



Informatica PowerCenter Express (Version 9.5.1)

Getting Started Guide



Informatica PowerCenter Express Getting Started Guide

Version 9.5.1
May 2013

Copyright (c) 2013 Informatica Corporation. All rights reserved.

This software and documentation contain proprietary information of Informatica Corporation and are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright law. Reverse engineering of the software is prohibited. No part of this document may be reproduced or transmitted in any form, by any means (electronic, photocopying, recording or otherwise) without prior consent of Informatica Corporation. This Software may be protected by U.S. and/or international Patents and other Patents Pending.

Use, duplication, or disclosure of the Software by the U.S. Government is subject to the restrictions set forth in the applicable software license agreement and as provided in DFARS 227.7202-1(a) and 227.7702-3(a) (1995), DFARS 252.227-7013(1)(ii) (OCT 1988), FAR 12.212(a) (1995), FAR 52.227-19, or FAR 52.227-14 (ALT III), as applicable.

The information in this product or documentation is subject to change without notice. If you find any problems in this product or documentation, please report them to us in writing.

Informatica, Informatica Platform, Informatica Data Services, PowerCenter, PowerCenterRT, PowerCenter Connect, PowerCenter Data Analyzer, PowerExchange, PowerMart, Metadata Manager, Informatica Data Quality, Informatica Data Explorer, Informatica B2B Data Transformation, Informatica B2B Data Exchange Informatica On Demand, Informatica Identity Resolution, Informatica Application Information Lifecycle Management, Informatica Complex Event Processing, Ultra Messaging and Informatica Master Data Management are trademarks or registered trademarks of Informatica Corporation in the United States and in jurisdictions throughout the world. All other company and product names may be trade names or trademarks of their respective owners.

Portions of this software and/or documentation are subject to copyright held by third parties, including without limitation: Copyright DataDirect Technologies. All rights reserved. Copyright © Sun Microsystems. All rights reserved. Copyright © RSA Security Inc. All Rights Reserved. Copyright © Ordinal Technology Corp. All rights reserved. Copyright © Aandacht c.v. All rights reserved. Copyright Genivia, Inc. All rights reserved. Copyright Isomorphic Software. All rights reserved. Copyright © Meta Integration Technology, Inc. All rights reserved. Copyright © Intalio. All rights reserved. Copyright © Oracle. All rights reserved. Copyright © Adobe Systems Incorporated. All rights reserved. Copyright © DataArt, Inc. All rights reserved. Copyright © ComponentSource. All rights reserved. Copyright © Microsoft Corporation. All rights reserved. Copyright © Rogue Wave Software, Inc. All rights reserved. Copyright © Teradata Corporation. All rights reserved. Copyright © Yahoo! Inc. All rights reserved. Copyright © Glyph & Cog, LLC. All rights reserved. Copyright © Thinkmap, Inc. All rights reserved. Copyright © Clearpace Software Limited. All rights reserved. Copyright © Information Builders, Inc. All rights reserved. Copyright © OSS Nokalva, Inc. All rights reserved. Copyright Edifecs, Inc. All rights reserved. Copyright Cleo Communications, Inc. All rights reserved. Copyright © International Organization for Standardization 1986. All rights reserved. Copyright © ej-technologies GmbH. All rights reserved. Copyright © Jaspersoft Corporation. All rights reserved. Copyright © is International Business Machines Corporation. All rights reserved. Copyright © yWorks GmbH. All rights reserved. Copyright © Lucent Technologies. All rights reserved. Copyright (c) University of Toronto. All rights reserved. Copyright © Daniel Veillard. All rights reserved. Copyright © Unicode, Inc. Copyright IBM Corp. All rights reserved. Copyright © MicroQuill Software Publishing, Inc. All rights reserved. Copyright © PassMark Software Pty Ltd. All rights reserved. Copyright © LogiXML, Inc. All rights reserved. Copyright © 2003-2010 Lorenzi Davide. All rights reserved. Copyright © Red Hat, Inc. All rights reserved. Copyright © The Board of Trustees of the Leland Stanford Junior University. All rights reserved. Copyright © EMC Corporation. All rights reserved. Copyright © Flexera Software. All rights reserved. Copyright © Jinfonet Software. All rights reserved. Copyright © Apple Inc. All rights reserved. Copyright © Telerik Inc. All rights reserved. Copyright © BEA Systems. All rights reserved.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>), and/or other software which is licensed under various versions of the Apache License (the "License"). You may obtain a copy of these Licenses at <http://www.apache.org/licenses/>. Unless required by applicable law or agreed to in writing, software distributed under these Licenses is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the Licenses for the specific language governing permissions and limitations under the Licenses.

This product includes software which was developed by Mozilla (<http://www.mozilla.org/>), software copyright The JBoss Group, LLC, all rights reserved; software copyright © 1999-2006 by Bruno Lowagie and Paulo Soares and other software which is licensed under various versions of the GNU Lesser General Public License Agreement, which may be found at <http://www.gnu.org/licenses/lgpl.html>. The materials are provided free of charge by Informatica, "as-is", without warranty of any kind, either express or implied, including but not limited to the implied warranties of merchantability and fitness for a particular purpose.

The product includes ACE(TM) and TAO(TM) software copyrighted by Douglas C. Schmidt and his research group at Washington University, University of California, Irvine, and Vanderbilt University, Copyright (©) 1993-2006, all rights reserved.

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (copyright The OpenSSL Project. All Rights Reserved) and redistribution of this software is subject to terms available at <http://www.openssl.org> and <http://www.openssl.org/source/license.html>.

This product includes Curl software which is Copyright 1996-2007, Daniel Stenberg, <daniel@haxx.se>. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://curl.haxx.se/docs/copyright.html>. Permission to use, copy, modify, and distribute this software for any purpose with or without fee is hereby granted, provided that the above copyright notice and this permission notice appear in all copies.

The product includes software copyright 2001-2005 (©) MetaStuff, Ltd. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://www.dom4j.org/license.html>.

The product includes software copyright © 2004-2007, The Dojo Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://dojotoolkit.org/license>.

This product includes ICU software which is copyright International Business Machines Corporation and others. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://source.icu-project.org/repos/icu/icu/trunk/license.html>.

This product includes software copyright © 1996-2006 Per Bothner. All rights reserved. Your right to use such materials is set forth in the license which may be found at <http://www.gnu.org/software/kawa/Software-License.html>.

This product includes OSSP UUID software which is Copyright © 2002 Ralf S. Engelschall, Copyright © 2002 The OSSP Project Copyright © 2002 Cable & Wireless Deutschland. Permissions and limitations regarding this software are subject to terms available at <http://www.opensource.org/licenses/mit-license.php>.

This product includes software developed by Boost (<http://www.boost.org/>) or under the Boost software license. Permissions and limitations regarding this software are subject to terms available at http://www.boost.org/LICENSE_1_0.txt.

This product includes software copyright © 1997-2007 University of Cambridge. Permissions and limitations regarding this software are subject to terms available at <http://www.pcre.org/license.txt>.

This product includes software copyright © 2007 The Eclipse Foundation. All Rights Reserved. Permissions and limitations regarding this software are subject to terms available at <http://www.eclipse.org/org/documents/epl-v10.php>.

This product includes software licensed under the terms at <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/hsqLicense.html>, <http://httpunit.sourceforge.net/doc/license.html>, <http://jung.sourceforge.net/license.txt>, http://www.gzip.org/zlib/zlib_license.html, <http://www.openldap.org/software/release/license.html>, <http://www.libssh2.org>, <http://slf4j.org/license.html>, <http://www.sente.ch/software/OpenSourceLicense.html>, <http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3-license-agreement>, <http://antlr.org/license.html>, <http://aopalliance.sourceforge.net/>, <http://www.bouncycastle.org/license.html>, <http://www.jgraph.com/jgraphdownload.html>, <http://www.jcraft.com/jsch/LICENSE.txt>, http://jotm.objectweb.org/bsd_license.html, <http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231>, <http://www.slf4j.org/license.html>, <http://nanoxml.sourceforge.net/orig/copyright.html>, <http://www.json.org/license.html>, <http://forge.ow2.org/projects/javaservice/>, <http://www.postgresql.org/about/licence.html>,

www.sqlite.org/copyright.html, <http://www.tcl.tk/software/tcltk/license.html>, <http://www.jaxen.org/faq.html>, <http://www.jdom.org/docs/faq.html>, <http://www.slf4j.org/license.html>; <http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License>; <http://www.keplerproject.org/md5/license.html>; <http://www.toedter.com/en/jcalendar/license.html>; <http://www.edankert.com/bounce/index.html>; <http://www.net-snmp.org/about/license.html>; <http://www.openmdx.org/#FAQ>; http://www.php.net/license/3_01.txt; <http://srp.stanford.edu/license.txt>; <http://www.schneier.com/blowfish.html>; <http://www.jmock.org/license.html>; <http://xsom.java.net>; and <http://benalman.com/about/license/>; <https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js>; <http://www.h2database.com/html/license.html#summary>; and <http://jsoncpp.sourceforge.net/LICENSE>.

This product includes software licensed under the Academic Free License (<http://www.opensource.org/licenses/afl-3.0.php>), the Common Development and Distribution License (<http://www.opensource.org/licenses/cddl1.php>) the Common Public License (<http://www.opensource.org/licenses/cpl1.0.php>), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (<http://www.opensource.org/licenses/bsd-license.php>) the MIT License (<http://www.opensource.org/licenses/mit-license.php>) and the Artistic License (<http://www.opensource.org/licenses/artistic-license-1.0>).

This product includes software copyright © 2003-2006 Joe Walnes, 2006-2007 XStream Committers. All rights reserved. Permissions and limitations regarding this software are subject to terms available at <http://xstream.codehaus.org/license.html>. This product includes software developed by the Indiana University Extreme! Lab. For further information please visit <http://www.extreme.indiana.edu/>.

This Software is protected by U.S. Patent Numbers 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,640,226; 6,789,096; 6,820,077; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; and 7,774,791, international Patents and other Patents Pending.

DISCLAIMER: Informatica Corporation provides this documentation "as is" without warranty of any kind, either express or implied, including, but not limited to, the implied warranties of noninfringement, merchantability, or use for a particular purpose. Informatica Corporation does not warrant that this software or documentation is error free. The information provided in this software or documentation may include technical inaccuracies or typographical errors. The information in this software and documentation is subject to change at any time without notice.

NOTICES

This Informatica product (the "Software") includes certain drivers (the "DataDirect Drivers") from DataDirect Technologies, an operating company of Progress Software Corporation ("DataDirect") which are subject to the following terms and conditions:

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.
2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.

Part Number: PCX-GSG-95100-0001

Table of Contents

Preface	iii
Informatica Resources.	iii
Informatica My Support Portal.	iii
Informatica Documentation.	iii
Informatica Web Site.	iii
Informatica Support YouTube Channel.	iii
Informatica Marketplace.	iv
Informatica Velocity.	iv
Chapter 1: Introduction to PowerCenter Express	1
PowerCenter Express Overview.	1
PowerCenter Express Example.	1
PowerCenter Express Architecture.	2
Informatica Developer.	3
Using Informatica Developer.	3
Informatica Developer User Interface.	4
Informatica Administrator.	5
Using Informatica Administrator.	5
Informatica Administrator User Interface.	6
Data Integration Process.	7
Chapter 2: Setting Up Informatica Developer	8
Setting Up Informatica Developer Overview.	8
Step 1. Log In to the Administrator Tool.	9
Step 2. Start Informatica Developer.	11
Step 3. Connect to the Repository.	12
Adding the Domain and Repository.	13
Connecting to the Model Repository.	16
Step 4. Create a Project.	16
Step 5. Create a Folder.	17
Step 6. Select the Default Data Integration Service.	18
PowerCenter Express Tips.	19
Chapter 3: Reviewing Source Data	20
Reviewing Source Data Overview.	20
Step 1. View a Mapping Source.	21
Step 2. Run a Column Profile.	24
PowerCenter Express Tips.	27

Chapter 4: Developing a Mapping.....	28
Developing a Mapping Overview.	28
Step 1. Create a Mapping.	29
Step 2. Add a Transformation to the Mapping.	29
Step 3. Add a Target to the Mapping.	33
Step 4. Run the Mapping and Review Results.	36
PowerCenter Express Tips.	38
Chapter 5: Running a Workflow.....	39
Running a Workflow Overview.	39
Step 1. Review the Mapping.	40
Step 2. Create a Workflow.	42
Step 3. Deploy and Run the Workflow.	44
PowerCenter Express Tips.	45
Chapter 6: Monitoring a Workflow.....	46
Monitoring a Workflow Overview.	46
Step 1. Access the Monitoring Tool.	47
Step 2. Monitor the Workflow and View Logs.	48
PowerCenter Express Tips.	49
Chapter 7: Creating a Connection.....	50
Creating a Connection Overview.	50
Step 1. Create a Connection.	51
PowerCenter Express Tips.	54
Appendix A: Glossary.....	55

Preface

The PowerCenter Express *Getting Started Guide* is written for developers. It provides a tutorial to help first-time users learn how to use Informatica Developer and Informatica Administrator for PowerCenter Express tasks. This guide assumes that you have an understanding of flat file concepts, relational database concepts, and the database engines in your environment.

Informatica Resources

Informatica My Support Portal

As an Informatica customer, you can access the Informatica My Support Portal at <http://mysupport.informatica.com>.

The site contains product information, user group information, newsletters, access to the Informatica How-To Library, the Informatica Knowledge Base, Informatica Product Documentation, and access to the Informatica user community.

Informatica Documentation

The Informatica Documentation team takes every effort to create accurate, usable documentation. If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com. We will use your feedback to improve our documentation. Let us know if we can contact you regarding your comments.

The Documentation team updates documentation as needed. To get the latest documentation for your product, navigate to Product Documentation from <http://mysupport.informatica.com>.

Informatica Web Site

You can access the Informatica corporate web site at <http://www.informatica.com>. The site contains information about Informatica, its background, upcoming events, and sales offices. You will also find product and partner information. The services area of the site includes important information about technical support, training and education, and implementation services.

Informatica Support YouTube Channel

You can access the Informatica Support YouTube channel at <http://www.youtube.com/user/INFASupport>. The Informatica Support YouTube channel includes videos about solutions that guide you through performing specific tasks. If you have questions, comments, or ideas about the Informatica Support YouTube channel, contact the Support YouTube team through email at supportvideos@informatica.com or send a tweet to [@INFASupport](https://twitter.com/INFASupport).

Informatica Marketplace

The Informatica Marketplace is a forum where developers and partners can share solutions that augment, extend, or enhance data integration implementations. By leveraging any of the hundreds of solutions available on the Marketplace, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at <http://www.informaticamarketplace.com>.

Informatica Velocity

You can access Informatica Velocity at <http://mysupport.informatica.com>. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions. If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

CHAPTER 1

Introduction to PowerCenter Express

This chapter includes the following topics:

- ◆ PowerCenter Express Overview, 1
- ◆ PowerCenter Express Architecture, 2
- ◆ Informatica Developer, 3
- ◆ Informatica Administrator, 5
- ◆ Data Integration Process, 7

PowerCenter Express Overview

Use PowerCenter Express to design and implement data integration solutions.

You can use PowerCenter Express to extract data from multiple sources, transform the data according to business logic that you build in the client application, and load the transformed data to targets. You can also run a profile to analyze the structure and content of your data, and to determine the quality of your data.

You can access data in relational databases, flat files, web services, and social media web sites.

PowerCenter Express includes the Informatica domain, application services that process data, repositories to store metadata, Informatica Administrator (the Administrator tool), and Informatica Developer (the Developer tool).

The Administrator tool, the repositories, and application services run on a single machine. The Developer tool can run on one or more machines.

PowerCenter Express Example

Organizations can use PowerCenter Express to perform data integration tasks. For example, use PowerCenter Express to consolidate data.

An organization wants to consolidate data from three flat files that contain customer records from different geographic areas. After analyzing the content of the data, the organization wants to load the data to a relational database table.

To complete this task, the organization uses the Developer tool to run a profile on the data, consolidate the data, and write the consolidated data to a relational database. Then, the organization uses the Administrator tool to monitor the progress of the profile job and the progress of the workflow that consolidates the data.

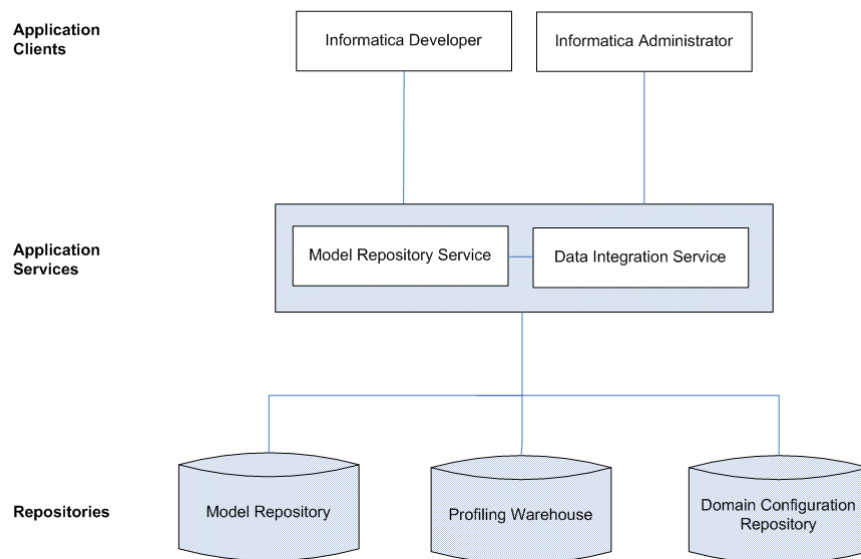
PowerCenter Express Architecture

The PowerCenter Express tools, application services, and repositories are components that run within the Informatica domain.

The Informatica domain is the fundamental administrative unit in Informatica. The Informatica domain contains the following components:

- ◆ Application clients. A group of clients that you use to access underlying Informatica functionality. Application clients make requests to the Service Manager or application services.
- ◆ Application services. A group of services that represent server-based functionality. You configure the application services that are required by the application clients that you use.
- ◆ Repositories. A group of relational databases that store metadata about objects and processes required to handle user requests from application clients.
- ◆ Service Manager. A service that is built in to the domain to manage all domain operations. The Service Manager runs the application services and performs domain functions including authentication, authorization, and logging.

The following figure shows the PowerCenter Express components that run within the Informatica domain:



The Informatica domain includes the following PowerCenter Express components:

Informatica Administrator

Informatica Administrator (the Administrator tool) is an application client that consolidates the administrative tasks for domain objects such as services, connections, and licenses. You manage the domain and the security of the domain through the Administrator tool.

Informatica Developer

Informatica Developer (the Developer tool) is an application client that you can use to design and implement data integration solutions.

Data Integration Service

The Data Integration Service is an application service that performs data integration tasks for Informatica Developer and external clients. Data integration tasks include previewing data and running profiles, mappings, and workflows.

Model Repository Service

The Model Repository Service is an application service that manages the Model repository.

Model repository

The Model repository is a relational database that stores the metadata for projects. The Model repository also stores run-time and configuration information for applications that are deployed to a Data Integration Service.

Domain configuration repository

The domain configuration repository is set of domain metadata tables stored in a relational database. Each time you make a change to the domain, the Service Manager writes the change to the domain configuration repository.

Profiling warehouse

The profiling warehouse is a relational database that the Data Integration Services uses to store profiling results.

Informatica Developer

Informatica Developer (the Developer tool) is an application client that you can use to design and implement data integration solutions.

You can use the Developer tool to import metadata, create connections, and create logical data objects. You can also use the Developer tool to create and run profiles, mappings, workflows.

Using Informatica Developer

Use the Developer tool to complete the tasks required to implement your data integration project.

You can use the Developer tool to complete the following tasks:

Import metadata for sources and targets.

Import metadata for sources and targets that you want to use in a mapping.

Define logical views of data.

A logical view of data describes the structure and use of data in an enterprise. You can create a logical data object model that shows the types of data your enterprise uses and how that data is structured. Then, you can create a mapping that links objects in a logical model to data sources or targets.

Run a profile.

Run a profile to analyze the structure and content of your data, and to determine the quality of your data.

Develop mappings.

Develop mappings to implement data integration tasks.

Create and run workflows.

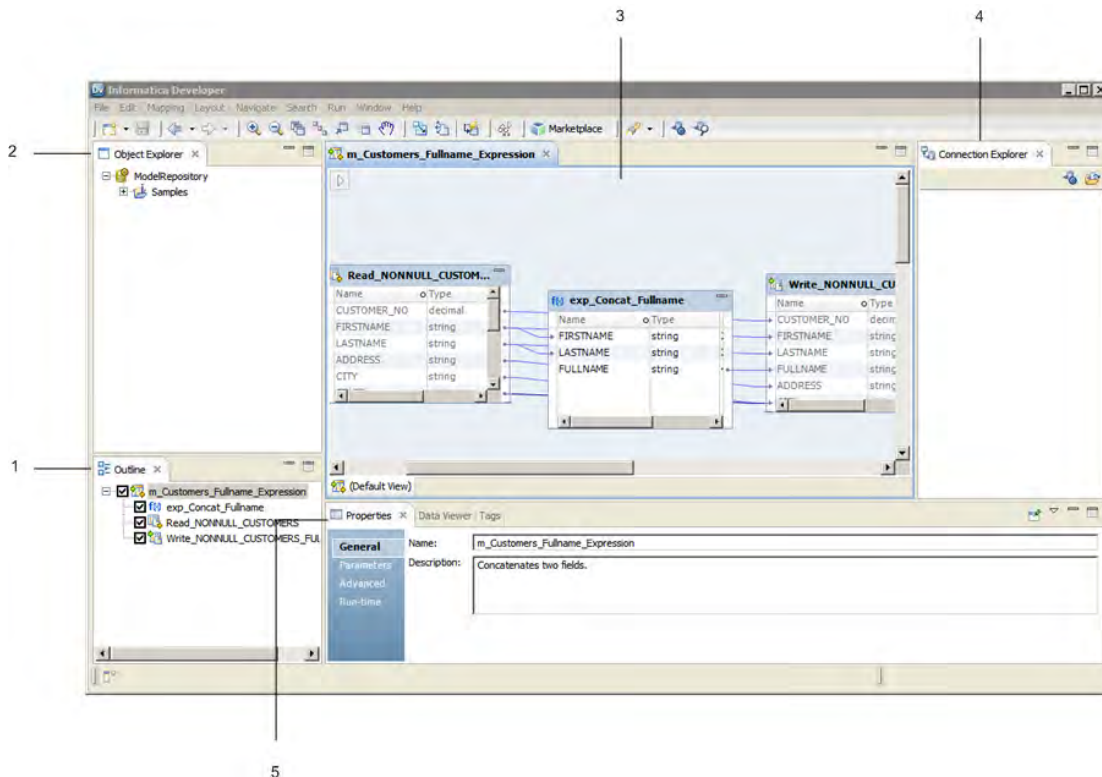
Create and run workflows to perform a sequence of events, tasks, and decisions that you can configure based on your business process requirements.

Informatica Developer User Interface

The Developer tool is an application that you use to design and implement data integration solutions. The Developer tool workbench includes an editor and views.

You edit objects, such as mappings, in the editor. The Developer tool displays views, such as the **Properties** view, based on which object is in focus in the editor and your selection of which views you want to display.

The following figure shows the Developer tool workbench:



1. Outline view
2. Object Explorer view
3. Editor
4. Connection Explorer view
5. Properties view

The Developer tool displays the following views by default:

Outline view

Displays objects that are dependent on an object selected in the **Object Explorer** view. By default, this view appears in the bottom left area of the Developer tool.

Object Explorer view

Displays projects, folders, and the objects within the projects and folders. By default, this view appears in the top left area of the Developer tool.

Connection Explorer view

Displays connections to relational databases. By default, this view appears in the top right area of the Developer tool.

Properties view

Displays the properties for an object that is in focus in the editor. By default, this view appears in the bottom area of the Developer tool.

You can hide views and move views to another location in the Developer tool workbench. Click **Window > Show View** to select the views you want to display.

The Developer tool workbench also displays the following views:

Cheat Sheets view

Displays the cheat sheet that you open. To open a cheat sheet, click **Help > Cheat Sheets** and select a cheat sheet.

Data Viewer view

Displays source data and previews the output of a transformation.

Help view

Displays context-sensitive online help.

Progress view

Displays the progress of operations in the Developer tool, such as a mapping run.

Search view

Displays the search results. You can also launch the search options dialog box.

Tags view

Displays tags that define an object in the Model repository based on business usage.

Validation Log view

Displays object validation errors.

Informatica Administrator

Informatica Administrator (the Administrator tool) is an application client that consolidates the administrative tasks for domain objects such as services, connections, and licenses.

You manage the domain and the security of the domain through the Administrator tool.

Using Informatica Administrator

Use the Administrator tool to manage the domain and the security of the domain.

Use the Administrator tool to complete the following types of tasks:

Domain administrative tasks

Manage logs, domain objects, and user permissions. Monitor jobs and applications that run on the Data Integration Service.

Security administrative tasks

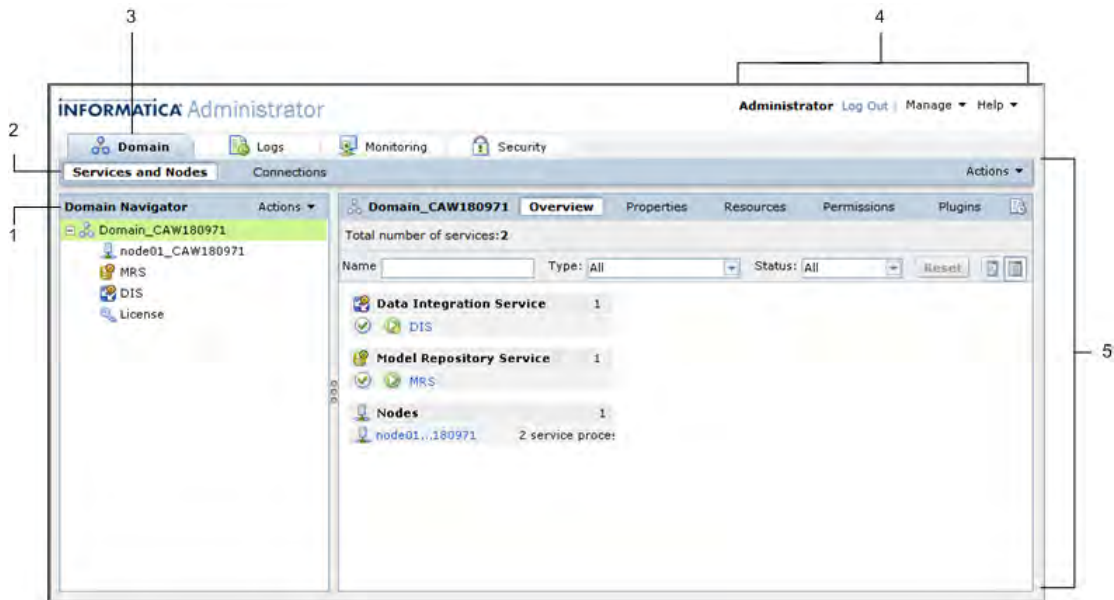
Manage users, groups, roles, and privileges.

Note: If you have PowerCenter Express Personal Edition, you do not have access to the administer security.

Informatica Administrator User Interface

The Administrator tool is an application that you use to manage the Informatica domain and the security of the Informatica domain. The Administrator tool interface contains tabs, header items, views, a navigator, and a contents panel.

The following figure shows the Administrator tool:



1. Navigator
2. View in the tab
3. Tab
4. Header area
5. Contents panel

The tabs and views that are available in the Administrator tool differs based on your product license and user permissions. The Navigator displays a hierarchy of objects. The types of objects in the navigator differ based on the tab that you select. The contents panel displays details about the object that you select in the navigator.

The Administrator tool has the following tabs:

Domain

View and edit the properties of the domain and objects within the domain.

Logs

View log events for the domain and services within the domain.

Monitoring

View the status of profile jobs, preview jobs, mapping jobs, and workflows for the Data Integration Service.

Security

Manage users, groups, roles, and privileges. If you have PowerCenter Express Personal Edition, you do not have access to the Security tab.

The Administrator tool has the following header items:

Log out

Log out of the Administrator tool.

Manage

Manage your account.

Help

Access help for the current tab, determine the Informatica version, and configure the data usage policy.

Data Integration Process

Use the PowerCenter Express client applications to complete your data integration solutions. Use the Developer tool to create connections, import metadata, run profiles, create mappings, and run mappings as part of workflows. Then, use the Administrator tool to monitor workflow progress.

Step 1. Create connections

Create connections to access data from relational databases, third-party web services, or social media web sites. Create a connection to create data objects, preview data, run profiles, and run mappings. The Developer tool uses the connection when you import a data object. The Data Integration Service uses the connection when you preview data, run profiles, and run mappings.

Step 2. Import metadata to create data objects

Import metadata to create data objects for sources and targets that you want to use in a mapping. When you develop a mapping, you can use data objects to define the input and output of the mapping.

Step 3. Run a profile

Run a profile to analyze the structure and content of your data, and to determine the quality of your data. When you run a profile, the Data Integration Service applies the profiling rules and runs the profile.

Step 4. Develop mappings

Develop mappings to implement data integration tasks. A mapping is a set of inputs and outputs that represent the data flow between sources and targets. Link the sources and targets with transformation objects that define the rules for data transformation. The Data Integration Service uses the instructions configured in the mapping to read, transform, and write data. You can add a mapping as a task in a workflow.

Step 5. Create and run workflows

Create a workflow to define a sequence of events, tasks, and decisions based on a business process. Then, deploy the workflow to the Data Integration Service and run the workflow. The Data Integration Service uses the instructions configured in the workflow to run the objects.

Step 6. Monitor workflows

Monitor the workflow instance run on the Monitoring tab of the Administrator tool. The Monitoring tab shows the status of running workflow and workflow object instances. You can abort or cancel a running workflow instance in the Monitoring tool. You can also use the Monitoring tool to view logs for workflow instances and to view workflow reports.

CHAPTER 2

Setting Up Informatica Developer

This chapter includes the following topics:

- ◆ Setting Up Informatica Developer Overview, 8
- ◆ Step 1. Log In to the Administrator Tool, 9
- ◆ Step 2. Start Informatica Developer, 11
- ◆ Step 3. Connect to the Repository, 12
- ◆ Step 4. Create a Project, 16
- ◆ Step 5. Create a Folder, 17
- ◆ Step 6. Select the Default Data Integration Service, 18
- ◆ PowerCenter Express Tips, 19

Setting Up Informatica Developer Overview

In this lesson, you start and set up the Developer tool. You log in to the Administrator tool to record information that you need to connect the Developer tool to the Informatica domain. To set up the Developer tool, you connect to the Model repository and create a project and folder to store your work. You select the default Data Integration Service to preview data and run mappings.

Lesson Concepts

The Informatica domain is a collection of services that perform data integration tasks and monitoring tasks. You manage the domain through the Administrator tool.

The Model Repository Service manages the Model repository. The Model repository is a relational database that stores the metadata for projects and folders. A project stores objects that you create in the Developer tool. A project can also contain folders that store related objects that are part of the same business requirement.

The Data Integration Service performs data integration tasks for the Developer tool. Data integration tasks include previewing data, and running profiles, mappings, and workflows.

Lesson Objectives

In this lesson, you complete the following tasks:

- ◆ Log in to the Administrator tool to record domain information.
- ◆ Start the Developer tool.
- ◆ Configure the Developer tool to connect to the Model repository.
- ◆ Create a project to store the objects that you create in the Developer tool.

- ◆ Create a folder in the project that can store related objects.
- ◆ Select the default Data Integration Service to preview data and to run mappings in the Developer tool.

Lesson Prerequisites

Before you start this lesson, verify the following prerequisites:

- ◆ You have installed Informatica PowerCenter Express.
- ◆ The Informatica services are running in the domain.

Lesson Timing

Set aside 10 to 15 minutes to complete the tasks in this lesson.

Step 1. Log In to the Administrator Tool

If you installed Informatica services and the Informatica client separately, you must log in to the Administrator tool to record the Informatica domain connection information. You must have the domain name, host name, and port number to add the domain to the Developer tool.

If you installed Informatica services and the Informatica client at the same time on the same machine, the installation process adds the domain to the Developer tool. You can skip this step and continue to “Step 2. Start Informatica Developer” on page 11.

1. From the Windows Start menu, click **Informatica PowerCenter Express > Launch Informatica Administrator**.

The default browser opens to the Administrator tool login page.

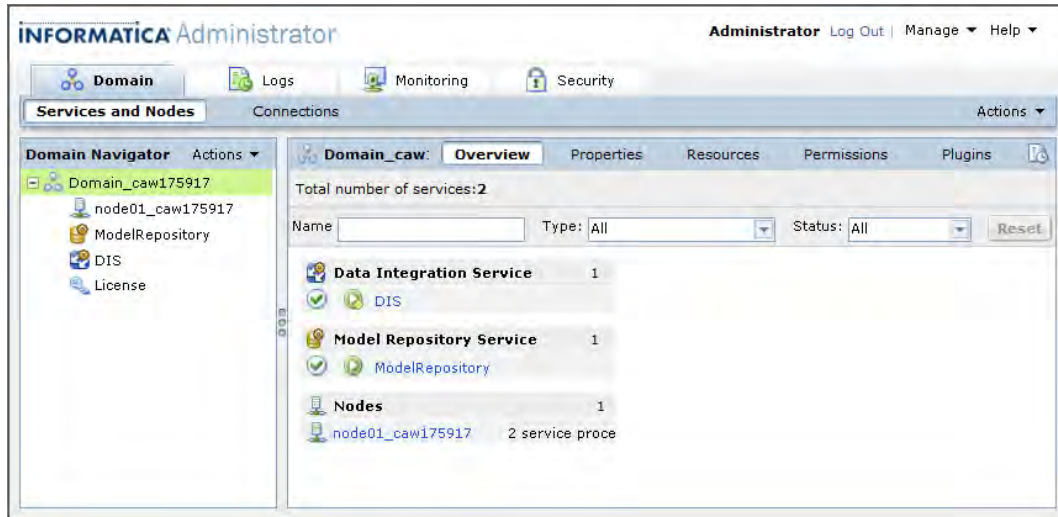
2. Enter the user name and password provided to you.
3. Click **Log In**.

The **Domain** tab of the Administrator tool opens.

4. Record the name of the domain that appears in the **Domain Navigator**.

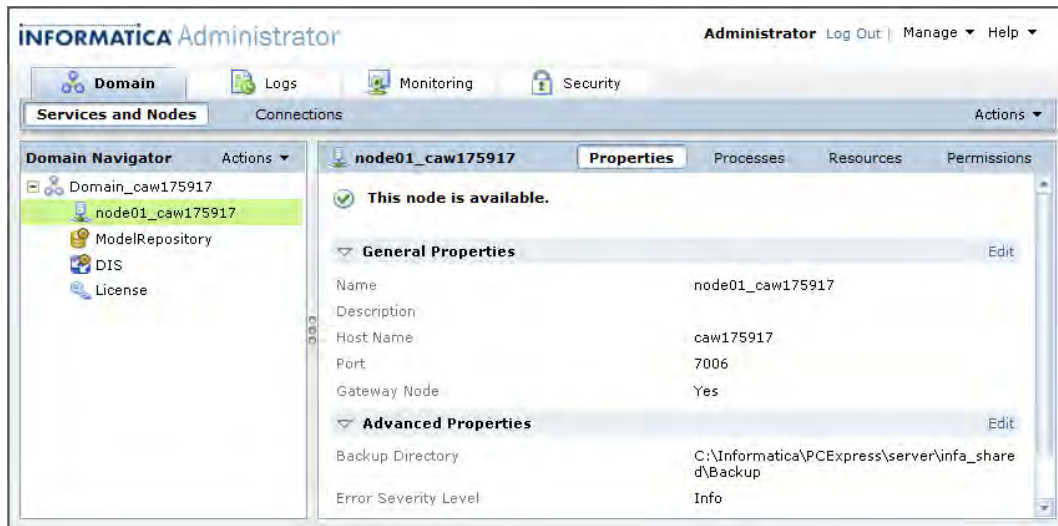
The default domain name is Domain_<machine_name>, where <machine_name> is the name of the machine where Informatica services is installed.

In the following figure, the domain name is Domain_caw175917:



5. In the **Domain Navigator**, select the node.
6. In the **Properties** view, record the host name and port in the **General Properties** section. The default host name is the name of the machine where Informatica services is installed.

In the following figure, the host name is caw175917 and the port number is 7006:



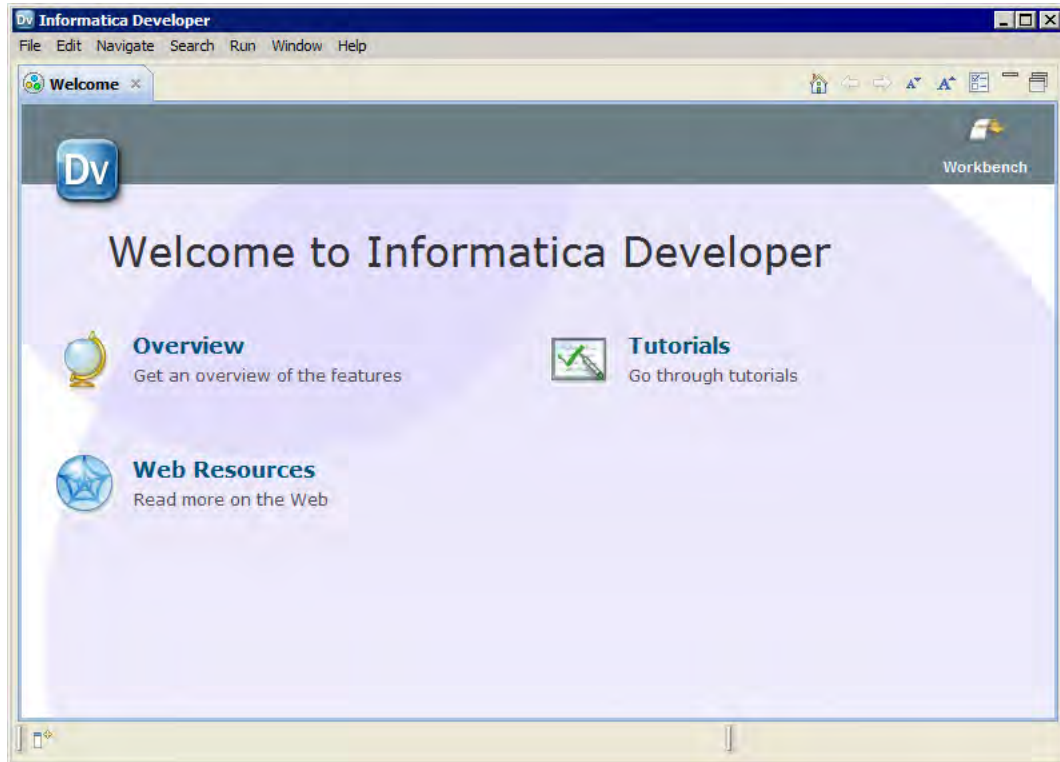
7. Click **Log Out** to log out of the Administrator tool.

Step 2. Start Informatica Developer

Start the Developer tool to begin the lesson.

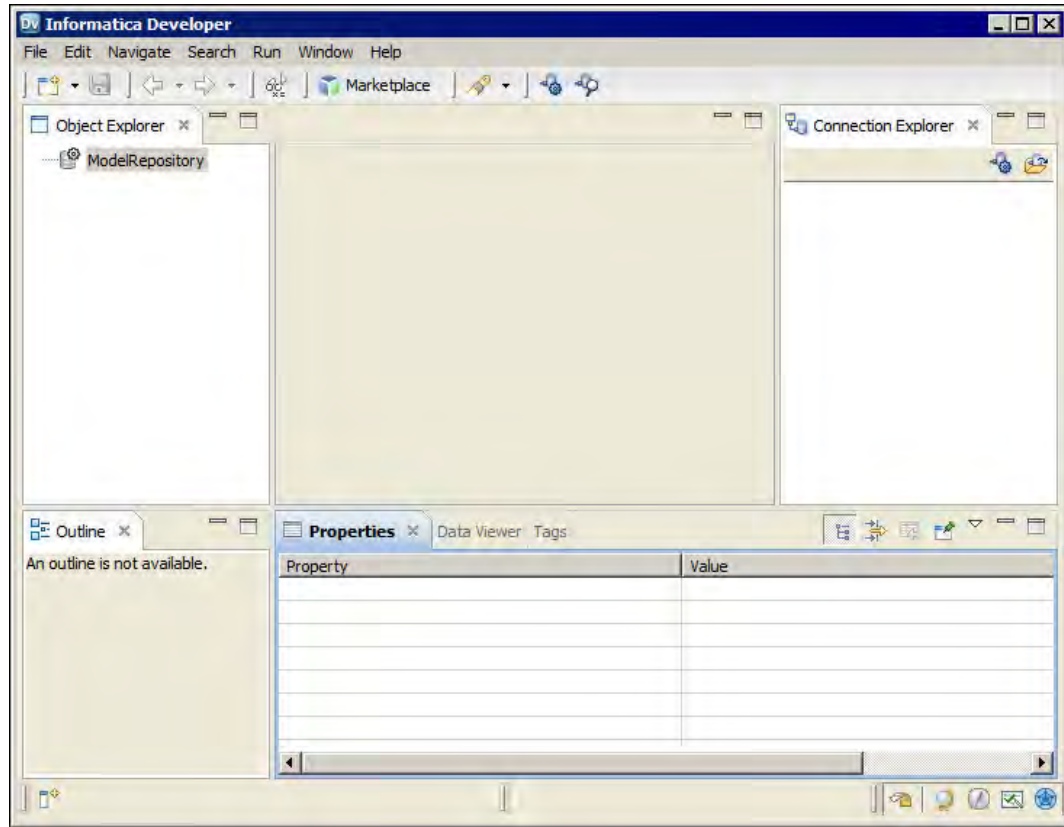
1. From the Windows Start menu, click **Informatica PowerCenter Express > Launch Informatica Developer**.

The **Welcome** page of the Developer tool appears. If you have started the Developer tool before, the Developer tool opens to the **Workbench**.



2. Click the **Workbench** link.

The Developer tool **Workbench** appears. If you installed Informatica services and the Informatica client separately, the **Object Explorer** view is empty. If you installed Informatica services and the Informatica client at the same time on the same machine, the **Object Explorer** view displays the Model repository with the default name ModelRepository.



Step 3. Connect to the Repository

Connect to the Model repository to create, view, and manage metadata.

The steps that you use to connect to the Model repository depend on the following installation methods:

You installed Informatica services and the Informatica client separately.

You must add the domain and Model repository the first time that you set up the Developer tool. For instructions, see “Adding the Domain and Repository” on page 13.

You installed Informatica services and the Informatica client at the same time on the same machine.

The installation process adds the domain and Model repository. You can connect to the Model repository the first time that you set up the Developer tool. For instructions, see “Connecting to the Model Repository” on page 16.

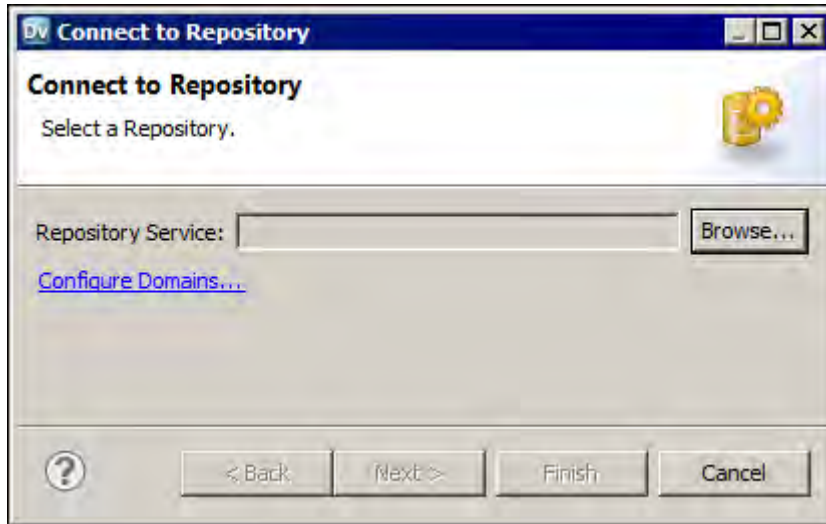
Adding the Domain and Repository

If you installed Informatica services and the Informatica client separately, add the domain and Model repository to the Developer tool.

When you add the domain, use the domain name, host name, and port number that you recorded in “Step 1. Log In to the Administrator Tool” on page 9.

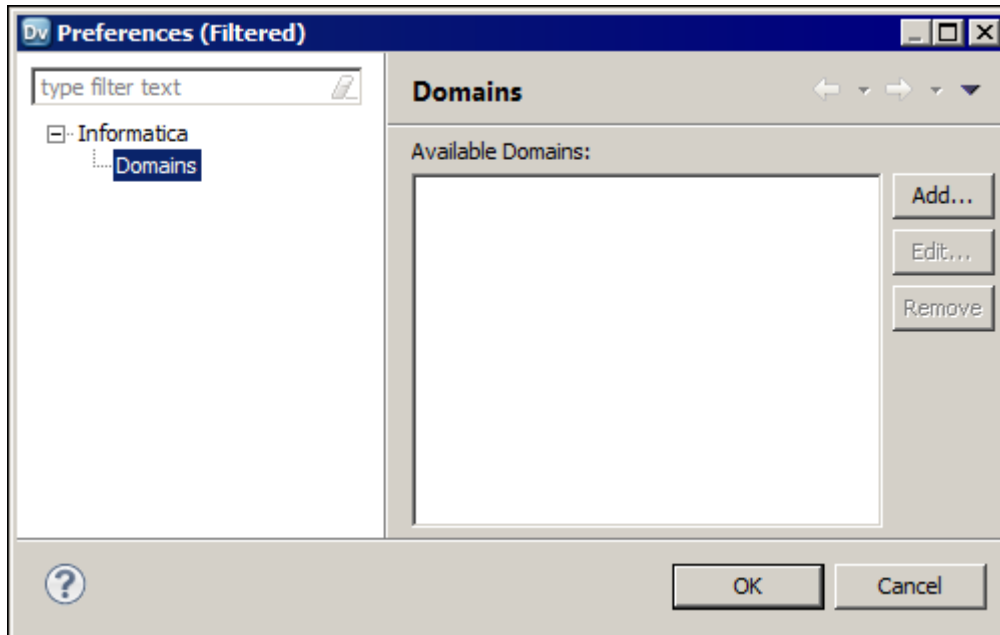
1. From the Developer tool menu, click **File > Connect to Repository**.

The **Connect to Repository** dialog box appears.



2. Click **Configure Domains**.

The **Preferences** dialog box appears.



3. Click **Add**.

The **New Domain** dialog box appears.

4. Enter the domain name, host name, and port number for the domain.

The following table lists the default values for the domain:

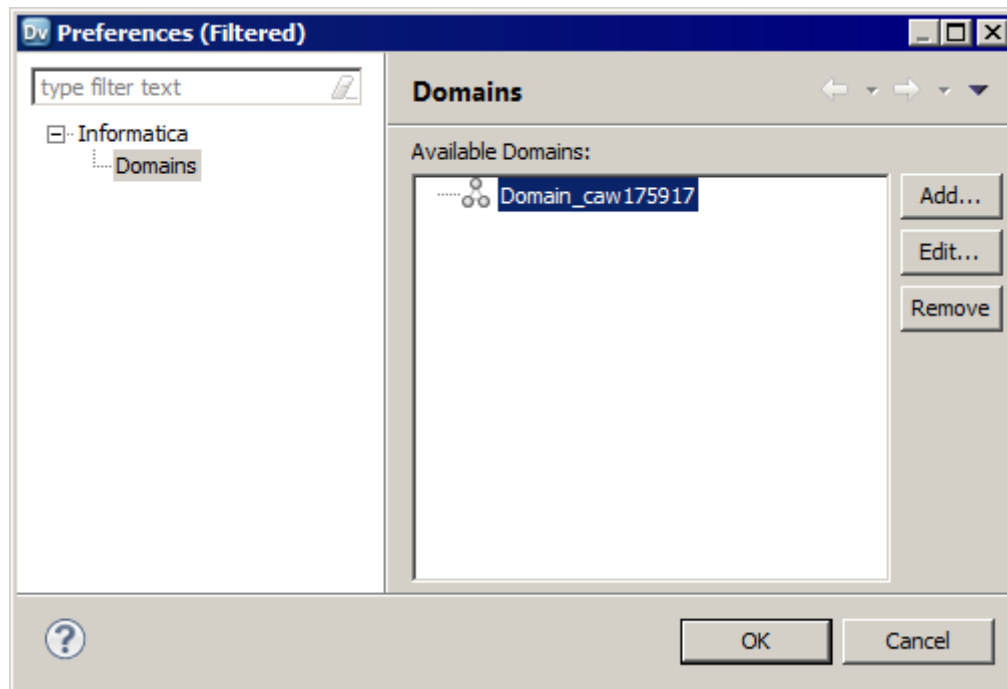
Property	Default Value	Example
Domain name	Domain_<machine name>	Domain_caw175917
Host name	<machine name>	caw175917
Port number	7006	7006

5. Click **Test Connection** to test the connection.

The Developer tool displays a message that states whether the connection is successful.

6. Click **Finish**.

The domain appears in the **Available Domains** panel of the **Preferences** dialog box.



7. Click **OK**.

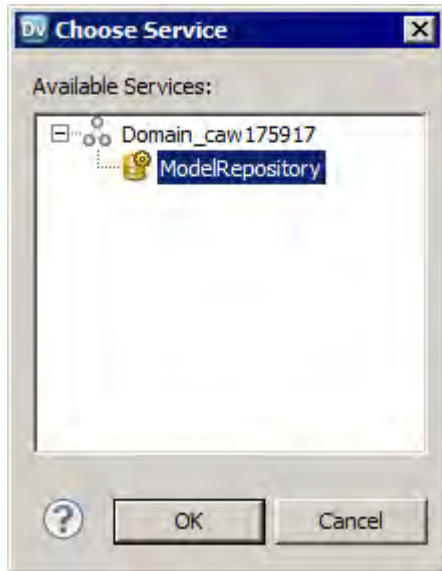
The **Connect to Repository** dialog box appears.

8. Click **Browse** to select a Model repository.

The **Choose Service** dialog box appears.

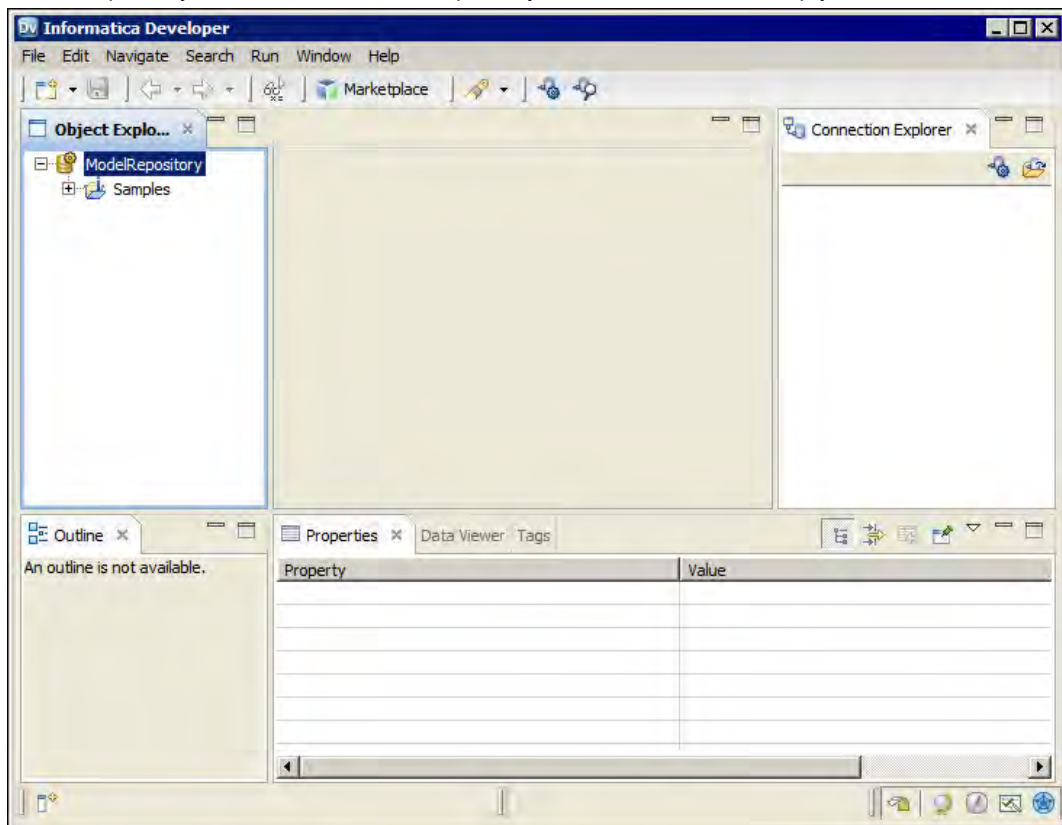
9. Expand the domain and select the Model repository.

The following figure shows the selected Model repository:



10. Click **OK** and then click **Next**.
The **Login** window appears.
11. Enter the user name and password provided to you.
12. Click **Finish**.

The Developer tool connects to the Model repository. The Object Explorer view of the Developer tool shows a Model Repository Service named ModelRepository. The other views are empty.

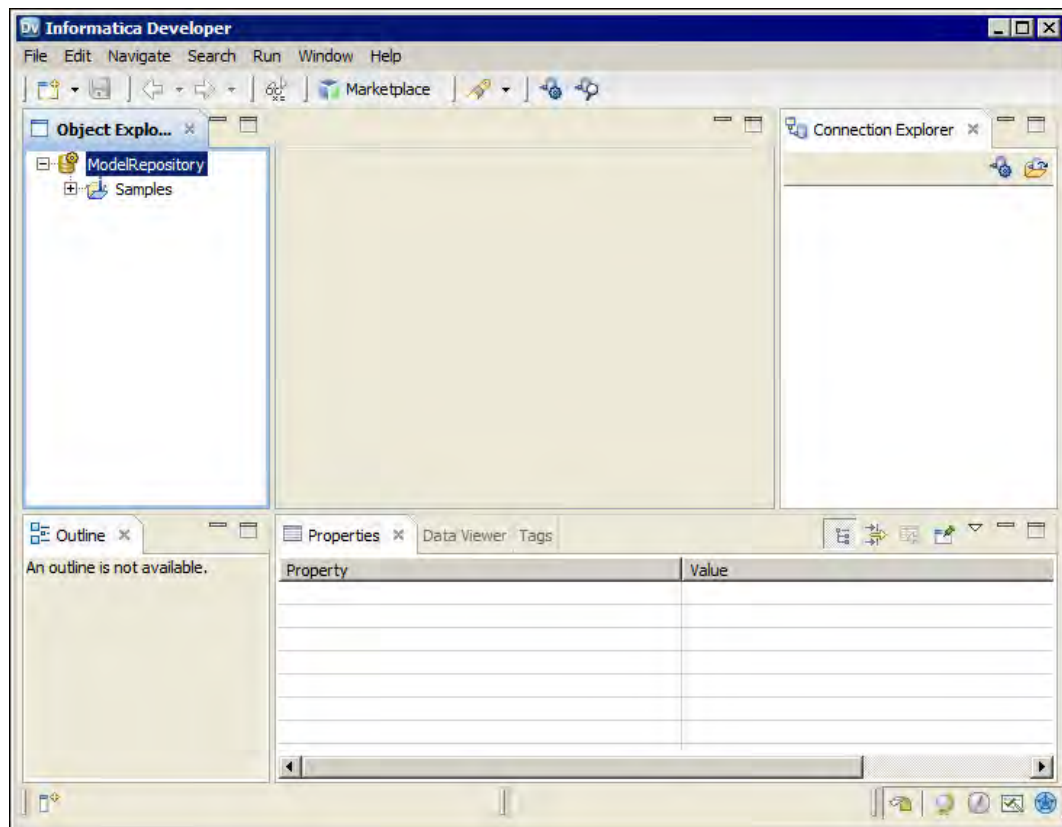


Connecting to the Model Repository

If you installed Informatica services and the Informatica client at the same time on the same machine, the installation process adds the domain and Model repository to the Developer tool. You can connect to the repository from the **Object Explorer** view.

1. In the **Object Explorer** view, right-click the Model repository and select **Connect**.
The **Connect to Domain** dialog box appears.
2. Enter the user name and password provided to you.
3. Click **OK**.

The Developer tool connects to the Model repository. The Model repository includes a Samples project that includes sample objects.



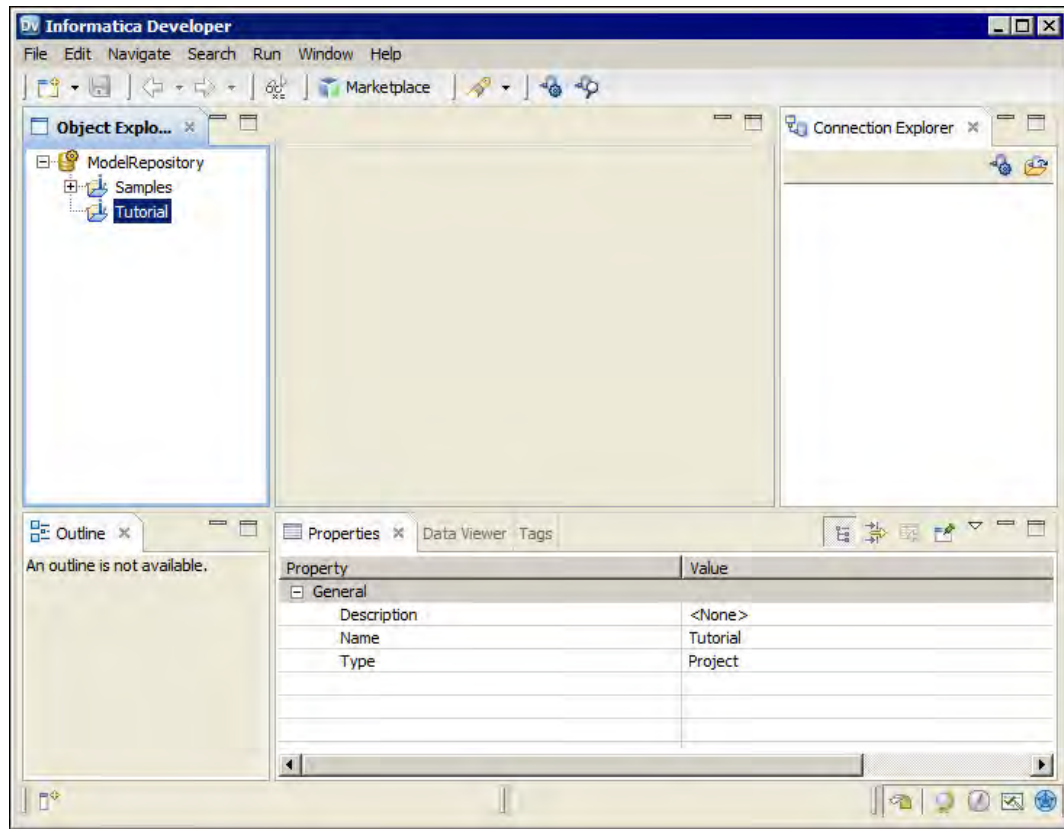
Step 4. Create a Project

Create a project in the Model repository to store objects that you create in the Developer tool. You can create one project for all lessons in this guide.

1. From the Developer tool menu, click **File > New > Project**.
The **New Project** dialog box appears.
2. Enter Tutorial for the project name.

3. Click **Finish**.

The Developer tool adds the Tutorial project under the Model repository in the **Object Explorer** view.



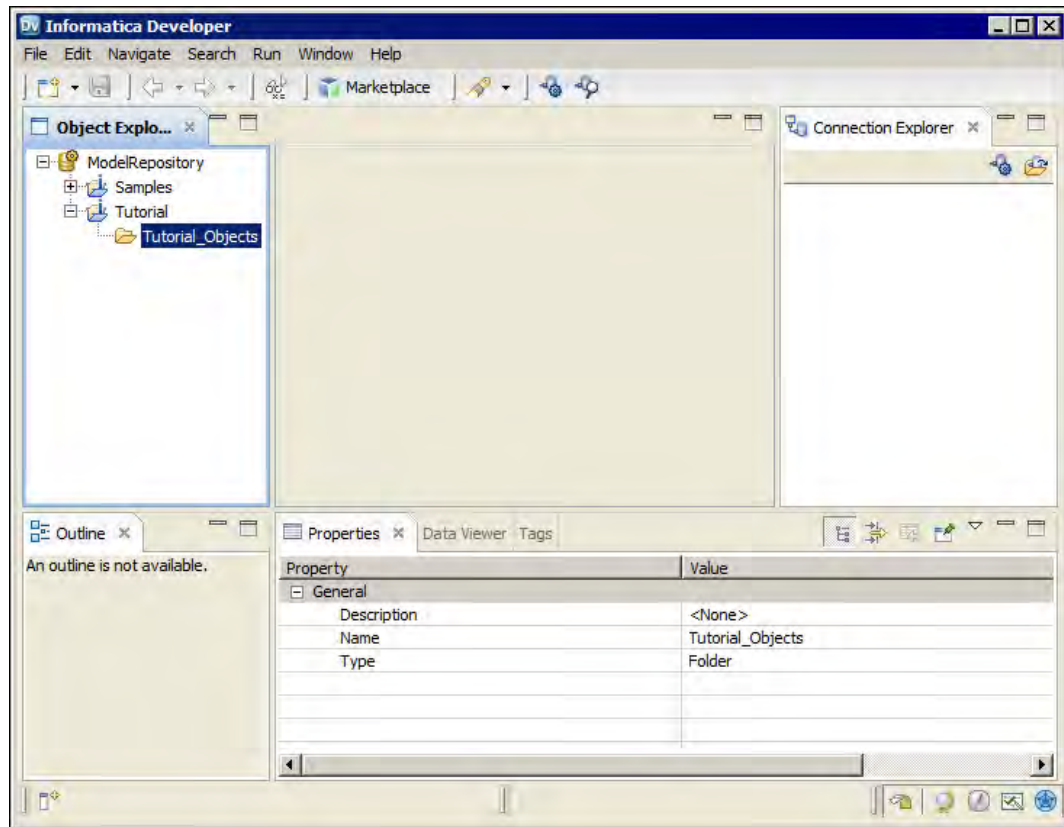
Step 5. Create a Folder

Create a folder to store related objects. You can also store objects in projects. However, folders are useful to organize and store many related objects. You can create one folder for all lessons in this guide.

1. In the **Object Explorer** view, select the Tutorial project.
2. Right-click and select **New > Folder**.
The **New Folder** dialog box appears.
3. Verify that the location indicates the Tutorial project, and enter Tutorial_Objects for the folder name.

4. Click **Finish**.

The Developer tool adds the Tutorial_Objects folder under the Tutorial project in the **Object Explorer** view.



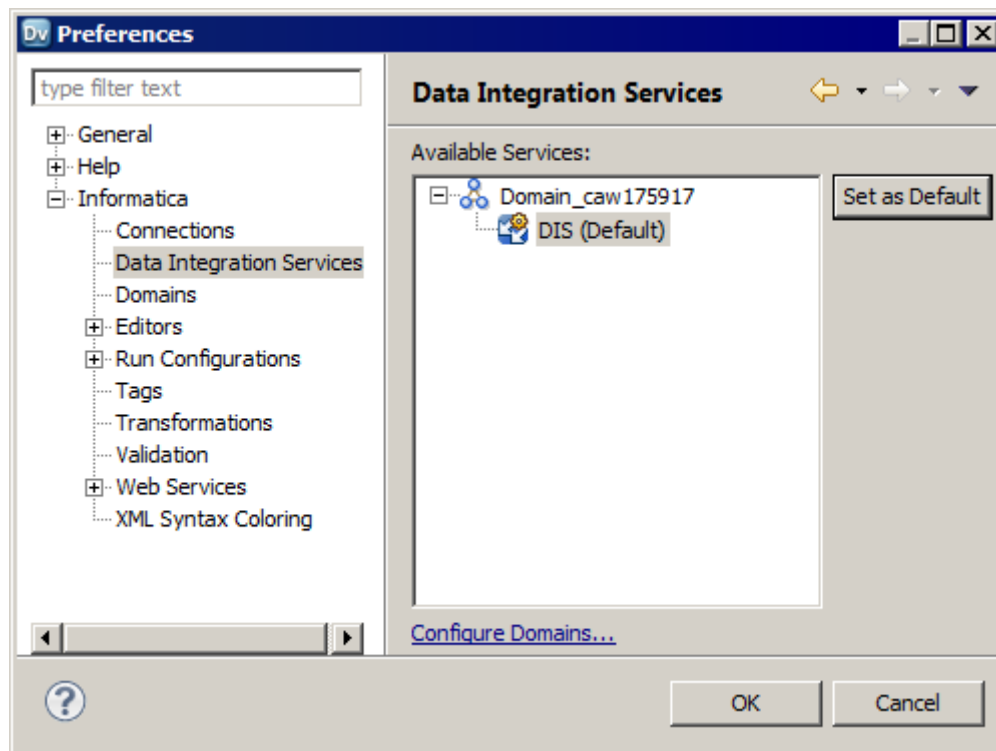
Step 6. Select the Default Data Integration Service

If you installed Informatica services and the Informatica client separately, select the default Data Integration Service that the Developer tool uses to preview data and run mappings.

If you installed Informatica services and the Informatica client at the same time on the same machine, the installation process selects the default Data Integration Service. You can skip this step and continue to the next lesson.

1. From the Developer tool menu, click **Window > Preferences**.
The **Preferences** dialog box appears.
2. Select **Informatica > Data Integration Services**.
3. Expand the domain.
4. Select the Data Integration Service and click **Set as Default**.

The following figure shows the selected default Data Integration Service:



5. Click **OK**.

PowerCenter Express Tips

Use the following tips to help you use the Developer tool more effectively.

Access Informatica Marketplace to search for data integration solutions.

The Informatica Marketplace provides prebuilt solutions to augment, extend, or enhance your data integration implementation.

To access Informatica Marketplace, click the **Marketplace** button on the toolbar. The Developer tool opens a **Marketplace** tab in the editor. You must register as a user before you can log in to the Marketplace for the first time. You might need to close the **Connection Explorer** view to better see the Marketplace content.

Reset the Developer tool perspective to the default values.

If you hide views or move views to another location in the Developer tool workbench, you can reset the Developer tool perspective to the default values. Click **Window > Reset Perspective**.

Use keyboard shortcuts.

You can use keyboard shortcuts to complete tasks in the Developer tool. To view the keyboard shortcuts, click **Help > Key Assist**.

CHAPTER 3

Reviewing Source Data

This chapter includes the following topics:

- ◆ Reviewing Source Data Overview, 20
- ◆ Step 1. View a Mapping Source, 21
- ◆ Step 2. Run a Column Profile, 24
- ◆ PowerCenter Express Tips, 27

Reviewing Source Data Overview

In this lesson, you view data in a sample flat file and run a column profile to evaluate the structure of the source data.

Lesson Concepts

A physical data object is the representation of data that is based on a flat file or relational database. You import physical data objects into the Developer tool so that you can read data from data sources, look up data from data sources, or write data to data sources. You can use a physical data object as a source or a target in a mapping. You can review the source data before you transform the data.

A profile is a set of metadata that describes the content and structure of a data set. Data profiling is often the first step in a project. You can run a profile to evaluate the structure of data and verify that data columns contain the types of information that you expect. If a profile reveals problems in data, you can define processes in your project to fix the problems. For example, if a profile reveals that a column contains values of greater than expected length, you can design data quality processes to remove or fix the problem values.

Lesson Objectives

In this lesson, you complete the following tasks:

- ◆ View data in the SALES_TRANSACTION data source.
- ◆ Run a column profile on the SALES_TRANSACTION data source.

Lesson Prerequisites

Before you start this lesson, complete the first lesson in this tutorial.

Lesson Timing

Set aside 15-20 minutes to complete the tasks in this lesson.

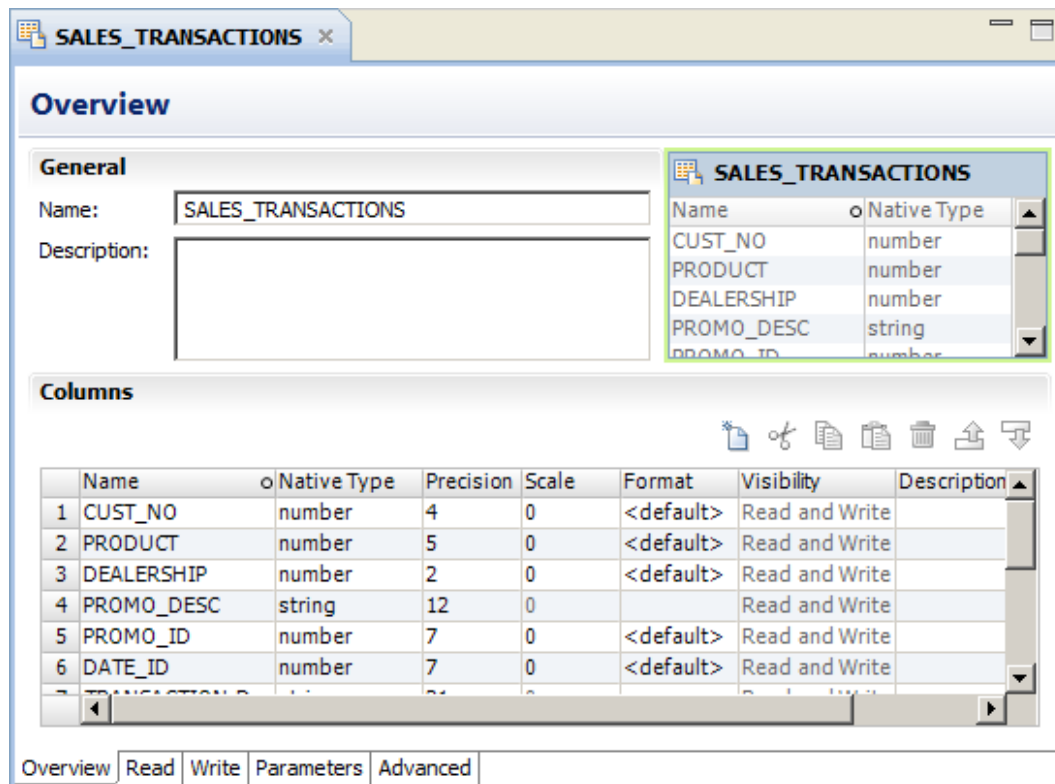
Step 1. View a Mapping Source

View a mapping source to review the data that the source contains.

1. Close the **Marketplace** tab and the **Connection Explorer** view.
2. In the **Object Explorer** view, expand Physical Data Objects in the SharedObjects folder located in the Samples project.
3. Double-click the SALES_TRANSACTIONS flat file data object.

The Developer tool opens the object. The **Overview** view displays the column properties configured for the flat file data object.

The following figure shows the **Overview** view:

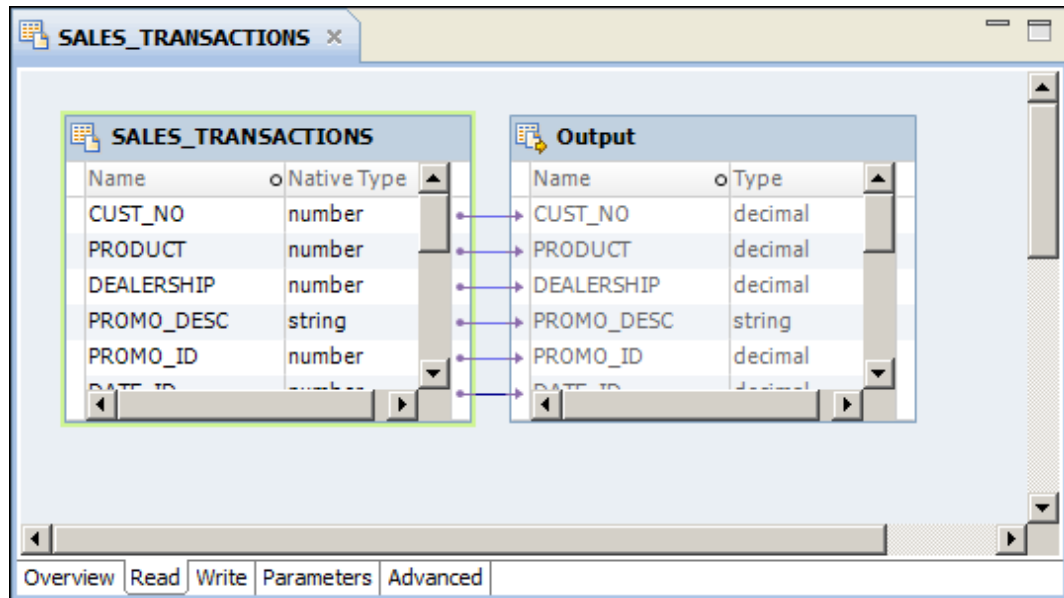


4. Select the SELLING_PRICE column.
5. Change the Native Type to number, the Precision to 8, and the Scale to 2.
6. Click **File > Save** to save the changes to the column.
7. Select the **Read** view.

The Data Integration Service uses the properties configured in the **Read** view when it reads data from the flat file. Select the SALES_TRANSACTIONS source transformation to view general, column, and format properties. The SALES_TRANSACTIONS source transformation displays the native datatypes that are specific to the flat file source.

Select the Output transformation to view run-time properties that the Data Integration Service uses when it reads data from the file. The Output transformation displays the transformation datatypes that the Data Integration Service uses to move data across platforms. When the Data Integration Service reads source data, it converts the native datatypes to the comparable transformation datatypes before transforming the data.

The following figure shows the **Read** view of the SALES_TRANSACTIONS data object:

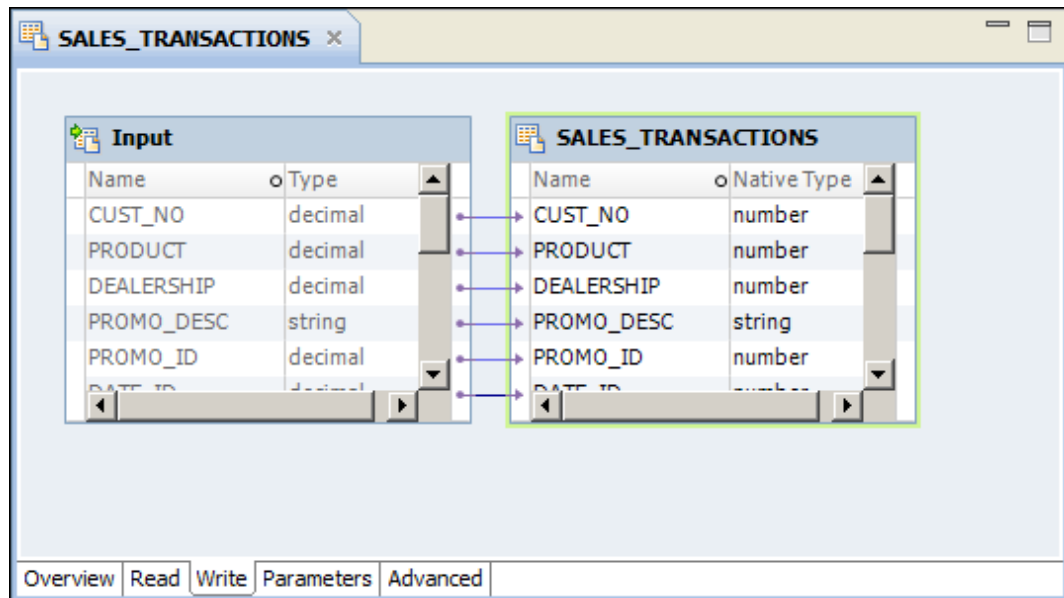


8. Select the **Write** view.

The Data Integration Service uses the properties configured in the **Write** view when it writes data to the flat file. Select the Input transformation to view run-time properties that the Data Integration Service uses when it writes data to the file. The Input transformation displays the transformation datatypes that the Data Integration Service uses to move data across platforms.

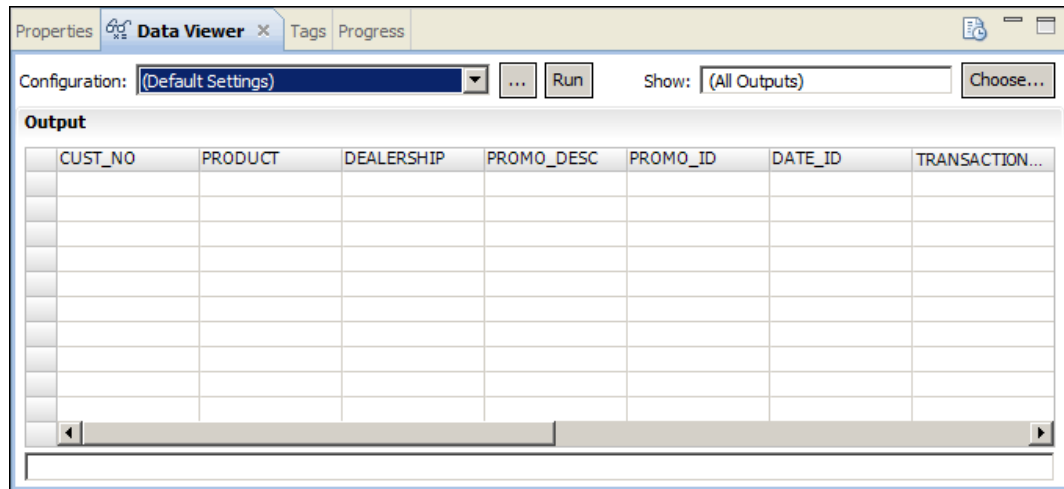
Select the target transformation to view general, column, and format properties. The target transformation displays the native datatypes that are specific to the flat file target.

The following figure shows the **Write** view of the SALES_TRANSACTIONS data object:



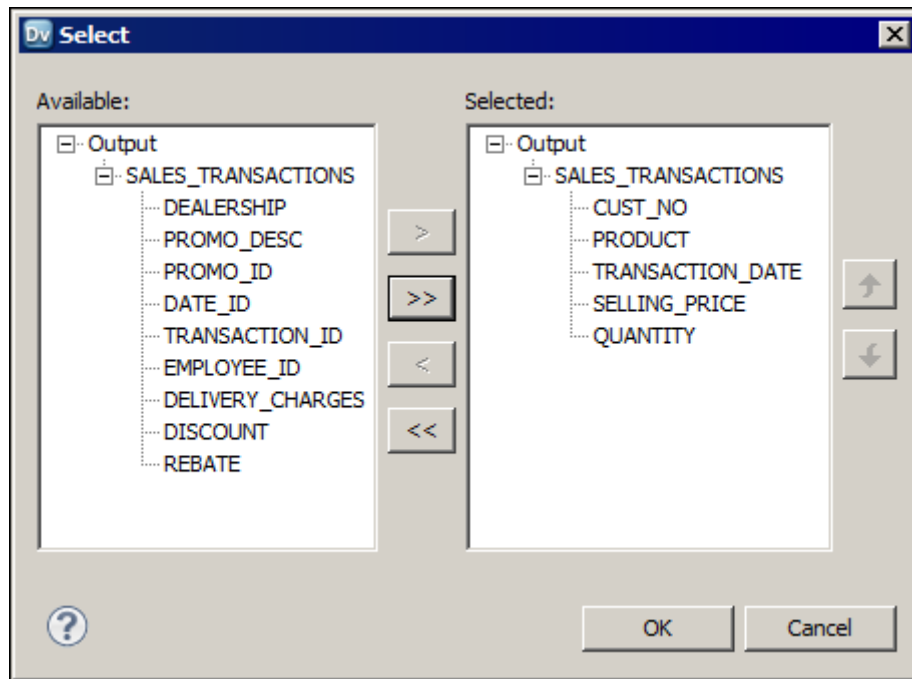
9. Select the **Data Viewer** view.

The following figure shows the **Data Viewer** view:



10. Click **Choose** to filter the data to view.
The **Select** dialog box appears.
11. Move columns to the Available panel so that the following columns display in the Selected panel:
 - ◆ CUST_NO
 - ◆ PRODUCT
 - ◆ TRANSACTION_DATE
 - ◆ SELLING_PRICE
 - ◆ QUANTITY

The following figure shows the selected columns:

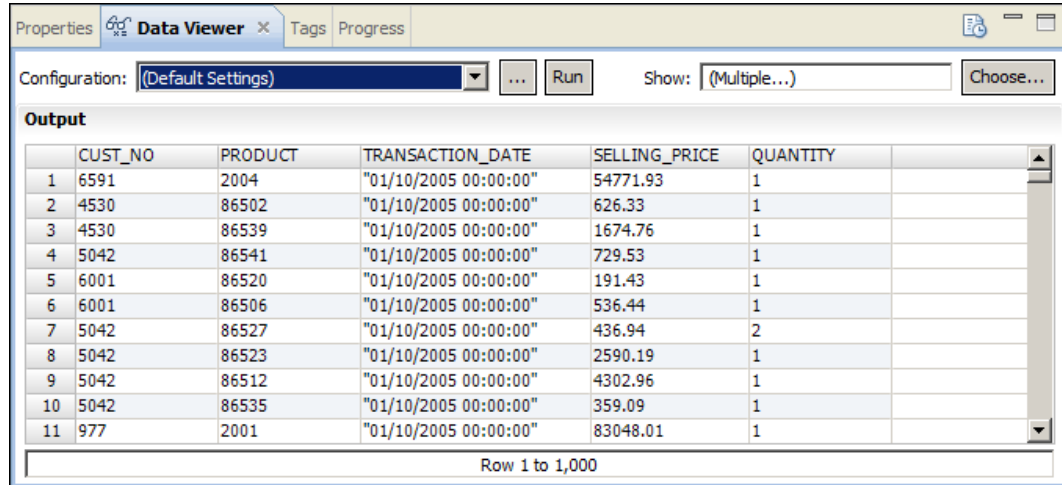


12. Click **OK**.

13. Click **Run** in the **Data Viewer** view.

The **Output** window displays the selected data in the SALES_TRANSACTIONS.dat file.

The following figure shows the previewed data in the **Data Viewer** view:



The screenshot shows the Data Viewer window with the following configuration: Configuration: (Default Settings), Run button, Show: (Multiple...), and Choose... button. The Output window displays a table with the following data:

	CUST_NO	PRODUCT	TRANSACTION_DATE	SELLING_PRICE	QUANTITY	
1	6591	2004	"01/10/2005 00:00:00"	54771.93	1	
2	4530	86502	"01/10/2005 00:00:00"	626.33	1	
3	4530	86539	"01/10/2005 00:00:00"	1674.76	1	
4	5042	86541	"01/10/2005 00:00:00"	729.53	1	
5	6001	86520	"01/10/2005 00:00:00"	191.43	1	
6	6001	86506	"01/10/2005 00:00:00"	536.44	1	
7	5042	86527	"01/10/2005 00:00:00"	436.94	2	
8	5042	86523	"01/10/2005 00:00:00"	2590.19	1	
9	5042	86512	"01/10/2005 00:00:00"	4302.96	1	
10	5042	86535	"01/10/2005 00:00:00"	359.09	1	
11	977	2001	"01/10/2005 00:00:00"	83048.01	1	

Row 1 to 1,000

Step 2. Run a Column Profile

A column profile analyzes the data quality of selected columns. Run a column profile on the SALES_TRANSACTIONS data source to view the patterns of data in the SELLING_PRICE column and the frequencies with which these values occur.

1. In the **Object Explorer** view, select the SALES_TRANSACTIONS flat file data object.
2. Right-click the object and select **Profile**.
The **New** window opens.
3. Select **Profile** and click **Next**.
The **New Profile** window opens.
4. Click **Browse**, select the Tutorial_Objects folder, and click **OK**.

The **New Profile** window shows the Tutorial_Objects folder in the Tutorial project as the location.

New Profile

Single Data Object Profile
Configure general properties.

Name: Profile_SALES_TRANSACTIONS

Location: ModelRepository\Tutorial\Tutorial_Objects **Browse...**

Description:

Data Objects:

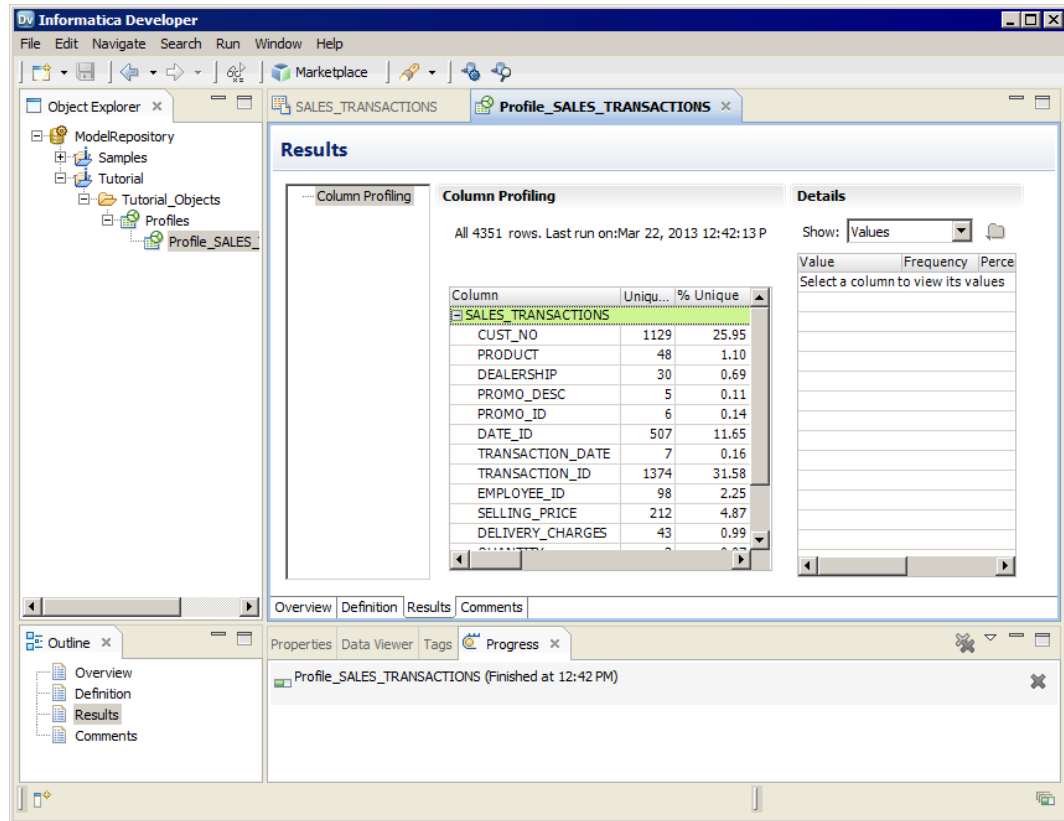
Samples\SharedObjects\SALES_TRANSACTIONS **Add...**
Remove

Run Profile on finish.

< Back **Next >** **Finish** **Cancel**

5. Click **Finish**.
The Profile_SALES_TRANSACTIONS profile opens and the profile runs.
6. Click **Window > Show View > Progress** to view the progress of the Profile_SALES_TRANSACTIONS profile.
The **Progress** view opens.
7. When the **Progress** view reports that the profile finishes running, select the **Results** view.

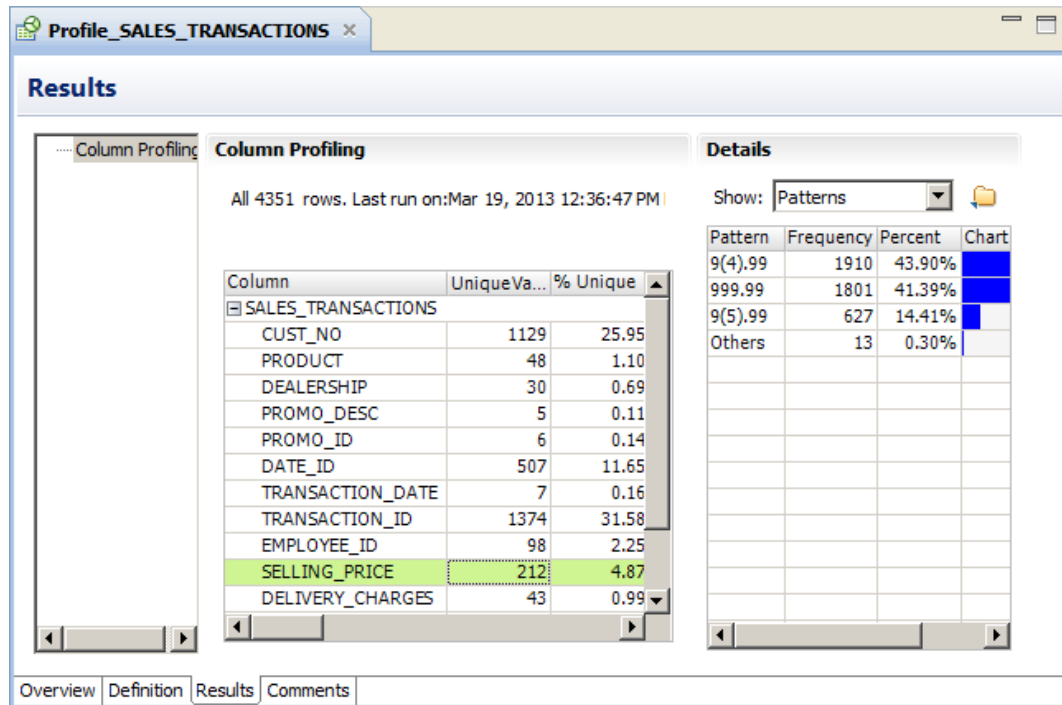
The following figure shows the **Results** view of the profile and the **Progress** view that displays below the open profile:



8. Under the **Column Profiling** section, select the **SELLING_PRICE** column.
9. Under the **Details** section, click the **Show** list and select **Patterns**.

The **Details** section shows the patterns found in the **SELLING_PRICE** column. The string 9(4).99 in the **Pattern** column refers to records that contain four-figure selling prices. The string 999.99 refers to records that contain three-figure selling prices. The string 9(5).99 refers to records that contain five-figure selling prices.

The following figure shows the patterns found in the SELLING_PRICE column:



10. In the **Pattern** column, double-click the string 9(4).99.

The **Data Viewer** view runs and displays records where the SELLING_PRICE column contains a four-figure price.

PowerCenter Express Tips

Use the following tip when you review source data.

Export data.

You can export the data that displays in the **Data Viewer** view to a tab-delimited flat file, such as a TXT or CSV file. Export data when you want to create a local copy of the data. To export the data, right-click a row of data in the **Data Viewer** view and select **Export Data**.

You can also export column values and column pattern data from profile results to a tab-delimited flat file, such as a TXT or CSV file. To export column values and column pattern data from a profile, click the **Export Patterns to**

File button () in the **Details** section of the **Results** view.

CHAPTER 4

Developing a Mapping

This chapter includes the following topics:

- ◆ Developing a Mapping Overview, 28
- ◆ Step 1. Create a Mapping, 29
- ◆ Step 2. Add a Transformation to the Mapping, 29
- ◆ Step 3. Add a Target to the Mapping, 33
- ◆ Step 4. Run the Mapping and Review Results, 36
- ◆ PowerCenter Express Tips, 38

Developing a Mapping Overview

In this lesson, you develop a mapping that aggregates data in a source flat file and then writes the results to a target flat file. You run the mapping and review the results.

Lesson Concepts

A mapping is a set of inputs and outputs that represent the data flow between sources and targets. You link the sources and targets with transformation objects that define the rules for data transformation. Use an Aggregator transformation in a mapping to perform aggregate calculations, such as averages and sums.

After you validate and save a mapping, you can run the mapping in the Developer tool. When you run a mapping, the Data Integration Service uses the instructions configured in the mapping to read, transform, and write data.

You can also add a mapping to an application and deploy the application to the Data Integration Service. Deploy a mapping to the Data Integration Service so that you can run mappings from the command line.

Lesson Objectives

In this lesson, you complete the following tasks:

- ◆ Create a mapping that uses the SALES_TRANSACTION flat file data object as a source.
- ◆ Add an Aggregator transformation to the mapping that calculates the total and average sales by dealership.
- ◆ Add a flat file data object to the mapping as a target so that the mapping writes the total and average sales to a flat file.
- ◆ Run the mapping and review the results in the target file.

Lesson Prerequisites

Before you start this lesson, complete the earlier lessons in this tutorial.

Lesson Timing

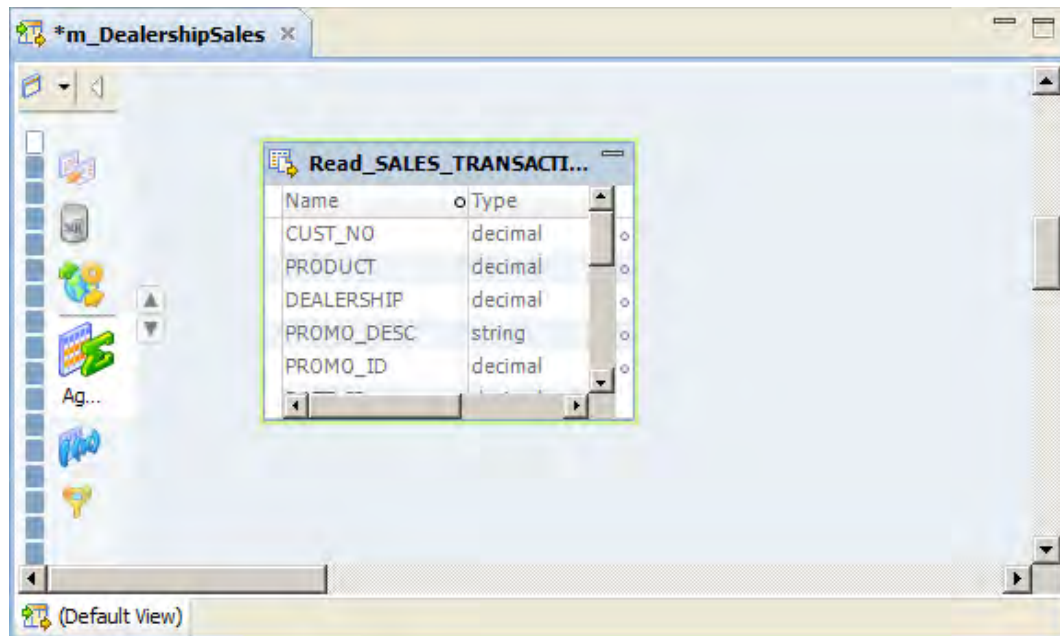
Set aside 25-30 minutes to complete the tasks in this lesson.

Step 1. Create a Mapping

Create a mapping and add the SALES_TRANSACTIONS flat file data object as a source.

1. In the **Object Explorer** view, select the Tutorial_Objects folder.
2. Right-click and select **New > Mapping**.
3. In the **Mapping** dialog box, enter m_DealershipSales for the mapping name and click **Finish**.
The empty mapping opens in the editor.
4. In the **Object Explorer** view, expand Physical Data Objects in the SharedObjects folder located in the Samples project.
5. Select the SALES_TRANSACTIONS flat file data object and drag it to the editor.
The **Add to Mapping** dialog box opens.
6. Select **Read** and click **OK**

The Read_SALES_TRANSACTIONS data object appears in the editor.

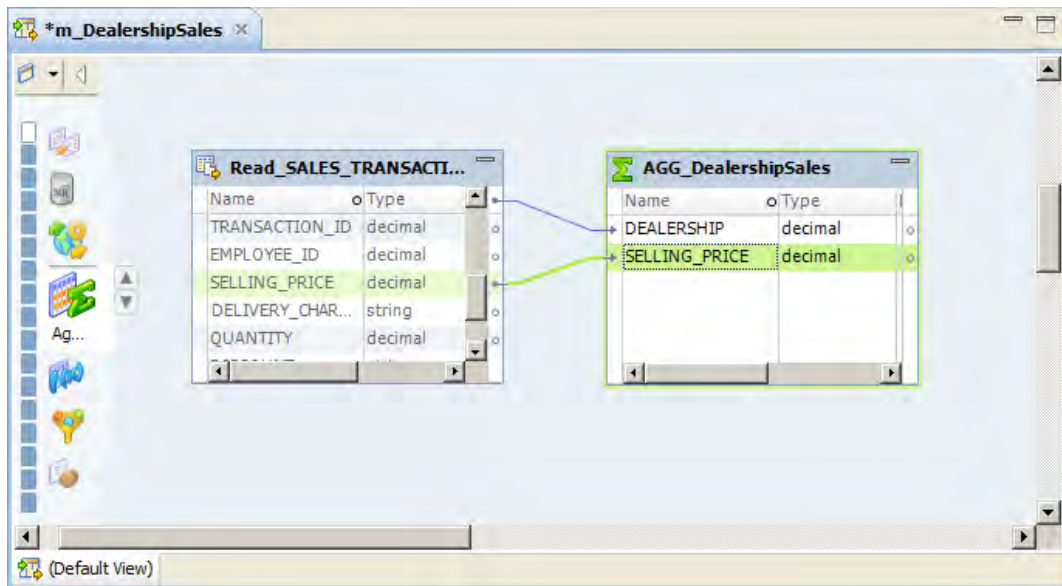


Step 2. Add a Transformation to the Mapping

Add an Aggregator transformation to the mapping to calculate the total and average sales by dealership.

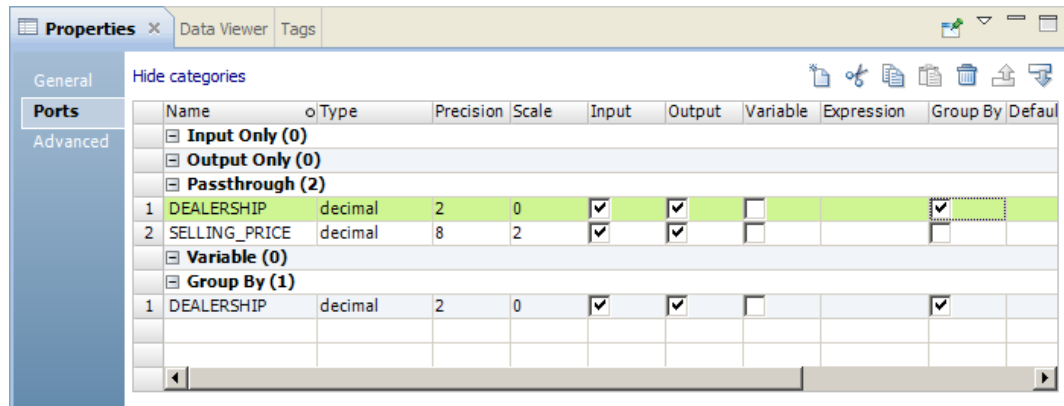
1. Right-click in the editor, and select **Add Transformation**.


2. Select the Aggregator transformation and click **OK**.
An empty Aggregator transformation appears in the editor.
3. Select the Aggregator transformation in the editor.
4. In the **Properties** view, click the **General** tab.
5. Enter AGG_DealershipSales for the transformation name.
6. In the editor, select the DEALERSHIP port from the Read_SALES_TRANSACTIONS data object and drag the port to the Aggregator transformation.
The Developer tool copies the DEALERSHIP port to the Aggregator transformation and links the ports.
7. Select the SELLING_PRICE port from the Read_SALES_TRANSACTIONS data object and drag the port to the Aggregator transformation.
The Developer tool copies the SELLING_PRICE port to the Aggregator transformation and links the ports.
The following figure shows the two ports linked to the Aggregator transformation:



8. In the **Properties** view of the Aggregator transformation, click the **Ports** tab.
9. For the DEALERSHIP port, select the checkbox in the **Group By** column.
When you group values, the Data Integration Service generates one row for each group.

The following figure shows the selected group by port:

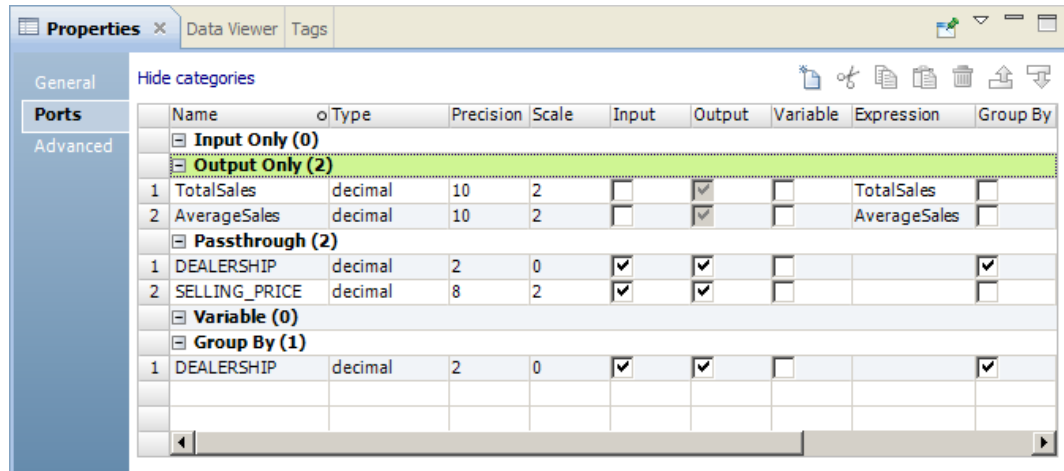



10. Select **Output Only** and click the **New** button () two times to create two output ports.
The Developer tool adds two ports under **Output Only** with default names of Field and Field1.
11. Select each output port and change the name, type, precision, and scale.

The following table describes the values to enter:

Port Name	Type	Precision	Scale
TotalSales	decimal	10	2
AverageSales	decimal	10	2

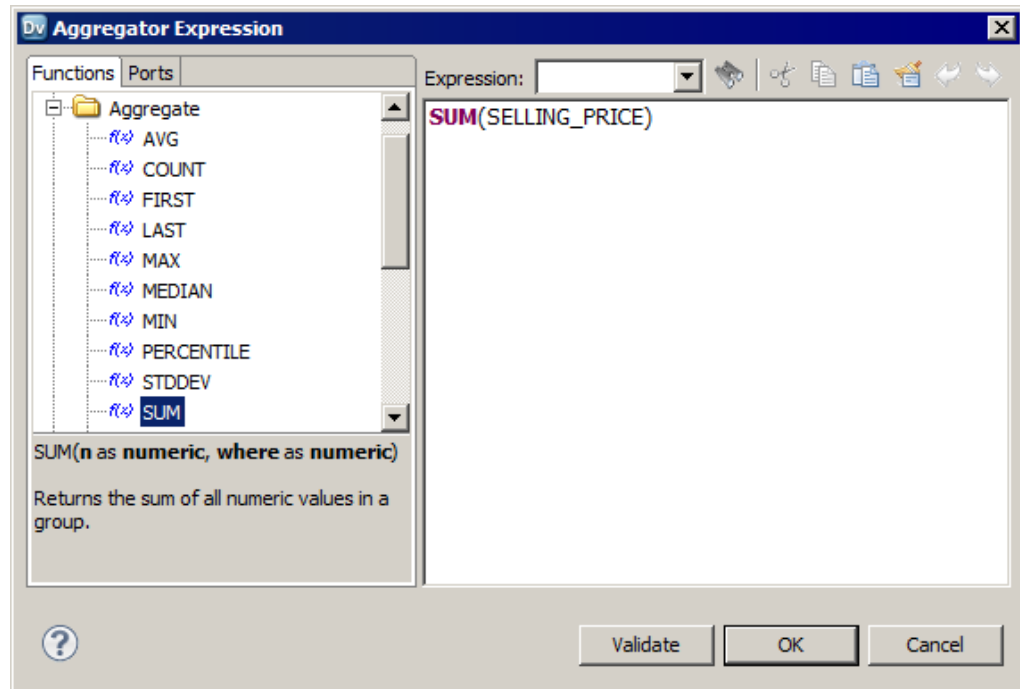
The following figure shows the completed output ports:




12. Select the TotalSales output port.
13. In the **Expression** column, click the **Open** button () to open the **Expression Editor**.
14. Use the point-and-click method in the **Expression Editor** to enter the expression.
 - a. Delete the TotalSales text in the **Expression** panel.
 - b. In the **Functions** tab, expand the Aggregate folder.

- c. Double-click the SUM function.
- d. Place the cursor inside the empty parentheses in the expression.
- e. Click the **Ports** tab.
- f. Double-click the SELLING_PRICE column.

The **Expression Editor** updates the expression to SUM(SELLING_PRICE).

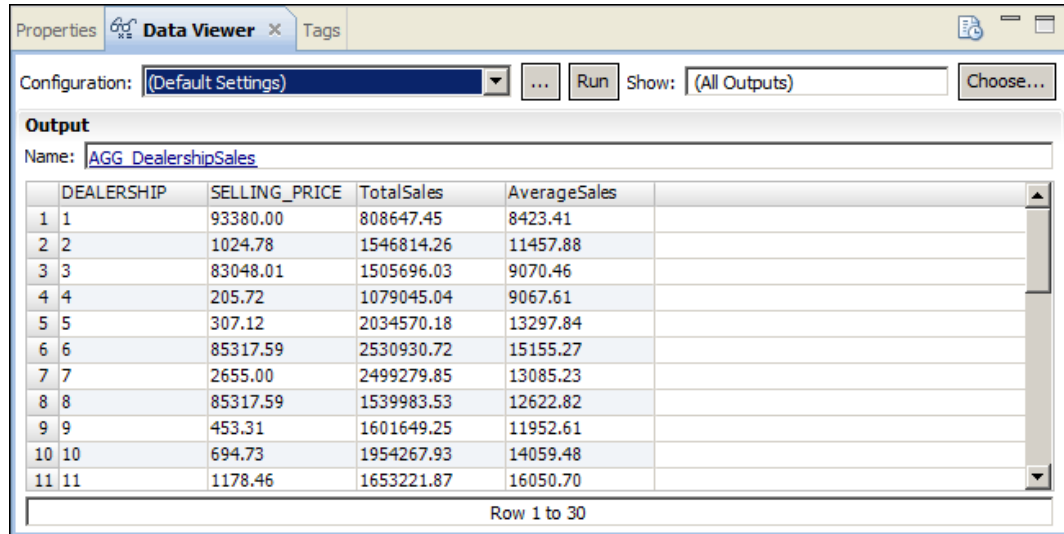


- g. Click **Validate** to validate the expression.
 - h. Click **OK** to close the **Expression Editor**.
15. Select the AverageSales output port.
 16. In the **Expression** column, click the **Open** button () to open the **Expression Editor**.
 17. Use the point-and-click method to enter the following expression:

```
AVG (SELLING_PRICE)
```
 18. Click **Validate** to validate the expression.
 19. Click **OK** to close the **Expression Editor**.

20. Preview the Aggregator transformation data to make sure that it contains the expected results. In the editor, right-click the Aggregator transformation and select **Run Data Viewer**.

The data calculated by the transformation appears in the **Data Viewer** view.



The screenshot shows the Data Viewer window with the following configuration and output:

Configuration: (Default Settings) Run Show: (All Outputs) Choose...

Output Name: AGG_DealershipSales

	DEALERSHIP	SELLING_PRICE	TotalSales	AverageSales
1	1	93380.00	808647.45	8423.41
2	2	1024.78	1546814.26	11457.88
3	3	83048.01	1505696.03	9070.46
4	4	205.72	1079045.04	9067.61
5	5	307.12	2034570.18	13297.84
6	6	85317.59	2530930.72	15155.27
7	7	2655.00	2499279.85	13085.23
8	8	85317.59	1539983.53	12622.82
9	9	453.31	1601649.25	11952.61
10	10	694.73	1954267.93	14059.48
11	11	1178.46	1653221.87	16050.70

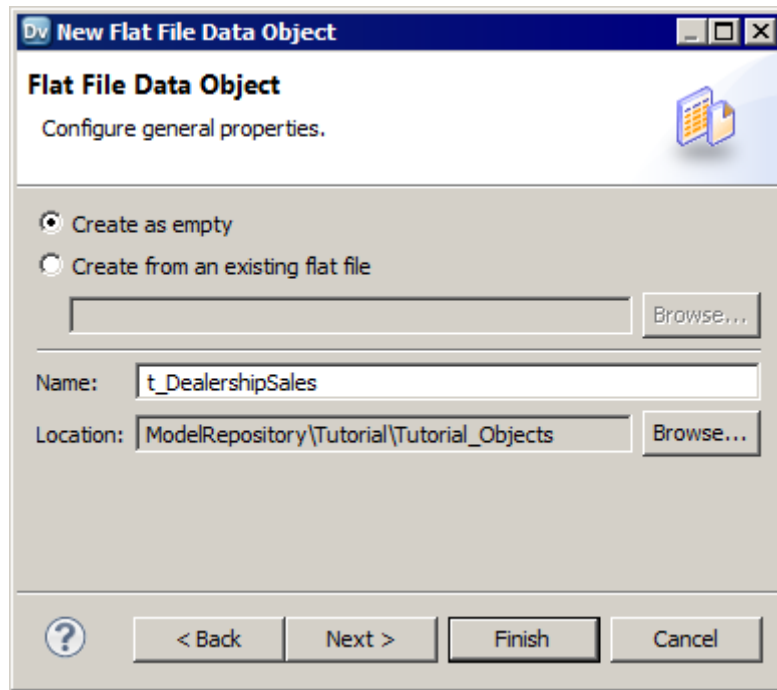
Row 1 to 30

Step 3. Add a Target to the Mapping

Create a flat file data object and add columns named DealerID, TotalSales, and AverageSales. Add the flat file data object to the mapping as a target so that the mapping writes the total and average sales to a flat file.


1. In the **Object Explorer** view, select the Tutorial_Objects folder.
2. Right-click and select **New > Data Object**.
3. Select **Flat File Data Object** and click **Next**.
4. In the **New Flat File Data Object** dialog box, select **Create as empty** and enter t_DealershipSales for the data object name.

The following figure shows the completed **New Flat File Data Object** dialog box:



5. Click **Finish**.

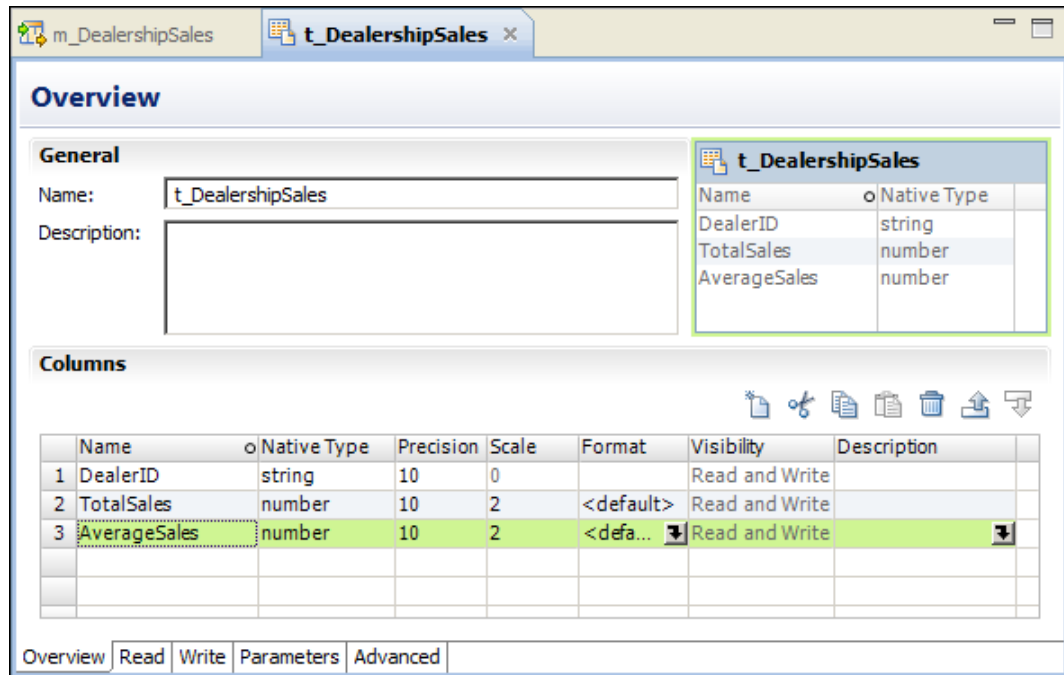
The `t_DealershipSales` data object opens.

6. In the **Columns** section of the **Overview** view, click the **New** button () three times to create three columns. The Developer tool adds three columns with default names of `Field`, `Field1`, and `Field2`.
7. Select each column and change the name, native type, precision, and scale.

The following table describes the values to enter:

Column Name	Native Type	Precision	Scale
DealerID	string	10	0
TotalSales	number	10	2
AverageSales	number	10	2

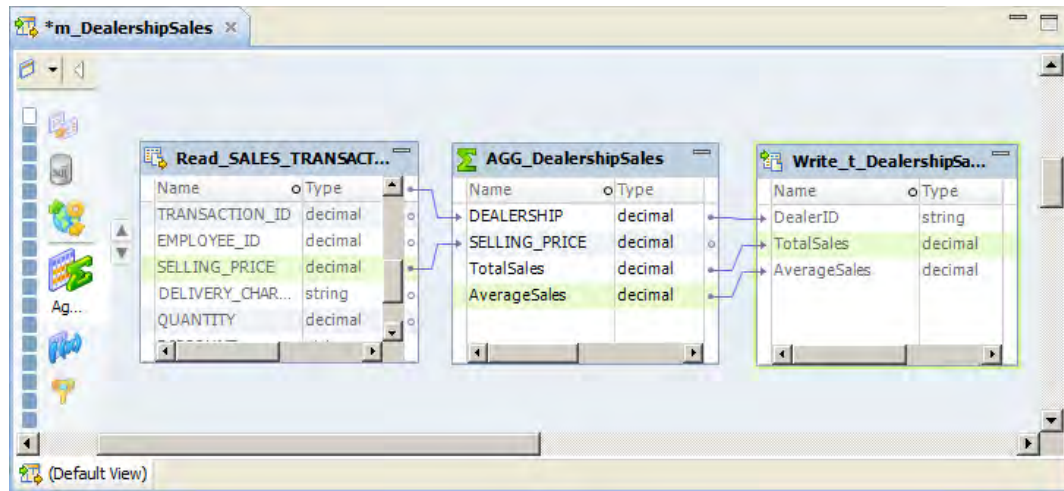
The following figure shows the completed columns:



8. Click **File > Save** to save the data object.
9. Close the t_DealershipSales data object.
10. In the **Object Explorer** view, select the t_DealershipSales data object and drag it to the editor of the m_DealershipSales mapping.
The **Add to Mapping** dialog box opens.
11. Select **Write** and click **OK**.
The Write_t_DealershipSales data object appears in the editor.
12. Drag ports from the Aggregator transformation to the flat file data object to link the ports.
The following table lists the ports to link:

Transformation Port	Write Data Object Port
DEALERSHIP	DealerID
TotalSales	TotalSales
AverageSales	AverageSales

The following figure shows the linked ports:



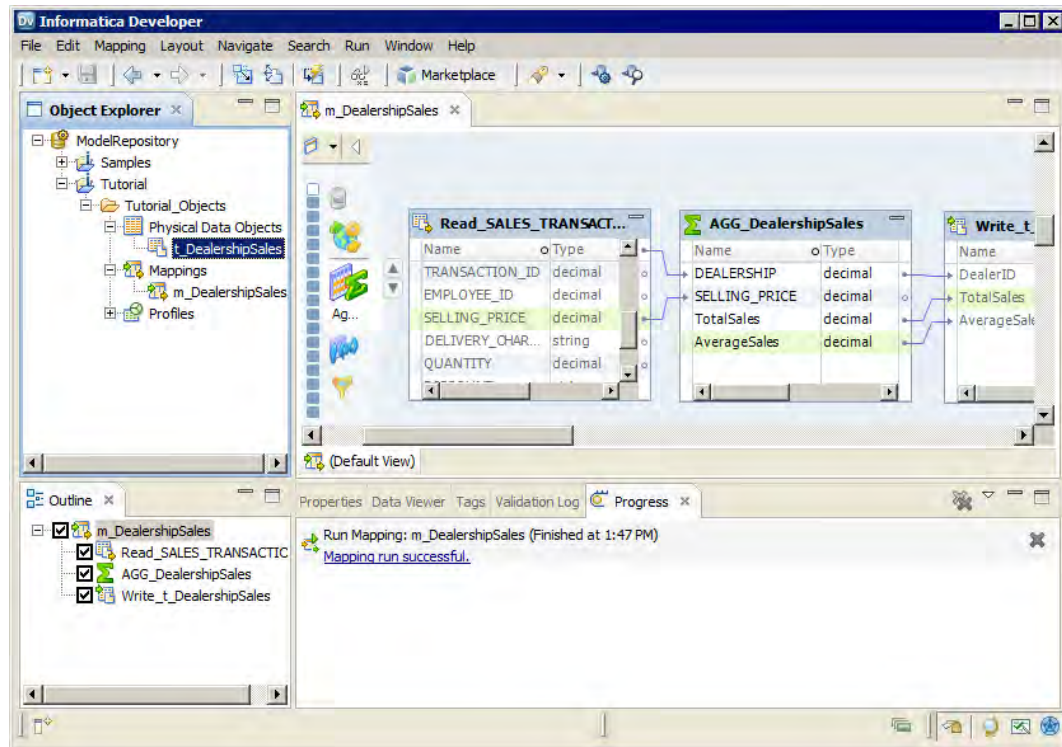
13. Right-click in the editor, and select **Validate**.
If errors appear in the **Validation Log** view, fix the errors and validate the mapping again.
14. When the mapping is valid, click **File > Save** to save the mapping.

Step 4. Run the Mapping and Review Results

Run the mapping to read, transform, and write the data to the target. View the mapping log events and review the results in the target file.

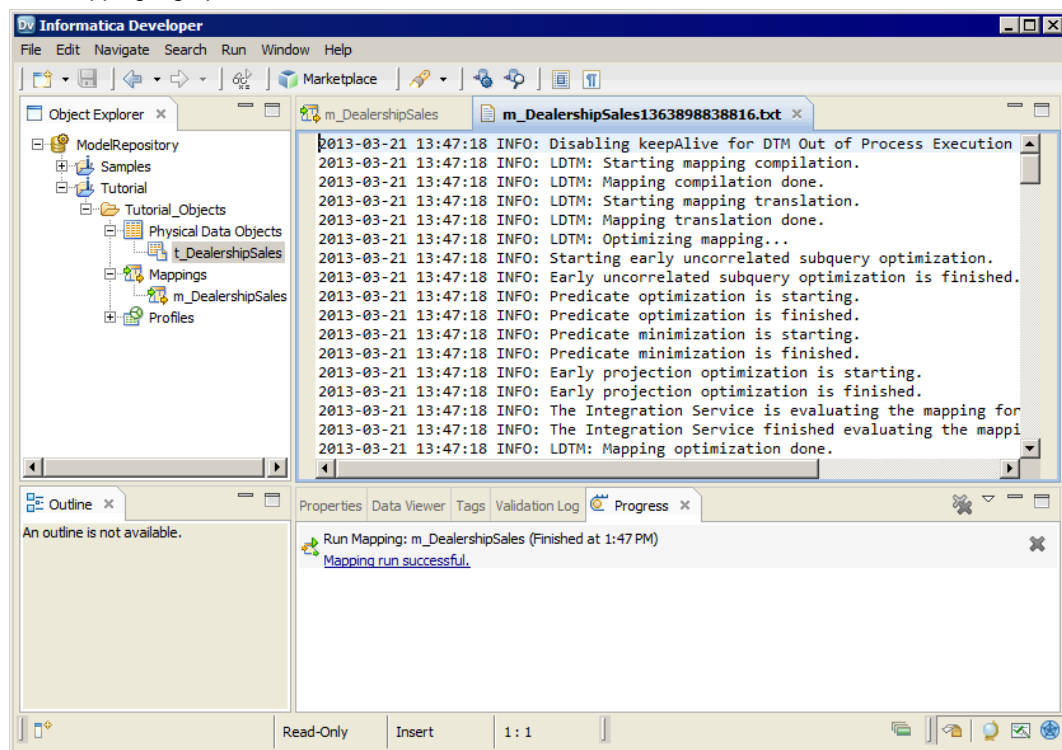
1. Right-click the editor of the m_DealershipSales mapping, and select **Run Mapping**.
The **Run Mapping** dialog box appears and displays the progress of the mapping run.
2. Click **Window > Show View > Progress** to view the progress of the mapping run.

The **Progress** view opens.



3. Click the link for the mapping run and select **Go to Log** to view the log events.

The mapping log opens.



4. To save the log to a file, click **File > Save a Copy As** and choose a directory.

5. To view the results written to the target file, navigate to the following target directory on the machine where Informatica services is installed:

```
<PowerCenter Express Installation Directory>\tomcat\bin\target
```

6. Open the t_DealershipSales.out file with a text editor.

Each line of the file lists a dealership ID, the total sales for the dealership, and the average sales for the dealership. For example, the first five lines of the file contain the following data:

```
1,808648.00,8423.42
2,1546814.00,11457.88
3,1505696.00,9070.46
4,1079045.00,9067.61
5,2034576.00,13297.88
```

PowerCenter Express Tips

Use the following tips when you develop mappings.

Align mapping objects as icons.

You can convert the mapping objects to icons and align the icons in the editor. To align mapping objects as icons, click **Layout > Arrange All Iconic**.

Navigate mapping objects with a quick outline.

A quick outline displays objects that are dependent on a mapping object selected in the editor. You can use the quick outline to sort dependent objects by name or by type, or to search for dependent objects. Select an object in the quick outline to navigate to the object in the editor.

To display the quick outline, select the mapping or an object in the mapping and then click **Navigate > Quick Outline**.

CHAPTER 5

Running a Workflow

This chapter includes the following topics:

- ◆ Running a Workflow Overview, 39
- ◆ Step 1. Review the Mapping, 40
- ◆ Step 2. Create a Workflow, 42
- ◆ Step 3. Deploy and Run the Workflow, 44
- ◆ PowerCenter Express Tips, 45

Running a Workflow Overview

In this lesson, you review a sample mapping that joins data from two sources. You add the mapping to a workflow. You deploy the workflow to the Data Integration Service and then run an instance of the workflow.

Lesson Concepts

A workflow is a graphical representation of a set of events, tasks, and decisions that define a business process. You add a Mapping task to a workflow to run a mapping from the workflow. You connect workflow objects with sequence flows to specify the order that the Data Integration Service runs the objects.

After you validate and save a workflow, you add the workflow to an application and deploy the application to the Data Integration Service. You run a workflow instance from the deployed application. The Data Integration Service uses the instructions configured in the workflow to run the events and tasks and to evaluate the decisions.

Lesson Objectives

In this lesson, you complete the following tasks:

- ◆ Review a sample mapping that joins data from two flat file sources.
- ◆ Create a workflow and add a Mapping task that runs the sample mapping.
- ◆ Deploy the workflow to the Data Integration Service and then run an instance of the workflow.

Lesson Prerequisites

Before you start this lesson, complete the earlier lessons in this tutorial.

Lesson Timing

Set aside 10-15 minutes to complete the tasks in this lesson.

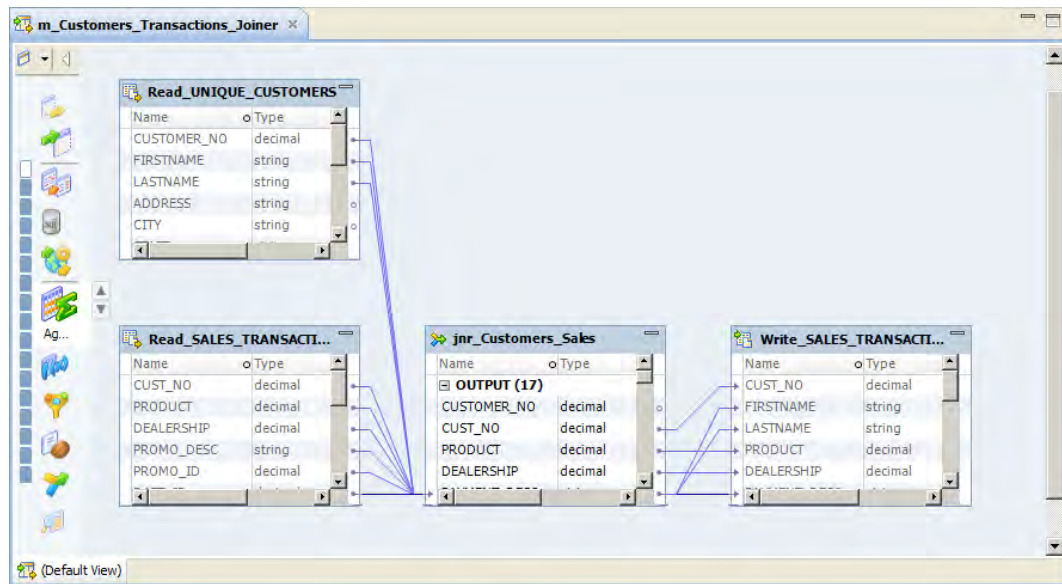
Step 1. Review the Mapping

Open and review a sample mapping that uses a Joiner transformation to join data from two flat file sources.

1. In the **Object Explorer** view, expand Mappings in the TransformationSamples folder located in the Samples project.
2. Double-click the m_Customers_Transactions_Joiner mapping.

The Developer tool opens the mapping in the editor. The mapping contains two flat file sources. The Read_UNIQUE_CUSTOMERS source contains information about each customer. The Read_SALES_TRANSACTIONS source contains information about each sales transaction, including a CUST_NO column that identifies the customer that made the transaction.

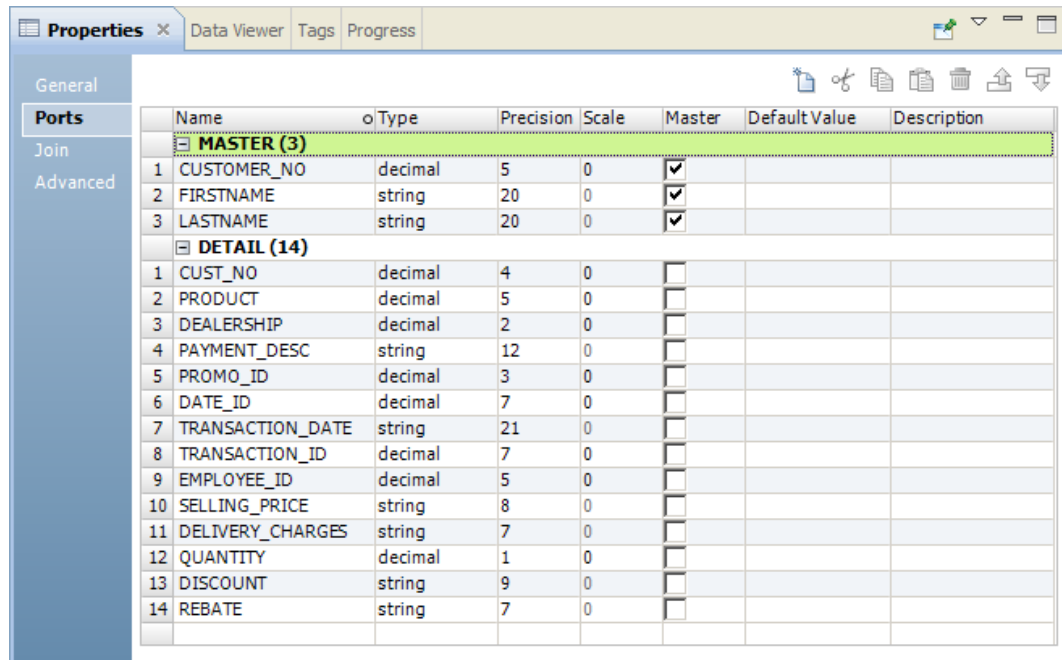
The following figure shows the m_Customers_Transactions_Joiner mapping open in the editor:



3. Select the Joiner transformation in the editor.
4. In the **Properties** view, click the **Ports** tab.

The ports include three master columns from the master source Read_UNIQUE_CUSTOMERS. The ports also include all columns from the detail source Read_SALES_TRANSACTIONS as detail ports.

The following figure shows the **Ports** tab in the **Properties** view for the Joiner transformation:



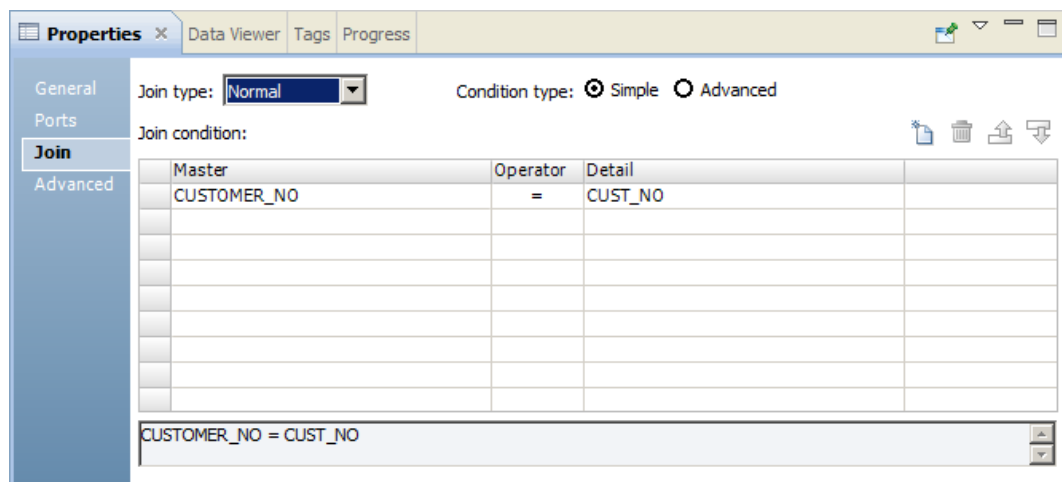
- Click the **Join** tab.

The transformation uses a normal join and ignores all rows of master and detail source data that do not match the join condition.

- Review the join condition.

The transformation uses the following join condition: `CUSTOMER_NO=CUST_NO`. The transformation uses `CUSTOMER_NO` as the matching column to join the customer first name and last name in the master source `Read_UNIQUE_CUSTOMERS` to the sales data in the detail source `Read_SALES_TRANSACTIONS`.

The following figure shows the **Join** tab in the **Properties** view for the Joiner transformation:

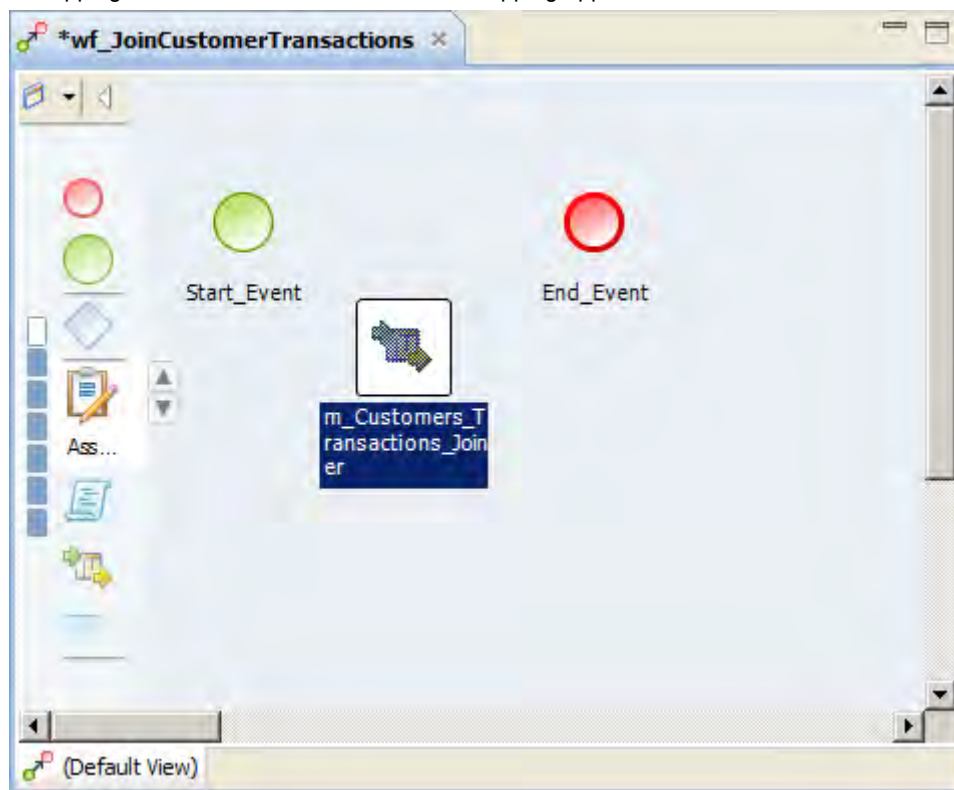


- Close the mapping in the editor.

Step 2. Create a Workflow

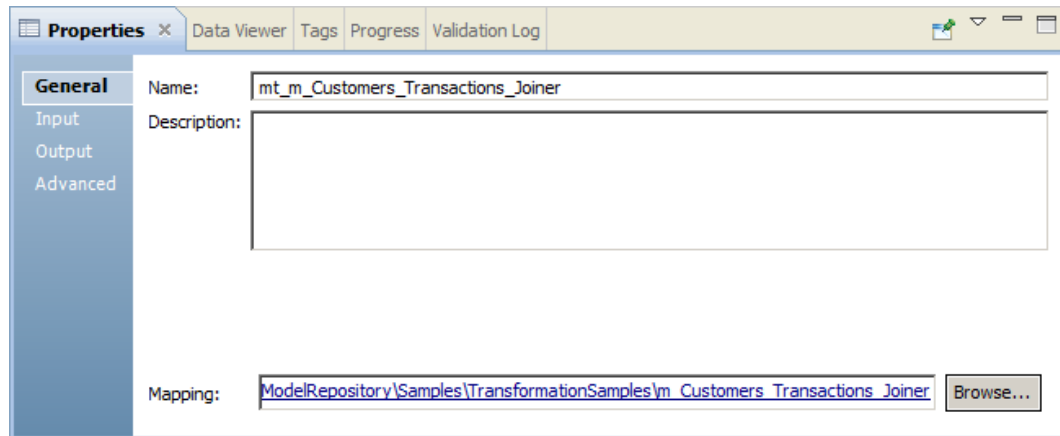
Create a workflow and add a Mapping task that runs the m_Customers_Transactions_Joiner sample mapping.

1. In the **Object Explorer** view, select the Tutorial_Objects folder.
2. Right-click and select **New > Workflow**.
3. In the **Workflow** dialog box, enter wf_JoinCustomerTransactions for the workflow name and click **Finish**.
A workflow with a Start event and an End event opens in the editor.
4. In the **Object Explorer** view, expand Mappings in the TransformationSamples folder located in the Samples project.
5. Select the m_Customers_Transactions_Joiner mapping and drag it to the editor.
A Mapping task with the same name as the mapping appears in the editor.



6. Select the Mapping task in the editor and click the **General** tab in the **Properties** view.
7. Enter mt_m_Customers_Transactions_Joiner for the Mapping task name.
8. Verify that the m_Customers_Transactions_Joiner mapping is selected.

The following figure shows the **General** tab in the **Properties** view for the Mapping task:

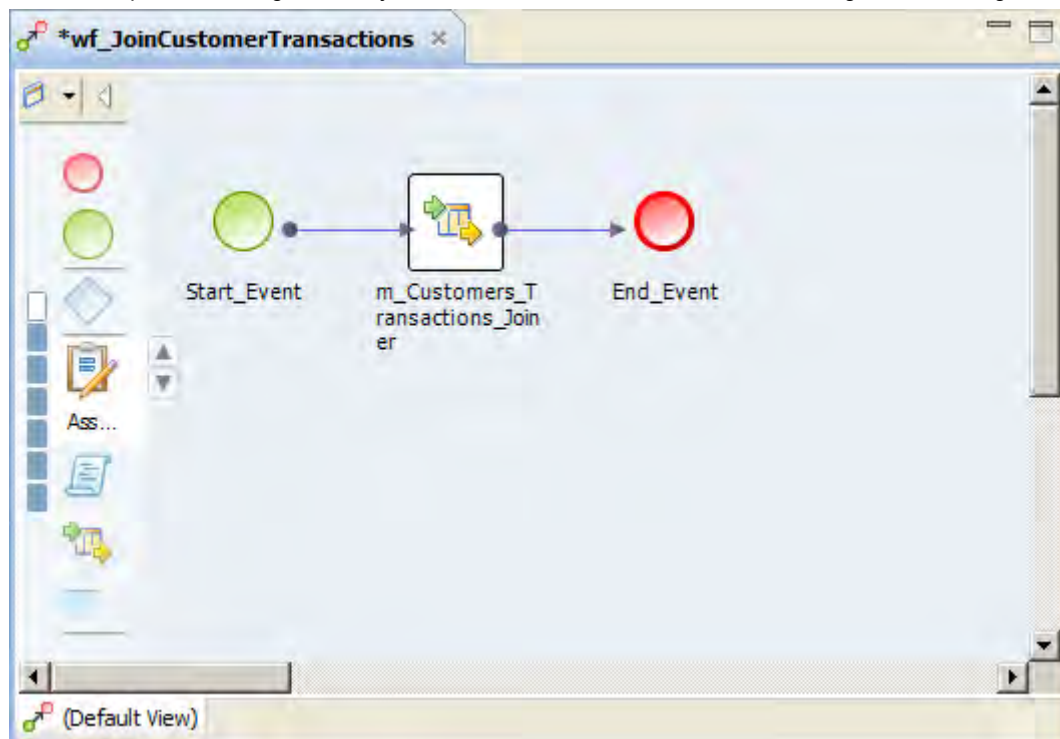


9. In the editor, hold the cursor over the Start event to display a yellow circle. Drag the yellow circle to the Mapping task.

A sequence flow appears between the objects. Sequence flows connect workflow objects and determine the order that the Data Integration Service runs the objects in the workflow.

10. Hold the cursor over the Mapping task to display a yellow circle. Drag the yellow circle to the End event.
11. Right-click in the editor, and select **Align All to Grid**.

The Developer tool arranges the objects in the editor based on data flow and aligns them to a grid.



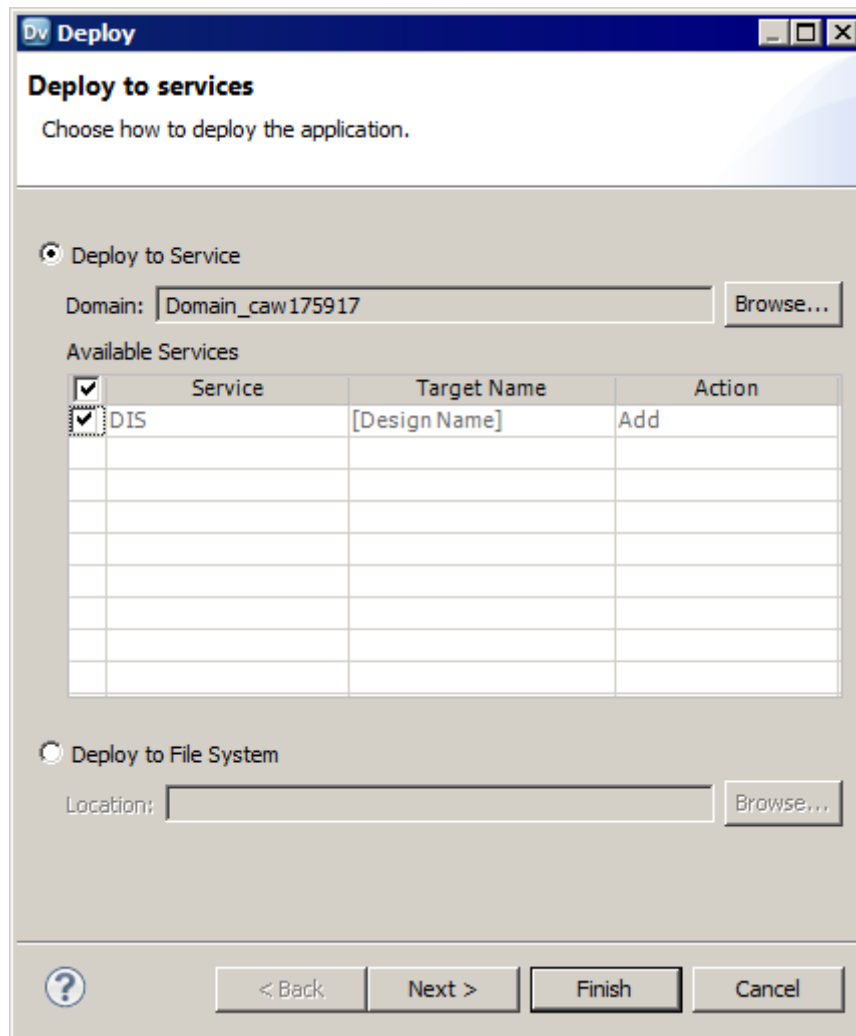
12. In the editor, right-click and select **Validate**.
If errors appear in the **Validation Log** view, fix the errors and validate the workflow again.
13. When the workflow is valid, click **File > Save** to save the workflow.

Step 3. Deploy and Run the Workflow

Deploy the workflow to the Data Integration Service and run an instance of the workflow.

1. In the **Object Explorer** view, expand Workflows in the Tutorial_Objects folder.
2. Right-click the wf_JoinCustomerTransactions workflow and select **Deploy**.
The **Deploy** dialog box opens.
3. Under **Deploy to Service**, click **Browse** to select the domain.
4. In the **Choose Domain** dialog box, select the domain and click **OK**.
5. In the **Deploy** dialog box, select the Data Integration Service under **Available Services**.

The following figure shows a completed **Deploy** dialog box:



6. Click **Next**.
7. Enter Tutorial_Application for the application name and click **Finish**.
8. In the **Deploy Completed** dialog box, click **Run Object** to run an instance of the deployed workflow.

9. In the **Run Object** dialog box, verify that the workflow is selected and then click **OK**.

The Developer tool deploys the workflow and then the Data Integration Service runs an instance of the workflow.

You can use the `infacmd wfs startWorkflow` command to run additional instances of the same workflow from the deployed application. For more information, see the PowerCenter Express *Command Reference*.

PowerCenter Express Tips

Use the following tips when you develop workflows.

Maximize the editor in the Developer tool when a workflow contains many objects.

If you are developing a workflow that contains many objects, you might want to maximize the editor so that you can view all objects in the workflow. To maximize the editor, click **Window > Navigation > Maximize Active View or Editor**.

Use the editor to find mapping and workflow objects that are open in the editor.

You can find mapping and workflow objects that are open in the editor. To display the find fields below the editor, select **Edit > Find/Replace**. To find an object, specify a search string and the types of objects to find.

CHAPTER 6

Monitoring a Workflow

This chapter includes the following topics:

- ◆ Monitoring a Workflow Overview, 46
- ◆ Step 1. Access the Monitoring Tool, 47
- ◆ Step 2. Monitor the Workflow and View Logs, 48
- ◆ PowerCenter Express Tips, 49

Monitoring a Workflow Overview

In this lesson, you access the Monitoring tool from the Developer tool. You use the Monitoring tool to monitor the workflow instance and to view the workflow and mapping logs.

Lesson Concepts

Use the Monitoring tool to monitor the status of running workflow instances. You can view the state, unique ID, start time, and elapsed time for each workflow instance. You can abort or cancel a running workflow instance.

You also use the Monitoring tool to view logs for workflow instances. The workflow log file includes information about errors, task processing, and expression evaluation in sequence flows. When a workflow instance includes a Mapping task, the Data Integration Service generates a separate log file for the mapping. The mapping log file includes any errors encountered during the mapping run and load summary and transformation statistics.

Lesson Objectives

In this lesson, you complete the following tasks:

- ◆ Access the Monitoring tool from the Developer tool.
- ◆ Monitor the workflow instance and view the workflow and mapping logs.

Lesson Prerequisites

Before you start this lesson, complete the earlier lessons in this tutorial.

Lesson Timing

Set aside 5-10 minutes to complete the tasks in this lesson.

Step 1. Access the Monitoring Tool

Access the Monitoring tool from the Developer tool to begin the lesson.

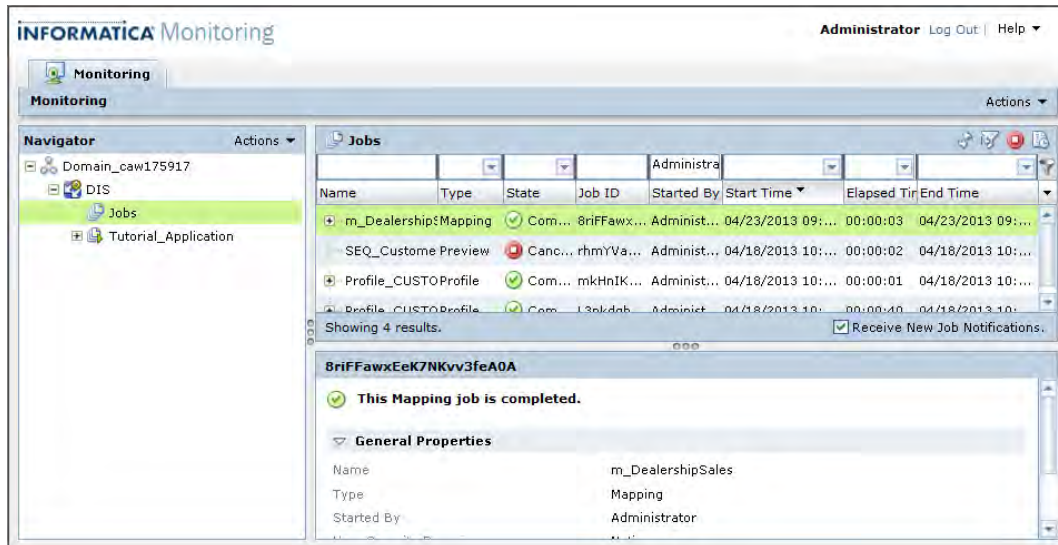
1. Click **Window > Show View > Progress** to open the **Progress** view.

2. Click the **Menu** button (☰) and select **Monitor Jobs**.

The **Choose Data Integration Service** dialog box opens.

3. Select the Data Integration Service to which you deployed the workflow, and then click **OK**.

The Monitoring tool opens.

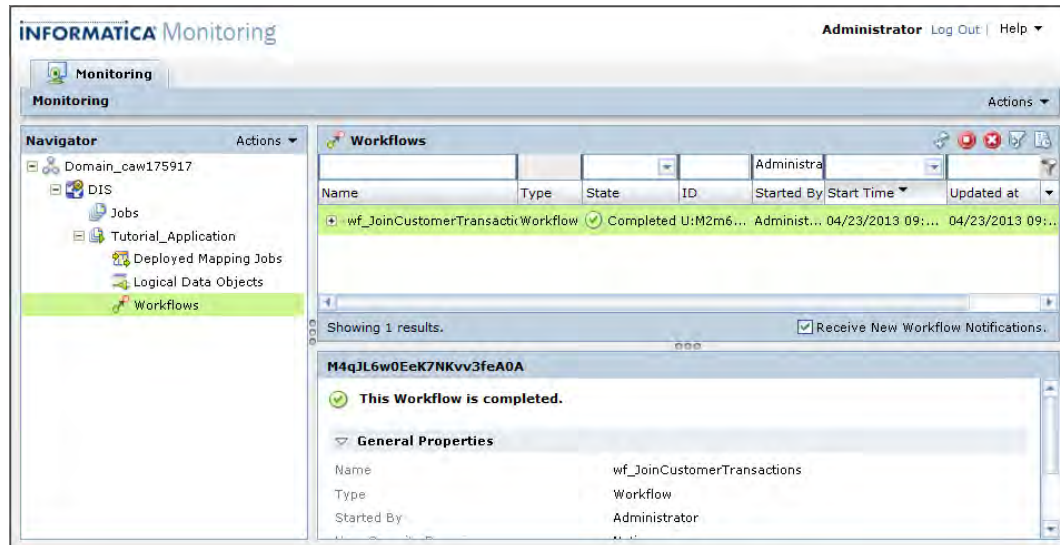


Step 2. Monitor the Workflow and View Logs

Monitor the workflow instance and view the workflow and mapping logs.

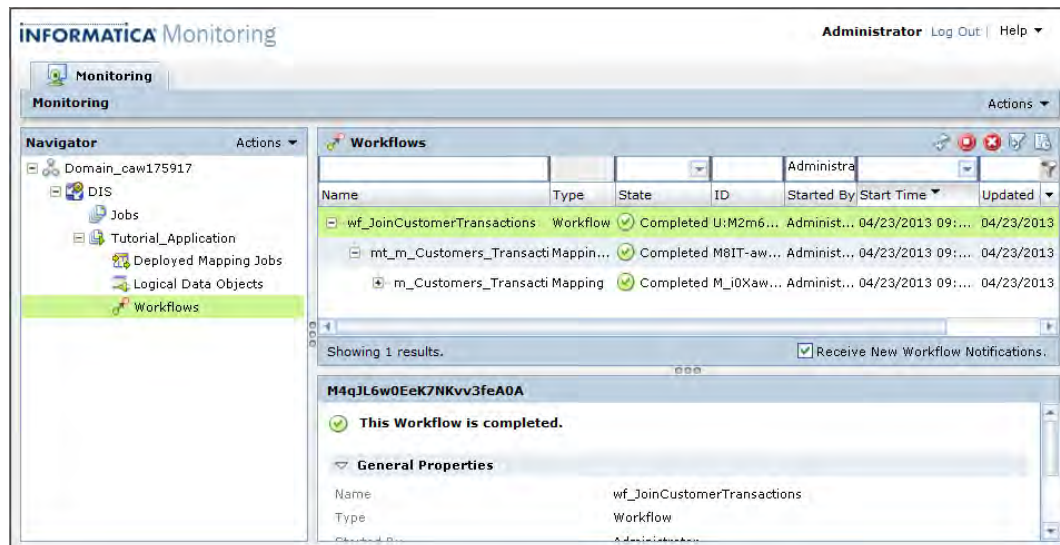
1. In the **Navigator** of the Monitoring tool, expand the application named Tutorial_Application, and then select **Workflows**.

The contents panel displays the workflow instance that was run from the deployed application.



2. In the contents panel, expand the workflow instance, and then expand the Mapping task.

The contents panel displays details about the workflow instance, the Mapping task, and the mapping that the Mapping task ran.



3. Select the workflow instance in the contents panel and then click **Actions > View Logs for Selected Object**.

The **File Download** dialog box appears. Notice that the name of the workflow log uses the following format:

<workflow instance ID>_log.txt

4. Click **Open** to view the workflow log file.
5. Close the workflow log file.

6. To view the mapping log, select the mapping in the contents panel and then click **Actions > View Logs for Selected Object**.
7. In the **File Download** dialog box, click **Open** to view the mapping log file.

PowerCenter Express Tips

Use the following tips when you monitor workflow instances.

Filter workflow instances by state.

When you monitor workflow instances, you can filter the workflow instances by state. For example, you might want to filter the workflow instances to display only the Failed workflow instances in the contents panel.

To filter workflow instances by state, click the arrow above the State column in the contents panel and select the state that you want to filter by.

View monitoring reports for the Data Integration Service or for an application.

You can view monitoring reports about objects that run on the Data Integration Service. For example, you can view the Longest Duration Workflows report to determine the workflows that were running the longest during the specified time period.

To view monitoring reports, select the Data Integration Service or an application in the **Navigator**. In the contents panel, click the **Reports** view.

CHAPTER 7

Creating a Connection

This chapter includes the following topics:

- ◆ Creating a Connection Overview, 50
- ◆ Step 1. Create a Connection, 51
- ◆ PowerCenter Express Tips, 54

Creating a Connection Overview

In this lesson, you create a connection to a relational database. After you create the connection, you can import data from the database as a physical data object, preview data in the database, run a profile on data in the database, and run mappings that read from and write to the database.

Lesson Concepts

A connection is a repository object that defines a connection in the domain configuration repository. Create connections to access data from relational databases, ODBC data sources, social media data sources, or third-party web services.

In this lesson, you create a connection to a relational database. You can create a connection to any of the following relational database types:

- ◆ IBM DB2
- ◆ Microsoft SQL Server
- ◆ Oracle

In addition, you can create a JDBC or an ODBC connection to access tables in a relational database through JDBC or ODBC. The properties that you configure for a connection depend on the database type.

The Developer tool stores connections in the domain configuration repository.

Note: All of the physical data objects in the Samples project in the Model repository are flat file data objects. Flat file data objects do not require a connection. As a result, the domain configuration repository does not contain sample connections.

Lesson Objectives

In this lesson, you complete the following task:

- ◆ Create a connection to a relational database.

Lesson Prerequisites

Before you start this lesson, verify the following prerequisites:

- ◆ You have the relational database host name and port number.
- ◆ You have a username and password to connect to the relational database.

Lesson Timing

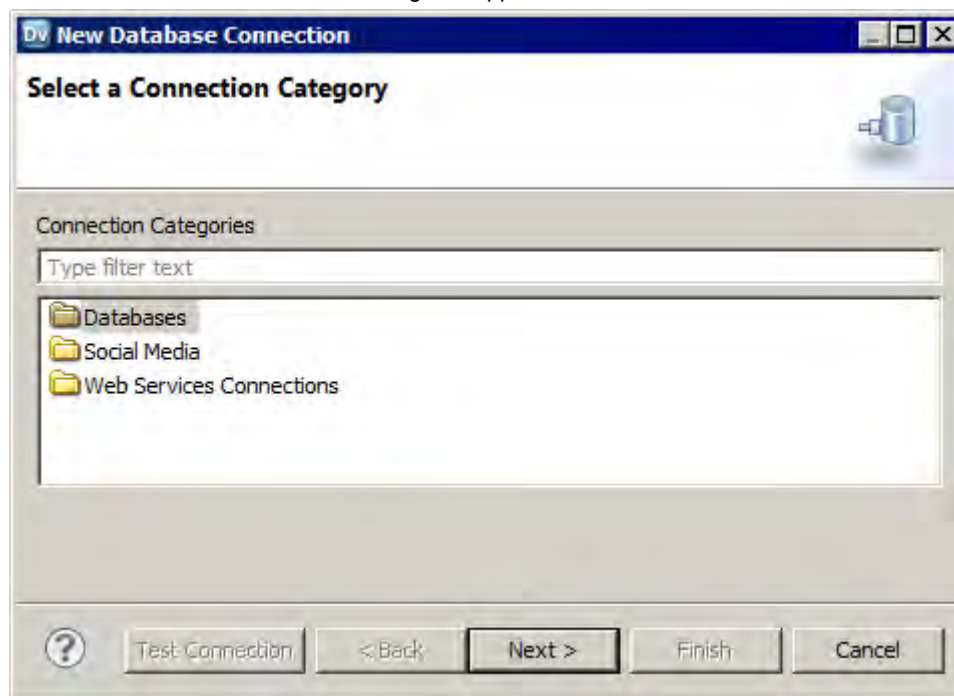
Set aside 5-10 minutes to complete the task in this lesson.

Step 1. Create a Connection

Use the Developer tool to create a connection to a relational database so that you can access data in the database.

1. In the Developer tool, click the **Create Connection** button () in the tool bar.

The **New Database Connection** dialog box appears.

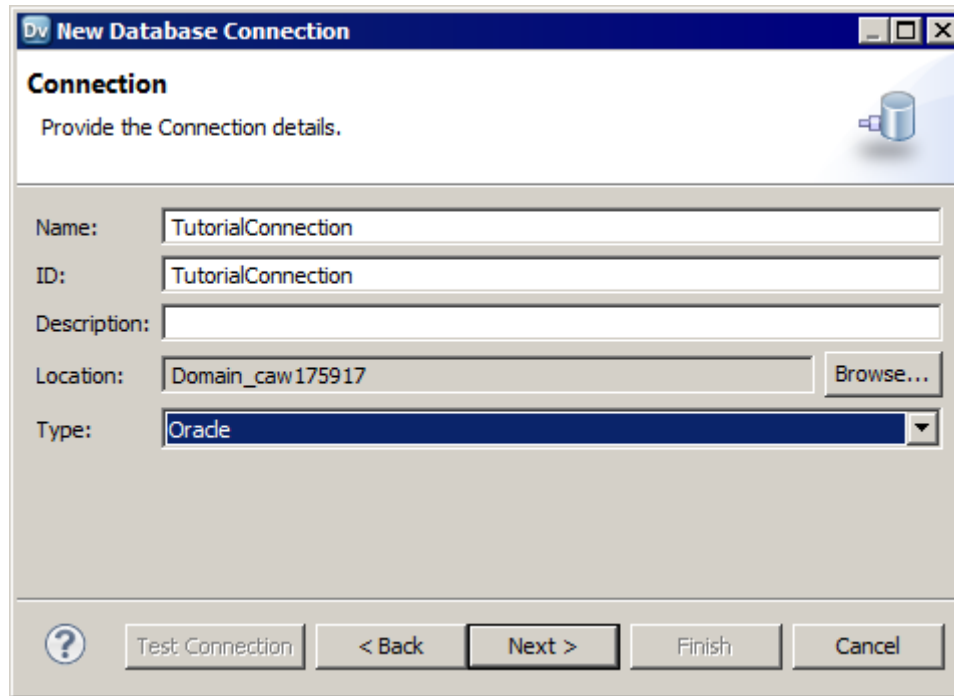


2. Select **Databases** for the connection category and click **Next**.
3. Enter a name for the connection.

The ID is a string that the Data Integration Service uses to identify the connection. You cannot change this property after you create the connection. Default value is the connection name.

4. Click **Browse** to select the domain in which the connection exists.
5. Select the database connection type.

The following figure shows Oracle selected as the database connection type:



6. Click **Next**.
7. Enter a username and password to access the database.
8. Configure the remaining connection properties. The properties depend on the connection type.

The following figure shows the properties for an Oracle connection:

New Database Connection

Connection Details
Provide connection details

User Name: TutorialUserName
Password: ●●●●●●●●
 Pass-through security enabled

Metadata Access
Connection String: jdbc:informatica:oracle://TutorialHost:1521;SID=TutorialSID

Data Access
Connection String: TutorialSID.informatica.com
Code Page: UTF-8 encoding of Unicode
Environment SQL:
Transaction SQL:
Retry Period: 0
 Parallel Mode
SQL identifier character: "" (quotes)
 Support mixed-case identifiers

? Test Connection < Back Next > Finish Cancel

9. Click **Test Connection** to verify that you entered the connection properties correctly and that you can connect.
10. Click **Finish**.
The **Add Connection** dialog box appears.
11. Click **Yes** to add the connection to the **Connection Explorer** view.

The new connection appears in the **Connection Explorer** view.

You can use the connection to import data from the database as a physical data object, preview data in the database, run a profile on data in the database, and run mappings that read from and write to the database.

PowerCenter Express Tips

Use the following tips when you manage connections.

Use the Connection Explorer view in the Developer tool to view data in a relational database.

You can add relational database connections to the **Connection Explorer** view. You can connect to the relational database by right-clicking the database in the **Connection Explorer** view and selecting **Connect**. Expand the database schema to view the tables, views, and synonyms in the schema.

Manage connections in the Developer tool and the Administrator tool.

You can use the Developer tool and the Administrator tool to create and manage connections. Any connection that you create in one tool is available in the other tool.

To manage connections in the Administrator tool, click the **Connections** view in the **Domain** tab.

APPENDIX A

Glossary

A

application

A deployable object that can contain data objects, mappings, and workflows.

application service

A service that runs on the node in the Informatica domain. You manage application services in Informatica Administrator or through the infacmd command program. Configure each application service based on your environment requirements.

associated service

An application service that is associated with another application service. For example, the Model Repository Service is associated with the Data Integration Service.

C

column profile

A type of profile that determines the characteristics of columns in a data source, such as value frequency, percentages, patterns, and datatypes.

customized data object

A physical data object that uses one or more related relational resources or relational data objects as sources. You use a customized data object to perform tasks such as join data from related resources or filter rows. Customized data object uses a single connection and SQL statement for the source tables.

D

Data Integration Service

An application service that performs data integration tasks for Informatica Developer. Data integration tasks include previewing data and running mappings, profiles, and workflows.

deploy

To make objects within an application accessible to end users. Depending on the types of objects in the application, end users can then run mappings or workflows.

domain

A domain is the fundamental administrative unit for the Informatica node and services.

DTM

The Data Transformation Manager process that reads, writes, and transforms data.

E

early projection optimization

Optimization method that reduces the amount of data that moves between transformations in the mapping. With early projection optimization, the Data Integration Service identifies unused ports and removes the links between the ports in a mapping.

early selection optimization

Optimization method that reduces the number of rows that pass through the mapping. With early selection optimization, the Data Integration Service moves filters closer to the mapping source in the pipeline.

F

folder

A container for objects in the Model repository. Use folders to organize objects in a project and create folders to group objects based on business needs.

I

Informatica Administrator

Informatica Administrator (the Administrator tool) is an application that consolidates the administrative tasks for domain objects such as services, nodes, licenses, and grids. You manage the domain and the security of the domain through the Administrator tool.

Informatica Developer

Informatica Developer (the Developer tool) is an application that you use to design data integration solutions. The Model repository stores the objects that you create in the Developer tool.

Informatica Monitoring tool

Informatica Monitoring tool (the Monitoring tool) is an application that provides a direct link to the Monitoring tab of the Administrator tool. The Monitoring tab shows properties, run-time statistics, and run-time reports about the integration objects that run on a Data Integration Service.

L

logical data object

An object that describes a logical entity in an organization. It has attributes and keys, and it describes relationships between attributes.

logical data object mapping

A mapping that links a logical data object to one or more physical data objects. It can include transformation logic.

logical data object model

A data model that describes data in an organization and the relationship between the data. It contains logical data objects and defines relationships between them.

logical data object read mapping

A mapping that provides a view of data through a logical data object. It contains one or more physical data objects as sources and a logical data object as the mapping output.

logical data object write mapping

A mapping that writes data to targets using a logical data object as input. It contains one or more logical data objects as input and a physical data object as the target.

M**mapping**

A set of inputs and outputs linked by transformation objects that define the rules for data transformation.

mapplet

A reusable object that contains a set of transformations that you can use in multiple mappings or validate as a rule.

Model Repository Service

An application service in the Informatica domain that runs and manages the Model repository. The Model repository stores metadata created by Informatica products in a relational database to enable collaboration among the products.

N**native authentication**

The authentication method used to authenticate users logging in to PowerCenter Express applications. The Service Manager stores group and user account information and performs authentication in the domain configuration database.

node

A logical representation of a machine or a blade. Each node runs a Service Manager that performs domain operations on that node.

P**permission**

The level of access a user has to a metadata object. Even if a user has the privilege to perform certain actions, the user may also require permission to perform the action on a particular object. You define object permissions in Metadata Manager. An object can inherit permissions from the containing folder or resource.

physical data object

A physical representation of data that is used to read from, look up, or write to resources.

pipeline branch

A segment of a pipeline between any two mapping objects.

predicate expression

An expression that filters the data in a mapping. A predicate expression returns true or false.

predicate optimization

Optimization method that simplifies or rewrites the predicate expressions in a mapping. With predicate optimization, the Data Integration Service attempts to apply predicate expressions as early as possible to increase mapping performance.

privilege

An action that a user can perform in Informatica Developer. You assign privileges to users and groups for the Informatica domain and the Repository Service.

privilege group

An organization of privileges that defines common user actions.

profile

An object that contains rules to discover patterns in source data. Run a profile to evaluate the data structure and verify that data columns contain the type of information that you expect.

project

The top-level container to store objects created in Informatica Developer. Create projects based on business goals or requirements.

R

result set caching

A cache that contains the results of each SQL data service query or web service request. With result set caching, the Data Integration Service returns cached results when users run identical queries. Result set caching decreases the run time for identical queries.

rule

Reusable business logic that defines conditions applied to source data when you run a profile. Use rules to further validate the data in a profile and to measure data quality progress. You can create a rule in Informatica Analyst or Informatica Developer.

S

semi-join optimization

Optimization method that reduces the number of rows extracted from the source. With semi-join optimization, the Data Integration Service modifies the join operations in a mapping. The Data Integration Service applies the semi-join optimization method to a Joiner transformation when a larger input group has rows that do not match a smaller input group in the join condition. The Data Integration Service reads the rows from the smaller group, finds the matching rows in the larger group, and performs the join operation.

P

pushdown optimization

Optimization method that pushes transformation logic to a source or target database. With pushdown optimization, the Data Integration Service translates the transformation logic into SQL queries and sends the SQL queries to the database. The database runs the SQL queries to process the data.

S

service process

A run-time representation of a service running on a node.

T

transformation

A repository object in a mapping that generates, modifies, or passes data. Each transformation performs a different function.