Informatica

Informatica® B2B Data Transformation Accelerator

9.0.1 HF1

Informatica® Conversion Agent by Informatica

9.0.1 HF1

User Guide

Informatica B2B Data Transformation Accelerator Conversion Agent by Informatica User Guide
9.0.1 HF19.0.1 HF1
February 2014

© Copyright Informatica LLC 2001, 2018

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. Copyright © PDFlib GmbH. All rights reserved. Copyright © Orientation in Objects GmbH. All rights reserved. Copyright © Tanuki Software, Ltd. All rights reserved. Copyright © Ricebridge. All rights reserved. Copyright © Sencha, Inc. 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 1996-2013, 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 and at http://www.eclipse.org/org/documents/edl-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/hsqlLicense.html, http:// httpunit.sourceforge.net/doc/ license.html, http://jung.sourceforge.net/license.txt , http://www.gzip.org/zlib/zlib_license.html, http://www.openldap.org/software/

release/license.html, http://www.libssh2.org, http://slf4j.org/license.html, http://www.sente.ch/software/OpenSourceLicense.html, http://fusesource.com/downloads/license-agreements/fuse-message-broker-v-5-3- license-agreement; http://antlr.org/license.html; http://aopalliance.sourceforge.net/; http://www.bouncycastle.org/licence.html; http://www.jgraph.com/jgraphdownload.html; http://www.jcraft.com/jsch/LICENSE.txt; http://jotm.objectweb.org/bsd_license.html; . http://www.w3.org/Consortium/Legal/2002/copyright-software-20021231; http://www.slf4j.org/license.html; http://nanoxml.sourceforge.net/orig/copyright.html; http://www.json.org/license.html; http://forge.ow2.org/projects/javaservice/, http://www.postgresql.org/about/licence.html, http://www.sqlite.org/copyright.html, http://www.tcl.tk/software/tcltk/license.html, http://www.jaxen.org/faq.html, http://www.jdom.org/docs/faq.html, http://www.slf4j.org/license.html; http://www.iodbc.org/dataspace/iodbc/wiki/iODBC/License; http://www.keplerproject.org/md5/license.html; http://www.toedter.com/en/jcalendar/license.html; http://www.edankert.com/bounce/index.html; http://www.net-snmp.org/about/license.html; http://www.openmdx.org/#FAQ; http://www.php.net/license/3_01.txt; http://srp.stanford.edu/license.txt; http://www.schneier.com/blowfish.html; http://www.jmock.org/license.html; http://xsom.java.net; http://benalman.com/about/license/; https://github.com/CreateJS/EaselJS/blob/master/src/easeljs/display/Bitmap.js; http://www.h2database.com/html/license.html#summary; http://jsoncpp.sourceforge.net/LICENSE; http://jdbc.postgresql.org/license.html; http://protobuf.googlecode.com/svn/trunk/src/google/protobuf/descriptor.proto; https://github.com/rantav/hector/blob/master/LICENSE; http://web.mit.edu/Kerberos/krb5-current/doc/mitK5license.html. and http://jibx.sourceforge.net/jibx-license.html.

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 new BSD License (http://opensource.org/licenses/BSD-3-Clause), the MIT License (http://www.opensource.org/licenses/mit-license.php), the Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) and the Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-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 product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

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; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; 7,774,791; 8,065,266; 8,150,803; 8,166,048; 8,166,071; 8,200,622; 8,224,873; 8,271,477; 8,327,419; 8,386,435; 8,392,460; 8,453,159; 8,458,230; and RE44,478, 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.

Publication Date: 2018-06-14

Table of Contents

Preface 7

Informatica Resources. 7

Informatica Network. 7

Informatica Knowledge Base. 7

Informatica Documentation. 7

Informatica Product Availability Matrixes. 8

Informatica Velocity. 8

Informatica Marketplace. 8

Informatica Global Customer Support. 8

Obtaining Conversion Agent Documentation. 8

Chapter 1: Understanding Data Transformation Accelerator. 9

Accelerator Overview. 9

Specifying a Mapper in Accelerator. 9

Integration with Other Data Transformation Modules. 10

Chapter 2: Installation. 11

System Requirements. 11

Platform. 11

Microsoft Excel. 11

Informatica Data Transformation 11

Other Microsoft Prerequisites. 11

XML File Viewer. 11

Upgrading Projects from Previous Accelerator Version. 12

Installing Accelerator. 12

Uninstalling Accelerator. 13

Chapter 3: Getting Started. 14

About Mappers. 14

Terminology. 14

Accelerator Window. 14

Informatica Tab. 15

Worksheets. 15

Document Actions Pane. 15

Tutorial: Creating a Mapper. 17

Input Files. 17

Specifying the Mapper. 18

Running and Testing the Mapper. 20

Chapter 4: Accelerator Projects. 23

Projects Overview. 23
Project Files. 23
Creating and Editing Projects. 24
Opening Accelerator from the Desktop and Creating a Project. 24
Opening Accelerator from the Desktop and Editing a Project. 24
Sharing Projects with Data Transformation Studio 24
Exporting a Mapper from Accelerator to the Studio. 24
Moving a Mapper Repeatedly Between Accelerator and the Studio. 25
Deploying a Mapper as a Data Transformation Service 25

Chapter 5: Specifying Mappers. 26

Mapper Specification Overview. 26
Informatica Tab. 27
Fundamental Procedures. 28
Schemas. 28
Aliases for XPaths. 29
Sample Data. 30
Mappings. 30
Automap. 31
Editing Operations. 31
Sequence of Mappings. 32
Navigating. 32
Advanced Mode. 33
Iterated Mappings. 34
Running a Mapper. 35
Test Driving a Mapper. 35
Using Expressions to Manipulate Data. 38
Defining an Expression. 38
Date and Time Functions. 40
User-Defined Functions. 42
Lookup Functions. 43
Using Conditions. 44
Using Groups. 45
Nested Groups. 46
Using Groups and Conditions to Define Alternative Mappings. 46
Using Groups to Control Iterations. 47
Selecting Occurrences of Elements. 48
Selecting by Occurrence Number. 48
Selecting by Key. 49
Reusable Mappers. 51
Specifying a Reusable Mapper. 52

Using a Reusable Mapper. 52
Exporting and Importing Reusable Entities. 53
Exporting Reusable Entities. 53
Importing Reusable Entities. 53
Chaining a Mapper with other Transformations. 53
Configuring a Pre-Mapper Transformation. 53
Configuring a Post-Mapper Transformation. 54
Effects of Pre- and Post-Mappers on Accelerator Behavior. 54

Appendix A: Properties Reference. 55

Overview of the Properties Reference. 55
Cover Worksheet. 55
Main Mapping Worksheet. 58
Reusable Mappers Worksheet. 60
Function Definitions Worksheet. 60
Keys Worksheet. 61
Lookups Worksheet. 61
Aliases Worksheet. 62

Index. 63

Preface

The *Data Transformation Libraries Guide* is written for developers and analysts who need to transform data to or from industry standard messaging formats. This guide explains how to use the library transformation components with the industry standards and how to customize the library components for specialized needs. This guide assumes that you understand the standards you are using and that you know how to deploy and run Data Transformation services.

Informatica Resources

Informatica Network

Informatica Network hosts Informatica Global Customer Support, the Informatica Knowledge Base, and other product resources. To access Informatica Network, visit https://network.informatica.com.

As a member, you can:

- Access all of your Informatica resources in one place.
- Search the Knowledge Base for product resources, including documentation, FAQs, and best practices.
- View product availability information.
- Review your support cases.
- Find your local Informatica User Group Network and collaborate with your peers.

Informatica Knowledge Base

Use the Informatica Knowledge Base to search Informatica Network for product resources such as documentation, how-to articles, best practices, and PAMs.

To access the Knowledge Base, visit https://kb.informatica.com. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team at KB_Feedback@informatica.com.

Informatica Documentation

To get the latest documentation for your product, browse the Informatica Knowledge Base at https://kb.informatica.com/_layouts/ProductDocumentation/Page/ProductDocumentSearch.aspx.

If you have questions, comments, or ideas about this documentation, contact the Informatica Documentation team through email at infa_documentation@informatica.com.

Informatica Product Availability Matrixes

Product Availability Matrixes (PAMs) indicate the versions of operating systems, databases, and other types of data sources and targets that a product release supports. If you are an Informatica Network member, you can access PAMs at https://network.informatica.com/community/informatica-network/product-availability-matrices.

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. 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 are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

If you have questions, comments, or ideas about Informatica Velocity, contact Informatica Professional Services at ips@informatica.com.

Informatica Marketplace

The Informatica Marketplace is a forum where you can find solutions that augment, extend, or enhance your Informatica implementations. By leveraging any of the hundreds of solutions from Informatica developers and partners, you can improve your productivity and speed up time to implementation on your projects. You can access Informatica Marketplace at https://marketplace.informatica.com.

Informatica Global Customer Support

You can contact a Global Support Center by telephone or through Online Support on Informatica Network.

To find your local Informatica Global Customer Support telephone number, visit the Informatica website at the following link:
http://www.informatica.com/us/services-and-training/support-services/global-support-centers.

If you are an Informatica Network member, you can use Online Support at http://network.informatica.com.

Obtaining Conversion Agent Documentation

On Windows platforms, the Conversion Agent documentation is supplied as online help with the software. You can download PDF copies of the Conversion Agent manuals from:

http://help.sap.com/saphelp_nwpi71/helpdata/en/43/fc39c16bfb025ee10000000a1553f7/frameset.htm

CHAPTER 1

Understanding Data Transformation Accelerator

This chapter includes the following topics:

- Accelerator Overview, 9
- Specifying a Mapper in Accelerator, 9
- Integration with Other Data Transformation Modules, 10

Accelerator Overview

Data Transformation Accelerator enables data analysts and transformation designers to specify, configure, and test mappers. Mappers are transformations that convert XML source documents to XML target documents.

Accelerator runs as an add-in within Microsoft Excel. The mapper specifications are Excel workbooks that you can share with other Excel users.

After you create a mapper in Accelerator, you can work with the mapper in the following ways:

- You can deploy the mapper as a service that runs in Data Transformation Engine.
- You can export the mapper to Data Transformation Studio. In the Studio, you can test and further configure the mapper, and you can integrate the mapper with other transformation components. You can move the mapper between Accelerator and the Studio and edit it in either environment.
- You can provide the Accelerator specification to the development team as an Excel workbook.

Specifying a Mapper in Accelerator

The following steps describe the general workflow to specify a mapper in Accelerator:

1. Outside Accelerator, prepare XML schemas of the source and target data, and prepare an XML file containing sample source data.
2. Open Accelerator and create a project.
3. Import the source and target schemas to the Accelerator project.
4. Import the sample source data to the Accelerator project.

5. On the Accelerator worksheets, map the source elements and attributes to the targets.
6. Test the mapper in Accelerator using the sample source data.
7. Deploy the mapper specification as a service to Data Transformation Engine, or export it to Data Transformation Studio for further testing and configuration.

Integration with Other Data Transformation Modules

Accelerator integrates with other Data Transformation modules in the following ways:

- From Accelerator, you can export a mapper to a Studio project. Later, you can import the project back to Accelerator, including changes that you implemented in the Studio.
- From Accelerator, you can deploy a mapper as a service to Data Transformation Engine.

CHAPTER 2

Installation

This chapter includes the following topics:

- System Requirements, 11
- Upgrading Projects from Previous Accelerator Version, 12
- Installing Accelerator, 12
- Uninstalling Accelerator, 13

System Requirements

Before installing Accelerator, your computer must have the following software installed.

Platform

Microsoft Windows XP or Microsoft Windows 7.

Microsoft Excel

Microsoft Excel 2007 or 2010. Be sure to install all Microsoft Excel updates.

Informatica Data Transformation

Data Transformation 9.0 or 9.0.1.

Other Microsoft Prerequisites

The Accelerator setup installs the following Microsoft components if they are not already installed:

- Microsoft .NET Framework 3.5 SP1
- Microsoft Visual Studio Tools for Office 3.0 runtime (VSTO 3.0)
- Microsoft Office Primary Interop Assemblies (PIA) 2007

XML File Viewer

On the Accelerator computer, associate XML files with Internet Explorer or another XML file viewer. Accelerator displays XML files in the viewer that you configure.

Upgrading Projects from Previous Accelerator Version

To upgrade from a previous Accelerator version, you must uninstall the old version before installing the new one. If you upgrade Data Transformation, you do not need to reinstall Accelerator.

Accelerator 9.0 and earlier saved projects as Excel 2003 XLS files. The current Accelerator version saves projects in the newer Excel XLSX format. If you are upgrading from Accelerator 9.0 or earlier, and you want to continue to use your existing XLS projects. you must upgrade the projects before you uninstall Accelerator or Excel 2003.

To upgrade existing XLS projects:

1. Open each XLS file in the previous Accelerator version.
2. Click Informatica > Export > Workbook Content.

   This saves the project content as a DTA file. Repeat for each XLS project.
3. Uninstall Excel 2003 and the old Accelerator version.
4. Install the new Excel version.
5. Install the new Accelerator version.
6. Open the new Accelerator version.

   Do not make any entries in the workbook.
7. On the Informatica tab, click Import Workbook Content, and browse to a DTA file.

   The Import Workbook Content command is available only in a completely blank workbook.
8. Save the project as an XLSX file.

   Repeat for each of the DTA files that you saved. You can now use the projects in the new Accelerator version.

Installing Accelerator

To install Accelerator:

1. Open Excel.
2. In Excel 2007, click the Office Button, and follow the links to Excel Options > Trust Center > Trust Center Settings > Macro Settings. Select the option to Trust Access to the VBA Project Object Model.

   In Excel 2010, display the File tab, and follow the links to Options > Trust Center > Trust Center Settings > Macro Settings. Select the option to Trust Access to the VBA Project Object Model.
3. Close Excel.
4. Unzip the Accelerator setup package to an empty directory.

   The package contains the Accelerator software and Microsoft prerequisites such as .NET Framework, VSTO, and PIA.

5. Double-click `setup.exe` and follow the on-screen instructions.

   The setup prompts to install the Microsoft prerequisites if they are not already installed. It then installs Accelerator.

   For convenient operation when you transfer workbooks between computers, we recommend installing in the default location. If you install in another location, Accelerator might need to update the workbooks when you transfer them.

Uninstalling Accelerator

To uninstall Accelerator:

1. In the Windows Control Panel, open Add or Remove Programs.
2. Select Data Transformation Accelerator and click Remove.
3. Follow the on-screen instructions.

CHAPTER 3

Getting Started

This chapter includes the following topics:

- About Mappers, 14
- Accelerator Window, 14
- Tutorial: Creating a Mapper, 17

About Mappers

To help you learn Accelerator, this chapter presents an overview of the Accelerator window. It presents a tutorial exercise where you create a simple mapper.

Terminology

Accelerator uses the following terminology:

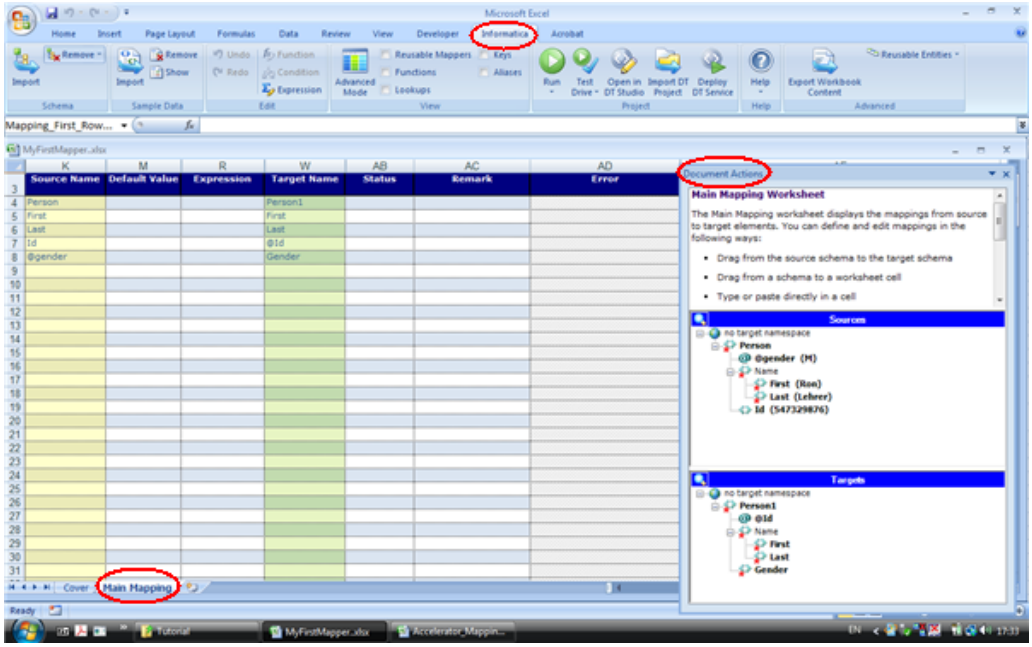
- Mapper. A transformation that can run in Data Transformation Engine. A mapper transforms a source XML document to a target XML schema.
- Mapping. A definition of how a source element or attribute is transformed to a target element or attribute. For example, you will define a mapping from the `gender` attribute of the source to the `Gender` element of the target.
- Specifying a mapper. The process of defining the set of mappings and how they operate. The result of the process is the completed mapper.

Accelerator Window

Accelerator displays an Excel window with the following special features:

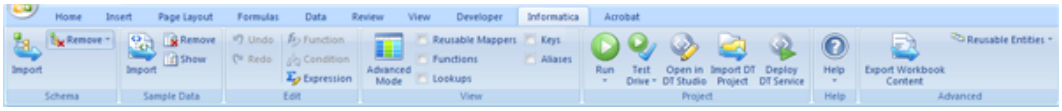
- An Informatica tab containing Accelerator commands.
- A Document Actions pane containing Accelerator tools, including explanations and schemas.

- Worksheets where you can configure the mapper features. The following figure shows a worksheet with mapper features:



Informatica Tab

The Informatica tab provides Accelerator commands that you can use to configure mappers. The following figure shows the Accelerator commands in the Informatica tab:



For more information about the Accelerator commands, see "Informatica Tab" on page 27.

Worksheets

When you open a new project, Accelerator displays the following worksheets:

- Cover worksheet, which displays general information and project options
- Main Mapping worksheet, where you specify a mapper

To expose additional worksheets, select the checkboxes in the View group of the Informatica ribbon. Deselects the checkboxes to hide the worksheets. For example, select the Functions checkbox to display the Functions worksheet.

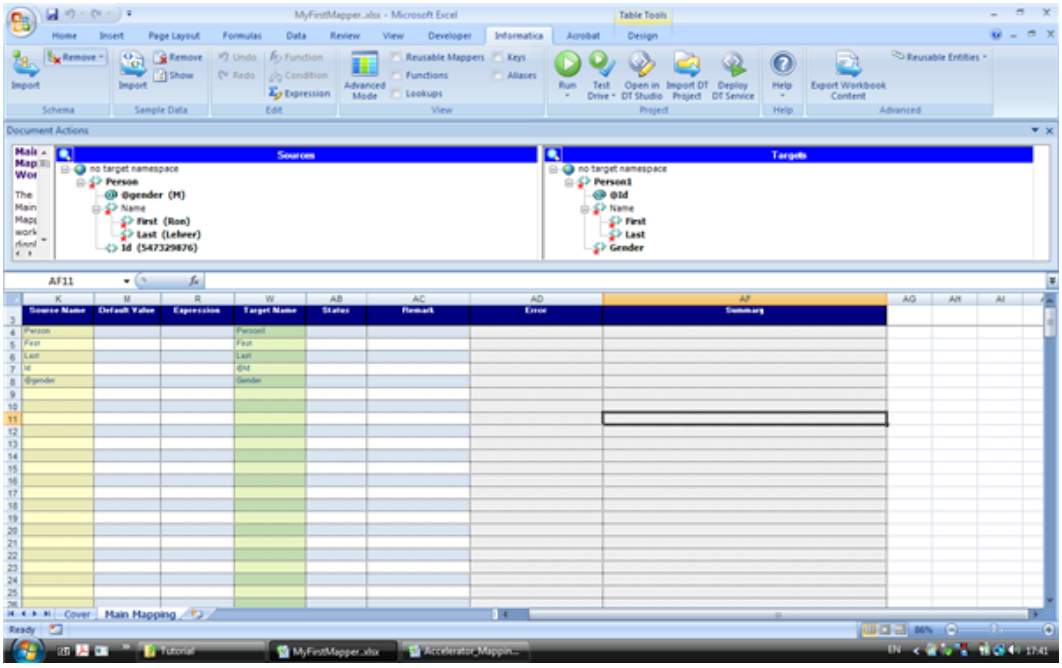
To switch between the worksheets, click the tabs at the bottom of the Excel window. If the tabs are not visible, maximize both the Excel window and the workbook window.

Document Actions Pane

By default, Excel displays the Document Actions pane on the right side of the window. You might find it convenient to dock the pane above the worksheet.To move the pane, drag the move handle near the upper left corner of the pane.

On the Main Mapping worksheet, the Document Actions pane displays a graphical view of the source and target schemas. It displays the sample source data next to the source elements and attributes.

The following figure shows the Document Actions pane with the source and target schemas:



The following table shows the symbols that the schemas use:

Symbol	Description	Meaning
<>	Brackets	Element
@	At symbol	Attribute
<>*	Red asterisk	Mandatory element
<>	Arrow	Multiple-occurrence element
Name	Normal font	Element or attribute that is not yet mapped
Last	Bold font	Element or attribute that is mapped
(Lehrer)	Parentheses	Sample data
[LastName]	Square brackets	Alias for an XPath
***	Additional symbols	Sequence or choice symbols, displayed if the XPath Format Option is set to DT XPath on the Cover worksheet

Tutorial: Creating a Mapper

In this section, you use Accelerator to create a simple mapper. The tutorial introduces the basic procedures for working with Accelerator.

Input Files

You can find the input files for the tutorial in the following subdirectory of the Accelerator installation directory:

```
Documentation\Tutorial\MyFirstMapper
```

The following table shows the input files:

File	Description
`Person.xsd`	Source schema
`Person.xml`	Sample source data
`Person1.xsd`	Target schema

In addition, the directory contains a sample Accelerator workbook, `MyFirstMapper.xlsx`, illustrating a solution to the tutorial.

Source Schema

The schema of the source documents is a file called `Person.xsd`. The following figure shows a sample schema named Person.xsd:

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Person">
    <xs:complexType>
      <xs:sequence>
        <xs:element name="Name">
          <xs:complexType>
            <xs:sequence>
              <xs:element name="First" type="xs:string"/>
              <xs:element name="Last" type="xs:string"/>
            </xs:sequence>
          </xs:complexType>
        </xs:element>
        <xs:element minOccurs="0" name="Id" type="xs:string"/>
        <xs:element minOccurs="0" name="Age" type="xs:integer"/>
      </xs:sequence>
      <xs:attribute name="gender" type="xs:string" use="optional"/>
    </xs:complexType>
  </xs:element>
</xs:schema>
```

Sample Source Data

The following figure shows the sample source document. The document conforms to the source schema. The following figure shows a sample source data file named Person.xml:

```
  <?xml version="1.0" encoding="windows-1252" ?>
- <Person gender="M">
  - <Name>
      <First>Ron</First>
      <Last>Lehrer</Last>
    </Name>
    <Id>547329876</Id>
  </Person>
```

Target Schema

The target schema differs from the source schema in the following ways:

- The document element is called `Person1` instead of `Person`.
- `Gender` is an element rather than an attribute.
- `Id` is an attribute rather than an element.
- The following figure shows a target schema named Person1.xsd:

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="Person1">
        <xs:complexType>
            <xs:sequence>
                <xs:element name="Name">
                    <xs:complexType>
                        <xs:sequence>
                            <xs:element name="First" type="xs:string"/>
                            <xs:element name="Last" type="xs:string"/>
                        </xs:sequence>
                    </xs:complexType>
                </xs:element>
                <xs:element name="Gender" type="xs:string"/>
            </xs:sequence>
            <xs:attribute name="ID" type="xs:string" use="optional"/>
        </xs:complexType>
    </xs:element>
</xs:schema>
```

Specifying the Mapper

To create a mapper, you must specify the following items:

- Source schema
- Target schema
- Mapping between the source elements and attributes and the target elements and attributes

In addition, you can import sample source data. Accelerator displays the sample data in the schema view. The data helps you to configure and test the mapper.

To specify the mapper:

1. On the Start menu, click Informatica > Data Transformation Accelerator > Data Transformation Accelerator.

   An Accelerator workbook opens in a Microsoft Excel environment.

2. Save the workbook with a name such as `MyFirstMapper.xlsx`.

3. On the Cover worksheet, enter a Project Name such as `MyFirstMapper`. The following figure shows the Cover worksheet with filled in fields:

General	
Project Name	MyFirstMapper
XPath display option	Parent XPath and node name
XPath format option	Standard XPath
Show aliases	Yes

4. Perform the following tasks to import the source and target schemas:
   - In the Schema group of the Informatica tab, click Import.
   - Click the Browse button and locate the `Documentation Tutorial\MyFirstMapper` directory.
   - Select `Person.xsd` and `Person1.xsd`. Press the CTRL key to select multiple files.
   - Click Open, and then click OK.

   Alternatively, you can import the schemas one at a time.

   Accelerator informs you when the import is complete.
5. Display the Main Mapping worksheet.
6. In the Document Actions pane, expand the Sources and Targets.

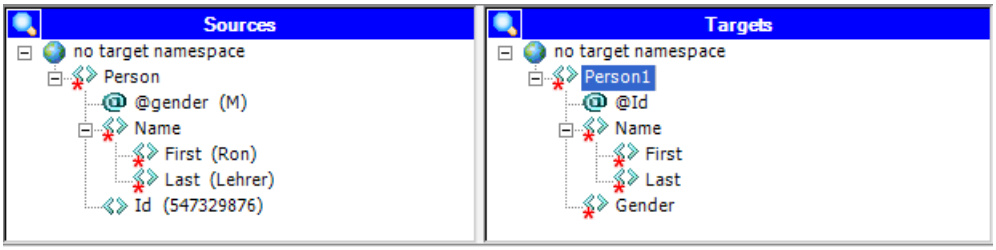
   The displays are graphical representations of the schemas. They appear under the `no target namespace` heading because the schemas do not define a namespace.

   By default, Accelerator displays both the `Person` schema and the `Person1` schema as both potential Sources and Targets.
7. To display only `Person` under Sources, right-click `Person` and click Set As Root.

   This hides the `Person1` schema.
8. To display only `Person1` under Targets, right-click `Person1` and click Set As Root.

   This hides the `Person` schema.
9. Import the sample source data:
   - In the Sample Data group of the Informatica tab, click Import.
   - Click the browse button and select `Person.xml`.
   - Click Open and then click OK.
   - Accelerator notifies you when the import is complete.

   The source schema displays the sample data. The following figure shows the source schema:



10. In the Sources, select the `Person` element, and drag it to the `Person1` element in the Targets.

This specifies a mapping from a `Person` document to a `Person1` document. The Main Mapping worksheet displays the mapping. The following figure shows the Mapping worksheet:

Source Name	Default Value	Expression	Target Name
Person			Person1

11. To map data from the source schema to the target schema, drag a source element or attribute to the corresponding target. For example, drag the `First` element, located under `Person` in the Sources pane, to the `First` element, located under `Person1` in the Targets pane.

    Accelerator adds a row to the Main Mapping worksheet. The row displays the mapping that you have defined.

12. Continue by mapping the source and target nodes. The following table shows the source and target node details:

Source	Target
`Person` (document element)	`Person1` (document element)
`First`	`First`
`Last`	`Last`
`@gender`	`Gender`
`Id`	`@Id`

    Accelerator adds a row to the Main Mapping worksheet for each mapping you specify. The following figure shows the Main Mapping worksheet with the details for the source and target nodes:

Source Name	Default Value	Expression	Target Name
Person			Person1
First			First
Last			Last
@gender			Gender
Id			@Id

13. Save the Excel workbook.

Running and Testing the Mapper

After you specify the mapper, you can run and test it in Accelerator. Optionally, you can export the mapper to Data Transformation Studio for further testing and editing.

From either Accelerator or the Studio, you can deploy the mapper as a Data Transformation service. Data Transformation Engine can run the service to perform production transformations.

To run and test the mapper:

1. On the Informatica tab, click Run.

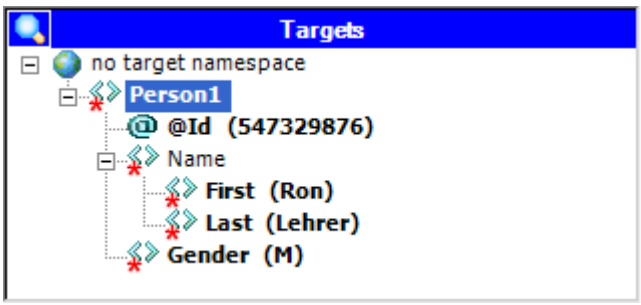
   Accelerator runs the mapper using the currently imported sample data.

   To run the mapper with different sample data, click the arrow under the Run button, and click Run with External Data. Accelerator prompts you for the source data file.

2. Accelerator displays the output in the default XML viewer.

3. Close the XML viewer and return to Accelerator.

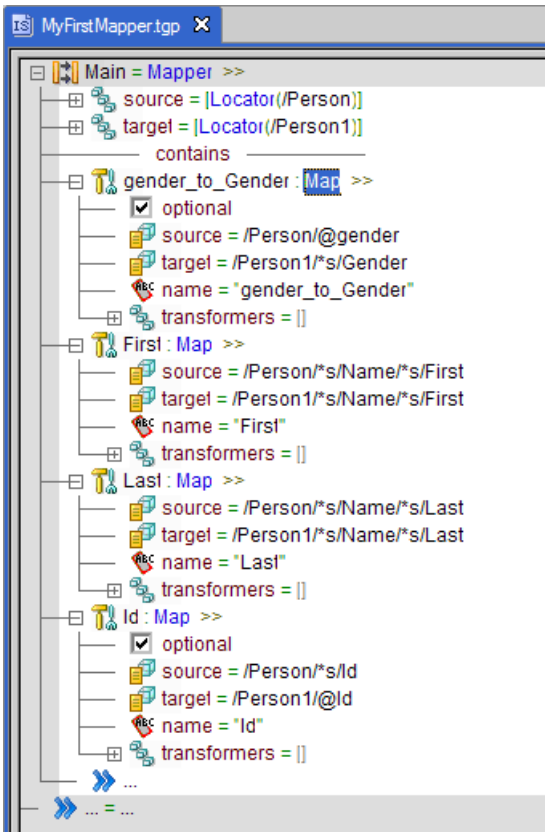
   Accelerator displays the output in the Targets window.



4. On the Informatica tab, click Open in DT Studio.

   This opens the Data Transformation Studio. The Data Transformation Explorer view displays the `MyFirstMapper` project.

   You can work with the project in the same way as any other Studio project. For example, you can display and edit the IntelliScript:

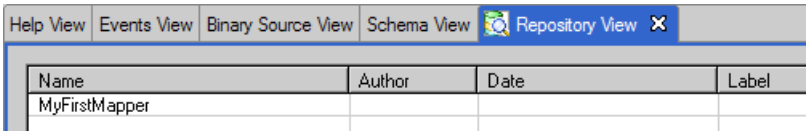


5. Return to Accelerator. On the Informatica tab, click Deploy DT Service.

   This deploys the mapper as a service that can run in Data Transformation Engine. You can view the service subdirectory in the Data Transformation repository directory. By default, the repository is in the following location:

   `c:\Informatica\9.5.1\DataTransformation\ServiceDB`

   You can also view the service in the Repository view of the Studio:



   For more information about running the service, see the *Data Transformation Engine Developer Guide*.

CHAPTER 4

Accelerator Projects

This chapter includes the following topics:

- Projects Overview, 23
- Project Files, 23
- Creating and Editing Projects, 24
- Sharing Projects with Data Transformation Studio , 24
- Deploying a Mapper as a Data Transformation Service , 25

Projects Overview

This chapter explains the procedures for working with Accelerator project files. You can open Accelerator from the desktop or from Data Transformation Studio. You can transfer projects to or from the Studio, and you can deploy projects as services that run in Data Transformation Engine.

Project Files

An Accelerator project is an Excel workbook, saved as an Excel XLSX file.

You can share the file with other Accelerator or Excel users. If you open an Accelerator file in a copy of Excel where Accelerator is not installed, it appears as an ordinary Excel workbook. You can view the mapper definition, but you cannot test the mapper, view the schemas, or use the other Accelerator functionality.

If you receive an Accelerator file as an email attachment, save it to a local disk before opening. You cannot open an Accelerator file directly from an email.

If you open a workbook that was previously edited on another computer, Microsoft Office might display customization or installation errors. You can safely ignore the Microsoft Office messages. They arise because the workbook contains a pointer to a customization path, which might be in a different location on the other computer. Accelerator updates the workbook to the new customization path. You can then edit and use the workbook.

Creating and Editing Projects

You can open Accelerator from the Windows desktop or Start menu. You can create an Accelerator project or edit an existing project.

Opening Accelerator from the Desktop and Creating a Project

To open Accelerator from the Windows desktop and create a project:

1. Double-click the Data Transformation Accelerator shortcut on the desktop, or open Informatica > Data Transformation Accelerator from the Start menu.

   Excel displays an Accelerator workbook.
2. Save the workbook as an Excel XLSX file.

Opening Accelerator from the Desktop and Editing a Project

To edit a project:

- Double-click the existing XLSX file, or open it from the Open command of Excel.

Sharing Projects with Data Transformation Studio

Within Accelerator, you can export a mapper project to the directory and file structure required by Data Transformation Studio. You can move a mapper from Accelerator to Studio and back to Accelerator, making as many round trips as you want. In this way, you can edit the mapper features in either Accelerator or the Studio as you find convenient.

Exporting a Mapper from Accelerator to the Studio

To export a mapper from Accelerator to the Studio:

1. From the desktop or Start menu, open Accelerator.
2. Create or edit a project, defining a mapper.
3. On the Cover worksheet, enter the Project Name.

   The Studio project will have this name. If a Studio project with this name already exists, the exported project overwrites the existing project.
4. Save the XLSX file.
5. On the Informatica tab, click Open in DT Studio.

   The system converts the Accelerator mapper to the Studio format. It saves the exported project as a directory in the Studio workspace, and it opens the Studio. You can test or edit the project in the Studio.

Moving a Mapper Repeatedly Between Accelerator and the Studio

To move a mapper repeatedly between Accelerator and the Studio:

1. Export a mapper from Accelerator to the Studio.

   You must start with a mapper created in Accelerator. If you create the original mapper in the Studio, you cannot import it reliably to Accelerator.
2. In the Studio, edit and test the mapper as desired.
3. Save the Studio project.
4. Return to Accelerator, and open the original XLSX file of the project if it is not already open.
5. On the Informatica tab, click Import DT Project.
6. Locate and open the CMW file of the Studio project.

   By default, the project is stored in the Studio workspace, `c:\Users\<User_Name>\Informatica\DataTransformation\950\workspace` .

   Accelerator imports the Studio project to the existing Accelerator project. The XLSX file is updated with the edited configuration.
7. In Accelerator, edit and save the resulting XLSX file.
8. Repeat this procedure for additional round trips.

   Note: The Studio has richer functionality than Accelerator. If you introduce Studio features that are not supported in Accelerator, such as global variables, they might be invisible in the Accelerator display. Nevertheless, you can use Accelerator to edit and test the mapper. Accelerator runs the complete mapper logic including the invisible features. The mapper behavior is identical to that in the Studio. If you export the mapper back to the Studio, the features are preserved and become visible again.

Deploying a Mapper as a Data Transformation Service

From Accelerator, you can deploy a mapper as a service that runs in Data Transformation Engine. Alternatively, you can export the mapper to Data Transformation Studio, and deploy it from the Studio.

To deploy a mapper:

1. In Accelerator, create or edit a mapper.
2. On the Cover worksheet, enter a Project Name.

   The name is used as the service name.
3. Save the project as an XLSX file.
4. On the Informatica tab, click Deploy DT Service.

CHAPTER 5

Specifying Mappers

This chapter includes the following topics:

- Mapper Specification Overview, 26
- Fundamental Procedures, 28
- Using Expressions to Manipulate Data, 38
- Using Conditions, 44
- Using Groups, 45
- Selecting Occurrences of Elements, 48
- Reusable Mappers, 51
- Exporting and Importing Reusable Entities, 53
- Chaining a Mapper with other Transformations, 53

Mapper Specification Overview

To specify and implement a mapper, perform the following steps:

1. Import source and target schemas.
2. Import sample source data.
3. Define mappings of source nodes to target nodes.
4. Depending on the required mapper logic, you can include the following optional features:
   - Expressions that manipulate the mapped data.
   - Conditions and alternatives that control the logic.
   - Groups of mappings that must succeed or fail together.
   - Reusable mappers and other reusable entities.
   - Selection logic for identifying the occurrences of iterated source and target elements.
5. Run and test the mapper.

This chapter explains the features and detailed procedures to perform each step.

Informatica Tab

The procedures and commands described in this chapter appear on the Informatica tab. The following table summarizes these options. Some of the options are active only under particular circumstances. For example, certain Edit options are active only if you expose a particular worksheet.

Group	Button	Description
Schema	Import	Imports source and target XML schemas (XSD files) to the Accelerator project.
	Remove	Removes an XML schema from the Accelerator project.
Sample Data	Import	Imports a sample XML source data file to the Accelerator project.
	Remove	Removes sample source data from the Accelerator project.
	Show	Displays sample source data in the default XML editor.
Edit	Undo	Undoes the last action.
	Redo	Redoes the last undone action.
	Function	Defines or edits a function in the selected row.
	Condition	Defines or edits a condition in the selected row.
	Expression	Defines or edits an expression in the selected row.
View	Advanced Mode	Displays additional columns and mapper configurations on the worksheets.
	Reusable Mappers	Exposes or hides the Reusable Mappers worksheet.
	Functions	Exposes or hides the Function Definitions worksheet.
	Keys	Exposes or hides the Keys worksheet.
	Aliases	Exposes or hides the Aliases worksheet.
Project	Run	Runs the mapper on the imported sample data. To run on other data, click the arrow below the button, and click Run with External Sample.
	Test Drive	Runs a mapper in Test Drive diagnostic mode on the imported sample data. To run on other data, click the arrow below the button, and click Test Drive with External Sample.
	Open in DT Studio	Exports the mapper to a Data Transformation Studio project and opens it in the Studio.
	Import DT Project	Imports a project from Data Transformation Studio.
	Deploy DT Service	Deploys the mapper as a service that can run in Data Transformation Engine.
Help	Help	Displays a Getting Started movie, online help, and Accelerator version information.
Advanced	Export Workbook Content	Saves the Accelerator project as a DTA file.

Group	Button	Description
	Import Workbook Content	Imports an Accelerator project from an existing DTA file. This button is displayed only in a new, empty workbook.
	Reusable Entities	Imports or exports reusable entities that you can share with other Accelerator projects.

Fundamental Procedures

This section describes the fundamental procedures to specify a mapper. These procedures are required in nearly every mapper project.

Schemas

Accelerator requires XML schemas that define the source and target XML structures. You can use any XML schema editor to prepare the schemas.

Note: The schemas must comply with the XML Schema recommendation, published by the World Wide Web Consortium, http://www.w3.org. The recommendation is sometimes called XSD, standing for the older name XML Schema Definition. For more information about the schema syntax, see http://www.w3schools.com, For more information about how Data Transformation and Data Transformation Accelerator use the schemas, see the *Data Transformation Studio User Guide*.

Importing Schemas

To specify a mapper, you must import the source and target schemas.

To import a schema:

1. In the Schema group of the Informatica tab, click Import.
2. Enter the path of a schema file, or click the browse button and select the schema.

   You can select and import multiple schemas in a single operation by using the CTRL key.
3. Click OK.

   Accelerator displays a message when the schema has been imported. A graphical representation of the schema appears in the Sources and Targets displays of the Main Mapping worksheet.

Source and Target Schemas

You make the selection of source and target when you specify a mapping between them.

You can map between nodes of the same schema, provided one is defined as the source, and the other as the target. You cannot, however, map from a root element to itself.

By default, Accelerator displays all imported schemas as potential Sources and Targets. You can simplify the Sources and Targets displays by hiding schemas or schema branches that do not participate in the mapping.

When you import sample source data, Accelerator detects the source schema to which it belongs, and it hides other schemas from the Sources display. You can redisplay them if required.

To show or hide schema branches:

1. In the Sources display, right-click the root of the source schema or branch, and click Set As Root.

   Other schemas and branches are hidden.

2. In the Targets display, right-click the root of the target schema or branch, and click Set As Root.

   Other schemas and branches are hidden.

3. To redisplay all branches, right-click the root node and click Show All.

Namespace Support

A schema can define a target namespace. To prevent conflicts between schemas, you can add a default namespace to each schema before you import it. Accelerator detects namespace conflicts, generating unique namespace aliases to prevent conflicts.

Schema Validation

Accelerator assumes that the schema syntax is valid. If a schema is not valid, Accelerator might give unexpected results.

Aliases for XPaths

Accelerator uses XPath syntax to identify schema nodes. The XPaths are displayed in many locations on the worksheets. For example, the advanced mode of the Main Mapping worksheet displays the source and target nodes as XPaths. If you insert expressions or conditions in a worksheet, the Expression and Condition columns display XPaths.

To simplify the display, you can define aliases for XPaths. For example, you can define the alias `PName` for the XPath `Person/Name`. In any location where you can enter `Person/Name`, you can enter `$PName` instead.

Aliases can help shorten long XPaths and hide path details. For example, a mapper might require the following XPaths:

```
Transaction/Message/Segment101/Person/Status/Family/Children/@number
Transaction/Message/Segment101/Person/Status/Family/Children/Child/@gender
```

If you define `Children` as an alias for `Transaction/Message/Segment101/Person/Status/Family/Children`, you can shorten the XPaths to the following values:

```
$Children/@number
$Children/Child/@gender
```

After you define an alias for a node, Accelerator displays the alias rather than the full XPath in every location where the node appears. You can use the Aliases option on the Cover worksheet to alter this behavior and display the full XPath.

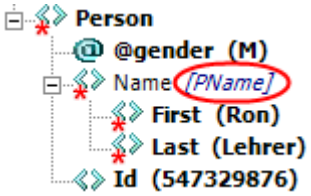
Defining an Alias in the Schema Display

To define an alias:

1. Right-click a node, and click Define alias.

2. Enter the alias, not including the $ symbol.

   The schema displays the alias.



Defining an Alias on the Aliases Worksheet

To define an alias:

1. On the Informatica tab, click Aliases.

   This adds an Aliases worksheet to the project.
2. Drag a node from a schema to the XPath column of the worksheet.
3. In the Name column, enter the alias, not including the $ symbol.

Using an Alias

Except in the schema displays, Accelerator prefixes aliases with a $ symbol. If you type an alias name in a cell, you must type the $ symbol.

Sample Data

You can import a sample XML data file to help configure and test a mapper. The sample data must conform to one of the imported schemas.

An Accelerator project can contain a single set of sample data. If you import sample data again, you overwrite the old data.

To import sample data:

1. In the Sample Data group of the Informatica tab, click Import.
2. Enter the path of an XML file conforming to one of the schemas, or click the browse button and select the file.
3. Click OK.

   Accelerator displays the sample data in the source schema. To display the sample data in the default XML viewer, on the Informatica tab, click Show.

Mappings

You must map the root nodes of the source and target and the nodes that contain the data. In certain cases, you must map other nodes.

Mapping Root Nodes

Accelerator must identify the top-level elements of the XML branches that you want to map. Map the root element of the source to the root element of the target.

If you forget to map the roots, Accelerator auto-generates a root mapping. Examine the auto-generated row and confirm that it is what you intended.

Mapping Data Nodes

To define a transformation from specific source data to specific target data, map the nodes that contain the data.

Mapping Other Nodes

You might need to map multiple-occurrence nodes, such as the parents of recurring branches. For more information, see "Selecting Occurrences of Elements" on page 48.

To define a mapping:

1. Display the Main Mapping worksheet.

   If the worksheet does not exist, you must import source and target schemas.
2. Expand the Sources and Targets schema displays.
3. Drag a source element or attribute to a target element or attribute.

   You can drag the top-level, branch, or leaf nodes.

   The Main Mapping worksheet displays a row for each mapping. The worksheet might also display auto-generated rows for parent elements of the ones that you have mapped.
4. Continue dragging elements or attributes, until you define all the required mappings.

   You can also drag from the source or target schema to the appropriate cell of the worksheet.

   You can type a mapping directly into the worksheet, provided that you use the appropriate XPath syntax.

Automap

Instead of defining explicit mappings between individual leaf nodes, you can map entire branches of the XML tree. Automap identifies matching nodes in the source and target branches. If it succeeds, it generates the leaf mapping automatically.

Use Automap when the source and target schemas contain branches that are identical or contain minor differences. For example, the target branch might be a subset of the source branch, or the source and target might have the same elements but different attributes.

To generate an Automap:

1. On the Informatica tab, select Advanced Mode.
2. Drag the parent element of a source branch to the parent element of a target branch.

   The Main Mapping worksheet displays a row for the parent mapping.
3. In the Automap column of the parent mapping, select Yes.
4. To run the mapper, on the Informatica tab, click Run.

   The Main Mapping worksheet displays the auto-generated mappings.
5. Review the generated mappings.

   If Automap failed to identify a mapping, you can create it explicitly by dragging the leaf nodes.

Editing Operations

You can insert, delete, cut, copy, or paste rows of the Main Mapping worksheet or of other Accelerator worksheets. Perform these operations by right-clicking the rows and choosing commands from the pop-up menu.

You can undo and redo some, but not all, Accelerator editing operations.

Note: Accelerator disables some Excel commands such as the commands on the Home tab to insert or delete rows. To delete rows, you cannot press the Delete key. Use the procedures described in this section instead.

Inserting a Row

To insert a mapping row:

1. Right-click the row below the one that you want to insert, and click Insert Row.
2. Drag the source and target to the new row.

Deleting Rows

To delete rows:

1. Select the rows.
2. Right-click and click Delete rows.

Copying Rows

To copy rows:

1. Select the rows.
2. Right-click and click Copy.
3. Select the row at the top of the new location, right-click, and click Paste. Accelerator overwrites the existing content with the copies content.

   Alternatively, select the row below the new location, right-click, and click Insert Copied Rows. Accelerator inserts new rows containing the copied content. It does not overwrite the existing content.

Undoing or Redoing an Operation

To undo or redo an operation:

1. To undo, click Informatica, click Undo on the Informatica tab, or press ALT + Z.
2. To redo, click Informatica, click Redo on the Informatica tab, or press ALT + Y.

Sequence of Mappings

By default, the Main Mapping worksheet displays the mappings in the sequence that you define them. The sequence is for display purposes only. It has no effect on the mapper output.

To help organize the display, you can insert a blank row and define a mapping in it. You can copy and paste rows to re-arrange them. For more information, see "Editing Operations" on page 31.

Navigating

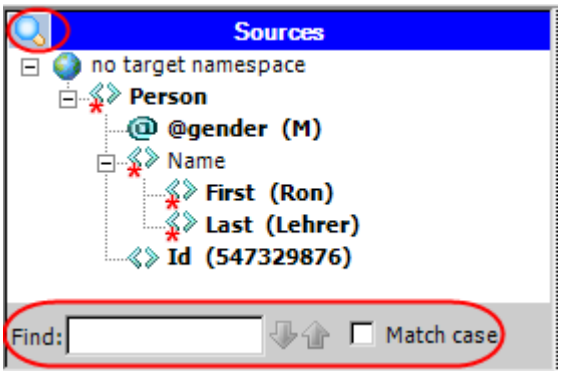
A row in the Main Mapping worksheet can contain a source node, a target node, and expressions that reference other nodes. You can navigate from the row to the associated nodes in the schema trees. Conversely, you can navigate from the schema trees to the mapping rows.

Searching for a Schema Node

To search the schemas:

1. To display the search options, click the icon at the upper left of the Sources or Targets display.

   To hide the search options, click the icon again.



2. Enter the text to find.

   You can enter all or part of a node name. For example, `ge` finds a node called `gender`.

3. For a case-sensitive search, select Match Case.
4. For a forward search, click the Find Next arrow. For a backward search, click the Find Previous arrow.

Navigating from Mapping Rows to the Schemas

To navigate:

1. To locate all nodes associated with a row in the Main Mapping Sheet, click the row.

   In the Sources and Targets displays, source and target nodes are highlighted in dark blue. Other participating nodes, such as those used in expressions, are highlighted in light blue.

2. To locate a specific node, right-click the row and click Find Node. Select the node from the list that appears.

Navigating from a Schema to the Mapping Rows

To navigate:

1. In the Sources or Targets display, right-click the node and click Next Row.

   Accelerator highlights the next mapping row where the node is used.

2. To find additional rows where the node is used, click Next Row again.
3. To search backwards, click Previous Row.

Advanced Mode

You can display the Main Mapping worksheet in basic mode or advanced mode. The basic mode displays the most frequently used columns and right-click menu commands. The advanced mode displays additional columns, details, and menu commands.

For example, in basic mode, the worksheet displays Source Name and Target Name columns:

Source Name	Default Value	Expression	Target Name
Person			Person1
First			First
Last			Last
@gender			Gender
Id			@Id

In Advanced Mode, the worksheet displays the XPath definition and additional information about the source and target:

Source			
XPath	Element Name	Cardinality	Default Value
/	Person	1:1	
/Person/Name	First	1:1	
/Person/Name	Last	1:1	
/Person	@gender	0:1	
/Person	Id	0:1	

For more information about the columns displayed in advanced mode, see "Main Mapping Worksheet" on page 58.

Note: Accelerator hides a few columns in both Basic and Advanced Mode. These columns are for internal use. Do not edit them.

To switch to Advanced Mode:

1. On the Informatica tab, select Advanced Mode.
2. To return to the basic mode, deselect Advanced Mode.

Iterated Mappings

The source and target schemas can permit elements to occur multiple times. By default, if you map multiple-occurrence elements, Accelerator iterates over the occurrences. It creates a target for each instance of the source.

For example, if the source is a class listing, it might contain multiple `Student` elements:

```
<Class>
  <Student>Robert</Student>
  <Student>Juanita</Student>
  <Student>Chang</Student>
  ...
</Class>
```

If you map the `Student` element to a multiple-occurrence target element, Accelerator maps each of the students.

You can control and modify the iteration behavior. For more information, see "Using Groups to Control Iterations" on page 47 and "Selecting Occurrences of Elements" on page 48.

Running a Mapper

For testing purposes, you can run a mapper in Accelerator. Accelerator generates the required Data Transformation files, such as a TGP script file, in a temporary storage location.

Alternatively, you can export the mapper to Data Transformation Studio or deploy it to Data Transformation Engine. For more information about these approaches, see Chapter 4, “Accelerator Projects” on page 23.

To run a mapper in Accelerator:

- To run the mapper with the currently imported sample data, display the Informatica tab and click Run.

  To run the mapper with other sample data, click the arrow under the Run button, click Run with External Sample, and browse to the sample data file.

  Accelerator displays the mapper output in the default XML viewer. It adds sample output data to the Targets schema display.

Test Driving a Mapper

When you run a complex mapper, it is sometimes not obvious how Accelerator processed a particular source node or generated a particular target. You can use the test-drive mode to trace and understand the processing.

If a mapper does not generate the expected results, the test-drive mode can help you to find configuration errors. If a mapper appears to operate correctly, the mode can help you to verify the transformation logic.

To run a mapper in test-drive mode:

1. Display the Main Mapping worksheet.
2. To run the mapper with the currently imported sample data, display the Informatica tab and click Test Drive.

   To run the mapper with other sample data, click the arrow under the Test Drive button, click Test Drive with External Sample, and browse to the sample data file.

   Accelerator runs the mapper. In the Document Actions pane, the Sources and Targets switch from a schema view to a view of the input and output XML. In addition, the pane displays a Test Drive Context area containing diagnostic messages.

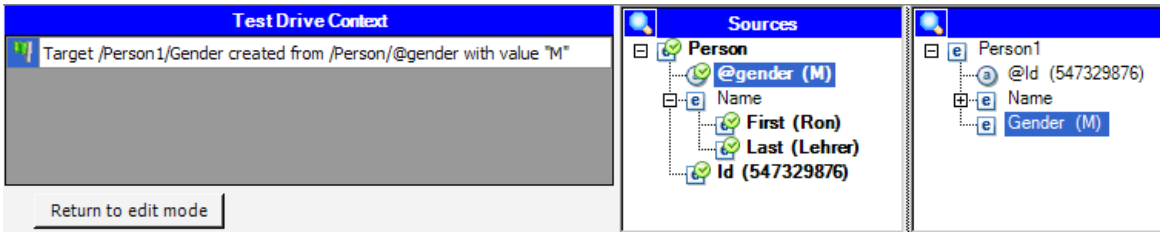
Using the Test Drive Context

The Test Drive Context area displays the details of how the mapper transforms a selected source node. There is a message for each context in which a source node participates.

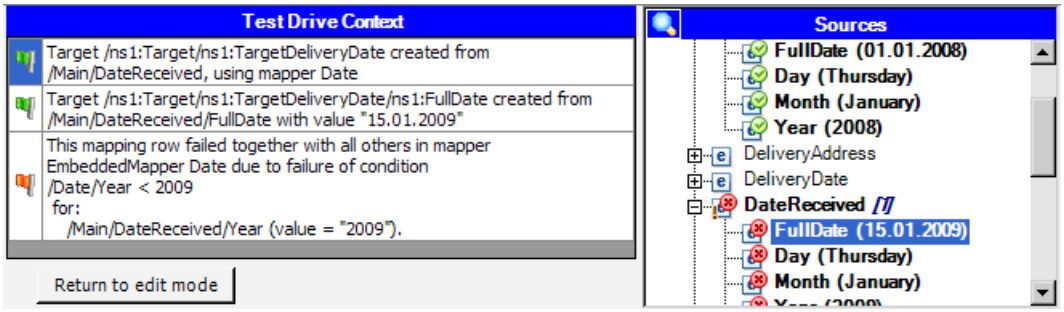
For example, a mapper might copy a source node called `Person/@gender` to a target node called `Person1/Gender`. In the sample data, the attribute has the value `M`.

If you select the source node, `Person/@gender`, Test Drive Context displays the context of the mapping row. The display includes the following message:

```
Target /Person1/Gender created from /Person/@gender with value "M"
```



In another example, if you select a node that participates in a condition, Test Drive Context displays a context message about the success or failure of the condition. If the condition succeeded, Test Drive Context displays another context message about the mapping result.



Test Drive Context labels each context with one of the following status symbols:

Symbol	Description	Meaning
	Success	Mapping succeeded.
	Failure	Mapping failed.

A failure does not necessarily imply an error in the mapper. The failure might be deliberate. For example, the data might not conform to a specified condition. Read the context message for information about the cause of the failure, and evaluate whether the mapper logic is correct.

To provide further information about the status, the Sources XML uses the following symbols:

Symbol	Description	Meaning
Last	Bold font	The node participates in at least one mapping row.
Name	Normal font	The node does not participate in any mapping row.
(Lehrer)	Parentheses	Input or output data
	Success	All mapping rows in which the node participates succeeded.
	Failure	At least one mapping row in which the node participates failed.
	Unreached	The node participates in at least one mapping row, but the mapper logic did not execute any such row. For example, the row belongs to a conditional group, and the condition failed.
	Nested failure	Expand the branch to view a nested node that has a Failure status.

The symbols can appear in combination. For example, a single node can display a success symbol and a nested-failure symbol. The symbols mean that the node was successfully mapped, but that a nested node failed.

Navigating in Test-Drive Mode

Accelerator synchronizes the view in the Sources, the Test Drive Context, and the worksheet. The synchronization enables you to navigate in many different ways:

- To display the contexts of a Sources node, select or right-click the node.
- To display the contexts of a mapping row, select or right-click the row.
- To highlight the node and mapping row corresponding to a particular context, select the context.

In each case, the selection jumps to the corresponding lines in all the displays. If a mapping is defined on the Reusable Mappers worksheet, the display jumps from the Main Mapping worksheet to Reusable Mappers.

The following table explains the navigation operations in greater detail:

Operation	Result in Sources	Result in Worksheet	Result in Test Drive Context
Select a Sources node.		Highlights the first row in which the node participates.	Displays the context of the Sources node and the highlighted row.
Right-click a Sources node, and click Next Mapping Row or Previous Mapping row.		Highlights the next or previous row in which the node participates.	Displays the context of the Sources node and the highlighted row.
Right-click a Sources node, and click Show Context.			Displays the contexts of all rows in which the node participates.
Select a mapping row.	Highlights the first node that participates in the row.		Displays the context of the first row execution.
Right-click a mapping row, and click Next Source Node or Previous Source Node.	Highlights the next or previous node that participates in the row.		Displays the context of the first row execution.
Right-click a mapping row, and click Show Context.	Highlights all nodes that participate in the row.		Displays the contexts of all row executions.
If Test Drive Context displays multiple contexts, select one of them.	Highlights the corresponding nodes.	Highlights the corresponding row.	

Searching the XML

You can search the Sources or the Targets for an element, attribute, or value.

To search the XML:

1. To display the search options, click the icon at the upper left of the Sources or Targets display.

   To hide the search options, click the icon again.
2. Enter the text to find.

In the Find Name box, you can enter an element name or an attribute name preceded by @. For example, `ID` finds an element called `ID`. `@ID` finds an attribute called `ID`.

In the Find Value box, you can enter the value of a node. For example, `RL1` finds any node that has the value `RL1`.

If you enter both a name and a value, Accelerator searches for the combination. For example, you can search for the name `@ID` with the value `RL1`.

3. For a case-sensitive search, select Match Case.
4. For a forward search, click the Find Next arrow. For a backward search, click the Find Previous arrow.

Returning to the Editing Mode

In the Test-Drive mode, you can view the mapper configuration, but you cannot edit it. To close the Test-Drive mode, click Return to Edit Mode.

Using Expressions to Manipulate Data

Accelerator provides an expression editor that you can use to manipulate the mapped data. Expressions can perform the following types of operations:

- Math functions, such as computing totals
- String functions, such as concatenation
- Logical functions, such as changing the output if a condition is satisfied
- Data type conversion functions, such as changing numbers to text

The following examples illustrate some expressions that you might define:

- Concatenate the source `FirstName` and `LastName` elements, and map the result to the target `FullName` element.
- If the source `Age` element is > 18, insert the word `adult` in the target. Otherwise, insert the word `child` in the target.

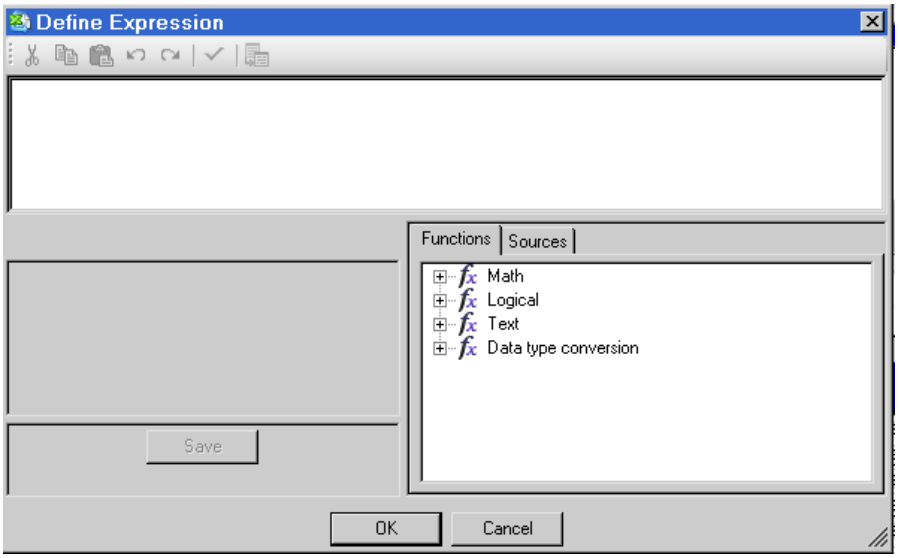
You can view a complete list of the supported functions in the Define Expression window.

Defining an Expression

To define an expression:

1. In the Main Mapping worksheet, select an empty row, or clear the content of the Source Name in an existing row.
2. In the same row, select the Expression cell.
3. On the Informatica tab, click Expression.

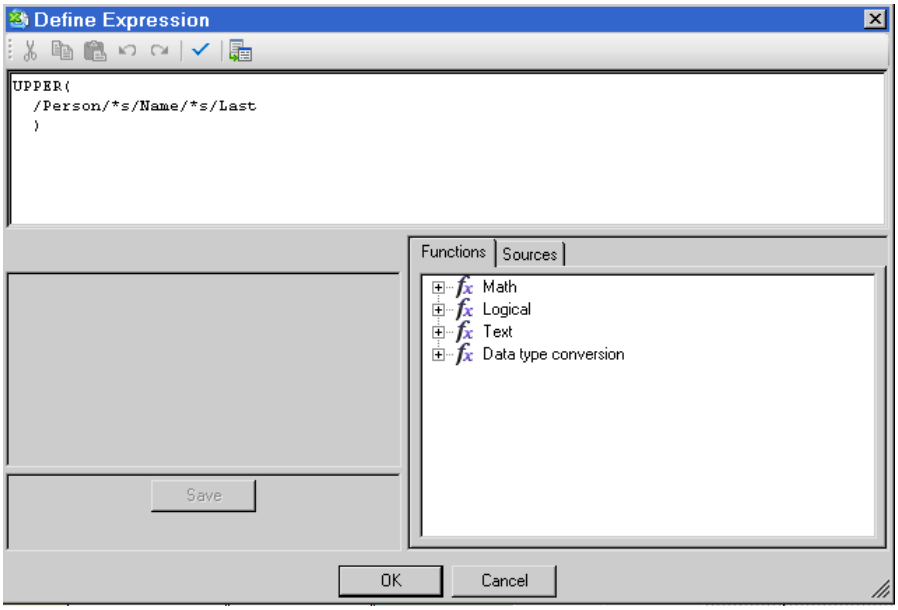
Accelerator displays the Define Expression window.



4. To define the required transformation:
   - Select a function from the list.
   - Fill in the parameters of the function by typing the necessary values or by clicking the Sources tab and dragging an element or attribute to the appropriate location.

   You can type literal values, for example:

   - Numbers: `0`, `10`.
   - Literal strings in quotes: `"dog"`, `"cat"`. If you type strings in text boxes, do not type the quotes. Accelerator adds the quotes.
   - Boolean values: `true`, `false`.



5. Click Save.
6. To display a complex expression clearly, click the Format Expression button.

For example, the button formats `ABS(FLOOR(SQRT(64)))` in the following way:

```
ABS (
  FLOOR (
    SQRT (
      64
    )
  )
)
```

7. Validate the expression by clicking the check mark.
8. Click OK.

   On the Main Mapping worksheet, the Expression cell displays the expression you have defined.

Date and Time Functions

Expressions can contain functions that operate on dates and times. The functions require format parameters that conform to the ICU conventions. For more information about ICU, see:

http://icu.sourceforge.net/apiref/icu4c/classSimpleDateFormat.htm

For example, the following expression converts a `Birthday` element from `15/03/2005` to `03/15/05`:

```
FORMAT_DATE(Person/Birthday, "dd/MM/yyyy", "MM/dd/yy")
```

The format parameters are `"dd/MM/yyyy"` and `"MM/dd/yy"`.

The following table lists the pattern symbols that you can use in format parameters:

Pattern Symbol	Meaning	Type	Example of Data that Matches the Pattern
G	Era designator	Text	AD
y	Year	Number	1996
u	Extended year	Number	-200, meaning 201 BC
M	Month in year	Text or number	July 07
d	Day in month	Number	10
h	Hour in AM/PM (1-12)	Number	12
H	Hour in day (0-23)	Number	0
m	Minute in hour	Number	30
s	Second in minute	Number	55
S	Fractional second	Number	978
E	Day of week	Text	Tuesday
e	Day of week (local 1-7)	Number	2

Pattern Symbol	Meaning	Type	Example of Data that Matches the Pattern
D	Day in year	Number	189
F	Day of week in month	Number	2, used in expressions such as the 2nd Wednesday in July
w	Week in year	Number	27
W	Week in month	Number	2
a	AM/PM marker	Text	PM
k	Hour in day (1-24)	Number	24
K	Hour in AM/PM (0-11)	Number	0
z	Time zone	Time	Pacific Standard Time
Z	Time zone (RFC 822)	Number	-0800
v	Time zone (generic)	Text	Pacific Time
g	Julian day	Number	2451334
A	Milliseconds in day	Number	69540000
' '	The text within single quotes is interpreted as a literal string	Text	'Today is 'dd/MM/yyyy generates output such as Today is 15/03/2005
''	Literal single quote	Text	'o''clock' generates the output o'clock

The count of pattern symbols further determines the format:

- **Text.** If the pattern contains four or more symbols, the data uses the full form. If the pattern contains fewer than four symbols, the data uses a short or abbreviated form. For example, if EEEE produces Monday, EEE produces Mon.
- **Numbers.** The number of pattern symbols is the minimum number of digits. Shorter numbers are padded with zeroes. For example, if m produces 6, mm produces 06.
- **Years.** The two-digit year is yy, and the four-digit year is yyyy. For example, if yy produces 05, yyyy produces 2005.
- **Months.** If M produces 1, then MM produces 01, MMM produces Jan, and MMMM produces January.

All non-alphabetic characters are interpreted as literals, even if they are not enclosed in single quotes. For example, dd/MM/yyyy HH:mm produces 15/03/2005 13:15, containing the /, space, and: characters.

User-Defined Functions

Use the Function Definitions worksheet to add custom functions to the Define Expression window.

To define a function:

1. In the View group of the Informatica tab, select Functions.

   This exposes the Function Definitions worksheet.

2. In the Edit group of the Informatica tab, click Function.

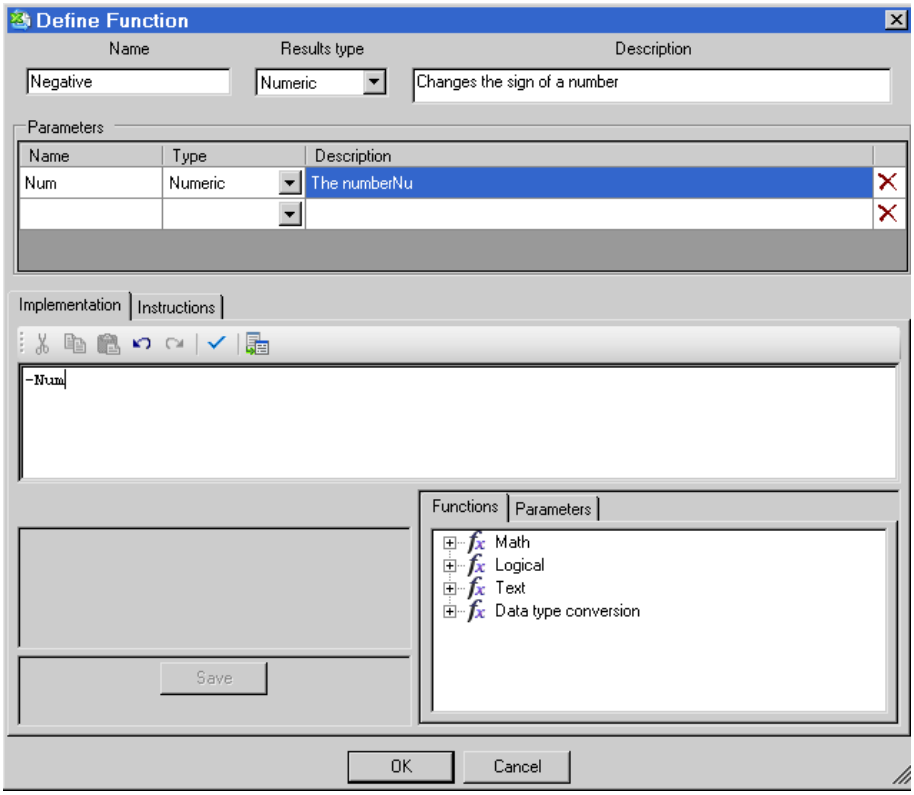
   The Define Function window appears.

3. Enter the function information:

   - **Name.** The name of the function.
   - **Results Type.** Select the data type that the function should return. To enable multiple output data types, set the type to Any.
   - **Description.** A comment describing the function.
   - **Parameters.** Enter the name, data type, and description of each input parameter. To enable multiple input data types, set the type to Any.

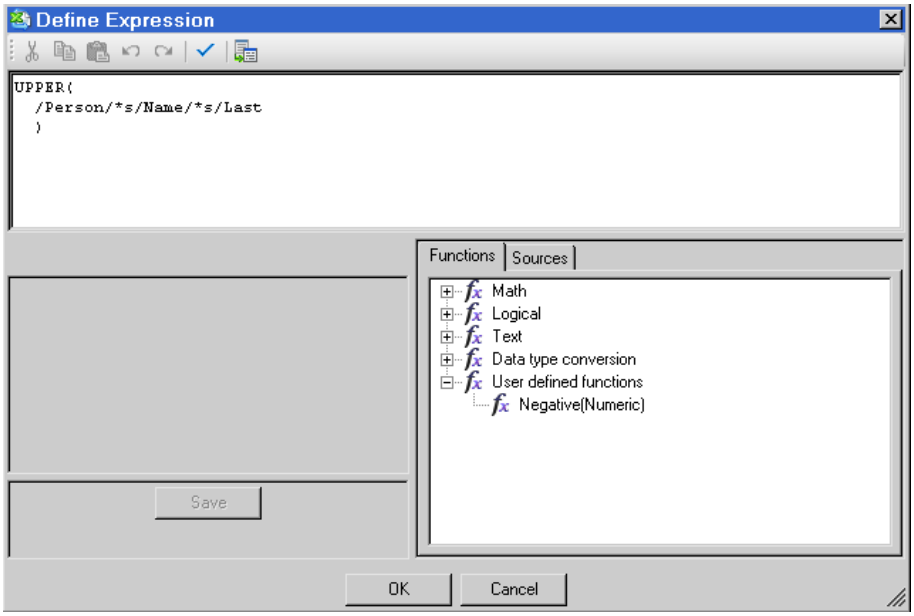
4. On the Implementation tab, enter the code that the function executes. You can use the same editing techniques as in the expression editor. In addition, you can drag parameters from the Parameters list to the Implementation tab.

   The following example illustrates a function that returns the negative of a number:



5. On the Instructions tab, you can enter additional comments or instructions on how to use the function.

6. When you use the expression editor, the user-defined function is available. You can use it in the same way as the built-in functions.



Lookup Functions

You can define a lookup table that contains input and output values. A mapper can use the lookup function to find the output that corresponds to a given input.

For example, you might define the following lookup table:

Input Value	Output Value
`M`	`Male`
`F`	`Female`

The mapper can look up `M` and `F` values that exist in the source data, and write `Male` and `Female` in the target.

To define a lookup table:

1. In the View group of the Informatica tab, select Lookups.

   This exposes the Lookups worksheet.

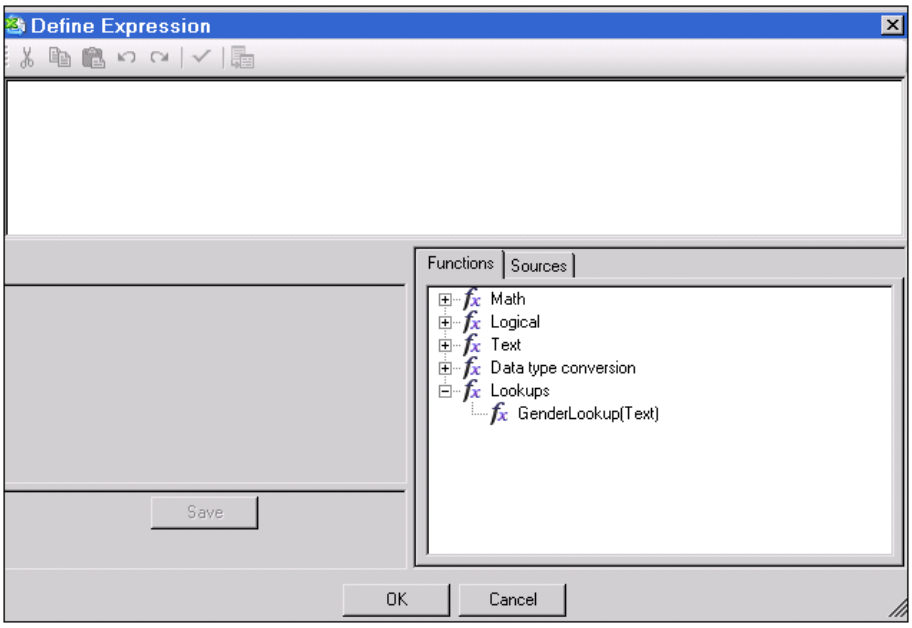
2. On each row of the table, enter an Input Value that can exist in the source data, and the corresponding Output Value that you want the function to return.

3. In the Name column, enter the same name in every row of the table.

   To define multiple lookup tables, use a different name for the rows of each table.

B	C	D
Name	Input value	Output value
GenderLookup	M	Male
GenderLookup	F	Female

4. When you use the expression editor, the lookup function is available. You can use it in the same way as the built-in functions.



Using Conditions

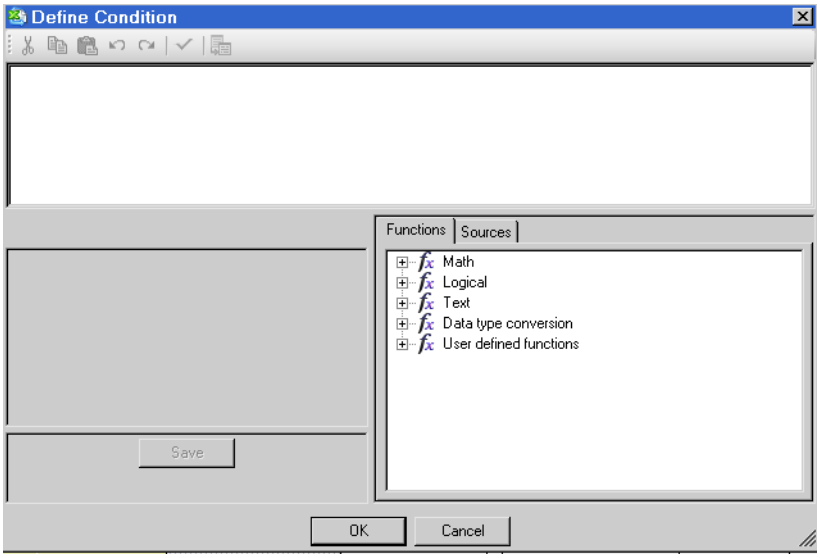
A condition is a logical test that controls a mapping. For example, you might map a source element to a target element only if the source is longer than three characters. Otherwise, the target is left empty.

For more information about ways to use conditions, see "Using Groups and Conditions to Define Alternative Mappings" on page 46.

To define a condition:

1. On the Informatica tab, select Advanced Mode.
2. Select the Condition cell of the row where you want to apply a condition.
3. On the Informatica tab, click Condition.

The Define Condition window appears.



4. Edit the condition.

   The editing procedure is the same as for expressions. For more information, see "Using Expressions to Manipulate Data" on page 38.

5. Click OK.

   The Condition cell displays the condition that you defined.

Using Groups

A group is a set of mappings that succeed or fail together.

To define a group, assign the Group Name column of each row. All rows with the same name are members of the group. If a mandatory row fails, the mapper does not execute any rows of the group.

For example, a source schema defines `First` and `Last` elements, storing first and last names. If a source document contains both `First` and `Last`, you want to map the elements. If either element is missing, you want Accelerator to ignore both mappings.

To do this, define the `First` and `Last` mappings as mandatory, and nest them in a group. If either `First` or `Last` does not exist, the group fails, and both mappings are not executed. Other mappings, not nested within the group, can still succeed.

You can nest groups. You can use groups to define alternatives and to control the iterations of a mapping.

To define a group:

1. On the Informatica tab, select Advanced Mode.
2. In each row of the group, enter the Group Name.

   In the above example, you might name the group `NamesMustExist`.

3. In the Mandatory cell of each row, select one of the following values:
   - Yes. If the row fails, it causes the entire group to fail.

- No. If the row fails, it does not cause the group to fail.
- As In Schema. Accelerator defines the row as mandatory if the source schema defines the node as mandatory.

Grouping	Mandatory	Source		
Group Name	Mandatory	XPath	Element Name	Cardinality
NamesMustExist	Yes	/	Person	1:1
NamesMustExist	No	/Person	@gender	0:1
NamesMustExist	Yes	/Person/Name	First	1:1
NamesMustExist	Yes	/Person/Name	Last	1:1

Nested Groups

To nest a group within another group, add a period followed by a suffix to the parent group name.

For example, if the parent group is called `Group1`, you might call the nested group `Group1.AA`.

You can nest groups to any level. For example, a group called `Group1.AA.BB` is nested within `Group1.AA`.

Using Groups and Conditions to Define Alternative Mappings

You can create multiple, alternative mappings from the source to the target. You can configure conditions that control which alternative is used.

To define the alternatives, create a group. On one row of the group, set the Group Type to Alternatives. Define each alternative mapping as a nested group. Accelerator attempts to execute the nested groups in alphanumeric sequence of their names, until one of them succeeds. If all the nested groups fail, the alternatives mapping fails.

To define alternative mappings:

1. Create a group.
2. Set the Group Type property to Alternatives.
3. Nested within the group, define nested groups.

   Each nested group is an alternative.

Online Sample

The following sample is available in the Accelerator installation directory:

```
Documentation\Tutorial\Alternatives\AlternativeSample.xlsx
```

The source document records telephone numbers in the following way:

```
<PhoneNumber type="home">1234567</PhoneNumber>
<PhoneNumber type="work">7654321</PhoneNumber>
<PhoneNumber type="mobile">4567123</PhoneNumber>
```

The target schema has different element for each type of number:

```
<Home>1234567</Home>
<Work>7654321</Work>
<Mobile>4567123</Mobile>
```

The sample specifies the following, alternative mappings:

- If the `PhoneNumber` element has `type="home"`, map it to `Home`.
- If the `PhoneNumber` has `type ="work"`, map it to `Work`.
- If the `PhoneNumber` has `type="mobile"`, map it to `Mobile`.

The following figure illustrates the configuration of these alternatives:

Grouping		Mandatory	Source ıs & Transf		Target		
Group Name	Group Type	Mandatory	XPath	Condition	XPath	Element Name2	Cardinality
Phones	Alternatives	Yes	/Source		/Target	Phone	1:unbounded
Phones.home		Yes	/Source	/Source/*s/PhoneNumber/@type = "home"	/Target/Phone	Home	1:1
Phones.mobile		Yes	/Source	/Source/*s/PhoneNumber/@type = "mobile"	/Target/Phone	Mobile	1:1
Phones.work		Yes	/Source	/Source/*s/PhoneNumber/@type = "work"	/Target/Phone	Work	1:1

The nested groups are mappings from `PhoneNumber` to `Home`, `Work`, and `Mobile`. Each group has a condition. For example, the condition on the mapping to `Home` is that `@type` has a value of `home`.

When you run the mapper, Accelerator detects that the nested mappings are alternatives. The mapper uses the first nested mapping that succeeds.

Using Groups to Control Iterations

If a group contains a multiple-occurrence source element, you can iterate the mappings over each occurrence. Accelerator executes all rows of the group for each instance of the source element.

To implement the iterations, use a controlled group. Mark the row containing the multiple-occurrence element as the controlling row. The group has the following behavior:

- For each occurrence of the source element in the controlling row, Accelerator iterates all rows of the group.
- If the controlling row defines a target, Accelerator creates a target occurrence for each iteration of the group.
- If the controlling row fails, the iteration fails.

  The row might fail, for example, if the source data is missing or a condition is false. In that case, Accelerator does not execute any rows of the group for the particular iteration. It continues with the next iteration.

To define a controlled group:

1. Create a group.
2. In the row containing the multiple-occurrence source element, set the Group Type to Controlled.

Selecting Occurrences of Elements

In many XML documents, an element occurs multiple times. For example, in the following document, there are multiple occurrences of the `Child` element:

```
<Children>
    <Child name="Eric" hobby="Swimming"/>
    <Child name="Elizabeth" hobby="Biking"/>
    <Child name="Mary" hobby="Painting"/>
    <Child name="Edward" hobby="Swimming"/>
</Children>
```

In such cases, a mapper must select the appropriate occurrences of the source and target elements. You can define the occurrences by number or by key.

Selecting by Occurrence Number

Suppose that the source and the target contain a multiple-occurrence `Child` element. You want to map the third `Child` element of the source to the first `Child` element of the target. You can define this mapping by number.

Online Sample

To view an example of selection by occurrence number, open the following project:

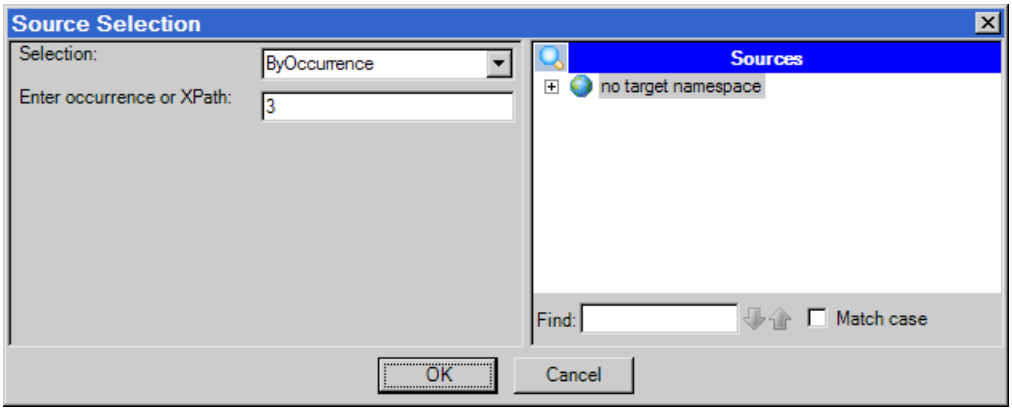
```
Documentation\Tutorial\ByOccurrence\ByOccurrence.xlsx
```

To select by occurrence number:

1. Define a mapping from a source element to a target element.

   Either or both elements can have multiple occurrences.
2. On the Informatica tab, select Advanced Mode.
3. If the source can have multiple occurrences, edit the Source Selection cell.
   - Click the Browse button in the cell.
   - Select ByOccurrence.
   - Enter the occurrence number, or drag an element or attribute that contains the occurrence number.

   For example, to map the third occurrence of the source element, select ByOccurrence and enter 3.



4. If the target element can have multiple occurrences, edit the Target Selection cell in the same way.

   Note: The ByOccurrence setting might affect other mapping rows that have the same source or target element. To prevent this, nest the ByOccurrence row in its own group. For more information, see "Using Groups" on page 45.

Selecting by Key

A `Key` is a set of one or more nodes that serves to uniquely define an occurrence of an element. You can use keys to identify the correct occurrence for each mapping.

Example

A source document contains a list of children and their hobbies:

```
<Children>
    <Child name="Eric" hobby="Swimming"/>
    <Child name="Elizabeth" hobby="Biking"/>
    <Child name="Mary" hobby="Painting"/>
    <Child name="Edward" hobby="Swimming"/>
</Children>
```

The target contains a node for each hobby, together with the names of the children who participate in that hobby:

```
<Hobbies>
  <Hobby name="Swimming">
    <Child name="Eric"/>
    <Child name="Edward"/>
  </Hobby>
  <Hobby name="Biking">
    <Child name="Elizabeth"/>
  </Hobby>
  <Hobby name="Painting">
    <Child name="Mary"/>
  </Hobby>
</Hobbies>
```

The goal is to map the name of each `Child` to the correct `Hobby` node using the name of the hobby as a key in the output document.

You can implement the mapping in the following way:

- In the target structure, define the Hobby/@name attribute as the key for Hobby nodes.
- Identify the Hobby node in which the Child should be inserted, according to the Hobby key.

Consider the following input node:

```
<Child name="Mary" hobby="Painting"/>
```

The input data is inserted into the following output node:

```
<Hobby name="Painting">
    <Child name="Mary"/>
<Hobby>
```

Accelerator uses the Hobby key to identify the output node. In this example, the input contains Child/@hobby = "Painting". Accelerator identifies the output node where Hobby/@name = "Painting".

Online Sample

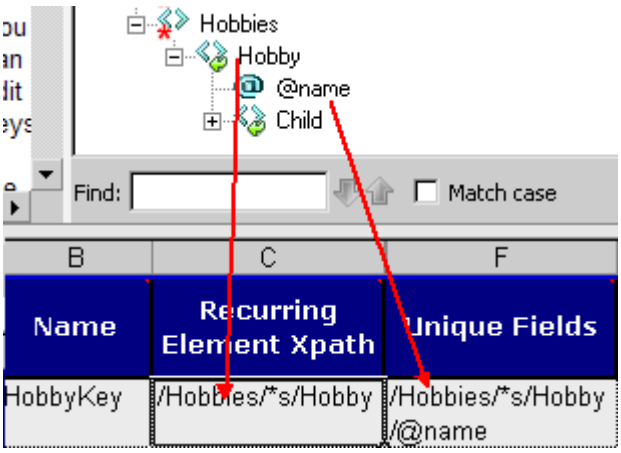
To view an implementation of the `Hobby` example, open the following project:

```
Documentation\Tutorial\ByKey\ByKey.xlsx
```

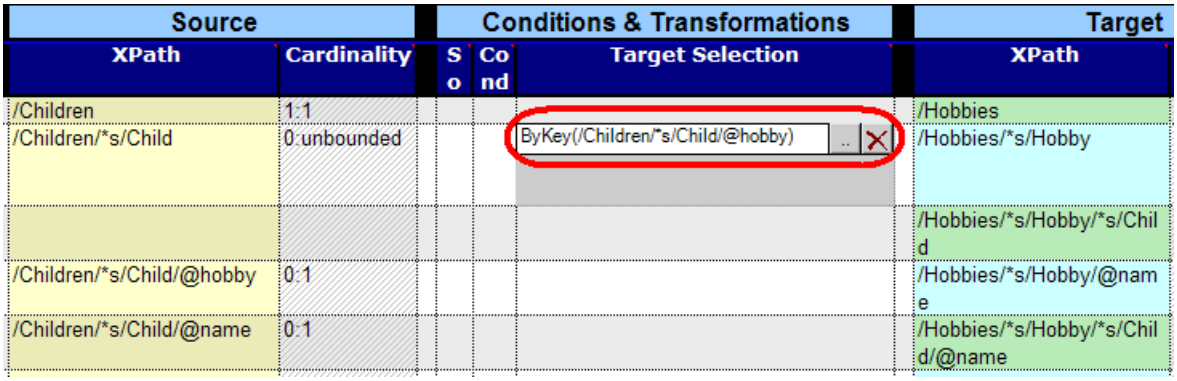
On the Main Mapping worksheet, the nodes are mapped in the following way:

Source	Target	Explanation
`Children/Child`	`Hobbies/Hobby`	Specifies that data in the multiple-occurrence `Child` element is mapped to data in the multiple-occurrence `Hobby` element.
`Children/Child/@name`	`Hobbies/Hobby/Child/@name`	Mapping of specific nodes in the `Child` element to specific nodes in the `Hobby` element.
`Children/Child/@hobby`	`Hobbies/Hobby/@name`	

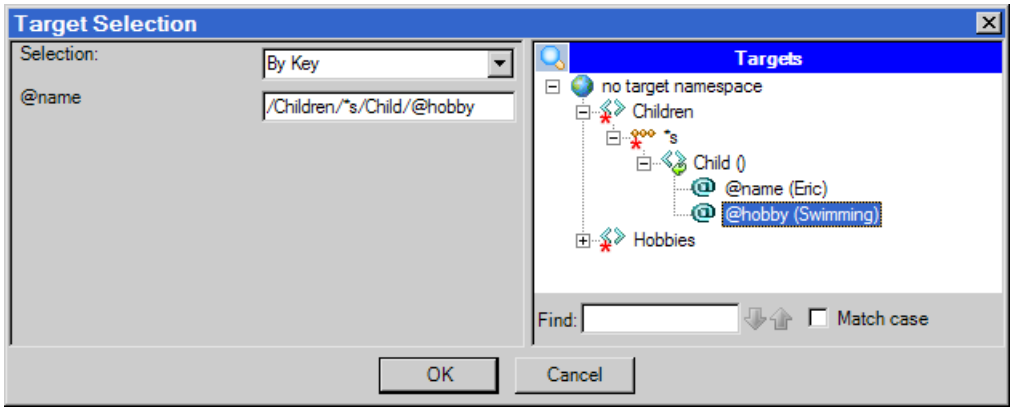
On the Keys worksheet, the Hobby/@name attribute is defined as a Unique Field identifying Hobby nodes. The identification has the name HobbyKey.



The Main Mapping worksheet uses the key to define the target of the `Hobby` mapping:



To display the details of the target selection, click the browse button:



To select by key:

1. Define a mapping from a source element to a target element.

   Either or both elements can have multiple occurrences.
2. On the Informatica tab, select Advanced Mode.
3. On the Informatica tab, select Keys.

   This exposes the Keys worksheet.
4. In the Name column, enter a name for the key.

   In the `Hobby` example, enter `HobbyKey`.
5. From the Document Actions pane, drag the multiple-occurrence element to the Recurring column.

   In the example, drag the `Hobbies/Hobby` element.
6. Drag one or more nodes that identify occurrences to the Unique Fields column.

   In the example, drag `Hobbies/Hobby/@name`.
7. Return to the Main Mapping worksheet, and select the row that maps the multiple-occurrence elements.

   In the example, select the row that maps `Children/Child` to `Hobbies/Hobby`.
8. In the Source Selection or Target Selection column, click the Browse button, select ByKey, and define the node whose value must equal the key.

   In the example, click Target Selection, and define that the `Children/Child/@hobby` node must equal the key.

Reusable Mappers

A reusable mapper is a mapper that can be used multiple times. In a complex project, a reusable mapper can map branches of the source and target schemas. You can insert the reusable mapper in the Main Mapping wherever the branches occur.

For example, a source might contain several `Date` elements, such as a start date, an end date, and milestone dates. All source dates are formatted in the following way:

```
<Date>
  <Day>21</Day>
  <Month>October</Month>
  <Year>2005</Year>
</Date>
```

The target has an analogous set of dates, formatted in the following way:

```
<Date>October 21, 2005</Date>
```

On the Reusable Mappers worksheet, you can define a reusable mapper that transforms a source `Date` to a target `Date`. The mapper contains an expression that concatenates the source `Day`, `Month`, and `Year`.

On the Main Mapping worksheet, you can insert the reusable mapper wherever a date occurs.

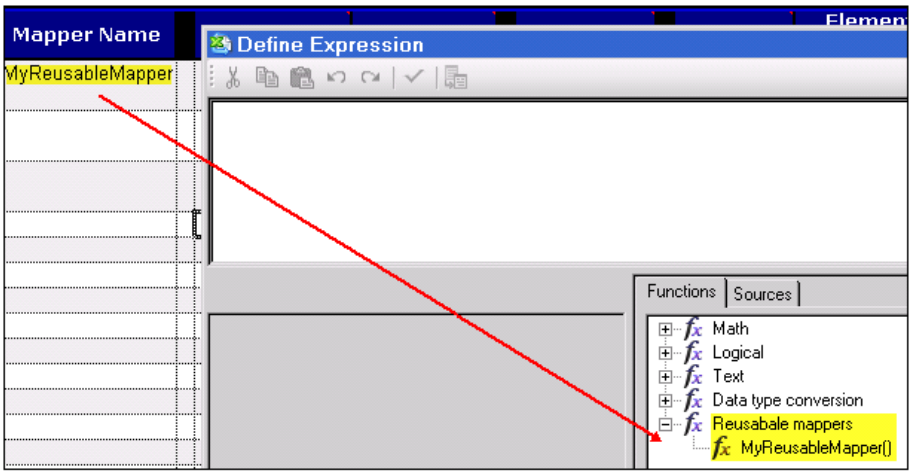
Specifying a Reusable Mapper

To specify a reusable mapper:

1. On the Informatica tab, select Reusable Mappers.
2. On the Reusable Mapper worksheet, specify a mapper.

   Use the same procedures as on the Main Mapping worksheet.
3. In the Mapper Name column, enter the same name in every row of the mapper.

   To define multiple reusable mappers, use a different name for the rows of each mapper.
4. On the Informatica tab, click Expression.
5. Verify that the reusable mapper appears in the function list.



Using a Reusable Mapper

To use a reusable mapper:

1. Display the Main Mapping sheet.
2. Map a branch of the source schema to a branch of the target schema.

   In the above example, map a source `Date` to a target `Date`.
3. Select the Expression column of the mapping row.
4. On the Informatica tab, click Expression.

   The Define Expression window appears.
5. Select the reusable mapper and click OK.

   The Main Mapping sheet displays the name of the reusable mapper. Accelerator uses the reusable mapper to transform the branch that you mapped.

Exporting and Importing Reusable Entities

You can use function definitions, lookups, and reusable mappers repeatedly in the same project. In addition, you can export them from a project and import them to other projects.

The entities are exported and imported by name. An imported entity overwrites an existing entity that has the same name.

Exporting Reusable Entities

To export reusable entities:

1. On the Informatica tab, click Reusable Entities > Export Reusable Entities.
2. Save the entities in an XML file.

Importing Reusable Entities

To import reusable entities:

1. On the Informatica tab, click Reusable Entities > Import Reusable Entities.
2. Open an XML file that you previously exported.

Chaining a Mapper with other Transformations

For testing and development purposes, you can configure an Accelerator project to run a mapper in sequence with pre-mapper and post-mapper transformations. By chaining the appropriate transformations, you can transform non-XML input or generate non-XML output. You can simulate the operation of a production application that chains multiple transformations on a single input.

For example, you can configure an Accelerator project to run the following transformations in sequence:

1. Run a deployed Data Transformation parser service on an input file that has any data format.
2. Run a mapper on the parser output.
3. Run a deployed Data Transformation serializer service on the mapper output, generating a final output file that has any data format.

Configuring a Pre-Mapper Transformation

To configure a pre-mapper transformation:

1. In Data Transformation Studio, define a transformation and deploy it as a Data Transformation service.

   The transformation output must conform to the source schema of the mapper.
2. On the Cover worksheet of the Accelerator project, click the Pre-Mapper Transformation Name, and select the deployed service.
3. To import a sample source file that the pre-mapper transformation can process, in the Sample Data group on the Informatica tab, click Import.

   For example, if the transformation is a parser that operates on text input, import a sample text file.

Configuring a Post-Mapper Transformation

To configure a post-mapper transformation:

1. In Data Transformation Studio, define a transformation and deploy it as a Data Transformation service.

   The transformation input must conform to the target schema of the mapper.
2. On the Cover worksheet of the Accelerator project, click the Post-Mapper Transformation Name, and select the deployed service.
3. On the Cover worksheet, enter the Output File Extension, including the leading period.

   For example, if the transformation is a serializer that generates text output, enter `.txt`.

Effects of Pre- and Post-Mappers on Accelerator Behavior

The pre- and post-mapper transformations affect the Accelerator behavior in the following ways:

- When you import or test the mapper on sample data, you must select an input file that the pre-mapper can process.
- The Show Sample Data command displays the pre-mapper input.
- The Run command displays the post-mapper output.
- The Test Drive command displays the following data:
  - The mapper XML input, which is the pre-mapper output.
  - The mapper XML output.
  - The post-mapper output.

In other respects, the pre- and post-mapper transformations have no effect on the Accelerator behavior:

- You must still import source and target XML schemas.
- The schemas still display XML sample data.

  The source XML is the pre-mapper output. The target XML is the mapper output.
- You must still configure the mapper to transform XML to XML.
- The Run command exports only the mapper. It does not export the pre- and post-mapper transformations.
- The Deploy DT Service command deploys only the mapper. The pre- and post-mapper transformations are already deployed.

APPENDIX A

Properties Reference

This appendix includes the following topics:

- Overview of the Properties Reference, 55
- Cover Worksheet, 55
- Main Mapping Worksheet, 58
- Reusable Mappers Worksheet, 60
- Function Definitions Worksheet, 60
- Keys Worksheet, 61
- Lookups Worksheet, 61
- Aliases Worksheet, 62

Overview of the Properties Reference

This appendix describes the complete set of properties that Accelerator displays on the various worksheets. Each property is a column of a worksheet.

Cover Worksheet

The Cover worksheet displays general information about a project.

Most options are identical to the project properties available in Data Transformation Studio. Only brief explanations are given here. For more information about the project properties, see the *Data Transformation Studio User Guide*.

Table 1. Cover Worksheet Properties

Group	Column Name	Description
General	Project Name	A name for the project, used if you export the project to Data Transformation Studio or deploy to the Engine. A project name must contain only English letters (A-Z, a-z), numerals (0-9), spaces, and the following symbols: `# % & + , - .  = @ [ ] _ { }`
	XPath display option	Defines the way the XPath of a node is displayed in the other worksheets. Select one of the following options: - **XPath only.** The full path of the node is displayed. - **Parent XPath and node name.** The parent Xpath and the node name are displayed in different columns.
	XPath format option	The display format of paths. Select one of the following options: - **Standard XPath.** Choice and sequence indicators are omitted. - **Data Transformation XPath.** Choice (`*c`) and sequence (`*s`) indicators are shown, as in Data Transformation Studio.
	Show aliases	If you define aliases for XPaths, select Yes to display the aliases on the other worksheets. Select No to ignore the aliases and display full XPaths.
	Generate trace log	Select Yes to create a log file in the Data Transformation reports location. The default log location is: `c:\Documents and Settings\<User Name>\Local Settings \Application Data\Informatica\Data Transformation Accelerator\Logs\<Excel File Name>.log`
	CM service output directory	Location where Accelerator deploys Data Transformation services. The default is the Data Transformation repository.
	Instance to execute	The name of the transformation. Do not change the default value, `Main`.
	Pre-mapper transformation name	The name of a deployed Data Transformation service that Accelerator runs before the mapper. The mapper runs on the output of the deployed service.
	Post-mapper transformation name	The name of a deployed Data Transformation service that Accelerator runs on the mapper output.
	Output file extension	If you define a post-mapper transformation, enter the extension of the output file. Include a leading period, for example, `.txt`.
XML Generation	XML output mode	Specifies the transformation behavior if a mapper does not map any data to a node that is defined in the target schema.
	Add default values for required nodes	Instructs a mapper to output default values that are required by the target schema.

Group	Column Name	Description
	Validate added nodes	Causes Data Transformation to validate the nodes that it adds because of the Full or Add Required output mode. If adding the node would invalidate the output XML, Data Transformation does not add it.
	Add XML root node	Applicable when mapping to two or more target schemas. If Yes, Accelerator generates a global root node for all the target schemas.
	Root node name	Name of the global root node.
	Add XML processing instruction	Controls whether Accelerator adds a processing instruction to the XML output.
	XML version	The version of XML that Accelerator adds to the processing instruction.
	XML encoding	The encoding that Accelerator adds to the processing instruction.
	XML custom encoding	If the XML encoding is Custom, enter the encoding name that Accelerator adds to the processing instruction.
	Add custom processing instructions	Controls whether Accelerator adds custom processing instructions to the output.
	Custom processing instructions	The custom processing instructions, including, the `<??>` symbols.
Input encoding	Extract codepage from source	If Yes, Accelerator uses a code page that is specified in the source document.
	Byte order	The byte order of binary data.
	Encoding profile	Options for the input document encoding.
	Custom encoding	The name of a custom encoding.
	Schema encoding	The encoding of special symbols. In the XML encoding, a symbol such as > is represented by `>`.
Working encoding	Encoding profile	Options for the working encoding.
	Custom encoding	The name of a custom encoding.
	Schema encoding	The encoding of special symbols. In the XML encoding, a symbol such as > is represented by `>`.
Output encoding	Byte order	The byte order of binary data.
	Encoding profile	Options for the output document encoding.
	Custom encoding	The name of a custom encoding.
	Schema encoding	The encoding of special symbols. In the XML encoding, a symbol such as > is represented by `>`.

Main Mapping Worksheet

The Main Mapping worksheet defines a mapper.

In basic mode, the Main Mapping worksheet displays the most frequently used columns and menu commands. In Advanced Mode, it exposes additional columns and ribbon commands. For more information about the modes, see "Advanced Mode" on page 33.

Table 2. Main Mapping Worksheet Properties

Group	Column	Shown in mode	Definition
Grouping	Group Name	Advanced	Creates a group of rows. Assign the same name to every row in the group. If one mandatory mapping in the group fails, the entire group fails.
	Group Type	Advanced	If Alternatives, subgroups belonging to this group are treated as alternative mappings. If Controlled, the row controls the behavior of the entire group.
Mandatory	Mandatory	Advanced	If Yes, and the mapping fails, the parent mapper or group fails. If No, the parent does not fail. The default is As In Schema, meaning that the mapping is mandatory if the source schema defines the node as mandatory. This property corresponds to `optional` in the Studio.
Source	Source name	Basic	Name of the source node.
	XPath Element Name	Advanced	The full path of the source node. Depending on the XPath Display Option that you select on the Cover worksheet, XPath and Element Name appear in a single column or in two columns.
	Cardinality	Advanced	The minimum and maximum occurrences of the data node. This property is fixed by the source schema and cannot be edited in Accelerator.
	Default Value	Basic Advanced	A default value for the source node. The default value applies if the node is missing from the source document.
Conditions and Transformations	Source Selection	Advanced	Defines how the mapper selects occurrences of multiple-occurrence source elements. For more information, see "Selecting Occurrences of Elements" on page 48. This property corresponds to a locator in the Studio.
	Condition	Advanced	Defines a logical condition for performing a mapping.
	Expression	Basic Advanced	Defines an expression that can modify the mapped data.
	Target Selection	Advanced	Defines how the mapper selects occurrences of multiple-occurrence target elements. For more information, see "Selecting Occurrences of Elements" on page 48. This property corresponds to a locator in the Studio.

Group	Column	Shown in mode	Definition
Target	Target Name	Basic	Name of the target node.
	XPath Element Name	Advanced	The full path of the target node. Depending on the XPath Display Option that you select on the Cover worksheet, XPath and Element Name appear in a single column or in two columns.
	Cardinality	Advanced	The minimum and maximum occurrences of the data node. This property is fixed by the source schema and cannot be edited in Accelerator.
General	Automap	Advanced	By default, Accelerator does not map nested nodes automatically. To map the nested nodes automatically, set the AutoMap column of the parent node to Yes. Accelerator adds a line to the Main Mapping worksheet for each nested node and labels it as `Auto generated` in the Status column. For more information, see "Automap" on page 31.
	Status	Basic Advanced	Status can have the following values: - **Empty.** There is no special status, and the mapping is active. Empty is the default value. - **Ignore.** Accelerator ignores the mapping. - **Auto deleted.** A mapping that was deleted outside Accelerator. This status is assigned if you export a mapper to the Studio, delete a mapping in the Studio, and re-import the mapper to Accelerator. Accelerator ignores the mapping. If you want to restore the mapping, copy the row content to a new row. - **Auto generated.** A required mapping that Accelerator detected, but you did not define explicitly.
	Remark	Basic Advanced	Comment.
	Error	Basic Advanced	Message displayed by Accelerator if an error occurs when the project is compiled or run.
Summary	Summary	Basic Advanced	Information supplied by Accelerator summarizing the mapping properties.

Reusable Mappers Worksheet

The Reusable Mappers worksheet defines mapper components that you can use multiple times. For more information, see "Reusable Mappers" on page 51.

Table 3. Reusable Mappers Worksheet Properties

Column	Definition
Mapper Name	Name of the reusable mapper.
(All other columns)	Identical to the Main Mapping worksheet. For more information about these columns, see "Main Mapping Worksheet" on page 58.

Function Definitions Worksheet

This worksheet displays user-defined functions. For more information, see "User-Defined Functions" on page 42.

Table 4. Function Definitions Worksheet Properties

Column	Definition
Function Name	Name of the user-defined function.
Function Prototype	The definition of the function and its parameters.
Function Body	Code that implements the function.
Ignore	If Yes, the mapper does not run the function.
Remarks	Comment.
Error	Message displayed by Accelerator if an error occurs when the project is run.

Keys Worksheet

This worksheet is used to define unique keys for identifying multiple-occurrence elements. For more information, see "Selecting by Key" on page 49.

Table 5. Keys Worksheet Properties

Column	Definition
Name	Name of the key.
Recurring Element XPath	The multiple-occurrence element.
Unique Fields	One or more fields that uniquely identify an occurrence of the element.
Ignore	If Yes, the mapper ignores the key.
Remark	Comment.
Error	Message displayed by Accelerator if an error occurs when the project is run.

Lookups Worksheet

This worksheet defines lookup tables that translate between input and output values. For more information, see "Lookup Functions" on page 43.

Table 6. Lookups Worksheet Properties

Column	Definition
Name	Name of the lookup table. Repeat the name in all rows of the same lookup function.
Input Value	Input of the lookup.
Output Value	Output of the lookup.

Aliases Worksheet

This worksheet defines aliases for XPaths. For more information, see "Aliases for XPaths" on page 29.

Table 7. Aliases Worksheet Properties

Column	Definition
Name	Name of the alias.
XPath	The XPath.
Remark	Comment.

INDEX

A

Accelerator
- deploying services 25
- expressions 38
- importing sample data 30
- integration with Studio 24
- introduction 9
- specifying mapping 30
- system requirements 11
- uninstalling 13

advanced mode
- displaying 33

alias
- for XPath 29

Aliases
- worksheet 62

alternatives
- in Accelerator 46

Automap
- mapping similar XML branches 31

B

basic mode
- displaying 33

By Occurrence
- in Accelerator 48

ByKey
- in Accelerator 48

C

chaining
- pre- and post-mapper transformations 53

controlled group
- mapping 47

copying
- rows 31

Cover
- worksheet 55

D

deleting
- rows 31

deploying services
- from Accelerator 25

diagnostics
- mappers 35

E

editing
- rows 31

exporting and importing 53

expression
- Accelerator 38

F

functions
- defining in Accelerator 42

Functions
- worksheet 60

G

groups
- in Accelerator 45
- nested 46

I

Importing
- XML schemas 28

Informatica menu
- options 27

inserting
- rows 31

integration
- Accelerator with Studio 24

iterated mappings
- multiple-occurrence elements 34

iterations
- controlling 47

K

keys
- multiple-occurrence elements 48

Keys
- worksheet 61

L

locators
- in Accelerator 48

lookup functions
- defining in Accelerator 43

lookups
- exporting and importing 53

Lookups
worksheet 61

M

Main Mapping
worksheet 44, 58
mapper
definition 14
occurrence of elements 48
specifying in Accelerator 18
mappers
creating in Accelerator 9
reusable 51
running
running mappers 35
mapping
alternative mapping 46
conditions 44
controlled group 47
definition 14
expressions 38
functions 42
groups 45
lookups 43
map similar branches 31
specifying 30
multiple-occurrence elements
iterated mapping 34

N

navigating
main-mapping worksheet 32
nested groups
defining 46

P

parser
chaining with mapper 53
post-mapper
transformation 53
pre-mapper
transformation 53
projects
Accelerator 23
creating Accelerator 24

R

reusable
mappers 51
reusable entities
exporting and importing 53
reusable mappers
exporting and importing 53
Reusable Mappers
worksheet 60
rows
inserting, copying, and deleting 31
running mappers
test-drive mode 35

S

sample data
importing to Accelerator 30
Schemas
source and target 28
sequence
of mapping rows 32
serializer
chaining with mapper 53
services
deploying from Accelerator 25
specifying a mapper
definition 14
Studio
integration with Accelerator 24
system requirements
Accelerator 11

T

test-drive mode
running mapper 35
troubleshooting
mappers 35

U

undo
editing operations 31
user interface 27

X

XLS files
Accelerator 23
XML schemas
XSD 28
XPath
alias for 29
XSD
XML schemas 28