Informatica

Informatica® Test Data Management
9.7.1 HotFix 1

Administrator Guide

Informatica Test Data Management Administrator Guide
9.7.1 HotFix 1
August 2016

© Copyright Informatica LLC 2003, 2018

This software and documentation contain proprietary information of Informatica LLC 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 LLC. 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, Informatica Master Data Management, and Live Data Map are trademarks or registered trademarks of Informatica LLC 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 © 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. Copyright © Scalable Systems, Inc. All rights reserved. Copyright © jQWidgets. All rights reserved. Copyright © Tableau Software, Inc. All rights reserved. Copyright© MaxMind, Inc. All Rights Reserved. Copyright © TMate Software s.r.o. All rights reserved. Copyright © MapR Technologies Inc. All rights reserved. Copyright © Amazon Corporate LLC. All rights reserved. Copyright © Highsoft. All rights reserved. Copyright © Python Software Foundation. All rights reserved. Copyright © BeOpen.com. All rights reserved. Copyright © CNRI. 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; http://jibx.sourceforge.net/jibx-license.html; https://github.com/lyokato/libgeohash/blob/master/LICENSE; https://github.com/hjiang/jsonxx/blob/master/LICENSE; https://code.google.com/p/lz4/; https://github.com/jedisct1/libsodium/blob/master/LICENSE; http://one-jar.sourceforge.net/index.php?page=documents&file=license; https://github.com/EsotericSoftware/kryo/blob/master/license.txt; http://www.scala-lang.org/license.html; https://github.com/tinkerpop/blueprints/blob/master/LICENSE.txt; http://gee.cs.oswego.edu/dl/classes/EDU/oswego/cs/dl/util/concurrent/intro.html; https://aws.amazon.com/asl/; https://github.com/twbs/bootstrap/blob/master/LICENSE; https://sourceforge.net/p/xmlunit/code/HEAD/tree/trunk/LICENSE.txt; https://github.com/documentcloud/underscore-contrib/blob/master/LICENSE, and https://github.com/apache/hbase/blob/master/LICENSE.txt.

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.

See patents at https://www.informatica.com/legal/patents.html.

DISCLAIMER: Informatica LLC 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 LLC 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: 2018-07-03

Table of Contents

Preface 8

Informatica Resources. 8

Informatica Network. 8

Informatica Knowledge Base. 8

Informatica Documentation. 9

Informatica Product Availability Matrixes. 9

Informatica Velocity. 9

Informatica Marketplace. 9

Informatica Global Customer Support. 9

Chapter 1: Introduction to TDM Administration 10

TDM Administration Overview. 10

TDM Architecture. 11

TDM Tools. 11

TDM Server. 12

TDM Services. 12

TDM Databases. 12

TDM Connections. 13

Test Data Manager. 14

Logging In to the Test Data Manager. 15

Changing Passwords. 15

Chapter 2: User and Role Administration 16

User and Role Administration Overview. 16

User and User Group Management. 17

Privileges. 17

Informatica Privileges. 18

Test Data Manager Service Privileges. 19

Optional Privileges. 26

Test Data Manager Service Custom Roles. 27

PowerCenter Permissions. 30

Folder Permissions. 30

Connection Object Permissions. 31

Project Permission. 31

Data Set Permission. 32

Chapter 3: Security Administration 33

Security Administration Overview. 33

Test Data Manager Authentication. 33

TDM Server Security. 34

Creating a Keystore File for SSL. . . . 34
Configuring TLS and SSL. . . . 34
Configuring SSL for the Command Line Interface. . . . 35
Password Encryption. . . . 35
Creating an Encrypted Password. . . . 35
Encrypting the Password for the Command Line Interface. . . . 35
Session Timeout. . . . 36
Configuring Session Timeout. . . . 36

Chapter 4: System Preferences. . . . 37

System Preferences Overview. . . . 37
General Properties. . . . 37
Data Domain Sensitivity. . . . 38
Project Configuration. . . . 38
Workflow Generation. . . . 38
Storage Tables for Repeatable Masking. . . . 40
Creating the Storage Tables. . . . 40
Naming Convention for Workflows, Mappings and Sessions. . . . 41
Creating a Naming Format For Workflows, Mappings and Sessions. . . . 42
Default Settings. . . . 42
Log Severity Level Settings. . . . 44
Data Discovery. . . . 45
Test Tool Integration. . . . 45

Chapter 5: TDM Server Administration. . . . 47

Starting and Stopping the TDM Server. . . . 47
Logs. . . . 47
Log Configuration. . . . 48
Application Logs. . . . 50
License Management. . . . 50

Chapter 6: Connections. . . . 51

Connections Overview. . . . 52
Connection Permissions. . . . 52
Connection Management. . . . 53
Creating a Connection. . . . 53
Copying a Connection. . . . 54
Importing a Connection. . . . 54
Editing a Connection. . . . 54
Editing Connection Permission. . . . 55
Deleting a Connection. . . . 55
DB2 for Linux, UNIX and Windows Connections. . . . 56
Informix Connections. . . . 57

JDBC Connections. 58
Microsoft SQL Server Connections. 59
Netezza Connections. 61
ODBC Connections. 64
Oracle Connections. 66
PowerExchange DB2 for i5/OS Connections. 68
PowerExchange DB2 for Linux, UNIX and Windows Connections. 70
PowerExchange DB2 for z/OS Connections. 72
PowerExchange Microsoft SQL Server Connections. 76
PowerExchange NRDB Batch Connections. 78
PowerExchange Oracle Connections. 80
PowerExchange Sybase Connections. 82
SAS Analytics Connections. 84
Sybase Connections. 85
Teradata Connections. 87
Teradata FastExport Connections. 88
Teradata FastLoad Connections. 89
Teradata MultiLoad Connections. 91
Teradata Parallel Transporter Connections. 93

Chapter 7: Dictionaries. 95

Dictionaries Overview. 95
Relational Dictionaries. 95
Adding a Relational Dictionary. 96
Flat File Dictionaries. 96
Flat File Dictionary Properties. 97
Adding a Flat File Dictionary. 97
Dictionary Management. 97

Chapter 8: Configuring the Test Data Warehouse. 99

Test Data Warehouse Overview. 99
Test Data Warehouse Architecture. 99
Test Data Warehouse Configuration. 101
Creating a Test Data Warehouse. 101
Adding an Existing Test Data Warehouse. 101
Editing a Test Data Warehouse Repository Password. 102
Editing a Test Data Warehouse Password. 102

Chapter 9: User-Defined Datatypes. 103

User-Defined Datatypes Overview. 103
Properties File. 103
Datatype Conversion Parameters. 104
Setting Up the Properties File for User-Defined Datatypes. 104

Creating Functions for User-Defined Datatypes in Oracle Applications. 105

Chapter 10: tdwcmd. 106

tdwcmd Overview. 106

Configuring tdwcmd. 106

Running tdwcmd. 107

Entering Options and Arguments. 107

Syntax Notation. 108

Version. 108

CreateTDWREPO. 109

CreateTestDataWarehouse. 111

List. 114

List Examples. 116

Index. 117

Preface

The Informatica *Test Data Management Administrator Guide* describes how to perform the administrator functions on the Test Data Manager tool. This guide is written for system administrators. It assumes knowledge of operating systems, database engines, flat files, and PowerCenter.

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.

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.
- 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.

CHAPTER 1

Introduction to TDM Administration

This chapter includes the following topics:

- TDM Administration Overview, 10
- TDM Architecture, 11
- Test Data Manager, 14

TDM Administration Overview

Test Data Management (TDM) provides data masking, data subset, data discovery, and data generation capabilities to manage non production data in your organization.

As a TDM administrator, you can perform the following tasks:

- Manage security and users. Create roles and assign privileges and roles to users from Informatica Administrator to distribute duties among users and restrict user access.
- Configure system preferences. You can set system preferences from Informatica Administrator to configure workflows for optimum performance. You can also set system preferences to configure data domain sensitivity levels for tracking the sensitive data that users want to mask.
- Create connections. Create connections to application services to perform data masking, data subset, data discovery, and data generation operations. Create connections to the source and target databases to perform these operations.
- Integrate a test tool with TDM. Integrate a test tool to directly copy flat file results to the test tool.
- Create a test data warehouse. Configure the required relational databases for the test data warehouse repository and the test data warehouse.
- Perform server management tasks. You can configure licenses and restart the TDM Server from the Informatica Administrator, and view server logs from Test Data Manager.

TDM Architecture

The TDM architecture consists of tools, the TDM server, the Test Data Manager Service and other application services, and databases.

The following image shows the components of TDM:

The TDM architecture consists of the tools that you use to modify the data.

The application services include Model Repository Service, Data Integration Service, PowerCenter Repository Service, PowerCenter Integration Service, and Test Data Manager Service. The Model Repository Service and Data Integration Service access the Model repository and profiling warehouse.

Test Data Manager accesses the Test Data Manager Service and uses the database content from the TDM repository and the test data warehouse repository. PowerCenter services access the PowerCenter repository. The application services access the domain configuration repository and Model repository that maintain connections and metadata for the domain.

TDM Tools

TDM tools consist of Test Data Manager, Informatica Administrator, and the PowerCenter Client.

You can use the following tools to perform administrative tasks for TDM:

Test Data Manager

A web-based application that you can use to create connections, configure the test data warehouse, integrate a test tool with TDM, and manage preferences and dictionaries.

Informatica Developer

A client application that you use to create and export profiles for data discovery.

Informatica Administrator

A web-based client that a domain administrator uses to manage application services and create users and user groups.

PowerCenter Client

A client application that you use to configure permissions on folders and connection objects for the PowerCenter repository.

TDM Server

The TDM Server runs the Test Data Manager and integrates with the Test Data Manager Service and other Informatica application services to perform data subset, data masking, data generation, and data discovery operations.

TDM Services

TDM application services consist of the Test Data Manager Service, profiling services, and PowerCenter application services. The application services are created in the Administrator tool.

TDM requires the following services:

PowerCenter Repository Service

An application service that manages the PowerCenter repository. The PowerCenter Repository Service accepts requests from the PowerCenter Integration Service when a workflow runs.

PowerCenter Integration Service

An application service that runs workflows that users generate in the Test Data Manager for data subset and data masking operations. When you configure workflow options, you can select the PowerCenter Integration Service that runs workflows for users.

Model Repository Service

An application service that manages the Model repository for data discovery operations.

Data Integration Service

An application service that performs data discovery operations. The Data Integration Service connects to the Model Repository Service to store metadata from data discovery profiles in the Model repository. When you run a profile, the Data Integration Service also stores data from data discovery profiles in the profiling warehouse.

Analyst Service

Manages the Analyst tool that TDM uses to link TDM global objects to objects in the Business Glossary. The license for the Analyst Service to use database content from the TDM repository.

Test Data Manager Service

The TDM application service that manages the TDM repository. The Test Data Manager accesses the Test Data Manager Service to use database content from the TDM repository.

TDM Databases

The databases component of TDM consists of the TDM repository, the Model repository, the PowerCenter repository, the profiling warehouse, the test data warehouse repository, and the domain configuration repository.

TDM uses the following databases:

TDM repository

A relational database that contains tables that TDM requires to run and the tables that store metadata.

Model repository

A relational database that stores table metadata for data discovery profiles and the connections that you create in Test Data Manager.

PowerCenter repository

A relational database that stores all of the metadata required to run workflows.

Profiling warehouse

A relational database that stores profile results for data discovery.

Test data warehouse repository

A relational database that stores the source table metadata and project metadata for data sets that you create in Test Data Manager.

Domain configuration repository

A relational database that stores the connections used to run profiles, users for the Informatica domain, and metadata for the Informatica domain.

TDM Connections

To perform data discovery, data subset, data masking, and data generation operations, you need a profiling connection, a repository connection, and source and target database connections.

To perform data discovery operations, an application requires connections to a database source and a Data Integration Service. To perform data subset and masking operations, workflows that you generate from plans require connections to services, the TDM repository, the PowerCenter repository, and source and target databases. To perform data generation operations, workflows that you generate from plans require connections to services, the TDM repository, the PowerCenter repository, and target schema metadata. Workflows for operations to store data in the test data warehouse require a connection to the test data warehouse repository.

TDM uses the following connections:

- Profiling connection. Create a connection to the Data Integration Service that you want to run data discovery operations. The profiling connection includes connection details for the Model Repository Service associated with the Data Integration Service.
- TDM repository connection. Create a connection to the TDM repository when you create the Test Data Manager Service from the Administrator tool. Test Data Manager uses the TDM repository to store the metadata from sources that you import.
- PowerCenter repository connection. Create a connection to the PowerCenter repository that you want to store workflow metadata. If the TDM repository and the PowerCenter repository exist in the same schema, copy the connection information for the TDM repository.
- Test data warehouse repository connection. Create a connection to the test data warehouse repository from the **Administrator | Test Data Warehouse** view. Test Data Manager stores the project metadata and source table metadata for data sets in the test data warehouse repository.
- Database connections. Create connections to the source and target databases to perform data discovery, data subset, and data masking operations. Create a connection to a database to configure it as the test data warehouse. Import target metadata schema into the TDM repository to perform data generation operations. When you create a database connection in Test Data Manager, you define connection information for workflows and profiles. When you run a workflow or profile, the Test Data Manager uses the connection information to create connection objects in the PowerCenter repository or the Model repository.
- Test tool connection. To run workflows to copy flat file results to a test tool, you must first integrate the test tool server with TDM. TDM connects to the integrated server to copy the flat file to the configured location.

Test Data Manager

Test Data Manager is a web-based interface that you use to manage connections, preferences and workflow settings.

Access Test Data Manager from a web browser.

The following image shows the Test Data Manager:

A workspace is a container for Test Data Manager components. You can click on connections and other Test Data Manager components to open them in another workspace.

The Test Data Manager contains views. By default, an administrator can access the **Administrator** view of the Test Data Manager. To access the other views in the Test Data Manager, assign the privileges for the other views to the administrator account.

The **Administrator** view contains the following tabs:

Preferences

Configure connections to application services and configure workflow preferences, data domain sensitivity levels, data discovery profiling, and default settings for data generation.

Connections

Configure connections to source and target databases.

Dictionaries

View and manage dictionaries to use in masking rules.

Application Logs

View and search application level logs.

Test Data Warehouse

Configure a test data warehouse.

The contents panel shows an overview of the items in a view.

The details panel shows additional details for a single item in the contents panel.

Logging In to the Test Data Manager

To access the Test Data Manager, enter the host name and port number of the TDM Server in a web browser.

To log in, enter a user name and password defined in Informatica Administrator.

1. In the address bar of a web browser, enter the Test Data Manager URL.
   - Use the following format if Transport Layer Security is enabled:

     `https://hostname:portnumber/tdm/`
   - Use the following format if Transport Layer Security is not enabled:

     `http://hostname:portnumber/tdm/`

   Where:
   - *hostname* is the host name or IP address of the machine where you installed the TDM Server.
   - *portnumber* is the port number. The default is 6643 if Transport Layer Security is enabled. The default is 6605 if Transport Layer Security is not enabled.

   For example, you might enter the following URL:

   `http://TXW1779:6643/tdm/`

   The **Login** dialog box of the Test Data Manager appears.
2. Enter the user name and password.

   Select the security domain. If the Informatica domain is configured to use LDAP authentication, the default security domain **Native**.
3. Click **Login**.

   The Test Data Manager appears.

To log out of the Test Data Manager, click **Logout**.

Changing Passwords

To change the password for the administrator account and for other users, use the PowerCenter Administrator tool.

CHAPTER 2

User and Role Administration

This chapter includes the following topics:

- User and Role Administration Overview, 16
- User and User Group Management, 17
- Privileges, 17
- PowerCenter Permissions, 30
- Project Permission, 31
- Data Set Permission, 32

User and Role Administration Overview

Test Data Manager users are stored in the Informatica domain. To perform tasks in the Test Data Manager, users need roles, privileges, and permissions. User roles, privileges, and permissions are assigned through the Administrator tool and the PowerCenter Client.

The Informatica administrator creates users and groups in the Administrator tool. Work with the Informatica administrator to edit any user or group in the Administrator tool. You cannot edit users or groups in the Test Data Manager.

Users need the following types of roles, privileges, and permissions:

Informatica privileges

Includes privileges to change passwords and to generate and start workflows. The Informatica administrator assigns privileges in the Administrator tool.

Test Data Manager Service roles and privileges

Includes roles and privileges to perform actions in the Test Data Manager. Assign Test Data Manager Service roles and privileges in the Administrator tool.

PowerCenter permissions

To generate and start workflows, users need folder and connection object permissions. Use the PowerCenter Client to assign permissions to users.

The Informatica administrator and the TDM administrator perform the following tasks to manage TDM users:

1. The Informatica administrator creates Test Data Manager users in the domain and assigns privileges to them.
2. The TDM administrator assigns Test Data Manager roles and privileges.

3. The Informatica administrator assigns folder and connection object permissions in the PowerCenter Client.

You can manage user roles, privileges, and permissions from the Administrator Tool. For more information, see the *Informatica Administrator Guide*.

User and User Group Management

After the Informatica administrator creates domain users, you can create users and grant access to Test Data Manager actions through roles and privileges.

After you install, the Informatica administrator has administrative privileges. Use the Informatica administrator to create users and user groups.

Note: Before you can create users and groups, the default Informatica administrator user must assign Security Administration privileges to the Test Data Administrator user.

Privileges

Privileges determine the tasks that users can perform in the Test Data Manager. Users require domain privileges and Test Data Manager privileges.

The Informatica administrator assigns domain privileges, and you assign Test Data Manager Service privileges. Domain privileges work in conjunction with Test Data Manager Service privileges. For example, a developer that creates data masking or data generation plans needs Test Data Manager Service privileges to create the plans in the Test Data Manager. The developer also needs domain privileges to generate and run the data masking or data generation operations.

Note: Administrators can create custom roles that contain privileges and assign roles to users from the Informatica Administrator.

Informatica Privileges

The Informatica administrator assigns PowerCenter Repository Service privileges to users that need to perform subset, masking, and generation operations. All users need the domain privilege to change passwords. Users do not need domain privileges to perform profiling operations.

PowerCenter Repository Service Privileges to Perform Subset, Masking, and Generation Operations

The following table lists the PowerCenter Repository Service privileges that users need to perform data subset, data masking, and data generation operations:

Privilege Group	Privileges
Runtime Objects	Create, Edit, and Delete Monitor Execute
Tools	Access Designer Access Repository Manager Access Workflow Manager Access Workflow Monitor
Design Objects	Create, Edit, and Delete
Sources and Targets	Create, Edit, and Delete
Folders	Create, Copy, and Manage Versions
Global Objects	Create Connections Manage Deployment Groups Execute Deployment groups Create Labels Create Queries

PowerCenter Repository Service Administrator Role to Generate and Run Workflows

To generate and run workflows, users must have the system-defined role **Administrator** on the PowerCenter Repository Service in addition to the required TDM privileges.

Privilege to Change Passwords

Test Data Manager users need the domain privilege, Access Informatica Administrator, to change their passwords in the Administrator tool.

Test Data Manager Service Privileges

Test Data Manager Service privileges determine the actions that users can perform using the Test Data Manager. A user with the privilege to perform certain actions requires permissions to perform the action on a particular object. Configure permissions on the Security tab of the Administrator tool.

The following table describes each Test Data Manager privilege group.

Privilege Group	Description
Administration	Includes privileges to create and manage connections, roles and assign privileges to users and user groups from the Informatica Administrator, manage repositories, add licenses, and set up workflow and project attributes. **Note:** Before you can create users and groups, the default Informatica administrator user must assign Security Administration privileges to the Test Data Administrator user.
Data Domains	Includes privileges to view and manage data domains in the Test Data Manager.
Data Masking	Includes privileges to view and manage masking rules and policy assignments in the Test Data Manager.
Data Subset	Includes privileges to view and manage subset objects including entities, groups and templates in the Test Data Manager.
Policies	Includes privileges to view and manage policies in the Test Data Manager.
Projects	Includes privileges to view and manage projects, audit and import metadata, and execute plans and workflows in the Test Data Manager.
Rules	Includes privileges to view and manage masking and generation rules in the Test Data Manager.
Data Generation	Includes privileges to view and manage test data generation in the Test Data Manager.

Administration Privilege Group

The privileges in the Administration privilege group determine the administration tasks that Test Data Administrators can perform.

The following table lists the privileges in the Administration privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
Manage Preferences	-	Write	User can perform the following actions on the Informatica Administrator and Test Data Manager: - Create roles. - Edit roles. - Delete roles. - View roles. - Associate roles to users. - Associate privileges to users. - Associate roles to user groups. - Associate privileges to user groups. - Add licenses. - Set up the TDM repository. - Set up the PowerCenter repository. - Set up data domain sensitivity levels. - Configure a test data warehouse repository. - Configure a test data warehouse. - Set up project custom attributes. - Set up workflow generation attributes. - Enable data discovery. - Set up profiling services. - View administration objects. - Configure keyword search indexing options.
View Connections	-	Read	User can perform the following actions on the Connections page in the Test Data Manager: - View connections. - Test connections.
Manage Connections	View Connections	Write	User can perform the following actions on the Connections page in the Test Data Manager: - Create connections. - Edit connections. - Delete connections. - View connections. - Test connections. - Configure a test data warehouse repository. - Configure a test data warehouse.

Data Domains Privilege Group

The privileges in the Data Domains privilege group determine the tasks that users can perform on data domains on the Policies page of the Test Data Manager.

The following table lists the privileges in the Data Domains privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Data Domains	-	Read	User can view data domains in the Test Data Manager.
Manage Data Domains	View Data Domains	Write	User can perform the following actions on data domains in the Test Data Manager: - Create data domains. - Edit data domains. - Delete data domains. - View data domains.

Data Masking Privilege Group

The privileges in the Data Masking privilege group determine the tasks that users can perform on the Project | Define | Data Masking view of the Test Data Manager. You can assign rules and polices to table columns from this view.

The following table lists the privileges in the Data Masking privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Data Masking	-	Read	User can view data masking assignments in the Test Data Manager.
Manage Data Masking	View Data Masking	Write	User can perform the following data masking assignment actions in the Test Data Manager: - Add rule and policy assignments. - Delete rule and policy assignments. - Override rule properties. - View data masking assignments.

Data Subset Privilege Group

The privileges in the Data Subset privilege group determine the tasks that users can perform on data subset objects in the Test Data Manager.

The following table lists the privileges in the Data Subset privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Data Subset	-	Read	User can perform the following data subset actions in the Test Data Manager: - View groups. - View templates - View entities. - View recent project objects.
Manage Data Subset	View Data Subset	Write	User can perform the following data subset actions in the Test Data Manager: - Create groups. - Edit groups. - Delete groups. - Add group parameters. - Create templates. - Edit templates. - Delete templates. - Add template parameters. - Create entity. - Edit entity. - Delete entity. - Add entity criteria. - Enable relationships. - Disable relationships. - Edit relationships - Review and act on changes. - Mark change review as complete.

Policies Privilege Group

The privileges in the Policies privilege group determine the tasks that users can perform on Policies in the Test Data Manager.

The following table lists the privileges in the Policies privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Policies	-	Read	User can view policies in the Test Data Manager.
Manage Policies	View Policies	Write	User can perform the following policy actions policies in the Test Data Manager: - Create policies. - Edit policies. - Delete policies. - View policies.

Projects Privilege Group

The privileges in the Projects privilege group determine the tasks that users can perform on Projects in the Test Data Manager.

The following table lists the privileges in the Projects privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Project	-	Read	User can perform the following actions on projects in the Test Data Manager: - View projects. - View plans. - View plan detail reports. - View plan audit reports. - View recent projects. - View data set versions.
Manage Project	View Project	Write	User can perform the following actions on projects in the Test Data Manager: - Create projects - Edit projects. - Delete projects - View projects. - Associate users to projects. - Associate user groups to projects. - Associate or remove rules to projects. - Associate or remove policies to projects - Create plans. - Edit plans. - Delete plans. - Generate plans. - Edit a data set version. - Delete a data set version.

Privilege	Includes Privileges	Permission	Description
Discover Project	-	Write	User can perform the following discover actions on projects in the Test Data Manager: - Classify tables. - Mark discovery as complete. - Associate data domains to columns. - Mark columns as restricted. - Mark columns as sensitive - Set similar value column - Remove similar value columns - Add primary keys - Remove primary Keys - Create logical constraints - View logical constraints - Edit logical Constraints - Delete Logical Constraints - View projects. - View profiled data domains. - Approve or reject profile data domains. - Mark data domain classification as complete. - View profiled primary keys. - Approve or reject profiled primary keys. - Mark primary key discovery as complete. - View profiled entities. - Approve or reject profiled entities. - Mark entity discovery as complete. - View project risk analysis. - View recent project sensitive data distribution.
Generate Project	-	Write	User can generate workflows in the Test Data Manager.
Execute Project	-	Write	User can perform the following execute actions on projects in the Test Data Manager: - Execute plans. - Execute workflows. - Stop workflows. - Abort workflows. - Recover workflows. - View plan execution. - Create a data set version. - Reset a data set version.
Monitor Project	-	Read	User can perform the following monitor actions on projects in the Test Data Manager: - Monitor project jobs. - View project job logs. - Monitor jobs across projects. - View job logs across projects.
Audit Project	-	Read	User can view recent activity on projects and plans in the Test Data Manager.
Import Metadata	-	Write	User can perform the following actions on projects in the Test Data Manager: - Import sources - Delete sources.

Note: A user with Manage Project privilege must have at least the following levels of privileges to be able to create a plan with each component.

- View connection from the Administration privilege group. To create a plan.
- View data subset from the Data Subset privilege group. To create a plan with subset components.
- View masking rules from the Rules privilege group. To create a plan with masking components.
- View generation rules from the Rules privilege group. To create a plan with generation components.

Rules Privilege Group

The privileges in the Rules privilege group determine the tasks that users can perform on data masking and data generation rules in the Test Data Manager.

The following table lists the privileges in the Data Masking privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Masking Rules	-	Read	User can view masking rules in the Test Data Manager.
Manage Masking Rules	View Masking Rules	Write	User can perform the following actions on data masking rules in the Test Data Manager: - Create masking rules. - Edit masking rules. - Delete masking rules. - View masking rules.
View Generation Rules	-	Read	User can view generation rules in the Test Data Manager.
Manage Generation Rules	View Generation Rules	Write	User can perform the following actions on data generation rules in the Test Data Manager: - Create generation rules. - Edit generation rules. - Delete generation rules. - View generation rules.

Data Generation Privilege Group

The privileges in the Data Generation privilege group determine the test data generation tasks that users can perform in the Test Data Manager.

The following table lists the privileges in the Data Generation privilege group and the permissions required to perform a task on an object:

Privilege	Includes Privileges	Permission	Description
View Data Generation	-	Read	User can view data generation rule assignments in the Test Data Manager.
Manage Data Generation	View Data Generation	Write	User can perform the following actions on data generation in the Test Data Manager: - View data generation rule assignments - Add data generation rule assignments. - Delete data generation rule assignments. - Override data generation rule assignments.

Optional Privileges

Based on the tasks performed, you might need to assign additional privileges to some users.

Users can link TDM global objects with business terms from a business glossary. To create, edit, and delete links to business terms, users need Model Repository Service and Analyst Services privileges. Users must have at least read permission on the glossary or the specific business term that they access. You can configure permission from the Analyst Tool.

Users can view terms linked to any object they have view privileges on. To create, edit, or delete a link to an object, users must have Manage privilege on the object.

The following table lists the minimum Model Repository Service privileges that users need to perform asset linking tasks:

Privilege Group	Privileges
Model Repository Service Administration	Access Analyst Access the Analyst Tool.

The following table lists the minimum Analyst Service privileges that users need to perform asset linking tasks:

Privilege Group	Privileges
Workspace Access	Glossary Workspace Access the glossary workspace within the Analyst Tool.

Users must have access permission to the Analyst Service to view or edit linked business terms. Assign a user access permission from the **Security** view in Informatica Administrator. To assign multiple users or user groups access permission, select the service in the Domain Navigator of the Informatica Administrator. Select the **Permissions** view and edit direct permissions.

For information on assigning permission from Informatica Administrator, see the *Informatica Administrator Guide*.

Test Data Manager Service Custom Roles

The Test Data Manager Service custom roles include the Test Data Administrator, Test Data Developer, Test Data Project DBA, Test Data Project Developer, Test Data Project Owner, Test Data Risk Manager, and Test Data Specialist.

Test Data Administrator

The following table lists the default privileges assigned to the Test Data Administrator custom role:

Privilege Group	Privilege Name
Projects	Audit Project
Administration	- View Connections - Manage Connections

Test Data Developer

The following table lists the default privileges assigned to the Test Data Developer custom role:

Privilege Group	Privilege Name
Policies	- View Policies - Manage Policies
Rules	- View Masking Rules - Manage Masking Rules - View Generation Rules
Data Domains	- View Data Domains - Manage Data Domains
Projects	Audit project

Test Data Project DBA

The following table lists the default privileges assigned to the Test Data Project DBA custom role:

Privilege Group	Privilege Name
Projects	- View Project - Execute Project - Monitor Project - Audit Project
Administration	- View Connections - Manage Connections

Test Data Project Developer

The following table lists the default privileges assigned to the Test Data Project Developer custom role:

Privilege Group	Privilege Name
Policies	View Policies
Rules	- View Masking Rules - View Generation Rules
Data Domains	View Data Domains
Projects	- View Project - Discover Project - Execute Project - Monitor Project - Audit Project - Import Metadata
Data Masking	- View Data Masking - Manage Data Masking
Data Subset	- View Data Subset - Manage Data Subset
Data Generation	- View Data Generation - Manage Data Generation
Administration	- View Connections - Manage Connections

Test Data Project Owner

The following table lists the default privileges assigned to the Test Data Project Owner custom role:

Privilege Group	Privilege Name
Policies	View Policies
Rules	- View Masking Rules - View Generation Rules
Data Domains	View Data Domains
Projects	- View Project - Manage Project - Discover Project - Execute Project - Monitor Project - Audit Project - Import Metadata
Data Masking	- View Data Masking - Manage Data Masking

Privilege Group	Privilege Name
Data Subset	- View Data Subset - Manage Data Subset
Data Generation	- View Data Generation - Manage Data Generation
Administration	- View Connections - Manage Connections

Test Data Risk Manager

The following table lists the default privileges assigned to the Test Data Risk Manager custom role:

Privilege Group	Privilege Name
Policies	View Policies
Rules	- View Masking Rules - View Generation Rules
Data Domains	View Data Domains
Projects	Audit project

Test Data Specialist

The following table lists the default privileges assigned to the Test Data Specialist custom role:

Privilege Group	Privilege Name
Policies	View Policies
Rules	- View Masking Rules - Manage Masking Rules - View Generation Rules - Manage Generation Rules
Data Domains	- View Data Domains - Manage Data Domains
Projects	- Manage Project - View Project - Discover Project - Execute Project - Monitor Project - Audit Project - Import Metadata
Data Masking	- View Data Masking - Manage Data Masking
Data Subset	- View Data Subset - Manage Data Subset

Privilege Group	Privilege Name
Data Generation	- View Data Generation - Manage Data Generation
Administration	- View Connections - Manage Connections

Note: If you have upgraded to Informatica service 9.6.1 HotFix 2 from Informatica service 9.6.1, a user with the Test Data Specialist role cannot create or delete data generation rules. The role does not include the Manage Data Generation privilege. To enable users with this role to create and delete data generation rules, you must manually edit the role. Log in to the Administrator tool and edit the Test Data Manager service custom role to include the Manage Generation Rules privilege from the Rules privilege group.

PowerCenter Permissions

To generate workflows, users need folder and connection object permissions. Assign permissions in the PowerCenter Client.

Folders organize and store metadata in the PowerCenter repository. Folders have the following types of permissions:

- Read. You can view the folder and objects in the folder.
- Write. You can create or edit objects in the folder.
- Execute. You can run or schedule workflows in the folder.

Connection objects are global objects that define connections in the PowerCenter repository. Connection objects have the following types of permissions:

- Read. View the connection object in the Workflow Manager and Repository Manager. When you have read permission, you can perform tasks in which you view, copy, or edit repository objects associated with the connection object.
- Write. Edit the connection object.
- Execute. Run sessions that use the connection object.

Folder Permissions

Each project created in the Test Data Manager is associated with a folder in the PowerCenter repository. To view objects in the folder and run workflows for objects in the folder, users need folder permissions.

When you create a project, the TDM Server creates a folder in the PowerCenter repository for the project. By default, the owner of a project is also the owner of the folder and has read, write, and execute permissions for the folder.

If the TDM repository is in a separate schema than the PowerCenter repository, the TDM repository contains a link to the folder in the PowerCenter repository.

Work with the Informatica administrator to assign folder permissions in the PowerCenter Repository Manager.

Connection Object Permissions

Users need connection object permissions to generate and run workflows.

Work with the Informatica administrator to assign the read and execute permissions to users that run workflows requiring the connection object. Assign write permissions to users that have the privilege to manage connections in the Test Data Manager.

Alternatively, the Informatica administrator can specify a default set of permissions for connection objects in the Workflow Manager. If enhanced security is enabled in the Workflow Manager, the Workflow Manager assigns the default permissions to users and user groups.

Project Permission

You can assign project permissions to control access to projects. A project owner and the domain administrator can assign and edit permission to users and user groups.

You can access and perform tasks in a project based on the permissions that you have.

Projects have the following levels of permission:

- Read
- Write
- Execute

To perform any task in a project, you must also have the minimum required level of privileges as a TDM user.

The following table lists the project permission levels, the tasks that you can perform with each level, and the minimum required privileges for each task:

Permission	Description	Minimum Required Privilege
Read	- Open and view the project. - Monitor logs for the project workflows.	- View project - Monitor project - Audit project
Write	- Open and view the project. - Monitor logs for the project workflows. - Import metadata. - Create entities, groups, and templates. - Assign rules. - Generate workflows. - Run profiles. - Copy the project. - Delete the project.	- View project - Monitor project - Audit project - Import metadata - Generate project - Manage project - Discover project
Execute	- Open and view the project. - Monitor logs for the project workflows. - Run workflows.	- View project - Monitor project - Audit project - Execute project

Assign and edit project permission from the **Permissions** tab in a project in Test Data Manager.

Data Set Permission

You can assign permissions to control access to data sets that you store in the test data warehouse.

When you run the plan that creates the data set, you become the owner of the data set. A data set owner and the domain administrator can add users and user groups and assign the required level of access. You can access data sets based on the permissions that you have. If you do not have the required permissions, you cannot open data sets that appear in search results.

Data sets have the following levels of permission:

- Read
- Write
- Execute

To perform any task that uses the data set, you must also have the minimum required level of privileges as a TDM user.

The following table lists the permission levels, the tasks that you can perform with each level, and the minimum required privileges for each task:

Permission	Description	Minimum Required Privilege
Read	- Open and view the data set. - Monitor logs for the data set.	- View project
Write	- Open and view the data set. - Monitor logs for the data set. - Edit the description and tags of the data set.	- View project - Manage project
Execute	- Open and view the data set. - Monitor logs for the data set. - Edit the description and tags of the data set. - Create a data set. - Reset a data set.	- View project - Execute project

The owner and the administrator can lock and unlock a specific version of a data set to restrict edit and reset.

CHAPTER 3

Security Administration

This chapter includes the following topics:

- Security Administration Overview, 33
- Test Data Manager Authentication, 33
- TDM Server Security, 34
- Password Encryption, 35
- Session Timeout, 36

Security Administration Overview

To add security to the TDM Server, you can configure Kerberos authentication, password encryption, the Secure Sockets Layer (SSL) protocol, and the Transport Layer Security (TLS) protocol.

You can implement Kerberos authentication for the Test Data Manager. TDM also supports Kerberos authentication for specific source and connection types.

You can also configure the TDM Server to use SSL and TLS. When you configure SSL and TLS, you ensure secure communication with the Informatica domain.

TDM also supports SSL authentication for source and target connections.

Implement password encryption for additional security. Use the command line interface to encrypt the password for the PowerCenter repository. If you implement SSL, you can also use the command line interface to encrypt the keystore password and the password for the command line interface

To log users out of the Test Data Manager after a period of inactivity, configure session timeout.

For information on how to configure Kerberos network authentication, see the *Informatica Security Guide*.

Test Data Manager Authentication

Kerberos is an authentication protocol that uses cryptography to verify the identity of clients and servers. If you use Kerberos to authenticate Test Data Manager users, you must edit configuration files for the TDM Server.

When you configure Kerberos for TDM, the Kerberos server verifies the identity of the TDM Server and the identity of the Test Data Manager users before establishing a secure connection between the two.

To use Kerberos authentication, configure the following files for the TDM Server:

- `krb5.conf`. Contains configuration information for Kerberos. This file is stored in the following location: `<INFA_HOME>/TDM/datadirect`.
- `org.eclipse.virgo.kernel.authentication.config`. Contains authentication properties for the TDM Server, including driver information. This file is stored in the following location: `<INFA_HOME>/TDM/configuration`.

Note: If you have multiple TDM servers, you must configure the files on each server.

TDM Server Security

TDM communication with the Informatica domain depends on the configuration of the domain. If the domain is configured to run in TLS mode, the communication with the domain is secure. You can configure TLS and SSL to ensure secure a connection for the Test Data Manager web application.

You configure TLS and SSL when you create the Test Data Manager Service. If you disable TLS and SSL when you create the service, you can configure TLS and SSL by editing the service properties. You must also create a keystore file. The keystore file stores security certificates for SSL encryption. Specify the security certificate that you want to use for SSL when you create the keystore file, and set a password for the keystore file.

Creating a Keystore File for SSL

Before you can enable SSL, you must export the SSL certificate that you want to use. Use the `keytool` utility bundled with Java to create a keystore file.

1. Open a command prompt in Windows or a terminal in Linux or UNIX.
2. Run the following command:

   ```
   keytool -import -file company_certificate.cer -keystore client.ks
   ```

3. The utility prompts you to enter a keystore password.

   The command creates a keystore file called `client.ks`.
4. Save the keystore file to a directory. You must provide the location of the keystore file when you create or edit the Test Data Manager Service properties.

Configuring TLS and SSL

Enable TLS and SSL for the Test Data Manager web application. Enable TLS and SSL when you create the Test Data Manager Service or edit the Test Data Manager Service properties.

1. Log in to the Informatica Administrator.
2. Select the Test Data Manager Service and click **Edit** the **Test Data Manager Server Configuration** tab.
3. Select the Enable Transport layer Security (TLS) check box and enter the following properties.
   - HTTPS Port. Port number for the HTTPS connection. The default is 6643.
   - Keystore File. The path for the keystore file with relation to the TDM installation directory.
   - Keystore Password. The keystore password.

- SSL Protocol. Secure Sockets Layer protocol to use. Default is TLS.

Configuring SSL for the Command Line Interface

Edit the `userConfig.ilm` file to enable SSL and TLS for the command line interface.

1. Open the following file in a text editor:

   ```
   <TDM Installation Directory>\utilities\ilmcli\conf\userConfig.ilm
   ```

2. Set the following property to true:

   ```
   isHTTPS=true
   ```

3. Save the `userConfig.ilm` file.
4. Restart the TDM Server.

Password Encryption

Use the command line interface to encrypt the keystore password, the PowerCenter repository password, and the password used by the command line interface to access the TDM repository.

Then, update the following configuration file that store the password:

- `<TDM Installation Directory>\utilities\ilmcli\conf\userConfig.ilm`. Stores the password used by the command line interface to access the TDM repository

After you edit the configuration files, restart the TDM Server.

Creating an Encrypted Password

Use the command line interface to create an encrypted password.

1. At the command line, switch to the directory where the command line executable is located. By default, the executable is installed in the following directory:

   ```
   <TDM Installation Directory>\utilities\ilmcli\bin
   ```

2. Enter the following command:

   ```
   ilmcmd -Encrypt password_to_encrypt
   ```

   The command returns the encrypted password.

Encrypting the Password for the Command Line Interface

You can encrypt the password used by the the command line interface to access the TDM repository. The password for the command line interface is stored in the `userConfig.ilm` file.

1. Open the following file in a text editor:

   ```
   <TDM Installation Directory>\utilities\ilmcli\conf\userConfig.ilm
   ```

2. Set the password to the encrypted value from the command line interface.
   For example, you might enter the following value for the password:

   ```
   password=uWlm059lmcj6QyLVzfpu6rK0BzpePJ472MBYOS85x6I=
   ```

3. Set the value of the following line to true:

   ```
   isPasswordEncrypted=true
   ```

4. Save the `userConfig.ilm` file.

Session Timeout

To determine how long an inactive user stays logged into the Test Data Manager, configure the session timeout.

The Test Data Manager logs a user out after a period of inactivity. You can alter the web.xml configuration file to change the amount of time that a user can remain inactive in the Test Data Manager.

The default session timeout is two minutes. The minimum is two minutes. Enter -1 to disable session timeout.

Configuring Session Timeout

Configure the Test Data Manager session timeout in the web.xml file.

1. Go to the TDM configuration directory:

   ```
   <TDM Installation Directory>/TDM/configuration
   ```

2. Find the web configuration file:

   web.xml

   Back up the file before you make any changes.

3. Use a text editor to edit the web.xml file. To change the session timeout, find the session-timeout property.

   The session-timeout property has a default of two minutes, as shown in the following example:

   ```
   <session-config>
           <session-timeout>2</session-timeout>
   </session-config>
   ```

   Change the session timeout to the amount of time in minutes that you want a user to remain inactive before the Test Data Manager logs out the user. Enter -1 to disable session timeout.

4. Save the web.xml file.

5. Restart the TDM Server.

CHAPTER 4

System Preferences

This chapter includes the following topics:

- System Preferences Overview, 37
- General Properties, 37
- Data Domain Sensitivity, 38
- Project Configuration, 38
- Workflow Generation, 38
- Naming Convention for Workflows, Mappings and Sessions, 41
- Default Settings, 42
- Log Severity Level Settings, 44
- Data Discovery, 45
- Test Tool Integration, 45

System Preferences Overview

System preferences determine global options that appear for all users of Test Data Manager. Configure system preferences for Test Data Manager before users complete any subset, masking, or generation operations.

System preferences include search index properties, data domain sensitivity levels, custom project fields, log severity level settings, default settings for data generation, test tool integration settings, Hive properties, and global workflow properties.

You configure system preferences in the **Administrator | Preferences** view.

General Properties

You cannot perform a search from the search field unless you index objects in the TDM repository at least once. You can schedule a date and time to perform a complete indexing of TDM objects. You can change the rate at which TDM indexes objects to optimize the keyword search return times.

You can set the start time for search indexing and the frequency of indexing from the **Administrator | Preferences** view. To define general properties, click **Edit** in the **General Properties** section.

The following table describes the general properties:

Property	Description
Search re-index frequency (Hours)	The rate at which TDM indexes objects. Enter the frequency in hours. The default is 24 hours.
Search index start time	The date and time at which TDM starts to index objects. You can choose to start indexing immediately or schedule indexing for a future date and time. Click the calendar to enter the value and then click **Done**. - Now. Select **Now** to begin indexing. - Browse the calendar and select a date and time to schedule indexing at a future date and time.

You can also click **Start Search Indexer** from the Actions menu on the**Administrator | Preferences** view to begin indexing.

Data Domain Sensitivity

When users create a data domain, they select the sensitivity level for all columns in the domain. By default, users can select a high, medium, or low sensitivity level. You can configure additional sensitivity levels available for all data domains.

You can configure additional sensitivity levels and edit the labels for the default levels. For example, you might want to add a level for critically sensitive columns such as columns that contain a Social Security number. You can add an additional sensitivity level named Critical.

When you change the sensitivity levels, the changed levels appear for all data domains in the TDM repository.

To configure sensitivity levels, click **Edit** in the **Data Domain Sensitivity Levels** section.

Project Configuration

A project is the top-level container that you can use to organize the components for data discovery, masking, subset, and generation operations. You can add custom optional fields that appear for all projects.

For example, you might want to add a custom optional field named "Business Unit" or "Organization ID" to all projects. The added fields display for all projects in the TDM repository. Users enter values for the optional project fields when they create or edit a project.

To add custom optional fields to all projects, click **Edit** in the **Project Configuration** section.

Workflow Generation

Before users generate workflows, define properties for all data subset and data masking workflows.

To define workflow properties, click **Edit** in the **Workflow Generation** section.

The following table describes the workflow generation properties:

Property	Description
Central seed value is a parameter	Select this option to enter a central seed value as a parameter.
Central Seed Value	Number that overrides the individual seed values that are configured for repeatable output for data masking components. Enter a central seed value to ensure that data masking components return the same masked values in different plans. When you set a central seed value, the PowerCenter Integration Service uses that value for all data masking components with repeatable output. The seed value modifies the logic which is used in data masking operations. For example, to mask the same CUST_ID column in four tables, such that all of them output the same masked values, set a central seed value. If you use a different seed value for each column, each column returns different masked values. The seed values that you specify ensure that columns return masked values consistently. Valid values are from 1 through 999. You can enter the central seed value as a parameter.
Cache Location	Directory for index and data cache files that the PowerCenter Integration Service uses when it runs workflows. Default is `<Informatica _Installation_Directory>/server/infa_shared/cache`.
Maximum Tables in a Mapping	Maximum number of tables that all workflows create for each session. You can override this value when you create a workflow. If you run the PowerCenter Integration Service on a machine with a fast processor, increase the number of tables to optimize performance. Default is 20.
Storage Connection	A connection to a database that contains storage tables. The Data Masking transformation maintains storage tables for repeatable substitution and expression masking between sessions. A storage table row contains the source column and a masked value pair. The storage tables for substitution and expression masking are separate tables IDM_SUBSTITUTION_STORAGE and IDM_EXPRESSION_STORAGE.
Staging Connection	A connection to a database for creating temporary staging tables. The default set in the Workflow Generation section appears during plan creation. You can change the connection used in a plan. Use this connection for scenarios that involve Cyclic Subset, Tending to Cyclic subset, Inplace Batch Updates, Non Relational Shuffle, and Hierarchical PWX.
Mapping Name	Format of the mapping name that TDM creates when you run a plan. You can enter an expression to create a naming format. TDM creates all mappings with names in the same format. Use the expression editor to create the format. If you do not enter a format, the plan assigns a random name.
Workflow Name	Format of the workflow name that TDM creates when you run a plan. You can enter an expression to create a naming format. TDM creates all workflows with names in the same format. Use the expression editor to create the format. If you do not enter a format, the plan assigns a random name.
Session Name	Format of the session name that TDM creates. You can enter an expression to create a naming format. TDM creates all sessions with names in the same format. Use the expression editor to create the format. If you do not enter a format, the plan assigns a random name.

Storage Tables for Repeatable Masking

To perform expression masking or substitution masking with repeatable values, you must assign a storage table for the repeatable values. You can create the table in the TDM repository database or in another database. TDM creates a default storage connection and creates storage tables for this connection.

If you choose to use a different storage connection, you must set this connection as the default connection and create the storage tables for the connection. Set the connection as the default storage and staging connections and create storage tables in Test Data Manager.

Staging connections and storage connections can be on Oracle, Sybase, IBM DB2, or Microsoft SQL Server databases. You can create a staging connection on Teradata and Teradata Parallel Transporter connections.

Note: To use a Teradata or a Teradata Parallel Transporter connection as a staging connection, the Teradata JDBC JAR files `tdgssconfig.jar` and `terajdbc4.jar` must be present in the following location: `<Informatica installation directory>\TDM\utilities\mapgen\thirdpartylib`. Create a `thirdpartylib` folder and copy the JAR files before you configure the connection as a staging connection.

You can create the storage tables for the required staging and storage connection from the **Administrator** view in Test Data Manager.

You can also use the SQL scripts that Informatica provides to create the storage tables. Use your database client tool to run the SQL script and create the table in the database. For example, you can use the SQL*Plus client for Oracle to create the storage table in an Oracle database.

Run the storage table SQL script for your database. For example, to create the substitution storage table in a Microsoft SQL Server database, run the *Substitution_SQL_Server.sql* file. The SQL scripts to create the storage table for repeatable expression masking are installed in the following TDM directory:

```
<TDM Installation Directory>/TDM/storage_script/expression_storage
```

After you run the expression storage script, verify that the script created the IDM_EXPRESSION_STORAGE table in the database. The SQL scripts to create the storage table for repeatable substitution masking are installed in the following TDM directory:

```
<TDM Installation Directory>/TDM/storage_script/substitution_storage
```

After you run the substitution storage script, verify that the script created the IDM_SUBSTITUTION_STORAGE table in the database.

Important: The expression and substitution storage tables can contain original values from the source database and the corresponding masked values from the target database. Ensure that access to the storage table is secure. Alternatively, you can create the expression or substitution storage table in a database with limited user access.

Creating the Storage Tables

Create storage tables if you do not want to use the default staging and storage connection and storage tables that TDM creates.

1. In the **Administrator | Preferences** view, click **Edit** in the **Workflow Generation** section.

   The **Edit Preferences** dialog box appears.
2. Select the required default storage connection from the list.
3. Click **Create Storage Tables** to create the storage tables for this connection.

Naming Convention for Workflows, Mappings and Sessions

To standardize names, you can customize the naming convention for the workflows, mappings, and sessions that TDM creates. All plans that you run use the format that you create. If you do not enter a format, the plan assigns a random name.

You can configure the naming format on the **Workflow Generation** tab of the **Administrator | Preferences** view. You can use functions and variables available in the expression editor to create an expression to define the naming format.

You can use the following functions in an expression:

Function	Description
CONCAT(string1, string2)	Creates a single character string by joining two or more strings.
LOWER(str)	Converts a string to lowercase.
SUBSTR(str,m,n)	Returns a portion of a string, beginning and ending at specified positions in the string. m = start index and n = end index. Indexes start from 0. The substring is calculated from start index to end index-1.
UPPER(STR,m,n)	Converts a string to uppercase.

You can use the following variables in an expression:

Variable Name	Description
$PROJECTNAME	The name of the project.
$PLANNAME	The name of the plan.
$SOURCECONNECTIONNAME	The name of the source connection in the plan.
$TARGETCONNECTIONNAME	The name of the target connection in the plan.
$SOURCESCHEMANAME	The name of the source schema in the plan.
$TARGETSCHEMANAME	The name of the target schema in the plan.
$TABLENAME	The name of the table in the workflow. If you use the variable $TABLENAME in an expression, the variable is ignored in the name in the following situations: - The expression defines the naming format for workflows and the workflow contains more than one table. - The expression defines the naming format for mappings and the mapping contains more than one table. - The expression defines the naming format for sessions, and the session contains more than one table, and there are multiple tables with the same name.

For example, the following expressions use the CONCAT function with variables:

- `CONCAT('w_', $PLANNAME,'_',$SOURCECONNECTIONNAME,'_',$SOURCESCHEMANAME)`
  When used as the format for workflow names, generates workflow names that begin with "w_" and include the plan, source connection, and source schema names.
- `CONCAT('m_', $PLANNAME,'_',$SOURCECONNECTIONNAME,'_',$TARGETSCHEMANAME)`
  When used as the format for mapping names, generates mapping names that begin with "m_" and include the plan, source connection, and target schema names.
- `CONCAT('s_', $PROJECTNAME,'_',$SOURCECONNECTIONNAME,'_',$SOURCESCHEMANAME)`
  When used as the format for session names, generates session names that begin with "s_" and include the project, source connection, and target schema names.

When you create a format that uses variables, TDM uses information that you enter in the plan settings section of the plan creation workflow.

A plan appends a counter to mapping and session names if the names are not unique across the plan. It appends a counter to workflow names if the plan contains multiple workflows or if a workflow by the same name exists. The name of a workflow, mapping, or session can be a maximum of 75 characters. In addition to this, a counter is appended if required.

Note: To prevent overwriting of workflows, ensure that you create an expression that generates unique workflow names.

Creating a Naming Format For Workflows, Mappings and Sessions

Create a format for workflow, mapping, and session names to standardize names of workflow objects and to easily identify objects. If you do not create a format, plans assign a random name.

1. Open the **Administrator | Preferences** view.
2. Click **Actions** > **Edit** and click the **Workflow Generation** tab.
3. Click Edit in the **Mapping Name**, **Workflow Name**, or **Session Name** field.

   The **Configure Expression** dialog box opens.
4. Select the required functions and variables from the **Functions** and **Variables** tabs and click the transfer arrow to move them to the **Expression** tab.
5. Configure the expression as required.
6. Click the **Validate Expression** button to verify that the expression is valid.
7. Click **OK**.

Default Settings

Before you create a plan to generate data, you define default plan settings in the **Administrator | Preferences** view. You can configure and update the settings when you create the plan. Use default settings to generate data for the columns that you have not assigned specific generation rules to.

To configure default settings, click **Edit** in the **Default Settings** section.

The following table describes the properties to configure the default settings:

Property	Description
Default Rule for Numeric Datatype	The default rule that you can select to generate data for numeric data types.
Default Rule for String Datatype	The default rule that you can select to generate data for string data types.
Default Rule for Date Datatype	The default rule that you can select to generate data for date data types.
Default Rule for Binary Datatype	The default rule that you can select to generate data for binary data types.
Default Primary Key Rule for Numeric Datatype	The default primary key rule that you can select to generate data for numeric data types.
Default Primary Key Rule for String Datatype	The default primary key rule that you can select to generate data for string data types.
Default Primary Key Rule for Date Datatype	The default primary key rule that you can select to generate data for date data types.
Default Primary Key Rule for Binary Datatype	The default primary key rule that you can select to generate data for binary data types.
Default Unique Key Rule for Numeric Datatype	The default unique key rule that you can select to generate data for numeric data types.
Default Unique Key Rule for String Datatype	The default unique key rule that you can select to generate data for string data types.
Default Unique Key Rule for Date Datatype	The default unique key rule that you can select to generate data for date data types.
Default Unique Key Rule for Binary Datatype	The default unique key rule that you can select to generate data for binary data types.
Apply Default Rule to	The default rule that you can apply to all the columns without generation rules. You can also apply the default rule to the columns that do not have null values.
Default Character Set	The character set that you can select. Default is ASCII.
Minimum	The minimum number of child records for a parent record. Default is 1.
Maximum	The maximum number of child record for a parent record. Default is 1.
Parent Records Without Children	The number of parent records that have no child.
Child Records Without a Parent	The number of child records that have no parent.

Log Severity Level Settings

You can set or update the log severity level of TDM modules to view logs and troubleshoot errors.

Some of the information or warning messages might not display the root cause of the problem. You can change the log levels for a module depending on the level of detail that you need. You can set the following severity levels:

- Error
- Warning
- Info
- Debug
- Trace

To set the log severity level of the modules in the application, click **Edit** in the **Log Severity Level Settings** section.

You can set log severity levels for the following modules:

All Modules

Contains logs from all the TDM modules. You can select a common severity level for all the TDM modules at a time.

Miscellaneous

Contains preferences management, connection management, asset linking, data source deletion, expression validation, authentication, and privileges and permissions logs.

Generation Rules

Contains data generation rules and components logs.

Command Line Utility

Contains command line program logs.

Metadata Import

Contains metadata import UI logs.

Project Discovery

Contains all the logs from the **Discover** tab of a project.

Masking Policies and Assignments

Contains masking policies, data domains, masking rules, masking rule assignments, policy assignment, and masking rule simulation logs.

TDM Data Model

Contains all the logs when you write to or read from the TDM data model.

TDM UI

Contains all the logs generated while rendering the TDM UI.

Plan

Contains plan management logs.

Project

Contains project management logs.

Monitor

Contains all the monitoring logs.

Search

Contains global search logs.

Subset

Contains logs from the data subset components such as entity, group, and template.

Import and Export

Contains XML import and export logs from UI and command line programs.

Offline Jobs

Contains offline import and workflow management job logs.

Data Discovery

TDM uses a Data Integration Service to perform data discovery. You must connect to a Data Integration Service within the Informatica domain and enable data profiling before users can run data discovery profiles. By default, data profiling is disabled in Test Data Manager.

You can connect to a Data Integration Service within the Informatica domain. Specify the Data Integration Service to connect to and enable data profiling when you create the Test Data Manager Service from the Administrator tool. You can edit the Test Data Manager Service properties and connect to a different Data Integration Service in the domain if you need to.

See the Informatica *Test Data Management Installation Guide* for information on how to edit the Test Data Manager Service.

Test Tool Integration

You can integrate the HP Application Lifecycle Management (ALM) tool with TDM. You can then copy flat file target results in TDM to a project in the integrated HP ALM server. You can copy results of subset, masking, and generation operations that have flat file targets. Use this feature to directly copy and maintain flat file results in an HP ALM server. You can then use the data to create and run test cases in HP ALM.

You can also integrate an HP ALM server that uses an HTTPS connection. TDM runs a separate job in the same workflow to copy the flat file to the HP ALM server in addition to the TDM server. You cannot export a flat file if you enable version control on the HP ALM server. You must disable version control and then run the TDM workflow.

You must run the workflow from Test Data Manager. The test tool integration job does not run when you run the workflow from the PowerCenter client.

To copy flat file results to an HP ALM server, you must configure the server information in TDM. Configure the server settings from the **Test Tool Settings** tab in the **Administrator | Preferences** view.

The following table describes the test tool integration properties:

Property	Description
Test Tool Integration	Select this option to enable test tool integration.
User Name	Required. The HP ALM server user name.
Password	Required. The password for the HP ALM server user name.
Target File Location	The default target folder location that TDM uses when you enable test tool integration. TDM sets the following target location: *${INFA_HOME}\TDM\tti* You cannot edit this location.
Host	Required. Host name of the HP ALM server.
Port	Required. Port number of the HP ALM server. For example, 8080.
Enable Transport Layer Security (TLS)	Select this option if the HP ALM server has secure access enabled.
HTTPS Port	Required if you enable TLS. Port number of the HP ALM server. For example, 8443.
Keystore File	Required if you enable TLS. You must copy the keystore file to the TDM server. Enter the complete path and the file name of the keystore file on the TDM server.
Keystore Password	Required if you enable TLS. Enter the password to the keystore file on the TDM server.

CHAPTER 5

TDM Server Administration

This chapter includes the following topics:

- Starting and Stopping the TDM Server, 47
- Logs, 47
- License Management, 50

Starting and Stopping the TDM Server

The TDM Server runs as the Test Data Manager Service in the Informatica domain. You can start and stop the TDM Server by enabling and disabling the Test Data Manager Service from the Administrator tool.

When you edit the TDM Server configuration files, you must restart the server to implement the changes.

You can change the startup and shutdown ports through the Test Data Manager Server Configuration properties page of the Test Data Manager Service.

Logs

TDM log files record events and tasks that the TDM Server and the Test Data Manager perform. You can configure log levels, log sizes, and archive settings for log files.

The TDM Server creates the following types of log files:

Client logs

Contain information about actions performed in the **Administrator** view of the Test Data Manager. The client logs also includes log information about login attempts and actions that users perform.

The client logs are stored in the following location:

```
<Informatica installation directory>/TDM/logs/tdm.log
```

Event logs

Contain TDM Server events, including startup information for the TDM Server.

The event logs are stored in the following location:

```
<Informatica installation directory>/TDM/logs/events.log
```

Server logs

Contain detailed information about all the actions that the TDM Server performs. You can find log information about user activities, such as the creation of rules and policies.

The server logs are stored in the following location:

```
<Informatica installation directory>/TDM/logs/log.log
```

Job logs

Contain detailed information about all the jobs that the TDM Server performs. You can find log information about every job that a user performs in TDM.

The job logs are stored in the following location:

```
<Informatica installation directory>/TDM/logs/jobLogs/job_<ID>.log
```

Console logs

Contain detailed information for all the jobs that the TDM Server performs. You can find log information about each job when TDM triggers a job. You can view the console logs even when the job logs are not present.

The console logs are stored in the following location:

```
<Informatica installation directory>/TDM/logs/jobLogs/console_<ID>.log
```

Profiling logs

Contain detailed information for all the profiling jobs that the TDM Server performs.

The profiling logs are stored in the following location:

```
<Informatica_Home>/tomcat/bin/disLogs
```

Test Data Manager Service logs

Contain detailed information for all the Test Data Manager Service startup, shut down, content creation, and content upgrade actions.

You can access all the Test Data Manager Service logs from the **Logs | Services** view of the Informatica Administrator tool.

Log Configuration

TDM uses the logback logging system to perform logging for the TDM Server. You can configure log levels and the location of log files.

To configure log files, modify the log configuration file. The log configuration file is stored in the following location: `<TDM Installation Directory>/configuration/serviceability.xml`

The changes that you make to `serviceability.xml` take effect when you restart the TDM Server.

Note: You can find out more information about the logback logging system at the following site: http://logback.qos.ch/

Log Levels

You can configure the level of logging for the client log, the event log, and the server log. You can set the severity levels for the TDM modules in Test Data Manager.

When you configure `serviceability.xml`, you can change the logging levels through the `logger level` property. If the log level for a particular log file is not specified, the TDM Server uses the value specified in the `root level` property as the log level.

The following table describes the log levels that you can set:

Log Level	Description
ALL	Contains messages for all log levels.
DEBUG	Indicates TDM Server operations at a detailed level. Debug messages generally record the success or failure of server operations. Debug messages have the lowest severity level.
INFO	Indicates that the TDM Server is performing an operation that does not indicate errors or problems. Information messages have the second lowest severity level.
WARN	Indicates that the TDM Server is performing an operation that may cause an error. Warning messages have the third highest severity level.
ERROR	Indicates that the TDM Server failed to perform an operation or respond to a request from a client application. Error messages have the second highest severity level.
FATAL	Fatal error occurred. Fatal error messages have the highest security level.
TRACE	Indicates TDM Server operations at a more specific level than the debug logs. Trace messages are generally trace code paths. Trace messages have the lowest severity level.
OFF	Turns logging off.

Log Location and Archives

You can change the location where the TDM Server writes log files. You can also configure log file archiving.

When you configure `serviceability.xml`, you can configure archiving for the client log, event log, the server log.

The following table describes the properties that you configure for the log location and archiving:

Property	Description
file	The file path for the log file with reference to the TDM installation directory.
FileNamePattern	The file path and the naming convention for archived log files with reference to the TDM installation directory. Use the `%i` modifier archive files by number. Use the `%d` modifier to archive files by date. To save directory space, use .gz or .zip when you archive files. For example, you might specify the following file name pattern: `<TDM installation directory>/serviceability/logs/tdm_%i.log.zip`
MaxFileSize	The maximum size for a log file. You can configure a file size in bytes, kilobytes, megabytes, or gigabytes. Default is 10 MB.
MaxIndex	The maximum number of log files that the TDM Server will create. The TDM Server will not create more than 12 log files even if you configure a size greater than 12. Default is 4.

Archive Example

Your organization wants to configure the archive specification for the server log, `tdm.log`.

You configure the maximum file size to be 8 MB, and you configure the maximum number of files to be seven. The TDM Server uses the following process to manage the server log:

1. The first time you start the TDM Server, it creates a file named `tdm.log`.
2. It continues to write to the file until the file reaches a size of 8 MB.
3. When the file reaches 8 MB, it renames the file to `tdm_1.log.zip`, and it creates another log named `tdm.log`.
4. When `TDM.log` reaches 8 MB, it renames `tdm_n.log.zip` to `tdm_n+1.log.zip`, and it creates another server log named `tdm.log`.
5. The TDM Server continues to rename the log files until it reaches the maximum index of seven, and it overwrites the oldest log file.

The following snippet of code shows the configuration for the server log.

```
<appender name="TDM_LOG_FILE" class="ch.qos.logback.core.rolling.RollingFileAppender">
            <file>serviceability/logs/tdm.log</file>
            <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
                  <FileNamePattern>serviceability/logs/tdm_%i.log.zip</FileNamePattern>
                  <MinIndex>1</MinIndex>
                  <MaxIndex>7</MaxIndex>
            </rollingPolicy>
            <triggeringPolicy
class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
                  <MaxFileSize>8MB</MaxFileSize>
            </triggeringPolicy>
            <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
                  <Pattern>[%d{yyyy-MM-dd HH:mm:ss.SSS}] %-5level %-28.28thread
%-64.64logger{64} User:X{username} %msg %ex%n</Pattern>
            </encoder>
      </appender>
```

Application Logs

Application logs record the internal errors that occur within Test Data Manager. Application logs do not contain any job-related logs.

Application logs contain the following severity levels: Debug, Info, Warning, and Error. View the application logs from the **Application Logs** tab. You can perform search, sort, and filter operations based on the date range, context, severity level, error description, and owner name. When you select an error message, the row expands to display the details of the error message. You can scroll down to 1000 lines to view the complete log. To refresh the log messages, click **Refresh**. To download the log file, click **Download**.

License Management

You can view details for license keys, and you can add or remove license keys from the Informatica Administrator.

For information on how to manage licenses from the Informatica Administrator, see the *Informatica Administrator Guide*.

CHAPTER 6

Connections

This chapter includes the following topics:

- Connections Overview, 52
- Connection Permissions, 52
- Connection Management, 53
- DB2 for Linux, UNIX and Windows Connections, 56
- Informix Connections, 57
- JDBC Connections, 58
- Microsoft SQL Server Connections, 59
- Netezza Connections, 61
- ODBC Connections, 64
- Oracle Connections, 66
- PowerExchange DB2 for i5/OS Connections, 68
- PowerExchange DB2 for Linux, UNIX and Windows Connections, 70
- PowerExchange DB2 for z/OS Connections, 72
- PowerExchange Microsoft SQL Server Connections, 76
- PowerExchange NRDB Batch Connections, 78
- PowerExchange Oracle Connections, 80
- PowerExchange Sybase Connections, 82
- SAS Analytics Connections, 84
- Sybase Connections, 85
- Teradata Connections, 87
- Teradata FastExport Connections, 88
- Teradata FastLoad Connections, 89
- Teradata MultiLoad Connections, 91
- Teradata Parallel Transporter Connections, 93

Connections Overview

Create connections to databases to perform data subset, masking, and generation operations, and to run profiles for discovery operations.

You can create connections in TDM. You can also import connections that you created in PowerCenter and Informatica Administrator. When you create a connection in Test Data Manager, TDM stores the connection information in the Model repository.

Connection requirements depend on the operations that you need to perform. To perform a data discovery operation, TDM requires a connection to a database source. To perform data subset and data masking, TDM requires connections to source and target databases. To perform data generation, TDM requires a connection to a target database.

TDM supports some databases for certain TDM operations only. When you create a connection, relevant properties appear in the connection configuration wizard. For example, if you create a connection to a database that you cannot run a profile on, the wizard does not provide the unique profiling connection parameters.

To create a connection for a MySQL database, you use the ODBC connection type. If you use the connection as a target in a plan in which you disable and enable the constraints, you must enter a target JDBC connection. You enter a target JDBC connection in the plan to provide a JDBC connection string. Use the JDBC connection type to create a MySQL connection to enter as the target JDBC connection. You cannot use a connection that you create using the JDBC connection type as a source or target connection.

TDM supports SSL authentication for source and target connections. TDM supports Kerberos network authentication for some connection types. You can enable Kerberos authentication in the connection configuration wizard.

To restrict access to connections, you can assign permissions to connections that you create in Test Data Manager.

Create and manage connections from the **Administrator | Connections** view in Test Data Manager.

Connection Permissions

When you create a connection in Test Data Manager you become the owner of the connection. As the owner of the connection, you can add users and user groups and assign the required levels of permission. You can change the owner of the connection. The domain administrator can also add and edit connection permissions and can change the owner of the connection.

You can access connections based on the permissions that you have.

Connections have the following levels of permission:

- Read
- Write
- Execute

To perform any task that uses the connection, you must also have the minimum required level of privileges as a TDM user.

The following table lists the connection permission levels, the tasks that you can perform with each level, and the minimum required privileges for each task:

Permission	Description	Minimum Required Privilege
Read	- View the connection in Test Data Manager.	- View connections
Write	- View the connection in Test Data Manager. - Update and delete the connection.	- View connections - Manage connections
Execute	- View the connection in Test Data Manager. - Update and delete the connection. - Import metadata. - Run profiles. - Generate workflows. - Run workflows. - Use relational dictionaries. - Perform rule simulation.	- View connections - Manage connections

Connection Management

You can create, copy, import, remove, and validate source and target connections in Test Data Manager.

You can manage connections in the **Administrator | Connections** view.

You can import connections from the PowerCenter repository and the domain configuration repository.

Select a connection, and click **Actions** > **Test Connection** to validate the connection.

You cannot assign an owner to connections that you create in the Administrator tool. When you import connections created in the Administrator tool, the user name that you enter when you create the Test Data Manager Service displays as the connection owner.

If the owner of the connection is not found in TDM or Workflow Manager, the user name that you enter when you create the Test Data Manager Service displays as the connection owner.

Creating a Connection

You can create a source or target connection in Test Data Manager.

Before you can test a Teradata or a Teradata Parallel Transporter connection, you must add the Teradata JDBC JAR files `tdgssconfig.jar` and `terajdbc4.jar` to the following location: `<Informatica installation directory>\TDM\lib\thirdparty`. Restart the Test Data Manager Service after you add the files.

Note: If you reuse the name of a connection previously created and deleted in Test Data Manager, a workflow that you use the connection in might fail. The workflow might fail because a connection object with the same name and different properties exists in the domain repository.

1. In the **Administrator | Connections** view, select **New Connection** from the **Actions** menu.

   A tab opens to display the new connection properties.
2. Select the connection type and define the connection name, description, and user information.

   The connection name must begin with an alphabetic character. If you enter a connection name that begins with a numeric character, a workflow that you use the connection in might fail.
3. Optional. Click **Change Owner** and select a different user as the connection owner.
4. Click **Next**.
5. Enter the connection properties.
6. Click **Test Connection** to test the connection.
7. Click **Finish** to save the connection.

   The connection is visible in the **Administrator | Connections** view.
8. Optional. Select the connection in the **Administrator | Connections** view, and click **Actions** > **Test Connection** to validate the connection.

Copying a Connection

You can create a copy of a connection. You might make a copy of a connection when you need to define a connection that is similar to an existing connection.

1. In the **Administrator | Connections** view, select a connection to copy.

   Do not open the connection.
2. Click **Actions** > **Duplicate**.

   The **Duplicate** dialog box appears.
3. Change the name and description for the connection.
4. Click **Save**.

   The connection appears in the connections list.

Importing a Connection

You can import a source or target connection in Test Data Manager.

1. In the **Administrator | Connections** view, select Import Connections from the **Actions** menu.

   The available source and target connections are imported into Test Data Manager from PowerCenter.
2. Go to the **Administrator | Connections** view to see a list of imported connections.

Editing a Connection

You can edit a connection to modify the connection properties.

1. In the **Administrator | Connections** view, click the connection that you want to edit.

   The connection opens in another tab.
2. Click **Actions** > **Edit**.

3. Modify the connection properties.
4. Click **Test Connection**.
5. Click **Save**.

Editing Connection Permission

Connection permissions determine the tasks that you can perform using the connection. You can edit the connection permission assigned to users and user groups from the **Permissions** tab of the connection.

1. In the **Administrator | Connections** view, click the connection that you want to edit.

   The connection opens in another tab.
2. Click the **Permissions** tab.

   A list of the TDM users or user groups with permissions for the connection appears.
3. Click **Edit** on the **Users** or **User Groups** tab.

   The **Edit Connection Permissions** dialog box opens.
4. To edit the permission of a user or user group, select the user or user group from the list and edit the permission as required. You must save the changes for each user or user group.
5. To delete a user or user group, select the user or user group from the list and click **Delete**.
6. To add a user or a user group:
   a. Click **Add Users** or **Add User Groups**.
   b. Select one or more users or user groups.
   c. Optional. From the list of permissions, select the required permissions if either of the following statements is true:
      - You selected a single user or user group.
      - You want to assign the same levels of permission to all selected users or user groups.
   d. Click **OK**. TDM adds the users or user groups to the list.
   e. Select each user or user group and assign the required permission levels. You must save the changes for each user or user group. Skip this step if you performed step c.
7. Click **OK**.

Deleting a Connection

You can delete a connection in Test Data Manager. When you delete a connection in Test Data Manager, the connection gets deleted from TDM, but the connection object does not get deleted from the domain repository. Consider editing the connection properties instead of deleting the connection.

1. In the **Administrator | Connections** view, select the connection that you want to delete.
2. Click **Actions** > **Delete**.
3. In the **Delete Connection** dialog box, click **Yes** to delete the connection.

DB2 for Linux, UNIX and Windows Connections

You can create a DB2 for Linux, UNIX and Windows connection in the Test Data Manager to perform data discovery, data subset, and data masking operations.

The following table describes the database connection properties for a DB2 for Linux, UNIX and Windows database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select DB2 for Linux, UNIX and Windows.
Description	The description of the connection. The description cannot exceed 255 characters.
Use Kerberos Authentication	Enables Kerberos Authentication. You cannot enter a user name and password if you select this check box.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Metadata Connection String	Required. The JDBC connection URL used to access metadata from the database. Enter jdbc:informatica:db2: //<hostname>:50000;databaseName=<dbname>. Used for all operations.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file. Used for all operations.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>. Used for all operations.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database. Used for all operations.

Property	Description
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction. Used for all operations.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Used for all operations. Default is 0.
Tablespace	The tablespace name of the database. Used for all operations.
Pass Through Security Enabled	Enables pass-through security for the connection. When you enable pass-through security for a connection, the domain uses the client user name and password to log into the corresponding database, instead of the credentials defined in the connection object. Used for data discovery operations. Default is disabled.

Informix Connections

You can create an Informix connection in the Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for an Informix database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Informix.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.

Property	Description
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.

JDBC Connections

Create a JDBC connection to perform TDM operations if the target ODBC connection requires a JDBC connection string.

For example, to create a connection for a MySQL database, you use the ODBC connection type. If you use the connection as a target in a plan in which you disable the indexes and constraints, you must enter a target JDBC connection. You enter a target JDBC connection in the **Target JDBC Connection** field in the plan to provide a JDBC connection string.
Use the JDBC connection type to create a MySQL database connection to use as the target JDBC connection. You cannot use a JDBC connection type as a source or target connection in a plan.

The following table describes the database connection properties for a JDBC database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select JDBC.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.

Property	Description
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Metadata Connection String	Required. The JDBC connection URL used to access metadata from the database. Enter the correct connection string for the database: - MySQL: jdbc:mysql://<hostname>:<port>/<dbname> - Oracle: jdbc:informatica:oracle://<hostname>:<port>;SID=<sid> - Microsoft SQL Server: jdbc:informatica:sqlserver://<hostname>:1433;SelectMethod=cursor;databaseName=<dbname> - DB2 for Linux, UNIX, and Windows: jdbc:informatica:db2://<hostname>:50000;databaseName=<dbname> - Sybase: jdbc:informatica:sybase://<hostname>:5000;DatabaseName=<databasename> - Teradata: jdbc:teradata://ServerHost/database=MyDatabaseName,tmode=ANSI,charset=UTF8
Database Type	Required. The type of database to which JDBC connects. The options are: - MySQL - Oracle - Microsoft SQL Server - DB2 for Linux, UNIX, and Windows - Sybase - Teradata Default is MySQL.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.

Microsoft SQL Server Connections

You can create a Microsoft SQL Server connection in Test Data Manager to perform data discovery, data subset, and data masking operations.

The following table describes the database connection properties for a Microsoft SQL Server database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Microsoft SQL Server.
Description	The description of the connection. The description cannot exceed 255 characters.
Use Kerberos Authentication	Enables Kerberos Authentication. You cannot enter a user name and password if you select this check box.
User Name	Required. The database user name.

Property	Description
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Metadata Connection String	Required. The JDBC connection URL used to access metadata from the database. Enter jdbc:informatica:sqlserver://<hostname>:1433;SelectMethod=cursor;databaseName=<dbname>. Used for data discovery operations.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file. Used for all operations.
Data Access Connection String	The connection string used to access data from the database. Enter <servername@dbname>. Used for all operations.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database. Used for all operations.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction. Used for all operations.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Used for all operations. Default is 0.
Domain Name	The name of the domain. Used for all operations.
Owner Name	The name of the owner of the schema.
Schema Name	The name of the schema in the database. Required in the following cases: - For the profiling warehouse and staging database if the schema name is different than the database user name - When you want to create and run a column profile Used for all operations.

Property	Description
Pass Through Security Enabled	Enables pass-through security for the connection. When you enable pass-through security for a connection, the domain uses the client user name and password to log into the corresponding database, instead of the credentials defined in the connection object. Used for data discovery operations. Default is disabled.
Use Trusted Connection	The Data Integration Service uses Windows authentication to access the Microsoft SQL Server database. The user name that starts the Data Integration Service must be a valid Windows user with access to the Microsoft SQL Server database.
Database Name	Required. Name of the database. If you do not enter a database name, connection-related messages do not show a database name when the default database is used. Used for all operations.
Server Name	Required. Database server name. Used for data subset and data masking operations.
Packet Size	Packet size for the transmission of data. Used to optimize the native Microsoft SQL Server drivers. Used for all operations. Default is 0.

Netezza Connections

You can create a Netezza connection in the Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a Netezza database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Netezza.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.

Property	Description
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.

You can transfer data in Netezza by using bulk mode. Use bulk mode to increase session performance.

In bulk mode, the PowerCenter Integration Service reads and writes Netezza data through an external table. An external table definition is stored within the Netezza database but the data is saved externally in a location that is accessible to the Netezza host or the client system. Create external tables to structure your loading operation and manipulate data by using Netezza SQL.

The following table describes Bulk Writer properties that you can enter for a Netezza connection to use the bulk loader option to write data:

Property	Description
Delimiter	The delimiter separates successive input fields. Set the delimiter to any value supported. The value must not be a part of the input data. Default is |.
Null value	The null value parameter of the external table. The PowerCenter Integration Service uses the null value internally. Maximum value is one character. Default is blank.
Escape Character	Escape character of the external table. If the data contains NULL, CR, and LF characters in the Char or Varchar field, you need to add an escape character for these fields before loading. Enter a backslash (\) as the escape character.
Quoted Value	The quoted value parameter of the external table. Select SINGLE or DOUBLE to enclose the field in single or double quotes. Select NO to omit quotes. Default is NO. The quoted value is not a part of the data.
Ignore Key Constraints	Ignores constraints on primary key fields. When you select this option, the PowerCenter Integration Service can write duplicate rows with the same primary key to the target. Default is disabled. The PowerCenter Integration Service ignores this value when the target operation is "update as update" or "update else insert."

Property	Description
Duplicate Row Handling	Determines how the PowerCenter Integration Service handles duplicate rows. Select one of the following values: - First Row. The PowerCenter Integration Service passes the first row to the target and rejects the rows that follow with the same primary key. - Last Row. The PowerCenter Integration Service passes the last duplicate row to the target and discards the rest of the rows. Default is First Row.
Bad File Name	Directory where the PowerCenter Integration Service creates a reject file. The file contains the rejected records. By default, the PowerCenter Integration Service creates a bad file in the following directory: $PMBadFileDir
Socket Buffer Size	Buffer size that the PowerCenter Integration Service uses to write data. Set the socket buffer size to 25 to 50 % of the DTM buffer size to increase session performance. You might need to test different settings for optimal performance. Enter a value between 4096 and 2147483648 bytes. Default is 8388608 bytes.
Control Character	The control character parameter of the external table to transfer data containing control characters. You can enter control characters for Char and Varchar fields. If you enter a control character, you must add an escape character for the NULL, CR, and LF fields. Default is TRUE.
CRINSTRING	The CRinstring parameter to transfer data containing carriage returns (CR). You can enter a non escape CR in Char or Varchar fields. To load the control characters present in the Char and Varchar fields, set the CTRLCHARS and CRINSTRING parameters to TRUE in the session properties for the Netezza source. Default is TRUE.

Pipe Directory Path	Path for the PowerCenter Integration Service to create the pipe for the external table. If you do not specify the path, the PowerCenter Integration Service uses the following directory to create the pipe for the external table:
	`<Informatica installation directory>/server/bin`
	Required if the machine hosting the PowerCenter Integration Service is on HP-UX and the following directory is on an NFS-mounted directory:
	`<Informatica installation directory>/server/bin>`
	Enter a path that does not use an NFS mount.

Error Log Directory Name	Error log directory can reside on the machine where the PowerCenter Integration Service runs. For example, you can use the following directory: $PMBadFileDir By default, the PowerCenter Integration Service creates the error log in the following directory on the machine hosting the NetezzaPerformance Server: /tmp The PowerCenter Integration Service creates a reject file in the error log directory if the data is not valid.

The following table describes Bulk Reader properties that you can enter for a Netezza connection to use the bulk loader option to read data:

Property	Description
Delimiter	The delimiter separates successive input fields. Set the delimiter to any value supported. The value must not be a part of the input data. Default is |.
Null value	The null value parameter of the external table. The PowerCenter Integration Service uses the null value internally. Maximum value is one character. Default is blank.
Escape Character	Escape character of the external table. If the data contains NULL, CR, and LF characters in the Char or Varchar field, you need to add an escape character for these fields before loading. Enter a backslash (\) as the escape character.
Socket Buffer Size	Buffer size that the PowerCenter Integration Service uses to read data. Set the socket buffer size to 25 to 50% of the DTM buffer size to increase session performance. You might need to test different settings for optimal performance. Enter a value between 4096 and 2147483648 bytes. Default is 8388608 bytes.

Pipe Directory Path	Path for the PowerCenter Integration Service to create the pipe for the external table. If you do not specify the path, the PowerCenter Integration Service uses the following directory to create the pipe for the external table:
	`<Informatica installation directory>/server/bin`
	Required if the machine hosting the PowerCenter Integration Service is on HP-UX and the following directory is on an NFS-mounted directory:
	`<Informatica installation directory>/server/bin>`
	Enter a path that does not use an NFS mount.

ODBC Connections

You can create an ODBC connection in the Test Data Manager to perform data discovery, data subset, and data masking operations.

The following table describes the database connection properties for an ODBC database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select ODBC.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.

Property	Description
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page used to read from a source database or write to a target database or file. Used for all operations.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>. Used for all operations.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database. Used for all operations.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction. Used for all operations.
Connection Retry Period	Number of seconds the Integration Service tries to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the integration object fails. Used for all operations. Default is 0.
Pass Through Security Enabled	Enables pass-through security for the connection. When you enable pass-through security for a connection, the domain uses the client user name and password to log into the corresponding database, instead of the credentials defined in the connection object. Used for data discovery operations. Default is disabled.
ODBC Provider	Required. The type of database to which ODBC connects. For pushdown optimization, specify the database type to enable the Integration Service to generate native database SQL. The options are: - Other. - Sybase. - MS SQL Server. Used for all operations. Default is Sybase.

Property	Description
SQL Identifier Character To Use	Required. The type of character used to identify special characters and reserved SQL keywords, such as WHERE. The Data Integration Service places the selected character around special characters and reserved SQL keywords. The Data Integration Service also uses this character for the Support Mixed-case Identifiers property. Select the following characters based on the database in the connection. - Double quotes " " - Single quotes ' ' - Back quotes ` ` - Brackets [ ] Used for data discovery operations. If you use MongoDB or Cassandra connections with special characters in the names, you must select the double quotes.
Support Mixed Case Identifiers	When enabled, the Data Integration Service places identifier characters around table, view, schema, synonym, and column names when generating and executing SQL against these objects in the connection. Use if the objects have mixed-case or lowercase names. By default, this option is not selected. Used for data discovery operations.

Oracle Connections

You can create an Oracle connection in the Test Data Manager to perform data discovery, data subset, and data masking operations.

The following table describes the connection properties for an Oracle database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Oracle.
Description	The description of the connection. The description cannot exceed 255 characters.
Use Kerberos Authentication	Enables Kerberos Authentication. You cannot enter a user name and password if you select this check box.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.

Property	Description
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Metadata Connection String	Required. The JDBC connection URL used to access metadata from the database. Enter jdbc:informatica:oracle://<hostname>:1521;SID=<sid>. Used for all operations.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file. Used for all operations.
Data Access Connection String	The native connection string to the database. Connection string to preview data and run mappings. Enter dbname.world from the TSNAMES entry. Used for all operations.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service executes the connection environment SQL each time it connects to the database. Used for all operations.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction. Used for all operations.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Used for all operations. Default is 0.
Enable Parallel Mode	Enables parallel processing when loading data into a table in bulk mode. Used for all operations. Default is disabled.
Pass Through Security Enabled	Enables pass-through security for the connection. When you enable pass-through security for a connection, the domain uses the client user name and password to log into the corresponding database, instead of the credentials defined in the connection object. Used for data discovery operations. Default is disabled.

PowerExchange DB2 for i5/OS Connections

You can create a PowerExchange DB2 for i5/OS connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange DB2 for i5/OS database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX DB2 for i5/OS.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database.
Library List	List of libraries that PowerExchange searches to qualify the table name for Select, Insert, Delete, or Update statements. PowerExchange searches the list if the table name is unqualified. If you specify multiple libraries, separate the library names with spaces and enclose the list in quotation marks (for example, "TGTLIB1 TGTLIB2 TGTLIB3"). **Note:** If you specify Library List and Database File Overrides and a table exists in both, Database Files Overrides takes precedence.
Isolation Level	Commit scope of the transaction. Select one of the following values: - None. - CS. Cursor stability. - RR. Repeatable read. - CHG. Change. - ALL. Default is None.
Array Size	Db2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from the DB2 source tables. Valid values are from 1 through 100000. Default is 25. Informatica recommends that you enter an array size no greater than 100.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.

Property	Description
Database Name	DB2 subsystem or database instance name.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is disabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.
Reject File	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the write mode is asynchronous with fault tolerance. To prevent the creation of the reject files, specify PWXDISABLE.
PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides: - `TCPIP_OP_TIMEOUT= network_operation_timeout ;` - `TCPIP_CON_TIMEOUT= connection_timeout`
Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.

Property	Description
Location	Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.
Database file overrides	Specifies the i5/OS database file override. The format is: `from_file/to_library/to_file/to_member` Where: - from_file is the file to be overridden. - to_library is the new library to use. - to_file is the file in the new library to use. - to_member is optional and is the member in the new library and file to use. *FIRST is used if nothing is specified. You can specify up to 8 unique file overrides on a single connection. A single override applies to a single source or target. When you specify more than one file override, enclose the string of file overrides in double quotes and include a space between each file override. **Note:** If both Library List and Database file overrides are specified and a table exists in both, the Database file overrides takes precedence.

PowerExchange DB2 for Linux, UNIX and Windows Connections

You can create a PowerExchange DB2 for Linux, UNIX and Windows connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange DB2 for Linux, UNIX and Windows database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX DB2 for Linux, UNIX and Windows.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database.

Property	Description
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Database Name	DB2 subsystem or database instance name.
Server Name	Database server name.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is disabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.

Property	Description
Reject File	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the write mode is asynchronous with fault tolerance. To prevent the creation of the reject files, specify PWXDISABLE.

PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides:
	`- TCPIP_OP_TIMEOUT= network_operation_timeout ;`
	`- TCPIP_CON_TIMEOUT= connection_timeout`

Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.
Location	Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.

PowerExchange DB2 for z/OS Connections

You can create a PowerExchange DB2 for z/OS connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange DB2 for z/OS database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX DB2 for z/OS.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database.
DB2 Subsystem ID	Required. Name of the DB2 subsystem.
Correlation ID	Value to use as the DB2 Correlation ID for DB2 requests. This value overrides the value that you specify for the SESSID statement in the DBMOVER configuration file.

Property	Description
Array Size	DB2 fetch array size, in number of rows, for DB2 bulk data movement operations that use the DB2 access method. The array size pertains to the DB2 multiple-row FETCH statements that PowerExchange uses to read data from the DB2 source tables. Valid values are from 1 through 100000. Default is 25. **Note:** You must have DB2 Version 8 new-function mode or a later DB2 version to use multiple-row FETCH statements.
Off Load Processing	For DB2 Version 8 new-function mode and later, specifies whether to use offload processing to move PowerExchange bulk data processing from the source system to the Integration Service machine. Select one of the following values: - No. Do not use offload processing. - Yes. Use offload processing. - Auto. PowerExchange determines whether to use offload processing. Default is No.
Worker Threads	When offload processing is enabled, specifies the number of threads that PowerExchange uses on the Integration Service machine to process bulk data. For optimal performance, this value should not exceed the number of installed or available processors on the Integration Service machine. Valid values are 1 through 64. Default is 0, which disables multithreading.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Database Name	DB2 subsystem or database instance name.
Compression	Select to compress source data during the PowerCenter session.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is disabled.

Property	Description
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.
Reject File	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the write mode is asynchronous with fault tolerance. To prevent the creation of the reject files, specify PWXDISABLE.
PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides: - `TCPIP_OP_TIMEOUT= network_operation_timeout ;` - `TCPIP_CON_TIMEOUT= connection_timeout`
Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.
Location	Required. Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.
Bulk Load	Whether PowerExchange loads data to the DB2 for z/OS targets with the DB2 LOAD utility. If you select this option, you can configure the remaining connection attributes that apply to the DB2 LOAD utility. Otherwise, PowerExchange ignores these attributes. Default is disabled.
File Name	The data set prefix PowerExchange uses to create the temporary files needed with the DB2 LOAD utility to load data into a DB2 table.
Space	Whether to allocate MVS space in tracks or cylinders: - TRACK. - CYLINDER. Default is TRACK.
Primary Space	Value for the primary space on MVS. Default is 5.
Secondary Space	Value for the secondary space on MVS. Default is 0.

Property	Description
Delete Temporary Files	How PowerExchange should handle the temporary files it creates with the DB2 LOAD utility to load data into a DB2 table. Select one of the following values: - NO. Does not delete the temporary files. - BEFORE. Deletes the temporary files before running the utility. - AFTER SUCCESS. Deletes the temporary files after running the utility if it ends with a return code 0. - AFTER. Deletes the temporary files after running the utility. Default is NO.
JCL Template	The name of the JCL template for the DB2 LOAD utility on the PowerExchange target system. Default is DB2LDJCL.
CTL Template	The name of the control file template for the DB2 LOAD utility on the PowerExchange target system. Default is DB2LDCTL.
Load Options	How PowerExchange should handle the data with the DB2 LOAD utility to load data into a DB2 table. Select one of the following values: - INSERT. - REPLACE. Default is INSERT.
Mode Type	How PowerExchange should run the DB2 LOAD utility to load data into a DB2 table. Select one of the following values: - TASK. Runs the LOAD utility as a sub-task under the PowerExchange Listener. - JOB. Submits a separate job to run the DB2 LOAD utility. - NOSUBMIT. Creates the files and JCL to run the DB2 LOAD utility, unless Mode Time is DATAONLY. Does not submit the load job. You must submit the job manually. Default is TASK.
Mode Time	How PowerExchange handles the execution of the DB2 LOAD utility. Select one of the following values: - WAIT. Waits for the job to end before returning control to PowerCenter. WAIT can only be specified with Mode Type JOB or TASK. - NO WAIT. Returns to PowerCenter without waiting for the job to end. NO WAIT can only be specified with Mode Type JOB or NO SUBMIT. - TIMED. Waits the number of seconds specified in Time before returning control to PowerCenter. TIMED can only be specified with Mode Type JOB. - DATA ONLY creates the data file only. Does not create the files and JCL to run the DB2 LOAD utility. Usually, this option is used with a Mode Type of NOSUBMIT. Default is WAIT. **Note:** If you enter WAIT, PowerExchange uses 99,999 seconds as the network operation timeout value instead of the value specified by the TCPIP_OP_TIMEOUT parameter of the PWX Override connection attribute. If you enter TIMED, PowerExchange adds 5 minutes to the network operation timeout value specified by the connection attribute.
Time	The wait time in seconds when you select JOB for the Mode Type and TIMED for Mode Time. Valid values are 1 to 99998. Default is 0.

PowerExchange Microsoft SQL Server Connections

You can create a PowerExchange Microsoft SQL Server connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange Microsoft SQL Server database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX Microsoft SQL Server.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database. **Note:** In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Array Size	Specifies the size of the storage array, in number of records, for Microsoft SQL Server bulk loads. Valid values are from 1 through 100000. Default is 50.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Database Name	Name of the database in the SQL Server instance.
Server Name	Name of the SQL Server instance.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.

Property	Description
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is disabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.
Reject File	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the write mode is asynchronous with fault tolerance. To prevent the creation of the reject files, specify PWXDISABLE.
PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides: - `TCPIP_OP_TIMEOUT= network_operation_timeout ;` - `TCPIP_CON_TIMEOUT= connection_timeout`
Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.
Location	Required. Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.

PowerExchange NRDB Batch Connections

You can create a PowerExchange NRDB Batch connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange NRDB Batch database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX NRDB Batch.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database. **Note:** In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Array Size	Storage array size, in number of records, that is used for partitioned or multithreaded sessions. For partitioned sessions, this array size is shared across the partitions. Valid values are from 1 through 100000. Default is 25. **Note:** To tune partitioned sessions, particularly when the Write Mode attribute specifies Confirm Write On, increase the array size.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.

Property	Description
Off Load Processing	Controls whether to use offload processing. Offload processing transfers PowerExchange bulk data processing from the source system to the Integration Service machine. Select one of the following values: - No. Do not use offload processing. - Auto. PowerExchange determines whether to use offload processing. - Filter After. Offloads the bulk data processing to the PowerCenter Integration Service machine, including the filtering of data. - Filter Before. Offloads processing to the PowerCenter Integration Service machine but continues to filter data on the source system. If you select Filter Before for an IMS unload file data source, PowerExchange changes the selection to Filter After. Default is No.
Worker Threads	The number of threads that PowerExchange uses on the Integration Service machine to process bulk data. For optimal performance, this value should not exceed the number of installed or available processors on the Integration Service machine. Valid values are from 1 through 64. Default is 0, which disables multithreading. If you use reader or writer pipeline partitioning, accept the default value of 0. You cannot use both multiple worker threads and partitioning.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is disabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.

Property	Description

PWX Override	PowerExchange connection override values, separated by semicolons. You can enter any of the following overrides:
	`TCPIP_OP_TIMEOUT=network_operation_timeout;`
	`TCPIP_CON_TIMEOUT=connection_timeout;`
	`WRT_ERROR_HANDLING=Y`
	If you use write partitioning and set the Write Mode attribute to Confirm Write On, you can specify `WRT_ERROR_HANDLIN=Y` to use an alternative method of error handling that is more efficient than the default method when the input data contains many errors. You can also specify this statement in the dbmover.cfg configuration file.

Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.
Location	Required. Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.
Get PowerExchange Log Entries	Specifies whether PWXPC writes PowerExchange messages related to a CDC session in the PowerCenter session log. Default is disabled.

PowerExchange Oracle Connections

You can create a PowerExchange Oracle connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange Oracle database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX Oracle.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.

Property	Description
Code Page	The code page that the Integration Service uses to extract data from the source database. **Note:** In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is enabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.
Reject File	Overrides the default prefix of PWXR for the reject file. PowerExchange creates the reject file on the target machine when the write mode is asynchronous with fault tolerance. To prevent the creation of the reject files, specify PWXDISABLE.

Property	Description

PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides:
	`- TCPIP_OP_TIMEOUT= network_operation_timeout ;`
	`- TCPIP_CON_TIMEOUT= connection_timeout`

Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. D Default is 0.
Location	Required. Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.
TNS Name	Oracle SID (instance name).

PowerExchange Sybase Connections

You can create a PowerExchange Sybase connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a PowerExchange Sybase database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select PWX Sybase.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database. **Note:** In Unicode mode, PWXPC sets the code page with this value overriding any code page specification in the PowerExchange configuration file.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.

Property	Description
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the data source or target if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Database Name	Name of the database in the Sybase instance.
Server Name	Name of the Sybase instance.
Compression	Select to compress source data during the PowerCenter session. Default is disabled.
Encryption Type	The type of encryption that the Integration Service uses. Select one of the following values: - None. - RC2. - DES. Default is None.
Encryption Level	The level of encryption that the Integration Service uses. If you select RC2 or DES for Encryption Type, select one of the following values to indicate the encryption level. - 1. Uses a 56-bit encryption key for DES and RC2. - 2. Uses a 168-bit triple encryption key for DES. Uses a 64-bit encryption key for RC2. - 3. Uses a 168-bit triple encryption key for DES. Uses a 128-bit encryption key for RC2. Ignored if you do not select an encryption type. Default is 1.
Interpret As Rows	Interprets the pacing size as rows or kilobytes. Select to represent the pacing size in number of rows. If you clear this option, the pacing size represents kilobytes. Default is enabled.
Convert Character Data As String	Convert character fields to string fields so that embedded nulls in data are processed as spaces. Default is disabled.
Write Mode	Mode in which the Integration Service sends data to the PowerExchange Listener. Configure one of the following write modes: - Confirm Write On. Sends data to the PowerExchange Listener and waits for a response before sending more data. Select if error recovery is a priority. This option might decrease performance. - Confirm Write Off. Sends data to the PowerExchange Listener without waiting for a response. Use this option when you can reload the target table if an error occurs. - Asynchronous With Fault Tolerance. Sends data to the PowerExchange Listener without waiting for a response. This option also provides the ability to detect errors. This provides the speed of Confirm Write Off with the data integrity of Confirm Write On. Default is Confirm Write On.
PWX Override	Specifies PowerExchange connection override values, separated by semicolons, including the following overrides: - `TCPIP_OP_TIMEOUT= network_operation_timeout ;` - `TCPIP_CON_TIMEOUT= connection_timeout`

Property	Description
Pacing Size	Amount of data that the source system can pass to the PowerExchange Listener. Configure the pacing size if an external application, database, or the Integration Service node is a bottleneck. The lower the value, the faster the performance. Enter 0 for maximum performance. Default is 0.
Location	Required. Location of the PowerExchange Listener node. The location is defined in the first parameter of the NODE statement in the PowerExchange dbmover.cfg configuration file.

SAS Analytics Connections

You can create an SAS Analytics connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for an SAS Analytics database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select SAS Analytics.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database.
Data Access Connection String	The connection string used to access data from the database. Enter <servername@dbname>.
Host Name	Host name of the machine that runs the SPI Server.
Port	Port number of the machine that runs the SPI Server. Must correspond to the port number used to configure the SPI Server.
Library Name	The library name from which PowerCenter reads or writes to the SAS table. You can override the library name in the session properties.
Post SAS Code	Value for the PostSASCode exit. Enter a SAS Macro or a SAS %include statement to run just after the reading of the SAS Source object. The SPI Server must be configured for SAS exits to run. The SAS Macro to be invoked must be defined in the autoexec.sas file that is part of the SPI Server configuration.

Property	Description
Pre SAS Code	Value for the PreSASCode exit. Enter a SAS Macro or a SAS %include statement to run just before the reading of the SAS Source object. The SPI Server must be configured for SAS exits to run. The SAS Macro to be invoked must be defined in the autoexec.sas file that is part of the SPI Server configuration.
Where Clause	Enter a selection of rows from the SAS source. This Where clause creates a WHERE statement in SAS.
Indexes	Index for the SAS connection definition.
Key Index	Key index for the SAS connection definition. The key index designates the key that identifies each row to use in an update else insert load.
Library Name PREC	Determines how the library name is represented in the mapping.

Sybase Connections

You can create a Sybase connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a Sybase database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Sybase.
Description	The description of the connection. The description cannot exceed 255 characters.
Use Kerberos Authentication	Enables Kerberos Authentication. You cannot enter a user name and password if you select this check box.
User Name	Required. The database user name.
User Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.

Property	Description
Metadata Connection String	Required. The JDBC connection URL used to access metadata from the database. Enter jdbc:informatica:sybase://<hostname>:5000;DatabaseName=<databasename>.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database. To run workflows with table or column names that are case sensitive, or contain special characters, or reserved keywords, you must set the value to *SET QUOTED_IDENTIFIER ON* .
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.
Database Name	Required. Name of the database. If you do not enter a database name, connection-related messages do not show a database name when the default database is used. Used for all operations.
Server Name	Required. Database server name.
Packet Size	Packet size for the transmission of data. Use to optimize the native drivers. Default is 0.

Teradata Connections

You can create a Teradata connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties for a Teradata database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Teradata.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page the Integration Service uses to read from a source database or write to a target database or file.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.
Environment SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the connection environment SQL each time it connects to the database.
Transaction SQL	SQL commands to set the database environment when you connect to the database. The PowerCenter Integration Service runs the transaction environment SQL at the beginning of each transaction.
Connection Retry Period	Number of seconds the Integration Service attempts to reconnect to the database if the connection fails. If the Integration Service cannot connect to the database in the retry period, the operation fails. Default is 0.

Property	Description
Database Name	Name of the database. This overrides the default database name in the ODBC entry. If you do not enter a database name, the Integration Service uses the default database name in the ODBC entry.
Data Source Name	Name of the Teradata ODBC data source.

Teradata FastExport Connections

You can create a Teradata FastExport connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties to access Teradata through FastExport:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Teradata FastExport.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	The code page that the Integration Service uses to extract data from the source database.
Database Name	The name of the Teradata database you want to connect to. The Integration Service generates the SQL statement using the database name as a prefix to the table name.
TDPID	Teradata database ID.

Property	Description
Tenacity	Number of hours that FastExport tries to log on to the Teradata database. When FastExport tries to log on but the maximum number of Teradata sessions is already running, FastExport waits for the amount of time defined by the Sleep option. After the Sleep time, FastExport tries to log on to the Teradata database again. FastExport repeats this process until it has either logged on for the required number of sessions or exceeded the Tenacity hours time period. Default is 4.
Max Sessions	Maximum number of FastExport sessions per FastExport job. Max Sessions must be between 1 and the total number of access module processes (AMPs) on your system. Default is 4.
Sleep	Number of minutes FastExport pauses before retrying a login. FastExport attempts a login until the login succeeds or the Tenacity hours elapse. Default is 6.
Data Encryption	Enables data encryption for FastExport. You can use data encryption with the version 8 Teradata client. Default is disabled.
Block Size	Maximum block size to use for the exported data. Default is 6400.
Log Table Name	Restart log table name. The FastExport utility uses the information in the restart log table to restart jobs that halt because of a Teradata database or client system failure. Each FastExport job should use a separate log table. If you specify a table that does not exist, the FastExport utility creates the table and uses it as the restart log. PowerCenter does not support restarting FastExport, but if you stage the output, you can restart FastExport manually.
Executable Name	Teradata command used to read the source data. Use the default value. Default is fexp.

Teradata FastLoad Connections

You can create a Teradata FastLoad connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties to access Teradata through FastLoad:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Teradata FastLoad.
Description	The description of the connection. The description cannot exceed 255 characters.

Property	Description
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Code Page	The code page that the Integration Service uses to extract data from the source database.
Database Name	The name of the Teradata database you want to connect to. The Integration Service generates the SQL statement using the database name as a prefix to the table name.
TDPID	Teradata database ID.
Tenacity	Number of hours that FastLoad tries to log in to the required FastLoad sessions when the maximum number of load jobs are already running on the Teradata database. When FastLoad tries to log in to a new session, and the Teradata database indicates that the maximum number of load sessions is already running, FastLoad logs off all new sessions that were logged in, delays for the number of minutes specified in the Sleep attribute, and then retries the login. FastLoad keeps trying until it logs in for the required number of sessions or exceeds the number of hours specified in the Tenacity attribute. Default is 4.
Max Sessions	Maximum number of FastLoad sessions per FastLoad job. Max Sessions must be between one and the total number of access module processes (AMPs) on your system. Default is 1.
Sleep	Number of minutes FastLoad pauses before retrying a login. FastLoad attempts a login until the login succeeds or the Tenacity hours elapse. Default is 6.
Error Limit	Maximum number of rows that FastLoad rejects before it stops loading data to the database table. Default is 1,000,000.
Check Point	Number of rows transmitted to the Teradata database between checkpoints. If processing stops while a FastLoad job is running, you can restart the job at the most recent checkpoint. Default is 0. If you enter 0, FastLoad does not perform checkpoint operations.
Drop Error Tables	Drops the FastLoad error tables before beginning the next session. FastLoad will not run if non-empty error tables exist from a prior job. Select this option to drop the tables, or clear it to keep the tables.
External Loader Executable	Name and optional file path of the Teradata external loader executable. If the external loader executable directory is not in the system path, you must enter the full path.
Is Staged	Method of loading data. Select Is Staged to load data to a flat file staging area before loading to the database. Otherwise, the data is loaded to the database using a named pipe.

Property	Description
Error Database	Error database name. Use this attribute to override the default error database name. If you do not specify a database name, the Integration Service uses the target table database.
Truncate Target Table	Truncates the target database table before beginning the FastLoad job. FastLoad cannot load data to non-empty tables.

Teradata MultiLoad Connections

You can create a Teradata MultiLoad connection in Test Data Manager to perform data subset and data masking operations.

The following table describes the database connection properties to access Teradata through MultiLoad:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Teradata MultiLoad.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Database Name	Name of the database. If you do not specify a database name, the Integration Service uses the target table database name defined in the mapping.
TDPID	Teradata database ID.
Tenacity	Amount of time, in hours, MultiLoad tries to log in to the required sessions. If a login fails, MultiLoad delays for the number of minutes specified in the Sleep attribute, and then retries the login. MultiLoad keeps trying until the login succeeds or the number of hours specified in the Tenacity attribute elapses. Default is 10000.

Property	Description
Max Sessions	Maximum number of MultiLoad sessions per MultiLoad job. Max Sessions must be between 1 and 32,767. Running multiple MultiLoad sessions causes the client and database to use more resources. Therefore, setting this value to a small number may improve performance. Default is 1.
Sleep	Number of minutes MultiLoad waits before retrying a login. MultiLoad tries until the login succeeds or the number of hours specified in the Tenacity attribute elapses. Sleep must be greater than 0. If you specify 0, MultiLoad issues an error message and uses the default value, 6 minutes. Default is 6.
Log Table Name	Log table database name. Use this attribute to override the default log table database name. If you do not specify a database name, the Integration Service uses the target table database.
Date Format	Date format. The date format in the connection object must match the date format you define in the target definition. The Integration Service supports the following date formats: - DD/MM/YYYY - MM/DD/YYYY - YYYY/DD/MM - YYYY/MM/DD
Error Limit	Total number of rejected records that MultiLoad can write to the MultiLoad error tables. Uniqueness violations do not count as rejected records. An error limit of 0 means that there is no limit on the number of rejected records.
Check Point	Interval between checkpoints. You can set the interval to the following values: - 60 or more. MultiLoad performs a checkpoint operation after it processes each multiple of that number of records. - 1-59. MultiLoad performs a checkpoint operation at the specified interval, in minutes. - 0. MultiLoad does not perform any checkpoint operation during the import task.
Load Mode	Mode to generate SQL commands: Insert, Delete, Update, Upsert, or Data Driven. When you select Data Driven loading, the Integration Service follows instructions in an Update Strategy or Custom transformation to determine how to flag rows for insert, delete, or update. The Integration Service writes a column in the target file or named pipe to indicate the update strategy. The control file uses these values to determine how to load data to the target. Default is Upsert.
Drop Error Tables	Drops the MultiLoad error tables before beginning the next session. Select this option to drop the tables or clear it to keep the tables. Default is disabled.
External Loader Executable	Name and optional file path of the Teradata external loader executable. If the external loader executable directory is not in the system path, you must enter the full path.
Is Staged	Method of loading data. Select Is Staged to load data to the flat file staging area before loading to the database. Otherwise, the data is loaded to the database using a named pipe. Default is disabled.
Error Database	Error database name. Use this attribute to override the default error database name. If you do not specify a database name, the Integration Service uses the target table database.

Property	Description
Work Table Database	Work table database name. Use this attribute to override the default work table database name. If you do not specify a database name, the Integration Service uses the target table database.
User Variables	User-defined variable used in the default control file.

Teradata Parallel Transporter Connections

You can create a Teradata Parallel Transporter connection in Test Data Manager to perform data subset and data masking operations. You cannot perform inplace masking and other tasks that require update override queries on Teradata Parallel Transporter connections.

The following table describes the database connection properties for a Teradata Parallel Transporter database:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. It cannot exceed 128 characters, start with a number, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Teradata Parallel Transporter.
Description	The description of the connection. The description cannot exceed 255 characters.
User Name	Required. The database user name.
Use Parameter in Password	Indicates the password for the database user name is a session parameter. $Param*Name*. Define the password in the workflow or session parameter file, and encrypt it using the *pmpasswd* CRYPT_DATA option. Used for data subset and data masking operations. Default is disabled.
Password	Required. The password for the database user name.
Owner	The owner of the connection. Default is the user that creates the connection. You can change the owner of the connection.
JDBC Login Password	Required if Use Parameter in Password is selected. The password for the JDBC user. Used for import from source and data discovery operations.
Code Page	Code page associated with the database. When you run a session that extracts from a Teradata source, the code page of the Teradata PT API connection must be the same as the code page of the Teradata source.
Data Access Connection String	The connection string used to access data from the database. Enter <database name>.

Property	Description
Database Name	Teradata database name. If you do not enter a database name, Teradata PT API uses the default login database name.
TDPID	Name of the Teradata database machine.
Tenacity	Amount of time, in hours, that Teradata PT API continues trying to log on when the maximum number of operations runs on the Teradata database. Must be a positive, non-zero integer. Default is 4.
Max Sessions	Maximum number of sessions that Teradata PT API establishes with the Teradata database. Must be a positive, non-zero integer. Default is 4.
Min Sessions	Minimum number of Teradata PT API sessions required for the Teradata PT API job to continue. Must be a positive integer between 1 and the Max Sessions value. Default is 1.
Sleep	Amount of time, in minutes, that Teradata PT API pauses before it retries to log on when the maximum number of operations runs on the Teradata database. Must be a positive, non-zero integer. Default is 6.
Data Encryption	Enables full security encryption of SQL requests, responses, and data. Default is disabled.
Block Size	Maximum block size, in bytes, Teradata PT API uses when it returns data to the Integration Service. Default is 6400.
System Operator	System operator to extract or load data. Select one of the following values: - Stream. Performs update, insert, upsert, and delete operations against Teradata database tables in near real-time mode. - Load. Bulk loads data into an empty Teradata database table. - Update. Performs update, insert, upsert, and delete operations against Teradata database tables. - Export. Extracts data from Teradata. Default is Stream.

CHAPTER 7

Dictionaries

This chapter includes the following topics:

- Dictionaries Overview, 95
- Relational Dictionaries, 95
- Flat File Dictionaries, 96
- Dictionary Management, 97

Dictionaries Overview

A dictionary is a flat file or relational table that contains substitute data and a serial number. You can use a dictionary to replace sensitive data in a table. You can use data from a dictionary to generate data.

Add a dictionary in the **Administrator | Dictionaries** view that you want to use in data masking or data generation rules. When you create a masking rule, you can define the dictionary that you want to use to mask sensitive data. Before you create a data generation rule, you must add dictionary to TDM.

The PowerCenter Integration Service generates a number and retrieves a dictionary row based on the serial number from the dictionary. It generates a hash key for repeatable masking or a random number for non-repeatable masking. You can configure an additional lookup condition if you configure repeatable masking.

When you use dictionaries for masking Hadoop sources, you need to activate the Content Management Service because the dictionary reference tables are created in the Content Management Service database.

You can create a generation rule to load the dictionary into target databases. The binary data type contains a serial number column that increases performance. You can configure serial number for relational databases when you create a data generation rule for binary data type.

Relational Dictionaries

A relational dictionary is a database table that you use as a dictionary. You can use a relational dictionary with email address and substitution masking. Use a relational dictionary when you create generation rules for string, date, numeric, and binary datatypes.

When you add a relational dictionary, you must define the datasource.

Adding a Relational Dictionary

Add a relational dictionary to use in masking rules. When you add a relational dictionary, you can define the connection to the dictionary.

1. In the **Administrator | Dictionaries** view, click **Actions** > **New Dictionary**.

   The **New Dictionary** tab appears.
2. Enter the name of the dictionary, an optional description of the dictionary, and the type of the dictionary.
3. Click **Select** to define a connection.

   The **Select Relational Dictionary** dialog box appears.
4. Select a datasource connection from the menu, and click **Next**.
5. Select a datasource, and click **Next**.
6. Select a table from the list of tables in the datasource, and click **Finish**.
7. Review the **Connection**, **Schema**, and **Table** properties you selected.
8. Click **Save**.

   A tab with the dictionary properties opens and the dictionary appears in the **Administrator | Dictionaries** view.

Flat File Dictionaries

A flat file dictionary is a file that contains dictionary information. You can use a flat file dictionary with email address and substitution masking. You can use a flat file dictionary to generate data for string, numeric, and date datatypes.

TDM installs with sample dictionary files that you can use for names and addresses. You can find the dictionary files in the following location:

```
<TDM installation directory>/TDM/infa_shared/LkpFiles
```

When you use a dictionary file in a masking or a generation rule, the Test Data Manager imports the structure of the dictionary file and stores the column names.

The following text shows a sample flat file dictionary that contains first name and gender:

```
SNO,GENDER,FIRSTNAME
1,M,Adam
2,M,Adeel
3,M,Adil
4,F,Alice
5,F,Alison
```

In this dictionary, the first field in the row is the serial number, and the second field is gender. The PowerCenter Integration Service looks up a dictionary record by serial number. You can add gender as a lookup condition. The PowerCenter Integration Service retrieves a row from the dictionary using a hash key, and it finds a row with a gender that matches the gender in the source data.

Flat File Dictionary Properties

Specify flat file dictionary properties to add a dictionary in the Test Data Manager.

The following table describes the properties of a flat file dictionary:

Property	Description
Name	The name of the dictionary.
Description	An optional description of the dictionary.
Type	The type of dictionary. Select Flat File.
File Location	The absolute path to the dictionary to import the file.
Overwrite File if Already Exists	Enable to overwrite a dictionary with the same name.
File Encoding	The file encoding type.

Adding a Flat File Dictionary

Add a flat file dictionary to use in masking rules. When you add a flat file dictionary, you can browse for and select a dictionary from your local computer.

1. In the **Administrator | Dictionaries** view, click **Actions** > **New Dictionary**.

   The **New Dictionary** tab appears.
2. Enter a name and optional description for the dictionary. Select **Flat File** as the dictionary type.
3. Click **Choose File** to browse to the location of the dictionary. Select the dictionary file, and click **Open**.
4. Optionally, select to overwrite the file if a dictionary with the same name exists.
5. Select the file encoding type.
6. Click **Save**.

   A tab with the dictionary properties opens and the dictionary appears in the **Administrator | Dictionaries** view.

Dictionary Management

You can add, edit, delete, sort, and filter dictionaries in the **Administrator | Dictionaries** view.

The **Administrator | Dictionaries** view contains a list of imported dictionaries. You can view the dictionary description and the dictionary type. You can filter the dictionaries by name, description, and type.

Select a dictionary to view the dictionary properties in the **Properties** pane below the dictionary list. Click **Used in Rules** to view the rules that use the dictionary. Click **Used in Projects** to view the projects that have rule assignments that use the dictionary.

You can edit and delete a dictionary through the **Actions** menu.

The following image shows the **Administrator | Dictionaries** view:

CHAPTER 8

Configuring the Test Data Warehouse

This chapter includes the following topics:

- Test Data Warehouse Overview, 99
- Test Data Warehouse Architecture, 99
- Test Data Warehouse Configuration, 101

Test Data Warehouse Overview

Create a test data warehouse to create a central and secure location to store, manage, and share test data. Store versions of test data as data sets. You can share data sets across users and reset a data set to a target to restore a test system to a required state.

Create the test data warehouse from the **Administrator** view in Test Data Manager.

The test data warehouse repository is a relational database that stores the source table metadata and the project metadata generated when you create a data set.

The test data warehouse stores the source data that you include in the data set.

To create a test data warehouse, you must enable the test data warehouse option when you create the Test Data Manager Service.

For information about how to enable the test data warehouse when you create the Test Data Manager Service, see the Test Data Management *Installation Guide*.

Note: If you delete the Test Data Manager Service and create it again after you have configured a test data warehouse, you must configure the test data warehouse again.

Test Data Warehouse Architecture

The test data warehouse architecture consists of a tool, server, services, and databases.

The tools component consists of the TDM client Test Data Manager. The TDM server forms the server component. The services component consists of the Test Data Manager Service and PowerCenter services.

The databases component consists of the TDM repository, test data warehouse repository, test data warehouse, and the PowerCenter repository.

The following image shows the architecture components for the test data warehouse:

When you run a plan to create a data set, the Test Data Manager Service stores the project and source table metadata in the test data warehouse repository. The service creates the tables in the test data warehouse. The PowerCenter Integration Service then moves the data into the tables in the test data warehouse.

The following table describes the architecture components:

Component	Description
Test Data Manager	A web-based client that you can use to perform data discovery, data subset, data masking, and data generation operations.
TDM server	Server that runs TDM and integrates with Informatica application services to perform data discovery, data subset, data masking, and data generation operations.
Test Data Manager Service	An application service that runs Test Data Manager and manages connections between service components and Test Data Manager users.
PowerCenter Repository Service	An application service that manages the PowerCenter repository where the metadata for workflows and mappings that you generate in Test Data Manager are stored.
PowerCenter Integration Service	An application service that runs sessions and workflows that you run from Test Data Manager.
TDM repository	A relational database that stores the components that you define in Test Data Manager, such as policies, projects, entities, and rules. The TDM repository stores metadata that you import into Test Data Manager from a source database or from the PowerCenter repository.
Test data warehouse repository	A relational database that stores the source table metadata and project metadata for data sets that you create in Test Data Manager.
Test data warehouse	A relational database that stores the source data that you include in a data set.
PowerCenter repository	A relational database that stores the metadata for workflows and mappings that you generate from Test Data Manager.

Test Data Warehouse Configuration

You must configure relational databases for the test data warehouse repository and the test data warehouse before you can store data in the test data warehouse. Multiple machines that host a Test Data Manager Service in a domain can use the same test data warehouse.

You can create a test data warehouse or add a test data warehouse that you configured for a different machine that hosts a Test Data Manager Service. Add an existing test data warehouse to store data from different machines that host a Test Data Manager Service in the same test data warehouse.

You can link a test data warehouse repository to a single test data warehouse.

You can create a test data warehouse repository on an Oracle, Microsoft SQL Server, or IBM DB2 database.

You can create a test data warehouse on an Oracle database.

Creating a Test Data Warehouse

To create a test data warehouse, configure a relational database to store the metadata and a database to store the data sets.

Create a connection in Test Data Manager for the database that you want to use as the test data warehouse.

1. From the **Actions** menu in the **Administrator | Test Data Warehouse** view, select **New Repository**.

   The **New Test Data Warehouse Repository** dialog box opens.
2. Select **New Repository**.
3. Enter a name for the test data warehouse repository.
4. Select the database type from the list of supported databases.
5. Enter the database user name and the password.
6. Enter the JDBC URL to connect to the database. Click **Next**.
7. Click **Add** to add a test data warehouse.
8. Enter a name and optional description for the test data warehouse.
9. Select the database type from the list of supported databases.
10. Select the connection that you created for the test data warehouse from the list of connections.
11. Click **OK** to create the test data warehouse.

    The test data warehouse appears in the list.
12. Select the test data warehouse from the list and click **Finish** to create the test data warehouse repository content.

The test data warehouse information appears in the **Test Data Warehouse** view.

Adding an Existing Test Data Warehouse

You can add a test data warehouse that you configured for a different Test Data Manager Service in the domain. Add an existing test data warehouse when you have more than one Test Data Manager Service within a domain, and you want all of the services to connect to the same test data warehouse.

Create a connection in Test Data Manager for the database that you want to use as the test data warehouse.

1. From the **Actions** menu in the **Administrator | Test Data Warehouse** view, select **New Repository**.

   The **New Test Data Warehouse Repository** dialog box opens.
2. Select **Use Existing Repository**.

3. In the **Test Data Warehouse Repository URL** field, enter the URL of the test data warehouse repository that you want to connect to, and click **Next**.
4. Select the test data warehouse from the list and click **Finish**. If you add a different connection as the test data warehouse, the change affects all machines that you have configured to use the same test data warehouse repository.

The test data warehouse information appears in the **Test Data Warehouse** view.

Editing a Test Data Warehouse Repository Password

If you change the database password, you must also update the password in Test Data Manager. Workflows will fail if you do not update the password in Test Data Manager.

Edit the password in the database before you edit the password in Test Data Manager.

1. To edit the test data warehouse repository password, click **Edit** on the **Test Data Warehouse Repository** tab in the **Administrator | Test Data Warehouse** view.

   The **Edit - Test Data Warehouse Repository Settings** dialog box opens.
2. Enter the password.
3. Click **OK**.

Editing a Test Data Warehouse Password

If you change the database password, you must also update the password and refresh the test data warehouse connection in Test Data Manager.

Edit the password in the database before you edit the password in Test Data Manager.

1. From the **Administrator | Connections** page, click the connection that you configured for the test data warehouse.

   The connection opens in a separate page.
2. Click **Actions** > **Edit**.

   The **Edit Connection** dialog box opens.
3. Enter the password that you entered in the database.
4. Test the connection and then click **Save**.
5. To refresh the test data warehouse, open the **Administrator | Test Data Warehouse** view.
6. Click **Edit** on the **Test Data Warehouse** tab.

   The **Edit Test Data Warehouse** dialog box opens.
7. Select the test data warehouse connection from the list of target connections.

   Click **OK**.

CHAPTER 9

User-Defined Datatypes

This chapter includes the following topics:

- User-Defined Datatypes Overview, 103
- Properties File, 103
- Creating Functions for User-Defined Datatypes in Oracle Applications, 105

User-Defined Datatypes Overview

A user-defined datatype (UDT) contains data with a format or structure defined by a user.

You can read from a relational source or write to a relational target that contains user-defined datatypes. Since the user-defined datatype is not a standard data type defined by the database server, you must provide a conversion method so that the PowerCenter Integration Service can process the data correctly.

You can configure a properties file to define how to convert user-defined datatypes. There must be a separate properties file for each type of database used as a source or target.

To use user-defined datatypes with Oracle Applications, run SQL scripts to create functions in the database and add additional lines to the properties file.

Properties File

The user-defined data type properties file contains the list of user-defined data types and the parameters required to convert the data types to a format that the PowerCenter Integration Service can process.

The name of the properties file must use the following format:
*DatabaseType*_UDT_Native_Conversions.properties

You can use a properties file to define conversion parameters for the following database types:

- Oracle. The properties file must be named Oracle_UDT_Native_Conversions.properties.
- Microsoft SQL Server. The properties file must be named Microsoft_SQL_Server_UDT_Native_Conversions.properties.
- IBM DB2. The properties file must be named DB2_UDT_Native_Conversions.properties.
- Sybase. The properties file must be named Sybase_UDT_Native_Conversions.properties.

The PowerCenter Integration Service looks for the properties files in the following folder:

```
<TDMInstallationDirectory>/utilities/mapgen/config
```

UDT functions for Oracle are available in TDM. To handle user-defined data types on other databases, you must write the functions.

Datatype Conversion Parameters

The properties file uses parameters to define how the PowerCenter Integration Service converts user-defined datatypes.

The datatype conversion parameters in the properties file must be in the following format:

```
UDTname=Datatype;Precision;SourceSQLFunction;TargetSQLFunction
```

The following table describes the parameters in the properties file:

Parameter	Description
UDTName	Name of the user-defined datatype used in the data source or target.
Datatype	Datatype to which the user-defined datatype is converted in PowerCenter.
Precision	Precision of the data after conversion.
SourceSQLFunction	SQL function used to convert the user-defined datatype in the data source to the datatype in PowerCenter. The source database must support the SQL function. The SQL functions that you can use depend on the database type of the data source.
TargetSQLFunction	SQL function used to convert the datatype in PowerCenter to the user-defined datatype in the target. The target database must support the SQL function. The SQL functions that you can use depend on the database type of the target.

The following example shows the datatype conversion parameters defined in a user-defined datatype properties file:

```
AMV_AQ_MSG_OBJECT_TYPE=clob;4000;SYS_XMLGEN(%port
%).GETCLOBVAL();CX_AMVAQMSGOBJECTTYPE(%port%)
AQ$_JMS_TEXT_MESSAGE=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();CX_AQ$JMSTEXTMESSAGE(%port
%)
AQ$_SIG_PROP=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();CX_AQ$SIGPROP(%port%)
AR_REV_REC_TYP=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();CX_ARREVRECTYP(%port%)
ASO_ORDER_FEEDBACK_TYPE=clob;4000;SYS_XMLGEN(%port
%).GETCLOBVAL();CX_ASOORDERFEEDBACKTYPE(%port%)
CACHESYNC_MESSAGE=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();CX_CACHESYNCMESSAGE(%port%)
XNP_MESSAGE_TYPE=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();CX_XNPMESSAGETYPE(%port%)
ROWID=varchar;255;ROWIDTOCHAR(%port%);CHARTOROWID(%port%)
TIMESTAMP(6)=varchar;255;to_char(%port%,'dd/mm/yyyy
hh24:mi.ss.ffffff');to_timestamp(%port%, 'dd/mm/yyyy hh24:mi.ss.ffffff')
xmltype=clob;4000;SYS_XMLGEN(%port%).GETCLOBVAL();xmltype(%port%)
```

Setting Up the Properties File for User-Defined Datatypes

If you use sources and targets that contain user-defined datatypes, you must set up the properties file to define how to convert the datatypes.

1. Verify that the TDM installation directory contains the following folder: `<TDMInstallationDirectory>/utilities/mapgen/config`

   If the folder does not exist, create it.

2. Create the user-defined datatype properties file.

   The properties file name must match the database for the source or target.

3. Add the list of user-defined datatypes to the properties file.

   Set the parameters for the user-defined datatypes.

When you generate a workflow, the PowerCenter Integration Service checks for user-defined datatypes in the sources and targets. It uses the conversion information in the properties file to convert user-defined datatypes in sources to PowerCenter datatypes and convert PowerCenter datatypes to user-defined datatypes in targets.

Creating Functions for User-Defined Datatypes in Oracle Applications

If you have a user-defined datatype and you want to use data subset or data masking with Oracle Applications, you must run SQL scripts to create functions in the database and print output that you add to the Oracle_UDT_Native_Conversions.properties file.

1. Run the CreateReplaceClobString.sql SQL script. You can find the SQL script in the following location:

   ```
   <TDMInstallationDirectory>\Content\Accelerators\Oracle Apps\SQL\
   ```

2. Run the OracleUDT_FunctionCreation.sql SQL script. OracleUDT_FunctionCreation.sql is in the same directory as CreateReplaceClobString.sql.

   OracleUDT_FunctionCreation.sql creates TargetSQLFunction and prints output.

3. Open the Oracle_UDT_Native_Conversions.properties file and add the output from OracleUDT_FunctionCreation.sql to the properties file.

CHAPTER 10

tdwcmd

This chapter includes the following topics:

- tdwcmd Overview, 106
- Configuring tdwcmd, 106
- Running tdwcmd, 107
- Entering Options and Arguments, 107
- Syntax Notation, 108
- Version, 108
- CreateTDWREPO, 109
- CreateTestDataWarehouse, 111
- List, 114

tdwcmd Overview

tdwcmd is a command line program that you can use to perform test data warehouse tasks.

Use the tdwcmd command to perform the following tasks:

- View the version details of the client on which the Test Data Manager Service is running.
- Create a test data warehouse repository.
- Create a test data warehouse.
- List the test data warehouse repository, test data warehouse, connection, and data set details.

You must have the required privileges to perform these tasks. For tasks related to projects, you must also have the required permission on the project.

Configuring tdwcmd

When TDM runs on HTTPS, you can configure tdwcmd before you run the tdwcmd commands.

Before you configure tdwcmd, import the browser certificate to the following location:

```
<TDM installation directory>/TDM/utilities/tdwcli/conf
```

To generate a keystore, you must import the browser certificate. The keystore is necessary to run the tdwcmd commands.

1. To set the password and generate a keystore, run the following command:

   ```
   keytool -import -file <Imported certificate> -keystore client.ks
   ```

2. Edit the userConfig.ilm file and add the keystore location for the javax.net.ssl.trustStore parameter. You can find the userConfig.ilm file in the following location:

   ```
   <TDM installation directory>/TDM/utilities/tdwcli/conf/userConfig.ilm
   ```

3. Add the password that you used to create the keystore for the javax.net.ssl.trustStorePassword parameter.

Running tdwcmd

Invoke tdwcmd from the command line. You can issue commands directly or from a script, batch file, or other program.

On Windows, tdwcmd is a batch file with a .bat extension. On UNIX, tdwcmd is a shell script with no extension.

1. At the command line, switch to the directory where the tdwcmd executable file is located.

   By default, tdwcmd is installed in the following directory:

   ```
   <TDM installation directory>\utilities\tdwcli\bin
   ```

2. Enter tdwcmd followed by the command name and the required options and arguments.

   For example:

   ```
   tdwcmd -command_name [-option1] argument_1 [-option2] argument_2...
   ```

To view the command line syntax, enter the following command:

```
tdwcmd -h
```

Entering Options and Arguments

The tdwcmd command line program uses a set of options and arguments.

Use the following rules when you enter the tdwcmd command options and arguments:

- To enter options, type a hyphen followed by the program syntax for the command. Use two consecutive hyphens for the long name of the option, and use a single hyphen for the short name of the option.
- You can enter options in any order.
- If an argument contains spaces, enclose the argument in double quotes.
- The first word after the option is the argument.
- Most options require arguments. You must separate options from arguments with a single space.
- Commands, options, and object names are case sensitive.

Syntax Notation

To use the tdwcmd command line program, review the syntax notation.

The following table describes the syntax notation for the tdwcmd command line program:

Convention	Description
--x or -x	An option placed before an argument. The option designates the parameter that you enter. For example, to enter the user name for tdwcmd, type --UserName or -un followed by the user name.
<x>	Required option. If you omit a required option, the command line program returns an error message.
<x | y>	Select between required options. If you omit a required option, the command line program returns an error message. If a pipe symbol (|) separates options, specify one option. If there are no pipe symbols to separate the options, specify all the options.
[x]	Optional option. The command runs whether or not you enter these options.
[x | y]	Select between optional options. For example, you can display help for all tdwcmd commands by using the --Help or -h option. `[--Help|-h]` The command runs whether or not you enter the optional parameter.
< <x | y> **|** <a | b> **>**	When a set contains subsets, indicate the supersets with bold brackets (**< >**). Separate the subsets with the bold pipe symbol (|).

Version

You can print the version information for the client and the version on which the Test Data Manager Service is running.

With the tdwcmd version command, you can perform the following tasks:

- Find the version of the client on which the Test Data Manager Service is running.
- Print the version information of the client.

The tdwcmd version command uses the following syntax:

```
tdwcmd
<--version>
```

The following table describes tdwcmd CreateTDWREPO options and arguments:

Option	Argument	Description
--version	n/a	View and print the version details of the client on which the Test Data Manager Service is running.

CreateTDWREPO

Create a test data warehouse repository to store the metadata for the data in the test data warehouse.

With the CreateTDWREPO command, you can perform the following tasks:

- Create a test data warehouse repository.
- Configure Test Data Manager Service details.
- Configure user authentication parameters.

The CreateTDWREPO command uses the following syntax:

```
tdwcmd
<--CreateTDWREPO | -CreateTDWREPO>
<--db_type | -dt> db_type
[--customDriver | -cd] customDriver
<--DBUsername | -du> DBUsername
<--DBPassword | -dp> DBPassword
<--DBUrl | -dl> DBUrl
<--DBConnString | -dc> DBConnString
[--DbSchema | -ds] DbSchema
[--DbTablespace | -dts] DbTablespace
[--idleTimeout | -it] idleTimeout
[--connPoolSize | -cp] connPoolSize
<--ServiceName | -sn> ServiceName
[--Help | -?,-h]
[--Verbose | -v]
<--HostName | -host_name> HostName
<--HTTPPort | -http_port> HTTPPort
<--HTTPSPort | -https_port> HTTPSPort
<--UserName | -un> UserName
<--Password | -pd> Password
<--SecurityDomain | -sdn>
SecurityDomain
<--DomainName | -dn> DomainName
<--TDMServiceName | -tsn> TDMServiceName
```

The following table describes the CreateTDWREPO command options and arguments:

Option	Argument	Description
--db_type -dt	db_type	Required. The type of database. You can use the following types of databases: Oracle, DB2, and SQL server.
--customDriver -cd	customDriver	The class name of the Custom JDBC driver that connects to the database. Required if you want to use a custom database driver.
--DBUsername -du	DBUsername	Required. Account for the repository database. Use the database client to set up this account.
--DBPassword -dp	DBPassword	Required. Repository database password for the database user.
--DBUrl -dl	DBUrl	Required. JDBC connect string to the database for the test data warehouse repository.
--DBConnString -dc	DBConnString	Required. The connection string to connect to the database.

Option	Argument	Description
--DbSchema -ds	DbSchema	Optional. The schema name for a Microsoft SQL Server database.
--DbTablespace -dts	DbTablespace	Required for a DB2 database. The name of the table space to create the tables.
--connPoolSize -cp	connPoolSize	Optional. Maximum number of idle connection instances that the Test Data Manager Service maintains for a database connection.
--idleTimeout -it	idleTimeout	Optional. The idle connection timeout for the database.
--ServiceName -sn	ServiceName	Required. The name of the test data warehouse repository.
--Help -?,-h	n/a	Optional. Displays or prints the help options.
--Verbose -v	n/a	Optional. Verbose output. Displays more than the minimum information about the object.
--HostName -host_name	HostName	Required. The TDM host name.
--HTTPPort -http_port	HTTPPort	Required if you do not specify an HTTPS port. Port number to connect to the Test Data Manager Service.
--HTTPSPort -https_port	HTTPSPort	Required if you do not specify an HTTP port. Port number to secure the connection to the Test Data Manager Service.
--UserName -un	UserName	Required. The Informatica domain user name.
--Password -pd	Password	Required. The password for Informatica domain user.
--SecurityDomain -sdn	SecurityDomain	Required. The Informatica security domain.
--DomainName -dn	DomainName	Required. The name of the TDM domain.
--TDMServiceName -tsn	TDMServiceName	Required. The name of the Test Data Manager Service.

CreateTDWREPO Examples

The following sample command creates the test data warehouse repository TDWREPO with Oracle database type:

```
tdwcmd --CreateTDWREPO --db_type oracle --DBUsername orauser --DBPassword orapass --
DBUrl jdbc:informatica:oracle://localhost:1521;SID=orcl --DBConnString orcl --
connPoolSize 5 --idleTimeout 320 --ServiceName TDWREPO -host_name localhost -http_port
```

```
6605 --UserName Administrator --Password Administrator --SecurityDomain Native -dn
Domain_SandboxVMLin_TDM961HF14  -tsn TDMS
```

The following sample command creates the test data warehouse repository TDWREPO with Microsoft SQL Server database type:

```
tdwcmd --CreateTDWREPO --db_type sqlserver --DBUsername <sqluser> --DBPassword
<sqlpassword> --DBUrl jdbc:informatica:sqlserver://<serverName>:
1433;SelectMethod=cursor;databaseName=<dbname> --DBConnString <serverName>@<dbName> --
connPoolSize 5 --ServiceName TDWREPO -host_name localhost -http_port 6605 --UserName
Administrator --Password Administrator --SecurityDomain Native -dn
Domain_SandboxVMLin_TDM961HF14 -tsn TDMS
```

CreateTestDataWarehouse

Create a test data warehouse to store the data.

With the CreateTestDataWarehouse command, you can perform the following tasks:

- Create a test data warehouse.
- Configure Test Data Manager Service details.
- Configure user authentication parameters.

The CreateTestDataWarehouse command uses the following syntax:

```
tdwcmd
<--CreateTestDataWarehouse | -CreateTestDataWarehouse>
<--TestDataWarehouseName | -twn> TestDataWarehouseName
<--DBUsername | -du> DBUsername
[--PasswordAsParam | -pp] PasswordAsParam
<--DBPassword | -dp> DBPassword
<--DBUrl | -dl> DBUrl
<--DBConnString | -dc> DBConnString
[--PassThroughSecurity | -pts] PassThroughSecurity
[--connection_environment_SQL | -ces] connection_environment_SQL
[--transaction_environment_SQL | -tes] transaction_environment_SQL
<--CodePage | -cp> CodePage
<--Kerberos | -kerberos> Kerberos
[--ConnectionRetryPeriod | -crp] ConnectionRetryPeriod
[--EnableParallelMode | -epm] EnableParallelMode
[--Help | -?,-h]
[--Verbose | -v]
<--HostName | -host_name> HostName
[--HTTPPort | -http_port] HTTPPort
[--HTTPSPort | -https_port] HTTPSPort
<--UserName | -un> UserName
<--Password | -pd> Password
<--SecurityDomain | -sdn>
SecurityDomain
<--DomainName | -dn> DomainName
<--TDMServiceName | -tsn> TDMServiceName
```

The following table describes the CreateTestDataWarehouse command options and arguments:

Option	Argument	Description
--TestDataWarehouseName -twn	TestDataWarehouseName	Required. The name of the test data warehouse.
--DBUsername -du	DBUsername	Required. Account for the test data warehouse. Use the database client to set up this account.
--PasswordAsParam -pp	PasswordAsParam	Optional. You can use a parameter for the password. Set the PasswordASParam option to True or False.
--DBPassword -dp	DBPassword	Required. The password for the database user.
--DBUrl -dl	DBUrl	Required. JDBC connect string to the database for the test data warehouse.
--DBConnString -dc	DBConnString	Required. The connection string to connect to the database.
--PassThroughSecurity -pts	PassThroughSecurity	Optional. Enable or disable pass-through security for the connection. When you enable pass-through security for a connection, the domain uses the client user name and password to log in to the corresponding database, instead of the credentials defined in the connection object.
--connection_environment_SQL -ces	connection_environment_SQL	Optional. SQL commands to set the database environment when you connect to the database. The Data Integration Service runs the connection environment SQL each time it connects to the database.
--transaction_environment_SQL -tes	transaction_environment_SQL	Optional. SQL commands to set the database environment when you connect to the database. The Data Integration Service runs the transaction environment SQL at the beginning of each transaction.
--CodePage -cp	CodePage	Required. The code page to read from or write to the database. To enter a code page description that contains a space or other nonalphanumeric character, enclose the name in quotation marks.
--Kerberos -kerberos	Kerberos	Required if you want to use Kerberos authentication. Set the Kerberos authentication to True or False.
--ConnectionRetryPeriod -crp	ConnectionRetryPeriod	Optional. The number of seconds that the service tries to reconnect to the database if the connection fails. If the service cannot connect to the database in the retry period, the operation fails.

Option	Argument	Description
--EnableParallelMode -epm	EnableParallelMode	Optional. Enables parallel processing when loading data into a table in bulk mode. Used for Oracle. Select True or False.
--Help -?, -h	N/A	Optional. Displays or prints the help options.
--Verbose -v	N/A	Optional. Verbose output. Displays more than the minimum information about the object.
--HostName -host_name	HostName	Required. The TDM host name.
--HTTPPort -http_port	HTTPPort	Required if you do not specify an HTTPS port. Port number to connect to the Test Data Manager Service.
--HTTPSPort -https_port	HTTPSPort	Required if you do not specify an HTTP port. Port number to secure the connection to the Test Data Manager Service.
--UserName -un	UserName	Required. The Informatica domain user name.
--Password -pd	Password	Required. The password for the Informatica domain user.
--SecurityDomain -sdn	SecurityDomain	Required. The Informatica security domain.
--DomainName -dn	DomainName	Required. The name of the TDM domain.
--TDMServiceName -tsn	TDMServiceName	Required. The name of the Test Data Manager Service.

CreateTestDataWarehouse Example

The following sample command creates the test data warehouse TDWWarehouse with Oracle database type:

```
tdwcmd --CreateTestDataWarehouse --TestDataWarehouseName TDWWarehouse --PasswordAsParam
false --DBUsername orauser --DBPassword orapass --DBUrl jdbc:informatica:oracle://
localhost:1521;SID=orcl --DBConnString orcl --PassThroughSecurity=false --CodePage "7-
bit ASCII" --Kerberos false --ConnectionRetryPeriod 180 --EnableParallelMode false -
host_name localhost -http_port 6605 --UserName Administrator --Password Administrator --
SecurityDomain Native -dn  Domain_SandboxVMLin_TDM961HF14 --TDMServiceName TDMS
```

List

You can search and list test data warehouse repository, connection, test data warehouse, and data set details.

You can search for the following object types in the repository:

- Test data warehouse repository
- Test data warehouse
- Connection
- Data set
- Data set version
- Data set version tables
- Test Data Manager Service details
- User authentication parameters

The list command uses the following syntax:

```
tdwcmd
<--list | -list>
[--Repository | -Repository]
[--TestDataWarehouse | -TestDataWarehouse]
[--Connection | -Connection]
<--Dataset | -Dataset>
[--TestDataWarehouseName | -twn] TestDataWarehouseName
<--DatasetVersion | -DatasetVersion>
[--TestDataWarehouseName | -twn] TestDataWarehouseName
[--DatasetName | -dn] DatasetName
<--DatasetVersionTable | -DatasetVersionTable>
[--TestDataWarehouseName | -twn] TestDataWarehouseName
[--TableDatasetName | -tdn] TableDatasetName
[--DatasetVersionNumber | -dvn] DatasetVersionNumber
<--HostName | -host_name] HostName
[--HTTPPort | -http_port] HTTPPort
[--HTTPSPort | -https_port] HTTPSPort
<--UserName | -un> UserName
<--Password | -pd> Password
<--SecurityDomain | -sdn> SecurityDomain
<--DomainName | -dn> DomainName
<--TDMServiceName | -tsn> TDMServiceName
[--Verbose | -v]
[--Help | -?, -h]
```

The following table describes the List command options and arguments:

Option	Argument	Description
--Repository -Repository	n/a	The test data warehouse repository.
--TestDataWarehouse -TestDataWarehouse	n/a	The relational database that stores the data related to data sets that you store in the test data warehouse.
--Connection -Connection	n/a	The connection to the data warehouse repository.
--Dataset -Dataset	n/a	Required. The data set that you create in the test data warehouse.

Option	Argument	Description
--TestDataWarehouseName -twn	TestDataWarehouseName	The name of the test data warehouse.
--DatasetVersion -DatasetVersion	n/a	Required. The version of the data set.
--DatasetName -dn	DatasetName	The name of the data set.
--DatasetVersionTable -DatasetVersionTable	n/a	Required. The version of the data set table.
--TableDatasetName -tdn	TableDatasetName	The name of the data set.
--DatasetVersionNumber -dvn	DatasetVersionNumber	The version number of the data set.
--HostName -host_name	HostName	Required. The TDM host name.
--HTTPPort -http_port	HTTPPort	Required if you do not specify an HTTPS port. Port number to connect to the TDM service.
--HTTPSPort -https_port	HTTPSPort	Required if you do not specify an HTTP port. Port number to connect to the Test Data Manager Service.
--UserName -un	UserName	Required. The Informatica domain user name.
--Password -pd	Password	Required. The password for the Informatica domain user.
--SecurityDomain -sdn	SecurityDomain	Required. The Informatica security domain.
--DomainName -dn	DomainName	Required. The name of the TDM domain.
--TDMServiceName -tsn	TDMServiceName	Required. The name of the Test Data Manager Service.
--Verbose -v	n/a	Optional. Verbose output. Displays more than the minimum information about the object.
--Help -?, -h	n/a	Optional. Displays or prints the help options.

List Examples

Listing a Data Set

The following sample command lists the data set details:

```
tdwcmd --list --Dataset -twn OracleWarehouse --HostName localhost --HTTPPort 6605 -un
Administrator -pd Administrator -sdn Native -dn  Domain_SandboxVMLin_TDM961HF14 --
TDMServiceName TDMS
```

Listing a Data Set Version

The following sample command lists the data set versions:

```
tdwcmd --list --DatasetVersion -twn OracleWarehouse --DatasetName Entity --HostName
localhost --HTTPPort 6605 -un Administrator -pd Administrator -sdn Native -dn
Domain_SandboxVMLin_TDM961HF14 --TDMServiceName TDMS
```

Listing a Data Set Version Table

The following sample command lists the data set version tables:

```
tdwcmd --list --DatasetVersionTable -twn OracleWarehouse -tdn Entity -dvn 1 -host_name
localhost -http_port 6605 --UserName Administrator --Password Administrator --
SecurityDomain Native -dn  Domain_SandboxVMLin_TDM961HF14 --TDMServiceName TDMS
```

INDEX

A

Administrator
  application logs 50
application logs
  internal errors 50
application services
  TDM 12
architecture
  TDM 11
authentication
  Kerberos 33

C

cache location
  workflows 38
central seed value
  configuring 38
CLI
  configuring SSL security 35
  tdwcmd description 106
configuring
  SSL security for CLI 35
connection properties
  DB2 for Linux, UNIX and Windows 56
  Informix 57
  JDBC 58
  Microsoft SQL Server 59
  Netezza 61
  ODBC 64
  Oracle 66
  PowerExchange DB2 for i5/OS 68
  PowerExchange DB2 for Linux, UNIX and Windows 70
  PowerExchange DB2 for z/OS 72
  PowerExchange Microsoft SQL Server 76
  PowerExchange NRDB Batch 78
  PowerExchange Oracle 80
  PowerExchange Sybase 82
  SAS Analytics 84
  Sybase 85
  Teradata 87
  Teradata FastExport 88
  Teradata FastLoad 89
  Teradata MultiLoad 91
  Teradata Parallel Transporter 93
connections
  copying 54
  creating 53
  importing 54
  TDM 13
createTDWREPO
  tdwcmd command 109
CreateTestDataWarehouse
  tdwcmd command 111

D

data discovery
  disabling 45
  enabling 45
  profile 45
data domains
  sensitivity level 38
databases
  TDM 12
Default Settings
  Default Rule 42
dictionaries
    flat file dictionaries
      adding 97
      properties 97
  management 97
    relational dictionaries
      adding 96
Dictionaries view
  description 97

E

encryption
  password 35
  password for the command line interface 35

F

files
  log 47

G

general properties
  properties 37

H

HTTPS
  configuring SSL for CLI 35

I

ilmcmd
  configuring SSL security 35

K

Kerberos
  authentication 33

L

license
  management 50
log
  severity level 44
log levels
  described 48
logging in
  Test Data Manager 15
logs
  access 47
  client 47
  configuring log levels 48
  event 47
  server 47

O

overview
  privileges 17
  Test Data Manager 14

P

password
  encryption 35
permissions
  connection 31
  connection object 30
  folder 30
preferences
  overview 37
privileges
  overview 16, 17
privileges, asset linking
  business glossary 26
projects
  configuration 38
  custom fields 38
properties file name
  user-defined datatypes 103

R

roles
  overview 16

S

search field
  properties 37
security
  overview 33
  TDM Server 34
seed value
  central 38
sensitivity level
  data domains 38
session timeout
  configuring 36
settings
  log severity level 44
SourceSQLFunction
  datatype conversion parameters 104
SSL
  configuring 34
  configuring SSL for CLI 35
  creating a keystore file 34
system preferences
  overview 37

T

TargetSQLFunction
  datatype conversion parameters 104
TDM
  application services 12
  architecture 11
  connections 13
  databases 12
  overview 10
  tools 11
TDM Server
  starting and stopping 47
tdwcmd
  arguments 107
  configure 106
  list 114
  options 107
  run command 107
  syntax notation 108
  version 108
tdwcmd commands
  createTDWREPO 109
  CreateTestDataWarehouse 111
  overview 106
Test Data Manager
  authentication 33
  logging in to 15
  overview 14
  session timeout 36
test data warehouse
  add test data warehouse repository 101
  create test data warehouse 101
  create test data warehouse repository 101
  edit 102
  password 102
  shared repository 101
test data warehouse repository
  edit 102
TLS
  configuring 34
tools
  TDM 11

U

UDT
  creating properties file 104
  datatype conversion parameters 104
  properties file name 103
  user-defined datatypes 103

UDTName
datatype conversion parameters 104
user-defined datatypes
creating a properties file 104
datatype conversion parameters 104
properties file name 103
relational databases 103
users
overview 17

W

workflow generation
properties 38
workflow objects
configure naming convention 41
workflows
cache location 38
properties 38