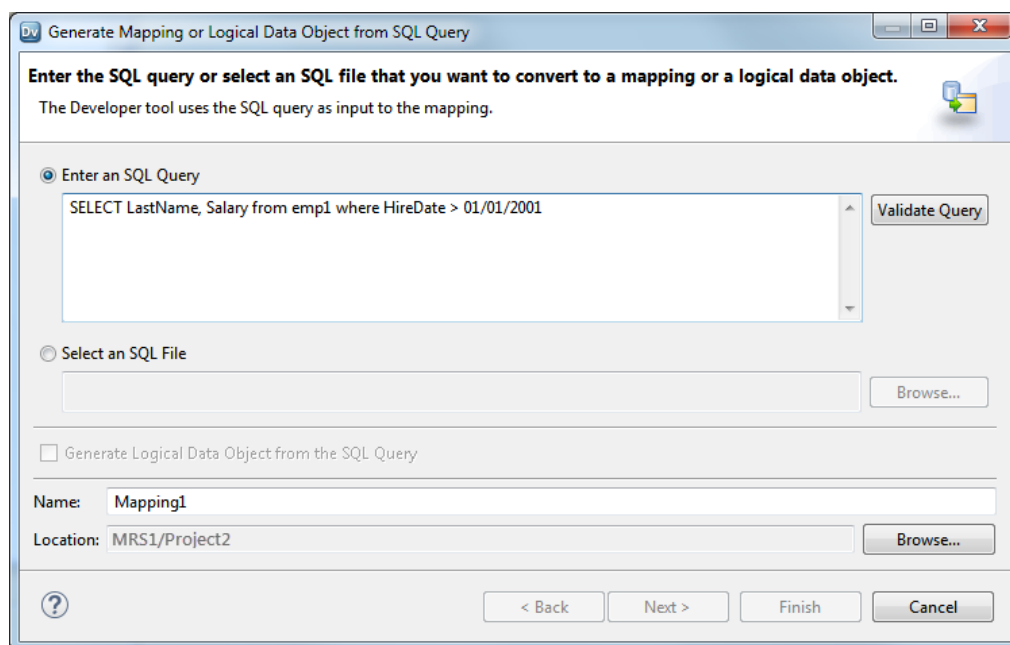
要使用 SQL 语句生成映射，请创建 SQL 语句。

您可以使用 SQL 查询工具或从头编写 SQL 语句来创建 SQL 语句。请遵循本文的语法准则。

注意：支持某些非 Informatica 函数。其他函数可在有效查询中使用，用于生成包含无效结果的映射。有关 SQL 语句中支持的函数的详细信息，请联系 Informatica 全球客户支持。

将 SQL 语句粘贴或导入到 Developer tool

1. 找到包含要导入的 SQL 语句的 SQL 文件，或将整个语句复制到剪贴板。
2. 在 Developer tool 中，单击**文件 > 新建 > 从 SQL 查询映射**
此时将打开**从 SQL 查询生成映射或逻辑数据对象**对话框。



3. 将查询导入到对话框。选择以下方法之一：
 - 选择**输入 SQL 查询**，然后将查询从剪贴板粘贴到编辑器。
 - 选择**选择 SQL 文件**，然后浏览并选择文件。
4. 单击**验证**。

Developer tool 会验证 SQL 语句。更正任何错误。
5. 如果您想生成逻辑数据对象而不是映射，请选择**从 SQL 查询生成逻辑数据对象**。

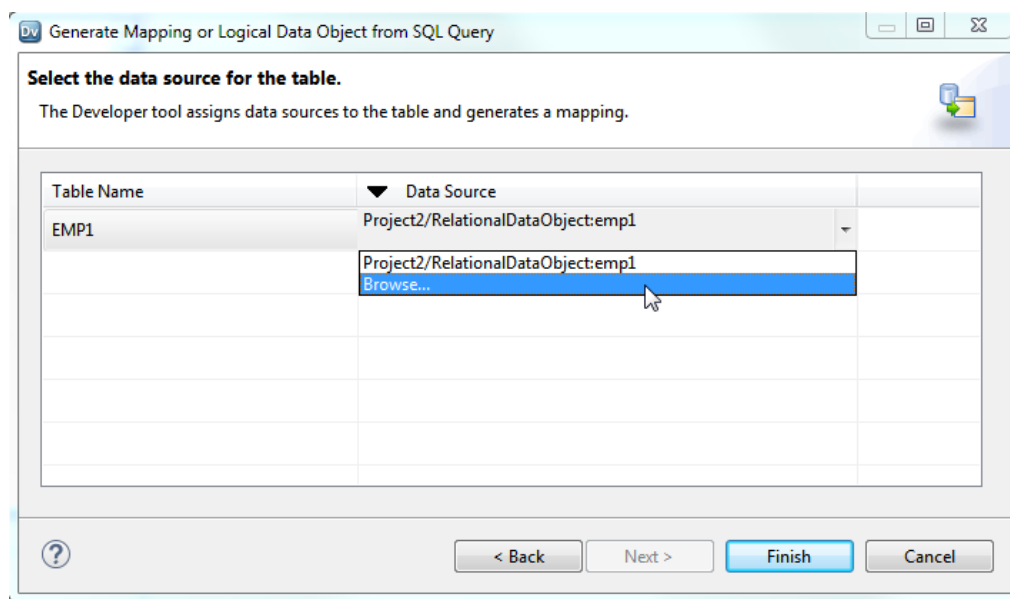
如果选择此选项，请执行以下步骤：

 1. （可选）重命名要创建的逻辑数据对象。
 2. （可选）单击**浏览**选择逻辑数据对象的位置，或者接受默认位置。
 3. 单击**验证**。

Developer tool 会验证 SQL 语句。更正任何错误。
6. 单击**下一步**。

此时会打开**选择表的数据源**对话框。
7. 要选择表的数据源，请单击数据源列，然后单击**浏览**。

下图显示了在哪个位置单击**浏览**以选择数据源：



8. 单击**完成**。

Developer tool 从 SQL 查询生成映射，并在编辑器中打开映射。

完成映射开发

创建映射后，请执行以下步骤来完成映射的开发：

1. 运行映射并查看结果。
2. 迭代编辑并重新运行映射，直至其满足要求。
3. 在数据集成服务中部署并运行映射。
您可以让映射自行部署，或将其包括在您部署的应用程序中。如果让映射自行部署，数据集成服务会创建一个应用程序以包含此映射。

有关映射、应用程序和部署的详细信息，请参阅《*Informatica 10.1.1 Developer tool 指南*》。

第 6 章

动态映射

本章包括以下主题：

- [动态映射概览, 100](#)
- [动态映射配置, 101](#)
- [动态源, 103](#)
- [动态目标, 106](#)
- [动态端口和生成的端口, 110](#)
- [动态表达式, 112](#)
- [输入规则, 113](#)
- [选择规则和端口选择器, 121](#)
- [设计时链接, 123](#)
- [运行时链接, 125](#)
- [动态映射故障排除, 128](#)

动态映射概览

动态映射是一种可接受源、目标和转换逻辑运行时更改的映射。使用动态映射可管理频繁的架构或元数据更改或者针对具有不同架构的数据源重用映射逻辑。配置规则、参数和一般转换属性，以创建动态映射。

对于源、目标或查找，如果数据源更改，可以配置一个映射，以动态地获取元数据运行时更改。在映射内配置参数、规则、端口和链接，以便在映射的所有阶段接收和传播更改。在再次运行映射前，不必手动同步数据对象及更新各转换。数据集成服务能动态地确定转换端口、端口中的转换逻辑及映射内的端口链接。

动态映射示例

您每周从不同部门收到您需要联接和汇总的客户数据。各部门可能定期更改源架构，以包含部门分析附加列。

为了接受数据源的更改，可以创建了动态映射。配置读取转换，以便在读取时获取数据对象列。创建输入规则，以包含您需要的列及排除所有其他列。

动态映射配置

如果源更改，可以配置读取转换以接受更改。例如，可以将转换配置为使用不同数据源或根据数据源更新数据对象。如果目标更改，可以将写入转换配置为接受目标更改。例如，可以将写入转换配置为根据关联的数据对象或映射流配置生成列。如果目标是关系型，可以在运行时创建或替换表。

在映射或 Mapplet 中配置转换，使其在整个映射中接收和传播更改。按数据流创建动态端口，以接收新列或发生更改的列。动态端口为各传入列生成端口。配置输入规则，以确定动态端口所接收的列及重命名或重新排序生成的端口。

使用动态端口或表达式中的选择规则创建动态表达式。当包含动态端口时，表达式针对动态端口生成的每个端口运行。当包含选择规则时，表达式针对规则中的每个端口运行。

当表达式、连接器或查找转换包含生成的端口时，可以配置端口选择规则接受映射运行时生成的端口的更改。例如，您需要进行销售数据计算，但各源的销售列名称不同。创建规则，选择正确的列进行计算。

可以在运行时用参数更改值。使用参数更改映射内源、目标、连接和规则等值。

转换可能改变，致使在设计映射时无法创建直接链接。如果在设计时无法创建链接，可以配置运行时链接。运行时链接使用策略或参数确定运行时转换组之间的链接端口。

动态数据源

可以将映射配置为接受运行时源和目标的更改。动态映射可以包含平面文件和关系数据源。可以按预期的更改类型使用参数并配置转换属性。

可以将映射配置为接受运行时下列数据源的更改：

源

动态源可以包含关系源和平面文件源。配置读取转换和物理数据对象，以接受运行时更改。可以根据文件或源连接的位置、传入源列的更改或数据对象更改源元数据。

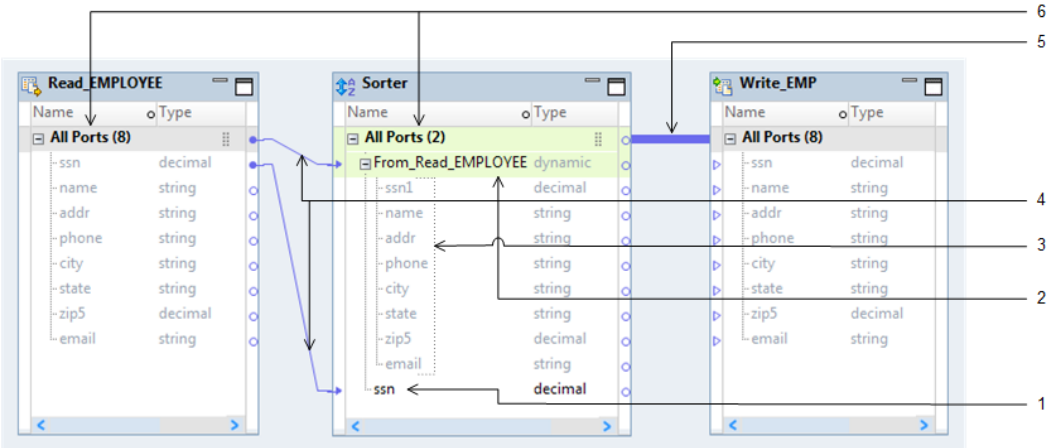
目标

动态目标可以包含关系目标和平面文件目标。可以根据映射流或关联的数据对象定义目标。也可以选择从数据源获取数据对象的列。可以用参数更改目标数据对象或目标连接等运行时属性。

动态映射端口和链接

为了处理元数据的更改，可以配置常规映射不包含的端口和链接类型。

动态映射中可见的端口和映射如下图所示：



- 1. 静态端口（端口）
- 2. 动态端口
- 3. 生成的端口
- 4. 设计时链接（链接）
- 5. 运行时链接
- 6. 端口组

静态端口（端口）

在任何类型的映射（动态或非动态）中可创建的端口。数据可进出端口，且不包含任何动态配置。

动态端口

转换中可以接收来自上游转换的一个或多个列的端口。动态端口可根据映射中传递的元数据来接收新列或发生更改的列。

生成的端口

表示动态端口内单一列的端口。动态端口会根据动态端口规则为每个列创建生成的端口。

设计时链接（链接）

为连接转换间传播数据的端口而创建的链接。在常规映射中也可创建这些链接。

运行时链接

转换组之间的链接，数据集成服务使用该链接根据策略和/或参数确定运行时要连接的端口。

端口组

在代表一行数据的映射中的端口集。在动态映射中，可以将一个组拉到下游转换中，以创建动态端口。

动态映射规则

在动态转换内创建规则，以控制动态端口接收的端口及其生成的端口。

可以配置下列类型的动态映射规则：

输入规则

输入规则定义动态端口生成的端口。可以选择包含或排除端口。也可以选择重命名及重新排序生成的端口。

选择规则和端口选择器

创建选择规则，以定义数据集成服务在运行时处理的生成的端口。在端口选择器内创建选择规则。端口选择器包含在表达式中或在联接或查找条件中可引用的端口。可以根据运行时预期的元数据更改，配置转换中的多个端口选择器。

动态映射中的参数

参数是可以在映射运行之间更改的常量值。使用动态映射中的参数可以更改平面文件或关系资源的源和目标。也可以用参数来更改输入规则、选择规则、转换属性和运行时链接。

下表列出了可以为动态映射组件创建的参数的功能：

动态映射组件	参数功能
汇总器转换	按端口更改组。
联接器转换	更改联接条件。
查找转换	更改查找条件。
等级转换	按端口更改组。
读取转换	创建参数以执行以下任务： <ul style="list-style-type: none">- 更改平面文件源的输入文件名或目录。- 更改关系源的连接。- 更改平面文件数据对象、自定义数据对象或关系数据对象。
规则	创建参数以执行以下任务： <ul style="list-style-type: none">- 按名称或模式更改输入规则条件。- 按名称或模式更改选择规则条件。
运行时链接	更改转换组之间链接的端口集。
排序器转换	更改排序键。
写入转换	创建参数以执行以下任务： <ul style="list-style-type: none">- 更改平面文件目标的输出文件名或目录。- 更改关系目标的连接。- 更改平面文件数据对象、自定义数据对象或关系数据对象。

相关主题：

- [“映射参数概览” 页面上 36](#)

动态源

动态源是指可在运行时更改的源。可以在映射中配置平面文件动态源或关系动态源。

可以按下列方式配置某源的动态运行时功能：

从数据源获取列。

当预期源在运行时较小更改时，可以配置读取转换以在运行时获取平面文件或关系目标列。在运行时，可以根据关系数据源或平面文件数据源的结构更新读取转换中的端口。

为确定源的平面文件名和目录而分配参数。

当平面文件源类似时，可以为文件名或目录分配参数。使用参数时，不必创建各源的数据对象。

为确定关系数据对象的资源、表所有者或目录而分配参数。

当关系源类似时，可以分配参数，以获取资源、连接和表所有者属性。

为确定用于文件源或关系源的数据对象而分配参数。

当预期源有较小更改时，可以根据关系数据源或平面文件数据源的结构，在运行时更新读取转换中的端口。

下表显示了可以在哪里配置源的动态运行时功能：

源动态运行时功能	配置
从数据源获取列。	为下列源类型配置读取转换的 数据对象 选项卡： <ul style="list-style-type: none">- 平面文件- 关系
为确定平面文件名和目录而分配参数。	为下列源类型配置物理数据对象的 高级 选项卡： <ul style="list-style-type: none">- 平面文件
为确定连接、所有者或资源而分配参数。	为下列源类型配置读取转换的 运行时 选项卡： <ul style="list-style-type: none">- 关系
为确定数据对象而分配参数。	为下列源类型配置读取转换的 数据对象 选项卡： <ul style="list-style-type: none">- 平面文件- 关系

从数据源获取列

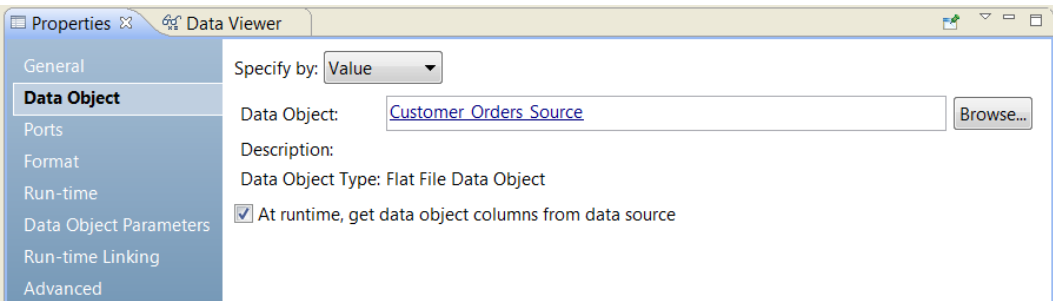
可以根据关系数据源或平面文件数据源的结构，更新运行时读取转换中的端口。可以用该方法更新运行时读取转换的实例。

当预期源有较小更改时，在运行时更新列。例如，您需要处理另一个组织的源，但该组织无法保证源文件中列的顺序。启用选项在运行时更新数据对象列时，数据集成服务会根据源数据的结构更改读取转换的端口。读取转换会将数据传递到动态映射中的下游转换进行处理。

在运行时更新数据对象列时，数据集成服务会更新运行时读取转换的实例。它不更新模型存储库中的元数据，因而您无法在 Developer tool 中看到更改。要更新模型存储库中的物理数据对象定义，请使用 Developer tool 中的同步选项。Developer tool 将重新导入物理数据对象元数据并更改元数据。

注意：如果为读取或查找转换创建自定义 SQL 查询，自定义查询的元数据可能与数据源列中刷新过的元数据冲突。如果该查询与数据集成服务将生成的默认查询不一致，映射会失败。

下图显示了在**数据对象**选项卡上启用该选项的位置：



数据集成服务会按架构确定关系源的结构，并检查**运行时**选项卡上显示的资源架构。然后，该服务会根据架构更新转换数据对象中的列。

数据集成服务根据平面文件物理数据对象配置方式确定平面文件源的结构。配置数据对象，以在运行时生成列名称。

在读取转换的**数据对象**选项卡上为平面文件源或关系源配置本功能。

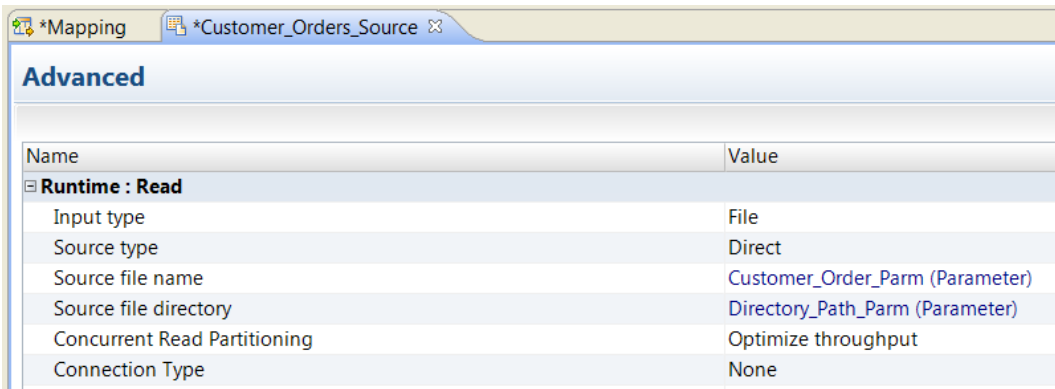
有关设置平面文件物理数据对象属性的详细信息，请参阅《*Informatica Developer Tool 指南*》。

为平面文件名分配参数

要运行具有相似平面文件源的动态映射，可以为文件名或目录分配参数。使用参数时，不必创建各源的数据对象。

可以参数化平面文件物理数据对象中的文件名和目录。可以先参数化属性，然后再从数据对象创建转换。在物理数据对象属性的高级选项卡上配置参数。从物理数据对象创建转换时，可以使用映射参数来覆盖参数默认值。

下图显示了物理数据对象的**高级**选项卡：



The screenshot shows the 'Advanced' tab of a configuration window for a physical data object. It contains a table with two columns: 'Name' and 'Value'.

Name	Value
Runtime : Read	
Input type	File
Source type	Direct
Source file name	Customer_Order_Parm (Parameter)
Source file directory	Directory_Path_Parm (Parameter)
Concurrent Read Partitioning	Optimize throughput
Connection Type	None

在物理数据对象的**高级**选项卡上为平面文件源配置本功能。

为关系源属性分配参数

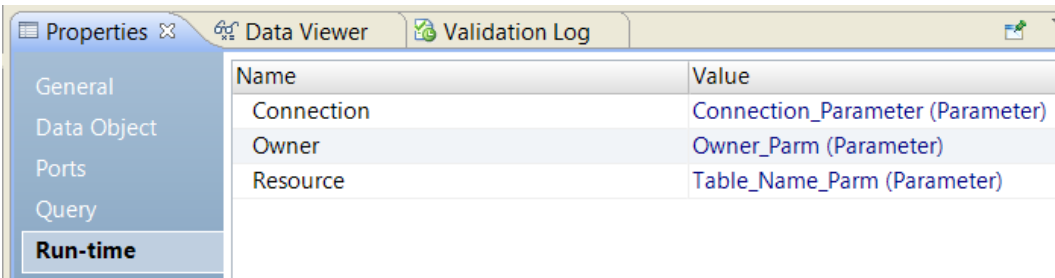
要运行具有相似关系源的动态映射，可以为参数分配读取转换中的资源、连接和表所有者属性。

使用资源参数运行相同数据库中不同但类似的表的映射。使用资源参数时，不必创建各源的数据对象。使用连接参数访问不同的数据库。针对多个关系源，可能需要运行唯一的 SQL 查询。

在转换属性的**运行时**选项卡上配置关系表参数。不能参数化关系物理数据对象中的这些属性。为读取转换中的属性创建参数时，需要创建映射参数。

默认情况下，为连接创建连接类型参数。您可分别为表名称和表所有者配置资源类型参数和字符串参数。

下图显示了读取转换的**运行时**选项卡：



The screenshot shows the 'Run-time' tab of a configuration window for a relationship source. It contains a table with two columns: 'Name' and 'Value'.

Name	Value
Connection	Connection_Parameter (Parameter)
Owner	Owner_Parm (Parameter)
Resource	Table_Name_Parm (Parameter)

在读取转换的**运行时**选项卡上为关系源配置本功能。

为源数据对象分配参数

可以为数据对象分配参数并更改读取转换运行时的源。

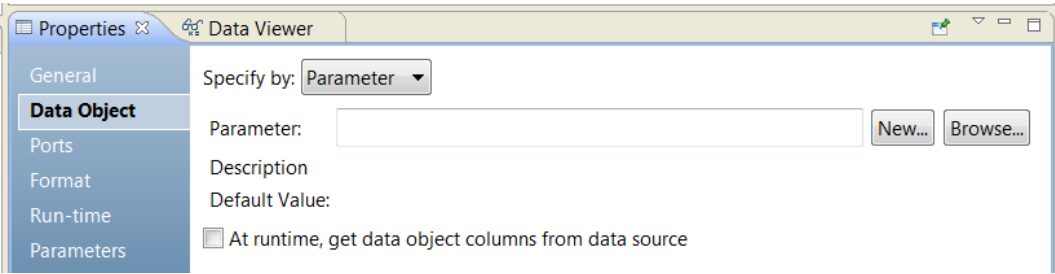
当每个数据源在模型存储库中的物理数据对象不同时，需要参数化数据对象。如果需要为平面文件或数据库表配置相同的转换，也可能需要参数化数据对象。参数化数据对象时，需使转换能够使用具有不同属性或唯一 SQL 查询的数据对象。

从某个物理数据对象创建读取转换时，转换属性的**数据对象**选项卡上会显示该数据对象的相关信息。您可以单击数据对象名称来查看模型存储库中的物理数据对象定义。

要参数化数据对象，请创建一个资源类型参数或浏览查找资源参数。参数默认值是模型存储库中物理数据对象的名称。创建默认参数值时，需从存储库中的数据对象列表中选择物理数据对象。

更改数据对象时，转换端口也会随之更改。可以查看转换属性中**端口**选项卡上的端口。

下图显示了按参数指定数据对象时出现的**数据对象**选项卡。



下表介绍了**数据对象**选项卡中的参数选项：

参数选项	说明
参数	您配置为数据对象的资源参数的名称。只读。
说明	参数的说明。只读。
新建	创建资源参数。在模型存储库中浏览参数默认值并选择数据对象。
浏览	浏览资源参数并选择该参数。
默认值	您为数据对象配置的资源参数的默认值。默认值为物理数据对象名称。只读。

在读取转换的**数据对象**选项卡上为平面文件源或关系源配置本功能。

动态目标

动态目标是指可在运行时更改的目标。可以将平面文件和关系目标配置成动态的。

运行映射时，动态目标能从物理数据目标（包括关系表、平面文件和自定义数据对象）获取元数据的更改。也可以根据上游列定义生成列。

可以按下列方式配置目标的动态运行时功能：

从数据源获取列。

当预期目标有较小更改时，可以配置写入转换，以在运行时获取关系对象列。将写入转换配置为从目标获取元数据时，可以配置写入转换进行动态更新，并保持与目标对象同步。

根据映射流定义目标列。

根据映射流定义列时，目标列由上游转换确定。

根据数据对象定义目标列。

根据数据对象定义列时，目标列由相关联的数据对象确定。

在运行时创建或替换关系目标表。

默认情况下，当您配置写入转换以在运行时创建或替换目标时，数据集成服务会根据数据对象创建目标。您也可以选择基于映射流创建目标，或者也可以定义 DDL 查询以基于该查询创建目标。

为确定关系数据对象的资源、表所有者或目录而分配参数。

当关系目标类似时，可以分配参数，以获取资源、连接和表所有者属性。

为确定用于文件源或关系目标的数据对象而分配参数。

可以将自定义数据对象创建为写入转换，并指定参数值作为转换目标。更改参数值时，使用该参数的所有对象的目标会更改。

下图显示了可以在哪里配置目标的动态运行时功能：

目标动态运行时功能	配置
从数据源获取列。	为下列目标类型配置写入转换的 数据对象 选项卡： <ul style="list-style-type: none">- 关系
根据数据对象或映射流定义目标列。	为下列目标类型配置写入转换的 端口 选项卡： <ul style="list-style-type: none">- 平面文件- 关系
在运行时创建或替换表。	为下列目标类型配置物理数据对象的 高级 选项卡： <ul style="list-style-type: none">- 关系
定义 DDL 查询以在运行时创建目标表。	为下列目标类型配置物理数据对象的 高级 选项卡： <ul style="list-style-type: none">- 关系- Hive
为确定连接、所有者或资源而分配参数。	为下列目标类型配置写入转换的 运行时 选项卡： <ul style="list-style-type: none">- 关系
为确定数据对象而分配参数。	为下列目标类型配置写入转换的 数据对象 选项卡： <ul style="list-style-type: none">- 平面文件- 关系

从数据源获取列

可以根据关系数据源的结构，更新运行时读取转换中的端口。

如果预期目标列有较小更改，请在运行时更新列。在运行时从数据源获取数据对象列时，数据集成服务会根据目标的结构创建运行时数据对象的实例。它不更新模型存储库中的元数据。

注意: 如果将写入转换配置为从数据源获取列并根据映射流定义目标，则映射会失败。

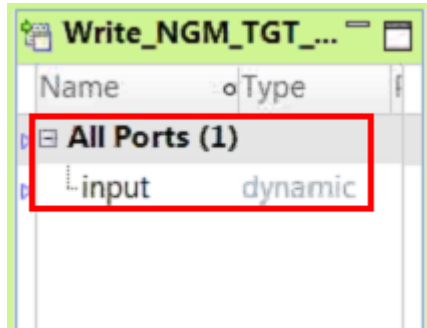
在写入转换的**数据对象**选项卡上为关系目标配置本功能。

根据映射流定义目标

根据映射流定义列时，目标列取决于上游转换。当上游转换更改端口顺序和元数据时，写入转换将提取更改。

如果上游转换中的键列与目标中的键列名称匹配，则创建或替换目标时可以保留这些键。

根据映射流定义目标列时写入转换的显示方式如下图所示：



注意：为了避免意外结果，对于请勿为根据映射流定义目标的写入转换配置运行时链接。

在写入转换的端口选项卡上为平面文件和关系目标配置本功能。

根据数据对象定义目标

可以将写入转换配置为根据关联的数据对象定义目标列。

根据数据对象定义目标列时，写入转换包含动态端口和生成的端口。

可以选择在运行时创建或替换目标。如果列名称匹配，则可以在创建或替换目标时保留目标键。可以配置规则以保证列名称匹配。

在写入转换的端口选项卡上为平面文件和关系目标配置本功能。

在运行时创建或替换目标

在运行时，数据集成服务可以创建表或丢弃并替换表。数据集成服务根据映射流或关联的数据对象创建或替换表。

配置写入转换以创建或替换目标时，数据集成服务丢弃全部与写入目标关联的现有目标表，并根据配置创建表，以使用数据对象或使用映射流。

当数据集成服务根据数据对象创建表时，表包含与数据对象中端口匹配的列。如果在运行时创建目标或用自定义数据对象替换目标，数据集成服务所创建表的名称为数据对象连接中引用的名称。

当数据集成服务根据映射流创建表时，表包含与写入转换中生成的端口匹配的列。

在数据对象的高级选项卡上配置本功能。

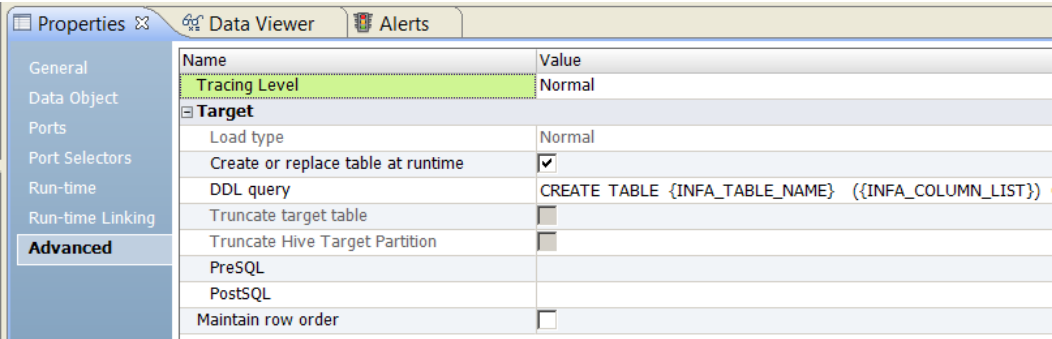
定义 DDL 查询

在运行时，数据集成服务可以基于所定义的 DDL 查询创建或替换关系目标表和 Hive 目标表。

配置写入转换以创建或替换目标时，数据集成服务在默认情况下会丢弃与写入对象关联的现有目标表。然后，数据集成服务会基于映射流或数据对象创建表。

如果要自定义表或指定其他参数（例如分区），您可以定义 DDL 查询，数据集成服务必须基于该查询创建或替换目标表。该表包含您在 DDL 查询中定义的列。

下图显示了 **DDL 查询** 字段：



可以在 DDL 查询中输入占位符。数据集成服务会在运行时将占位符替换为实际值。例如，如果表包含 50 个列，则您可以输入占位符，而不必在 DDL 查询中输入所有列名称。

可以在 DDL 查询中输入以下占位符：

INFA_TABLE_NAME

在运行时提取目标表名称。

INFA_COLUMN_LIST

在运行时提取目标表中列的列表。

INFA_PORT_SELECTOR

添加端口选择器。

必须用两个大括号将占位符括起来。例如，{INFA_TABLE_NAME}。

可以在数据对象的**高级**选项卡上配置本功能。

在运行时创建或替换目标的规则和准则

在运行时创建或替换目标时，请考虑下列规则和准则：

- 如果目标表对数据库中其他表有循环相关性，则数据库无法执行丢弃或创建表的命令，且映射失败。
- 当数据集成服务替换目标时，它不保留目标表的索引和权限。
- 如果未将写入转换配置为具有动态端口，数据集成服务会根据数据对象创建具有链接和未链接端口的目标。它将数据写入链接的端口。
- 即使数据对象中的资源同物异名或是视图，数据集成服务也会创建一个表。尽管每个连接均可指向不同的数据库实例，但动态映射中所有连接的数据库类型必须全部相同。

为关系目标属性分配参数

要运行具有相似关系目标的动态映射，可以为参数分配写入转换中的资源、连接和表所有者属性。

使用资源参数运行相同数据库中不同但类似的表的映射。使用资源参数时，不必创建各目标的数据对象。使用连接参数访问不同的数据库。

无法参数化物理数据对象中的这些属性。为写入转换中的属性创建参数时，需要创建映射参数。

默认情况下，为连接创建连接类型参数。您可分别为表名称和表所有者配置资源类型参数和字符串参数。

在写入转换的**运行时**选项卡上为关系目标配置本功能。

为目标数据对象分配参数

可以为自定义数据对象分配参数并更改写入转换运行时的源。

对于一个以上的目标数据源，当在模型存储库中有自定义数据对象时，应参数化数据对象。更改参数值时，使用该参数的所有对象的目标均会更改。

从某个自定义数据对象创建写入转换时，转换属性的**数据对象**选项卡上会显示该数据对象的相关信息。您可以单击数据对象名称，从模型存储库查看定义。要参数化数据对象，请创建一个资源类型参数或浏览查找资源参数。参数默认值是模型存储库中自定义数据对象的名称。创建默认参数值时，需要从存储库中的数据对象列表中选择自定义数据对象名称。

更改数据对象时，转换端口也会随之更改。可以查看转换属性中**端口**选项卡上的端口。

下表介绍了**数据对象**选项卡中的参数选项：

参数选项	说明
参数	您配置为数据对象的资源参数的名称。只读。
说明	参数的说明。只读。
新建	创建资源参数。在模型存储库中浏览参数默认值并选择数据对象。
浏览	浏览资源参数并选择该参数。
默认值	您为数据对象配置的资源参数的默认值。默认值为自定义数据对象名称。只读。

在写入转换的**数据对象**选项卡上为关系目标配置本功能。

动态目标的规则和准则

- 使用动态目标时，请考虑以下规则和准则：
- 预览动态目标时，Developer tool 不刷新架构定义。如果由于架构更改（例如配置为在运行时从数据源获取列或替换目标）导致不匹配，则会发生数据预览故障。手动同步读取或写入转换。如果继续出错，请运行映射以查看结果。
 - 如果动态目标对于传入数据来说太小，映射会失败，并显示消息指示值对于列来说太大。
 - 目标表中的数据类型可能与写入转换中的数据类型不同。在运行映射时，数据集成服务可能会更改上游转换与目标表之间的数据类型。

动态端口和生成的端口

可以在转换中创建动态端口，以便从上游转换接收新列或已更改的列。动态端口根据输入规则接收一列或多列并生成端口。输入规则确定动态端口接收和生成的列。

使用动态端口可执行以下任务：

接收新列和已更改的列。

为了从动态源或参数化源获取数据，需在下游转换中创建一个动态端口以接收新列和发生更改的列。如果映射包含动态源中的数据，下游转换中的动态端口则会自动获取所有新列或发生更改的列。例如，如果新列

“title” 添加到了动态源中，读取转换则会将这个新列传递到动态端口，而且动态端口会为“title” 列创建一个生成的端口。

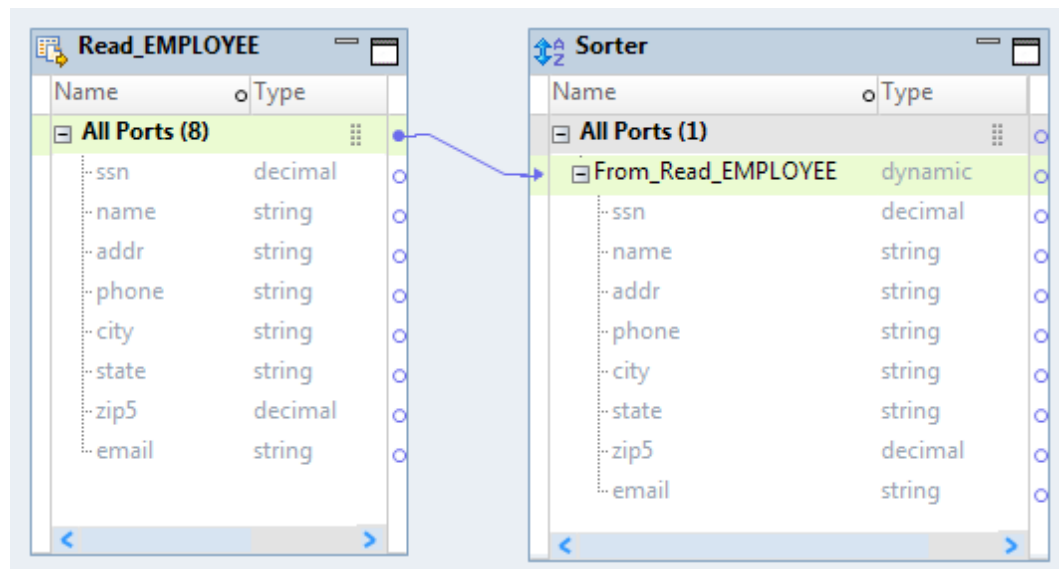
根据输入规则筛选列。

为了只处理转换中某种类型的列，需要创建一个动态端口并定义输入规则来筛选列。例如，映射源包含小数、字符串和日期/时间数据类型的列。但您只需处理小数数据类型列的数据，您可以创建一个动态端口，并定义输入规则以仅包含小数列。

针对表达式转换中的多个端口重复计算

为了对多个端口执行同一计算，需要在动态表达式中使用动态端口。动态表达式会针对动态端口中的每个端口运行一次并将结果返回到动态输出端口。

下图显示了动态端口 From_Read_Employee 和生成的端口：



动态端口和生成的端口的配置

可以从读取转换的“所有端口”组、上游转换的某个组或上游转换中的某个动态端口创建动态端口。Developer tool 将以数据类型值作为动态值来创建动态端口。可以在一个转换中创建多个动态端口。

使用**新建**按钮创建端口时，Developer tool 会分配一个默认名称。为了确保各转换中端口名称的唯一性，需要重命名动态端口。将相同名称的端口添加到转换中时，Developer tool 会向动态端口或生成的端口附加一个编号以解决端口命名冲突。

可以在下列转换中创建动态端口：

- 汇总器
- 表达式
- 筛选器
- 联接器
- 查找
- 等级
- 读取
- 路由器
- 序列生成器

排序器
更新策略
写入

如果映射所含转换无法包括动态端口，可能需要在源元数据更改时手动更新映射。

注意：端口属性的任何更改都会传播到管道中生成的端口。不必手动传播更改的端口属性。

动态端口和生成的端口的规则和准则

使用动态端口和生成的端口时，请考虑以下规则和准则：

- 无法将生成的端口链接到虚拟表映射中的输出转换。
- 无法将生成的端口链接到操作映射中的故障、输入或输出转换。

动态表达式

在动态输出端口中配置表达式时，该表达式将成为动态表达式。一个动态表达式可以生成多个输出端口。

可以在动态表达式中引用端口选择器或动态端口。端口选择器或动态端口包含多个端口时，动态表达式将针对每个端口运行。

配置动态表达式时，Developer tool 将不会验证生成的端口对于该表达式而言是否为有效类型。例如，如果在一个需要字符串类型的表达式中引用一个包含小数类型端口的端口选择器，该表达式在设计时将显示为有效。

示例

表达式转换具有以下生成的输入端口：

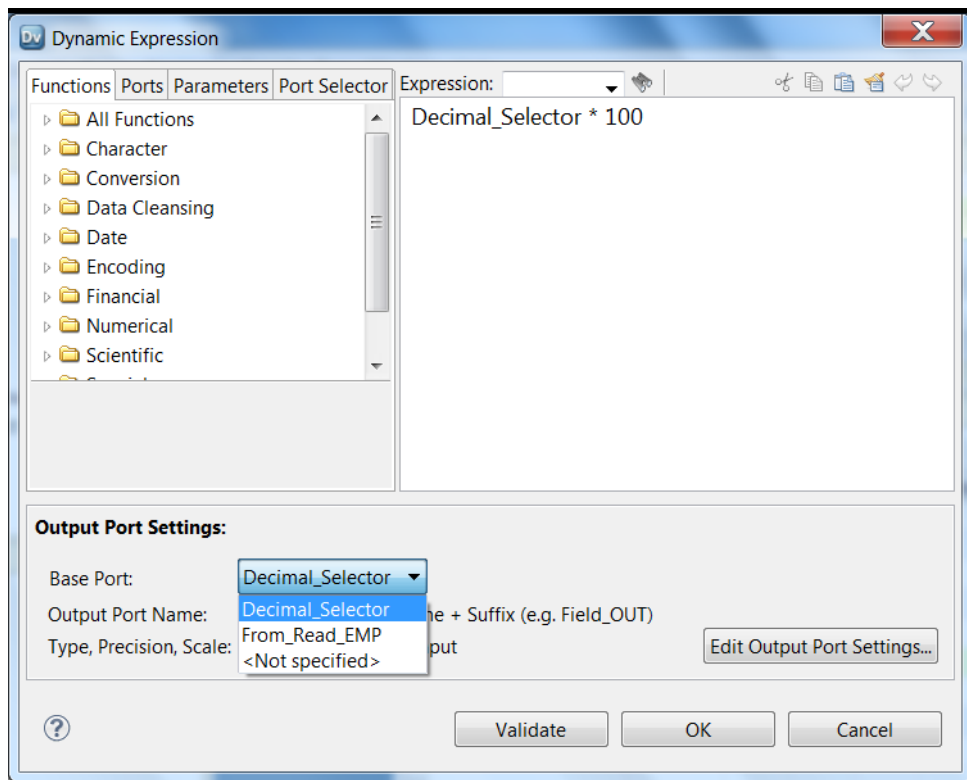
```
EMPNO    Decimal
NAME      String
SALARY    Decimal
DEPTNO    Decimal
```

该转换包含名为 MyDynamicPort 的动态输出端口。输出端口返回动态表达式的结果。动态表达式将端口选择器中每个端口的值乘以 100。该表达式为端口选择器中的每个端口分别运行一次。每个实例可以返回一个不同的结果。表达式转换为每个结果生成一个单独的输出端口。

Decimal_Selector 端口选择器具有一个包括小数数据类型端口的选择规则：

```
EMPNO    Decimal
SALARY    Decimal
DEPTNO    Decimal
```

下图显示了一个引用 Decimal_Selector 端口选择器的动态表达式：



编辑输出端口设置以更改输出端口名称和输出端口属性。您还可以选择基本端口。

输入规则

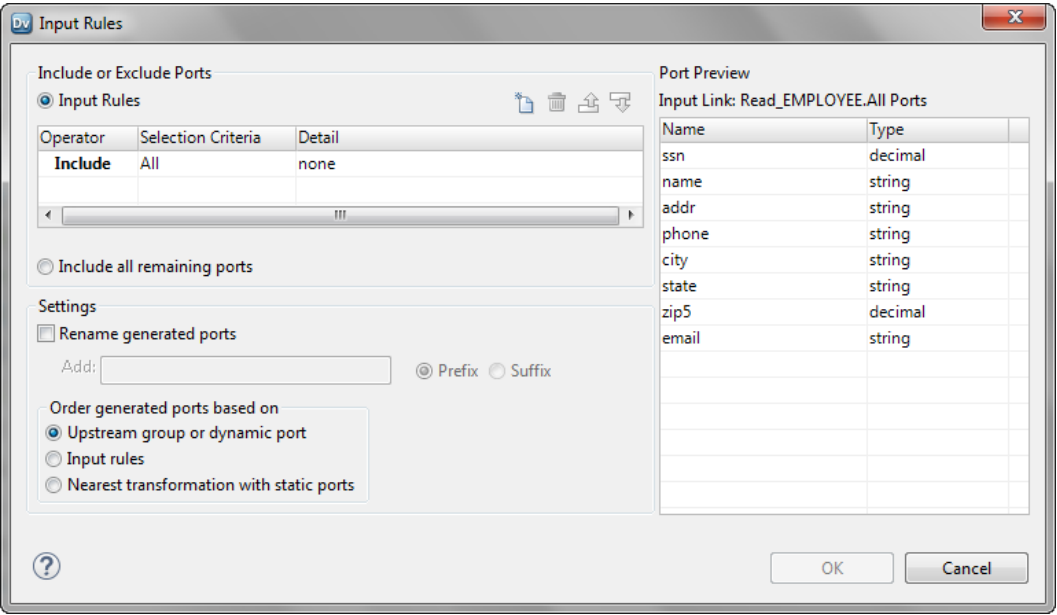
输入规则是动态端口中用于定义要生成的端口的一套条件。可以配置输入规则，以筛选动态端口所接收的列并为筛选后的列创建生成的端口。可使用输入规则在映射中向下游传播特定列。

要处理动态端口中的某些列或某种类型的列，可以定义输入规则以便根据名称或数据类型来筛选列。例如，员工数据库表包含小数、字符串和日期/时间数据类型的列。但您只需处理以 SAL 开头的小数数据类型列的数据。这种情况下，您可以创建一个动态端口，并定义输入规则以便仅包含符合此条件的列。

输入规则配置

使用**输入规则**对话框可定义要包含的端口、重命名生成的端口、更改生成的端口顺序以及查看规则结果。

下图显示了**输入规则**对话框，其默认输入规则为包含上游转换中的所有端口：



配置输入规则时，请配置下列属性：

包含或排除端口

根据端口名称或数据类型指定要在动态端口中包含或排除哪些端口。可以定义多个规则。数据集成服务会以规则在输入规则列表中显示的顺序来应用规则。默认输入规则为包含所有端口。为动态端口至少创建一个包含输入规则。

包含所有剩余端口

添加从转换的其他动态端口中排除的端口。当转换中包含多个动态端口时，可以包含最后一个动态端口的上游转换中所剩余的所有端口。

重命名生成的端口

为生成的端口名称添加前缀或后缀。使用前缀或后缀指示生成端口的转换或确保每个转换中的端口名称均唯一。

重新排序生成的端口

根据输入的规则顺序显示生成的端口。默认情况下，Developer tool 会以端口在上游转换中显示的顺序来显示端口。您还可以根据读取转换中的端口顺序来更改顺序。但是，如果一个或多个中游转换具有动态端口和静态端口，Developer tool 会按照它们在具有静态端口的最近的上游转换中出现的顺序显示端口。

配置规则后，可以预览生成的端口，以验证规则的组合情况。数据集成服务按规则在**输入规则**对话框中显示的顺序来评估规则。可以更改规则顺序，以确保它们以正确的顺序运行。

包含或排除端口

可以根据端口名称或数据类型包含或排除端口。每个输入规则均使用运算符和选择条件来筛选端口。可以定义多个规则。数据集成服务会以规则在输入规则列表中显示的顺序来应用规则。默认输入规则为包含所有端口。

配置以下输入规则设置以确定要包含或排除哪些端口：

运算符

确定是包含还是排除端口。默认设置为包含端口。

选择条件

确定是根据端口名称还是数据类型来筛选端口。选择选择条件时，根据条件会显示输入规则详细信息对话框。例如，在**输入规则详细信息: 按名称列表**对话框中提供**名称**选择条件详细信息

详细信息

确定根据您为端口名称或数据类型提供的详细信息筛选哪些端口。

下表介绍了选择条件以及如何指定条件的详细信息：

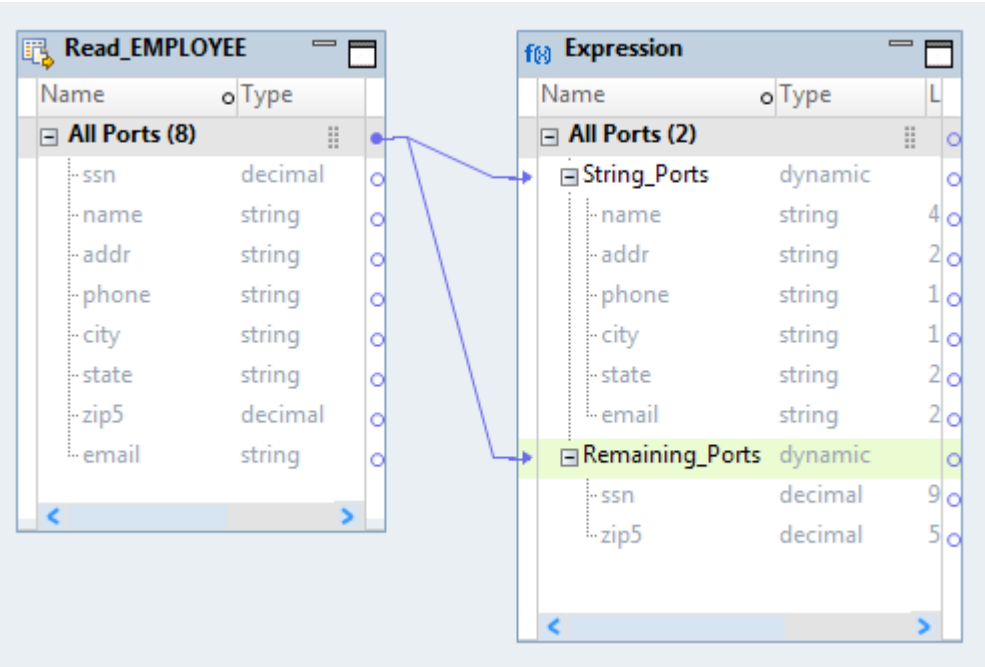
选择条件	说明	条件详细信息
全部	包括所有端口。请勿将此选择条件与排除运算符一起使用。	无需指定任何详细信息。
名称	根据端口名称筛选端口。	从值列表中选择端口名称或使用一个类型为“端口”或“端口列表”的参数。 注意: 名称值不区分大小写。
类型	根据端口的数据类型筛选端口。	从列表中选择数据类型。
模式	根据端口名称的模式筛选端口。	选择前缀、后缀或正则表达式作为端口名称的模式类型。然后，输入模式值或使用类型为“字符串”的参数。 注意: 模式值不区分大小写。

包含所有剩余端口

转换包含多个动态端口时，可以将最后一个动态端口配置为包含其他动态端口中未包含的所有端口。

例如，您要删除表中字符串列的前导空格并将字符串数据的输出与所有其他列的数据一起写入到目标。在表达式转换中，创建两个动态端口。配置输入规则，将所有字符串数据包含在一个端口中，而将所有剩余日期放入另一个端口。为最后一个动态端口选择**包含所有剩余端口**选项。

下图显示了表达式转换中的两个动态端口 String_Ports 和 Remaining_Ports:

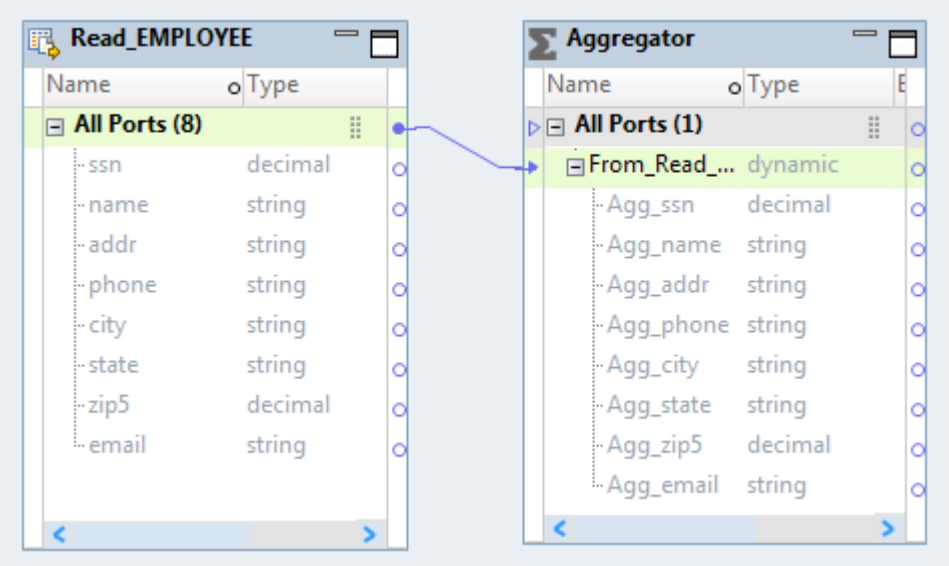


重命名生成的端口

可以通过设置生成唯一的端口名称，该设置使用前缀或后缀重命名生成的端口。

例如，可以添加 Agg_ 前缀，以指示端口在汇总器转换中生成。

下图显示了汇总器转换中使用 Agg_ 前缀进行重命名的生成的端口：



将同名的端口添加到转换中时，Developer tool 会向生成的端口附加一个编号以解决端口命名冲突。如果数据集成服务无法在运行时解决端口冲突，则可能需要重命名生成的端口。如果映射使用动态源，数据集成服务在运行

时可能遇到端口名称冲突。如果数据集成服务遇到端口名称冲突，它会尝试重命名生成的端口。如果数据集成服务无法解决端口名称冲突，映射会失败。在下列情况下，映射会失败：

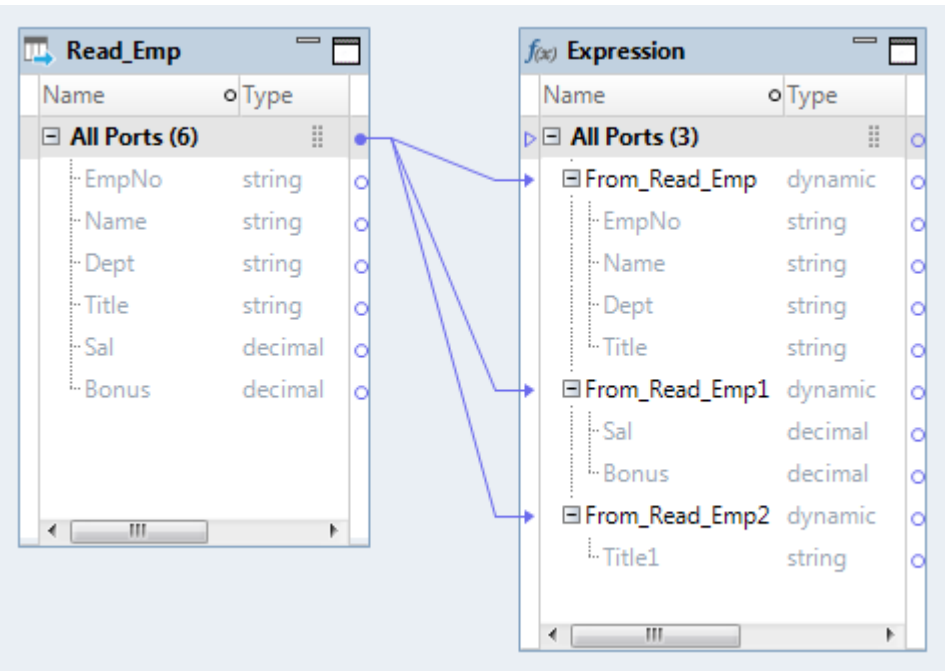
- 重命名的生成的端口与某个静态端口之间存在未解析的链接。
- 转换属性（如分组依据端口或联接条件）使用已重命名生成的端口。

要避免映射故障，请重命名生成的端口，以确保每个转换中的名称均唯一。

示例 - 重命名生成的端口

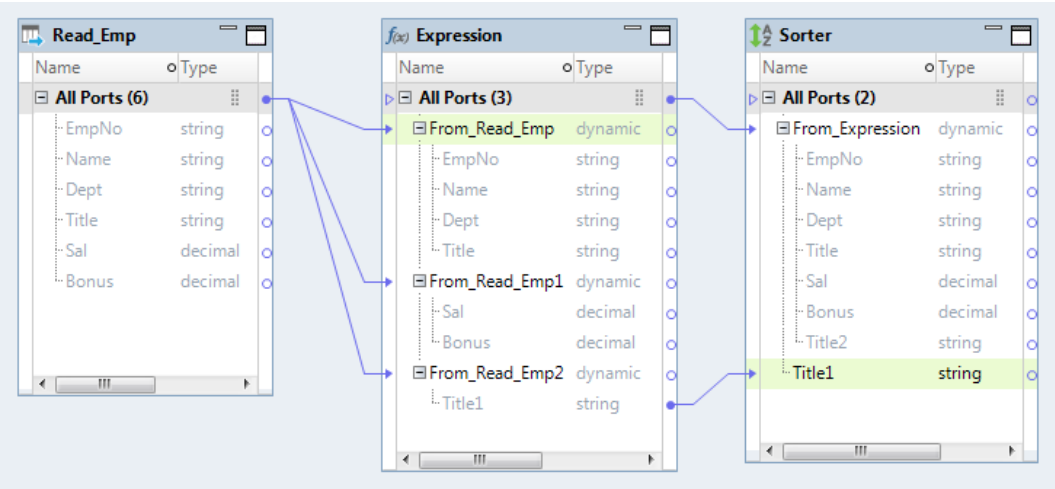
表达式转换包含三个动态端口。“From_Read_Emp”和“From_Read_Emp2”动态端口包含生成的端口“Title”。为避免名称冲突，Developer tool 会将“From_Read_Emp2”中生成的端口重命名为“Title1”。

下图显示了表达式转换中的已重命名生成的端口 Title1：



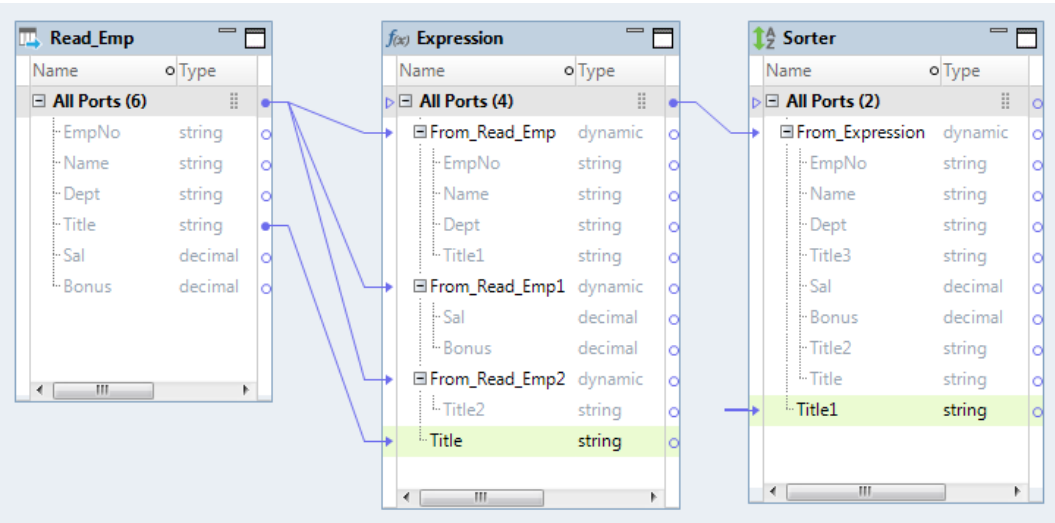
可以将表达式转换中生成的端口“Title1”链接到排序器转换中的端口“Title”，并可使用“Title1”作为排序键。

下图显示了从表达式转换中生成的端口到排序器转换中的端口的链接：



如果添加了另一个从读取转换中的端口“Title”到表达式转换中的端口“Title”的链接，Developer tool 则会重命名生成的端口。“From_Read_Emp”动态端口中生成的端口将重命名为“Title1”，“From_Read_Emp2”动态端口中生成的端口则重命名为“Title2”。排序器转换中的“Title1”链接将显示为未解析。

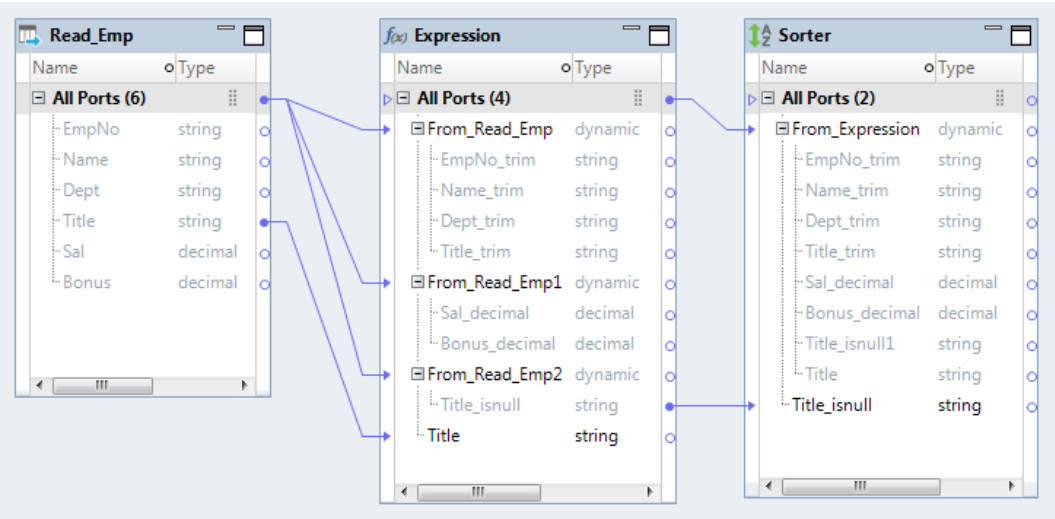
下图显示了读取转换和表达式转换之间的新链接、Developer tool 在表达式转换中重命名的生成的端口以及排序器转换的未解析链接：



在运行时映射会失败，因为用作排序键的生成的端口可能并不是要使用的端口。

要避免映射故障，请重命名生成的端口，以确保每个转换中的名称均唯一。例如，若要裁减“From_Read_Emp”动态端口的字符串端口中的前导空格，需向生成的端口添加后缀_trim。若要查找“From_Read_Emp2”动态端口中的端口是否具有空值，需向生成的端口添加后缀_isnull。

下图显示了在表达式转换中重命名的生成的端口：



重新排序生成的端口

要重新排序生成的端口，您可以使用根据输入规则顺序或读取转换中端口顺序来重新排序端口的设置。默认情况下，Developer tool 会以生成的端口在上游转换中显示的顺序来显示这些端口。

您可以选择以下选项之一来重新排序生成的端口：

上游组或动态端口

以端口在上游转换的组或动态端口中显示的顺序来显示端口。这是默认选项。

输入规则

根据动态端口的输入规则的顺序显示生成的端口。

数据集成服务按**输入规则**对话框中列出的顺序读取规则。审查端口顺序，并根据输入规则的顺序对其重新排序。可确保数据集成服务按所需顺序处理端口和规则。对端口重新排序也有助于查看和分析结果。

具有静态端口的最近转换

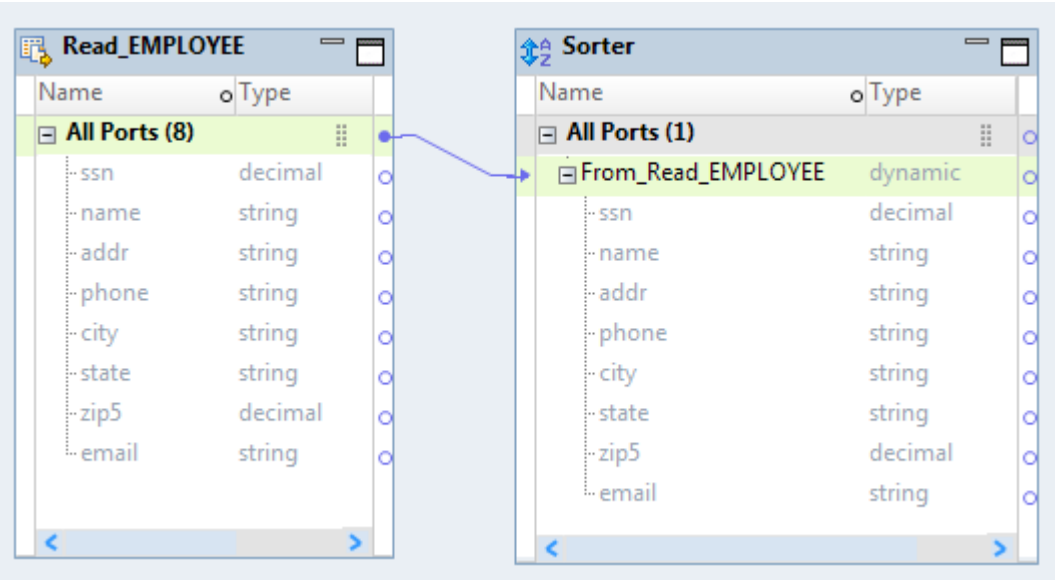
根据读取转换中的端口顺序显示生成的端口。

根据此选项重新排序端口有助于保留源中端口的原始顺序。但是，如果一个或多个中游转换具有动态端口和静态端口，Developer tool 会按照它们在具有静态端口的最近的上游转换中出现的顺序显示端口。此选项仅当映射具有单管道时有效。

示例 - 重新排序生成的端口

员工平面文件源有许多经常更改的列。您希望按姓名排序员工，并希望在查看员工数据时将员工姓名显示在第一位且后跟员工所在的城市。此外，由于您不想分析小数类型列的数据，还希望将这些列移至末尾。

下图显示了动态端口 From_Read_EMPLOYEE，其中生成的端口以原始顺序显示：

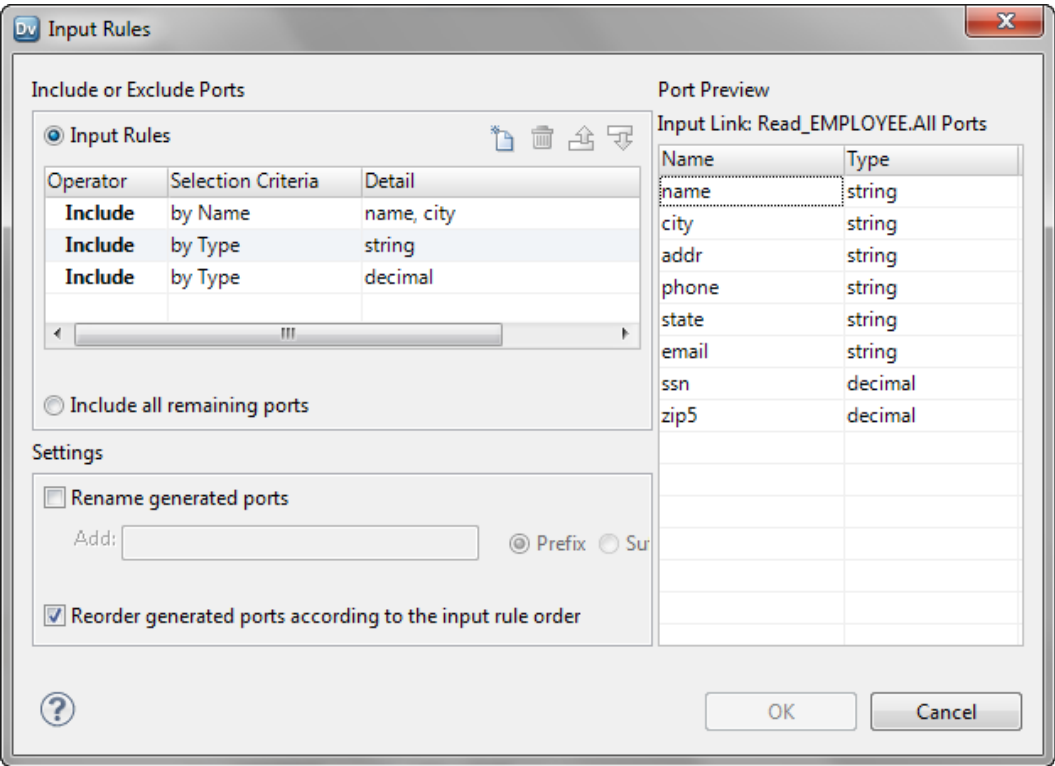


配置下列输入规则：

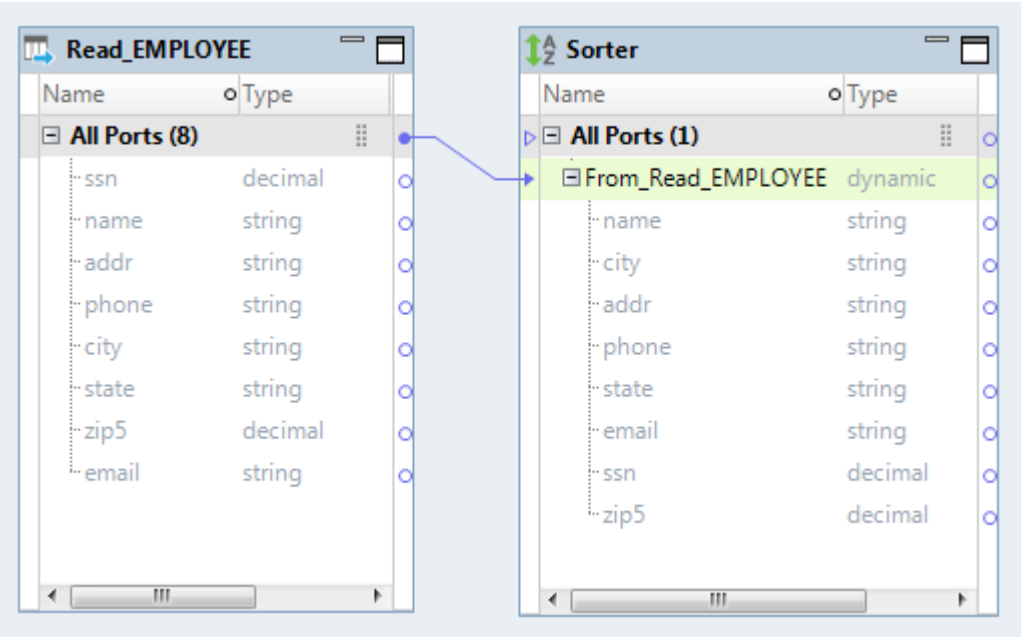
- 按“name”和“city”包含端口。
- 按字符串类型包含所有端口。
- 按小数类型包含所有端口。

然后选择根据输入规则顺序重新排序端口。要验证顺序是否正确，可以预览端口顺序。

下图显示了输入规则的顺序：



下图显示了根据输入规则设置重新排序的生成的端口。



选择规则和端口选择器

转换具有生成的端口时，您需要对该转换进行配置，以便在生成的端口发生变化时转换成功运行。您可以使用端口选择器来确定要在动态表达式、查找条件或连接器条件中使用的端口。

端口选择器为一套定义端口的选择规则。在表达式中引用端口选择器。生成的端口在动态映射中发生变化时，端口选择器可以在其中包含不同端口。您可以在表达式转换、查找转换或连接器转换中创建一个端口表达式。这些转换包含可以在端口选择器中引用所有端口的表达式。

可在以下映射对象中配置端口选择器：

表达式转换

可在动态表达式中引用端口选择器。在表达式中引用端口选择器时，表达式将对端口选择器中的每个端口运行。对于端口选择器中的每个端口，动态表达式会将结果返回到单独的输出端口。如果转换具有多个引用了端口选择器的表达式，则转换会为每个表达式返回额外的输出端口。

连接器转换

可在一个联接条件中引用两个端口选择器。分别为主组和详细信息组定义一个端口选择器。数据集成服务将根据端口选择器中的端口顺序来比较主组中的每个端口和详细信息组中的端口。可以选择一种运算符类型来比较每对端口。每个端口选择器具有的端口数必须相同。

例如，您可以配置名为 Master-SelectorX 和 Detail-SelectorY 的两个端口选择器，前者包含端口 A、B 和 C，后者包含端口 D、E 和 F。如果联接条件为 Master-SelectorX = Detail-SelectorY，则 Developer tool 会创建以下联接条件：A = D AND B = E AND C = F。

查找转换

可以配置符合查找条件的端口的端口选择器。数据集成服务将根据每个端口选择器中的端口顺序来比较输入端口选择器中的每个端口和查找端口选择器中的端口。每个端口选择器具有的端口数必须相同。

写入转换

可为写入转换中的端口配置端口选择器。

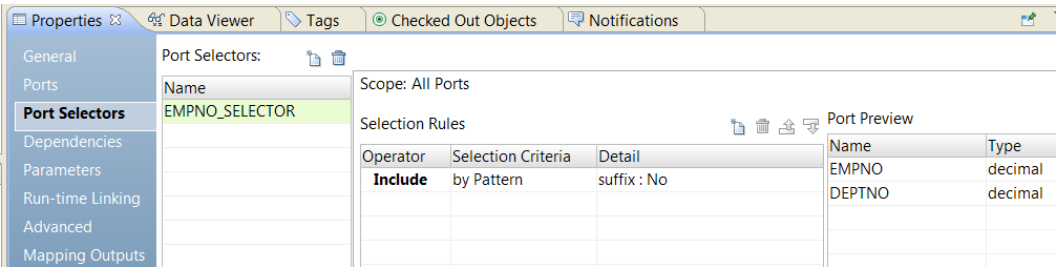
将数据写入到关系目标或 Hive 目标时，您可以选择在运行时创建或替换目标表。您可以定义 DDL 查询，数据集成服务必须在运行时基于该查询创建或替换目标表。还可以在 DDL 查询中配置端口选择器。

端口选择器配置

配置端口选择器时，可以定义选择规则来确定要包括哪些生成的端口。选择规则与您可以为动态端口配置的输入规则类似。

端口选择器可以包括静态端口或生成的端口。在**端口选择器**选项卡上配置端口选择器。

下图显示了**端口选择器**选项卡：



配置端口选择器的以下属性：

名称

标识端口选择器。可以在一个转换中创建多个端口选择器，然后在表达式中引用它们。

范围

标识要应用端口选择器的一组端口。为联接器或查找转换创建端口选择器时，必须选择范围。这些转换包含多个输入组。联接器转换具有“主”或“详细信息”范围。查找转换具有“导入”或“查找”范围。表达式转换包含一个输入组。范围始终是“所有端口”。

选择规则

确定要包括在端口选择器中的端口。创建选择规则时，**端口预览**面板显示当前输入端口中符合条件的端口。这些端口可能会发生变化。配置选择规则以接受来自不同源的端口。

选择规则

与端口选择器相关联的选择规则决定了端口选择器中要包括的端口。

创建选择规则时，**端口预览**面板显示当前输入端口中符合条件的端口。这些端口可能会发生变化。配置选择规则以接受来自不同源的端口。

请根据以下条件创建选择规则：

运算符

包括或排除选择规则返回的端口。默认为包括。必须先包括端口，然后才能排除端口。

选择条件

要创建的选择规则的类型。可根据端口类型或列名称创建规则。要根据列名称包括端口，请搜索特定名称或搜索名称中的字符模式。

详细信息

要应用到选择条件的值。如果选择条件是按列名称，则配置要搜索的字符串或名称。如果选择条件是按端口类型，则选择要包括的端口类型。

下表介绍了选择条件以及如何指定条件的详细信息：

选择条件	说明	详细信息
全部	包括所有端口。	无需详细信息。
名称	根据端口名称筛选端口。	从值列表中选择端口名称或使用一个类型为“端口”或“端口列表”的参数。
类型	根据每个端口的数据类型筛选端口。	从列表中选择数据类型。
模式	按照名称中的字符串或按照正则表达式来筛选端口。	选择前缀、后缀或正则表达式作为端口名称的模式类型。然后，输入模式值或使用类型为“字符串”的参数。

示例 - 选择规则和端口选择器

您将映射配置为使用动态源，但各源文件中含薪资信息的列有不同的名称。不同源的列名称分别是 Salary、Monthly_Salary 或 Base_Salary。

要运行带任何薪资端口名称的表达式，请执行下列任务：

- 1. 创建名为“Salary_PortSelector”的端口选择器。
- 2. 创建选择规则，以接受带“Salary”后缀的任何端口名称。
- 3. 配置该表达式以包含端口选择器名称，而非具体的列名称。该表达式使用以下语法：

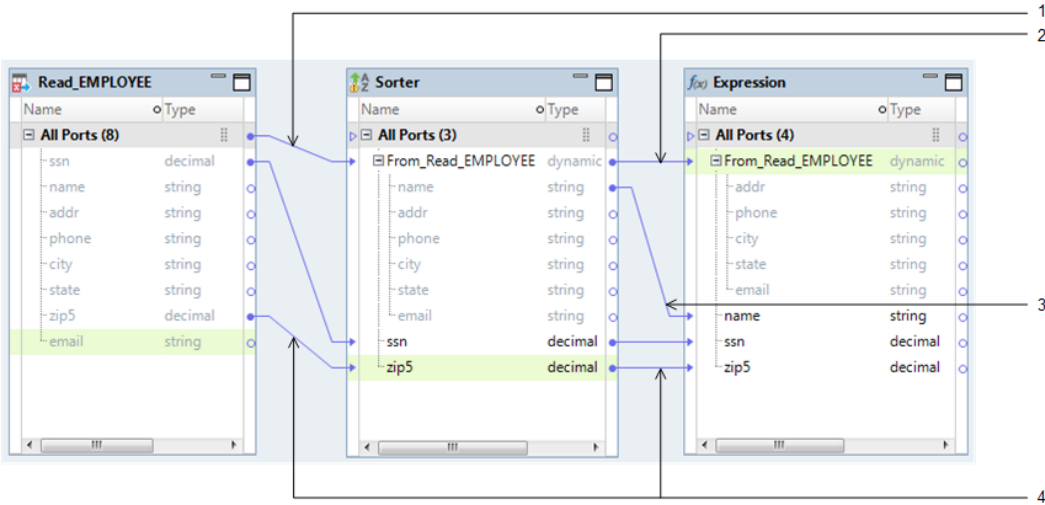
```
Salary_PortSelector * 12
```

设计时链接

设计动态映射时，可以创建不同类型的链接。可以创建两个端口之间、组和动态端口之间、两个动态端口之间或生成的端口与某个静态端口之间的链接。

设计时链接（链接）指在映射内创建的直接链接。转换可能改变，致使在设计映射时无法创建直接链接。如果在设计时无法创建链接，可以配置运行时链接，在运行时确定链接端口。

下图显示了动态映射中的链接：



1. 组和动态端口之间的链接
2. 两个动态端口之间的链接
3. 生成的端口与某个静态端口之间的链接
4. 两个端口之间的链接

设计映射时，可以创建下列类型的链接：

链接组和动态端口。

组和动态端口之间的链接可传播一列或多列数据。组可以包含一个或多个端口和动态端口。动态端口的输入规则决定了显示在动态端口下的生成的端口。默认规则是将组中的所有列均作为生成的端口包含在下游转换的动态端口中。

例如，上图显示了从读取转换中的“所有端口”组到排序器转换中的动态端口“From_Read_EMPLOYEE”的链接。排序器转换中动态端口“From_Read_EMPLOYEE”的输入规则包含字符串端口。

链接两个动态端口。

两个动态端口之间的链接可传播一列或多列数据。动态端口的输入规则决定了显示在动态端口下的生成的端口。默认规则是将上游动态端口中的所有列均作为生成的端口包含在下游转换的动态端口中。

例如，上图显示了从排序器转换中的动态端口“From_Read_EMPLOYEE”到表达式转换中的另一动态端口“From_Read_EMPLOYEE”的链接。表达式转换中动态端口的输入规则包含字符串端口并排除“name”端口。

链接生成的端口与某个静态端口。

从生成的端口到某个端口的链接可传播单列数据。

例如，上图显示了从排序器转换中动态端口“From_Read_EMPLOYEE”下生成的端口“name”到表达式转换中端口“name”的链接。

链接两个静态端口。

按照与其他映射相同的方式链接转换之间的端口。

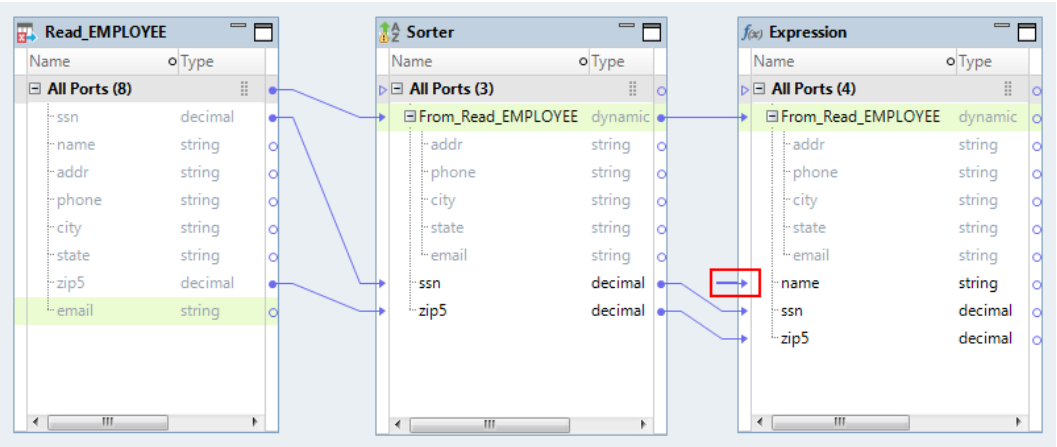
链接解析

动态端口内生成的端口可根据动态源或输入规则更改。

如果从某个生成的端口创建了一个链接且该端口不再可用，Developer tool 则会将该端口的链接显示为未解析链接。

例如，如果将排序器转换中动态端口“From_Read_EMPLOYEE”的输入规则更新为排除“name”端口，Developer tool 则会将该链接更改为未解析链接。

下图显示了表达式转换中端口“name”的未解析链接：



当您验证映射时，Developer tool 将针对未解析链接显示一条警告消息。如果生成的端口在映射运行之时可用，数据集成服务则会解析链接并处理映射。但是，如果数据集成服务无法解析链接，映射将失败。必须删除未解析链接才能成功运行映射。右键单击转换并选择**清除未解析的链接**可清除转换中任何未解析的链接。

运行时链接

运行时链接是端口可在运行时发生更改的组之间的链接。数据集成服务根据策略和参数确定运行时要链接的端口。

如果上游转换中的端口可在运行时发生更改，则需在两组映射对象之间创建运行时链接。如果端口可以在运行时更改，则设计映射时无法链接端口。创建运行时链接，该链接可使用参数和链接策略确定在运行时链接哪些端口。

在下列情况下，创建运行时链接：

读取转换配置为从数据源获取列或使用参数定义的源。

例如，读取转换在运行时使用参数更改源或从源获取元数据更改。下游转换通过一个可在两次映射运行之间发生更改的生成的端口中的端口接收数据。创建并配置下游转换的运行时链接。在运行时，数据集成服务根据链接策略或参数值连接端口。

写入转换配置为从数据源或数据对象获取列或使用参数定义的目标。

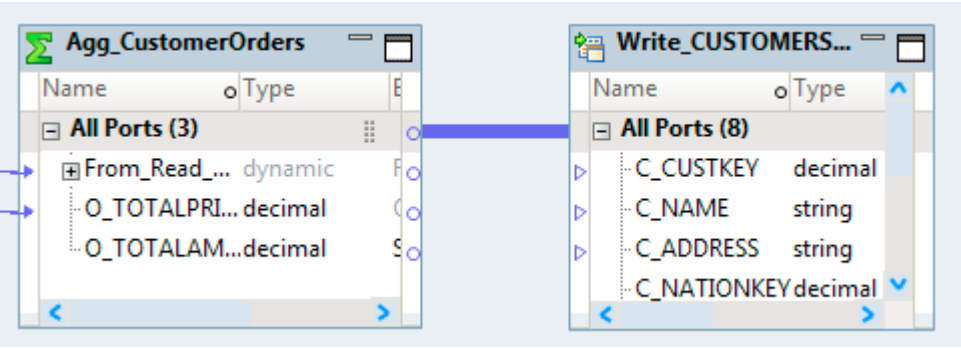
例如，写入转换可根据关联的数据对象定义列。写入转换在运行时使用参数更改目标或从目标获取元数据更改。创建并配置写入转换的运行时链接。

注意：根据映射流定义目标列时，请勿创建写入转换的运行时链接。

在运行时，数据集成服务会根据链接策略或参数值连接端口并将数据传递到下游端口。

如果上游转换中的端口可在运行时发生更改，则需在两组转换之间创建运行时链接。数据集成服务会根据您定义的参数和/或链接策略确定在运行时链接哪些端口。运行时链接在映射编辑器中以粗线显示。

下图显示了汇总器转换和写入转换之间的运行时链接：

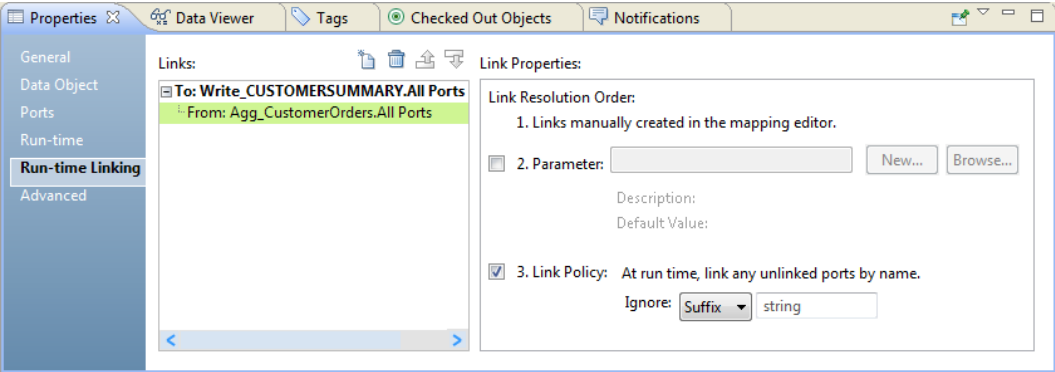


运行时链接配置

配置运行时链接属性可确定运行时要在映射对象之间链接哪些端口。可以定义一个参数或者选择链接策略，也可以同时使用二者来确定要链接的端口。

使用**运行时链接**对话框或**运行时链接**选项卡可以配置运行时链接属性。

下图显示了写入转换的**运行时链接**选项卡：



在运行时，数据集成服务按下列顺序在端口之间建立链接并对其进行解析：

1. 您在映射编辑器中手动创建的链接。
2. 基于您为运行时链接所配置的参数的链接。
3. 基于您为运行时链接所配置的链接策略的链接。

配置运行时链接的下列属性：

选择转换

在**链接**区域中，单击**新建**按钮，然后在“新链接”对话框中选择要在运行时从中链接端口的转换。**链接**区域列出了当转换包含传入运行时链接时链接源自的组端口。

配置参数

如果可在映射运行之间更改端口名称且您知道端口名称值，请使用参数。使用“输入链接集”类型的参数按名称值连接映射运行之间的端口。

例如，可创建一个名为 Cust_InputLinkSet 的新“输入链接集”类型参数并提供如下默认值：C_Name -> Cust_name。在运行时，数据集成服务在端口 C_Name 和 Cust_name 之间创建链接。可按如下方式更改运行下一映射的参数值：CustFirstName -> Cust_name

配置链接策略

链接策略按名称链接任何未链接的端口。按映射流定义目标列时，映射会传播来自源或上游对象的所有端口。使用运行时链接策略可以传播某些类型的或具有某些名称的端口。当端口具有匹配名称时，请选择链接策略。如果您希望自动链接具有匹配名称的端口，请使用链接策略。

可以将链接策略配置为忽略端口名称中的后缀或前缀。例如，如果将链接策略配置为忽略后缀“_OUT”，数据集成服务则会将 SALARY_OUT 链接至 SALARY。

根据链接组时采取的操作，链接属性显示在**运行时链接**对话框中或**属性**中的**运行时链接**选项卡上：

- 按住 Ctrl 键并将组从上游转换拖动到下游转换中的某个组可打开**运行时链接**对话框。
- 若无法创建动态端口，可以从上游转换拖动某个组。然后，在**链接**对话框中选择**创建运行时链接**以打开**运行时链接**对话框。
- 选择下游转换**属性**视图中的**运行时链接**选项卡。

示例 - 运行时链接

映射有一个可重用的汇总器转换，该转换按部门计算总薪资，并从具有生成的端口的表达式转换接收员工数据。

表达式转换可能从动态表达式生成以下输出口：

Read_EMP 源 1:

EMPNO_OUT
NAME_OUT
SALARY_OUT
DEPTNO_OUT

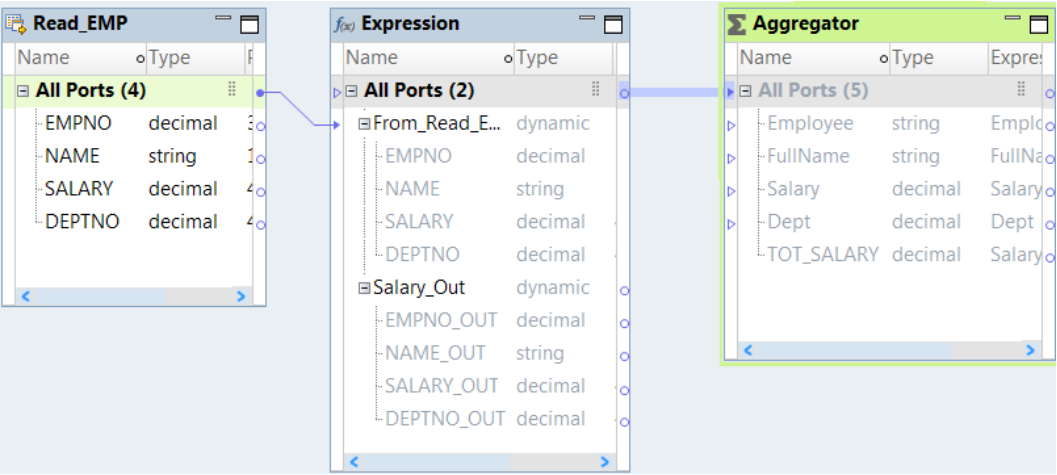
Read_EMP 源 2:

EMPNUM_OUT
FULLNAME_OUT
SALARY_OUT
DEPT_OUT

汇总器转换实例没有动态端口。

设置汇总器转换实例的运行时链接属性，以接收 EMPNO_OUT 或 EMPNUM_OUT 员工编号、NAME_OUT 或 FULLNAME_OUT 字符串、SALARY 编号以及 DEPTNO_OUT 或 DEPT_OUT 部门编号。

下图显示了表达式转换和汇总器转换之间的链接：



动态映射故障排除

在设计和测试动态映射时，请考虑以下故障排除提示：

我的映射中的动态端口包含 XML 数据类型的列，并且发生映射故障。

如果满足下列任何条件，则无法通过映射传播 XML 数据：

- 读取或写入转换配置为从数据源获取列。
- 写入转换配置为从数据流获取列。
- 目标数据对象配置为在运行时创建或更改目标。

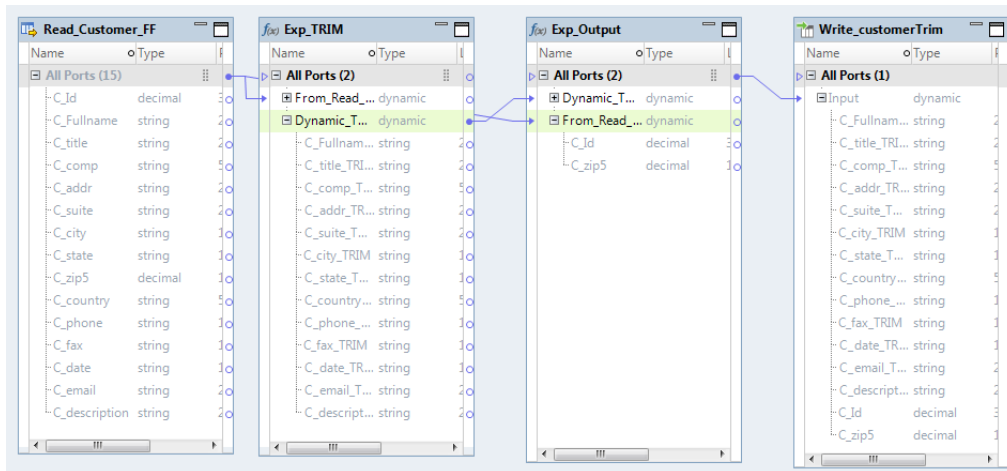
我的动态映射因参数和链接出现运行时错误而失败。我想确保在运行映射之前参数和链接解析成功。

运行动态映射时，数据集成服务会编译映射以完成下列任务：

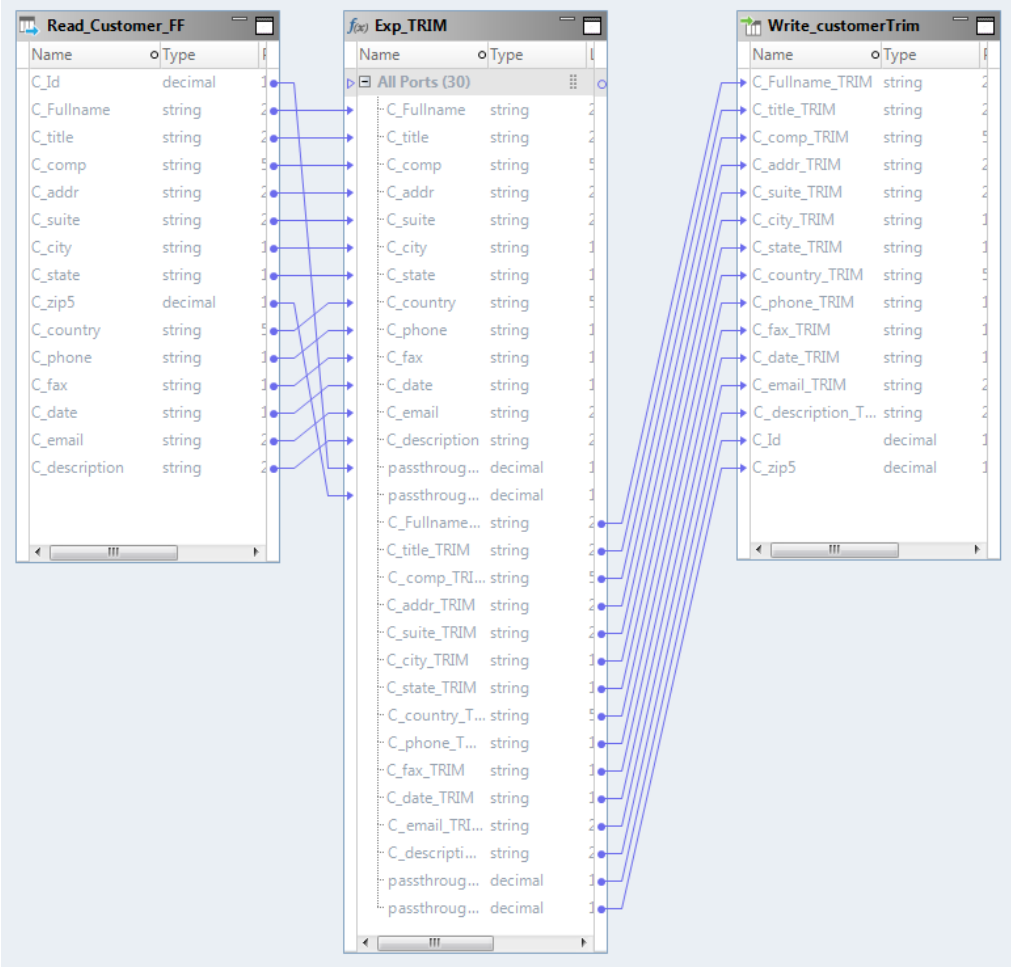
- 解析参数值。
- 展开动态端口并将生成的端口转换为静态端口。
- 链接静态端口。
- 解析运行时链接以连接端口。

可以查看优化映射以了解映射的已编译版本。右键单击映射编辑器中的空白区域，并单击**显示优化映射**。数据集成服务会生成优化映射。可以审阅优化映射，修复发现的问题，然后运行映射。

下图显示了一个所含转换具有动态端口的动态映射：



下图显示了一个动态映射的已编译版本，其中的生成端口已转换为静态端口且已链接：



第 7 章

如何开发和运行动态映射

本章包括以下主题：

- [开发和运行动态映射, 130](#)
- [配置动态源, 131](#)
- [创建动态端口, 132](#)
- [使用输入规则配置动态端口, 134](#)
- [创建端口选择器, 137](#)
- [创建动态表达式, 138](#)
- [配置动态目标, 140](#)
- [创建和配置运行时链接, 144](#)
- [验证和运行动态映射, 147](#)

开发和运行动态映射

开发动态映射可管理源的元数据更改或者针对不同的源和目标重用数据集成逻辑。针对可能存在元数据更改的相同或不同源和目标运行动态映射。

下表列出了开发和运行动态映射的高级任务。任务及其执行顺序取决于映射方案和计划在映射中使用的转换。

任务	引用
创建映射并添加映射对象。	“创建映射” 页面上 26 “将对象添加到映射” 页面上 26
为读取转换或查找转换配置动态源，以在运行时从平面文件或关系源获取元数据更改。	“配置动态源” 页面上 131
在转换中创建动态端口并链接各端口。	“创建动态端口” 页面上 132
定义动态端口的输入规则以确定要创建哪些生成的端口。 <ul style="list-style-type: none">- 定义包含或排除端口的输入规则。- 重命名生成的端口。- 或者，重新排序生成的端口。	“使用输入规则配置动态端口” 页面上 134
配置转换。	有关在映射中配置转换的详细信息，请参阅《 <i>Informatica Developer 转换指南</i> 》。

任务	引用
(可选) 创建要在联接器转换、查找转换或表达式转换的转换逻辑中使用的端口选择器。	Creating Port Selectors
(可选) 创建要在表达式转换中使用的动态表达式。	“创建动态表达式” 页面上 138
按如下方式配置要写入至动态目标的写入转换： <ul style="list-style-type: none"> - 从关联的数据对象定义列定义并从目标文件获取元数据更改，或从上游转换的映射流定义列定义。 - 在运行时针对表示关系目标的写入转换创建或替换目标表。 	“配置动态目标” 页面上 140
创建和配置运行时链接以确定在运行时链接哪些端口。	“创建和配置运行时链接” 页面上 144
确定参数在映射中的使用位置后，创建并分配参数。 <ul style="list-style-type: none"> - 将源配置为参数 - 将目标配置为参数 - 将转换属性配置为参数 	“如何配置参数” 页面上 60
验证、编译并运行动态映射。	第 7 章，“如何开发和运行动态映射” 页面上 130

配置动态源

可以将映射中的读取转换和查找转换配置为在转换源发生更改时动态更新元数据（包括列名称）。

为动态映射配置读取转换和查找转换时，可以使用以下一种或多种方法：

使用参数作为源

使用参数值作为读取或查找转换源时，需要选择一个参数名称以引用在存储库的其他位置中定义的源数据对象。

将读取转换和查找转换配置为在运行时获取元数据

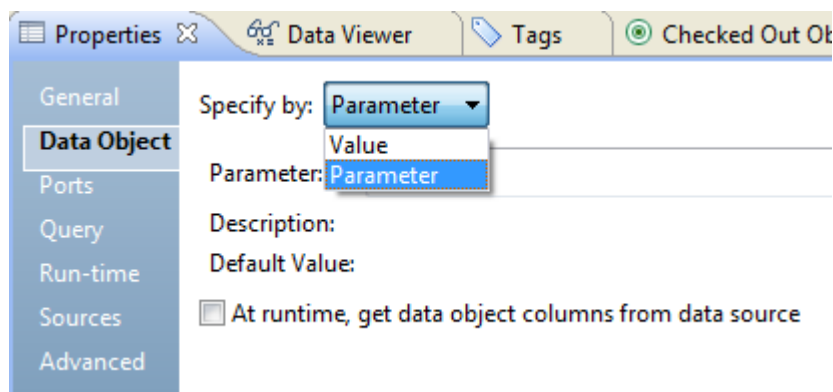
将读取转换或查找转换配置为在运行时从源获取数据对象列时，映射将在运行之时刷新端口定义。

使用参数作为动态映射源

可以使用参数作为动态映射源对象的源。

1. 在映射编辑器中，选择源对象。
2. 在属性视图中，单击**数据对象**选项卡。

3. 要针对映射运行之间的源对象使用不同的值，请在**指定依据**列表中选择**参数**。



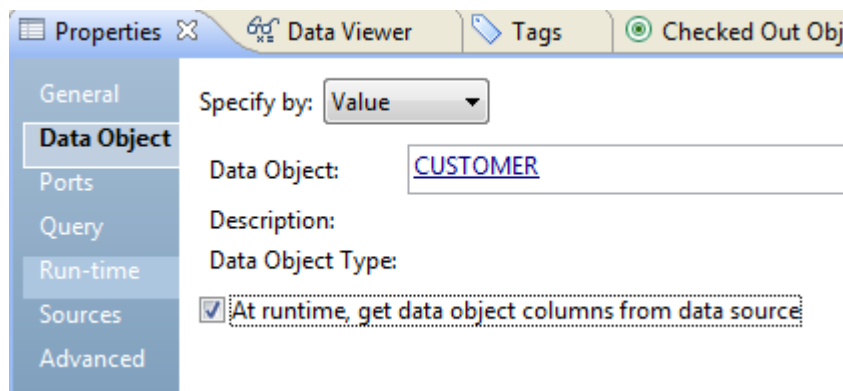
4. 单击**新建**创建一个新参数，或单击**浏览**选择现有参数。

配置源以在运行时获取元数据更改

可以为映射中的源对象配置数据源，以在运行时获取元数据更改。

如果在开发映射之后数据源列的元数据发生了更改，则映射可能会过期。配置数据源时可以使用在映射运行时获取此数据的选项。

1. 在映射编辑器中，选择源对象。
2. 在**属性**视图中，单击**数据对象**选项卡。
3. 要在运行时从数据源文件动态获取列，请选择**在运行时，从数据源获取数据对象列**。



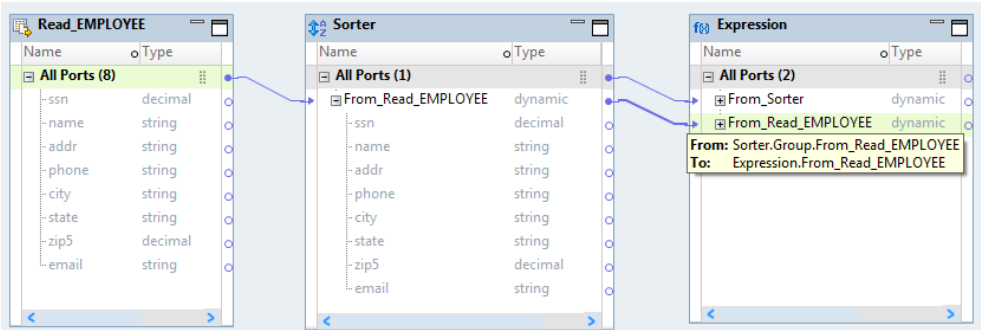
创建动态端口

创建动态端口可从上游转换接收多个列。这些列可在运行时更改。可以在一个转换中创建多个动态端口。

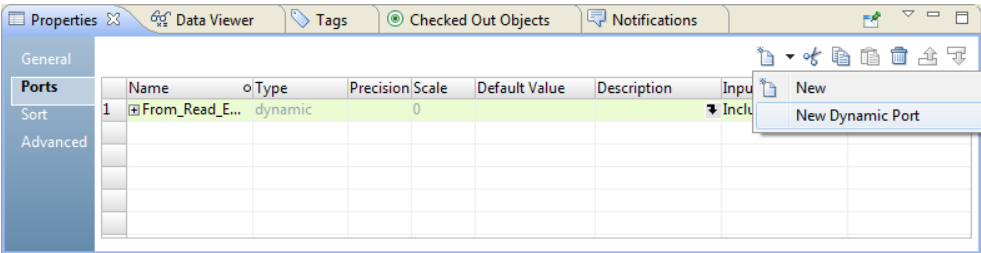
1. 请按照以下方法创建动态端口：
 - 拖动“所有端口”组或来自另一转换的动态端口。

Developer tool 会为上游转换中的所有列创建具有生成端口的动态端口并链接这些端口。可以通过更改输入规则筛选生成的端口。

下图显示了排序器转换和表达式转换中的动态端口：

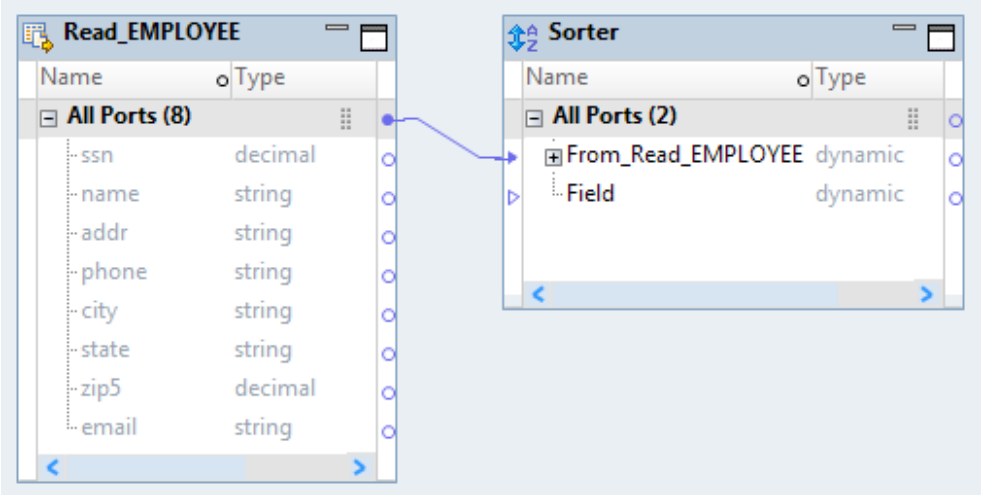


- 在转换的属性视图中，选择端口选项卡上的新建动态端口。



Developer tool 会创建一个可配置的空动态端口。必须手动链接端口才能创建生成的端口。

下图显示了排序器转换中无生成端口的新动态端口：



- 也可以更改动态端口名称并添加端口说明。
生成的端口将从上游转换继承端口属性且不可编辑。

使用输入规则配置动态端口

可以定义输入规则，以指定通过管道中的动态端口生成并传播哪些端口。

使用**输入规则**对话框可定义动态端口的输入规则、重命名生成的端口以指示端口在映射中的位置、更改生成端口的顺序以及查看规则结果。可以添加多个用于包含和排除端口的规则。数据集成服务会按规则在列表中的显示顺序来应用规则。

1. 打开“输入规则”对话框。
2. 为转换中的每个动态端口定义一个或多个输入规则。对于每个输入规则，请执行以下步骤：
 - a. 为输入规则选择运算符和选择条件。
 - b. 如果选择“名称”选择条件，请按名称或参数指定条件详细信息。
 - c. 如果选择“类型”选择条件，请从列表中选择端口的数据类型。
 - d. 如果选择“模式”选择条件，请选择模式类型并指定模式字符串作为值或参数。或者，为转换中的最后一个动态端口定义输入规则以包含上游转换中的所有剩余端口。
3. 重命名生成的端口。
4. （可选）重新排序生成的端口。
5. 验证输入规则结果和输入规则设置。

步骤 1. 打开“输入规则”对话框

打开**输入规则**对话框以定义或编辑输入规则。

- ▶ 通过以下方式打开**输入规则**对话框：
 - 右键单击转换中的动态端口，然后选择**编辑输入规则**。
 - 在转换的**端口**选项卡上，单击动态端口的**输入规则**。
- 此时将显示**输入规则**对话框，其中的默认输入规则为“包含所有”。

步骤 2. 定义输入规则

定义输入规则以包含或排除动态端口从上游转换接收的端口。

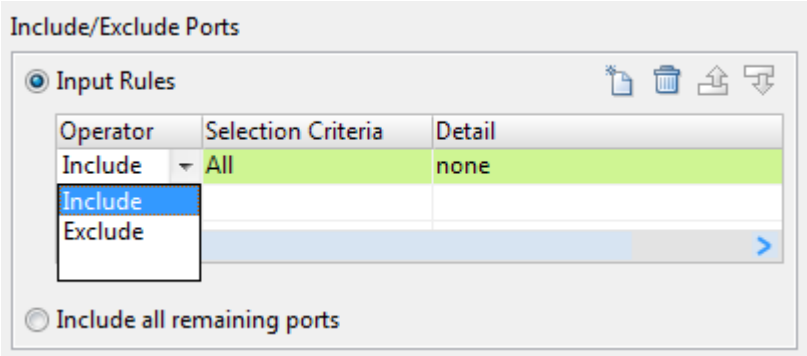
1. 在**包含或排除端口**区域中，选择**输入规则**。
2. 要编辑默认输入规则，请执行以下步骤：
 - a. 为输入规则选择运算符和选择条件。
 - b. 配置选择条件详细信息。
3. （可选）按规则在 Developer tool 中的预期运行顺序添加多个输入规则。
 - a. 单击**新建**为输入规则添加一个新行。
 - b. 对于每个输入规则，请选择运算符和选择条件并指定条件详细信息。
4. 要仅包含上游转换中的剩余端口，请执行以下步骤：
 - a. 创建另一个动态端口或选择转换中的最后一个动态端口。
 - b. 选择**包含所有剩余端口**。此规则包含上游转换中不属于其他动态端口的端口。

步骤 2a. 选择运算符和选择条件

选择运算符和选择条件，前者用于包含或排除端口，后者用于根据端口名称或数据类型筛选端口。

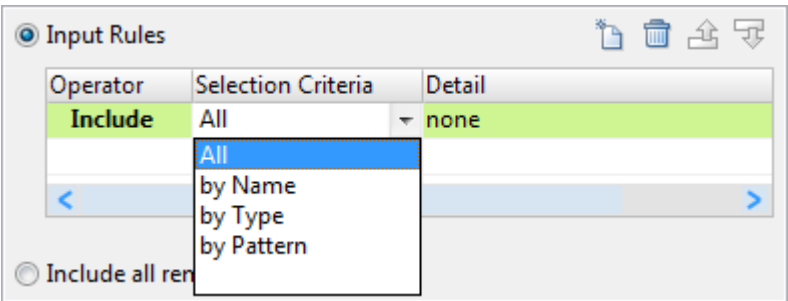
1. 从**运算符**列中，选择**包含**或**排除**运算符。

运算符用于确定规则是必须包含还是排除端口。



2. 在**选择条件**列中，选择下列选项之一：

- **全部**。包括所有端口。请勿将此选项与“排除”运算符一起使用。
- **名称**。根据端口名称包含或排除端口。
- **类型**。根据端口的数据类型包含或排除端口。
- **模式**。根据端口名称的模式包含或排除端口。



3. 在**详细信息**列中，单击“详细信息”箭头以提供选择条件详细信息。

此时将显示选择条件的**输入条件详细信息**对话框。

步骤 2b. 配置名称选择条件详细信息

如果为输入规则选择了名称选择条件，请从值列表中选择端口名称。或者，使用端口或端口列表类型的参数指定可在运行时更改的端口名称。

1. 在**输入规则详细信息:按名称**对话框中，从**指定依据**列表中选择下列项之一：

- **值**。输入端口名称，或从列表中选择端口名称。
- **参数**。创建一个新参数或选择“端口列表”类型的现有参数。

2. 通过以下方法之一指定端口名称值：

- 在**名称**框中输入端口名称，然后单击**添加**。
- 单击**选择**，在**端口**对话框中选择端口名称，然后单击**确定**。

3. 要为端口名称创建一个新参数，请执行以下步骤：
 - a. 单击**新建**。
 - b. 在**参数**对话框中，输入参数名称。
 - c. （可选）输入参数说明。
 - d. 输入端口名称参数的默认值。也可以单击**选择从端口列表中选择端口名称**。
 - e. 单击**确定**。
4. 要为端口名称选择一个现有参数，请执行以下步骤：
 - a. 单击**浏览**。
 - b. 在**分配参数**对话框中，选择一个参数。
 - c. （可选）从此对话框创建新参数或编辑参数。
 - d. 单击**确定**。

步骤 2c. 配置类型选择条件详细信息

如果为输入规则选择了类型选择条件，请从数据类型列表中选择类型。

1. 在**输入规则详细信息:按类型**对话框中，从列表中选择数据类型。
2. 单击**确定**。

步骤 2d. 配置模式选择条件详细信息

如果为输入规则选择了模式选择条件，请为端口名称选择模式类型。输入模式值或使用“字符串”类型的参数指定可在运行时更改的值。

1. 在**输入规则详细信息:按模式**对话框中，从**模式类型**列表中选择下列项之一：
 - **前缀**。包含或排除以前缀字符串开头的端口名称。
例如，如果输入了前缀值 E，则输入规则将筛选以 E 或 e 开头的端口名称，如 EmpNo、empName 和 EmpTitle。
 - **后缀**。包含或排除以后缀字符串结尾的端口名称。
例如，如果输入了后缀值 E，则输入规则将筛选以 E 或 e 结尾的端口名称，如 empname 和 EMPTITLE。
 - **正则表达式**。包含或排除遵循某种模式的端口名称。
例如，如果输入了值 E.*No，则输入规则将筛选以 E 开头且以 No 结尾的端口名称，如 ENo、EmpNo 和 EmployeeNo。
2. 从**指定依据**列表中选择下列项之一：
 - **值**。为模式输入字符串值。
 - **参数**。创建一个新参数或选择字符串类型的现有参数。
3. 在**字符串框**中指定模式值，然后单击**确定**。
4. 要为模式创建一个新参数，请执行以下步骤：
 - a. 单击**新建**。
 - b. 在“参数”对话框中，输入参数名称。
 - c. （可选）输入参数说明。
 - d. 输入模式的默认值并输入精度值。
 - e. 单击**确定**。

5. 要为端口名称选择一个现有参数，请执行以下步骤：
 - a. 单击**浏览**。
 - b. 在“分配参数”对话框中，选择一个参数。
 - c. （可选）从此对话框创建新参数或编辑参数。
 - d. 单击**确定**。

步骤 3. 重命名生成的端口

重命名生成的端口，以确保转换中的端口名称唯一。

1. 在**设置**区域中，选择**重命名端口**。
2. 选择是添加前缀还是后缀来重命名生成的端口。
3. 为生成的端口添加前缀或后缀文本。

重命名的端口将显示在**端口预览**区域中。

步骤 4. 重新排序生成的端口

重新排序生成的端口以有效地查看和分析结果。

- ▶ 在**设置**区域中，选择**根据输入规则顺序重新排序生成的端口**。

重新排序的端口将显示在**端口预览**区域中。生成的端口将按输入规则顺序（而非其在上游转换中的显示顺序）来显示。

步骤 5. 验证动态端口配置

根据您为动态端口定义的规则和设置查看生成的端口。

- ▶ 在**输入规则**对话框的**端口预览**区域中，验证动态端口的输入规则设置结果。

创建端口选择器

使用端口选择器可确定要在动态表达式、查找条件或联接器条件中使用的端口。

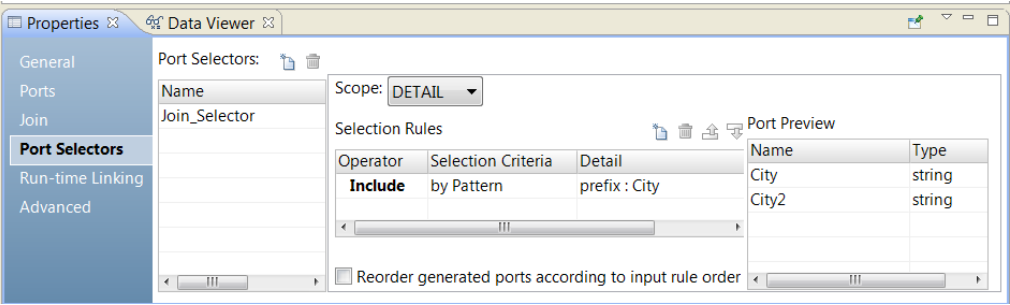
1. 单击**端口选择器**选项卡。
2. 在**端口选择器**区域中，单击**新建**。

Developer tool 使用包含所有端口的默认选择规则来创建端口选择器。
3. 在**端口选择器**区域内，将端口选择器名称更改为唯一名称。
4. 如果您要使用联接器转换或查找转换，请选择范围。

可用端口会根据所选的端口组发生相应变化。
5. 在**选择规则**区域内，选择**运算符**。
 - 包含。创建一个包含端口选择器端口的规则。必须先包括端口，然后才能排除端口。
 - 排除。创建一个在端口选择器中排除特定端口的规则。
6. 选择**选择条件**。

- 按名称。按名称选择特定端口。您可以在范围内从端口列表中选择端口名称。
- 按类型。按类型选择端口。可以选择一个或多个数据类型。
- 按模式。按端口名称中的字符模式选择端口。可以使用特定字符进行搜索，也可以创建正则表达式。

下图显示了“端口选择器”选项卡：



7. 单击**详细信息**列。
输入规则详细信息对话框随即打开。
8. 选择作为端口筛选依据的值。
 - 按名称。选择按值或参数来创建端口列表。单击**选择**在列表中选择端口。
 - 按类型。从列表中选择一个或多个数据类型。**端口预览**区域显示所选类型的端口。
 - 按模式。选择在端口名称的前缀或后缀中搜索特定字符模式。或者，选择创建在搜索时使用的正则表达式。配置参数或者配置在搜索时使用的模式。

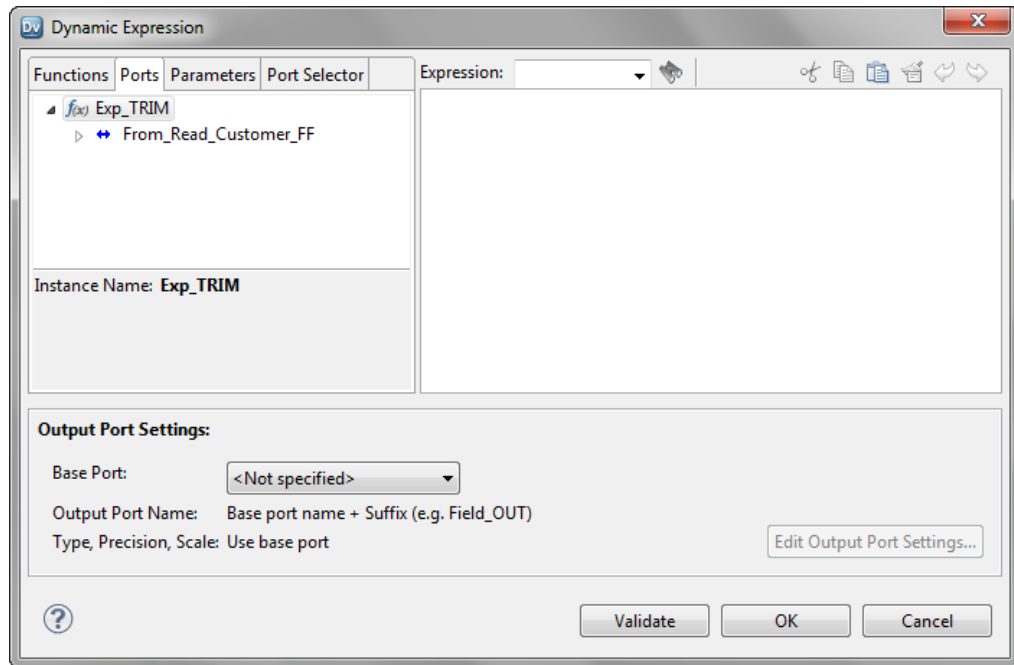
配置规则时，**端口预览**区域会显示端口选择器中的端口。
9. 要重新排序端口选择器中的端口，请选择**根据输入规则顺序重新排序生成的端口**。

创建动态表达式

在表达式转换中创建动态表达式，以便针对动态端口或端口选择器中的每个端口运行一次该表达式。动态表达式将结果返回到每个实例的单独生成的端口。

1. 在表达式转换中，转到**属性**视图，然后单击**端口**选项卡。
2. 单击**新建动态端口**。
Developer tool 使用默认属性创建动态端口。
3. 重命名该动态端口并禁用输入选项。
动态端口必须为输出口。
4. 在动态输出端口的**表达式**列中，单击**打开按钮** (🔗)。

此时将显示**动态表达式**对话框：

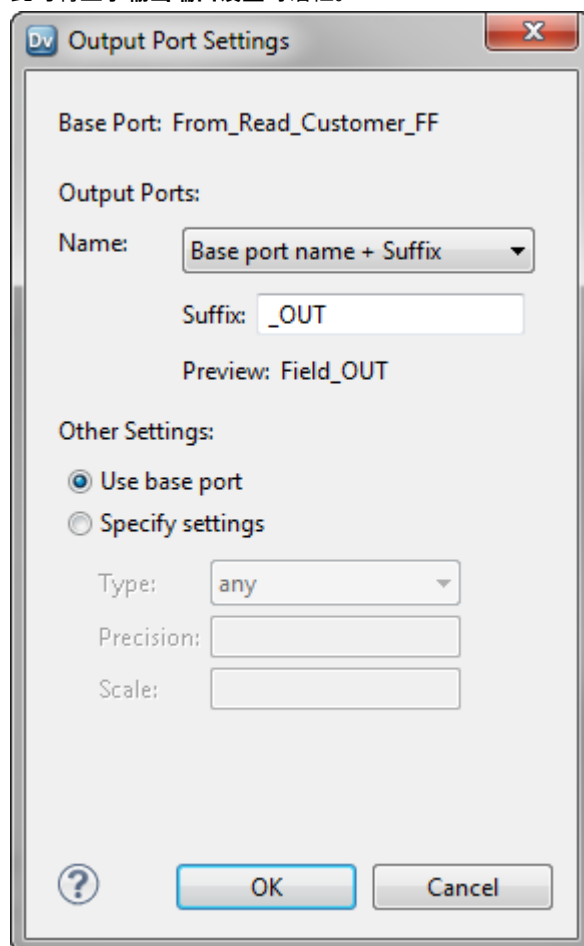


5. 在表达式编辑器中，输入表达式。该表达式可以包含端口选择器或动态端口。
例如 LTRIM(RTRIM(Dynamic_Customer))，其中 Dynamic_Customer 是动态端口。
6. 单击**验证**以验证表达式。
7. 单击**确定**退出**验证表达式**对话框。
8. 在**输出端口**设置区域中，从**基本端口**列表中选择动态输出端口，或者选择在表达式中引用的端口选择器。
Developer tool 根据您的选择内容生成输出端口。

9. 按照以下步骤重命名输出端口：

- a. 单击**编辑输出端口设置**。

此时将显示**输出端口设置**对话框。



- b. 在**名称**列表中，选择一个选项并输入前缀或后缀的值。如果已选择**固定字符串 + 自动编号**，请输入输出端口名称的文本。例如，如果您为输出端口名称输入 TRIM，输出端口名称将显示为 TRIM1、TRIM2、TRIM3。
- c. 或者，在**其他设置**区域中选择**指定设置**以更改输出端口的类型、精度和小数位数。默认情况下，输出端口使用基本端口的设置。
- d. 单击**确定**。

10. 单击**确定**退出**动态表达式**编辑器。

配置动态目标

可以将写入转换配置为在运行时当目标元数据更改时从目标接收列。或者，将参数指定为目标数据对象以便分配不同的值。也可以指定写入转换针对端口定义是使用关联对象还是使用映射流。

为动态映射配置写入转换时，可以使用以下一种或多种方法：

使用参数作为目标

将参数指定为目标的基础数据对象，以便通过参数更改写入转换的架构。

在运行时从目标获取数据对象列

启用在运行时从目标获取数据对象列的选项，以便使用目标架构中的更改动态更新写入转换端口。

定义 DDL 查询以在运行时创建或替换目标

如果选择在运行时创建或替换目标，则您可以定义 DDL 查询以基于所定义的查询创建目标。可以为关系目标和 Hive 目标定义 DDL 查询。

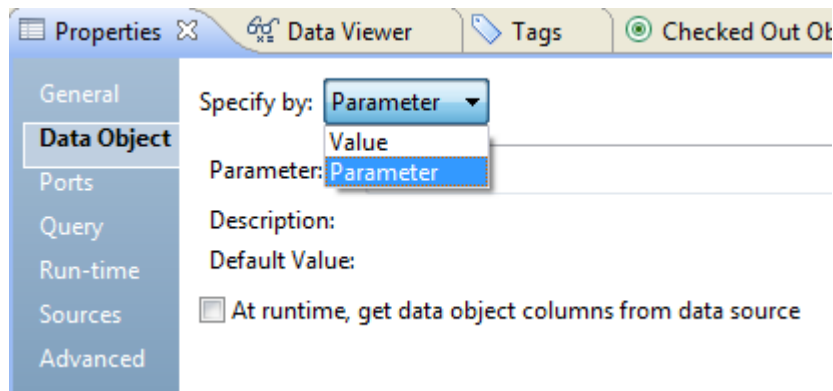
从映射流定义写入转换端口

选择从映射流定义端口时，数据集成服务会根据上游列定义来定义写入转换端口。目标列将在运行时动态更新。

使用参数作为动态映射目标

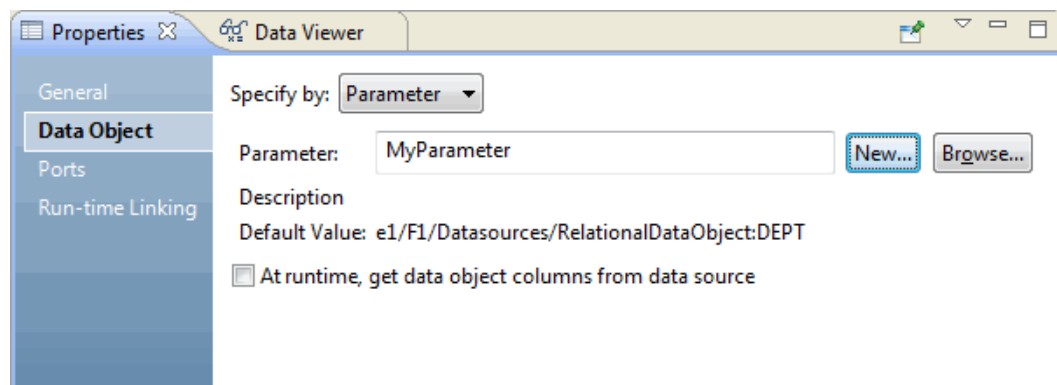
可以使用参数作为转换的数据对象，然后在运行时更改参数。

1. 在映射编辑器中选择写入转换。
2. 在**属性**视图中，单击**数据对象**选项卡。
3. 在**指定依据**列表中选择**参数**。



4. 选择以下选项之一：
 - 单击**新建**创建一个参数。命名参数，然后浏览并选择参数的默认值。
 - 单击**浏览**以选择现有参数。

下图显示了使用参数作为数据源的转换：

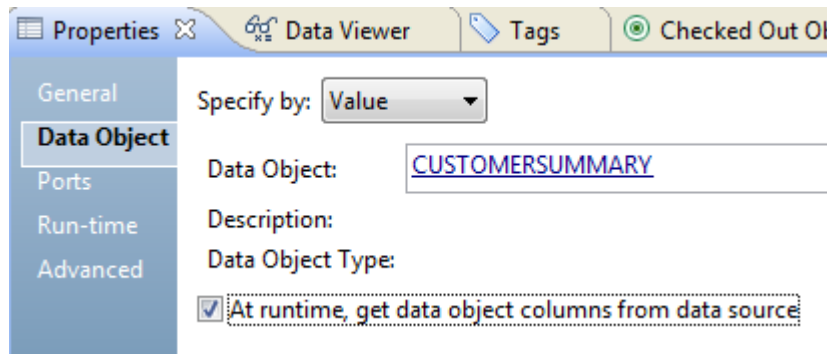


在运行时从数据源获取目标对象列

可以启用在运行时从数据源获取数据对象列的选项。

选择在运行时从数据源获取数据对象列的选项时，映射会在运行时将数据对象列提取到转换。如果数据源列和元数据发生更改，则映射会提取更改的信息。

1. 在**属性**视图中，单击**数据对象**选项卡。
2. 选择**在运行时，从数据源获取数据对象列**。



定义 DDL 查询以在运行时创建或替换目标

如果选择在运行时创建或替换目标，则您可以定义 DDL 查询，数据集成服务必须在运行时基于该查询创建或替换目标表。可以为关系目标和 Hive 目标定义 DDL 查询。可以在 DDL 查询中输入占位符和参数。

1. 在**属性**视图中，单击**高级**选项卡。
2. 选择**在运行时创建或替换表**选项。

DDL 查询字段可用。

3. 单击**编辑**。

此时将显示 **DDL 查询**对话框。

4. 在编辑器中输入 DDL 查询。

可以在 DDL 查询中输入占位符。数据集成服务会在运行时将占位符替换为实际值。例如，如果表包含 50 个列，则您可以输入占位符，而不必在 DDL 查询中输入所有列名称。

可以在 DDL 查询中输入以下占位符：

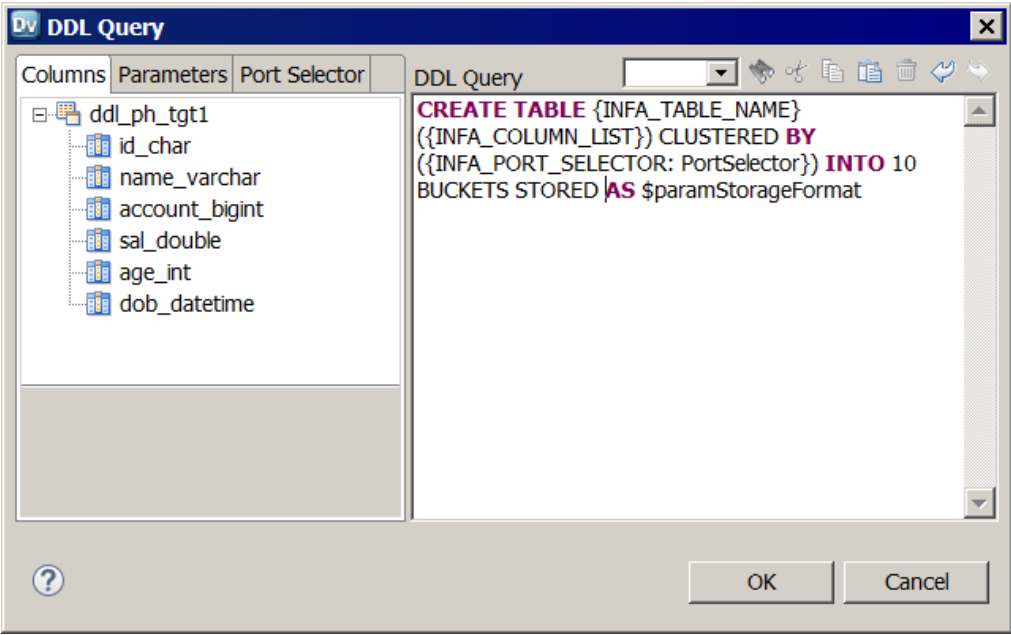
- INFA_TABLE_NAME。在运行时提取目标表名称。
- INFA_COLUMN_LIST。在运行时提取目标表中列的列表。
- INFA_PORT_SELECTOR。添加端口选择器。

注意：占位符名称区分大小写。必须用两个大括号将占位符括起来。例如，{INFA_TABLE_NAME}。

还可以执行以下步骤以定义 DDL 查询。

- 要添加列名称，请在**列**选项卡中双击一个列。
- 要定义参数，请单击**参数**选项卡，然后双击一个参数名称。也可以单击**管理参数**以添加、编辑或删除参数。
- 要配置端口选择器，请单击**端口选择器**选项卡，然后双击一个端口选择器。也可以单击**新建**以配置新的端口选择器。

下图显示了用于创建 Hive 目标表的 DDL 查询：



该图中的 DDL 查询包含 INFA_TABLE_NAME、INFA_COLUMN_LIST 和 INFA_PORT_SELECTOR 占位符。该查询还包含一个用于定义存储格式的参数。

如果您不输入 DDL 查询，则数据集成服务会基于映射流或数据对象创建目标。

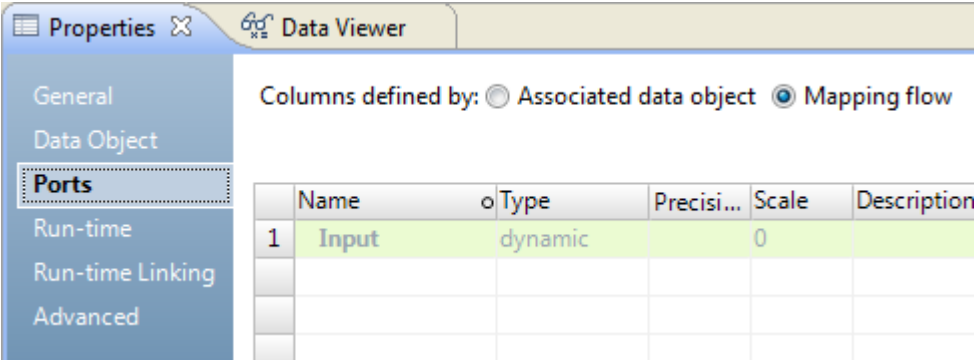
5. 单击**确定**以保存 DDL 查询。

定义写入转换端口

按映射流定义目标对象列，以使上游映射对象可以更新写入转换的传入端口。

1. 在**属性**视图中，单击**端口**选项卡。
2. 选择**列定义依据：映射流**。

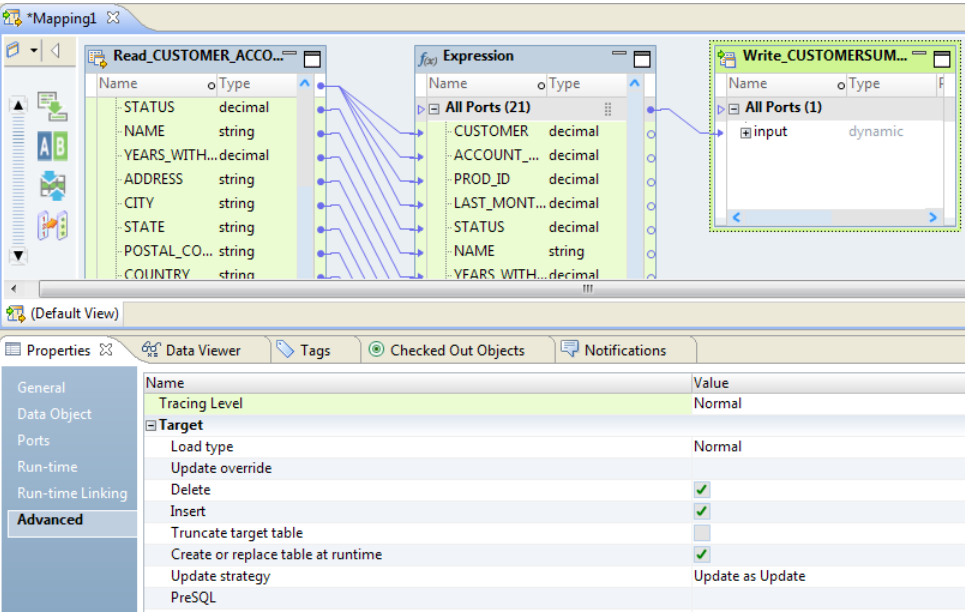
下图显示了**端口**选项卡，其中填充的端口是按关联数据对象定义的：



3. 启用动态端口和目标：
 - a. 将上游端口拖动到写入转换的**输入**窗格。
目标将从上游映射对象获取列定义。
 - b. 在**属性**视图中，单击**高级**选项卡。

c. 选择在运行时创建或替换表。

下图显示了目标对象“高级”选项卡中的在运行时创建或替换表选项：



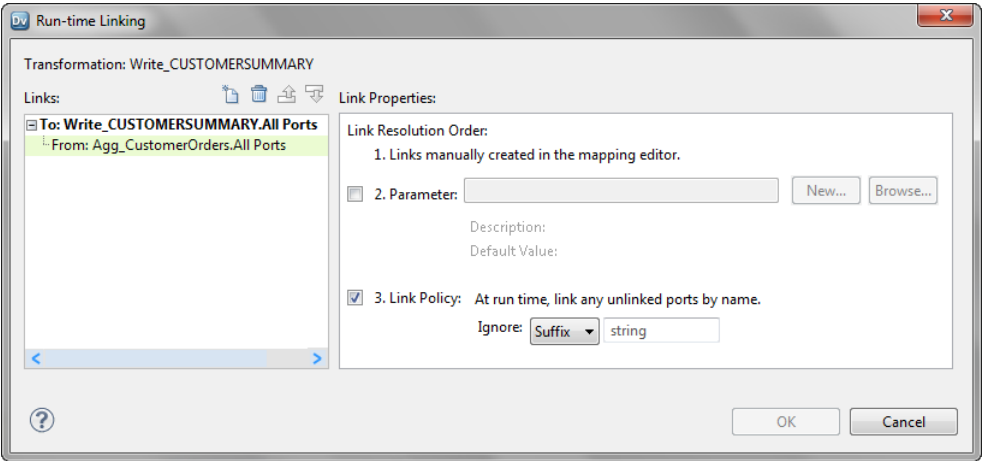
在运行时，数据集成服务会创建目标表，或者丢弃并替换目标表。

注意：当映射包含多个目标且这些目标的列是按同一物理数据对象定义时，请仅为一个目标启用在运行时创建并替换表选项。如果为多个目标启用了此选项，则映射所创建表的元数据将仅与其中一个目标匹配，而且映射将失败。

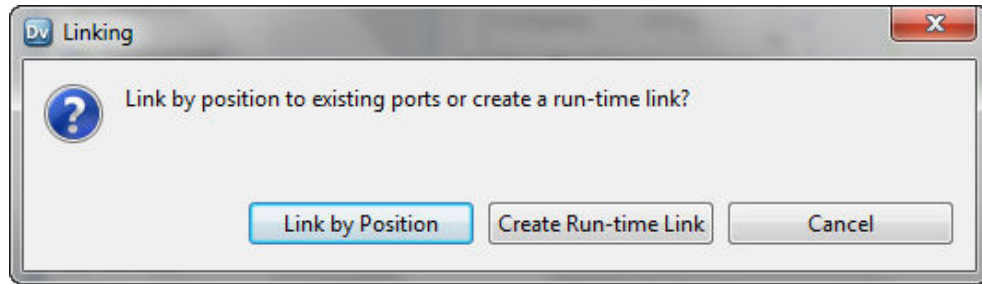
创建和配置运行时链接

配置转换组之间的运行时链接可在运行时根据参数和/或链接策略链接端口。

- 1. 可通过以下方式创建运行时链接：
 - 按住 Ctrl 键并将组拖动到动态映射中的下游转换。此时将显示运行时链接对话框。




- 要创建写入转换或可重用转换的运行时链接，请将组从上游转换拖动到可重用转换或写入转换的某个组。然后，在**链接**对话框中选择**创建运行时链接**以打开**运行时链接**对话框。
- 在要为其创建运行时链接的下游转换中，转至**属性**视图并单击**运行时链接**选项卡。

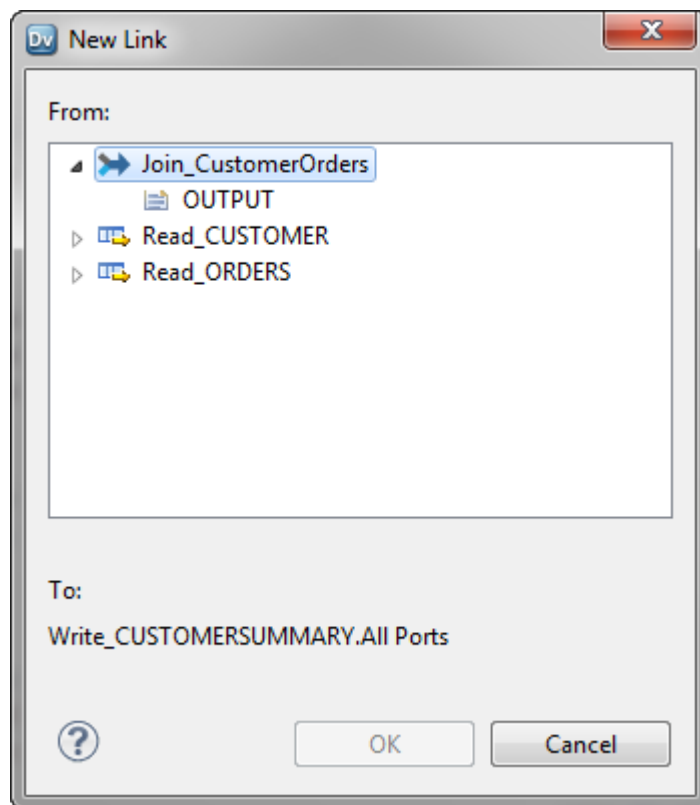


2. 在**链接属性**区域中，选择以下一个或两个选项以确定在运行时链接哪些端口：
 - **参数**。如果端口名称可在映射运行之间更改且您知道端口名称，请使用参数。可以创建一个新参数，也可以选择“输入链接集”类型的现有参数。
 - **链接策略**。使用链接策略可自动按名称链接端口。默认情况下，此选项处于选中状态。如果端口名称包含前缀或后缀，请输入要忽略的字符串。
3. 要创建“输入链接集”类型的新参数，请执行以下步骤：
 - a. 单击**新建**。
 - b. 在**参数**对话框中，输入参数名称。
例如，Cust_InputLinkSet。
 - c. （可选）输入参数说明。
 - d. 以逗号分隔的端口对形式输入参数默认值。
例如，按如下所示输入默认值：
C_NAME->Cust_name, C_ACCTBAL->Cust_acctbal
 - e. 单击**确定**。
4. 要选择“输入链接集”类型的现有参数，请执行以下步骤：
 - a. 单击**浏览**。
 - b. 在**分配参数**对话框中，选择一个参数。
 - c. （可选）从此对话框创建新参数或编辑参数。
 - d. 单击**确定**。

5. (可选) 要从**运行时链接**对话框添加另一运行时链接, 请执行以下步骤:

- a. 单击**新建按钮** (), 该按钮位于**链接**区域中。

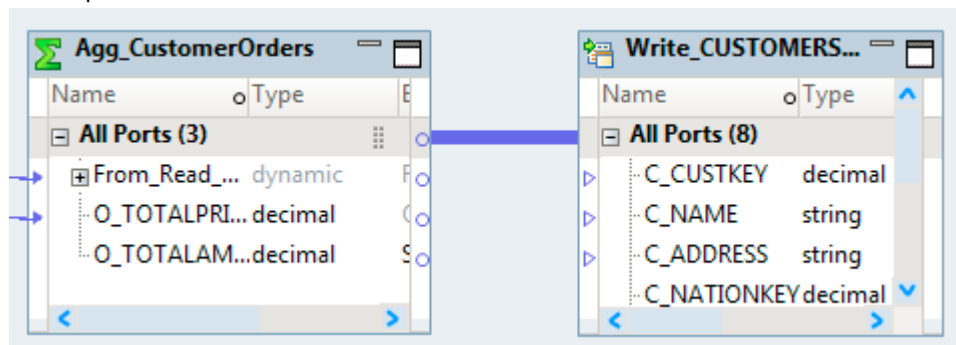
此时将显示**新链接**对话框。



- b. 从动态映射中的另一转换创建一个组。

6. 单击**确定**以创建运行时链接。

Developer tool 即会在组之间创建运行时链接。



7. 要编辑现有运行时链接, 请右键单击链接并选择**运行时链接**

此时将显示**运行时链接**对话框, 可在其中更改用于确定要链接哪些端口的选项。

创建运行时链接

可通过以下方式创建运行时链接：打开“运行时链接”对话框或使用“属性”视图中的“运行时链接”选项卡。???

► 可通过以下方式创建运行时链接：

- 1. 将“所有端口”组拖动到动态映射中的下游转换。
此时将显示**链接**对话框。
- 2. 单击**创建运行时链接**。
- 按住 Ctrl 键并将“所有端口”组拖动到动态映射中的下游转换。
此时将显示**运行时链接**对话框。
- 1. 选择要为其创建运行时链接的下游转换。
- 2. 在“属性”视图中，

验证和运行动态映射

验证映射以确保数据集成服务能够读取和处理整个映射，然后运行映射以将转换的数据写入至目标。

1. 打开映射，单击**编辑 > 验证**。
如果**验证日志**视图中显示错误，请修复错误并重新验证映射。
2. 如果映射有效，请单击**文件 > 保存**以保存映射。
3. 单击**运行 > 运行映射**。
运行映射窗口将显示映射运行的进度。该映射将运行，并将输出写入到目标文件。
4. 单击**窗口 > 显示视图 > 进度**以查看映射运行的进度。
此时将打开**进度**视图。
5. 更改映射运行之间的参数值。
6. 源架构或参数值发生更改后，验证并重新运行映射。

第 8 章

动态映射使用案例

本章包括以下主题：

- [使用案例：用于处理关系源中元数据更改的动态映射, 148](#)
- [使用案例：将动态映射重用于不同的源和目标, 159](#)

使用案例：用于处理关系源中元数据更改的动态映射

您是帮助组织汇总所有客户订单的开发者。该组织每周都会从不同部门收到客户数据表和客户订单数据表。各个部门会经常更改表中列的顺序或者添加一些新列。您需要开发一个能接受更改源架构并汇总所有客户订单的动态映射。

源表

CUSTOMER 和 ORDERS 是映射中读取转换的源表。

下表列出了将 C_CUSTKEY 列作为主键的 CUSTOMER 表的列和元数据：

名称	本地类型	精度	小数位数
C_CUSTKEY	number(p,s)	38	0
C_NAME	varchar2	25	0
C_ADDRESS	varchar2	40	0
C_NATIONKEY	number(p,s)	38	0
C_PHONE	varchar2	15	0
C_ACCTBAL	number(p,s)	10	2
C_MKTSEGMENT	varchar2	10	0

下表列出了 ORDERS 表的列和元数据：

名称	本地类型	精度	小数位数
O_ORDERKEY	number(p,s)	38	0
O_CUSTKEY	number(p,s)	38	0
O_ORDERSTATUS	varchar2	1	0
O_TOTALPRICE	number(p,s)	10	2
O_ORDERDATE	date	19	0
O_ORDERPRIORITY	varchar2	15	0
O_CLERK	varchar2	15	0
O_SHIPPRIOIRITY	number(p,s)	30	0

目标表

CUSTOMERSUMMARY 是映射中写入映射的目标表。

下表列出了 CUSTOMERSUMMARY 表的列和元数据：

名称	本地类型	精度	小数位数
C_CUSTKEY	number(p,s)	38	0
C_NAME	varchar2	25	0
C_ADDRESS	varchar2	40	0
C_NATIONKEY	number(p,s)	38	0
C_PHONE	varchar2	15	0
C_ACCTBAL	number(p,s)	10	2
C_MKTSEGMENT	varchar2	10	0
C_TOTALAMOUNT	number(p,s)	10	2

动态映射

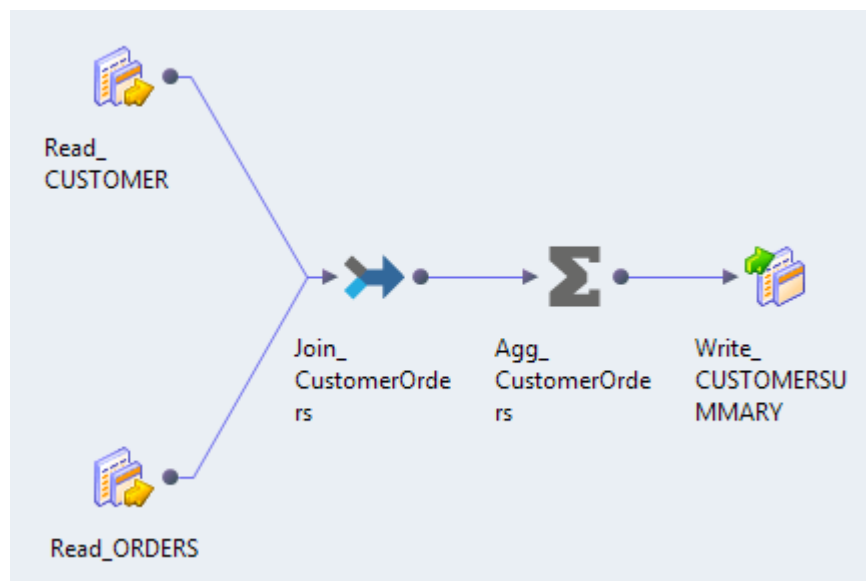
创建映射 m_CustomerLoad 并配置下列动态映射功能：

- 可从动态源中读取数据的读取转换
- 下游转换中可传递新列和已更改列的动态端口
- 可向动态目标写入数据的写入转换
- 可在运行时将端口连接到写入转换的运行时链接

当您运行映射时，数据集成服务将执行以下任务：

1. 提取源文件中数据对象的结构和元数据更改。
2. 通过动态端口将新列和已更改列传递到每个转换。
3. 将新端口和已更改端口连接到写入转换。
4. 将已转换数据写入到目标。

下图显示了映射中的对象：



该映射包含以下对象：

Read_CUSTOMER

表示关系源 CUSTOMER 的读取转换。在关系表中，每个客户对应一个单独的行。

Read_ORDERS

表示关系源 ORDERS 的读取转换。在关系表中，每个客户订单对应一个单独的行。

Join_CustomerOrders

用于联接 CUSTOMER 和 ORDERS 源的联接器转换。

Agg_CustomerOrders

用于汇总客户订单总额的汇总器转换。

Write_CUSTOMERSUMMARY

表示关系目标 CUSTOMERSUMMARY 的写入转换。关系表包含一个映射列，用于写入按客户分组的订单总额的汇总值。

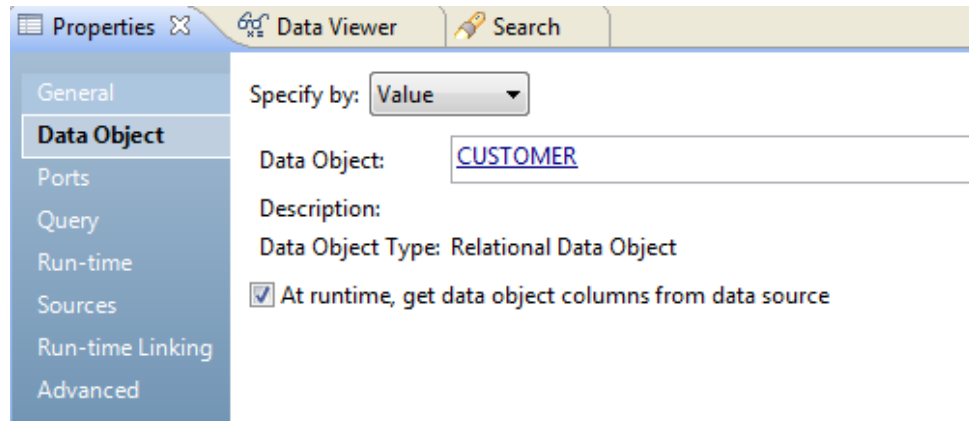
步骤 1. 配置读取转换

配置读取转换，让它在运行时直接从动态源提取列和元数据更改。

1. 添加两个读取转换，分别表示 CUSTOMER 和 ORDERS 关系数据对象。
2. 配置 Read_CUSTOMER 转换，让它在运行时直接从源中提取列和元数据更改。
 - a. 选择 Read_CUSTOMER 转换。

- b. 在属性视图中，单击数据对象选项卡。
- c. 选择在运行时，从数据源获取数据对象列。

下图显示了 Read_CUSTOMER 转换的“数据对象”选项卡设置：



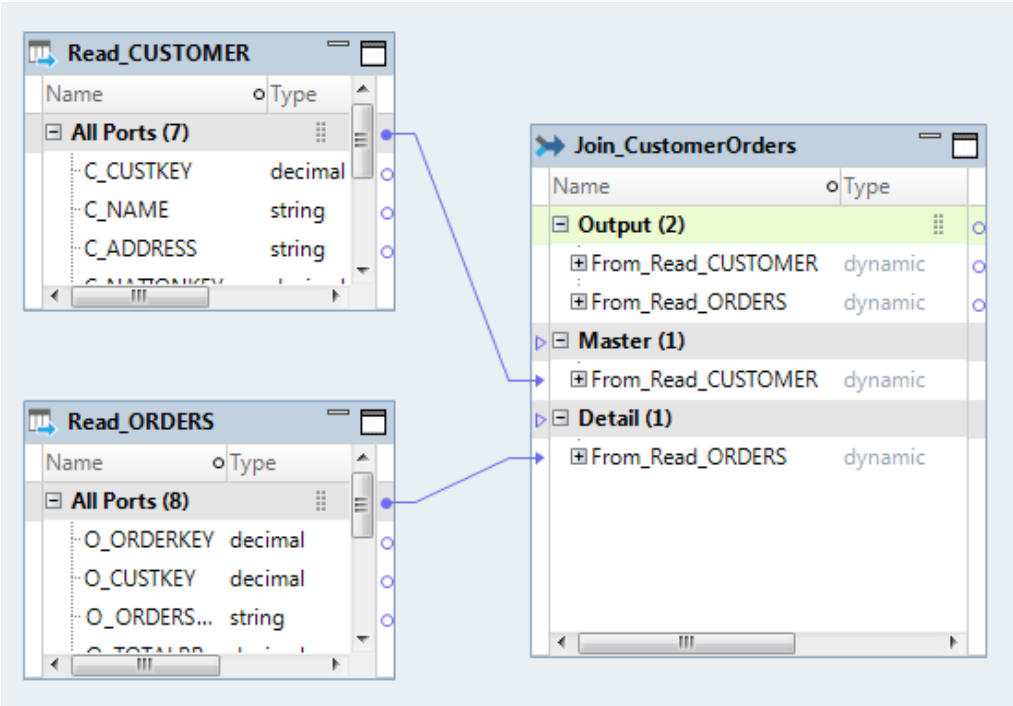
3. 配置 Read_ORDERS 转换，让它在运行时直接从源中提取列和元数据更改。
 - a. 选择 Read_ORDERS 转换。
 - b. 在属性视图中，单击数据对象选项卡。
 - c. 选择在运行时，从数据源获取数据对象列。

步骤 2. 配置联接器转换

向映射中添加一个联接器转换，并为其配置动态端口以接收来自读取转换的所有新列和已更改列。定义联接条件，以便联接 CUSTOMER 和 ORDERS 这两个源表。

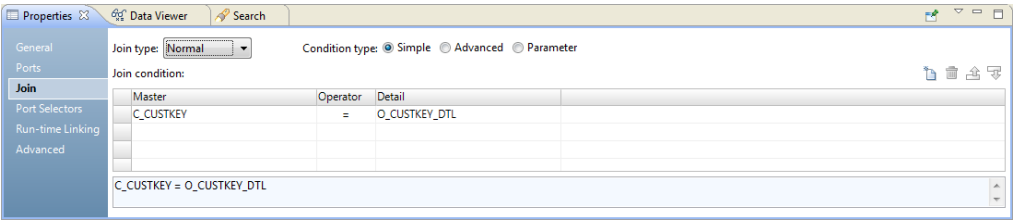
1. 将联接器转换 Join_CustomerOrders 添加到映射。
 2. 在联接器转换中创建动态端口：
 - a. 在 Read_Customer 转换中，将“所有端口”组拖动到联接器转换中的“主”组。
Developer tool 会在“主”组和“输出”组中创建一个动态端口 From_Read_CUSTOMER。
 - b. 在 Read_Orders 转换中，将“所有端口”组拖动到联接器转换中的“详细信息”组。
Developer tool 会在“详细信息”组和“输出”组中创建一个动态端口 From_Read_ORDERS。
- 动态端口会将相应读取转换中的所有端口作为生成的端口包含在内。

下图显示了读取转换中的“所有端口”组已链接到联接器转换中的两个动态端口：



3. 在属性视图中，单击**联接**选项卡。
4. 单击**新建**按钮，然后将联接条件定义为 C_CUSTKEY = O_CUSTKEY_DTL。

下图显示了已定义联接条件的**联接**选项卡：

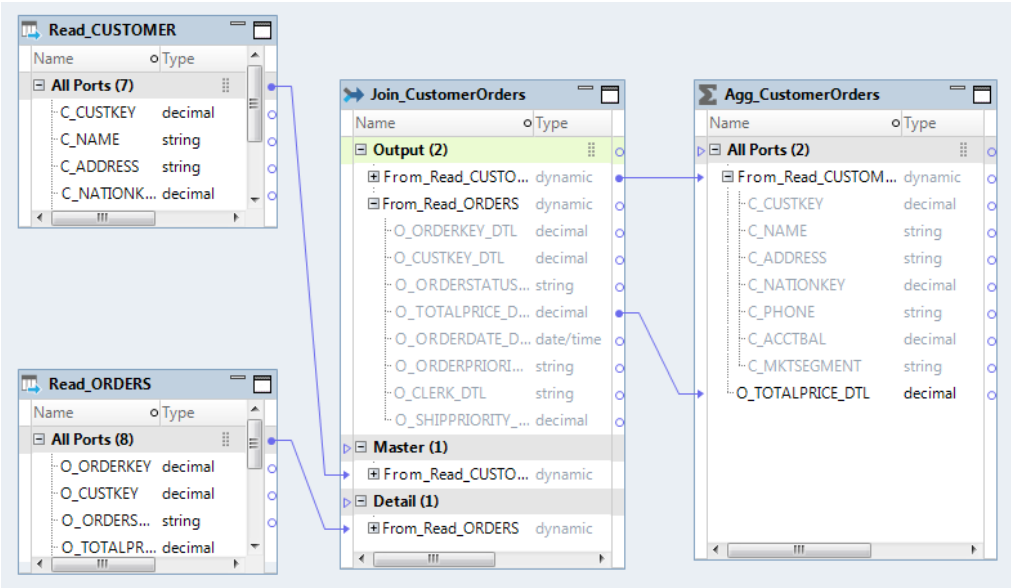


步骤 3. 配置汇总器转换

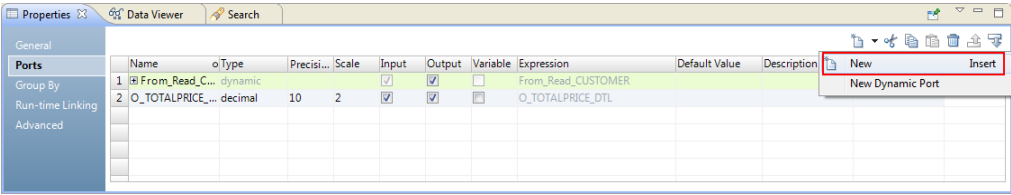
向映射中添加一个汇总器转换，并为其配置动态端口以接收来自联接器转换的所有新列和已更改列。创建一个汇总表达式，以计算客户订单的总价格并按照客户对汇总进行分组。

1. 将汇总器转换 Agg_CustomerOrders 添加到映射。
2. 在汇总器转换中创建动态端口：
 - a. 从联接器转换中的“输出”组中，将 From_Read_CUSTOMER 动态端口拖动到汇总器转换。
动态端口 From_Read_CUSTOMER 将显示在汇总器转换中。
 - b. 从联接器转换中“输出”组的 From_Read_ORDERS 动态端口中，将 O_TOTALPRICE_DTL 生成端口拖动到汇总器转换。

下图显示了联接器转换中的端口已链接到汇总器转换：



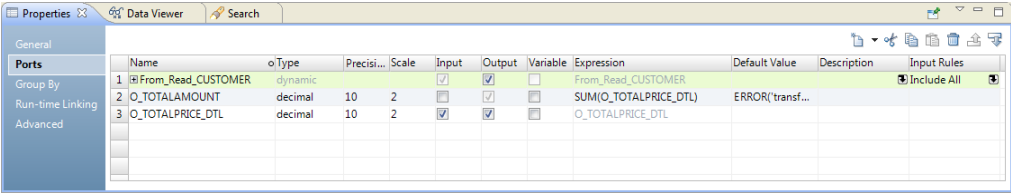
3. 在属性视图中，单击端口选项卡。
4. 单击新建按钮以创建一个用于汇总订单价格的端口。



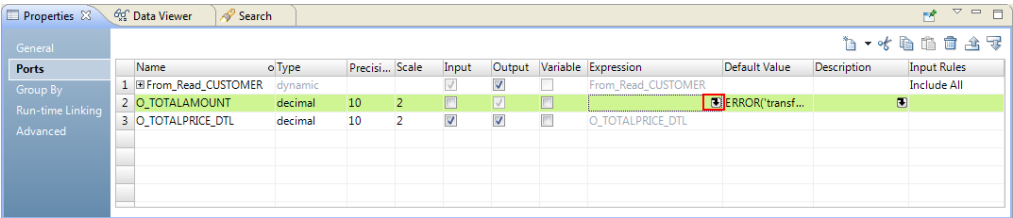
Developer tool 会创建一个名为 Field 的新端口。

5. 选择此新端口，然后如下所示更改列值：
 - 名称：O_TOTALAMOUNT
 - 类型：小数
 - 精度：10
 - 小数位数：2
 - 输入：清除所选内容可将该端口设为仅输出端口。

下图显示了汇总器转换中的端口：



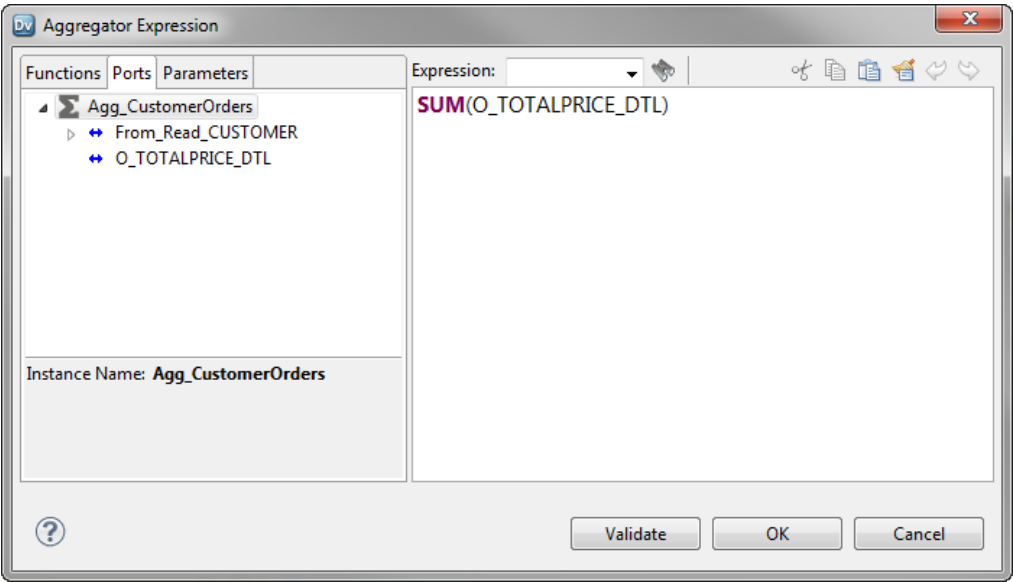
6. 在 O_TOTALAMOUNT 端口的“表达式”列中，单击**打开**按钮。



Name	oType	Precisi...	Scale	Input	Output	Variable	Expression	Default Value	Description	Input Rules
1 From_Read_CUSTOMER	dynamic			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		From_Read_CUSTOMER			Include All
2 O_TOTALAMOUNT	decimal	10	2	<input type="checkbox"/>	<input checked="" type="checkbox"/>		ERROR('transf...			
3 O_TOTALPRICE_DTL	decimal	10	2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		O_TOTALPRICE_DTL			

此时将显示**汇总器表达式**窗口。

7. 将编辑器中的现有表达式替换为以下表达式：SUM(O_TOTALPRICE_DTL)



8. 单击**验证**以验证表达式。

9. 单击**确定**。

10. 单击**确定**退出**汇总器表达式**编辑器。

11. 在**属性**视图中，单击**分组依据**选项卡。

12. 如下所示指定分组端口，以按照细分市场汇总总价格：

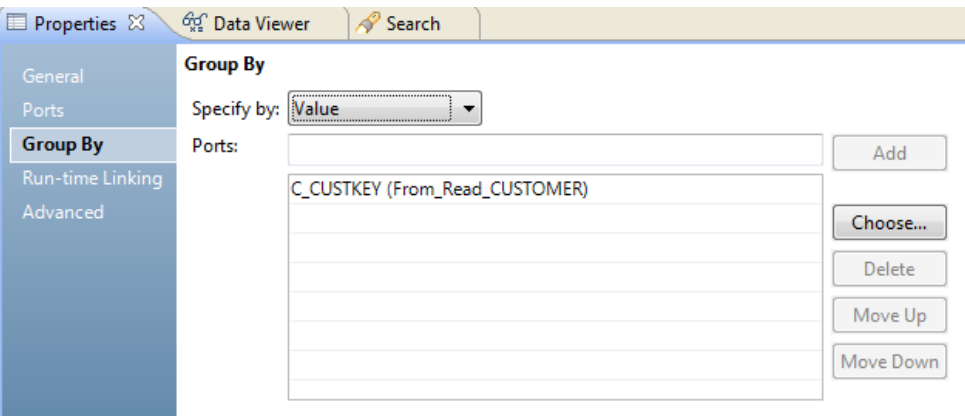
a. 确保选择了**指定依据**列表中的**值**。

b. 单击**选择**。

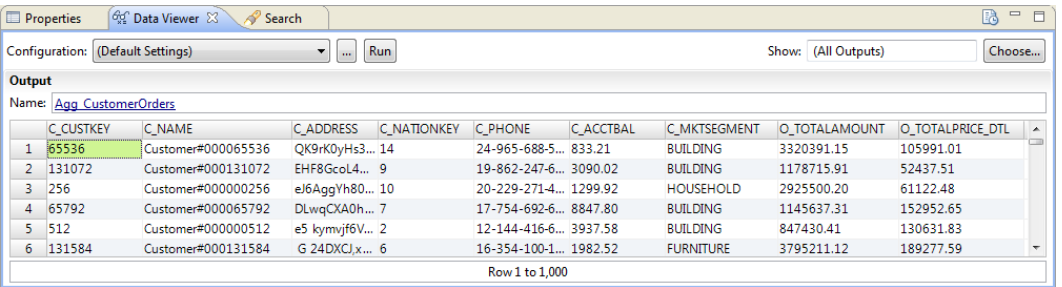
此时将显示**端口**对话框。

- c. 选中 C_CUSTKEY 旁的复选框，然后单击**确定**。

下图显示了选定的分组端口：



可以预览汇总器转换数据，确保其中包含预期结果。在映射编辑器中，右键单击汇总器转换，然后选择**运行数据查看器**。此时将在**数据查看器**视图中显示通过转换计算得出的数据。



	C_CUSTKEY	C_NAME	C_ADDRESS	C_NATIONKEY	C_PHONE	C_ACCTBAL	C_MKTSEGMENT	O_TOTALAMOUNT	O_TOTALPRICE_DTL
1	65536	Customer#000065536	QK9rk0yHs3...	14	24-965-688-5...	833.21	BUILDING	3320391.15	105991.01
2	131072	Customer#000131072	EHF8GcoL4...	9	19-862-247-6...	3090.02	BUILDING	1178715.91	52437.51
3	256	Customer#000000256	eJ6AggYh80...	10	20-229-271-4...	1299.92	HOUSEHOLD	2925500.20	61122.48
4	65792	Customer#000065792	DLwqCXA0h...	7	17-754-692-6...	8847.80	BUILDING	1145637.31	152952.65
5	512	Customer#000000512	e5 kymvjf6V...	2	12-144-416-6...	3937.58	BUILDING	847430.41	130631.83
6	131584	Customer#000131584	G 24DXCJ,x...	6	16-354-100-1...	1982.52	FURNITURE	3795211.12	189277.59

步骤 4. 配置写入转换

添加写入转换，然后将该写入转换配置为在运行时直接从目标中提取列更改。

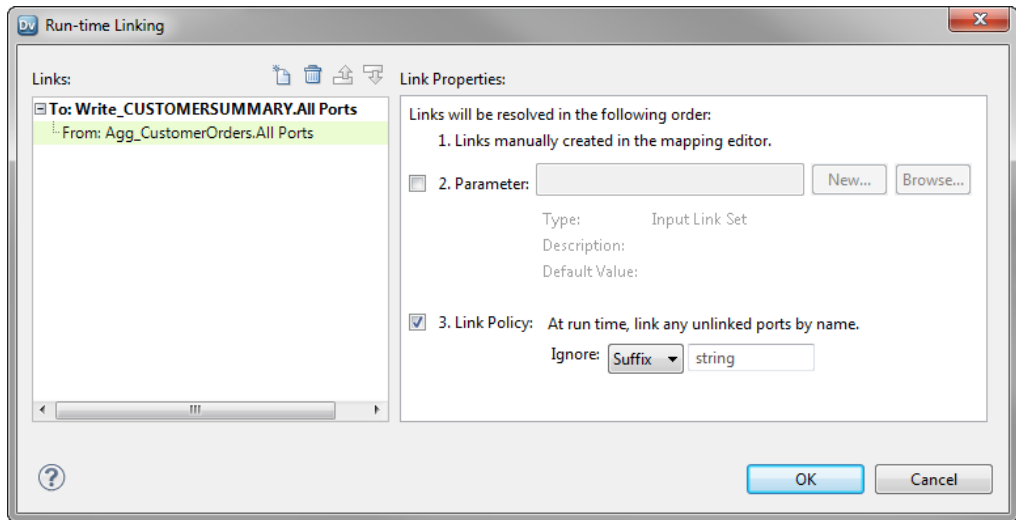
1. 将 CUSTOMERSUMMARY 关系数据对象作为写入转换添加。
该写入转换将在编辑器中显示为 Write_CUSTOMERSUMMARY。
2. 验证写入转换已配置为自动重新导入元数据更改。
 - a. 在**属性**视图中，单击**常规**选项卡。
 - b. 确保选择了**同步输入**端口。
3. 将写入转换配置为在运行时直接从目标表中获取列。
 - a. 在**属性**视图中，单击**数据对象**选项卡。
 - b. 选择在**运行时**，从**数据源**获取**数据对象**列。

步骤 5. 创建并配置运行时链接

创建指向写入转换的运行时链接，然后将链接策略配置为在运行时按照端口名称建立并解析链接。

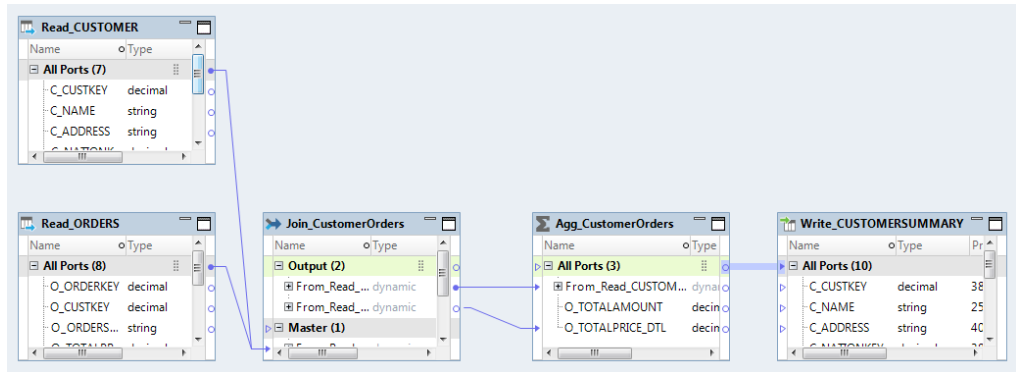
1. 按住 Ctrl 键并将汇总器转换中的“所有端口”组拖动到写入转换的“所有端口”组。
此时将显示**运行时链接**对话框。

2. 确认已选择“链接属性”区域中的**链接策略**，以在运行时按照名称自动链接各个端口。



3. 单击**确定**。

Developer tool 会在汇总器转换和写入转换之间创建运行时链接。



步骤 6. 验证并运行映射

验证并运行该映射。预览目标数据对象中的数据以验证结果。

1. 在映射编辑器中，单击**编辑 > 验证**。
2. 如果映射有效，请单击**文件 > 保存**以保存映射。
3. 单击**运行 > 映射**。

运行映射窗口将显示映射运行的进度。该映射将运行，并将输出写入到目标文件。

4. 在**对象浏览器**视图中，找到项目中的 CUSTOMERSUMMARY 数据对象并双击该数据对象。
该数据对象随即在编辑器中打开。
5. 单击**窗口 > 显示视图 > 数据查看器**。
此时将显示**数据查看器**视图。

6. 在数据查看器视图中，单击运行。

数据查看器视图将运行并显示数据。

在本示例中，C_TOTALAMOUNT 列显示客户订单的汇总总价格。

Output							
Name: CUSTOMERSUMMARY							
	C_CUSTKEY	C_NAME	C_ADDRESS	C_NATIONKEY	C_PHONE	C_ACCTBAL	C_TOTALAMOUNT
1	287	Customer#000...	KTsaTAJRC0e...	4	14-330-840-6321	1734.18	MACHINERY 701351.00
2	1055	Customer#000...	Z3AggyEMPM...	7	17-802-131-7180	639.93	HOUSEHOLD 1549236.00
3	32	Customer#000...	jD2xZzi UmlD...	15	25-430-914-2194	3471.53	BUILDING 1336868.00
4	544	Customer#000...	Jv7vcm,oE,HEy...	5	15-572-651-1323	4974.68	AUTOMOBILE 2900638.00
5	289	Customer#000...	NUilehg0nVOk...	10	20-456-773-7693	-215.75	AUTOMOBILE 2893675.00
6	545	Customer#000...	AsYw6k,nDUQ...	10	20-849-123-8918	7505.33	AUTOMOBILE 975375.00
7	1057	Customer#000...	xyV8 FbW4xSj...	24	34-750-735-1314	-377.11	AUTOMOBILE 2838452.00
8	34	Customer#000...	Q6G9wZ6dncz...	15	25-344-968-5422	8589.70	HOUSEHOLD 4295230.00
9	290	Customer#000...	8OIPT9G 8UqV...	4	14-458-625-5633	1811.35	MACHINERY 618490.00
10	1058	Customer#000...	R0NIEcSVDQ4r...	19	29-818-620-9637	6807.55	MACHINERY 1252089.00

步骤 7. 在源架构发生更改后运行映射

提供客户数据表和客户订单数据表的部门在两个表中新增了一个“注释”列。查看动态映射中的列更改，然后验证并重新运行该映射。可以预览目标数据对象中的数据以验证更新后的结果。

下表列出了已新增 C_COMMENT 列的更新后 CUSTOMER 表的列和元数据：

名称	本地类型	精度	小数位数
C_CUSTKEY	number(p,s)	38	0
C_NAME	varchar2	25	0
C_ADDRESS	varchar2	40	0
C_NATIONKEY	number(p,s)	38	0
C_PHONE	varchar2	15	0
C_ACCTBAL	number(p,s)	10	2
C_MKTSEGMENT	varchar2	10	0
C_COMMENT	varchar2	117	0

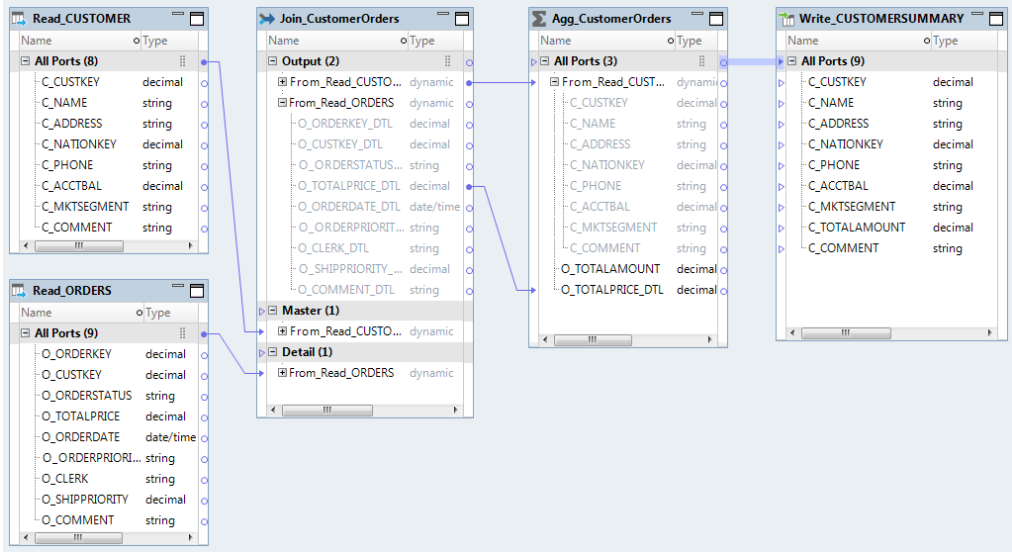
下表列出了已新增 O_COMMENT 列的更新后 ORDERS 表的列和元数据：

名称	本地类型	精度	小数位数
O_ORDERKEY	number(p,s)	38	0
O_CUSTKEY	number(p,s)	38	0
O_ORDERSTATUS	varchar2	1	0
O_TOTALPRICE	number(p,s)	10	2

名称	本地类型	精度	小数位数
O_ORDERDATE	date	19	0
O_ORDERPRIORITY	varchar2	15	0
O_CLERK	varchar2	15	0
O_SHIPPRIOIRITY	number(p,s)	30	0
O_COMMENT	varchar2	79	0

1. 在映射编辑器中，查看映射的更改。

读取和写入转换将自动反映新列。联接器和汇总器转换中的动态端口将自动包含来自相应读取转换的新 C_COMMENT 和 O_COMMENT 列。



2. 要验证已更改的映射，请单击编辑 > 验证。

3. 如果映射有效，请单击文件 > 保存以保存映射。

4. 单击运行 > 映射。

运行映射窗口将显示映射运行的进度。该映射将运行，并将输出写入到目标文件。

5. 在对象浏览器视图中，找到项目中的 CUSTOMERSUMMARY 数据对象并双击该数据对象。

该数据对象随即在编辑器中打开。

6. 单击窗口 > 显示视图 > 数据查看器。

此时将显示数据查看器视图。

7. 在数据查看器视图中，单击运行。

数据查看器视图将运行并显示数据。

8. 验证在源架构发生更改后该映射显示预期结果。

C_TOTALAMOUNT 列显示客户订单的汇总总价格。

使用案例：将动态映射重用于不同的源和目标

您是帮助组织清理不同数据文件以删除字符串值开头和结尾空格的开发者。数据文件有不同的列名称并有字符串类型的多个列。您需要开发一个动态映射以删除来自不同源的字符串开头和结尾的空格并将输出写入到不同目标。

源文件

源文件是所含字符串数据的开头和结尾存在空格的平面文件。读取转换的源文件包含 Customer_FF 和 orders_FF。

在示例过程中，第一次映射运行从 Customer_FF 文件中读取数据，第二次映射运行从 orders_FF 文件中读取数据。

Customer_FF 列和数据

Customer_FF 包含以下列：

```
C_Id
C_Fullname
C_title
C_comp
C_addr
C_suite
C_city
C_state
C_zip5
C_country
C_phone
C_fax
C_date
C_email
C_description
```

其中，C_ID 和 C_zip5 列为数字数据类型，而其他列为字符串数据类型。

Customer_FF 包含以下数据：

```
C_Id,C_Fullname,C_title,C_comp,C_addr,C_suite,C_city,C_state,C_zip5,C_country,C_phone,C_fax,C_date,C_email,C_description
1, Smith John,Account Executive,DKR MANAGEMENT COMPANY INC,100 High Street,5406,Anytown,TN,22342,USA,4047668150,2124031386,31/08/1985,bwilliams@yahoo.com, ACTIVE
2,Balasubramanian Krishna,Account Executive,EASTON & COMPANY,71 Congress Parkway,789,Bangalore,Karnataka,38103,India,4046345228,4151689756,29/10/1985,bmatthewc@univ.edu, ACTIVE
3, Johnson Lars,Regional Sales Exec,GREATER BAY BANCORP,123 Snow St.,43543,St. Paul,MN,55103,USA,4046581534,6122945948,7/9/1992, ehpuniv.edu,INACTIVE
4,Zogby Kevin,Regional Sales Exec, HEWLETT-PACKARD,317 29th. St.,5856,San Francisco,CA,94116,USA,4042662730,4155466814,7/8/1985,grobertwuniv.edu, ACTIVE
5,Franklin Roosevelt,Sales Representative,JAYD TRADING,1511 Wacker Dr,6334,Chicago,IL,60606,USA,7703965851,2065075486,20/10/1982,trichard@univ.edu,INACTIVE
6, Cruz Emilio,Sales Representative,JEFFERSON-PILOT LIFE INSURANCE,700 Ponce de Leon Blvd,757,Miami,FL,33134,USA,4043500799,2127655499,31/07/1983,ahelle@mailcity.com, ACTIVE
7, King BB,Sales Representative,KUWAIT PETROLEUM CORPORATION,18 Beale St,967,Memphis,TN,38103,USA,4046243979,2151717120,27/09/1989, glizziem@univ.edu ,INACTIVE
8,Presley Elvis,Sales Representative,PRINCIPIA PARTNERS,45 N Green St.,43546,Tupelo,MS,38804,USA,4043733125,3311313591,26/07/1992,, ACTIVE
9,Olson Floyd,Acct MGR., SOLITON ASSOCIATES INC.,21 Lake Harriet Pkwy,869790,Mineapolis,MN,55410,USA,7706425402,3232429056,27/08/1993,,INACTIVE
10,Chu Steven,Account Executive,WQXR,2100 Sepulveda Blvd,3434,Los Angeles,CA,90049,USA,4042319005,2126509756,29/09/1988,akennetha@univ.edu, ACTIVE
```

例如，第一行和第三行在名称的开头存在空格：

```
1, Smith John,
3, Johnson Lars,
```

orders_FF 列和数据

orders_FF 包含以下列：

OrderID
Customer_ID
Company
CompanyAddress
CompanyCity
CompanyState
CompanyZip
OrderContact
DeliveryAddress
DeliveryCity
DeliveryState
PaymentType
PaymentTerms
Title
DeliveryOption
DeliveryVendor
ConfirmationCode
OrderAmount
OrderType
ProductDescription

其中，Customer_ID 列为数字数据类型，而其他列为字符串数据类型。

orders_FF 包含以下数据：

0-5079,10110085,JOSEPH TAL LYON & ROSS,96 FISHER ROAD, MAHWAH,NJ,7430,PARKE PERSLEY OR RAYFORD
LECROY,96 FISHER ROAD,MAHWAH,NJ,American Express,CHARGE,Account Executive,UPA,United Parcel Service
Air,44162,\$21.00 ,Generic,O/L/B P/W L/S TAWNY SHIMMER .08 OZ.
0-6658,10110086,NRCA,10255 W.HIGGINS RD., ROSEMONT,IL,60018-5607,ROLANDA SORTO,10255 W.HIGGINS
RD.,ROSEMONT,IL,American Express,CHARGE,Account Executive,UPA,United Parcel Service Air,
44163,\$56.40 ,Generic,O-L.B PW LIPSTYLO LASTING PERFECTION .08 OZ.
0-8195,10110087,POND EQUITIES,4522 FT. HAMILTON PKWY., BROOKLYN,NY,11219, KONSTANTIN PEDDICORD,4522
FT. HAMILTON PKWY.,BROOKLYN,NY,American Express,CHARGE,Account Executive,UPA,United Parcel Service
Air,44164,\$78.00 ,Generic,O/L/B P/W L/S TAWNY SHIMMER LASTING PERFECTION LIPSTYLO TAWNY SHIMMER .08
OZ.
0-9130,10110088, SCHRODER & COMPANY ,787 SEVENTH AVENUE, NEW YORK,NY,10019,GIORGIA
TWITCHELL,787 SEVENTH AVENUE,NEW YORK,NY,American Express,CHARGE,Account Executive,UPA,United Parcel
Service Air,44165,\$14.00 ,Generic,A/COL L PERFECTION L/S REF P SUPREME LASTING PERFECTION LIPSTYLO
TAWNY SHIMMER .08 OZ.
0-9352,10110089,YUASA TRADING COMPANY (AMERICA),150 EAST 52ND STREET,NEW YORK,NY,10005,STEFFI
MCGLOWN,150 EAST 52ND STREET,NEW YORK,NY,American Express,CHARGE,Account Executive,UPA,United Parcel
Service Air,44166,\$54.00 ,Generic,O/L/B L PERFECTION REF LIPSTYLO COFFEE PEACH SUPREME .08 OZ.
0-9517,10110090,DAI ICHI KANGYO BANK,1 WORLD TRADE CENTRE SUITE 49 - 11,NEW YORK,NEW YORK,
10048,AIKEN DOBRICK,1 WORLD TRADE CENTRE SUITE 49 - 11,NEW YORK,NEW YORK,American
Express,CHARGE,Account Executive,UPR,United Parcel Service Red,44167,\$58.00 ,Generic,LASTING
PERFECTION LIP COLOR HOLLYWOOD GLAMOUR 1.7 G MAUVE ICE #752
0-9639,10110091,FIRST GLOBAL SECURITIES,614 EAST COLORADO BLVD.,PASADENA,CA,91101, KIRSTENI SIPPEL,
614 EAST COLORADO BLVD.,PASADENA,CA,American Express,CHARGE,Account Executive,FSO,Federal Express
Overnight,44168,\$24.00 ,Generic,A/COL L PERFECTION L/S REF P SUPREME .08 OZ.
0-9761,10110092,MILTON PARTNERS,56 MASON STREET, GREENWICH ,CT,6830,ORLANTA DYSON,56 MASON
STREET,GREENWICH,CT,American Express,CHARGE,Account Executive,UPI,United Parcel Service
International,44169,\$75.20 ,Generic,LASTING PERFECTION LIPSTYLO PEACH SU .08 OZ.
0-9883,10110093, TAX ANALYSTS ,6830 N. FAIRFAX DRIVE,ARLINGTON,VA,22213,NEWLIN MCCART,6830 N.
FAIRFAX DRIVE,ARLINGTON,VA,American Express,CHARGE,Account Executive,FSO,Federal Express Overnight,
44170,\$275.40 ,Generic,O/L/B L PERFECTION L/STYLO REF P SUPRE
0-5438,10110094,VECTORMEX,535 MADISON AVENUE,NEW YORK,NY,10022,LONNA HUGGINS,535 MADISON AVENUE,NEW
YORK,NY,American Express,CHARGE,Account Executive,FSO,Federal Express Overnight,
44171,\$60.00 ,Generic,LASTING PERFECTION DOUBLE PERFORMANCE LIPSTICK PEACH SUPREME .08 OZ.

例如，第四行在公司名称的开头和结尾存在空格：

0-9130,10110088, SCHRODER & COMPANY ,

目标文件

目标文件是一个平面文件，在删除字符串值开头和结尾的空格之后映射会将数据写入到此文件中。创建 customerTrim.csv 文件作为目标数据对象的目标文件。

如果使用不同的数据源，使用参数可在运行时更改输出文件名。数据集成服务会根据目标文件名的参数值创建输出文件，然后将该文件保存到系统上安装了 Informatica 服务的目标目录中。

动态映射

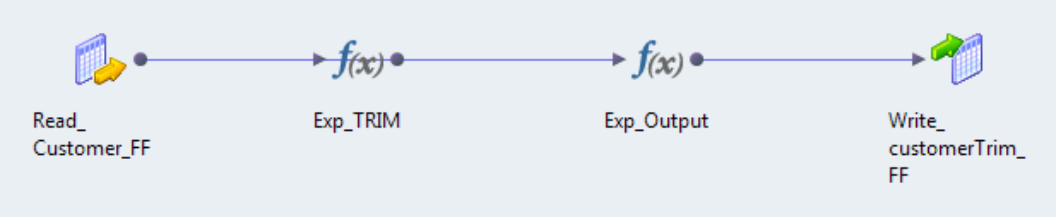
创建映射 m_Replication_Template 并配置下列动态映射功能：

- 使用数据对象参数从不同源读取数据的读取转换
- 下游转换中可传递新列或已更改列的动态端口
- 包含可删除字符串中前导和尾随空格的动态表达式的表达式转换
- 根据映射流创建目标列并将目标数据对象中的参数用于目标文件名的写入转换

当您运行映射时，数据集成服务将执行以下任务：

1. 根据源数据对象的参数值从相应源文件中读取数据。
2. 通过动态端口将新列和已更改的列传递到下游转换。
3. 扩展动态表达式并处理动态端口中每个生成端口的表达式函数。
4. 在写入转换中根据映射流创建列，并按照参数值将转换的数据写入适当的目标文件。

下图显示了映射中的对象：



该映射包含以下对象：

Read_Customer_FF

表示平面文件源的读取转换。平面文件包含的字符串数据具有前导和结尾空格。

Exp_TRIM

包含可删除字符串类型端口的前导和尾随空格的动态表达式的表达式转换。

Exp_Output

包含已转换字符串端口和源对象中剩余端口的表达式转换。

Write_customerTrim_FF

表示平面文件目标的写入转换。映射将输出写入到平面文件目标。

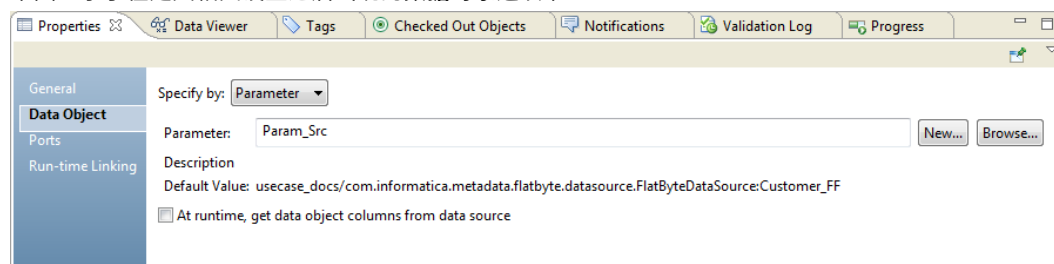
步骤 1. 配置 Read_Customer_FF 读取转换

将 Read_Customer_FF 读取转换配置为使用资源类型的参数，以更改映射运行之间的源数据对象。

1. 添加表示 Customer_FF 平面文件数据对象的读取转换。
该读取转换将在编辑器中显示为 Read_Customer_FF。
2. 在属性视图中，单击数据对象选项卡。

3. 在**指定依据**列表中选择**参数**。
4. 单击**新建**创建一个新参数。
此时将显示**参数**对话框。
5. 将参数名称输入为 Param_Src。
6. 单击**默认值**中的**浏览**。
7. 在**选择位置**对话框中，选择要作为默认值提供的**数据对象**。
示例默认值是 MRS//Cust_Dept/Customer_FF，其中 MRS 为模型存储库服务，Cust_Dept 为存储 Customer_FF 数据对象的项目。可以在运行映射时更改参数值。

下图显示了在定义相关设置之后出现的**数据对象**选项卡：



步骤 2. 配置 Exp_TRIM 表达式转换

将表达式转换 Exp_TRIM 添加到映射，并配置转换以删除字符串开头和结尾的空格。

1. 创建可从读取转换中接收列的动态端口，并定义仅包含字符串端口的输入规则。
2. 创建动态输出端口并定义动态表达式，以删除字符串开头和结尾的空格。

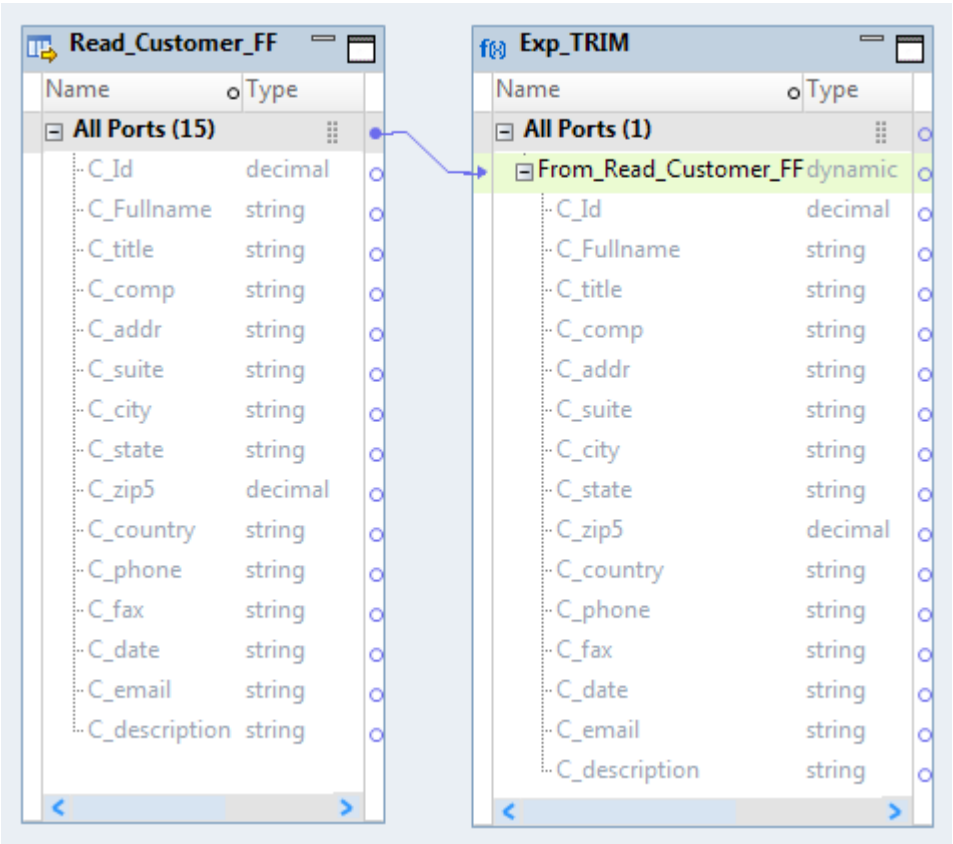
创建动态端口并定义输入规则

创建可从读取转换接收列的动态端口。定义输入规则，以仅包含动态端口中的字符串端口。

1. 将 Read_Customer_FF 转换中的“所有端口”组拖动到 Exp_TRIM 转换中的“所有端口”组。

Developer tool 会在 Exp_TRIM 转换中创建动态端口 From_Read_CUSTOMER_FF。

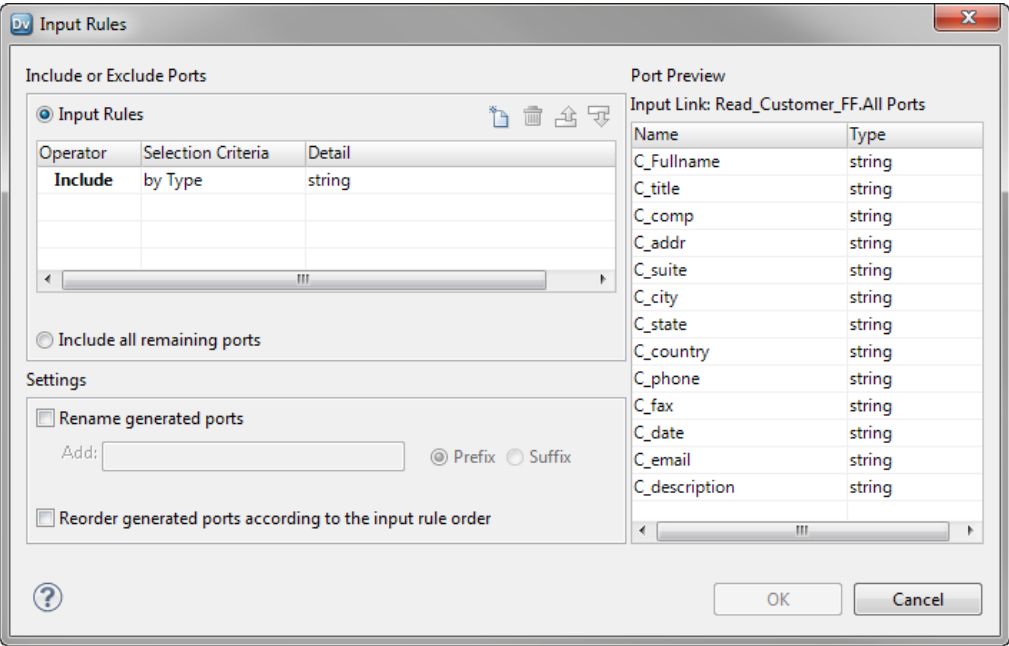
下图显示了 Exp_TRIM 转换中的动态端口，该端口将读取转换中的所有端口作为生成的端口包含在内：



2. 右键单击该动态端口，然后选择**编辑输入规则**。
此时将显示**输入规则**对话框。
3. 从**选择条件**列中选择**按类型**。
4. 单击**详细信息**按钮，选择要包含的数据类型。
5. 在**输入规则详细信息:按类型**对话框中，从列表中选择**字符串**数据类型。

6. 验证端口预览区域，以确保仅显示字符串端口。

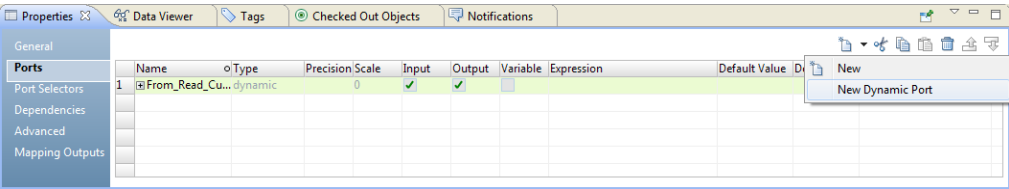
下图显示了输入规则对话框的端口预览区域中的已更新输入规则和字符串端口。



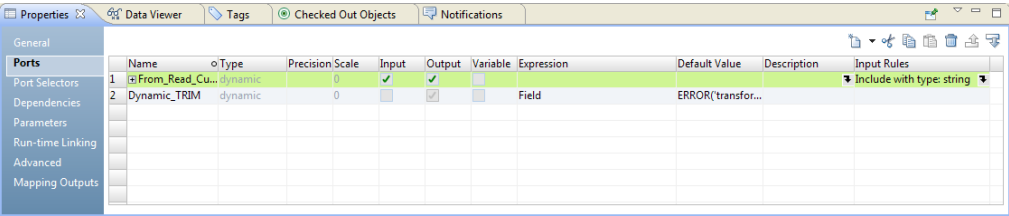
创建动态端口并定义动态表达式

将动态端口创建为 Exp_TRIM 转换中的仅输出端口。定义动态表达式以删除字符串开头和结尾的空格。

1. 在 Exp_TRIM 转换的属性视图中，单击端口选项卡。
2. 单击新建动态端口。

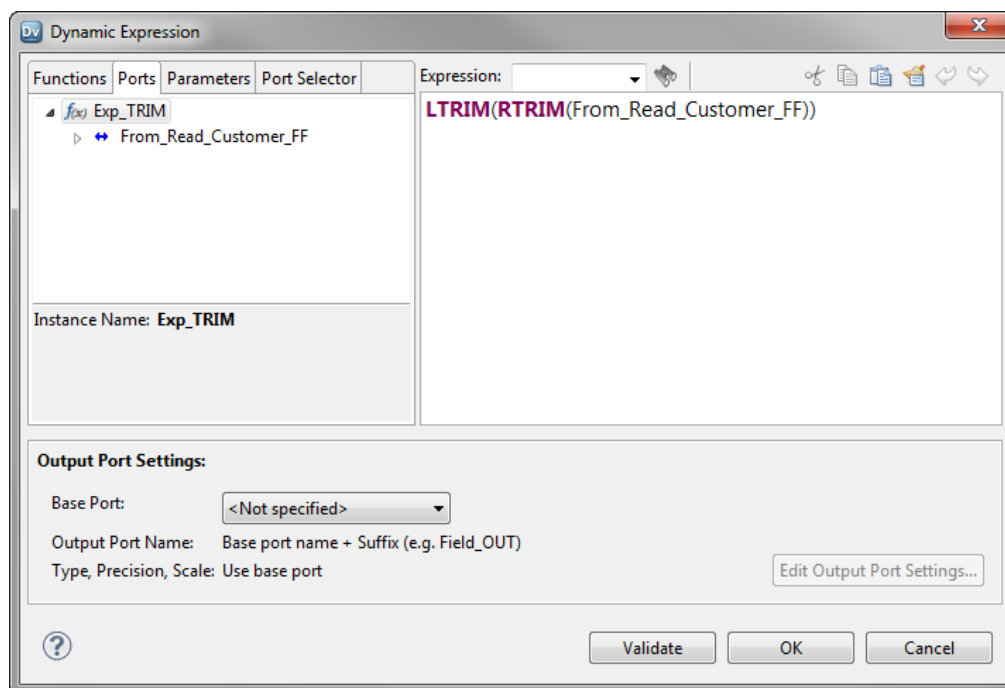


3. 清除输入列，以将此端口设为仅输出端口。
4. 将所创建的动态端口重命名为 Dynamic_TRIM。



5. 在 Dynamic_TRIM 动态端口的表达式列中，单击打开按钮 (🔍)。
- 此时将打开动态表达式窗口。

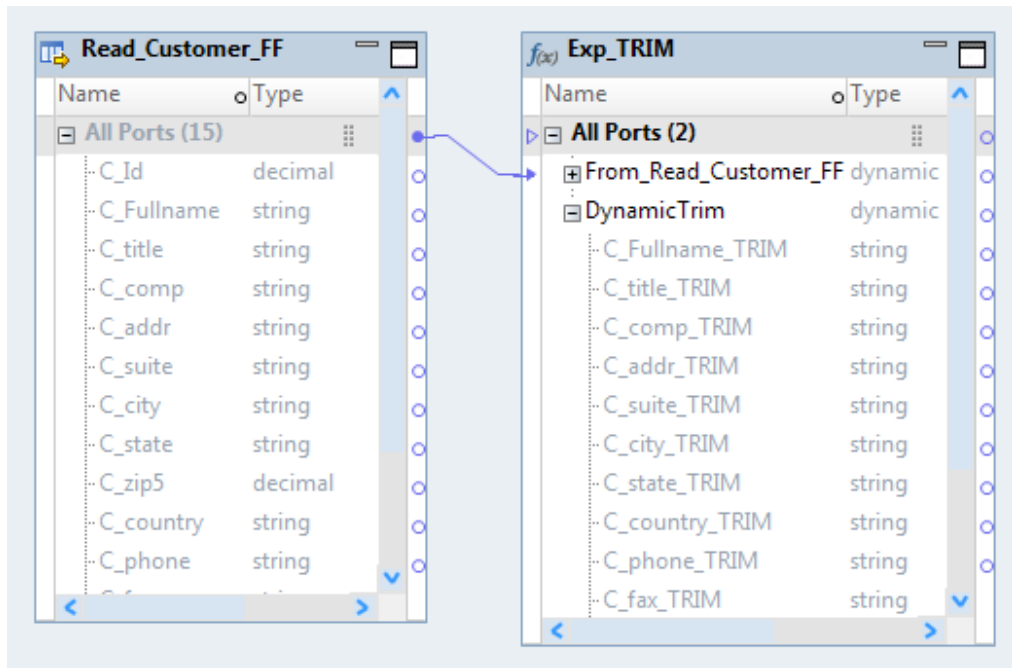
6. 将编辑器中的现有表达式替换为以下表达式：LTRIM(RTRIM(From_Read_Customer_FF))



7. 单击**验证**以验证表达式。
8. 单击**确定**退出**验证表达式**对话框。
9. 如下所示重命名表达式的输出端口：
- 在**输出端口**设置区域中，选择基本端口作为 From_Read_Customer_FF。
 - 单击**编辑输出端口设置**。
此时将显示**输出端口设置**对话框。
 - 在**名称**列表中，选择**基本端口名称 + 后缀**。
 - 在**后缀**框中输入 _TRIM。
 - 单击**确定**。

10. 单击**确定**退出**动态表达式编辑器**。

下图显示了包含已重命名生成端口的 `Dynamic_TRIM` 动态端口。

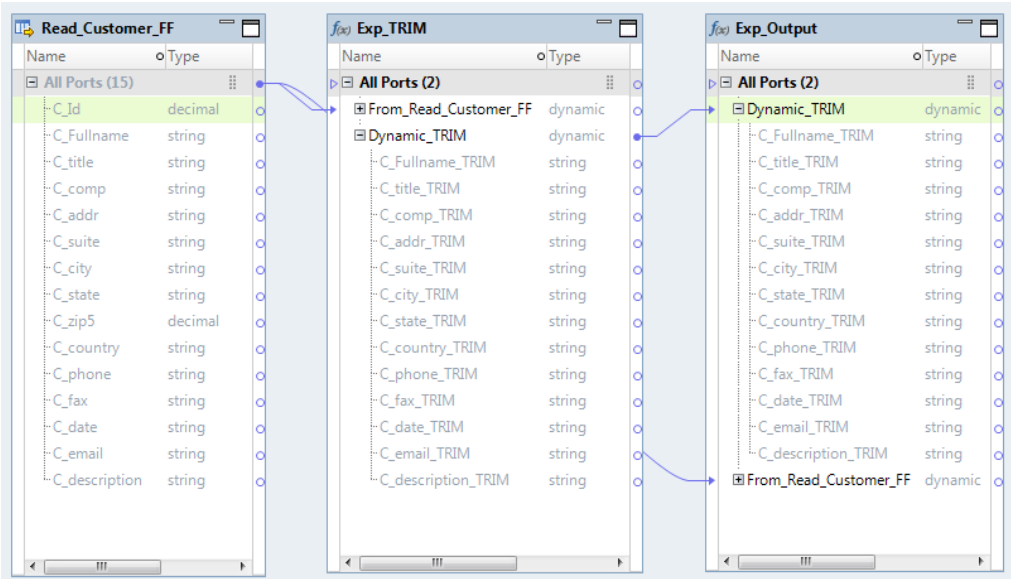


步骤 3. 配置 `Exp_Output` 表达式转换

将表达式转换 `Exp_Output` 添加至映射。创建可从 `Exp_TRIM` 转换获取输出端口动态端口。创建另一个可从读取转换获取端口的动态端口，并定义仅包含未用端口的输入规则。

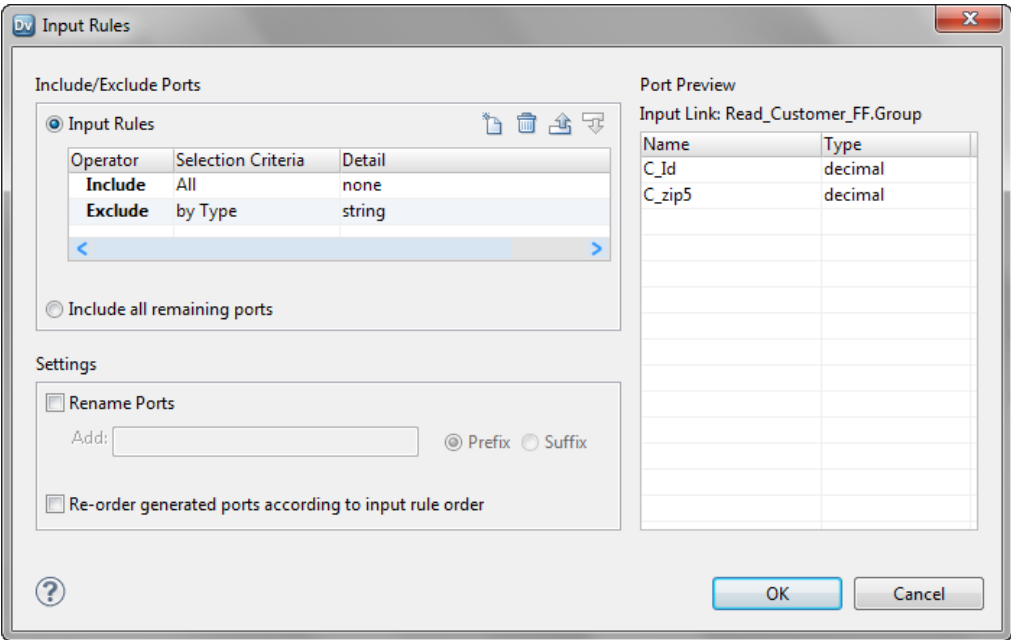
1. 从 `Exp_TRIM` 转换中，将 `DynamicTrim` 动态端口拖动到 `Exp_Output` 转换中的“所有端口”组。
Developer tool 会在 `Exp_Output` 转换中创建动态端口 `DynamicTrim`。
2. 从 `Read_Customer_FF` 转换中，将“所有端口”组拖动到 `Exp_Output` 转换中的“所有端口”组。
Developer tool 会在 `Exp_Output` 转换中创建动态端口 `From_Read_Customer_FF`。

下图显示了 Exp_Output 转换中的两个动态端口。



3. 右键单击 From_Read_CUSTOMER_FF 动态端口，选择**编辑输入规则**。
此时将显示**输入规则**对话框。
4. 单击**新建**图标以添加输入规则。
5. 在**运算符**列中选择**排除**。
6. 在**选择条件**列中选择**类型**。
7. 单击**详细信息**箭头以选择要包含的数据类型。
8. 在**输入规则详细信息:按类型**对话框中，从列表中选择**字符串**数据类型。
9. 验证**端口预览**区域以确保不显示字符串端口。

下图显示了**输入规则**对话框的**端口预览**区域中的已更新输入规则和端口。





步骤 4. 配置 Write_customerTrim_FF 写入转换

创建 customerTrim_FF 数据对象，并将其配置为对输出文件名称使用字符串类型的参数。配置 Write_customerTrim_FF 转换，以根据 Exp_Output 转换中的列在运行时创建目标文件。

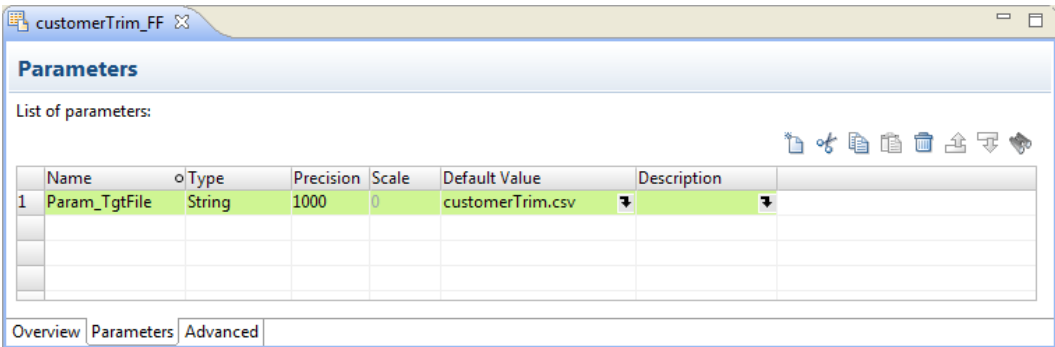
将数据对象配置为使用参数

创建 customerTrim_FF 数据对象以作为写入转换添加到映射中，然后将该数据对象配置为对输出文件名称使用字符串类型的参数。

1. 根据 customerTrim.csv 文件创建 customerTrim_FF 数据对象。
2. 要对输出文件使用参数，请执行以下步骤：
 - a. 在数据对象的**参数**选项卡中，单击**新建按钮** () 以创建一个新参数。
 - b. 在**名称**列中，将参数名称更改为 Param_TgtFile。
 - c. 在**默认值**列中，单击**打开按钮** ()。
此时将显示**编辑参数值**窗口。
 - d. 将默认文件名值输入为 customerTrim.csv，然后单击**确定**。

3. 保存 customerTrim_FF 数据对象。

下图显示了包含新参数的**参数**选项卡：

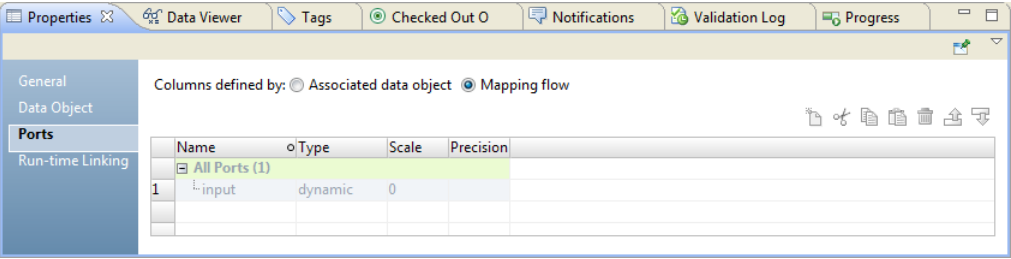


从映射流创建目标列

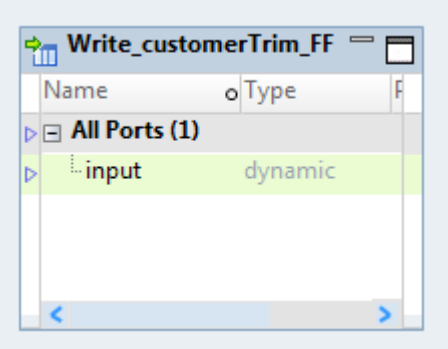
将写入转换添加到映射中，然后将 Write_customerTrim_FF 转换配置为在运行时根据 Exp_Output 转换中的列创建目标文件。

1. 将 customerTrim_FF 数据对象作为写入转换添加到映射中。
2. 在写入转换的**属性**视图中，单击**端口**选项卡。
3. 选择**映射流**选项以定义目标列。
Developer tool 会在 Write_customerTrim_FF 转换中创建动态端口 **input**。

下图显示了在选择该选项之后出现的端口选项卡：

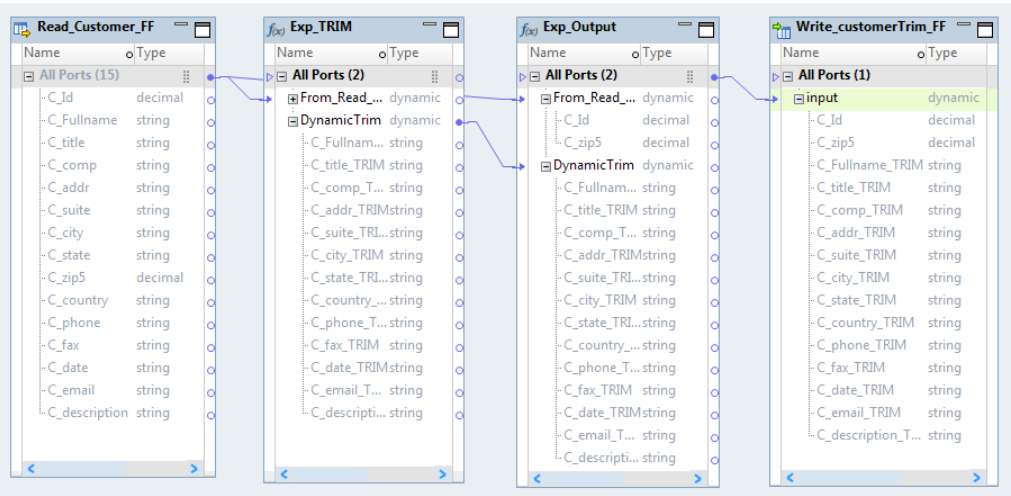


下图显示了 Write_customerTrim_FF 转换中的新动态端口 **input**：



4. 在 Exp_Output 转换中，将“所有端口”组拖动到 Write_customerTrim_FF 转换中的 input 端口。
Developer tool 会创建一个链接，并将 Exp_Output 转换“所有端口”组中的列传送到写入转换的输入动态端口。

下图显示了已配置写入转换的 m_Replication_Template 映射：



步骤 5.验证并保存映射

使用源数据对象和目标文件的默认参数值验证并运行 m_ReplicationTemplate 映射，以查看结果。

1. 在映射编辑器中，单击 **编辑 > 验证**。
2. 如果映射有效，请单击 **文件 > 保存** 以保存映射。

步骤 6. 针对不同的源和目标运行动态映射

在开发动态映射后，可以运行映射以访问不同的源，并根据参数值写入不同的目标。

运行 Customer_FF 源的映射

使用源数据对象和目标文件的默认参数值运行 m_ReplicationTemplate 映射，以查看结果。映射读取 Customer_FF 源文件并写入 customerTrim.csv 目标文件。

1. 单击**运行 > 映射**。

运行映射窗口将显示映射运行的进度。该映射将运行，并将输出写入到目标文件。

2. 要查看写入目标文件的结果，请导航到系统上安装了 Informatica 服务的目标目录：

<Informatica Installation Directory>\tomcat\bin\target

3. 打开 customerTrim.csv 文件，确认字符串值的开头和末尾没有空格。

此文件的每一行按照列在目标对象中的显示顺序列出各列的数据，如 C_Id、C_zip5、C_Fullname、C_title、C_comp 等。例如，此文件的前五行包含以下已删除字符串开头和结尾空格的数据：

```
1,22342,Smith John,Account Executive,DKR MANAGEMENT COMPANY INC,100 High Street,5406,Anytown,TN,USA,
4047668150,2124031386,31/08/1985,bwilliams@yahoo.com,ACTIVE
2,38103,Balasubramanian Krishna,Account Executive,EASTON & COMPANY,71 Congress Parkway,
789,Bangalore,Karnataka,India,4046345228,4151689756,29/10/1985,bmatthewc@univ.edu,ACTIVE
3,55103,Johnson Lars,Regional Sales Exec,GREATER BAY BANCORP,123 Snow St.,43543,St. Paul,MN,USA,
4046581534,6122945948,7/9/1992,ehpuniv.edu,INACTIVE
4,94116,Zogby Kevin,Regional Sales Exec,HEWLETT-PACKARD,317 29th. St.,5856,San Francisco,CA,USA,
4042662730,4155466814,7/8/1985,grobertwuniv.edu,ACTIVE
5,60606,Franklin Roosevelt,Sales Representative,JAYD TRADING,1511 Wacker Dr,6334,Chicago,IL,USA,
7703965851,2065075486,20/10/1982,trichard@univ.edu,INACTIVE
```

更改参数值

更改源数据对象和目标数据对象输出文件名的参数值。

1. 要更改源数据对象的参数值，请执行以下步骤：

a. 在映射的**属性**视图中，单击**参数**选项卡。

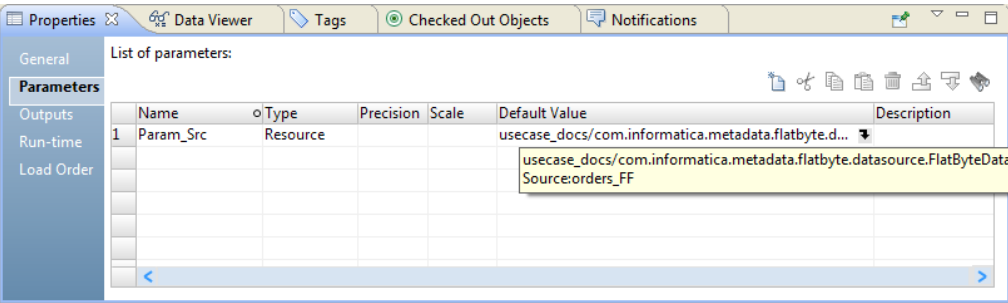
b. 找到源对象的 Param_Src 参数。

c. 在**默认值**列中，单击**打开按钮** (🔗)。

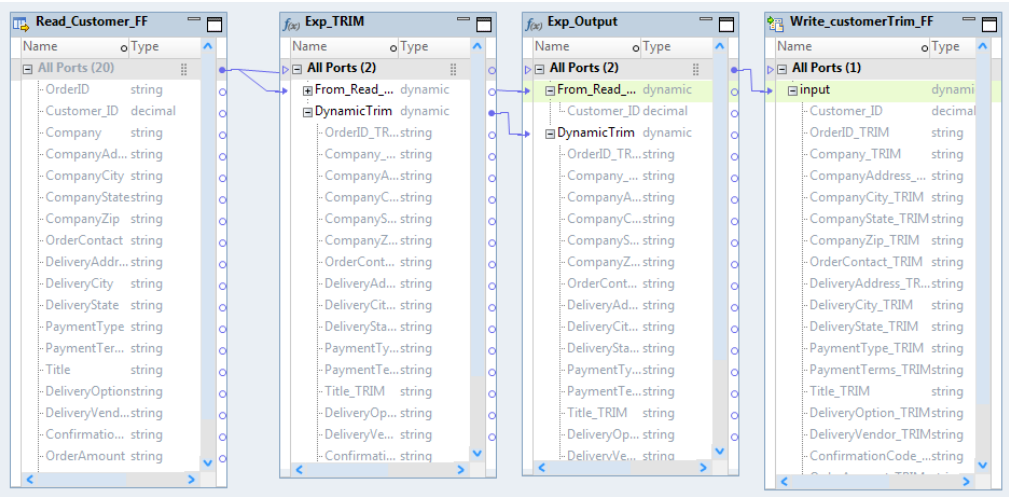
此时将显示**选择位置**对话框。

d. 选择 orders_FF 数据对象。

下图显示了包含已更新默认值的映射的**参数**选项卡：

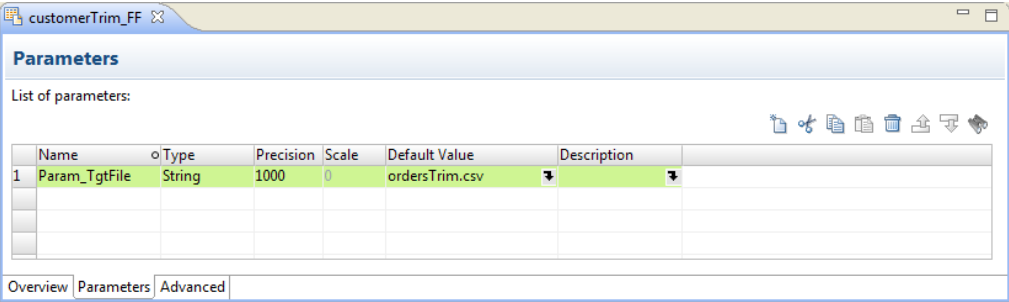


下图显示的映射反映了读取转换的 orders_FF 数据对象中的端口。动态端口反映了新生成的端口。



2. 要更改目标文件名的参数值，请执行以下步骤：
- a. 打开 customerTrim_FF 目标数据对象。
 - b. 在数据对象的参数选项卡中，找到目标文件名的 Param_TgtFile 参数。
 - c. 在默认值列中，单击打开按钮 (🔍)。
 - 此时将显示编辑参数值窗口。
 - d. 将默认文件名值更改为 ordersTrim.csv，然后单击确定。

下图显示了包含已更新默认值的 customerTrim_FF 数据对象的参数选项卡：



运行 orders_FF 源的映射

验证映射并运行不同源和目标的 m_ReplicationTemplate 映射。映射读取 orders_FF 源文件并写入 ordersTrim.csv 目标文件。

- 1. 在映射编辑器中，单击 **编辑 > 验证**。
- 2. 如果映射有效，请单击 **文件 > 保存** 以保存映射。
- 3. 单击 **运行 > 映射**。
- 运行映射**窗口将显示映射运行的进度。该映射将运行，并将输出写入到目标文件。
- 4. 要查看写入目标文件的结果，请导航到系统上安装了 Informatica 服务的目标目录：
 <Informatica Installation Directory>\tomcat\bin\target
- 5. 打开 ordersTrim.csv 文件，确认字符串值的开头和末尾没有空格。

该文件的每一行按照列在目标对象中的显示顺序列出各列的数据，如 Customer_Id、Order_ID、Company、CompanyAddress、CompanyCity 等。例如，此文件的前五行包含以下已删除字符串开头和结尾空格的数据：

```
10110085,0-5079,JOSEPH TAL LYON & ROSS,96 FISHER ROAD,MAHWAH,NJ,7430,PARKE PERSLEY OR RAYFORD LECROY,
96 FISHER ROAD,MAHWAH,NJ,American Express,CHARGE,Account Executive,UPA,United Parcel Service Air,
44162,$21.00,Generic,0/L/B P/W L/S TAWNY SHIMMER .08 OZ.
10110086,0-6658,NRCA,10255 W.HIGGINS RD.,ROSEMONT,IL,60018-5607,ROLANDA SORTO,10255 W.HIGGINS
RD.,ROSEMONT,IL,American Express,CHARGE,Account Executive,UPA,United Parcel Service Air,
44163,$56.40,Generic,0-L.B PW LIPSTYLO LASTING PERFECTION .08 OZ.
10110087,0-8195,POND EQUITIES,4522 FT. HAMILTON PKWY.,BROOKLYN,NY,11219,KONSTANTIN PEDDICORD,4522
FT. HAMILTON PKWY.,BROOKLYN,NY,American Express,CHARGE,Account Executive,UPA,United Parcel Service
Air,44164,$78.00,Generic,0/L/B P/W L/S TAWNY SHIMMER LASTING PERFECTION LIPSTYLO TAWNY SHIMMER .08
OZ.
10110088,0-9130,SCHRODER & COMPANY,787 SEVENTH AVENUE,NEW YORK,NY,10019,GIORGIA TWITCHELL,787
SEVENTH AVENUE,NEW YORK,NY,American Express,CHARGE,Account Executive,UPA,United Parcel Service Air,
44165,$14.00,Generic,A/COL L PERFECTION L/S REF P SUPREME LASTING PERFECTION LIPSTYLO TAWNY SHIMMER .
08 OZ.
10110089,0-9352,YUASA TRADING COMPANY (AMERICA),150 EAST 52ND STREET,NEW YORK,NY,10005,STEFFI
MCGLOWN,150 EAST 52ND STREET,NEW YORK,NY,American Express,CHARGE,Account Executive,UPA,United Parcel
Service Air,44166,$54.00,Generic,0/L/B L PERFECTION REF LIPSTYLO COFFEE PEACH SUPREME .08 OZ.
```

第 9 章

映射管理

本章包括以下主题：

- [映射管理概览, 173](#)
- [查看映射作业的属性, 173](#)
- [查看映射作业的摘要统计信息, 174](#)
- [查看映射作业的详细统计信息, 174](#)
- [查看映射作业的日志, 175](#)
- [重新发出已部署映射作业, 175](#)
- [取消映射作业, 175](#)
- [拒绝文件, 176](#)

映射管理概览

当您运行 Ad Hoc 映射作业或者将映射部署到数据集成服务时，可以在 Monitoring 工具或 Administrator 工具中监视作业。如果数据集成服务无法将行写入到目标，您也可以查看与拒绝文件中的行有关的信息。您必须具有相应的权限才能监视作业或查看拒绝文件。

可以在以下位置监视映射作业：

- 监视工具。在 Developer tool 中，单击**进度**视图中的**菜单**按钮，并选择**监视作业**。选择运行映射作业的数据集成服务，然后单击**确定**。此时将打开 Monitoring 工具。
- Administrator 工具。要在 Administrator 工具中监视映射，单击**监视**选项卡。

监视某个映射作业时，可以查看该作业的摘要统计信息或执行统计信息。**摘要统计信息**视图显示域中映射作业状态的图形概览。可使用**执行统计信息**视图查看映射属性和统计信息、查看作业日志、取消作业或重新发出已部署映射。

查看映射作业的属性

监视 Ad Hoc 映射作业或已部署映射作业时，可以查看作业的属性。属性包括作业 ID、启动作业的用户以及作业的持续时间。

1. 单击**执行统计信息**视图。
2. 在域导航器中，展开一个数据集成服务。

- 3. 选择 **Ad Hoc 作业** 文件夹，或者展开一个应用程序并选择**已部署映射作业**。
此时内容面板中将显示作业列表。内容面板会显示一些属性，例如作业的名称、状态、ID 和持续时间。
- 4. 在内容面板中选择一个作业。
详细信息面板中将显示此作业的属性。

查看映射作业的摘要统计信息

可以查看 Ad Hoc 映射作业或已部署映射作业的吞吐量和资源使用情况统计信息。

- 1. 单击**执行统计信息**视图。
- 2. 在域导航器中，展开一个数据集成服务。
- 3. 选择 **Ad Hoc 作业** 文件夹，或者展开一个应用程序并选择**已部署映射作业**。
此时内容面板中将显示作业列表。
- 4. 在内容面板中选择一个作业。
详细信息面板中将显示此作业的属性。
- 5. 单击详细信息面板中的**摘要统计信息**视图。

摘要统计信息视图会显示源和目标的吞吐量及资源使用情况统计信息。

您也可以按照升序或降序顺序对统计信息进行排序。单击列标题将按升序顺序对列进行排序。再次单击列标题将按降序顺序对列进行排序。

查看映射作业的详细统计信息

可以查看在单独的本地进程中运行的 Ad Hoc 映射作业或已部署映射作业的吞吐量和资源使用情况的图表。将显示运行时间超过一分钟的作业的详细统计信息。

- 1. 单击**执行统计信息**视图。
- 2. 在域导航器中，展开一个数据集成服务。
- 3. 选择 **Ad Hoc 作业** 文件夹，或者展开一个应用程序并选择**已部署映射作业**。
此时内容面板中将显示作业列表。
- 4. 在内容面板中选择一个作业。
详细信息面板中将显示此作业的属性。
- 5. 单击详细信息面板中的**详细统计信息**视图。

详细统计信息视图会显示吞吐量图和资源使用情况图。

您也可以在**详细统计信息**视图中完成以下任务：

任务	说明
放大图形	将光标悬停在图形上，然后单击放大镜图标。
将放大的图形的一部分放大	拖动光标选择要放大的区域。

任务	说明
在吞吐量图中的行数和字节数之间切换	单击“字节数”选项或“行数”选项。
选择在吞吐量图上绘制的统计信息	在吞吐量字段中，选择要查看的源和目标。

查看映射作业的日志

您可以下载作业的日志以查看作业详细信息。

1. 单击**执行统计信息**视图。
2. 在域导航器中，展开一个数据集成服务。
3. 选择 **Ad Hoc 作业** 文件夹，或者展开一个应用程序并选择**已部署映射作业**。
此时内容面板中将显示作业列表。
4. 在内容面板中选择一个作业。
5. 单击**操作 > 查看选定对象的日志**。
此时将显示一个对话框，包含用于打开或保存日志文件的选项。

重新发出已部署映射作业

如果映射作业失败，可以重新发出已部署映射作业。重新发出已部署映射作业后，数据集成服务将重新运行该作业。

1. 单击**执行统计信息**视图。
2. 在域导航器中，展开一个数据集成服务。
3. 展开一个应用程序，然后选择**已部署映射作业**。
内容面板将显示已部署映射作业列表。
4. 选择一个已部署的映射作业。
5. 单击**操作 > 重新发出选定对象**。

取消映射作业

可以取消正在运行的 Ad Hoc 映射作业或已部署映射作业。如果作业停止响应或者所需完成时间过长，您可能希望取消作业。

1. 单击**执行统计信息**视图。
2. 在域导航器中，展开一个数据集成服务。
3. 选择 **Ad Hoc 作业** 文件夹，或者展开一个应用程序并选择**已部署映射作业**。
此时内容面板中将显示作业列表。

- 4. 在内容面板中选择一个作业。
- 5. 单击操作 > 取消选定对象。

拒绝文件

在映射运行过程中，数据集成服务会为映射中的每个目标实例创建一个拒绝文件。如果数据集成服务无法将某行写入到目标，就会把被拒绝的行写入到拒绝文件。拒绝文件和映射日志包含可帮助确定拒绝原因的信息。

如果拒绝文件不包含任何拒绝的行，则数据集成服务会删除映射运行末尾的拒绝文件。

每次运行映射时，数据集成服务都会将拒绝的数据附加到拒绝文件中。根据问题来源，您可以更正映射和目标数据库以阻止后续映射中的拒绝。

拒绝文件的位置

数据集成服务会为映射中的每个目标实例创建拒绝文件。数据集成服务将在目标拒绝文件目录中创建拒绝文件。

在平面文件的运行时属性内或映射的关系目标内配置目标拒绝文件目录。默认情况下，数据集成服务会在 RejectDir 系统参数定义的目录中创建拒绝文件。数据集成服务将在目标实例名称后命名拒绝文件。拒绝文件的默认名称为 <file_name>.bad。

当数据集成服务为目标创建多个分区时，数据集成服务会为每个分区创建一个单独的拒绝文件，其名称为 <file_name><partition_number>.bad。例如，三个分区可能会写入名称分别为 MyOutput1.bad、MyOutput2.bad 和 MyOutput3.bad 的拒绝文件中。

拒绝文件的内容

找到拒绝文件后，您可以使用支持该拒绝文件代码页的文本编辑器读取该文件。

拒绝文件包含由写入器或目标数据库拒绝的数据行。数据集成服务会将整行写入拒绝文件。但是，问题通常集中在该行内的某一列。要帮助确定哪一列导致该行被拒绝，拒绝文件所包含的指示器可提供有关每一列的详细信息。

拒绝文件包含以下指示器：

行指示器

拒绝文件的每一行的第一列为行指示器。行指示器定义了该行是被标记为插入、更新、删除还是拒绝。

列指示器

列指示器在每一数据列后显示。列指示器定义该列是包含有效、溢出、空的还是被截断的数据。

行指示器

拒绝文件中的第一列为行指示器。行指示器是一个定义数据行更新策略的标志。

下表描述了拒绝文件中的行指示器：

行指示器	含义	拒绝者
0	插入	写入器或目标
1	更新	写入器或目标

行指示器	含义	拒绝者
2	删除	写入器或目标
3	拒绝。由更新策略表达式标记为拒绝。	写入器
4	回滚插入	写入器
5	回滚更新	写入器
6	回滚删除	写入器
7	已提交的插入	写入器
8	已提交的更新	写入器
9	已提交的删除	写入器

以下的示例拒绝文件显示了每一行的行指示器“0”，其指示该行的插入更新策略：

```
0,D,1921,D,Nelson,D,William,D,415-541-5145,D
0,D,1922,D,Page,D,Ian,D,415-541-5145,D
0,D,1923,D,Osborne,D,Lyle,D,415-541-5145,D
0,D,1928,D,De Souza,D,Leo,D,415-541-5145,D
0,D,2001123456789,0,S. MacDonald,D,Ira,D,415-541-514566,T
```

列指示器

列指示器会在每一列数据后显示。列指示器定义数据是有效、溢出、为空还是被截断。

下表描述了拒绝文件中的列指示器：

列指示器	数据类型	写入器视为
D	有效数据。	正常数据。写入器会将其传递给目标数据库。除非出现数据库错误（例如找到重复键），否则目标会接受数据。
N	空值。列包含一个空值。	正常数据。写入器将其传递给目标，如果目标数据库不接受空值，则会拒绝该数据。
T	已截断。字符串数据已超过为列指定的精度，因此值已被截断。	错误数据，如果您已将映射目标配置为拒绝溢出或截断数据。

空列会显示在拒绝文件中，其中的列以逗号进行标记。以下示例显示了一个周围均为良好数据的空列：

```
0,D,5,D,,N,5,D
```

列指示器“D”也将每个行指示器后显示。以下示例在行指示器“0”后显示了列指示器“D”：

```
0,D,2001123456789,0,S. MacDonald,D,Ira,D,415-541-514566,T
```

写入器或目标数据库都可以拒绝行。请查看日志以确定拒绝原因。

第 10 章

导出到 PowerCenter

本章包括以下主题：

- [导出到 PowerCenter：概览, 178](#)
- [PowerCenter 版本兼容性, 179](#)
- [Mapplet 导出, 179](#)
- [导出包含参数的映射, 180](#)
- [导出到 PowerCenter：选项, 180](#)
- [将对象导出到 PowerCenter, 181](#)
- [导出限制, 182](#)
- [导出到 PowerCenter：规则和准则, 183](#)
- [导出到 PowerCenter：故障排除, 184](#)

导出到 PowerCenter：概览

可以将 Developer tool 中的对象导出到 PowerCenter® 中使用。

可以导出以下对象：

- 映射。将映射导出为 PowerCenter 映射或 Mapplet。
- Mapplet。将 Mapplet 导出为 PowerCenter Mapplet。
- 逻辑数据对象模型。将逻辑数据对象导出到 PowerCenter Mapplet。

可以将对象导出到 PowerCenter 存储库或 XML 文件。如果将对象导出到 XML 文件，PowerCenter 用户可以将该文件导入到 PowerCenter 存储库。

导出对象时，请指定相应的导出选项，例如 PowerCenter 版本、转换映射和 Mapplet 的方式以及是否导出引用表。

可以导出包含参数的映射和 Mapplet。将映射导入到 PowerCenter 存储库时，参数会解析为默认值。

必须在托管 PowerCenter Express 客户端的计算机上安装 Metadata Converter，才能将对象从 PowerCenter Express 导出到 PowerCenter。

PowerCenter 版本兼容性

要验证对象与特定 PowerCenter 版本的兼容性，请设置 PowerCenter 版本兼容性级别。该兼容性级别将应用于可以在 Developer 工具中查看的所有映射、Mapplet 和逻辑数据对象模型。

可以配置 Developer 工具以针对特定的 PowerCenter 版本进行验证，也可以配置该工具跳过版本兼容性验证。默认情况下，Developer 工具不会针对任何 PowerCenter 版本对对象进行验证。

在将对象导出到 PowerCenter 前，请将兼容性级别设置为某个 PowerCenter 版本。设置兼容性级别之后，Developer 工具将在您验证映射、Mapplet 或逻辑数据对象模型时执行两次验证检查。Developer 工具首先验证对象在 Developer 工具中是否有效。如果对象有效，Developer 工具接下来将验证该对象对于导出到 PowerCenter 选定版本是否有效。您可以在[验证日志](#)视图中查看兼容性错误。

设置兼容性级别

通过设置兼容性级别，可以针对某个 PowerCenter 版本验证映射、Mapplet 和逻辑数据对象模型。如果未选择任何项，Developer 工具将在您验证对象时跳过版本兼容性验证。

1. 单击**编辑 > 兼容性级别**。
2. 选择兼容性级别。

Developer 工具将在菜单中选定的兼容性级别旁边放置一个点。该兼容性级别将应用于可以在 Developer 工具中查看的所有映射、Mapplet 和逻辑数据对象模型。

Mapplet 导出

导出 Mapplet 或将映射导出为 Mapplet 时，导出进程将创建 Mapplet 中的对象。导出进程还将重命名某些 Mapplet 对象。

导出进程可能会在导出 XML 文件中创建以下 Mapplet 对象：

表达式转换

导出进程将立即在 Mapplet 中每个输入转换的下游和每个输出转换的上游创建表达式转换。导出进程将按以下方式命名表达式转换：

Expr_<InputOrOutputTransformationName>

表达式转换包含传递端口。

输出转换

如果导出 Mapplet 并将目标转换为输出转换，导出进程将为每个目标创建输出转换。导出进程将按以下方式命名输出转换：

<MappletInstanceName>_<TargetName>

导出进程将在导出 XML 文件中重命名以下 Mapplet 对象：

Mapplet 输入和输出转换

导出进程将按以下方式命名 Mapplet 输入和输出转换：

<TransformationName>_<InputOrOutputGroupName>

Mapplet 端口

导出进程将按以下方式重命名 Mapplet 端口：

<PortName>_<GroupName>

导出包含参数的映射

您可以导出包含参数的映射或 Mapplet，也可以将其导入到 PowerCenter。

如果导出包含参数的映射或 Mapplet，则当您将其参数导入到 PowerCenter 时，参数会解析为其默认值。导入进程可以解析任何包含参数的 SQL 表达式。

系统参数将解析为等效的 PowerCenter 系统参数。如果 PowerCenter 没有等效的系统参数，则将系统参数引用导入到 PowerCenter 后，该系统参数引用会保留在映射中。您需要编辑映射并更改引用。

不能将映射输出导出到 PowerCenter。如果映射包含映射输出，则将该映射导入到 PowerCenter 后，该映射在其中无效。

导出到 PowerCenter：选项

导出要在 PowerCenter 中使用的对象时，必须指定导出选项。

下表介绍了导出选项：

选项	说明
项目	模型存储库中从中导出对象的项目。
目标版本	PowerCenter 发行版本。
将选定对象导出到文件	将对象导出到 PowerCenter XML 文件。如果选择了此选项，请指定导出 XML 文件的名称和位置。
将选定对象导出到 PowerCenter 存储库	<p>将对象导出到 PowerCenter 存储库。如果选择了此选项，请为 PowerCenter 存储库指定以下连接详细信息：</p> <ul style="list-style-type: none">- 主机名。PowerCenter 域网关主机名。- 端口号。PowerCenter 域网关的 HTTP 端口号。- 身份验证类型。选择以下值之一：Kerberos 单点登录、本地或 LDAP。- 身份验证类型。选择以下值之一：本地或 LDAP。- 用户名。存储库用户名。- 密码。存储库用户名对应的密码。 <p>注意：如果身份验证类型为本地或 LDAP，请指定用户名和密码。</p> <ul style="list-style-type: none">- 安全域。如果身份验证类型为 LDAP，请指定 LDAP 安全域名。否则，请输入“Native”。- 存储库名称。PowerCenter 存储库名称。
发送到存储库文件夹	将对象导出到 PowerCenter 存储库中的指定文件夹。
使用控制文件	使用指定的 <i>pmrep</i> 控制文件将对象导出到 PowerCenter 存储库。
将导出的映射转换为 PowerCenter Mapplet	<p>将开发程序工具映射转换为 PowerCenter Mapplet。</p> <p>开发程序工具将映射中用作源和目标的数据对象转换为 PowerCenter Mapplet 中的输入和输出转换。</p>

选项	说明
转换目标 Mapplet	将 Mapplet 中用作目标的数据对象转换为 PowerCenter Mapplet 中的输出转换。 PowerCenter Mapplet 不能包含目标。如果导出对象包括含有目标的 Mapplet，并且您未选择此选项，导出进程将失败。
导出引用数据	导出供导出对象中的转换使用的所有引用表数据。
引用数据位置	开发程序工具导出的引用表数据的位置。开发程序工具将引用表数据导出为一个或多个字典文件。输入一个指向计算机上托管开发程序工具的目录的路径。
代码页	PowerCenter 存储库的代码页。

将对象导出到 PowerCenter

将映射、Mapplet 或逻辑数据对象模型导出到 PowerCenter 时，可以将对象导出到文件或 PowerCenter 存储库。

在导出对象之前，请将兼容性级别设置为适当的 PowerCenter 版本。验证对象是否与该 PowerCenter 版本兼容。

1. 单击**文件 > 导出**。
此时将显示**导出**对话框。
2. 选择 **Informatica > PowerCenter**。
3. 单击**下一步**。
此时将显示**导出到 PowerCenter**对话框。
4. 选择模型存储库中从中导出对象的项目。
5. 选择要将对象导出到的 PowerCenter 版本。
6. 选择要将对象导出到的位置。可以将对象导出到 PowerCenter 存储库的 XML 文件。
 - 要将对象导出到文件，请指定 XML 文件的名称和位置。
 - 要将对象导出到 PowerCenter 存储库，请单击**浏览**以指定存储库的连接详细信息。
7. 如果导出到 PowerCenter 存储库，请选择 PowerCenter 存储库中的文件夹或 *pmrep* 控制文件以定义将对象导入 PowerCenter 的方式。
8. 选择**将导出的映射转换为 PowerCenter Mapplet** 以将开发程序工具映射转换为 PowerCenter 中的 Mapplet。
9. 选择**转换目标 Mapplet** 以将 Mapplet 中用作目标的数据对象转换为 PowerCenter Mapplet 中的输出转换。
10. 选择**导出引用数据** 以将导出对象中的转换所使用的所有引用表数据导出。
11. 如果要导出引用数据，请指定开发程序工具导出引用表数据的位置。
12. 选择 PowerCenter 存储库的代码页。
13. 单击**下一步**。
开发程序工具将提示您选择要导出的对象。

14. 选择要导出的对象，然后单击**完成**。

开发程序工具会将对象导出到选定的位置。

如果将对象导出到文件，则可以将文件中的对象导入到 PowerCenter 存储库。

如果导出了引用表数据，则可以将引用数据文件复制到计算机上托管 Informatica 服务的 PowerCenter 目录结构中。引用数据文件的位置必须对应于模型存储库中的引用表对象位置。

例如，将引用数据文件复制到以下位置：

<PowerCenter 安装目录>\services\<模型存储库项目名称>\<文件夹名称>

导出限制

将模型存储库对象导出到 PowerCenter 时，某些模型存储库对象可能无法导出到 PowerCenter 存储库。如果映射或 Maplet 中包含任何在 PowerCenter 中无效的对象，则不能将其导出。

不能将以下对象导出到 PowerCenter：

名称过长的对象

PowerCenter 用户不能导入对象名称超出 80 个字符的映射、Maplet 或其中的对象。

包含动态端口的映射或 Maplet

不能导出包含动态端口的映射或 Maplet。

包含数据处理器转换的映射或 Maplet

不能导出包含数据处理器转换的映射或 Maplet。

包含使用系统参数的判定转换的映射或 Maplet

如果映射或 Maplet 包含判定转换且转换脚本包含系统参数，则不能导出该映射或 Maplet。导出操作无法将系统参数转换为 PowerCenter 可以使用的值。在导出包含使用系统参数的判定转换的映射或 Maplet 之前，请使用适当值替换参数。

返回映射输出的映射或 Maplet

如果映射或 Maplet 将返回映射输出，则 PowerCenter 用户不能导入该映射或 Maplet。

包含使用特定联接条件的联接器转换的映射或 Maplet

如果映射和 Maplet 包含的联接器转换使用了在 PowerCenter 中无效的联接条件，则您不能导出该映射和 Maplet。在 PowerCenter 中，用户可基于指定的主源和详细信息源之间的相等性定义联接条件。在 Developer tool 中，可以定义其他联接条件。例如，可以基于主源和详细信息源之间的相等性或不相等性定义联接条件。可以定义包含转换表达式的联接条件。还可以定义联接条件（例如 $1 = 1$ ）以使联接器转换执行交叉联接。

这些类型的联接条件在 PowerCenter 中无效。因此，如果映射或 Maplet 包含的联接器转换使用了这些类型的联接条件，则不能将其导出到 PowerCenter。

包含带有重命名端口的查找转换的映射或 Maplet

PowerCenter 集成服务根据转换和查找条件中的查找端口来查询查找源。因此，查找转换中的端口名称必须与查找源中的列名称相匹配。

包含使用特定自定义 SQL 查询的查找转换的映射或 Maplet

Developer tool 用于在查找转换中验证 SQL 查询语法的规则与 PowerCenter 不同。在 Developer tool 中编写的自定义 SQL 查询使用 AS 关键字或计算出的字段，在 PowerCenter 中无效。如果映射或 Maplet 包含的查找转换带有使用 AS 关键字或计算所得字段的 SQL 查询，则不能将其导出到 PowerCenter。

包含在 PowerCenter 中不可用的源的映射或 Mapplet

如果要导出的映射或 Mapplet 包含在 PowerCenter 中不可用的源，则该映射或 Mapplet 将无法导出。

不能导出包含以下源的映射或 Mapplet：

- 复杂文件数据对象
- DataSift
- Web 内容-Kapow Katalyst

连接端口的 Mapplet

如果要导出的 Mapplet 包含多组输入转换，并且不同输入组中的端口连接到同一下游转换，则导出进程将失败。

包含未连接的查找转换的嵌套 Mapplet

如果要导出的映射或 Mapplet 中包含使用未连接的查找转换的其他 Mapplet，则导出进程将失败。

包含 SAP 源的映射

如果导出的映射包含 SAP 源，Developer tool 将导出映射，而不导出 SAP 源。将映射导入到 PowerCenter 存储库时，PowerCenter 客户端将导入映射，而不导入源。输出窗口将显示一条消息，指示映射无效。必须在 PowerCenter 中手动创建该 SAP 源，并将其添加到映射。

包含 Timestamp with Time Zone 或 Timestamp with Local Time Zone 的映射

如果从 Developer tool 导入的映射包含具有时区的时间戳或具有本地时区的时间戳类型的数据，PowerCenter 客户端将无法转换该映射。

导出到 PowerCenter：规则和准则

由于开发程序工具和 PowerCenter 之间存在差异，某些开发程序工具对象可能与 PowerCenter 不兼容。

将对象导出到 PowerCenter 时请遵循以下规则和准则：

验证 PowerCenter 版本。

验证要从开发程序工具导出的对象在目标 PowerCenter 版本中是否兼容。

验证对象名称是否唯一。

将对象导出到 PowerCenter 存储库时，如果 PowerCenter 对象名称与导出对象名称相同，导出进程将替换该 PowerCenter 对象。

验证代码页是否兼容。

如果开发程序工具和 PowerCenter 使用的代码页不兼容，导出进程将失败。

验证精度模式。

默认情况下，开发程序工具在启用高精度的模式下运行映射或 Mapplet，PowerCenter 在禁用高精度的模式下运行会话。如果采用不同精度模式运行开发程序工具映射和 PowerCenter 会话，会产生不同结果。为了避免产生不同结果，请以相同精度模式运行对象。

复制引用数据。

如果要导出的映射或 Mapplet 中包含使用了引用表的转换，则必须将引用表复制到 PowerCenter 集成服务可以访问的目录。将引用表复制到 INFA_CONTENT 环境变量中所定义的目录。如果未设置 INFA_CONTENT，请将引用表复制到以下 PowerCenter 服务目录：

\$INFA_HOME\services\<开发程序工具项目名称>\<开发程序工具文件夹名称>

导出到 PowerCenter: 故障排除

导出对象名称过长的 Mapplet 时，导出进程失败。

导出 Mapplet 或将映射导出为 Mapplet 时，导出进程将创建或重命名 Mapplet 中的某些对象。导出进程可能会在导出 XML 文件中创建表达式或输出转换。导出进程还将重命名输入转换、输出转换和 Mapplet 端口。

要生成表达式转换的名称，导出进程将在输入和输出转换名称后附加字符。如果导出 Mapplet 并将目标转换为输出转换，导出进程会将 Mapplet 实例名称与目标名称合并以生成输出转换名称。导出进程重命名输入转换、输出转换和 Mapplet 端口时，会将组名称附加到对象名称。

如果现有对象名称过长，导出对象名称可能会超出导出 XML 文件或 PowerCenter 存储库中的对象名称长度限制（最多 80 个字符）。如果对象名称超出了 80 个字符，导出进程将失败并显示内部错误。

如果导出 Mapplet 时导出进程返回了内部错误，请检查输入转换、输出转换、目标和端口的名称。如果名称过长，请将其缩短。

第 11 章

从 PowerCenter 导入

本章包括以下主题：

- [从 PowerCenter 导入：概览, 185](#)
- [替代属性, 186](#)
- [冲突解决方案, 186](#)
- [导入摘要, 186](#)
- [数据类型转换, 187](#)
- [对转换进行转换, 187](#)
- [导入包含参数的映射, 193](#)
- [PowerCenter 存储库连接参数, 194](#)
- [连接分配, 195](#)
- [从 PowerCenter 导入对象, 196](#)
- [导入限制, 197](#)
- [导入性能, 197](#)

从 PowerCenter 导入：概览

可以将对象从 PowerCenter 存储库导入到模型存储库。导入进程会先对 PowerCenter 存储库对象进行验证并将其转换为模型存储库对象，然后再将其导入。

从 PowerCenter 导入对象时，请选择要导入的对象以及模型存储库中的目标位置。导入进程会提供相应选项来解决导入期间的对象名称冲突。

也可以选择将模型存储库中的连接分配给 PowerCenter 对象。可以将一个连接同时分配给多个 PowerCenter 对象。

可以导入包含参数的映射。导入包含可重用转换的映射时，导入进程会导入 PowerCenter 映射参数，并生成转换级别参数，进而将 PowerCenter 映射参数绑定至转换级别参数。如果映射包含不可重用转换，则输入进程将在映射级别创建参数。

导入进程完成后，可以查看导入摘要。

必须在托管 PowerCenter Express 客户端的计算机上安装 Metadata Converter，才能将对象从 PowerCenter 导入到 PowerCenter Express。

替代属性

在导入进程中，可以选择保留或忽略 PowerCenter 对象的替代属性。默认情况下，导入进程会保留 PowerCenter 对象的替代属性。

如果 PowerCenter 映射替代了源属性和目标属性，则导入进程就会创建与 PowerCenter 映射具有相同替代属性值的数据对象。如果 PowerCenter 映射替代了源属性和目标属性，则导入进程会创建一个与 PowerCenter 映射具有相同替代属性值的数据对象。导入进程会在 PowerCenter 对象的名称上附加一个数字，然后创建数据对象。

冲突解决方案

当从 PowerCenter 导入的对象的名称与模型存储库中的某个对象的名称相同时，可以解决对象名称冲突。

可以选择以下冲突解决方案选项：

重命名目标中的对象

使用默认的命名约定重命名 PowerCenter 存储库对象，然后将其导入。默认的冲突解决方案为重命名相应对象。

替换目标中的对象

将模型存储库对象替换为 PowerCenter 存储库对象。

重用目标中的对象

在映射中重用模型存储库中的对象。

重要说明：对于冲突解决方案，模型存储库不会区分映射和 Mapplet。例如，如果导入的 Mapplet 和存储库包含同名映射，系统会提示您解决冲突。如果选择了替换对象，则导入过程会将映射替换为 Mapplet。

导入摘要

将 PowerCenter 对象导入模型存储库后，导入进程会创建导入摘要。

如果存在转换错误，可以将导入摘要保存到文件中。导入摘要包含导入状态、未转换的对象数、导入后无效的对象数以及转换错误。导入后，也可以在 Developer 工具中验证这些对象，以查看验证错误。

数据类型转换

某些 PowerCenter 数据类型在模型存储库中无效。如果导入的 PowerCenter 对象的数据类型无效，则导入进程会在模型存储库中将其转换为可比的有效数据类型。

下表列出了在导入进程中转换为相应模型存储库数据类型的 PowerCenter 存储库数据类型：

PowerCenter 存储库数据类型	模型存储库数据类型
Real	Double
Small Int	Integer
Nstring	String
Ntext	Text

对转换进行转换

导入进程会根据兼容性对 PowerCenter 转换进行转换。某些转换与模型存储库不兼容。其他转换在导入时存在一定限制。

下表介绍了在导入时存在限制的转换或无法导入的 PowerCenter 转换：

PowerCenter 转换	导入操作
汇总器	导入时存在限制。
数据屏蔽	无法导入。
外部过程	无法导入。
HTTP	无法导入。
身份识别	无法导入。
Java	导入时存在限制。
联接器	导入时存在限制。
查找	导入时存在限制。
规范器	导入时存在限制。
等级	导入时存在限制。
序列生成器	导入时存在限制。
排序器	导入时存在限制。
源限定符	导入时存在限制。源和“源限定符”转换将作为一个数据对象导入。

PowerCenter 转换	导入操作
存储过程	无法导入。
事务控制	无法导入。
SQL	导入时存在限制。
联合	导入时存在限制。
非结构化数据	无法导入。
XML 解析器	无法导入。
XML 生成器	无法导入。

转换属性限制

基于转换属性导入某些 PowerCenter 转换时存在限制。

导入进程可能会根据某些转换属性的兼容性执行以下某个操作：

- 忽略。忽略转换属性，导入对象。
- 在内部进行转换。随对象导入转换属性，但 Developer tool 不显示该属性。
- 使导入失败。使对象导入失败，映射无效。

汇总器转换

下表介绍了汇总器转换属性的导入操作：

转换属性	导入操作
转换范围	忽略。

Java 转换

在 Java 转换中，端口必须为输入端口或输出端口。如果 Java 转换既有输入端口，又有输出端口，导入将失败。

下表介绍了 Java 转换属性的导入操作：

转换属性	导入操作
类名称	忽略。
函数标识符	忽略。
生成事务	忽略。
必须阻止输入	忽略。
是否可分区	忽略。
语言	忽略。

转换属性	导入操作
模块标识符	忽略。
输出具有确定性	忽略。
输出具有可重复性	忽略。
要求每个分区一个线程	忽略。
运行时位置	忽略。
更新策略转换	忽略。

连接器转换

下表介绍了连接器转换属性的导入操作：

转换属性	导入操作
主空值排序	在内部进行转换。
空值排序详情	在内部进行转换。
转换范围	在内部进行转换。

查找转换

下表介绍了查找转换属性的导入操作：

转换属性	导入操作
缓存文件名前缀	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
查找缓存初始化	忽略。
查找缓存目录名称	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
启用查找缓存	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
查找数据缓存大小	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
查找索引缓存大小	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
查找源为静态	忽略。
查找 SQL 替代	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入至自定义 SQL 查询。
查找源筛选器	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
预构建查找缓存	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。

转换属性	导入操作
从查找源重新缓存	如果作为独立的转换进行转换则忽略，如果在映射中转换则导入。
陈旧时重新缓存	忽略。
子秒精度	忽略。
同步动态缓存	忽略。

规范器转换

将规范器转换导入到 Developer tool 中时，规范器转换会与一个输入组和至少一个输出组一起导入。

如果导入的规范器转换不属于映射，Developer tool 会将所有输入端口都置于规范器转换的输入组中。Developer tool 将根据输出端口的规范器转换规则创建默认输出组。如果规范器转换中没有要导入的输出端口，Developer tool 会在导入的规范器转换中创建默认输出组。

当规范器转换属于映射时，Developer tool 可能会根据映射中下游转换或目标的链接创建多个输出组。有关多组转换与目标间链接的详细规则和准则信息，请参阅《*Developer 转换指南*》。

导入包含可重用规范器转换的映射时，Developer tool 会将该转换导入为可重用转换，并将映射中的可重用规范器转换实例替换为不可重用的转换实例。Developer tool 会生成从不可重用规范器转换到下游转换和目标的新链接。

在 PowerCenter 中，规范器转换至少具有一个生成的键端口，但在 Developer tool 中，该转换不包含生成的键端口。从 PowerCenter 导入规范器转换时，Developer tool 会忽略生成的键端口。

下表介绍了规范器转换属性的导入操作：

转换属性	导入操作
重置	忽略。
重新开始	忽略。

等级转换

下表介绍了等级转换属性的导入操作：

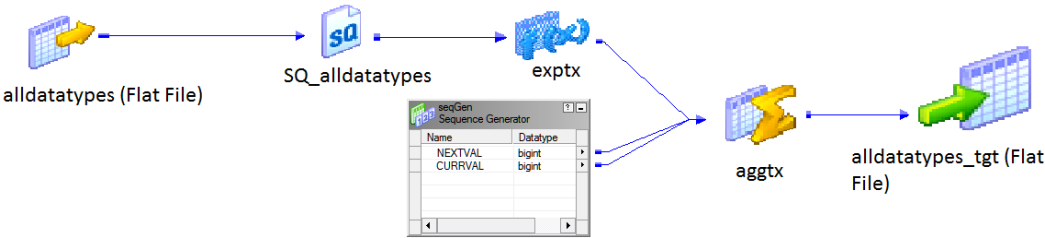
转换属性	导入操作
转换范围	忽略。

序列生成器转换

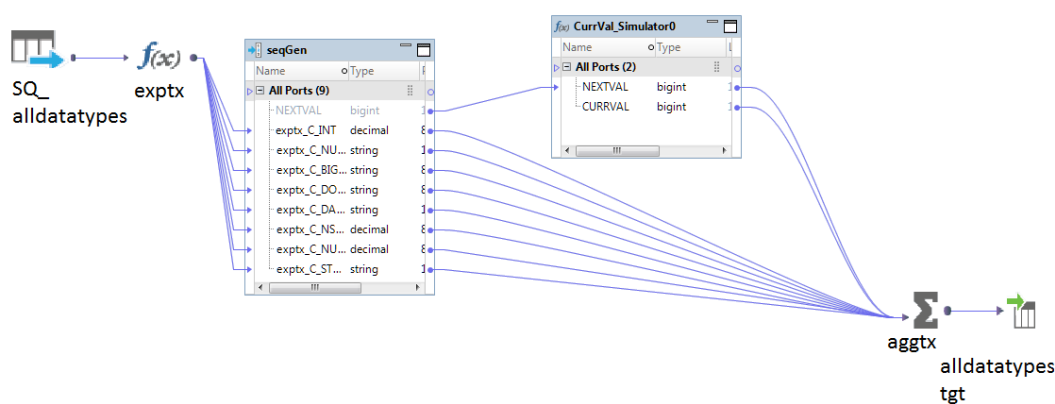
在 PowerCenter 中，序列生成器转换具有以下两个端口：CURRVAL 和 NEXTVAL。在 Developer tool 中，序列生成器转换仅具有一个端口 NEXTVAL。导入映射中的序列生成器转换时，Developer tool 将创建一个表达式转换以设置 CURRVAL 端口值并将其传递至下游转换。

例如，您在 PowerCenter 中的映射具有一个序列生成器转换、一个汇总器转换和一个表达式转换。源数据将依次传递至表达式转换、汇总器转换和目标。序列生成器转换将为每个具有 CURRVAL 端口的行添加一个序列号。

下图显示了具有序列生成器转换、汇总器转换和表达式转换的 PowerCenter 映射：



下图显示了导入到 Developer tool 中的映射：



导入映射时，Developer tool 会将序列生成器转换中的 NEXTVAL 数据传递至表达式转换以设置 CURRVAL 值。

Developer tool 会将映射中的可重用序列生成器转换导入为具有可重用序列数据对象的不可重用转换。如果导入的可重用序列生成器转换不属于映射，Developer tool 会创建一个可重用序列数据对象。

下表介绍了序列生成器转换属性的导入操作：

转换属性	导入操作
当前值	忽略。
缓存值数	忽略。

排序器转换

下表介绍了排序器转换属性的导入操作：

转换属性	导入操作
转换范围	忽略。

源限定符转换

下表介绍了源限定符转换属性的导入操作：

转换属性	导入操作
已排序端口数	忽略。

SQL 转换

下表介绍了 SQL 转换属性的导入操作：

转换属性	导入操作
自动提交	忽略。
类名称	忽略。
连接类型	如果设置为动态连接对象或完整动态连接信息，则使导入失败。
数据库类型	对于 Sybase、Informix 或 Teradata，使导入失败。
函数标识符	忽略。
生成事务	忽略。
必须阻止输入	忽略。
是否可分区	忽略。
语言	忽略。
最大连接池	忽略。
模块标识符	忽略。
输出具有确定性	忽略。
输出具有可重复性	忽略。
要求每个分区一个线程	忽略。
运行时位置	忽略。
SQL 模式	对于脚本模式，使导入失败。
转换范围	忽略。
将数据库连接失败视为致命错误	在内部进行转换。
更新策略转换	忽略。
使用连接池	忽略。

联合转换

下表介绍了联合转换属性的导入操作：

转换属性	导入操作
类名称	忽略。
函数标识符	忽略。
生成事务	忽略。
必须阻止输入	忽略。
是否可分区	忽略。
语言	忽略。
模块标识符	忽略。
输出具有确定性	忽略。
输出具有可重复性	忽略。
要求每个分区一个线程	忽略。
运行时位置	忽略。
转换范围	忽略。
更新策略转换	忽略。

导入包含参数的映射

可以导入包含参数的 PowerCenter 映射或 Mapplet。

当 PowerCenter 映射或 Mapplet 包含参数时，导入进程将在 PowerCenter 映射参数和引用这些参数的可重用转换之间创建参数绑定。导入进程将在转换级别生成参数。

PowerCenter 不需要参数的初始值。导入不包含初始值的 PowerCenter 参数时，导入进程将基于参数数据类型为参数分配默认值。

如果导入包含变量的映射，则导入进程不会转换变量。映射转换或许会成功，但该映射在模型存储库中无效。您可以更改映射，使其在模型存储库中使用参数而不是变量。

部分 PowerCenter 映射参数启用了 IsExprVar 属性。此属性指示 PowerCenter 集成服务必须在解析表达式之前展开参数。IsExprVar 属性在模型存储库中无效。如果导入启用了此参数属性的映射，转换将成功，但该映射无效。

导入含参数化源所有者名称属性的 Netezza 和 Teradata 对象时，导入进程不改变属性。导入进程也忽略 Teradata 映射中的任何参数化目标表名称。

导入包含系统定义的参数的映射

可以导入包含某些系统定义的参数的 PowerCenter 映射或 Mapplet。如果参数与模型存储库中有效的系统定义的参数匹配，则导入进程会导入这些参数。

导入进程将在系统定义的参数和引用这些参数的可重用转换之间创建参数绑定。

如果 PowerCenter 映射包含某个系统定义的参数，而此参数在模型存储库中没有等效的系统定义的参数，则转换不会失败。导入进程将复制映射属性，并使用参数名称作为属性值。但是，导入的映射无效。您可以创建用户定义的参数以替换属性值，或者可以更改映射逻辑。

可以导入以下系统定义的参数：

- \$PMMappingName
- \$PMIntegrationServiceName
- \$PMRepositoryUserName
- \$SESSSTARTTIME
- \$SYSDATE

不能导入以下 PowerCenter 系统定义的参数：

- \$PMFolderName
- \$PMRepositoryServiceName
- \$PMSessionName
- \$PMSessionRunMode
- \$PMTAB_ALL_DATA_TYPES@TableName
- \$PMTGT_ALL_DATA_TYPES@TableName
- \$PMWorkflowName
- \$PMWorkflowRunId
- \$PMWorkflowRunInstanceName

PowerCenter 存储库连接参数

从 PowerCenter 存储库导入对象时，必须指定存储库连接参数。开发程序工具使用导入参数连接至 PowerCenter 存储库。

下表介绍了导入参数：

参数	说明
主机名	PowerCenter 域网关主机名。
端口号	PowerCenter 域网关 HTTP 端口号。
版本号	PowerCenter 发行版本。

参数	说明
身份验证类型	连接到 PowerCenter 存储库需要的用户身份验证类型。选择以下值之一：Kerberos 单点登录、本地或 LDAP。 连接到 PowerCenter 存储库需要的用户身份验证类型。选择以下值之一：本地或 LDAP。 注意: 如果身份验证类型为本地或 LDAP，请指定用户名和密码。
用户名	PowerCenter 存储库用户名。
密码	PowerCenter 存储库用户名的密码。
安全域	如果身份验证类型为 LDAP，请指定 LDAP 安全域名。否则，请输入”本地“。
存储库名称	PowerCenter 存储库名称。
代码页	PowerCenter 存储库的代码页。

连接分配

从 PowerCenter 导入数据源及其他对象时，您可以指定数据源连接类型。

例如，您可以在连接到 Oracle 数据库的 PowerCenter 中创建源定义、目标定义或查找转换。在 Developer tool 中导入这些对象时，请使用**连接分配**对话框为每个对象指定连接类型。

为 PowerCenter 存储库对象选择连接类型时，请使用以下方法之一分配连接：

将单个连接同时分配给多个 PowerCenter 对象。

可以将单个连接分配给未分配连接的所有源、所有目标、所有查找转换或所有对象。也可以将单个连接分配给名称与指定名称模式匹配的所有对象。从**选择**列表中选择一个选项，然后单击**分配连接**。

将单个连接分配给不同对象类型的多个 PowerCenter 对象。

在**选择**列表中选择**自定义**，选择多个 PowerCenter 对象，然后单击**分配连接**。

将连接分配给 PowerCenter 对象。

选择一个 PowerCenter 对象并单击**连接名称**列中的**打开**按钮。

您可以将不同于原始连接类型的连接分配给对象。如果 Developer tool 不支持原始连接类型，您可能需要执行此操作。如果分配不同于原始连接类型的连接，Developer tool 会发出连接类型不匹配的警告。

如果您选择忽略警告并继续操作，导入进程将成功并为导入的数据源分配新连接。如果所选连接的元数据与导入的数据源的架构匹配，则源或目标有效。

注意: 对于使用 Developer tool 不支持的连接类型创建的查找转换，有关原始连接类型的信息将不可用，并且 Developer tool 不会警告连接不匹配。

从 PowerCenter 导入对象

可以将对象从 PowerCenter 存储库导入到模型存储库。

从 PowerCenter 导入对象之前，请先连接到目标模型存储库。

1. 选择**文件 > 导入**。

此时将显示**导入**对话框。

2. 选择 **Informatica > PowerCenter**。

3. 单击**下一步**。

此时将显示**从 PowerCenter 导入**对话框。

4. 指定 PowerCenter 存储库的连接参数。

5. 单击**测试连接**。

Developer tool 将测试与 PowerCenter 存储库的连接。

6. 如果成功连接到 PowerCenter 存储库，请单击**确定**。单击**下一步**。

Developer tool 将显示 PowerCenter 存储库中的文件夹，并提示您选择要导入的对象。

7. 选择要导入的一个或多个对象。

8. 单击**下一步**

9. 在模型存储库中选择导入对象的目标位置。

10. 为对象名称冲突选择一个冲突解决方案选项。选择重命名、替换或重用目标模型存储库中的对象。

- 要使用默认的命名约定重命名 PowerCenter 存储库对象然后将其导入模型存储库，请选择**重命名目标中的对象**选项。默认的冲突解决方案为重命名相应对象。
- 要将模型存储库对象替换为 PowerCenter 存储库对象，请选择**替换目标中的对象**选项。
- 要在映射中重用模型存储库中的对象而不导入 PowerCenter 对象，请选择**重用目标中的对象**选项。

11. 单击**下一步**。

Developer tool 将显示 PowerCenter 对象以及相关对象。

12. 单击**忽略替代属性**以忽略可重用 PowerCenter 源、目标和转换的替代属性。默认情况下，进程会保留替代属性。

13. 如果导入 IBM DB2 对象，请选择 DB2 对象类型。您可以选择以下对象类型之一：LOW、z/OS 或 i5/OS。

14. 单击**下一步**。

15. 指定 PowerCenter 存储库对象的模型存储库连接详细信息。

16. 此时将显示**选择连接**对话框。选择一个连接，然后单击**确定**。

17. 单击**下一步**。

Developer tool 将生成导入摘要并列出发要导入的 PowerCenter 对象以及相关对象。

18. 单击**转换检查**，验证对象能否作为有效的模型存储库对象导入。

Developer tool 将显示转换检查摘要报告，其中包含转换检查的结果。

19. 单击**确定**。单击**完成**。

在导入过程中 Developer tool 将显示进度信息。Developer tool Developer tool 将 PowerCenter 对象和相关对象导入到模型存储库，并生成最终的导入摘要报告。

20. 单击**保存**并指定一个文件名，以便在出现转换错误时保存导入摘要。

导入限制

导入 PowerCenter 对象时，应遵循以下限制：

源和目标

- 如果从 PowerCenter 9.1.0 版或更早版本导入源或目标，则导入进程无法验证与对象关联的连接类型是否有效。
- 如果 PowerCenter 存储库的版本早于 9.5.0，则 IBM DB2 源数据库名称或 IBM DB2 目标名称必须以“DB2”开头，才能设置 DB2 类型。
- 如果平面文件源的行分隔符无效，则导入进程会将其更改为默认值。
- 从 PowerCenter 导入 Teradata 源时，导入进程忽略以下属性：
 - 列的非空属性
 - 输出是确定性的
 - 输出具有可重复性
 - 已排序端口数
 - 外键
- 从 PowerCenter 导入 Teradata 目标时，导入进程忽略以下属性：
 - 列的非空属性
 - 更新替代
 - 目标表前缀
 - 外键
- 从 PowerCenter 导入 Netezza 源时，导入进程忽略以下属性：
 - 外键
 - 输出是确定性的
 - 输出具有可重复性
 - 已排序端口数
- 从 PowerCenter 导入 Netezza 目标时，导入进程忽略以下属性：
 - 外键
 - 更新替代

转换

- 转换中的表达式最多只能包含 4,000 个字符。
- 在导入过程中，SQL 转换或查找转换的数据库类型将转换为 ODBC。
- 如果将转换的数据缓存大小或索引缓存大小设置为无效值，则导入进程会将此值更改为 Auto。

映射

- 一个映射只能包含一个管道。

导入性能

如果要导入的映射大于 68 MB，请通过命令行导入该映射以获得最佳性能。

提示: 可以使用以下命令行选项: ImportFromPC

第 12 章

性能优化

本章包括以下主题：

- [性能优化概览, 199](#)
- [优化方法, 200](#)
- [优化器级别, 203](#)
- [为 Developer tool 映射设置优化器级别, 204](#)
- [为已部署的映射设置优化器级别, 204](#)

性能优化概览

数据集成服务可优化映射以提高其性能。

数据集成服务可以执行以下优化：

筛选数据以减少需要处理的行数。

数据集成服务会应用优化方法来尝试减少要处理的数据量。当运行映射时，您可以选择一种优化器级别，用于确定数据集成服务能够对映射应用的优化方法。例如，数据集成服务可以使用早期选择优化以将筛选器移至更靠近源的位置。可以使用下推优化将转换逻辑推送至数据库。可以使用基于成本的优化方法更改联接处理顺序。

数据集成服务可以对一个映射同时应用多种优化方法。例如，当您选择普通优化器级别时，数据集成服务会应用早期计划、谓词优化、早期选择、分支修剪或推入优化方法。

确定分区策略以最大化并行处理能力。

如果启用了分区选项，则数据集成服务可以实现映射的并行最大化。数据集成服务可动态确定映射的分区策略。分区策略包括分区点的位置、每个管道阶段的最佳分区数以及能在各个分区点对数据进行最佳重新分配的分区类型。有关分区的详细信息，请参阅 [“已分区的映射概览” 页面上 227](#)。

数据集成服务会应用优化方法来尝试减少要处理的数据量。当运行映射时，您可以选择一种优化器级别，用于确定数据集成服务能够对映射应用的优化方法。例如，数据集成服务可以使用早期选择优化以将筛选器移至更靠近源的位置。可以使用下推优化将转换逻辑推送至数据库。可以使用基于成本的优化方法更改联接处理顺序。

数据集成服务可以对一个映射同时应用多种优化方法。例如，当您选择普通优化器级别时，数据集成服务会应用早期计划、谓词优化、早期选择、分支修剪或推入优化方法。

也可以设置映射中关系源、逻辑数据对象、物理数据对象和虚拟表的约束，以筛选不必要的行。数据集成服务可以处理约束，以提高映射性能。

优化方法

数据集成服务应用优化方法以减少映射中要处理的行数。您可以配置映射的优化器级别以限制数据集成服务应用的优化方法。

数据集成服务可以应用以下优化方法：

- 下推优化
- 早期计划优化
- 早期选择优化
- 分支修剪优化
- 推入优化
- 谓词优化
- 全局谓词优化
- 基于成本的优化
- 船舶数据中心联接优化
- 半联接优化

数据集成服务可以对一个映射同时应用多种优化方法。例如，当您选择普通优化器级别时，数据集成服务会应用早期计划优化、谓词优化、全局谓词优化、分支修剪优化、早期选择优化或推入优化等方法。

早期计划优化方法

数据集成服务应用早期计划优化方法时，将标识未使用的端口并删除这些端口之间的链接。

早期计划优化方法通过减少数据集成服务跨转换移动的数据量来提高性能。数据集成服务处理映射时，会将来自映射中所有已连接端口的数据从一个转换移至另一个转换。在复杂的大型映射中，或者在使用嵌套 Mapplet 的映射中，某些端口可能不向目标提供数据。数据集成服务会标识不向目标提供数据的端口。数据集成服务标识未使用的端口后，将从映射中删除所有未使用的端口之间的链接。

数据集成服务不删除所有链接。例如，数据集成服务不删除以下链接：

- 连接到有副作用的转换的链接。
- 连接到可以调用 `ABORT()` 或 `ERROR()` 函数、发送电子邮件或调用存储过程的转换的链接。

如果数据集成服务确定转换中的所有端口都未使用，则将删除指向包含最少数据的端口的链接外的所有转换链接。数据集成服务不会从映射中删除未使用的转换。

Developer 工具默认启用此优化方法。

早期选择优化方法

数据集成服务应用早期选择优化方法时，将拆分、移动或删除映射中的筛选器转换。数据集成服务会将筛选器移至映射上方，更靠近源的位置。

如果筛选条件是一个连词，数据集成服务可能会拆分筛选器转换。例如，数据集成服务可能会将筛选条件“`A>100 AND B<50`”拆分为两个更简单的条件“`A>100`”和“`B<50`”。数据集成服务拆分筛选器时，会将简化的筛选器移至映射管道上方，更靠近源的位置。拆分筛选器时，数据集成服务会分别将筛选器移至管道上方。

当您在 Developer tool 中选择了普通或完整优化器级别时，将默认启用早期选择优化方法。如果筛选器转换前面显示的转换具有副作用，数据集成服务会忽略早期选择优化。数据集成服务无法确定 SQL 转换、Web 服务使用者转换和 Java 转换是否具有副作用。如果这些转换没有副作用，则可为其配置早期选择优化。

如果优化未提高性能，可以禁用早期选择。默认情况下，数据集成服务会启用此优化方法。

分支修剪优化方法

数据集成服务可以对不向映射中的目标提供任何行的转换应用分支修剪优化方法。

如果筛选条件对数据行的计算结果为 FALSE，则数据集成服务可能会删除筛选器转换。例如，一个映射有两个筛选器转换，可从两个关系源中筛选数据。一个筛选器转换的筛选条件为 Country=US，另一个筛选器转换的筛选条件为 Country=Canada。联合转换可联接两个关系源，且筛选条件为 Country=US。数据集成服务可能会从映射中删除筛选条件为 Country=Canada 的筛选器转换。

在您选择普通或完整优化器级别时，Developer 工具默认启用分支修剪优化方法。如果通过将优化器级别设置为最低或无仍无法提高优化性能，则您可以禁用分支修剪。

谓词优化方法

数据集成服务应用谓词优化方法时，将检查映射生成的谓词表达式。数据集成服务将确定其能够简化还是重写表达式以提高映射性能。

数据集成服务运行映射时，将针对映射源生成查询，并根据映射内部的映射逻辑和转换针对查询结果执行操作。查询和操作通常包括谓词表达式。谓词表达式表示数据必须满足的条件。谓词表达式的示例包括筛选器和联接器转换中的筛选器和联接条件。

通过谓词优化方法，数据集成服务还将尝试尽可能早地在映射中应用谓词表达式，以提高映射性能。

数据集成服务根据现有谓词表达式推理关系并创建新的谓词表达式。例如，某个映射中包含具有联接条件“A=B”的联接器转换和具有筛选条件“A>5”的筛选器转换。数据集成服务可以将“B>5”添加到联接条件。

数据集成服务在能够对映射同时应用谓词优化方法和早期选择优化方法时应用这两种方法。例如，数据集成服务通过谓词优化方法创建新的筛选条件时，还会尝试通过早期选择方法将其移至映射中的上游。与单独应用其中任何一种方法相比，同时应用这两种优化方法可以提高映射性能。

如果应用程序提高了性能，数据集成服务将应用谓词优化方法。如果应用程序更改了映射结果或者降低了映射性能，数据集成服务将不应用此方法。数据集成服务默认应用此优化方法。

基于成本的优化方法

使用基于成本的优化方法时，数据集成服务将评估映射，生成语义上等价的映射，并运行具有最佳性能的映射。基于成本的优化可缩短执行相邻内部联接操作和完整外部联接操作的映射的运行时间。

语义上等价的映射是指执行相同的功能并生成相同结果的映射。要生成语义上等价的映射，数据集成服务会将原始映射分为多个片段。数据集成服务然后确定其能够优化的映射片段。

优化过程中，数据集成服务可能会添加、删除片段中的转换或者对其进行重新排序。数据集成服务验证已优化片段是否与原始片段生成相同的结果，并生成使用已优化片段的备用映射。

如果数据集成服务确定经过排序的合并联接性能高于嵌套循环联接性能，则它还可以应用经过排序的合并联接。执行联接之前，经过排序的合并联接会使用排序顺序来排列两个数据集。嵌套循环联接则使用嵌套循环来联接两个数据集。数据集成服务可能会使用源中的排序信息，或者如果数据排序的成本低于嵌套循环联接处理的成本，则它可能会创建排序器转换。

数据集成服务生成的全部或大部分映射在语义上都等同于原始映射。数据集成服务使用剖析统计信息或数据库统计信息计算原始映射和每个备用映射的成本。然后标识运行最快的映射。数据集成服务针对最佳备用映射执行验证检查以确保其有效，并生成与原始映射相同的结果。

数据集成服务在内存中缓存最佳备用映射。运行映射时，数据集成服务将检索备用映射并运行该映射，而非运行原始映射。

Developer tool 默认不启用此方法。

船舶数据中心联接优化方法

船舶数据中心联接优化方法会尝试定位与较大数据集相邻的较小数据集，以缩短联接处理时间。当两个表的大小相差很大时，数据集成服务会尝试应用船舶数据中心联接优化方法。

例如，数据集成服务可以应用船舶数据中心联接优化方法，以将包含 10,000 行的主表与包含 1,000,000 行的详情表相联接。数据集成服务将在包含较大的详情表的数据库中创建一个临时的暂存表，以执行船舶数据中心联接。然后，数据集成服务会将较小的主表复制到一个临时表，并将该临时表中的数据与较大的详情表中的数据相联接。数据集成服务执行联接操作后，联接器转换逻辑会在数据库中进行处理。

应用船舶数据中心联接优化方法之前，数据集成服务将执行分析以确定船舶数据中心联接优化是否可行且值得执行。如果分析确定此方法可能会提高性能，数据集成服务将对映射应用此方法。数据集成服务随后将重新分析映射，以确定是否有其他执行船舶数据中心联接优化的机会。如果合适，则会执行其他优化。

Developer tool 默认不启用此方法。

提高性能的船舶数据中心联接要求

船舶数据中心联接优化方法有时无法提高性能。以下因素会影响通过船舶数据中心联接优化实现的映射性能：

- 联接器转换主源具有的行数必须远少于详细信息源。
- 详细信息源必须足够大，这样才有必要进行优化。如果详细信息源不够大，则数据集成服务会发现从主源和详细信息源读取所有数据而不应用船舶数据中心联接优化方法速度更快。

船舶数据中心联接优化规则和准则

如果转换满足以下要求，则数据集成服务可以对联接器转换应用船舶数据中心联接优化：

- 联接类型必须为普通、主外联接或详细外联接。
- 详细信息管道必须源自关系源。
- 如果映射使用基于目标的提交，则联接器转换范围必须为“全部输入”。
- 主管道和详细信息管道不能共享任何转换。
- 映射在详细信息源与联接器转换之间不能包含分支。
- 如果包含联接详细信息端的数据库为不支持 Unicode 编码的 IBM DB2 数据库，数据集成服务将无法应用船舶数据中心联接优化方法。

半联接优化方法

半联接优化方法尝试通过修改映射中的联接操作来减少从源中提取的数据量。

在以下情况下，数据集成服务将对联接器转换应用半联接优化方法：一个输入组具有的行数远大于其他输入组，而且根据联接条件，较大组中的许多行在较小组中都没有匹配行。数据集成服务尝试通过读取较小组中的行、查找较大组中匹配的行并执行联接操作来减小一个联接操作数的数据集的大小。减小数据集的大小可提高映射性能，因为数据集成服务不再从较大的组源中读取不必要的行。数据集成服务将联接条件移至较大的组源，并仅读取与较小的组匹配的行。

应用半联接优化方法之前，数据集成服务将执行分析以确定半联接优化方法是否可行且值得执行。如果分析确定此方法可能会提高性能，数据集成服务将对映射应用此方法。数据集成服务随后分析映射，以确定是否有其他执行半联接优化的机会。如果合适，则会执行其他优化。

Developer tool 默认不启用此方法。

已提高性能的半联接优化要求

半联接优化方法有时无法提高性能。以下因素影响通过半联接优化实现的映射性能：

- 联接器转换主源具有的行数必须远少于详细信息源。
- 详细信息源必须足够大，以弥补优化。数据集成服务应用半联接优化时，此方法会将部分开销时间添加到映射处理过程。如果详细信息源非常小，应用半联接方法所需的时间可能会超过处理详细信息源中的所有行所需的时间。
- 数据集成服务必须能够为联接器转换获取源行计数统计信息，以便正确地比较常规联接操作与半联接操作的时间要求。

半联接优化规则和准则

如果转换满足以下要求，数据集成服务可以对联接器转换应用半联接优化：

- 联接类型必须为普通、主外联接或详细外联接。联接器转换无法执行完整外联接。
- 详细信息管道必须源自关系源。
- 联接条件必须是有效的“排序-合并-联接”条件。即，每个子句必须是一个主端口和一个详细信息端口的等式。如果存在多个子句，这些子句必须通过 AND 联接。
- 如果映射不使用基于目标的提交，则联接器转换范围必须为“所有输入”。
- 主管道和详细信息管道不能共享任何转换。
- 映射在详细信息源与联接器转换之间不能包含分支。

查看优化映射

您可以查看一个优化映射来确定优化方法如何影响映射。

- ▶ 右键单击编辑器中的空白区域，然后单击**显示优化映射**。
数据集成服务会生成优化映射。

优化器级别

数据集成服务根据您配置的优化器级别优化映射。如果希望映射使用除了普通级别之外的优化器级别，那么您可以配置优化器级别。默认情况下，每个映射都使用普通优化器级别。

您可以选择以下优化器级别之一：

无

数据集成服务不应用任何优化。

最小

数据集成服务应用早期计划优化方法。

普通

数据集成服务将应用早期计划、早期选择、分支修剪、推入、全局谓词优化和谓词优化等方法。“普通”为默认优化级别。

完整

数据集成服务将应用基于成本的优化、早期计划、早期选择、分支修剪、谓词、推入、半联接和船舶数据中心联接优化等方法。

从**运行**菜单或 Developer tool 中的映射编辑器运行映射时，数据集成服务会应用普通优化器级别。从**运行**菜单运行映射时，数据集成服务将应用映射配置中的优化器级别。从命令行运行映射时，数据集成服务在应用程序中应用映射部署属性中的优化级别。

注意：数据集成服务不会将下推优化方法与优化器级别一起应用。您可以在映射运行时属性中为映射配置下推优化。

相关主题：

- [“下推优化概览” 页面上 205](#)

为 Developer tool 映射设置优化器级别

如果通过“运行”菜单或映射编辑器运行映射，则 Developer tool 将以普通优化器级别运行映射。要以不同的优化器级别运行映射，请从**运行配置**对话框运行映射。

1. 打开映射。
2. 选择**运行 > 打开运行对话框**。
此时将显示**运行配置**对话框。
3. 选择包含要应用的优化器级别的映射配置，或创建一个映射配置。
4. 单击**高级**选项卡。
5. 更改优化器级别。
6. 单击**应用**。
7. 单击**运行**以运行映射。

Developer tool 将以所选映射配置中的优化器级别运行映射。

为已部署的映射设置优化器级别

通过在应用程序中更改映射的部署属性，可以设置从命令行运行映射时要使用的优化器级别。

映射必须位于一个应用程序中。

1. 打开包含映射的应用程序。
2. 单击**高级**选项卡。
3. 选择优化器级别。
4. 保存应用程序。

更改优化器级别后，必须重新部署应用程序。

第 13 章

下推优化

本章包括以下主题：

- [下推优化概览, 205](#)
- [下推类型, 205](#)
- [转换逻辑下推, 207](#)
- [到源的下推优化, 208](#)
- [下推优化表达式, 212](#)
- [比较数据集成服务的输出和源, 226](#)

下推优化概览

数据集成服务应用下推优化时，会将转换逻辑推送到源数据库。数据集成服务会将转换逻辑转换为 SQL 查询，并将 SQL 查询发送至数据库。源数据库运行 SQL 查询以处理转换。

如果源数据库处理转换逻辑的速度能够比数据集成服务快，下推优化可提高映射性能。数据集成服务从源中读取的数据也较少。

数据集成服务推送至源数据库的转换逻辑量取决于数据库、转换逻辑和映射配置。数据集成服务将处理所有无法推送至数据库的转换逻辑。

为映射配置下推优化时，数据集成服务会通过从源到目标或者直至到达无法推送至源数据库的下游转换来分析优化映射。数据集成服务将为每个下推了转换逻辑的源生成并执行 SELECT 语句。然后，会读取此 SQL 查询的结果并处理映射中的剩余转换。

相关主题：

- [“优化器级别” 页面上 203](#)

下推类型

在映射运行时属性中选择了下推类型后，数据集成服务会将下推优化应用到该映射。

您可以选择下列下推类型：

- 无。此选项表示为映射选择的是“无”下推类型。
- 源。数据集成服务会尝试将尽可能多的转换逻辑下推至源数据库。

- 完整。数据集成服务会将全部转换逻辑推送至源数据库。

还可为下推类型创建字符串参数并使用以下参数值：

- Full
- 源
- 无

完整下推优化

当应用完整下推优化时，数据集成服务会将映射中的全部转换逻辑推送至源数据库。您可以在映射运行时属性中配置完整下推。

当源和目标位于同一个数据库，或者诸如汇总器转换和筛选器转换等转换在源数据库中进行处理并减少了移动的数据量时，完整下推优化是理想的优化方法。例如，如果映射包含 Teradata 源和 Teradata 目标，则您可以配置完整下推优化，以将 Teradata 源数据库中所有待处理的转换逻辑推送至 Teradata 目标数据库。

如果对包含更新策略转换的映射配置完整下推，则您必须确定该映射的下推兼容性。

在以下情况下，数据集成服务可以下推包含更新策略转换的映射：

- 连接到更新策略转换的目标转换接收多个没有相同键的行。
- 连接到更新策略转换的目标转换接收多个具有相同键且可重新排序的行。

在以下情况下，数据集成服务无法下推包含更新策略转换的映射：

- 连接到更新策略转换的目标转换接收多个具有相同键但不可重新排序的行。

您还可以在映射中使用下推兼容性参数。可以使用以下参数值：

- noMultipleRowsWithSameKeyOnTarget
- reorderAllowedForMultipleRowsWithSameKey
- reorderNotAllowedForRowsWithSameKey

数据集成服务可对以下源使用完整下推优化：

- Oracle
- IBM DB2
- Microsoft SQL Server
- Teradata
- Netezza
- Greenplum
- SAP HANA

完整下推优化的规则和准则

配置完整下推优化时请考虑以下规则和准则：

- 数据集成服务可以将映射中的所有转换逻辑推送到 IBM DB2、Oracle、Microsoft SQL Server 和 ODBC 源，例如 Teradata、Greenplum、Netezza 和 SAP HANA。
- 当您为包含更新策略转换的映射配置完整下推优化时，只能对 Oracle 和 Teradata 应用“更新否则插入”策略。

源下推

当应用源下推时，数据集成服务会分析源到目标的映射，在到达下游转换之前，它无法推送至源数据库。

数据集成服务将基于可以推送至数据库的每个转换的转换逻辑来生成和执行 SELECT 语句。然后，会读取此 SQL 查询的结果并处理剩余转换。

如果源和目标驻留在不同的数据库中，您可以将映射配置为使用源下推。例如，如果映射包含 Teradata 源和 Oracle 目标，则您可以配置源下推，以将部分待处理的转换逻辑推送至 Teradata 源。

配置下推

您可以在映射运行时属性中配置一个要进行下推优化的映射。

1. 打开一个映射。
2. 在**属性**选项卡上，选择**运行时**。
3. 选择一个下推类型或分配一个下推参数：
 - **无**。数据集成服务不会将映射逻辑下推至源数据库。
 - **完整**。数据集成服务会将全部映射逻辑推送至源数据库。
 - **源**。数据集成服务会将除目标以外的全部映射逻辑推送至源数据库。
 - **分配参数**。选择您为下推类型配置的参数或者创建一个新参数并单击**确定**。
4. 或者，如果您选择完整下推优化并且映射包含更新策略转换，则可以选择一个下推兼容性选项或者分配一个下推兼容性参数。
 - **多个行不具有相同键**。连接到更新策略转换的目标转换收到具有相同键的多个行。数据集成服务可以下推更新策略转换。
 - **可对具有相同键的多个行进行重新排序**。连接到更新策略转换的目标转换收到可以重新排序并具有相同键的多个行。数据集成服务可以下推更新策略转换。
 - **无法对具有相同键的多个行进行重新排序**。连接到更新策略转换的目标转换收到无法重新排序并具有相同键的多个行。数据集成服务无法下推更新策略转换。
 - **分配参数**。选择您为下推兼容性配置的参数或者创建一个参数并单击**确定**。

转换逻辑下推

数据集成服务会使用下推优化将转换逻辑推送至源数据库。数据集成服务推送至源数据库的转换逻辑量取决于数据库、转换逻辑和映射配置。数据集成服务将处理所有无法推送至数据库的转换逻辑。

数据集成服务可以将以下转换逻辑推送至源数据库：

- 汇总器
- 表达式
- 筛选器
- 联接器
- 查找
- 排序器
- 联合

在以下情况下，数据集成服务无法将转换逻辑推送至源：

- 源包含一个使用二进制数据类型的列。
- 源是自定义数据对象，其中包含用于表达式转换逻辑或联接器转换逻辑的筛选条件或用户定义联接。
- 源位于不同的数据库管理系统上，或者为联接器转换逻辑或联合转换逻辑使用不同的连接。

数据集成服务会处理无法推送至源的映射逻辑。

到源的下推优化

数据集成服务可以将转换逻辑推送至不同的源，例如关系源和使用数据库特定 ODBC 驱动程序的源。数据集成服务推送的转换逻辑类型取决于源类型。

数据集成服务可以将转换逻辑推送至以下类型的源中：

- 关系源
- 使用本地数据库驱动程序的源
- PowerExchange® 非关系源
- 使用数据库特定 ODBC 驱动程序的源
- SAP 源

到关系源的下推优化

数据集成服务可以将转换逻辑推送至使用本地驱动程序或数据库特定 ODBC 驱动程序的关系源。

数据集成服务可以将汇总器、表达式、筛选器、联接器、排序器和联合转换逻辑推送至以下关系源：

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

将汇总器转换逻辑推送至关系源时，如果传递端口是分组端口，则传递端口有效。转换语言包括可以在汇总器转换中使用的汇总函数。

下表显示了在 IBM DB2 关系源中有效的汇总函数：

汇总函数	DB2-LUW	DB2i	DB2z/os
AVG	是	是	是
COUNT	是	是	是

汇总函数	DB2-LUW	DB2i	DB2z/os
FIRST	否	否	否
LAST	否	否	否
MAX	是	是	是
MEDIAN	否	否	否
MIN	是	是	是
PERCENTILE	否	否	否
STDDEV	是	是	是
SUM	是	是	是
VARIANCE	是	是	是

下表显示了在 Greenplum、Hive、MSSQL、Oracle、Sybase 和 Teradata 关系源中有效的汇总函数：

汇总函数	Greenplum	Hive	MSSQL	Oracle	Sybase	Teradata
AVG	是	是	是	是	是	是
COUNT	是	是	是	是	是	是
FIRST	否	否	否	否	否	否
LAST	否	否	否	否	否	否
MAX	是	是	是	是	是	是
MEDIAN	否	否	否	是	否	否
MIN	是	是	是	是	是	是
PERCENTILE	否	否	否	否	否	否
STDDEV	是	是	是	是	否	是
SUM	是	是	是	是	是	是
VARIANCE	是	是	是	是	否	是

下表显示了在 MSSQL、Oracle 和 Sybase 关系源中有效的汇总函数：

汇总函数	MSSQL	Oracle	Sybase
AVG	是	是	是
COUNT	是	是	是

汇总函数	MSSQL	Oracle	Sybase
FIRST	否	否	否
LAST	否	否	否
MAX	是	是	是
MEDIAN	否	是	否
MIN	是	是	是
PERCENTILE	否	否	否
STDDEV	是	是	否
SUM	是	是	是
VARIANCE	是	是	否

关系源具有用于处理空值的默认配置。默认情况下，一些数据库会将空值视为低于其他任何值，而另一些数据库会将空值视为高于其他任何值。可以将排序器转换逻辑推送至关系源，并在源使用默认空值排序时获取准确结果。

如果为不同的输出行配置排序器转换，则必须启用区分大小写的排序才能将转换逻辑推送至 DB2、Sybase、和 Oracle 源。

数据集成服务无法将任何包含十进制数据类型的函数推送到 Hive 源。

到本地源的下推优化

如果数据集成服务将转换逻辑推送至使用本地驱动程序的关系源，则数据集成服务将生成使用本地数据库 SQL 的 SQL 语句。

数据集成服务可以将汇总器、表达式、筛选器、联接器、排序器和联合转换逻辑推送至以下本地源：

- IBM DB2 for Linux、IBM DB2 for UNIX 和 IBM DB2 for Windows（“DB2 for LUW”）
- Microsoft SQL Server。如果在 Windows 上运行数据集成服务，则数据集成服务可以使用到 Microsoft SQL Server 的本地连接。
- Oracle

数据集成服务可以将筛选器转换逻辑推送至以下本地源中：

- IBM DB2 for i5/OS
- IBM DB2 for z/OS

到 PowerExchange 非关系源的下推优化

对于 z/OS 系统上的 PowerExchange 非关系数据源，数据集成服务会将筛选器转换逻辑推送至 PowerExchange。PowerExchange 会将逻辑转换成源可以处理的查询。

数据集成服务可以为以下类型的非关系源推送筛选器转换逻辑：

- IBM IMS

- 序列数据集
- VSAM

到 ODBC 源的下推优化

数据集成服务可以将转换逻辑推送至使用特定于数据库的 ODBC 驱动程序的数据库。如果选择**其他**作为 ODBC 提供程序，则数据集成服务无法将转换逻辑推送至源。

使用特定于数据库的 ODBC 驱动程序连接到源时，数据集成服务会使用本地数据库 SQL 生成 SQL 语句。

可以在 ODBC 连接对象中指定 ODBC 提供程序。

可以为以下 ODBC 连接类型配置特定的 ODBC 提供程序：

- Greenplum
- Microsoft SQL Server
- Netezza
- SAP HANA
- Sybase ASE
- Teradata

到 SAP 源的下推优化

对于包含列名称、运算符和文字字符串的表达式，数据集成服务可以将筛选器转换逻辑推送至 SAP 源。如果数据集成服务将转换逻辑推送至 SAP，则将表达式中的文字字符串转换为 SAP 数据类型。

如果 TO_DATE 将 DATS、TIMS 或 ACCP 数据类型的字符串转换为以下日期格式之一，则数据集成服务可以推送包含 TO_DATE 函数的筛选器转换逻辑：

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

如果将 TO_DATE 函数应用于除 DATS、TIMS 或 ACCP 之外的数据类型，或者如果 TO_DATE 将字符串转换为数据集成服务无法推送至 SAP 的格式，则数据集成服务将处理转换逻辑。数据集成服务将处理包含其他 Informatica 函数的转换逻辑。数据集成服务将处理包含其他 Informatica 函数的转换逻辑。

筛选器转换表达式可以包括多个由 AND 或 OR 分隔的条件。如果条件适用于多个 SAP 表，则数据集成服务可以在 SAP 数据对象使用 Open SQL ABAP 联接语法时，将转换逻辑推送至 SAP。在 SAP 数据对象的读取操作中配置 Select 语法模式。

SAP 数据类型例外

当源无法处理筛选器转换的转换逻辑而且转换表达式包含以下数据类型时，数据集成服务将处理该转换逻辑：

- RAW
- LRAW
- LCHR

下推优化表达式

如果转换包含源支持的运算符和函数，则数据集成服务可以将转换逻辑推送至源数据库。数据集成服务通过确定数据库中的等价运算符和函数将转换表达式转换成查询。如果不存在等价运算符或函数，则数据集成服务将处理转换逻辑。

如果源使用 ODBC 连接，并且在 ODBC 连接对象中配置了特定于数据库的 ODBC 提供程序，则数据集成服务会将源视为本地源类型。

函数

Informatica 函数对 z/OS 上的非关系源不可用。下表显示了可用于 IBM DB2 源的下推优化的 Informatica 函数：

函数	DB2 for i5/OS ¹	DB2 for LUW	DB2 for z/OS ¹
ABORT()	否	否	否
ABS()	否	是	否
ADD_TO_DATE()	是	是	是
AES_DECRYPT()	否	否	否
AES_ENCRYPT()	否	否	否
ASCII()	是	是	是
AVG()	是	是	是
CEIL()	是	是	是
CHOOSE()	否	否	否
CHR()	否	是	否
CHRCODE()	否	是	是
COMPRESS()	否	否	否
CONCAT()	是	是	是
COS()	是	是	是
COSH()	是	是	是
COUNT()	是	是	是
CRC32()	否	否	否
CREATE_TIMESTAMP_TZ()	否	否	否
CUME()	否	否	否
DATE_COMPARE()	是	是	是

函数	DB2 for i5/OS ¹	DB2 for LUW	DB2 for z/OS ¹
DATE_DIFF()	否	否	否
DECODE()	否	是	否
DECODE_BASE64()	否	否	否
DECOMPRESS()	否	否	否
ENCODE_BASE64()	否	否	否
ERROR()	否	否	否
EXP()	否	是	否
FIRST()	否	否	否
FLOOR()	否	是	否
FV()	否	否	否
GET_DATE_PART()	是	是	是
GET_TIMESTAMP()	否	否	否
GET_TIMEZONE()	否	否	否
GREATEST()	否	否	否
IIF()	否	是	否
IN()	否	是	否
INDEXOF()	否	否	否
INITCAP()	否	否	否
INSTR()	是	是	是
IS_DATE()	否	否	否
IS_NUMBER()	否	否	否
IS_SPACES()	否	否	否
ISNULL()	是	是	是
LAST()	否	否	否
LAST_DAY()	否	否	否
LEAST()	否	否	否
LENGTH()	是	是	是

函数	DB2 for i5/OS ¹	DB2 for LUW	DB2 for z/OS ¹
LN()	是	是	是
LOG()	是	是	是
LOWER()	是	是	是
LPAD()	否	否	否
LTRIM()	是	是	是
MAKE_DATE_TIME()	否	否	否
MAX()	是	是	是
MD5()	否	否	否
MEDIAN()	否	否	否
METAPHONE()	否	否	否
MIN()	是	是	是
MOD()	是	是	是
MOVINGAVG()	否	否	否
MOVINGSUM()	否	否	否
NPER()	否	否	否
PERCENTILE()	否	否	否
PMT()	否	否	否
POWER()	是	是	是
PV()	否	否	否
RAND()	否	否	否
RATE()	否	否	否
REG_EXTRACT()	否	否	否
REG_MATCH()	否	否	否
REG_REPLACE	否	否	否
REPLACECHR()	否	否	否
REPLACESTR()	否	否	否
REVERSE()	否	否	否

函数	DB2 for i5/OS ¹	DB2 for LUW	DB2 for z/OS ¹
ROUND(DATE)	否	否	是
ROUND(NUMBER)	是	是	是
RPAD()	否	否	否
RTRIM()	是	是	是
SET_DATE_PART()	否	否	否
SIGN()	是	是	是
SIN()	是	是	是
SINH()	是	是	是
SOUNDEX()	否	是 ¹	否
SQRT()	否	是	否
STDDEV()	是	是	是
SUBSTR()	是	是	是
SUM()	是	是	是
SYSTIMESTAMP()	是	是	是
TAN()	是	是	是
TANH()	是	是	是
TO_BIGINT	是	是	是
TO_CHAR(DATE)	是	是	是
TO_CHAR(NUMBER)	是	是 ²	是
TO_DATE()	是	是	是
TO_DECIMAL()	是	是 ³	是
TO_DECIMAL38()	否	否	否
TO_FLOAT()	是	是	是
TO_INTEGER()	是	是	是
TO_TIMESTAMP_TZ()	否	否	否
TRUNC(DATE)	否	否	否
TRUNC(NUMBER)	是	是	是

函数	DB2 for i5/OS ¹	DB2 for LUW	DB2 for z/OS ¹
UPPER()	是	是	是
VARIANCE()	是	是	是

¹ 仅当函数包括在筛选器转换逻辑中时，数据集成服务才可以将这些函数推送至源。

² 如果此函数使用小数或浮点参数，则仅当函数包含在筛选器转换逻辑中时，数据集成服务才可以推送此函数。

³ 如果此函数使用字符串参数，则仅当函数包含在筛选器转换逻辑中时，数据集成服务才可以推送此函数。

下表显示了可用于 Greenplum、Hive、Microsoft SQL Server、Netezza、Oracle、SAP、SAP HANA、Sybase ASE 和 Teradata 源下推优化的 Informatica 函数：

函数	Greenplum	Hive	Microsoft SQL Server	Netezza	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
ABORT()	否	否	否	否	否	否	否	否	否
ABS()	是	是	是	是	是	否	是	是	是
ADD_TO_DATE()	是	否	是	是	是	否	否	是	是
AES_DECRYPT()	否	否	否	否	否	否	否	否	否
AES_ENCRYPT()	否	否	否	否	否	否	否	否	否
ASCII()	是	否	是	是	是	否	否	是	否
AVG()	是	是	是	是	是	否	是	是	是
CEIL()	是	是	是	是	是	否	是	是	是
CHOOSE()	否	否	否	否	否	否	否	否	否
CHR()	是	否	是	是	是	否	是	是	否
CHRCODE()	是	否	是	是	是	否	是	是	否
COMPRESS()	否	否	否	否	否	否	否	否	否
CONCAT()	是	是	是	是	是	否	是	是	是
COS()	是	是	是	是	是	否	是	是	是
COSH()	是	否	是	是	是	否	是	是	是
COUNT()	是	是	是	是	是	否	是	是	是
CRC32()	否	否	否	否	否	否	否	否	否
CREATE_TIMESTAMP_TZ()	否	否	否	否	是	否	否	否	否
CUME()	否	否	是	否	否	否	否	否	否

函数	Greenplum	Hive	Microsoft SQL Server	Netezza	Oracle	SAP 1	SAP HANA	Sybase ASE	Teradata
DATE_COMPARE()	是	否	是	是	是	否	是	是	是
DATE_DIFF()	否	否	否	否	否	否	是	否	否
DECODE()	是	是	是	是	是	否	是	是	是
DECODE_BASE64()	否	否	否	否	否	否	否	否	否
DECOMPRESS()	否	否	否	否	否	否	否	否	否
ENCODE_BASE64()	否	否	否	否	否	否	否	否	否
ERROR()	否	否	否	否	否	否	否	否	否
EXP()	是	是	是	是	是	否	否	是	是
FIRST()	否	否	否	否	否	否	否	否	否
FLOOR()	是	是	是	是	是	否	是	是	是
FV()	否	否	否	否	否	否	否	否	否
GET_DATE_PART()	是	否	是	是	是	否	是	是	是
GET_TIMESTAMP()	否	否	否	否	是	否	否	否	否
GET_TIMEZONE()	否	否	否	否	否	否	否	否	否
GREATEST()	否	否	否	否	是	否	否	否	否
IIF()	是	是	是	是	是	否	是	是	是
IN()	否	否	是	否	是	否	否	是	是
INDEXOF()	否	否	否	否	否	否	否	否	否
INITCAP()	是	否	否	是	是	否	否	否	否
INSTR()	否	否	是	是	是	否	否	是	是
IS_DATE()	否	否	否	否	否	否	否	否	否
IS_NUMBER()	否	否	否	否	否	否	否	否	否
IS_SPACES()	否	否	否	否	否	否	否	否	否
ISNULL()	否	是	是	否	是	否	是	是	是
LAST()	否	否	否	否	否	否	否	否	否
LAST_DAY()	否	否	否	是	是	否	是	否	否

函数	Greenplum	Hive	Microsoft SQL Server	Netezza	Oracle	SAP ↓	SAP HANA	Sybase ASE	Teradata
LEAST()	否	否	否	否	是	否	否	否	否
LENGTH()	是	是	是	是	是	否	是	是	是
LN()	是	是	是	否	是	否	是	是	是
LOG()	是	是	是	是	是	否	是	是	是
LOWER()	是	是	是	是	是	否	是	是	是
LPAD()	是	是	否	是	是	否	是	否	否
LTRIM()	是	是	是	是	是	否	是	否	是
MAKE_DATE_TIME()	否	否	否	否	否	否	否	否	否
MAX()	是	是	是	是	是	否	是	是	是
MD5()	否	否	否	否	否	否	否	否	否
MEDIAN()	否	否	否	否	是	否	否	否	否
METAPHONE()	否	否	否	否	否	否	否	否	否
MIN()	是	是	是	是	是	否	是	是	是
MOD()	是	是	是	是	是	否	是	是	是
MOVINGAVG()	否	否	否	否	否	否	否	否	否
MOVINGSUM()	否	否	否	否	否	否	否	否	否
NPER()	否	否	否	否	否	否	否	否	否
PERCENTILE()	否	否	否	否	否	否	否	否	否
PMT()	否	否	否	否	否	否	否	否	否
POWER()	是	是	是	是	是	否	是	是	是
PV()	否	否	否	否	否	否	否	否	否
RAND()	否	否	否	否	否	否	否	否	否
RATE()	否	否	否	否	否	否	否	否	否
REG_EXTRACT()	否	否	否	否	否	否	否	否	否
REG_MATCH()	否	否	否	否	否	否	否	否	否
REG_REPLACE	否	否	否	否	否	否	否	否	否

函数	Greenplum	Hive	Microsoft SQL Server	Netezza	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
REPLACECHR()	否	否	否	否	否	否	否	否	否
REPLACESTR()	否	否	否	否	否	否	否	否	否
REVERSE()	否	否	否	否	否	否	否	否	否
ROUND(DATE)	否	否	否	否	是	否	否	否	否
ROUND(NUMBER)	是	是	是	是	是	否	是	是	是
RPAD()	是	是	否	是	是	否	是	否	否
RTRIM()	是	是	是	是	是	否	是	是	是
SET_DATE_PART()	否	否	否	否	否	否	否	否	否
SIGN()	是	否	是	是	是	否	是	是	是
SIN()	是	是	是	是	是	否	是	是	是
SINH()	是	否	是	是	是	否	是	是	是
SOUNDEX()	否	否	是	否	是	否	否	是	否
SQRT()	是	是	是	是	是	否	是	是	是
STDDEV()	是	否	是	是	是	否	是	是	是
SUBSTR()	是	是	是	是	是	否	是	是	是
SUM()	是	是	是	是	是	否	是	是	是
SYSTIMESTAMP()	是	否	是	是	是	否	是 ²	是	否
TAN()	是	否	是	是	是	否	是	是	是
TANH()	是	否	是	是	是	否	否	是	是
TO_BIGINT	是	是	是	是	是	否	是	是	是
TO_CHAR(DATE)	是	否	是	是	是	否	是	是	是
TO_CHAR(NUMBER)	是	否	是	是	是	否	是	是	是
TO_DATE()	是	否	是	是	是	是	是	是	是
TO_DECIMAL()	是	否	是	是	是	否	是	是	是
TO_DECIMAL38()	是	是	是	是	是	是	是	是	是
TO_FLOAT()	是	否	是	是	是	否	是	是	是

函数	Greenplum	Hive	Microsoft SQL Server	Netezza	Oracle	SAP ¹	SAP HANA	Sybase ASE	Teradata
TO_INTEGER()	是	是	是	是	是	否	是	是	是
TO_TIMESTAMP_TZ()	否	否	否	否	是	否	否	否	否
TRUNC(DATE)	是	否	否	是	是	否	是	否	否
TRUNC(NUMBER)	是	否	是	是	是	否	是	是	是
UPPER()	是	是	是	是	是	否	是	是	是
VARIANCE()	是	否	是	是	是	否	否	是	是

¹. 仅当这些函数包含在筛选器转换逻辑中时，数据集成服务才可将其推送至源。

². SYSTIMESTAMP() 仅支持 SS 参数。

下表汇总了 Informatica 函数对下推优化的可用性。

函数	DB2 for LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
ABS()	是	是	是	是	是
ADD_TO_DATE()	是	是	否	是	是
ASCII()	是	是	否	是	是
CEIL()	是	是	否	是	是
CHR()	是	是	否	是	是
CONCAT()	是	是	否	是	是
COS()	是	是	是	是	是
COSH()	是	是	否	是	是
DATE_COMPARE()	是	是	是	是	是
DECODE()	是	是	是	是	是
EXP()	是	是	是	不适用	是
FLOOR()	否	是	否	是	是
GET_DATE_PART()	是	是	否	是	是
IIF()	是	是	是	否	是
IN()	否	是	是	否	是
INITCAP()	否	否	否	是	否

函数	DB2 for LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
INSTR()	是	是	否	是	是
ISNULL()	是	是	是	是	是
LAST_DAY()	否	否	否	是	不适用
LENGTH()	是	是	否	是	是
LN()	是	否	否	是	是
LOG()	是	是	否	是	是
LOOKUP()	否	否	是	否	否
LOWER()	是	是	是	是	是
LPAD()	否	否	否	是	不适用
LTRIM()	X	是	否	是	是
MOD()	X	是	否	是	是
POWER()	X	是	否	是	是
ROUND(DATE)	否	否	否	是	不适用
ROUND(NUMBER)	是	是	否	是	是
RPAD()	否	否	否	是	否
RTRIM()	是	是	否	是	是
SIGN()	是	是	否	是	是
SIN()	是	是	是	是	是
SINH()	是	是	否	是	是
SOUNDEX()	是 ¹	是	否	是	是
SQRT()	是	是	是	是	是
SUBSTR()	是	是	否	是	是
SYSDATE()	是	是	否	是	是
SYSTIMESTAMP()	是	是	否	是	是
TAN()	是	是	是	是	是
TANH()	是	是	否	是	是
TO_BIGINT	是	是	否	是	是

函数	DB2 for LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
TO_CHAR(DATE)	是	是	否	是	是
TO_CHAR(NUMBER)	是 ²	是	否	是	是
TO_DATE()	是	是	否	是	是
TO_DECIMAL()	是 ³	是	否	是	是
TO_FLOAT()	是	是	否	是	是
TO_INTEGER()	是	是	否	是	是
TRUNC(DATE)	否	否	否	是	否
TRUNC(NUMBER)	是	是	否	是	是
UPPER()	是	是	是	是	是

.¹ 仅当函数包括在筛选器转换逻辑中时，数据集成服务才可以将这些函数推送至源。

.² 如果此函数使用小数或浮点参数，则仅当函数包含在筛选器转换逻辑中时，数据集成服务才可以推送此函数。

.³ 如果此函数使用字符串参数，则仅当函数包含在筛选器转换逻辑中时，数据集成服务才可以推送此函数。

Hive 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Hive 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Hive 源处理转换逻辑：

- LTRIM 将空格用作第二个参数。
- RTRIM 将空格用作第二个参数。

对以下函数使用日期数据类型时，数据集成服务无法为 Hive 源处理转换逻辑：

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

IBM DB2 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 IBM DB2 for i5/OS、DB2 for LUW 和 DB2 for z/OS 源。在某些情况下，数据集成服务无法将受支持的函数推送至 IBM DB2 for LUW 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 IBM DB2 源处理转换逻辑：

- ADD_TO_DATE 或 GET_DATE_PART 返回采用毫秒或纳秒精度的结果。
- LTRIM 包含多个参数。
- RTRIM 包含多个参数。

- TO_BIGINT 在 DB2 for LUW 源上将字符串转换为长整型值。
- TO_CHAR 将日期转换为字符串并指定了不受 DB2 支持的格式。
- TO_DATE 将字符串转换为日期并指定了一个不受 DB2 支持的格式。
- TO_DECIMAL 在不使用小数位数参数的情况下将字符串转换为十进制值。
- TO_FLOAT 将字符串转换为双精度浮点数。
- TO_INTEGER 在 DB2 for LUW 源上将字符串转换为整数值。

Microsoft SQL Server 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Microsoft SQL Server 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Microsoft SQL Server 源处理转换逻辑：

- IN 包括 CaseFlag 参数。
- INSTR 包含三个以上的参数。
- LTRIM 包含多个参数。
- RTRIM 包含多个参数。
- TO_BIGINT 包含多个参数。
- TO_INTEGER 包含多个参数。

Netezza 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Netezza 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Netezza 源处理转换逻辑：

- SYSTIMESTAMP 包含 YYYY-MM-DD HH24:MI:SS.US 格式的日期。
- TO_CHAR(DATE) 和 TO_DATE() 包含 YYYY-MM-DD HH24:MI:SS.US 格式的日期，精确到子秒。

Oracle 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Oracle 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Oracle 源处理转换逻辑：

- ADD_TO_DATE 或 GET_DATE_PART 返回采用亚秒精度的结果。
- ROUND 将值舍入为秒或亚秒。
- SYSTIMESTAMP 返回采用微秒精度的日期和时间。
- TRUNC 截断秒或亚秒。

ODBC 函数异常

如果 IN 函数的 CaseFlag 参数是零以外的数字，则数据集成服务将为 ODBC 处理转换逻辑。

注意：如果 ODBC 连接对象属性包括特定于数据库的 ODBC 提供程序，则数据集成服务会将源视为本地源类型。

当您将连接对象中的 ODBC 提供程序指定为**其他**时，数据集成服务无法将 EXP() 函数推送到 Teradata 源。将 ODBC 提供程序设置为 **Teradata** 以推送 EXP() 函数。

Sybase ASE 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Sybase ASE 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Sybase ASE 源处理转换逻辑：

- IN 包括 CaseFlag 参数。
- INSTR 包含两个以上的参数。
- LTRIM 包含多个参数。
- RTRIM 包含多个参数。
- TO_BIGINT 包含多个参数。
- TO_INTEGER 包含多个参数。
- TRUNC(Numbers) 包含多个参数。

Teradata 函数异常

在某些情况下，数据集成服务无法将受支持的函数推送至 Teradata 源。

如果表达式中包含使用以下逻辑的受支持函数，则数据集成服务将为 Teradata 源处理转换逻辑：

- ADD_TO_DATE 包含的属性不包括 YEAR 和 MONTH。
- IN 包括 CaseFlag 参数。
- INSTR 包含两个以上的参数。
- LTRIM 包含多个参数。
- ROUND 包含多个参数。
- RTRIM 包含多个参数。

运算符

下表按源类型汇总了 Informatica 运算符的可用性。每一列显示了数据集成服务是否可以将运算符推送至源。

注意：非关系源是 z/OS 上的 IMS、VSAM 和序列数据集。

运算符	DB2 for LUW	DB2 for i5/OS 或 DB2 for z/OS	Greenplum	Hive	Microsoft SQL Server	非关系*	Oracle	SAP*	SAP HANA	Sybase ASE	Teradata
+ - *	是	是	是	是	是	是	是	否	是	是	是
/	是	是	是	是	是	否	是	否	是	是	是
%	是	是	是	是	是	否	是	否	是	是	是
	是	是	是	是	是	否	是	否	是	是	是

运算符	DB2 for LUW	DB2 for i5/OS 或 DB2 for z/OS	Greenplum	Hive	Microsoft SQL Server	非关系*	Oracle	SAP*	SAP HANA	Sybase ASE	Teradata
= > < >= <=	是	是	是	是	是	是	是	是	是	是	是
<>	是	是	是	是	是	否	是	是	是	是	是
!=	是	是	是	是	是	是	是	是	是	是	是
^=	是	是	是	是	是	否	是	是	是	是	是
AND OR	是	是	是	是	是	是	是	是	是	是	是
NOT	是	是	是	是	是	否	是	否	是	是	是

* 仅当运算符包括在筛选器转换逻辑中时，数据集成服务才可以将这些运算符推送至源。

运算符	DB2 for LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
+ - *	是	是	是	是	是
/	是	是	是	是	是
%	是	是	否	是	是
	是	是	否	是	是
= > < >= <=	是	是	是	是	是
<>	是	是	是	是	是
!=	是	是	是	是	是
^=	是	是	是	是	是

运算符	DB2 for LUW	Microsoft SQL Server	ODBC	Oracle	Sybase ASE
AND OR	是	是	是	是	是
NOT	是	是	是	是	是

比较数据集成服务的输出和源

如果处理相同的转换逻辑，则数据集成服务和源可能会产生不同的结果。如果数据集成服务将转换逻辑推送至源，则转换逻辑的输出可能不同。

在以下情况中转换逻辑的输出可能不同：

区分大小写

数据集成服务和数据库在处理是否区分大小写方面可能会有所不同。例如，数据集成服务使用区分大小写的查询，但数据库不区分大小写。某个筛选器转换使用以下筛选条件：IIF(col_varchar2 = 'CA', TRUE, FALSE)。需要数据库返回与“CA”匹配的行。但是，如果将此转换逻辑推送至不区分大小写的数据库，则将返回与值“Ca”、“ca”、“cA”和“CA”匹配的行。

转换为字符值的数值

数据集成服务和数据库可以将相同的数值转换为不同格式的字符值。数据库可能会将数值转换为不可接受的字符格式。例如，一个表包含数字 1234567890。数据集成服务将该数字转换为字符值时，会插入字符“1234567890”。但是，数据库可能会将该数字转换为“1.2E9”。这两组字符表示相同的值。

TO_CHAR 和 TO_DATE 函数的日期格式

如果数据集成服务将函数推送至数据库，则使用 TO_CHAR 或 TO_DATE 函数中的日期格式。使用 TO_DATE 函数可比较日期或时间值。使用 TO_CHAR 比较日期或时间值时，数据库可能会在某些值（例如个位数的月、日或小时）的前面添加一个空格或零。添加空格或前导零后，数据库的比较结果可能与数据集成服务的结果有所不同。

精度

数据集成服务和数据库可能会对特定的数据类型使用不同的精度。转换数据类型使用默认的数值精度，该精度可能与本地数据类型的精度有所不同。如果数据库使用与数据集成服务不同的精度，则结果可能有所不同。

SYSTIMESTAMP 函数

使用 SYSTIMESTAMP 时，数据集成服务将返回运行服务进程的节点的当前日期和时间。但是，将转换逻辑推送至数据库时，数据库将返回托管该数据库的计算机的当前日期和时间。如果托管该数据库的计算机与运行数据集成服务进程的计算机所在的时区不同，则结果可能有所不同。

如果将 SYSTIMESTAMP 推送至 IBM DB2 或 Sybase ASE 数据库，并且指定了 SYSTIMESTAMP 的格式，则数据库将忽略该格式并返回完整的时间戳。

LTRIM、RTRIM 或 SOUNDEX 函数

将 LTRIM、RTRIM 或 SOUNDEX 推送至数据库时，数据库会将参数 (') 视作空值，但数据集成服务会将参数 (') 视作空格。

Oracle 源上的 LAST_DAY 函数

将 LAST_DAY 推送至 Oracle 时，Oracle 将返回精确到秒的日期。如果输入日期包含亚秒，则 Oracle 会将日期截减至秒。

第 14 章

已分区的映射

本章包括以下主题：

- [已分区的映射概览, 227](#)
- [每个管道阶段一个线程, 228](#)
- [每个管道阶段多个线程, 229](#)
- [已分区平面文件源, 230](#)
- [已分区关系源, 231](#)
- [已分区平面文件目标, 233](#)
- [已分区关系目标, 236](#)
- [已分区转换, 237](#)
- [保持已分区映射中的顺序, 239](#)
- [替代映射的最大并行数值, 240](#)
- [对已分区映射进行故障排除, 243](#)

已分区的映射概览

如果您具有分区选项，管理员可以使数据集成服务在运行映射时实现并行最大化。管理员实现并行最大化后，数据集成服务可将基本数据动态划分为多个分区并同时处理所有分区。

如果映射处理大数据集或者包含执行复杂计算的转换，则映射可能要花费较长时间进行处理并导致数据吞吐量降低。当您为这些映射启用分区时，数据集成服务将使用额外的线程处理映射，从而可以优化性能。

要启用分区，管理员和开发人员需执行下列任务：

管理员在 Administrator 工具中将数据集成服务的最大并行数值设置为大于 1 的值。

最大并行数值确定了处理单个管道阶段的最大并行线程数。管理员可根据运行映射的节点上可用的 CPU 数量增大**最大并行数**属性值。

或者，开发人员也可以在 Developer tool 中设置映射的最大并行数值。

默认情况下，每个映射的**最大并行数**属性均设置为“自动”。每个映射都使用为数据集成服务定义的最大并行数值。

开发人员可以在映射运行时属性中更改最大并行数值，以定义特定映射的最大值。当数据集成服务和映射的最大并行数设置为不同的整数值时，数据集成服务将使用两者之中的较小值。

禁用了某个映射的分区时，数据集成服务会将该映射划分为多个管道阶段并使用一个线程来处理每个阶段。

启用了某个映射的分区时，数据集成服务会使用多个线程来处理每个映射管道阶段。

数据集成服务可以为映射创建分区，这些分区具有物理数据作为输入和输出。在映射运行期间，数据集成服务可以使用多个分区来完成以下操作：

- 从平面文件、IBM DB2 for LUW 或 Oracle 源中读取数据。
- 运行转换。
- 向平面文件、IBM DB2 for LUW 或 Oracle 目标写入数据。

每个管道阶段一个线程

最大并行数设置为 1 时，将禁用分区。数据集成服务会将映射划分为多个管道阶段并使用一个线程来处理每个阶段。

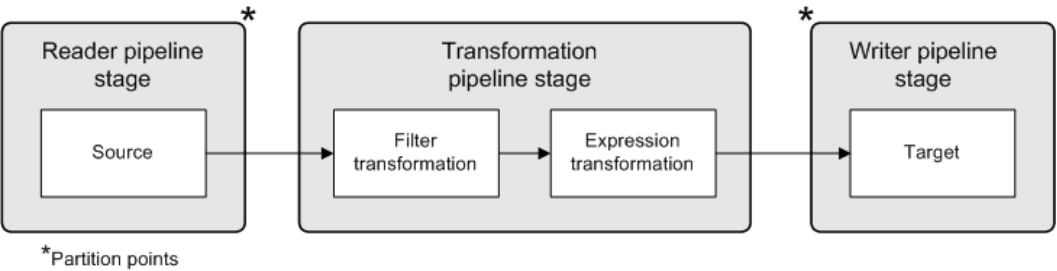
每个映射包含一个或多个管道。管道包含一个读取转换和从该读取转换接收数据的所有转换。数据集成服务将一个映射管道分为多个管道阶段，然后并行执行每个管道阶段的提取、转换和加载操作。

分区点标记了管道中的边界并将管道划分为不同阶段。对于每个映射管道，数据集成服务会在读取转换之后写入转换之前添加分区点，从而创建多个管道阶段。

每个管道阶段将在下列线程之一中运行：

- 读取器线程：控制数据集成服务从源提取数据的方式。
- 转换线程：控制数据集成服务在管道中处理数据的方式。
- 写入器线程：控制数据集成服务将数据加载到目标的方式。

下图显示了某个映射划分为读取器管道阶段、转换管道阶段和写入器管道阶段：



因为该管道包含三个阶段，所以数据集成服务可以同时处理三个行集并优化映射性能。例如，读取器线程处理第三个行集时，转换线程处理第二个行集，而写入器线程处理第一个行集。

下表显示了多个线程同时处理三个行集的方式：

读取器线程	转换线程	写入器线程
行集 1	-	-
行集 2	行集 1	-
行集 3	行集 2	行集 1
行集 4	行集 3	行集 2
行集 n	行集 (n-1)	行集 (n-2)

如果映射管道包含执行复杂计算的转换，则转换管道阶段可能需要花费较长时间。为优化性能，数据集成服务在某些转换前添加了分区点，以创建额外的转换管道阶段。

每个管道阶段多个线程

最大并行数设置为大于 1 的值时，将启用分区。数据集成服务会将映射划分为多个管道阶段并使用多个线程来处理每个阶段。任何管道阶段中的线程数等于该阶段中的分区数。

实现并行最大化后，数据集成服务可在运行时自动执行下列任务：

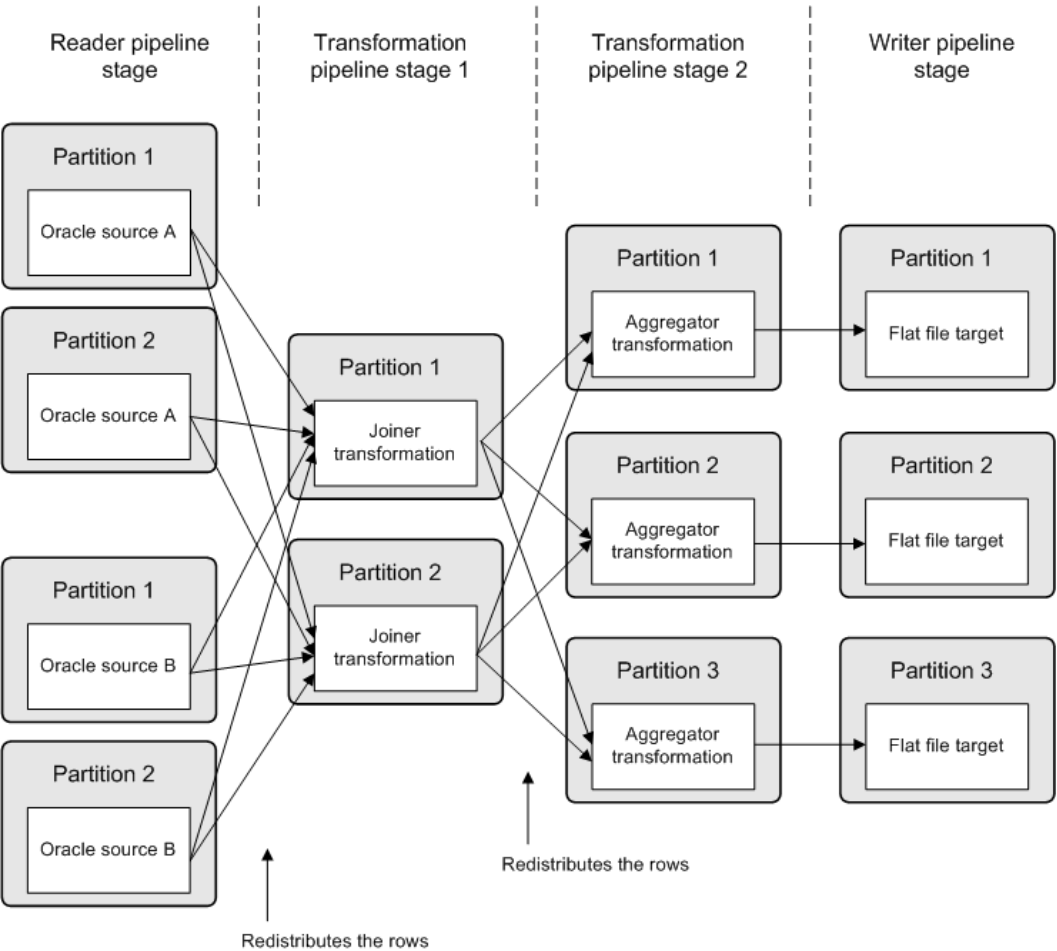
将数据划分为多个分区。

数据集成服务会将基本数据动态地划分为多个分区，并同时运行这些分区。数据集成服务将决定每个管道阶段的最佳线程数量。用于单个管道阶段的线程数量不能超过最大并行数值。数据集成服务可以为每个管道阶段使用不同数量的线程。

在各分区点重新分发数据。

数据集成服务会根据转换要求动态确定在分区点重新分发数据的最佳方式。

下图显示了在多个分区间为每个管道阶段分发数据的映射示例：



在上图中，数据集成服务的最大并行数为三个。此映射的最大并行数为“自动”。数据集成服务将映射分隔为四个管道阶段，并使用总共 12 个线程运行映射。在每个管道阶段，数据集成服务会执行以下任务：

- 在读取器管道阶段，数据集成服务查询 Oracle 数据库系统，以发现两个源表（源 A 和源 B）包含两个数据库分区。数据集成服务为每个数据库分区使用一个读取器线程。
- 在第一转换管道阶段，数据集成服务在两个线程间将数据重新分发给联接条件的组行。
- 在第二转换管道阶段，数据集成服务确定汇总器转换的最佳线程数量为三个。该服务在三个线程间将数据重新分发给汇总表达式的组行。
- 在写入器管道阶段，数据集成服务不需要在目标分区点重新分发各个行。单个分区中的所有行在错过目标分区点后将保留在该分区。

已分区平面文件源

当启用了分区的映射从平面文件源进行读取时，数据集成服务可以使用多个线程来读取该文件源。

数据集成服务可以为下列类型的平面文件源创建分区：

- 直接文件
- 间接文件
- 文件目录
- 命令
- Hadoop 分布式文件系统 (HDFS) 中的文件或文件目录

当数据集成服务使用多个线程来读取文件源时，该服务会创建多个指向源的并发连接。默认情况下，数据集成服务不会保留行顺序，因为数据集成服务不是按顺序读取文件或文件列表中的行。要在多个线程读取单个文件源时保留行顺序，请配置并发读取分区。

当数据集成服务使用多个线程来读取直接文件时，该服务会创建多个读取器线程来并发读取该文件。

当数据集成服务使用多个线程来读取间接文件或文件目录时，该服务会创建多个读取器线程来并发读取列表或目录中的文件。数据集成服务可能会使用多个线程来读取单个文件。或者，数据集成服务也可能使用单个线程来读取列表或目录中的多个文件。

并发读取分区

要在多个线程读取单个文件源时保留行顺序，请为平面文件数据对象配置**并发读取分区**属性以保留该顺序。

在平面文件数据对象的**高级**属性中配置**并发读取分区**属性。Find the property in the **Runtime:Read** section.

为**并发读取分区**属性选择以下选项之一：

优化吞吐量

多个分区读取单个文件源时数据集成服务不保留行顺序。如果多个分区读取单个文件源的顺序不重要，则可以使用此选项。

默认选项。

保持相对顺序

保留每个分区读取的输入行的排序顺序。

下表显示了一个文件源的排序顺序示例（其中 10 行由两个分区读取）：

分区	读取的行数
分区 #1	1,3,5,8,9
分区 #2	2,4,6,7,10

保持绝对顺序

保留所有分区读取的全部输入行的排序顺序。在具有被动转换的传递映射中，写入目标的行顺序与输入行的顺序相同。

下表显示了一个文件源的排序顺序示例（其中 10 行由两个分区读取）：

分区	读取的行数
分区 #1	1,2,3,4,5
分区 #2	6,7,8,9,10

已分区关系源

当启用了分区的映射从 IBM DB2 for LUW 或 Oracle 源进行读取时，数据集成服务可以使用多个线程来读取关系源。数据集成服务会为每个线程创建一个单独的数据库连接。

注意: 如果映射从除 DB2 for LUW 或 Oracle 以外的关系源进行读取，数据集成服务将使用一个线程从该源读取数据。数据集成服务可以使用多个线程处理余下的映射管道阶段。

数据集成服务会查询 DB2 for LUW 或 Oracle 数据库系统以查找分区信息。如果源表支持数据库分区，则数据集成服务可以使用多个线程从数据库中对应的节点读取已分区数据。数据集成服务会为每个读取器线程生成一个 SQL 查询。

数据集成服务使用的读取器线程数取决于下列情况：

数据库分区数量小于或等于最大并行数值。

数据集成服务为每个数据库分区使用一个读取器线程。数据集成服务会为每个读取器线程分发一个数据库分区。

对于使用复合分区的 Oracle 源，数据集成服务会针对每个数据库子分区使用一个读取器线程。例如，如果一个 Oracle 源包含三个分区，且每个分区包含两个子分区，数据集成服务将使用六个读取器线程。

数据库分区数的大于最大并行数值。

数据集成服务将按最大并行数值所定义的数量使用读取器线程。数据集成服务会为某些读取器线程分发多个数据库分区。例如，一个 DB2 for LUW 源具有五个数据库分区，且最大并行数值为 3。数据集成服务将使用三个读取器线程。数据集成服务会将两个数据库分区分发给第一个读取器线程和第二个读取器线程。服务会将一个数据库分区分发给第三个读取器线程。

无数据库分区。

数据集成服务使用一个线程从源读取数据。数据集成服务可以使用多个线程处理余下的映射管道阶段。

分区的关系连接类型

数据集成服务可以使用多个线程根据用于连接到数据库的连接类型读取 DB2 for LUW 或 Oracle 关系源。

您可以使用以下任何连接类型连接至 DB2 for LUW 或 Oracle 数据库：

- DB2 for LUW 连接或 Oracle 连接
- JDBC 连接
- ODBC 连接

要使用多个线程来读取 DB2 for LUW 或 Oracle 关系源，关系数据对象必须使用 DB2 for LUW 或 Oracle 连接。

如果 DB2 for LUW 或 Oracle 关系数据对象使用 JDBC 或 ODBC 连接，数据集成服务将使用一个线程来读取源。数据集成服务可以使用多个线程处理余下的映射管道阶段。

已分区关系源的 SQL 查询

数据集成服务使用多个线程来读取关系源时，会为每个读取器线程生成一个 SQL 查询。

如果数据库源的数据库分区数大于最大并行数值，则数据集成服务会将数据分发给各读取器线程。数据集成服务可以生成读取多个数据库分区的 SQL 查询。Oracle 源包含子分区时，数据集成服务可以生成读取多个数据库子分区的 SQL 查询。

DB2 for LUW 或 Oracle 源示例

最大并行数值为 3，并且关系源具有 5 个数据库分区。数据集成服务对这些数据库分区运行 SQL 查询时，第一个和第二个读取器线程会从两个数据库分区接收数据。第三个读取器线程会从一个数据库分区接收数据。在此示例中，读取转换中的简单查询未启用“选择相异”选项。

使用 DB2 for LUW 源时，数据集成服务会为第一个读取器线程生成类似于下列语句的 SQL 语句：

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

使用 Oracle 源时，数据集成服务会为第一个读取器线程生成类似于下列语句的 SQL 语句：

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

具有子分区的 Oracle 源的示例

Oracle 源具有五个分区 (1-5)，且每个分区中有两个子分区 (a 和 b)。最大并行数值为 3。第一个读取器线程会从四个数据库子分区接收数据。第二个读取器线程会从三个数据库子分区接收数据。在此示例中，读取转换中的简单查询未启用“选择相异”选项。

数据集成服务会为第一个读取器线程生成类似于下列语句的 SQL 语句：

```
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
```

关系源分区的规则和准则

当启用分区的映射从关系源中读取数据时，请考虑下列规则和准则：

- 数据集成服务使用一个线程来读取源，但在以下情况下可以为余下的映射管道阶段使用多个线程：
 - 映射从 DB2 for LUW 或 Oracle 以外的关系源中读取数据。
 - 映射使用 JDBC 或 ODBC 连接从 DB2 for LUW 或 Oracle 源中读取数据。
 - 该映射将转换逻辑推送到源数据库。

- 使用读取转换中的简单查询来选择要作为排序依据的端口或者配置一个用户定义的联接。
- 使用读取转换中的高级查询来创建自定义 SQL 查询。
- 如果使用读取转换中的简单查询来创建提示、选择相异值或者输入源筛选器，则数据集成服务会使用多个线程来读取源。数据集成服务会将提示、相异值或源筛选器添加到为每个分区生成的 SQL 查询。

已分区平面文件目标

当启用了分区的映射向平面文件目标写入数据时，数据集成服务可以使用多个线程来向该文件目标写入数据。

数据集成服务可以为平面文件或 Hadoop 分布式文件系统 (HDFS) 中的文件创建分区。

可以将平面文件数据对象配置为具有文件或命令输出类型。当平面文件数据对象具有文件输出类型时，数据集成服务会将目标数据写入至平面文件。如果多个线程向该平面文件目标写入数据，每个线程均会将目标输出写入一个单独的文件。每个线程将使用以下格式来命名该文件：

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

例如，三个线程可以向名称为 MyOutput1.out、MyOutput2.out 和 MyOutput3.out 的文件写入数据。

可以配置多个输出文件目录以优化性能，也可以配置平面文件数据对象以向单个合并文件写入数据。

当平面文件数据对象具有命令输出类型时，数据集成服务会将目标数据输出至命令或合并命令，而非平面文件或合并文件。如果多个分区向平面文件目标写入数据，则可使用命令来处理单个分区的目标数据或处理所有目标分区的合并数据。

优化分区文件目标的输出文件目录

默认情况下，如果平面文件数据对象具有文件输出类型，则每个线程均会将目标输出写入至单独的文件。要优化多个线程向一个文件目标写入数据时的性能，请配置多个输出文件目录。

当多个线程向单个目录写入数据时，映射可能会因输入/输出 (I/O) 争用而遇到瓶颈问题。当多个线程同时向文件系统写入数据时，将出现 I/O 争用。

当您配置了多个目录时，数据集成服务会以循环方式确定每个线程的输出目录。例如，您将一个平面文件数据对象配置为使用 directoryA 和 directoryB 作为目标目录。如果数据集成服务使用四个线程向文件目标写入数据，则第一个和第三个写入器线程会将目标文件写入到 directoryA。第二个和第四个写入器线程会将目标文件写入 directoryB。

如果数据集成服务不使用多个线程向目标写入数据，服务会将输出文件写入第一个列出的目录。

在平面文件数据对象的**高级**属性中配置输出文件目录。在**运行时: 写入**部分中查找**输出文件目录**属性。默认情况下，该属性被配置为使用为数据集成服务定义的系统参数值。如果管理员为数据集成服务的**目标目录**属性输入了多个以分号分隔的目录，则使用默认的 TargetDir 系统参数值。

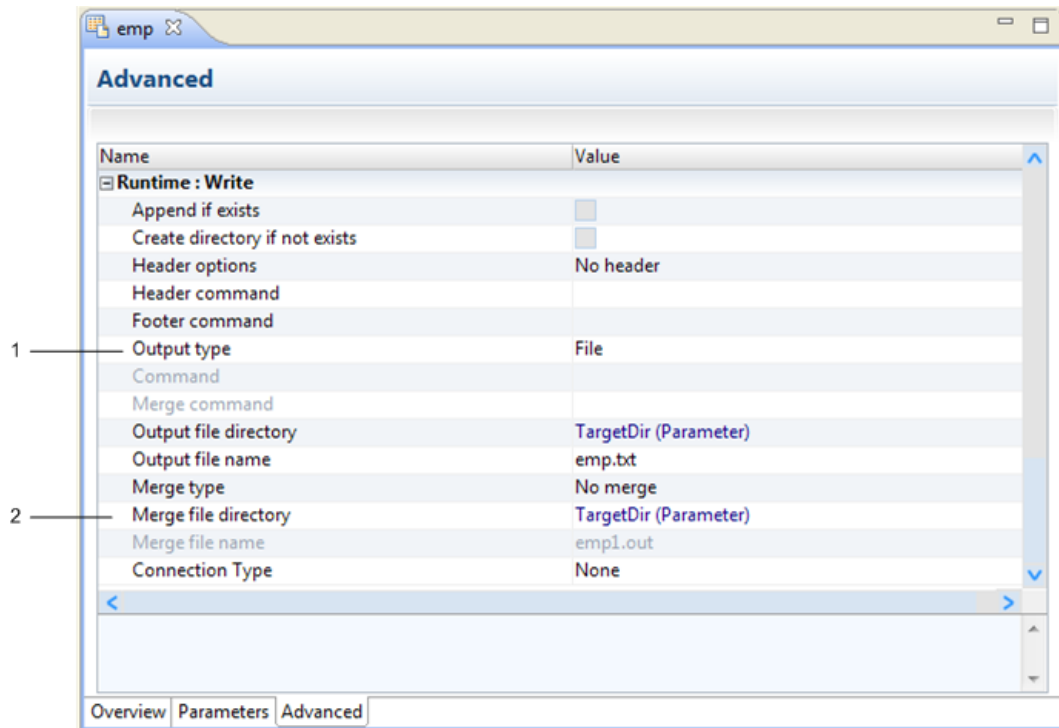
可以输入其他值以配置多个特定于平面文件数据对象的输出文件目录。为属性或分配给属性的用户定义参数输入多个由分号分隔的目录。

已分区文件目标的合并选项

默认情况下，如果平面文件数据对象具有文件输出类型，则每个线程均会将目标输出写入至单独的文件。您可以合并分区的目标数据。合并目标数据时，数据集成服务将为所有目标分区创建单个合并文件。

在平面文件数据对象的**高级**属性中配置合并选项。在**运行时: 写入**部分中查找合并属性。

下图显示了平面文件数据对象的高级属性中的合并选项：



1. 文件输出类型

2. 合并选项

为**合并类型**属性选择以下选项之一：

不合并

数据集成服务会并发针对每个分区将目标输出写入一个单独的文件。

默认选项。

连续

数据集成服务会为每个分区创建一个输出文件，然后将这些文件合并为单个合并文件。数据集成服务使用输出文件名称和输出文件目录值来创建各个目标文件。数据集成服务按每个写入器线程的完成顺序将每个分区的输出数据连续添加到合并文件。例如，如果 Partition2 的写入器线程在 Partition1 的线程之前完成，则数据集成服务会按如下顺序向合并文件添加数据：Partition2、Partition1。

文件列表

数据集成服务会为每个分区创建一个目标文件，然后创建一个包含各个文件路径的文件列表。数据集成服务使用输出文件名称和输出文件目录值来创建各个目标文件。如果将目标文件写入合并目录或合并目录的子目录，则文件列表将包含相对路径。否则，文件列表包含绝对路径。如果将这些目标文件用作其他映射的源文件，则可将此文件用作源文件。

并发

数据集成服务会并发地将所有目标分区的数据写入合并文件。数据集成服务不会为每个分区创建中间文件。因为数据集成服务会并发地为所有分区写入合并文件，所以合并文件中的数据顺序可能不是连续的。

如果配置平面文件数据对象以合并目标数据，可以编辑**合并文件目录**和**合并文件名称**属性的默认值（可选）。

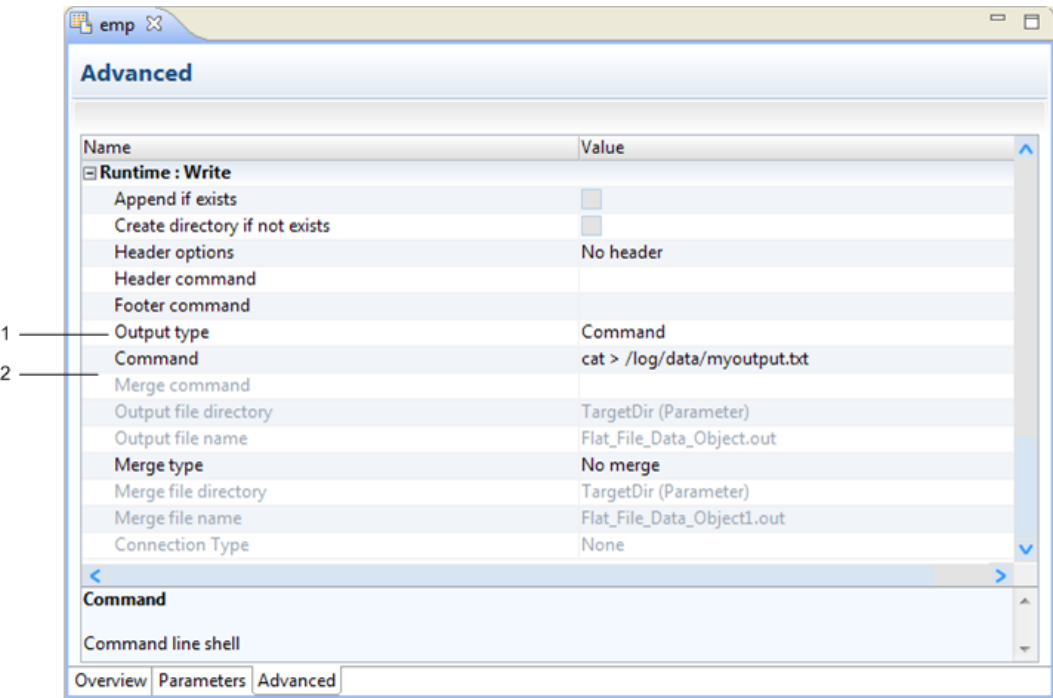
如果配置平面文件数据对象以合并目标数据并且数据集成服务不为目标创建分区，则数据集成服务将忽略合并选项。服务会写入**输出文件名称**和**输出文件目录**属性中定义的文件。

已分区文件目标的命令

当平面文件数据对象具有命令输出类型时，可以使用命令来处理单个分区的目标数据或处理映射中所有目标分区的合并数据。数据集成服务将数据发送至命令或合并命令，而不是发送至平面文件或合并文件。

在平面文件数据对象的**高级**属性中配置处理分区数据的命令。在**运行时: 写入**部分中查找命令属性。

下图显示了一个被配置为使用命令处理单个分区目标数据的平面文件数据对象：



- 1. 命令输出类型
- 2. 命令选项

在 UNIX 中，请使用任何有效的 UNIX 命令或 Shell 脚本。在 Windows 中，请使用任何有效的 DOS 或批处理文件。

可以使用命令来处理下列类型的目标数据：

单个分区的数据

输入每个写入器线程单独运行的命令。映射运行时每个线程均将目标数据发送至该命令。每个线程针对不同的数据集运行同一命令。

输入命令时，必须考虑到运行映射的操作系统。例如，如果输入了命令 `cat > /log/data/myoutput.txt`，则多个线程将写入同一个文件，这将导致操作系统错误。如果输入了命令 `cat >> /log/data/myoutput.txt`，则多个线程会将数据附加到同一文件，这样会降低导致操作系统错误的可能性。

要将单个分区的目标数据发送至命令，请选择**输出类型**属性的命令，并为**合并类型**属性选择“不合并”。为**命令**属性输入一个命令。

合并所有分区的数据

输入处理所有写入线程的合并数据的合并命令。数据集成服务必须对命令使用并发合并类型才能处理合并数据。映射运行时每个线程会同时将目标数据发送至合并命令。合并命令只针对所有数据运行一次。该命令可能无法保持目标数据的顺序。

要将所有分区的合并数据发送至合并命令，请选择**输出类型**属性的命令，并为**合并类型**属性选择“并发”。为**合并命令**属性输入一个命令。

已分区关系目标

当启用了分区的映射向 IBM DB2 for LUW 或 Oracle 目标写入数据时，数据集成服务可以使用多个线程来向关系目标写入数据。数据集成服务会为每个线程创建一个单独的数据库连接。

注意: 如果映射向除 DB2 for LUW 或 Oracle 以外的关系目标写入数据，数据集成服务将使用一个线程来向该目标写入数据。数据集成服务可以使用多个线程处理余下的映射管道阶段。

当目标为 Oracle 时，数据集成服务将按照最大并行数值所定义的数量使用写入器线程。如果 Oracle 关系表具有分区，数据库会将数据路由到正确的分区。

目标为 DB2 for LUW 时，数据集成服务会查询 DB2 for LUW 系统以查找分区信息。数据集成服务会将已分区的数据加载到目标数据库中的对应节点。

数据集成服务用于 DB2 for LUW 目标的写入器线程数取决于下列情况：

数据库分区数量小于或等于最大并行数值。

数据集成服务将针对每个数据库分区使用一个写入器线程。每个写入器线程写入到一个数据库分区。

数据库分区数的大于最大并行数值。

数据集成服务将按照最大并行数值所定义的数量使用写入器线程。数据集成服务会为某些写入器线程分发多个数据库分区。例如，一个 DB2 for LUW 目标具有五个数据库分区，且最大并行数值为 3。数据集成服务将使用三个写入器线程。数据集成服务会将两个数据库分区分发给第一个写入器线程和第二个写入器线程。服务会为将一个数据库分区分发给第三个写入器线程。

无数据库分区。

数据集成服务将按照最大并行数值所定义的数量使用写入器线程。

但是，当目标具有数据库分区时，您可以优化加载性能。在这种情况下，每个写入器线程均连接至包含数据库分区的 DB2 for LUW 节点。由于写入器线程均连接至不同的 DB2 for LUW 节点，而不是所有线程都连接至一个主节点，所以提高了性能。

分区的关系连接类型

数据集成服务可以使用多个线程根据用于连接到数据库的连接类型来写入到 DB2 for LUW 或 Oracle 关系目标。

您可以使用以下任何连接类型连接至 DB2 for LUW 或 Oracle 数据库：

- DB2 for LUW 连接或 Oracle 连接
- JDBC 连接
- ODBC 连接

要使用多个线程来写入 DB2 for LUW 或 Oracle 关系目标，关系数据对象必须使用 DB2 for LUW 或 Oracle 连接。

如果 DB2 for LUW 或 Oracle 关系数据对象使用 JDBC 或 ODBC 连接，数据集成服务将使用一个线程向目标写入数据。数据集成服务可以使用多个线程处理余下的映射管道阶段。

关系目标分区的规则和准则

为写入到关系目标的映射启用分区时，请注意以下规则和准则：

- 数据集成服务使用一个线程向目标写入数据，但是在以下情况下可以为余下的映射管道阶段使用多个线程：
 - 映射写入到 DB2 for LUW 或 Oracle 以外的关系目标。
 - 映射使用 JDBC 或 ODBC 连接来写入到 DB2 for LUW 或 Oracle 目标。

- DB2 for LUW 目标表分区键为小数列时，请为映射启用高精度。如果分区键为小数列而您没有为映射启用高精度，则数据集成服务可能会使映射失败。

已分区转换

当启用了分区的映射包含支持分区的转换时，数据集成服务可以使用多个线程来运行转换。

数据集成服务会确定是否需要在转换处添加额外的分区点，然后确定该转换管道阶段的最佳线程数量。数据集成服务还会确定是否需要在分区点重新分发数据。例如，数据集成服务可能会在汇总器转换重新分发数据以对汇总表达式的各行进行分组。

某些转换不支持分区功能。当启用了分区的映射包含不支持分区的转换时，数据集成服务将使用一个线程运行转换。数据集成服务可以使用多个线程运行余下的映射管道阶段。

以下转换不支持分区：

- 关联
- 合并
- 异常
- 匹配（针对字段匹配分析而配置时）
- REST Web 服务使用者
- 未连接的查找
- Web 服务使用者

已分区转换的约束

某些支持分区的转换需要进行特定的配置。如果启用了分区的映射包含的转换具有不受支持的配置，数据集成服务将使用一个线程运行转换。数据集成服务可以使用多个线程处理余下的映射管道阶段。

下列转换需要进行特定配置才能支持分区：

- 汇总器转换必须包含一个分组依据端口。汇总器转换不得包含传递端口。汇总器转换不得包括可逐行计算累加值和平均值的数值函数。
- 表达式转换不得包括以下类型的函数或变量：
 - 逐行计算累加值和平均值的数值函数。
 - 有多个线程处理转换时可能返回不同结果的特殊函数。
 - 依赖于上一行的值的局部变量。
- 判定转换、Java 转换和 SQL 转换必须启用**可分区**属性。
- 连接器转换必须包含一个使用等式运算符的联接条件。如果联接条件包含多个相等条件，则这些条件必须使用 AND 运算符进行组合。
- 等级转换必须包含一个分组依据端口。

转换的缓存分区

缓存分区功能为每个处理汇总器转换、连接器转换、等级转换、查找转换或排序器转换的分区创建一个单独的缓存。在缓存分区过程中，每个分区在单独的缓存中存储不同的数据。每个存储包含特定分区所需的行。

通过缓存分区，可以并行让每个线程查询单独的缓存，因此可以优化映射性能。数据集成服务为映射创建了分区时，会始终为已分区的汇总器转换、连接器转换、等级转换和排序器转换使用缓存分区功能。数据集成服务可能会将缓存分区用于已分区的查找转换。

在下列情况下，数据集成服务会将缓存分区用于已连接的查找转换：

- 查找条件仅包含等式运算符。
- 已连接的查找转换查找关系表中的数据时，数据库将配置为区分大小写的比较。

例如，如果查找条件包含字符串端口而数据库未配置为区分大小写的比较，则数据集成服务不会使用缓存分区。

当数据集成服务不对查找转换使用缓存分区时，运行查找转换的所有线程将共享同一缓存。每个线程将按顺序依次查询同一缓存。

注意：数据集成服务不对未连接的查找转换使用缓存分区，因为该服务使用一个线程来运行未连接的查找转换。

已分区缓存的缓存大小

数据集成服务为汇总器转换、连接器转换、等级转换、查找转换和排序器转换使用缓存分区功能时，服务会在各分区间划分缓存大小。

可以在转换高级属性中配置缓存大小。您可以输入以字节为单位的数值，也可以选择**自动**让数据集成服务在运行时决定缓存大小。

如果输入了某个数值，则数据集成服务会于运行时在该数目的转换线程之间划分缓存大小。例如，将转换缓存大小配置为 2,000,000 字节。数据集成服务使用四个线程来运行转换。该服务会划分缓存大小值，以使每个线程最多使用 500,000 字节的缓存大小。

如果选择**自动**，数据集成服务将在运行时决定转换的缓存大小。然后该服务会在该数目的转换线程间划分缓存大小。

优化分区缓存目录

为了在汇总器转换、连接器转换、等级转换和排序器转换的缓存分区期间获得最佳性能，请配置多个缓存目录。

当数据集成服务使用缓存分区功能并且必须将溢出值存储在缓存文件中时，转换线程将数据写入缓存目录。当多个线程向单个目录写入数据时，映射可能会因输入/输出 (I/O) 争用而遇到瓶颈问题。当多个线程同时向文件系统写入数据时，将出现 I/O 争用。

如果配置了多个缓存目录，数据集成服务将以循环方式确定每个转换线程的缓存目录。例如，您将一个汇总器转换配置为使用 directoryA 和 directoryB 作为缓存目录。如果数据集成服务使用四个线程运行该汇总器转换，则第一个和第三个转换线程会将溢出值存储在 directoryA 的缓存文件中。第二个和第四个转换线程将溢出值存储在 directoryB 中的缓存文件中。

如果数据集成服务不为汇总器转换、连接器转换、等级转换或排序器转换使用缓存分区，服务会将溢出值存储在第一个列出的目录中的缓存文件中。

注意：查找转换只能使用单个缓存目录。

在汇总器转换、连接器转换或等级转换的高级属性的**缓存目录**属性中配置缓存目录。在排序器转换的高级属性的**工作目录**属性中配置缓存目录。默认情况下，**缓存目录**和**工作目录**属性被配置为使用为数据集成服务定义的系统参数值。如果管理员为数据集成服务的**缓存目录**或**临时目录**属性输入了多个以分号分隔的目录，则使用默认的 CacheDir 或 TempDir 系统参数值。

可以输入不同值以配置多个特定于转换的缓存目录。为属性或分配给属性的用户定义参数输入多个由分号分隔的目录。

为转换禁用分区

已分区的判定转换、Java 转换或 SQL 转换可能不会在每次映射运行时都返回相同的结果。您可以为这些转换禁用分区，以便数据集成服务使用一个线程处理转换。数据集成服务可以使用多个线程处理余下的映射管道阶段。

在 Java 转换或 SQL 转换中，**可分区**高级属性默认处于选中状态。清除该高级属性将为转换禁用分区。

在判定转换中，**可分区**高级属性默认处于已清除状态。选择该高级属性将为转换启用分区。

您可能希望为转换禁用分区的原因取决于该转换的类型。

判定转换

您可能会希望为使用数值函数的判定转换禁用分区。数值函数 CUME、MOVINGSUM 和 MOVINGAVG 会逐行计算累加值和平均值。如果已分区的判定转换包含这些函数之一，每个线程将单独处理该函数。每个函数使用数据的子集而不是所有数据来计算结果。因此，使用 CUME、MOVINGSUM 或 MOVINGAVG 函数的已分区的转换在每次映射运行时可能不会返回相同的计算结果。

Java 转换

当 Java 代码要求使用一个线程处理转换时，为 Java 转换禁用分区。

SQL 转换

当 SQL 查询要求使用一个线程处理转换时，为 SQL 转换禁用分区。或者，您可能会为了仅建立一个与数据库的连接而希望为 SQL 转换禁用分区。

保持已分区映射中的顺序

您可以在具有已排序的平面文件源、已排序的关系源或排序器转换的映射中设置顺序。数据集成服务向映射添加分区点时，可能会重新分发数据，并失去之前在映射中设置的顺序。要保持已分区映射中的顺序，您必须指定某些转换和写入转换来保持行顺序。

您可以指定以下映射对象保持输入数据的行顺序：

- 表达式转换
- Java 转换
- 序列生成器转换
- SQL 转换
- 写入转换

例如，如果一个关系目标具有一个数据库触发器，而该触发器依赖于按已排序顺序写入的数据，则配置写入转换来保持行顺序。

当您配置写入转换来保持行顺序时，数据集成服务将使用单个线程向目标写入数据。如果某个使用已排序输入的汇总器转换优先于写入转换，数据集成服务将使用单个线程来处理汇总器转换和目标。

当您配置了所有其他转换来保持行顺序时，数据集成服务将在保持顺序的同时为转换管道阶段确定最佳线程数量。

用于配置转换来保持行顺序的方法取决于以下对象类型：

表达式转换、序列生成器转换或 SQL 转换

在表达式转换、序列生成器转换或 SQL 转换的**高级属性**中选择**保持行顺序**属性。

Java 转换

在 Java 转换的**高级属性**中选择**无状态**属性。

写入转换

在写入转换的**高级属性**中选择**保持行顺序**属性。

保持稳定排序

保持已分区映射中的顺序时，数据集成服务不执行稳定排序，而是根据排序键来保持行顺序。但是，如果多行具有相等的排序键值，则行在输出中的相对显示顺序可能与其在输入中的相对显示顺序不同。

例如，启用了分区的映射从包含以下数据且已排序的平面文件源进行读取：

```
Order_ID,Item_ID,Item,Quantity,Price
45,000468,ItemD,5,0.56
45,123456,ItemA,5,3.04
41,456789,ItemB,2,12.02
43,123456,ItemA,3,3.04
```

该映射包含一个指定 Order_ID 作为排序键且方向为降序的排序器转换。当数据集成服务使用多个线程运行排序器转换时，该服务可能并不会保持具有相同 Order_ID 值的行的相对顺序。例如，该服务可能按如下顺序将行写入至合并的目标文件：

```
Order_ID,Item_ID,Item,Quantity,Price
45,123456,ItemA,5,3.04
45,000468,ItemD,5,0.56
43,123456,ItemA,3,3.04
41,456789,ItemB,2,12.02
```

要保持稳定排序，请通过将映射的**最大并行数**运行时属性设置为 1 来禁用映射分区。

替代映射的最大并行数值

默认情况下，每个映射的**最大并行数**属性均设置为“自动”。每个映射都使用为数据集成服务定义的最大并行数值。您可以替代最大并行数值来为特定映射定义最大值。

当数据集成服务和映射的最大并行数设置为不同的整数值时，数据集成服务将使用两者之中的较小值。

鉴于以下原因，您可能希望替代映射的**最大并行数**属性：

运行一个导致线程数大于 CPU 所能处理数量的复杂映射。

能够为整个映射管道运行的并行线程总数为并行值与管道阶段数的乘积。每个分区点均会添加一个额外的管道阶段。包含多个汇总器或连接器转换的复杂映射可能具有很多管道阶段。大量管道阶段会导致数据集成服务使用的线程数超过 CPU 能够处理的数量。

每个管道阶段的并行线程数减少映射性能更能令人满意。

单个映射运行使用较少数量的并行线程时，更多线程可用于数据集成服务运行其他作业。

您希望为某个转换定义一个建议的并行值。

如果您替代某个映射的最大并行数，您可以为该映射中的特定转换定义一个建议的并行值。当转换包含许多端口或执行复杂计算时，您可能会希望定义一个建议的并行值来优化转换的性能。

您想要为地址验证器或匹配转换定义一个执行实例值。

如果您替代了映射的最大并行数，数据集成服务将考虑该映射中地址验证器或匹配转换的执行实例值。您可能希望定义一个执行实例值来优化转换的性能。

转换的建议并行值

如果替代了映射的**最大并行数**运行时属性，则可在映射运行时属性中定义特定转换的**建议的并行值**属性。

只要可以对转换进行分区，数据集成服务便会考虑针对此转换管道阶段的线程数量建议的并行值。例如，如果您将映射配置为保持行顺序，数据集成服务可能需要使用一个线程进行转换。

如果映射的**最大并行数**运行时属性设置为“自动”，则无法为映射中的任何转换定义建议的并行值。在定义转换的建议并行值后，如果将映射最大并行值设为“Auto”，则数据集成服务忽略建议并行值。

当转换包含许多端口或执行复杂计算时，您可能会希望定义一个建议的并行值来优化转换的性能。

例如，如果一个启用了分区的映射处理小型数据集，数据集成服务可能会决定一个线程足以处理表达式转换的管道阶段。但是，如果表达式转换包含许多复杂计算，转换的管道阶段仍然可能需要很长处理时间。输入的建议并行值可以大于 1，但要小于为映射或数据集成服务定义的最大并行数值。数据集成服务将使用针对表达式转换的线程数量建议的并行值。

替代映射的最大并行数时，可以为转换的**建议的并行值**属性配置以下值：

建议的并行值	说明
1	数据集成服务使用一个线程运行转换。
自动	数据集成服务考虑为映射和数据集成服务定义的最大并行数。该服务将使用其中的最低值来确定运行转换的最佳线程数。 此为每个转换的默认设置。
大于 1	数据集成服务会考虑为转换定义的建议并行值、为映射定义的最大并行数和为数据集成服务定义的最大并行数。该服务将使用其中的最低值作为运行转换的线程数。

可以针对以下转换在映射运行时属性中定义**建议的并行值**属性：

- 汇总器
- 表达式
- 筛选器
- Java
- 连接器
- 查找
- 规范器
- 等级
- 路由器
- 序列生成器
- 排序器
- SQL
- 联合
- 更新策略

地址验证器和匹配转换的执行实例

如果您替代了映射的**最大并行数**运行时属性，数据集成服务将考虑为地址验证器或匹配转换定义的**执行实例**高级属性的值。

只要转换可以分区，数据集成服务就会为该转换管道阶段的线程数考虑执行实例值。例如，如果您将映射配置为保持行顺序，数据集成服务可能需要使用一个线程进行转换。

配置转换以进行标识匹配分析时，可以增加匹配转换上的执行实例的数量。配置转换以进行字段匹配分析时，不能增加匹配转换上的执行实例的数量。在字段匹配分析中，匹配转换使用一个执行实例。

如果映射的**最大并行数**运行时属性设置为“自动”，数据集成服务将忽略为地址验证器或匹配转换定义的执行实例值。

替代映射的最大并行数时，可以为地址验证器或匹配转换的**执行实例**高级属性配置以下值：

执行实例值	说明
1	数据集成服务使用一个线程运行转换。 地址验证器转换的默认设置。
自动	数据集成服务考虑为映射和数据集成服务定义的最大并行数。该服务将使用其中的最低值来确定运行转换的最佳线程数。 标识匹配分析中的匹配转换的默认设置。
大于 1	数据集成服务考虑为转换定义的执行实例数、为映射定义的最大并行数以及为数据集成服务定义的最大并行数。该服务将使用其中的最低值作为运行转换的线程数。

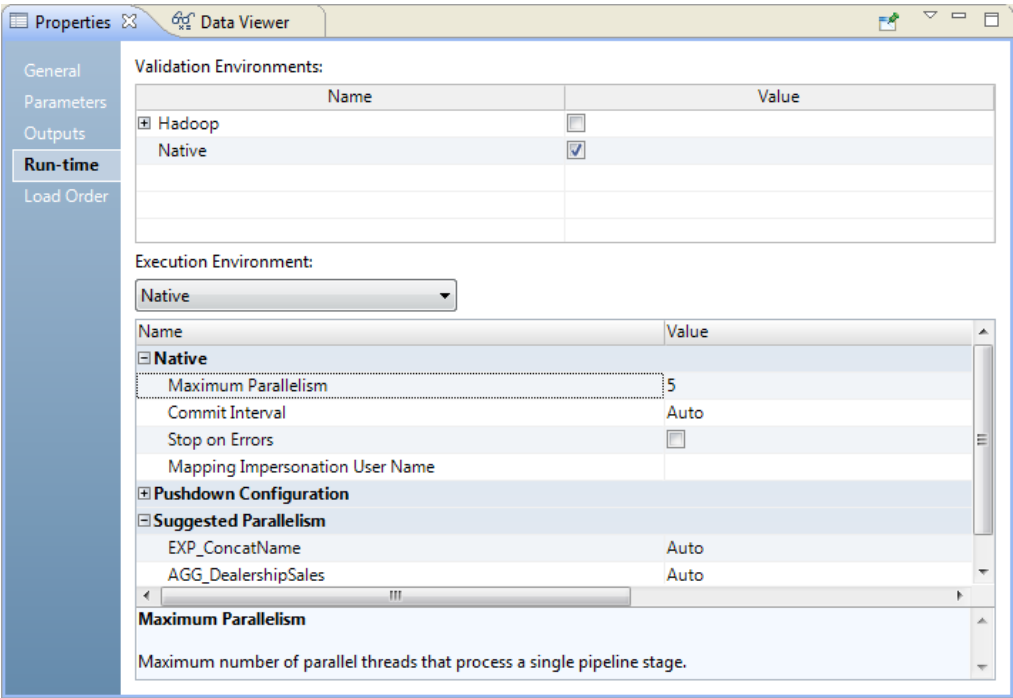
注意：在计算地址验证器转换的最佳线程数时，数据集成服务还会考虑内容管理服务上的“地址对象计数上限”属性。“地址对象计数上限”属性决定了可以在映射中并发运行的最大地址验证实例数。地址对象计数上限值必须大于或等于数据集成服务上的最大并行数值。

替代最大并行数值

要替代最大并行数值，请将映射运行时属性中的最大并行数值设置为大于 1 的整数值，但该值同时要小于为数据集成服务设置的值。

1. 打开映射。
2. 在**属性**视图中，单击**运行时**选项卡。
3. 针对**执行环境**选择**本地**。
4. 对于**最大并行数**属性，输入一个大于 1 的整数值，但该值同时要小于为数据集成服务设置的值。
或者，您可以向属性分配一个用户定义的参数，然后在参数集或参数文件中定义该参数值。
5. 要为映射中的特定转换定义建议的并行值，请在**建议的并行值**部分中为转换输入一个大于 1 的整数值。

下图显示了一个替代了最大并行数值且转换采用了默认建议并行值的映射：



- 6. 保存映射。
- 7. 要为针对标识匹配分析而配置的地址验证器或匹配转换定义执行实例值，请完成以下步骤：
 - a. 打开地址验证器或匹配转换。
 - b. 在高级视图中，为**执行实例**属性输入一个大于 1 的整数值。
 - c. 保存转换。

对已分区映射进行故障排除

针对以下情况的解决方法对于已分区映射的故障排除可能会有所帮助。

数据集成服务的“最大并行数”属性设置为大于 1 的值，映射的“最大并行数”属性设置为“自动”。但是，分区功能在映射运行时处于禁用状态。

要运行具有多个分区的映射，分配给数据集成服务的许可证必须包含分区。如果许可证不包含分区，数据集成服务会将以下消息写入映射日志，并使用一个线程来处理每个映射管道阶段：

WARNING: The Integration Service cannot create partitions for the mapping because the license assigned to the Integration Service does not include partitioning.
INFO: Partitioning is disabled for the mapping.

Developer tool 命名约定

本章包括以下主题：

- [转换命名约定, 244](#)
- [对象类型命名约定, 246](#)
- [工作流对象命名约定, 246](#)

转换命名约定

下表包含转换类型的标准命名约定：

转换类型	建议命名约定
关联	AST_<转换名称>
地址验证器	AGG_<转换名称>
汇总器	AGG_<转换名称>
关联	AST_<转换名称>
离散记录异常	EXC_<转换名称>
大小写转换器	CCO_<转换名称>
分类器	CLA_<转换名称>
比较	CMP_<转换名称>
合并	CNS_<转换名称>
数据屏蔽	DMK_<转换名称>
数据处理器	DPR_<转换名称>
判定	DEC_<转换名称>
重复记录异常	EXC_<转换名称>
表达式	EXP_<转换名称>

转换类型	建议命名约定
筛选器	FIL_<转换名称>
Java	JTX_<转换名称>
联接器	JNR_<转换名称>
键生成器	KGN_<转换名称>
标签创建器	LAB_<转换名称>
查找	LKP_<转换名称>
匹配	MAT_<转换名称>
合并	MRG_<转换名称>
规范器	NRM_<转换名称>
解析器	PRS_<转换名称>
等级	RNK_<转换名称>
读取	SRC_<转换名称>
REST 提供程序	RESTP_<转换名称>
REST Web 服务使用者	RESTWS_<转换名称>
路由器	RTR_<转换名称>
序列生成器	SEQ_<转换名称>
排序器	SRT_<转换名称>
SQL	SQL_<转换名称>
标准创建器	STD_<转换名称>
联合	UN_<转换名称>
更新策略	UPD_<转换名称>
Web 服务使用者	WSC_<转换名称>
加权平均值	WAV_<转换名称>
写入	WRT_<更新类型>_<目标名称>

对象类型命名约定

下表包含存储库对象的标准命名约定：

Developer tool 对象	建议命名约定
应用程序	APP_<应用程序名称>
自定义数据对象	CDO_<数据对象名称>
逻辑数据对象	LDO_<数据对象名称>
逻辑数据对象模型	LDOM_<模型名称>
映射	M_<进程>_<源系统>_<目标名称>
Mapplet	MPLT_<Mapplet 名称>
物理数据对象	PDO_<数据对象名称>
配置文件	PRFL_<配置文件名称>
规则	Rule_<规则名称>
结果卡	SCD_<结果卡名称>
SQL 数据服务	SDS_<数据服务名称>
目标	T_<目标名称>
虚拟架构	VS_<架构名称>
虚拟存储过程	VSP_<过程名称>
虚拟表	VT_<表名称>
Web 服务	WS_<Web 服务名称>

工作流对象命名约定

工作流对象是事件、任务或网关。下表包含工作流对象的标准命名约定：

工作流对象	建议命名约定
分配任务	AST_<说明>
命令任务	CMT_<说明>
独占网关任务	EXG_<说明>
人工任务	HT_<异常表名称>_<说明>

工作流对象	建议命名约定
映射任务	MT_<说明>
通知任务	NTF_<说明>
工作流	WF_<工作流名称>

索引

B

绑定

Mapplet 输出至映射 [77, 88](#)
映射输出至工作流变量 [87](#)

绑定参数

实例值 [66](#)

半联接优化

说明 [202](#)

保留映射输出

绑定至任务输入 [86](#)
规则和准则 [75](#)
如何 [85](#)
使用 infacmd [74](#)
说明 [73](#)

表达式

使用参数 [42](#)
下推优化 [212](#)

表达式参数

描述 [43](#)

表达式转换

动态表达式 [112](#)
已分区 [237](#)
映射输出视图 [71](#)

并发读取分区

说明 [230](#)

并行

减少映射 [240](#)

并行数

建议的 [23](#)
最大值 [22, 23](#)

C

参数

动态映射 [103](#)
关系表属性 [105](#)
目标数据对象 [110](#)
平面文件源 [105](#)
源数据对象 [106](#)
表名称和资源 [46](#)
从 PowerCenter 导入 [193](#)
导出到 PowerCenter [180](#)
逻辑数据对象 [51](#)
平面文件分隔符 [47](#)
如何配置 [60](#)
实例值 [66](#)
虚拟表映射 [51](#)
在 Mapplet 中 [49](#)
在 SQL 语句中 [44](#)
在表达式中 [42](#)

查找转换

缓存分区 [238](#)

重命名生成的端口

示例 [117](#)

船舶数据中心联接优化

说明 [202](#)

出错时停止

映射属性 [18](#)

映射 [22](#)

处理线程

映射 [228](#)

从 PowerCenter 导入

参数转换 [193](#)
冲突解决方案 [186](#)
导入对象 [196](#)
导入限制 [197](#)
导入性能 [197](#)
概览 [185](#)
系统定义的参数 [194](#)
选项 [194](#)
转换类型转换 [187](#)

CUME 函数

分区约束 [239](#)

参数集

创建 [67](#)
概览 [52](#)

参数实例值

设置 [48](#)

参数文件

结构 [53](#)
示例 [55](#)
项目元素 [54](#)
XML 架构定义 [53](#)
应用程序元素 [55](#)
创建 [59](#)
从 Developer tool 导出 [58](#)
目的 [53](#)
运行工作流 [53](#)
运行映射 [53](#)

错误

读取器 [22](#)
写入器 [22](#)
转换 [22](#)

D

等级转换

多个缓存目录 [238](#)
缓存分区 [238](#)
已分区 [237](#)

地址验证器转换

执行实例 [242](#)

动态表达式

示例 [112](#)
创建 [138](#)
概览 [112](#)

动态端口

创建 [132](#)
概览 [110](#)

- 动态端口 (续)
 - 配置 [134](#)
- 动态端口和生成的端口
 - 配置 [111](#)
 - 动态映射
 - 转换 [111](#)
- 动态目标
 - 根据数据对象定义 [108](#)
 - 根据映射流定义 [108](#)
 - 在运行时创建或替换 [108](#)
 - 在运行时获取列 [107](#)
- 动态映射
 - 包含或排除端口 [114](#)
 - 包含所有剩余端口 [115](#)
 - 参数 [103](#)
 - 参数化目标数据对象 [110](#)
 - 参数化源名称 [105](#)
 - 参数化源数据对象 [106](#)
 - 重命名生成的端口 [116](#)
 - 端口选择器 [122](#)
 - 链接解析 [124](#)
 - 配置概览 [101](#)
 - 设计时链接 [123](#)
 - 输入规则 [113-115](#)
 - 重新排序生成的端口 [119](#)
 - 参数化目标名称 [109](#)
 - 创建动态端口 [132](#)
 - 创建运行时链接 [144](#)
 - 定义输入规则 [134](#)
 - 动态端口 [110](#)
 - 端口和链接 [102](#)
 - 概览 [100](#)
 - 故障排除 [128](#)
 - 目标对象 [106](#)
 - 配置写入转换 [140](#)
 - 生成的端口 [110](#)
 - 输入规则 [113-115](#)
 - 选择规则 [122](#)
 - 验证 [147](#)
 - 源 [103](#)
 - 运行 [147](#)
 - 运行时链接 [125, 126](#)
 - 开发和运行 [130](#)
- 动态映射规则
 - 概览 [102](#)
- 动态映射组件
 - 规则 [102](#)
 - 数据源 [101](#)
- 动态源
 - 说明 [103](#)
 - 在运行时获取列 [104](#)
- 端口列表参数
 - 描述 [47](#)
- 端口选择器
 - 示例 [123](#)
 - 选择规则 [122](#)
 - 创建 [137](#)
 - 说明 [121](#)
 - 位于动态表达式中 [112](#)
 - 选择规则 [122](#)
- 端口预览
 - 端口选择器 [121](#)
- 导出
 - 到 PowerCenter [178](#)
- 导出到 PowerCenter
 - 版本兼容性 [179](#)
 - 参数转换 [180](#)
 - 导出对象 [181](#)

- 导出到 PowerCenter (续)
 - 规则和准则 [183](#)
 - 故障排除 [184](#)
 - 设置兼容性级别 [179](#)
 - 限制 [182](#)
 - 选项 [180](#)
 - 概览 [178](#)
- 端口
 - 连接验证 [18](#)
- 段
 - 复制 [16](#)
 - 在映射中 [16](#)
- 对象
 - 命名约定 [246](#)

F

- 反馈绑定
 - 说明 [73](#)
- 范围
 - 端口选择器 [122](#)
- 非关系源
 - 下推优化 [210](#)
- 分区
 - 保持稳定排序顺序 [240](#)
 - 表达式转换 [237](#)
 - 并发读取 [230](#)
 - 等级转换 [237](#)
 - 地址验证器转换 [242](#)
 - 关系连接类型 [232, 236](#)
 - 故障排除 [243](#)
 - 合并文件目标 [233](#)
 - 缓存 [238](#)
 - 缓存大小 [238](#)
 - 汇总器转换 [237](#)
 - IBM DB2 for LUW 目标 [236](#)
 - IBM DB2 for LUW 源 [231](#)
 - Java 转换 [237](#)
 - 减少映射 [240](#)
 - 拒绝文件 [176](#)
 - 联接器转换 [237](#)
 - Oracle 源 [231](#)
 - 判定转换 [237](#)
 - 平面文件目标 [233](#)
 - 平面文件源 [230](#)
 - 匹配转换 [242](#)
 - SQL 转换 [237](#)
 - 维持行顺序 [239](#)
 - 映射 [229](#)
 - 转换 [237](#)
 - 最大并行数 [229](#)
- 分区点
 - 说明 [228](#)
- 分区约束
 - 表达式转换 [237](#)
 - 等级转换 [237](#)
 - 关系目标 [236](#)
 - 关系源 [232](#)
 - 汇总器转换 [237](#)
 - Java 转换 [237](#)
 - 联接器转换 [237](#)
 - 判定转换 [237](#)
 - 数值函数 [239](#)
 - SQL 转换 [237](#)
- 分支修剪优化
 - 说明 [201](#)

G

- 工作流变量
 - 将映射输出绑定至 [87](#)
- 管道阶段
 - 说明 [228](#)
- 关系目标
 - 分区约束 [236](#)
 - 拒绝文件 [176](#)
 - 已分区 [236](#)
- 关系数据对象
 - 拒绝文件 [176](#)
 - 已分区 [231](#), [236](#)
- 关系源
 - 分区约束 [232](#)
 - 下推优化 [210](#)
 - 已分区 [231](#)
- 规则和准则
 - 在运行时创建目标 [109](#)
- 故障排除
 - 动态映射 [128](#)
 - 将对象导出到 PowerCenter [184](#)
- 公开为映射参数
 - 描述 [49](#)
 - 任务说明 [65](#)
 - 说明 [48](#)

H

- Hadoop
 - 验证环境 [21](#)
 - 执行环境 [21](#)
 - Hive 连接 [23](#)
- 行指示器
 - 拒绝文件 [176](#)
- 函数
 - 下推优化 [212](#)
 - 在源中可用 [212](#)
- Hive
 - 连接 [23](#)
 - 执行环境 [23](#)
- Hive 源
 - SQL 查询中的参数 [45](#)
- 缓存大小
 - 分区 [238](#)
- 缓存分区
 - 缓存大小 [238](#)
 - 说明 [238](#)
- 缓存目录
 - 多个目录 [238](#)
 - 优化 [238](#)
- 汇总函数
 - 说明 [73](#)
- 汇总类型
 - 映射输出 [70](#)
- 汇总器转换
 - 多个缓存目录 [238](#)
 - 缓存分区 [238](#)
 - 已分区 [237](#)

I

- IBM DB2 for LUW 目标
 - 已分区 [236](#)
- IBM DB2 for LUW 源
 - 已分区 [231](#)

- IBM DB2 源
 - 下推优化 [210](#)
- infacmd
 - 使用参数集 [52](#)

J

- Java 转换
 - 禁用分区 [239](#)
 - 已分区 [237](#)
- 建议的并行值
 - 转换 [241](#)
- 基于成本的优化
 - 说明 [201](#)
- 拒绝文件
 - 查看 [176](#)
 - 查找 [176](#)
 - 读取 [176](#)
 - 分区 [176](#)
 - 行指示器 [176](#)
 - 列指示器 [177](#)
 - 目标 [176](#)
- 加载顺序
 - 约束 [23](#)

K

- 开发
 - 映射 [26](#)

L

- 链接解析
 - 动态映射 [124](#)
- 联接器转换
 - 多个缓存目录 [238](#)
 - 缓存分区 [238](#)
 - 已分区 [237](#)
- listMappingPersistedOutputs
 - 描述 [74](#)
- 逻辑数据对象
 - 使用参数 [51](#)
 - 映射输出 [79](#)

M

- Mapplet
 - 创建 [34](#)
 - 导出到 PowerCenter [179](#)
 - 概览 [30](#)
 - 规则 [35](#)
 - 结合使用参数与 [49](#)
 - 类型 [31](#)
 - 生成 [32](#)
 - 输出 [32](#)
 - 输入 [32](#)
 - 验证 [34](#)
 - 说明 [16](#)
- Mapplet 参数
 - 示例 [50](#)
- Mapplet 输出
 - 绑定至映射 [76](#), [77](#)
 - 绑定至映射输出 [88](#)
 - 定义 [89](#)

- Mapplet 输出 (续)
 - 如何绑定至映射 [91](#)
- Microsoft SQL Server 源
 - 下推优化 [210](#), [211](#)
 - 下推优化 [210](#), [211](#)
- MOVINGAVG 函数
 - 分区约束 [239](#)
- MOVINGSUM 函数
 - 分区约束 [239](#)
- 目标
 - 合并文件 [233](#)
 - 已分区关系 [236](#)
 - 已分区平面文件 [233](#)
- 目标提交间隔
 - 映射属性 [18](#)
- 命名约定
 - 对象类型 [246](#)
 - 转换 [244](#)
- 目标加载顺序约束
 - 创建 [27](#)
 - 规则和准则 [24](#)
 - 示例 [24](#)
 - 说明 [23](#)

O

- Oracle 源
 - 下推优化 [210](#)
 - 已分区 [231](#)

P

- 排序列表参数
 - 描述 [47](#)
- 排序器转换
 - 缓存分区 [238](#)
- 排序顺序
 - 保持已分区映射中的 [239](#)
- 判定转换
 - 禁用分区 [239](#)
 - 已分区 [237](#)
- 平面文件分隔符
 - 使用参数 [47](#)
- 平面文件目标
 - 多个输出目录 [233](#)
 - 合并分区 [233](#)
 - 拒绝文件 [176](#)
 - 已分区 [233](#)
- 平面文件数据对象
 - 拒绝文件 [176](#)
- 平面文件源
 - 已分区 [230](#)
- 匹配转换
 - 执行实例 [242](#)
- 普通优化级别
 - 说明 [203](#)

R

- 日期参数
 - 有效格式 [39](#)
- 如何
 - 保留映射输出 [85](#)
 - 定义 Mapplet 输出 [89](#)
 - 将 Mapplet 输出绑定至映射 [88](#)

- 如何 (续)
 - 将保留的输出绑定至输入任务 [86](#)
 - 将映射输出绑定至工作流变量 [87](#)
 - 配置参数 [60](#)
 - 配置映射输出 [79](#)
 - 在 Mapplet 中定义输出表达式 [90](#)
 - 在映射中定义输出表达式 [83](#)

S

- SAP HANA 源
 - 下推优化 [211](#)
- SAP 源
 - 下推优化 [211](#)
- 生成的 Mapplet
 - 创建 [33](#)
 - 概览 [32](#)
 - 规则和准则 [33](#)
 - 验证错误 [33](#)
- 生成的端口
 - 重命名 [134](#)
 - 重新排序 [134](#)
 - 概览 [110](#)
- 示例
 - 重命名生成的端口 [117](#)
 - 重新排序生成的端口 [119](#)
 - 动态表达式 [112](#)
 - 动态映射的选择规则 [123](#)
 - 运行时链接 [127](#)
- 输出绑定
 - 对话框说明 [77](#)
- 输出表达式
 - 如何配置 [83](#)
 - 在 Mapplet 中配置 [90](#)
- 输出视图
 - 描述 [70](#)
- 输出文件目录
 - 多个目录 [233](#)
 - 优化 [233](#)
- 数据对象
 - 在运行时获取列 [104](#)
- 顺序
 - 保持已分区映射中的 [239](#)
 - 已分区映射中的稳定排序 [240](#)
- 输入规则
 - 包含或排除端口 [114](#)
 - 包含所有剩余端口 [115](#)
 - 重命名生成的端口 [116](#)
 - 配置 [114](#)
 - 定义 [134](#)
 - 概览 [113](#)
- 输入链接集参数
 - 描述 [47](#)
- SQL 语句
 - 参数 [44](#)
 - 使用 Hive 的参数 [45](#)
- SQL 转换
 - 禁用分区 [239](#)
 - 已分区 [237](#)
- Sybase ASE 源
 - 下推优化 [211](#)
- 实例值
 - 参数设置 [48](#), [66](#)

T

提交间隔
目标 [22](#)

U

updateMappingPersistedOutputs
描述 [74](#)

W

完整优化级别
说明 [203](#)
稳定排序顺序
保持已分区映射中的 [240](#)

X

线程
 处理映射 [228](#)
项目元素
 参数文件 [54](#)
下推优化
 表达式 [212](#)
 关系源 [208](#), [210](#)
 SAP 源 [211](#)
 概览 [205](#)
 Greenplum 源 [211](#)
 关系源 [208](#), [210](#)
 函数 [212](#)
 IBM DB2 源 [210](#)
 Microsoft SQL Server 源 [210](#), [211](#)
 ODBC 源 [211](#)
 Oracle 源 [210](#)
 SAP HANA 源 [211](#)
 Sybase ASE 源 [211](#)
 下推类型 [205](#)
 运算符 [224](#)
 z/OS 上的非关系源 [210](#)
 源 [208](#)
下推优化方法
 配置下推 [207](#)
 完整下推 [206](#)
 源下推 [207](#)
写入转换
 动态 [107](#)
性能优化
 半联接优化方法 [202](#)
 船舶数据中心联接优化方法 [202](#)
 分支修剪优化方法 [201](#)
 基于成本的优化方法 [201](#)
 谓词优化方法 [201](#)
 下推优化 [205](#)
 优化方法 [200](#)
 优化级别 [203](#)
 早期计划优化方法 [200](#)
 早期选择优化方法 [200](#)
系统定义的参数
 导入 [194](#)
选择规则
 动态映射 [122](#)
 示例 [123](#)
 端口选择器 [122](#)

选择条件
 端口选择器 [122](#)
虚拟表映射
 配置参数 [51](#)

Y

映射, 动态
 故障排除 [128](#)
映射参数
 创建位置 [39](#)
 分配位置 [39](#)
 概览 [36](#)
 infacmd [52](#)
 系统 [37](#)
 在虚拟表映射中 [51](#)
 类型 [38](#)
 用户定义 [38](#)
映射管道
 说明 [228](#)
映射管理
 概览 [173](#)
映射任务输入
 将保留的输出绑定至 [86](#)
映射输出
 绑定至 Mapplet [76](#)
 保留 [73](#)
 保留准则 [75](#)
 部署更改 [75](#)
 定义 [70](#)
 概览 [69](#)
 汇总类型 [70](#)
 逻辑数据对象 [79](#)
 配置步骤 [79](#)
 如何定义输出 [82](#)
 使用 infacmd [74](#)
 输出表达式 [71](#)
 输出视图 [70](#)
 系统定义 [69](#)
 用户定义 [70](#)
应用程序元素
 参数文件 [55](#)
优化
 半联接优化方法 [202](#)
 船舶数据中心联接优化方法 [202](#)
 分支修剪优化方法 [201](#)
 基于成本的优化方法 [201](#)
 映射性能方法 [200](#)
 早期计划优化方法 [200](#)
 早期选择优化方法 [200](#)
优化级别
 说明 [203](#)
源
 已分区关系 [231](#)
 已分区平面文件 [230](#)
运算符
 下推优化 [224](#)
 在源中可用 [224](#)
运行时链接
 动态映射 [126](#)
 链接策略 [126](#)
 示例 [127](#)
 创建 [144](#)
 概览 [125](#)
验证环境
 Hadoop [21](#)
 本地执行环境 [21](#)

- 映射
 - 出错时停止 [18](#)
 - 处理线程 [228](#)
 - 对象验证 [18](#)
 - 分区点 [228](#)
 - 管道 [228](#)
 - 环境 [18](#)
 - 减少并行值 [240](#)
 - 拒绝文件 [176](#)
 - 目标提交间隔 [18](#)
 - 谓词优化方法 [201](#)
 - 验证环境 [18](#)
 - 已分区 [229](#)
 - 优化方法 [200](#)
 - 优化映射 [203](#)
 - 运行时属性 [18](#)
 - 执行环境 [18](#)
 - 最大并行数 [18](#), [229](#)
 - 表达式验证 [18](#)
 - 创建 [26](#)
 - 创建约束 [27](#)
 - 从 SQL 查询 [93](#)
 - 对象 [15](#)
 - 概览 [14](#)
 - 工作流 [26](#)
 - 开发 [15](#), [26](#)
 - 连接对象 [27](#)
 - 连接验证 [18](#)
 - 模拟 [22](#)
 - 目标加载顺序约束 [23](#)
 - 视图 [17](#)
 - 添加对象 [26](#)
 - 选项卡 [17](#)
 - 验证 [17](#), [28](#)
 - 运行 [29](#)
- 映射段
 - 已介绍 [16](#)
- 映射模拟
 - 用户名 [22](#)
- 映射视图
 - 已介绍 [17](#)
- 用户名
 - 映射模拟 [22](#)
- 约束
 - 插入行 [23](#)

- 约束 (续)
 - 创建 [27](#)
 - 更新行 [23](#)
 - 目标加载顺序 [23](#)
 - 删除行 [23](#)

Z

- 在运行时创建目标
 - 规则和准则 [109](#)
- 早期计划优化
 - 说明 [200](#)
- 早期选择优化
 - 说明 [200](#)
- 指定依据
 - 值或参数 [43](#)
- 执行实例
 - 地址验证器转换 [242](#)
 - 匹配转换 [242](#)
- 最大并行数
 - 说明 [229](#)
 - 替代 [241](#)
 - 映射运行时 [18](#)
- 最小优化级别
 - 说明 [203](#)
- 执行环境
 - Hadoop [21](#)
 - 验证环境 [21](#)
 - Hadoop [21](#)
- 重新排序生成的端口
 - 示例 [119](#)
 - 动态映射 [119](#)
- 转换
 - 建议的并行值 [23](#), [241](#)
 - 已分区 [237](#)
 - 命名约定 [244](#)
 - 已介绍 [16](#)
- 字符串参数
 - 精度限制 [38](#)