Informatica

Informatica® Test Data Management
10.2.1

Administrator Guide

Informatica Test Data Management Administrator Guide
10.2.1
May 2018

Publication Date: 2018-06-26

Table of Contents

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, and flat files.

Informatica Resources

Informatica Network

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

As a member, you can:

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

Informatica Knowledge Base

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

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

Informatica Documentation

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

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

Informatica Product Availability Matrixes

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

Informatica Velocity

Informatica Velocity is a collection of tips and best practices developed by Informatica Professional Services. Developed from the real-world experience of hundreds of data management projects, Informatica Velocity represents the collective knowledge of our consultants who have worked with organizations from around the world to plan, develop, deploy, and maintain successful data management solutions.

If you are an Informatica Network member, you can access Informatica Velocity resources at http://velocity.informatica.com.

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

Informatica Marketplace

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

Informatica Global Customer Support

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

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

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

CHAPTER 1

Introduction to TDM Administration

This chapter includes the following topics:

TDM Administration Overview

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

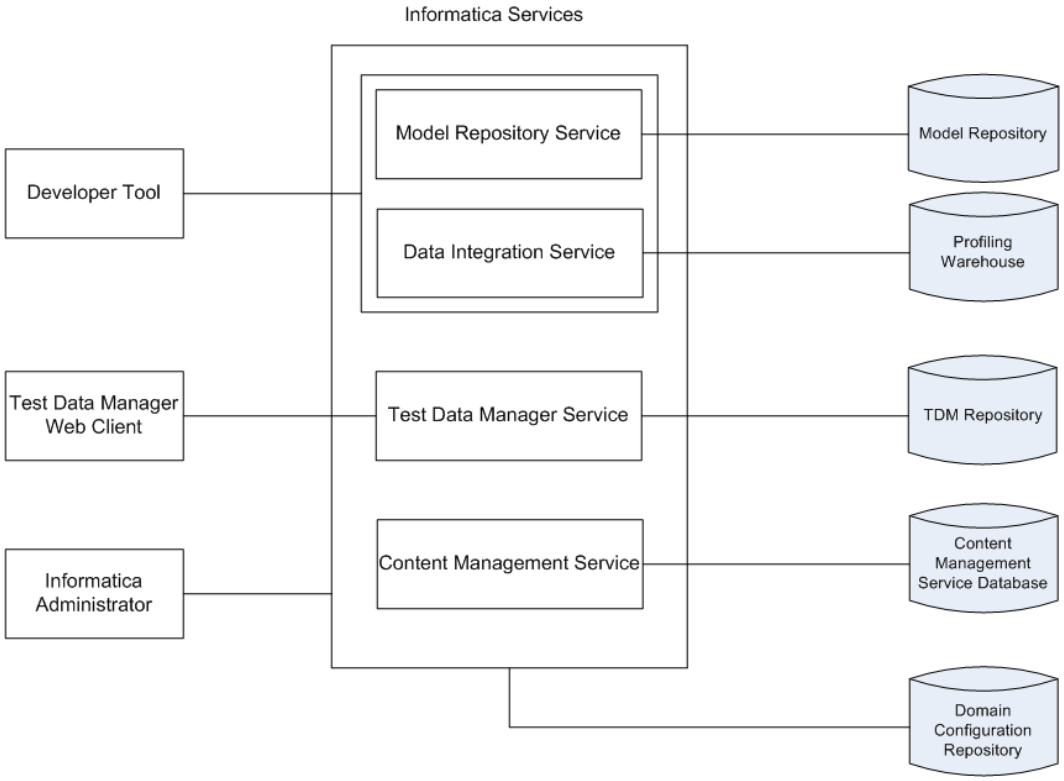
As a TDM administrator, you can perform the following tasks:

- 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 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, and data discovery operations. Create connections to the source and target databases to perform these operations.
- Perform server management tasks. You can configure licenses and restart the TDM Server from Informatica Administrator.
- Configure log level settings. You can set severity levels to generate logs for the different TDM operations.

TDM Architecture

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

The following image shows the components of TDM:



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, and data masking operations.
Developer Tool	A thick client that you use to create and run profiles to analyze the data.
Informatica Administrator	A web application that you can use to manage, monitor, deploy, and undeploy data flows.
Model Repository Service	An application service that manages the Model repository.
Data Integration Service	An application service that performs data integration tasks for the Developer tool and external clients.
Test Data Manager Service	An application service that runs Test Data Manager and manages connections between service components and Test Data Manager users.

Component	Description
Content Management Service	An application service that manages reference data. It fetches dictionary reference data from the reference data warehouse when you use relational dictionaries to mask Hadoop source connections.
TDM repository	A relational database that stores the components that you define in Test Data Manager, such as policies, projects, and rules. The TDM repository stores metadata that you import into Test Data Manager from a source database or from the Model repository.
Profiling warehouse	A relational database that stores profile results.
Model repository	A relational database that stores the table metadata for data discovery profiles. The Model repository also stores connection information for connections that you create in TDM.
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 Tools

TDM tools consist of Test Data Manager, Informatica Developer, and Informatica Administrator.

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, 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.

TDM Server

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

TDM Services

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

TDM requires the following services:

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. The Data Integration Service performs data movement and data masking operations in the Hadoop environment. To run a Hadoop plan, TDM uses the Data Integration Service to push down the transformation logic into Hadoop clusters.

Test Data Manager Service

The TDM application service that manages the TDM repository. Test Data Manager accesses the Test Data Manager Service to use database content from the TDM repository and to connect to other services to perform TDM operations.

TDM Databases

The databases component of TDM consists of the TDM repository, the Model repository, the profiling warehouse, 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.

Profiling warehouse

A relational database that stores profile results for data discovery.

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, and data masking 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, and source and target databases.

TDM uses the following connections:

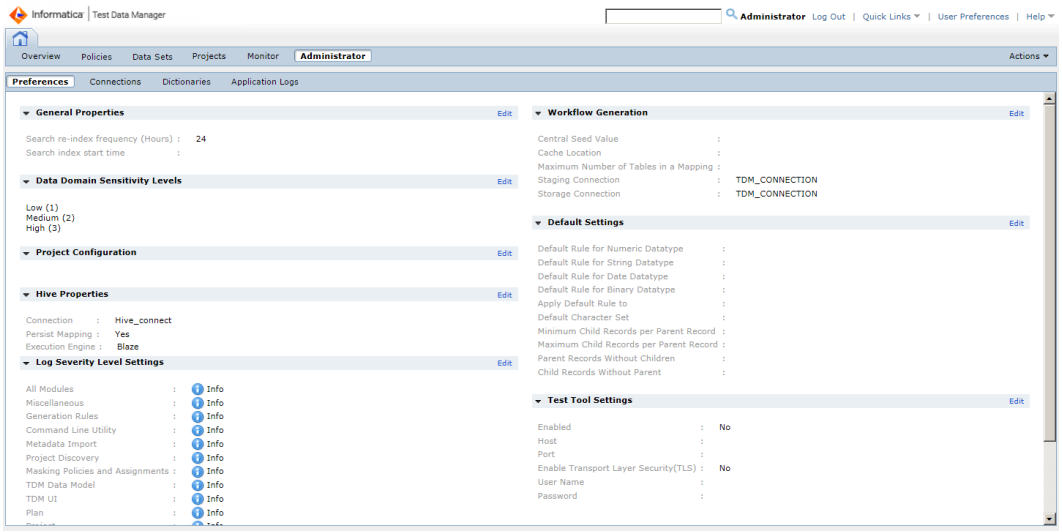
- Profiling connection. Create a connection to the Data Integration Service that you want to run data discovery operations. The profiling connection includes connection details for the Model Repository Service associated with the Data Integration Service.
- TDM repository connection. Create a connection 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.
- Database connections. Create connections to the source and target databases to perform data discovery, data subset, and data masking 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, Test Data Manager uses the connection information to create connection objects in the Model repository.

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 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.

Test Data Manager contains views. By default, an administrator can access the **Administrator** view of Test Data Manager. To access the other views in 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 Hive properties.

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.

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 Test Data Manager

To access 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 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**.

   Test Data Manager opens.

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

Changing Passwords

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

CHAPTER 2

User and Role Administration

This chapter includes the following topics:

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.

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.

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.

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 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 operations.

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

Data Integration Service Privilege

The following table lists the actions that users can perform with the privilege in the Application Administration privilege group:

Privilege Name	Description
Manage Applications	User can perform the following actions: - Back up and restore an application to a file. - Deploy an application to a Data Integration Service and resolve name conflicts. - Start an application after deployment. - Find an application. - Start or stop objects in an application. - Configure application properties.

The following table lists the required permissions and the actions that users can perform with the privilege in the Profiling Administration privilege group:

Privilege Name	Permission On	Description
Drilldown and Export Results	Read on project Execute on relational data source connection is also required to drill down on live data	User can perform the following actions: - Drill down profiling results. - Export profiling results.

Model Repository Service Privileges

The Model repository object permissions determine the tasks that users can perform on objects in projects.

The following table lists the required permissions and the actions that users can perform with the Model Repository Service privileges:

Privilege	Permission	Description
N/A	Read on project	User can view projects and objects in projects.
N/A	Write on project	User can create, edit, and delete objects in projects.
N/A	Grant on project	User can grant and revoke permissions on projects for users and groups.
Access Developer	N/A	User can access the Model repository from the Developer tool.
Create, Edit, and Delete Projects	N/A	User can perform the following actions: - Create projects. - Upgrade the Model Repository Service.
Create, Edit, and Delete Projects	Write on project	User can perform the following actions: - Edit projects. - Delete projects if the user created the projects.
Show Security Details	N/A	User can view the following details: - Names of projects for which users do not have read permission. - Error and warning message details.

Test Data Manager Service Privileges

Test Data Manager Service privileges determine the actions that users can perform using Test Data Manager. Configure privileges 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 groups in the Test Data Manager.
Policies	Includes privileges to view and manage policies in the Test Data Manager.

Privilege Group	Description
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 rules 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 data domain sensitivity levels. - 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.

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 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. - 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.
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.
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 project risk analysis. - View recent project sensitive data distribution. - Delete tables.
Generate Project	-	Write	User can generate workflows in the Test Data Manager.

Privilege	Includes Privileges	Permission	Description
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. - View plan execution.
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. - Delete tables.

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.

Rules Privilege Group

The privileges in the Rules privilege group determine the tasks that users can perform on data masking 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.

Test Data Manager Custom Roles

The Test Data Manager 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, Test Data Specialist, and Test Engineer.

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 - Manage Preferences

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
Data Domains	- View Data Domains - Manage Data Domains
Rules	- View Masking Rules - Manage Masking Rules
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
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
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
Data Domains	View Data Domains

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

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

Test Engineer

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

Privilege Group	Privilege Name
Projects	- View Project - Monitor Project

Model Repository Permissions

To generate workflows, users need folder and connection permissions. Assign permissions in the Model Repository Service.

Folder Permissions

Each project created in Test Data Manager is associated with a folder in the Model repository. Each data masking task that you perform is associated with a folder in the Model 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 Model 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 on the folder.

When you run a data masking task, the TDM server creates a folder in the Model repository. By default, the user who runs the first task on the data set is also the owner of the folder and has read, write, and execute permissions on the folder.

Work with the Informatica administrator to assign folder permissions in the Model Repository Service.

Connection Permissions

Users need connection permissions to generate and run workflows.

Permissions control the level of access that a user or group has on the connection.

You can configure permissions on a cluster configuration object, Hadoop, Hive, and relational connections in the Administrator tool.

You can assign different permission types to users to perform the following actions:

Action	Permission Types
View all connection metadata, except passwords, such as connection name, type, description, connection strings, and user names.	Read
Edit all connection metadata, including passwords. Delete the connection. Users with Write permission inherit Read permission.	Write
Access the physical data in the underlying data source defined by the connection. Users can preview data, run a mapping, run a mapping in a workflow mapping task, run a scorecard, or run a profile that uses the connection.	Execute

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. - Delete tables. - Create groups. - 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.

CHAPTER 3

Security Administration

This chapter includes the following topics:

Security Administration Overview

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

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. 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.

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 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

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, or masking operations.

System preferences include search index properties, data domain sensitivity levels, custom project fields, log severity level 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, and subset 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.

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.

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 group.

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.

Hive Properties

Set the Hive properties from the **Administrator | Preferences** view. You can modify the Hive properties at the plan level.

You can configure the following Hive properties:

Connection

The Hadoop connection that you need to run a Hadoop plan.

Persist Mapping

Optional. Stores the mappings in the Model repository for future use.

Execution Engine

The run-time engine that runs the mapping. Select Blaze, Spark, or Hive.

High Group List Directory Location

The path to the High Group List directory on the Hadoop cluster. The High Group List contains the Social Security numbers that are issued by the Social Security Administration.

Persist Mapping

You can choose to store the mappings in the Model repository.

You can enable the Persist Mapping option from the Hive Properties in the **Administrator | Preferences** view. Default is disabled. A TDM user can override this setting at the plan level. You can choose to persist mappings in the Model repository so that the mappings are available for future use. You can persist mapping if you want to troubleshoot a problem. After you persist mapping, you can view and edit the mappings.

CHAPTER 5

TDM Server Administration

This chapter includes the following topics:

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 Test Data Manager. The client logs also include log information about successful and failed login tries, timeouts, and actions that users perform.

The client logs are stored in the following location:

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

You can view information about successful logins, session timeout, and logout information from the **Application Logs** tab of the **Administrator** view in Test Data Manager. You can also view log information on unsuccessful login tries. The Owner column does not display information for unsuccessful login tries.

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/logback.xml

The changes that you make to logback.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 `logback.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 `logback.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>/configuration/logs/tdm_%i.log`

Property	Description
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>logs/tdm.log</file>
 <rollingPolicy class="ch.qos.logback.core.rolling.FixedWindowRollingPolicy">
     <FileNamePattern>logs/tdm_%i.log</FileNamePattern>
     <MinIndex>1</MinIndex>
     <MaxIndex>10</MaxIndex>
 </rollingPolicy>
 <triggeringPolicy class="ch.qos.logback.core.rolling.SizeBasedTriggeringPolicy">
     <MaxFileSize>10MB</MaxFileSize>
 </triggeringPolicy>
 <encoder class="ch.qos.logback.classic.encoder.PatternLayoutEncoder">
     <Pattern>[%d{yyyy-MM-dd HH:mm:ss.SSS}]\t[%logger]\t[%X{username}]\t[%X{context}]
\t[%level]\t[%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

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

You can create connections in TDM. You can also import connections that you created in 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.

TDM supports some databases for certain TDM operations only. When you create a connection, relevant properties appear in the connection configuration wizard.

TDM supports SSL authentication for source and target connections.

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 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.

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 includes the connection 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 Model repository.
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.

Property	Description
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 Data 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 Data 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.

Hadoop Connections

A Hadoop connection is a cluster type connection. In the Administrator tool, you must create a cluster configuration for the Hadoop clusters. Create and manage Hadoop connections from Test Data Manager.

The following table describes Hadoop connection properties:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
ID	String that the Data Integration Service uses to identify the connection. The ID is not case sensitive. It must be 255 characters or less and must be unique in the domain. You cannot change this property after you create the connection. Default value is the connection name.
Connection Type	Required. The connection type. Select Hadoop.

Property	Description
Description	The description of the connection. The description cannot exceed 4000 characters.
Use Kerberos Authentication	Enables Kerberos authentication for Hadoop connections.

Hadoop Properties

The following table describes the cluster and connection properties that you configure for Hadoop:

Property	Description
Cluster Configuration	The name of the cluster configuration object associated with the Hadoop environment.
Cluster Environment Variable	Environment variables used in the cluster. Specify any custom environment variables in the Hadoop connection. During runtime, the specified environment variables are combined with the default environment variables based on the cluster configuration associated with the Hadoop connection. For example, you can specify ORACLE_HOME, ODBCHOME, or DB2_HOME.
Cluster Library Path	Library paths for the cluster environment variables.
Impersonation User Name	Required if the Hadoop cluster uses Kerberos authentication. Hadoop impersonation user. The user name that the Data Integration Service impersonates to run mappings in the Hadoop environment. The Data Integration Service runs mappings based on the user that is configured. Refer the following order to determine which user the Data Integration Services uses to run mappings: 1. Operating system profile user. The mapping runs with the operating system profile user if the profile user is configured. If there is no operating system profile user, the mapping runs with the Hadoop impersonation user. 2. Hadoop impersonation user. The mapping runs with the Hadoop impersonation user if the operating system profile user is not configured. If the Hadoop impersonation user is not configured, the Data Integration Service runs mappings with the Data Integration Service user. 3. Informatica services user. The mapping runs with the operating user that starts the Informatica daemon if the operating system profile user and the Hadoop impersonation user are not configured.
Temporary Table Compression Codec	Hadoop compression library for a compression codec class name.
Codec Class Name	Codec class name that enables data compression and improves performance on temporary staging tables.
Hive Staging Database Name	Namespace for Hive staging tables. Use the name default for tables that do not have a specified database name.

Property	Description
Hadoop Engine Custom Properties	Custom properties that are unique to the Hadoop connection. You can specify multiple properties. Use the following format: `<property1>=<value>` To specify multiple properties use `&:` as the property separator. If more than one Hadoop connection is associated with the same cluster configuration, you can override configuration set property values. Use Informatica custom properties only at the request of Informatica Global Customer Support.
Write Reject Files to Hadoop	If you use the Blaze engine to run mappings, select the check box to specify a location to move reject files. If checked, the Data Integration Service moves the reject files to the HDFS location listed in the property, Reject File Directory. By default, the Data Integration Service stores the reject files based on the RejectDir system parameter.
Reject Files Directory	The directory for Hadoop mapping files on HDFS when you run mappings.

Hive Configuration

You can use the values for Hive configuration properties from hive-site.xml or mapred-site.xml located in the following directory on the Hadoop cluster: `/etc/hadoop/conf/`.
The following table describes the connection properties that you configure to push mapping logic to the Hadoop cluster:

Property	Description
Environment SQL	SQL commands to set the Hadoop environment. The Data Integration Service executes the environment SQL at the beginning of each Hive script generated in a Hive execution plan. The following rules and guidelines apply to the usage of environment SQL: - Use the environment SQL to specify Hive queries. - Use the environment SQL to set the classpath for Hive user-defined functions and then use environment SQL or PreSQL to specify the Hive user-defined functions. You cannot use PreSQL in the data object properties to specify the classpath. The path must be the fully qualified path to the JAR files used for user-defined functions. Set the parameter `hive.aux.jars.path` with all the entries in `infapdo.aux.jars.path` and the path to the JAR files for user-defined functions. - You can use environment SQL to define Hadoop or Hive parameters that you want to use in the PreSQL commands or in custom queries. - If you use multiple values for the environment SQL, ensure that there is no space between the values.
Hive Warehouse Directory on HDFS	Required. The absolute HDFS file path of the default database for the warehouse that is local to the cluster. If you do not configure the Hive warehouse directory, the Hive engine first tries to write to the directory specified in the cluster configuration property `hive.metastore.warehouse.dir`. If the cluster configuration does not have the property, the Hive engine writes to the default directory `/user/hive/warehouse`.

Property	Description
Hive JDBC Connection String	The JDBC URI to connect to the Hive server. To connect to HiveServer, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database name to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details.
Engine Type	The engine that the Hadoop environment uses to run a mapping on the Hadoop cluster. You can choose MRv2 or Tez. You can select Tez if it is configured for the Hadoop cluster. Default is MRv2.
Hive Engine Custom Properties	Custom properties that are unique to the Hive connection. You can specify multiple properties. Use the following format: `<property1>=<value>` To specify multiple properties use `&:` as the property separator. If more than one Hive connection is associated with the same cluster configuration, you can override configuration set property values. Use Informatica custom properties only at the request of Informatica Global Customer Support.

Blaze Engine

The following table describes the connection properties that you configure for the Blaze engine:

Property	Description
Blaze Staging Directory	The HDFS file path of the directory that the Blaze engine uses to store temporary files. Verify that the directory exists. The YARN user, Blaze engine user, and mapping impersonation user must have write permission on this directory. Default is `/blaze/workdir`. If you clear this property, the staging files are written to the Hadoop staging directory `/tmp/blaze_<user name>`.
Blaze Service User Name	The operating system profile user name for the Blaze engine.
Minimum Port	The minimum value for the port number range for the Blaze engine. Default is 12300.
Maximum Port	The maximum value for the port number range for the Blaze engine. Default is 12600.
YARN Queue Name	The YARN scheduler queue name used by the Blaze engine that specifies available resources on a cluster.

Property	Description
Blaze Job Monitor Address	The host name and port number for the Blaze Job Monitor. Use the following format: `<hostname>:<port>` Where - <hostname> is the host name or IP address of the Blaze Job Monitor server. - <port> is the port on which the Blaze Job Monitor listens for remote procedure calls (RPC). For example, enter: `myhostname:9080`
Blaze Service Custom Properties	Custom properties that are unique to the Blaze engine. To enter multiple properties, separate each name-value pair with the following text: `&:`.Use Informatica custom properties only at the request of Informatica Global Customer Support.

Spark Engine

The following table describes the connection properties that you configure for the Spark engine:

Property	Description
Spark Staging Directory	The HDFS file path of the directory that the Spark engine uses to store temporary files for running jobs. The YARN user, Data Integration Service user, and mapping impersonation user must have write permission on this directory. By default, the temporary files are written to the Hadoop staging directory `/tmp/spark_<user name>`.
Spark Event Log Directory	Optional. The HDFS file path of the directory that the Spark engine uses to log events.
YARN Queue Name	The YARN scheduler queue name used by the Spark engine that specifies available resources on a cluster. The name is case sensitive.
Spark Execution Parameters	An optional list of configuration parameters to apply to the Spark engine. You can change the default Spark configuration properties values, such as `spark.executor.memory` or `spark.driver.cores`. Use the following format: `<property1>=<value>` To enter multiple properties, separate each name-value pair with the following text: `&:`

HDFS Connections

You can add and use Hadoop Distributed File System (HDFS) connections.

Use an HDFS connection to access data in the Hadoop cluster. The HDFS connection is a file system type connection.
In the Administrator tool, you must create a cluster configuration for the Hadoop clusters. Create and manage HDFS connections in Test Data Manager.

The following table describes HDFS connection properties:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select HDFS.
Description	The description of the connection. The description cannot exceed 765 characters.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
User Name	Required. User name to access HDFS.
NameNode URI	Required. The URI to access HDFS. Use the following format to specify the NameNode URI in Cloudera and Hortonworks distributions: hdfs://<namenode>:<port> Where - <namenode> is the host name or IP address of the NameNode. - <port> is the port that the NameNode listens for remote procedure calls (RPC). Use one of the following formats to specify the NameNode URI in MapR distribution: – - maprfs:/// - maprfs:///mapr/my.cluster.com/ Where my.cluster.com is the cluster name that you specify in the mapr-clusters.conf file.
Directory	Required. The path to the HDFS directory.

Hive Connections

Use the Hive connection to access a Hive database. A Hive connection is a database type connection. In the Administrator tool, you must create a cluster configuration for the Hadoop clusters. Create and manage Hive connections from Test Data Manager.

The following table describes Hive connection properties:

Property	Description
Name	Required. Name of the connection. The name is not case sensitive and must be unique within the domain. The name cannot exceed 128 characters, contain spaces, or contain the following special characters: ~ ` ! $ % ^ & * ( ) - + = { [}] | \ : ; " ' < , > . ? /
Connection Type	Required. The connection type. Select Hive.
Description	The description of the connection. The description cannot exceed 4000 characters.

Property	Description
Use Kerberos Authentication	Enables Kerberos authentication for Hadoop connections.
Owner	The owner of the connection. Default is the user who creates the connection. You can change the owner of the connection.
Connection Modes	Hive connection mode. Select at least one of the following options: - Access Hive as a source or target. Select this option if you want to use the connection to access the Hive data warehouse. If you want to use Hive as a target, you must enable the same connection or another Hive connection to run mappings in the Hadoop cluster. - Use Hive to run mappings in Hadoop cluster. Select this option if you want to use the connection to run mappings in the Hadoop cluster. You can select both the options. Default is **Access Hive as a source or target**.
User Name	The name of the user that the Data Integration Service impersonates to run mappings on a Hadoop cluster. The user name depends on the JDBC connection string that you specify in the Metadata Connection String or Data Access Connection String for the native environment. If the Hadoop cluster uses Kerberos authentication, the principal name for the JDBC connection string and the user name must be the same. Otherwise, the user name depends on the behavior of the JDBC driver. With Hive JDBC driver, you can specify a user name in many ways and the user name can become a part of the JDBC URL. If the Hadoop cluster does not use Kerberos authentication, the user name depends on the behavior of the JDBC driver. If you do not specify a user name, the Hadoop cluster authenticates jobs based on the following criteria: - The Hadoop cluster does not use Kerberos authentication. It authenticates jobs based on the operating system profile user name of the machine that runs the Data Integration Service. - The Hadoop cluster uses Kerberos authentication. It authenticates jobs based on the SPN of the Data Integration Service.
Environment SQL	SQL commands to set the Hadoop environment. In native environment type, the Data Integration Service executes the environment SQL each time it creates a connection to a Hive metastore. If you use the Hive connection to run mappings in the Hadoop cluster, the Data Integration Service executes the environment SQL at the beginning of each Hive session. The following rules and guidelines apply to the usage of environment SQL in both connection modes: - Use the environment SQL to specify Hive queries. - Use the environment SQL to set the classpath for Hive user-defined functions and then use environment SQL or PreSQL to specify the Hive user-defined functions. You cannot use PreSQL in the data object properties to specify the classpath. The path must be the fully qualified path to the JAR files used for user-defined functions. Set the parameter hive.aux.jars.path with all the entries in infapdo.aux.jars.path and the path to the JAR files for user-defined functions. - You can use environment SQL to define Hadoop or Hive parameters that you want to use in the PreSQL commands or in custom queries. If you use the Hive connection to run mappings in the Hadoop cluster, the Data Integration service executes only the environment SQL of the Hive connection. If the Hive sources and targets are on different clusters, the Data Integration Service does not execute the different environment SQL commands for the connections of the Hive source or target.

Properties to Access Hive as Source or Target

The following table describes the connection properties that you configure to access Hive as a source or target:

Property	Description
Metadata Connection String	The JDBC connection URI used to access the metadata from the Hadoop server. You can use PowerExchange for Hive to communicate with a HiveServer service or HiveServer2 service. To connect to HiveServer, specify the connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database name to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. To connect to HiveServer 2, use the connection string format that Apache Hive implements for that specific Hadoop Distribution. For more information about Apache Hive connection string formats, see the Apache Hive documentation. For user impersonation, you must add `hive.server2.proxy.user=<xyz>` to the JDBC connection URI. If you do not configure user impersonation, the current user's credentials are used connect to the HiveServer2. If the Hadoop cluster uses SSL or TLS authentication, you must add `ssl=true` to the JDBC connection URI. For example: `jdbc:hive2://<hostname>:<port>/<db>;ssl=true` If you use self-signed certificate for SSL or TLS authentication, ensure that the certificate file is available on the client machine and the Data Integration Service machine. For more information, see the *Informatica Big Data Management Hadoop Integration Guide*.
Bypass Hive JDBC Server	JDBC driver mode. Select the check box to use the embedded JDBC driver mode. To use the JDBC embedded mode, perform the following tasks: - Verify that Hive client and Informatica services are installed on the same machine. - Configure the Hive connection properties to run mappings on a Hadoop cluster. If you choose the non-embedded mode, you must configure the Data Access Connection String. Informatica recommends that you use the JDBC embedded mode.
Data Access Connection String	The connection string to access data from the Hadoop data store. To connect to HiveServer, specify the non-embedded JDBC mode connection string in the following format: `jdbc:hive2://<hostname>:<port>/<db>` Where - <hostname> is name or IP address of the machine on which HiveServer2 runs. - <port> is the port number on which HiveServer2 listens. - <db> is the database to which you want to connect. If you do not provide the database name, the Data Integration Service uses the default database details. To connect to HiveServer 2, use the connection string format that Apache Hive implements for the specific Hadoop Distribution. For more information about Apache Hive connection string formats, see the Apache Hive documentation. For user impersonation, you must add `hive.server2.proxy.user=<xyz>` to the JDBC connection URI. If you do not configure user impersonation, the current user's credentials are used connect to the HiveServer2. If the Hadoop cluster uses SSL or TLS authentication, you must add `ssl=true` to the JDBC connection URI. For example: `jdbc:hive2://<hostname>:<port>/<db>;ssl=true` If you use self-signed certificate for SSL or TLS authentication, ensure that the certificate file is available on the client machine and the Data Integration Service machine. For more information, see the *Informatica Big Data Management Hadoop Integration Guide*.

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.
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.
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.

Property	Description
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.
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.

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.

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.
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 Data 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 Data 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.
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.
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.
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 Data Integration Service executes 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 Data 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.

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.

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: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 Data 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 Data 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.

CHAPTER 7

Dictionaries

This chapter includes the following topics:

Dictionaries Overview

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

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

The Data 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.

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.

Dictionary Management

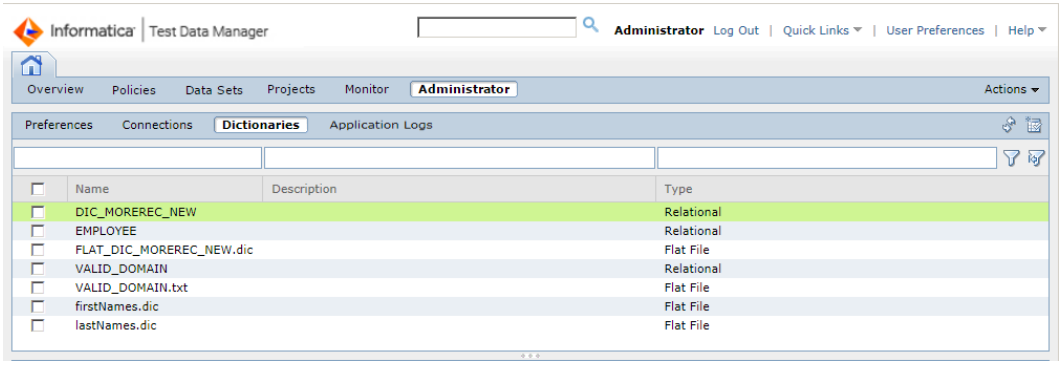
You can add, edit, delete, sort, and filter dictionaries in the **Administrator | Dictionaries** view.

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:



INDEX

P

R

S

T

U