Informatica®

Informatica® Test Data Management
10.2.1

User Guide

Informatica Test Data Management User Guide
10.2.1
May 2018

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

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

NOTICES

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

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

The information in this documentation is subject to change without notice. If you find any problems in this documentation, report them to us at infa_documentation@informatica.com.

Informatica products are warranted according to the terms and conditions of the agreements under which they are provided. INFORMATICA PROVIDES THE INFORMATION IN THIS DOCUMENT "AS IS" WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING WITHOUT ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND ANY WARRANTY OR CONDITION OF NON-INFRINGEMENT.

Publication Date: 2018-06-25

Table of Contents

Preface 12

Informatica Resources. 12

Informatica Network. 12

Informatica Knowledge Base. 12

Informatica Documentation. 12

Informatica Product Availability Matrixes. 13

Informatica Velocity. 13

Informatica Marketplace. 13

Informatica Global Customer Support. 13

Chapter 1: Introduction to Test Data Management. 14

Test Data Management Overview. 14

Test Data Management Use Cases. 15

Using TDM for Security and Compliance in Application Testing Environments. 15

TDM Architecture. 16

TDM Tools. 17

TDM Server. 17

Application Services. 17

TDM Databases. 18

TDM Connections. 18

TDM Process. 19

Create a Data Masking Policy. 20

Create a Project and Import Metadata. 20

Discover Source Information. 20

Define Data Masking and Data Subset Operations. 20

Create a Plan for Data Masking and Data Subset. 21

Monitor the Workflow. 21

Chapter 2: Test Data Manager. 22

Test Data Manager Overview. 22

Test Data Manager User Interface. 23

Views. 23

Search Field. 24

Advanced Search Text Filter. 24

Quick Links Menu. 24

User Preferences Menu. 25

Actions Menu. 25

Data Filter. 25

Overview View. 25

Dashboards. 25

Project Risk Analysis. 26

Project Assignments and Project Objects. 26

Recent Project Sensitive Data Distribution. 26

Recent Activities. 26

Plan Execution. 26

Global Components. 26

Recent Projects. 27

Policies View. 27

Projects View. 27

Monitor View. 27

Administrator View. 28

Expression Builder. 28

Logging In to Test Data Manager. 29

Chapter 3: Projects. 31

Projects Overview. 31

Project Components. 32

Project Logs. 33

Project Management. 33

Creating a Project. 33

Editing a Project. 34

Copying a Project. 35

Deleting a Project. 35

Configuring Project Log Locations. 35

Exporting a Project. 35

Importing a Project. 36

Data Sources. 36

TDM Source Definitions. 36

Hive and HDFS Data Sources. 37

Importing Data Sources. 39

Delete a Table. 40

Deleting a Table. 40

Project Permission and Security. 40

Project Permissions. 41

Updating User and Group Security. 41

Chapter 4: Policies. 43

Policies Overview. 43

Policies View. 43

Policies Task Flow. 44

Rules. 44

Data Domains. 44

Apply Masking Rules to a Data Domain. 45

Metadata and Data Patterns for Data Domains. 45
Regular Expression Syntax. 45
Data Patterns 46
Metadata Patterns. 46
Data Domain Options. 46
Creating a Data Domain. 47
Copying a Data Domain. 47
Editing a Data Domain. 48
Deleting a Data Domain. 48
Policy Packs. 48
PII Policy Pack. 49
PHI Policy Pack. 50
PCI Policy Pack. 51
Import and Export. 51
Exporting Policy Components. 52
Importing Policy Components. 52
Data Domain Import. 52
Policy Management. 53
Creating a Policy. 53
Copying a Policy. 54
Editing a Policy. 54
Deleting a Policy. 54

Chapter 5: Data Discovery. 56

Data Discovery Overview. 56
Data Discovery Sources. 57
Rules and Guidelines for Data Discovery Sources. 57
Discover View. 57
Column Properties. 58
Sensitive Columns. 59
Computed Columns. 59
Value Cascades. 59
Auto Cascades. 61
Data Discovery Task Flow. 61
Data Domain Discovery. 61
Data Domain Profiling on Hive and HDFS Sources. 62
Data Domain Profile Sampling Options. 62
Assigning a Data Domain to Multiple Columns. 62
Manually Updating the Column Data Domain. 63
Data Domain Assignments Import and Export. 63
Profile Management. 64
Creating a Data Domain Profile. 64
Editing a Profile. 65

Deleting a Profile. 65
Profile Import. 65
Importing a Profile. 66
Apply the Results. 66
Data Domain Discovery Results. 66
Project Tables. 68
Table Classification. 68
Constraints. 68
Manually Add Keys to Tables in a Project. 69
Creating a Primary Key Constraint. 69
Creating a Logical Relationship Between Tables. 69
Creating a Unique Key Constraint. 70
Chapter 6: Creating a Data Subset. 71
Data Subset Overview. 71
Data Subset Process Flow. 71
Data Subset Components. 72
Groups. 72
Creating a Group. 73
Editing a Data Subset Component. 73
Exporting a Data Subset Component. 73
Importing a Data Subset Component. 74
Copying a Data Subset Component. 74
Deleting a Data Subset Component. 74
Creating a Data Subset. 75
Chapter 7: Performing a Data Masking Operation. 76
Data Masking Overview. 76
Data Masking Task Flow. 77
Data Masking Rules. 77
Standard Masking Rules 77
Mapplet Rules. 78
Advanced Masking Rules. 78
Masking Rule Assignments. 79
Creating and Assigning Data Masking Rules. 79
Creating a Standard Masking Rule. 80
Creating a Mapplet Masking Rule. 81
Creating an Advanced Masking Rule. 81
Adding Data Masking Rules to a Project. 82
Assigning a Standard Masking Rule. 82
Assigning a Custom Masking Rule. 83
Assigning an Advanced Masking Rule. 83
Modifying Data Masking Rules and Assignments. 84

Available Data Masking Rules . . . 84
Editing a Masking Rule. . . . 85
Copying a Masking Rule. . . . 85
Deleting a Masking Rule. . . . 85
Overriding a Masking Rule. . . . 86
Assigning Single and Multiple Masking Rules. . . . 86
Deleting Masking Rule Assignments. . . . 87
Performing a Data Masking Operation. . . . 88
Data Masking Components. . . . 88

Chapter 8: Data Masking Techniques and Parameters. . . . 90
Data Masking Techniques and Parameters Overview. . . . 91
Data Masking Techniques. . . . 91
Data Masking Parameters. . . . 92
Repeatable Output. . . . 93
Exception Handling. . . . 94
Custom Masking. . . . 94
Custom Masking Parameters. . . . 95
Advanced Masking. . . . 95
Advanced Masking Parameters. . . . 96
Advanced Masking Example. . . . 97
Credit Card Masking. . . . 97
Credit Card Masking Parameters. . . . 98
Email Masking. . . . 98
Email Masking Parameters. . . . 98
Encryption Masking. . . . 100
Expression Masking. . . . 100
Expression Masking Parameters. . . . 101
Rules and Guidelines for Expression Masking. . . . 101
IP Address Masking. . . . 101
Key Masking. . . . 102
Mask Format. . . . 102
Source String Characters. . . . 103
Result String Replacement Characters. . . . 103
Date Key Masking . . . 103
Numeric Key Masking Parameters. . . . 104
String Key Masking Parameters. . . . 104
Nullification Masking. . . . 104
Phone Masking. . . . 105
Random Masking. . . . 105
Range Masking. . . . 105
Blurring. . . . 106
Mask Format. . . . 106

Source String Characters. 107

Result String Replacement Characters. 107

Date Random Masking Parameters. 108

Numeric Random Masking Parameters. 108

String Random Masking Parameters. 109

Shuffle Masking. 109

Shuffle Masking Parameters. 110

SIN Masking. 110

SSN Masking. 111

Substitution Masking. 111

Substitution Masking Parameters. 112

URL Masking. 112

Name Substitution Example 113

Add a Dictionary in Test Data Manager. 113

Creating the Substitution Rule. 114

Creating the Advanced Masking Rule. 114

Shuffle Address Example. 118

Creating the Shuffle Rule. 119

Create the Advanced Masking Rule. 119

Chapter 9: Plans and Workflows. 125

Plans and Workflows Overview. 125

Plans and Workflows Task List. 126

Workflow Connections. 126

Plan Components. 126

Persist Mapping. 127

Plan Settings. 127

Connection Options. 128

Target Options. 128

Advanced Options. 128

Hadoop Plan Settings. 129

Masking Components. 130

Subset Components. 130

Hadoop Components. 131

Component Criteria. 131

Disabling Masking for a Column. 131

Source Settings. 131

Connection Properties. 132

Target Properties. 133

Advanced Properties. 134

Hadoop Data Source Settings. 134

Plan Management. 135

Creating a Data Masking and Data Subset Plan. 135

Creating a Hadoop Plan. 136
Copying a Plan. 136
Exporting a Plan. 137
Importing a Plan. 137
Deleting a Plan. 137
Workflow Generation. 137
Generating a Workflow. 138
Workflow View. 138
Executing a Workflow. 139
Workflow Executions View. 139
Workflow Tasks. 140
Workflow Properties Panel. 140

Chapter 10: Monitor. 142
Monitor Overview. 142
Jobs. 142
Job Details. 143
Monitor Tasks. 143
Logs. 144
Severity Levels. 144
Viewing the Log Messages. 145
Monitoring for Hadoop. 145

Chapter 11: Reports. 146
Reports Overview. 146
Audit Trail Report. 147
Running an Audit Trail Report. 147
Data Masking Report. 147
Running the Data Masking Report. 147
Plan Audit Report. 147
Running a Plan Audit Report. 148
Plan Detail Report. 149
Running the Plan Detail Report. 149

Chapter 12: ilmcmd 150
ilmcmd Overview. 150
Configuring ilmcmd. 151
Running ilmcmd. 151
Entering Options and Arguments. 151
Syntax Notation. 152
Delete. 152
Delete Examples. 153
Export. 153

Export Examples. 154
Import. 155
Import Examples. 155
Search. 156
Search Examples. 157
Workflow. 157
Workflow Examples. 158

Appendix A: Data Type Reference. 159
Data Type Reference Overview. 159
Oracle. 159
Microsoft SQL Server. 162
DB2 for Linux, UNIX, and Windows. 163
Sybase ASE. 165
HDFS. 167
Hive. 167
Flat File. 168

Appendix B: Data Type Reference for Hadoop. 169
Data Type Reference for Hadoop Overview. 169
Oracle. 170
Microsoft SQL Server. 172
DB2 for Linux, UNIX, and Windows. 173
Sybase ASE. 174
Hive. 176
HDFS. 177
Flat File. 177

Index. 178

Preface

The Informatica *Test Data Management User Guide* describes how to protect sensitive data and create lean non-production systems for test and development. It shows you how to implement data subset, data masking, and data discovery operations. This guide is written for users who use Test Data Manager. 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 Test Data Management

This chapter includes the following topics:

- Test Data Management Overview, 14
- Test Data Management Use Cases, 15
- TDM Architecture, 16
- TDM Process, 19

Test Data Management Overview

Test Data Management (TDM) integrates with Informatica applications to manage nonproduction data in an organization.

With TDM, an organization can create a smaller copy of the production data and mask the sensitive data. An organization can discover the sensitive columns in the test data, and ensure that the sensitive columns are masked in the test data.

Organizations create multiple copies of application data to use for testing and development. Organizations often maintain strict controls on production systems, but data security in nonproduction systems is not as secure. An organization must maintain knowledge of the sensitive columns in the production data and ensure that sensitive data does not appear in the test environment. Development must not have to rewrite code to create test data.

Manage data discovery, data subset, and data masking in Test Data Manager.

Data discovery

Use data discovery to run sensitive field profiles to identify the columns that contain sensitive data. Use the profile results to determine which columns to mask and which data masking techniques to apply. Define data domains to identify sensitive data columns by patterns in the data or the column metadata. When you apply data masking, you can apply the same rule to multiple columns in the same data domain.

Data subset

Use data subset to create a small environment for testing and development. You can define the type of data that you want to include in the subset database. You might create a subset database with data based on time, function, or geographic location. For example, a time-based subset database might include recent payment transactions from all invoice data in a production system.

Data masking

Create data masking rules to apply to source columns and data domains. You can apply different masking techniques such as substitution masking, shuffle masking, key masking, and encryption. You can configure repeatable results in the masked data. You can assign multiple rules to the same column.

To perform data subset and masking operations, you can generate and run workflows from data subset and data masking plans in Test Data Manager.

You can perform data masking and data movement on Big Data Edition Hadoop clusters. Use Hadoop sources to lower the cost of raw data storage and to solve large scale analytics by using the distributed computing capabilities of Hadoop. For example, when you move sensitive data into Hadoop, you can classify data for analytics, provision data for testing, or other purposes.

Use Hadoop to improve the speed of processing large volumes of structured and unstructured data. For example, you work with heterogeneous data sets and you want to normalize and correlate data sets of the size of terabytes or petabytes. The analytics results processed on Hadoop are faster and cost-effective, and you can extract the analytics results to a conventional database.

TDM includes the ilmcmd command line program. Run ilmcmd commands to perform a subset of the Test Data Manager tasks from the command line.

TDM users have roles and privileges that determine the tasks that they can perform through Test Data Manager or the ilmcmd command line program. The administrator manages roles and privileges for users from the Informatica Administrator.

Test Data Management Use Cases

Uses cases for Test Data Management include security and compliance in application testing environments.

Using TDM for Security and Compliance in Application Testing Environments

Testing teams need to be compliant and maintain strict controls on the data used in testing environments.

You must identify sensitive information in huge volumes of production data and then ensure that you do not expose the information in the test data. This is a challenge and a compliance and security risk.

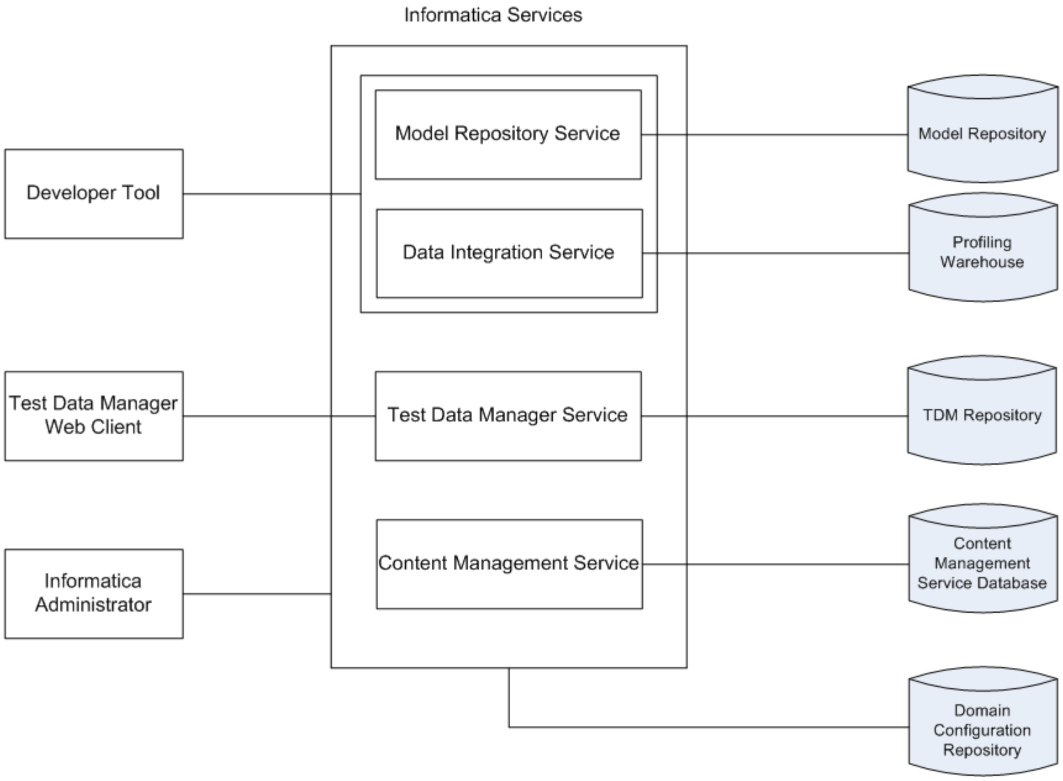
You can perform the following tasks to help with security and compliance in a test environment:

- To analyze the data and identify sensitive information in the data, run profiles in TDM. You can compare the data against multiple compliance regulation standards to ensure that the data meets standard compliance regulations. For example, run a profile on the data to identify columns that contain sensitive data according to PII, PCI, or PHI compliance regulations.
- To remove sensitive data that you identify, run a data masking operation in TDM. You can use standard masking formats such as credit card numbers or Social Security numbers to mask data or create customized formats to mask data.

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:


Informatica Services
Developer Tool
Test Data Manager Web Client
Informatica Administrator
Model Repository Service
Data Integration Service
Test Data Manager Service
Content Management Service
Model Repository
Profiling Warehouse
TDM Repository
Content Management Service Database
Domain Configuration Repository

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

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

Test Data Manager

A web-based client application that you can use to configure data masking, data subset, and profiles for data discovery. You can also configure connections, and manage project permissions for users and user groups.

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 is the interface between Test Data Manager and the application services.

Application Services

TDM uses Informatica services. Create the services in the Administrator tool.

TDM uses the following application services:

Model Repository Service

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

Data Integration Service

An application service that performs the 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 create a Data Integration Service in the Administrator tool, you select the data profiling warehouse to store data from data discovery profiles. The Data Integration Service performs data movement and data masking operations in the Hadoop environment.

Test Data Manager Service

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

TDM Databases

TDM connects to databases for metadata, profiling, TDM configuration, and domain configuration.

TDM needs connections to 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. When you perform data masking and data movement operations on Hadoop, you can choose to store the mappings in the Model repository for future use.

Profiling warehouse

A relational database that stores profile results for data discovery.

Domain configuration repository

A relational database that stores connections and metadata for the Informatica domain.

TDM Connections

TDM connects to databases, repositories, and services to perform data subset, masking, and profiles for discovery operations. The connection requirements are based on the operations that you need to perform.

To perform data discovery operations, TDM requires connections to a database source and a Data Integration Service.

To perform data masking and data movement operations on Hadoop, TDM requires connections to Hadoop sources and a Data Integration Service.

Workflows for data subset and masking operations require connections to services, the TDM repository, and external database sources.

Service Connections

TDM requires connections to the following services:

Data Integration Service

TDM requires a connection to a Data Integration Service for data discovery operations. The Data Integration Service is the service in the Informatica domain that performs the data discovery operation. The Data Integration Service performs data movement and data masking operations in the Hadoop environment.

Model Repository Service

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

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.

Repository Connections

TDM requires connections to repositories.

TDM accesses the following repositories:

TDM repository

The Data Integration Service stores TDM components in the TDM repository.

A data masking or data subset workflow requires a connection to the TDM repository.

Model repository

When you run profiles to discover data, the TDM Server sends a request to the Data Integration Service to extract data for the source tables. The Data Integration Service sends a request to its associated Model Repository Service to load the metadata for the tables to the Model repository. When you perform data masking and data movement operations on Hadoop, you can choose to store the mappings in the Model repository for future use.

Profiling warehouse

The Data Integration Service loads the profile results to the profiling warehouse. When you create a Data Integration Service in the Administrator tool, you configure a profiling warehouse.

Database Connections

TDM requires a connection to a database source to perform data discovery operations. It also requires connections for relational sources and targets for data subset and data masking.

Configure connections in the Administrator view of Test Data Manager. To connect TDM to databases, you do not need to install a separate driver.

When you generate workflows from data subset or masking plans, you must select connections for relational sources and targets. Select these connections in Test Data Manager when you create a plan. TDM tests the connections when it generates workflows from plans, and the Data Integration Service uses the connections when it runs the workflows.

TDM Process

Run a profile against source data, create a subset of the data, and mask the subset data.

The TDM process includes the following high-level steps:

1. Create policies that define the types of data you want to mask and the rules that you might use to mask the data.
2. Create a project and import data sources.
3. Optionally, discover information about the source data. Run profiles for data and metadata discovery to discover data domains.
4. Define data subset operations and data masking operations. Define the tables that you want to include in the subset database and the relationships between the tables. Assign data masking rules to columns in the source data.
5. Generate and run the workflow for data masking or data subset.
6. Monitor the workflow.

Create a Data Masking Policy

Design policies to mask specific types of data. A policy includes the data domains that describe the data that you want to mask. A policy does not contain any data source. You can apply a policy to more than one project in Test Data Manager.

Define data domains to group sensitive fields by column name or by the column data. Define patterns in the column name or the column data using regular expressions. A data domain also contains masking rules that describe how to mask the data.

To design a data masking rule, select a built-in data masking technique in Test Data Manager. A rule is a data masking technique with specific parameters. You can create data masking rules with mapplets imported into TDM from the Developer tool.

Create a Project and Import Metadata

Create a project to organize the components for data discovery, masking, and subset operations.

Import data sources in the project. Create a target schema. TDM overwrites any data that already exists in the target schema. Import metadata for the sources on which you want to perform data subset or masking operations. You can import metadata from the Model repository or an external database source.

When you import source metadata from the Model repository, the TDM Server sends a request to the Model Repository Service to extract source metadata from the Model repository. The Model Repository Service loads the source metadata to the TDM repository. When you import external database metadata, the TDM Server extracts metadata from the source tables and loads it into the TDM repository.

Discover Source Information

You can run profiles to discover data domains in source tables.

You can run a data domain profile to search for columns in the source data to add to each data domain. Use data domain profile results to determine which columns to mask with the same masking rules.

When you run profiles for data discovery, the TDM Server sends a request to the Data Integration Service to extract data from the source tables. The Data Integration Service loads the profile results to the profiling warehouse. When you add constraints to tables, the TDM Server stores the constraints in the TDM repository. The TDM server does not update the data sources.

Define Data Masking and Data Subset Operations

To define data subset operations, define the tables that you want to include in the subset database and the relationships between the tables. To perform data masking operations, create a plan to run the masking operations. Add policies for masking the data. You can also add rules that are not in policies.

Perform the following tasks in Test Data Manager to define the data masking and data subset operations:

1. Create groups to define the tables that you want to copy to the subset database. A group defines a set of unrelated tables.
2. Assign data masking rules to columns in the data source.
3. Create a data subset plan, and add the groups to it.
4. Create a data masking plan and assign the policies and rules to the plan that you want to apply.

The TDM Server stores projects, groups, and plans in the TDM repository. When you generate and run workflows from plans, the Data Integration Service runs the workflows and loads the data into the target database.

Create a Plan for Data Masking and Data Subset

Create a plan for the data masking and data subset operations. A plan includes the components that you need to generate a workflow. You can combine a data masking and a data subset operation in the same plan, or you can create separate plans.

1. Create a data subset plan and add the groups to it.
2. Create a data masking plan and assign the policies and rules to the plan that you want to apply.
3. Generate a workflow from the plan.
4. Run the workflow.

When you generate and run workflows from plans, the Data Integration Service runs the workflows and loads the data into the target database.

Monitor the Workflow

Monitor workflow progress and monitor progress and logs of other jobs such as importing metadata and profiling in the Monitor view. Each workflow appears as a job in the Monitor view.

Access the Monitor view to determine the status of the workflow jobs. You can run the Row Count Report on a successfully run workflow to view the number of rows that a plan affects. View the workflow job status. Access the TDM job log to troubleshoot problems.

CHAPTER 2

Test Data Manager

This chapter includes the following topics:

- Test Data Manager Overview, 22
- Test Data Manager User Interface, 23
- Overview View, 25
- Policies View, 27
- Projects View, 27
- Monitor View, 27
- Administrator View, 28
- Expression Builder, 28
- Logging In to Test Data Manager, 29

Test Data Manager Overview

Test Data Manager is a web-based user interface that you use to perform data discovery, data subset, and data masking operations. Open a view in Test Data Manager based on the task you need to perform.

A compliance officer uses Test Data Manager **Policies** view to create policies, data masking rules, and data domains. The compliance officer assigns the data masking rules to data domains.

A developer uses the **Projects** view in Test Data Manager to define a project and import the data sources to the project. The developer runs profiles to discover data domain assignments. The developer assigns rules to columns for data masking operations. The developer creates plans, generates workflows, and runs the workflows.

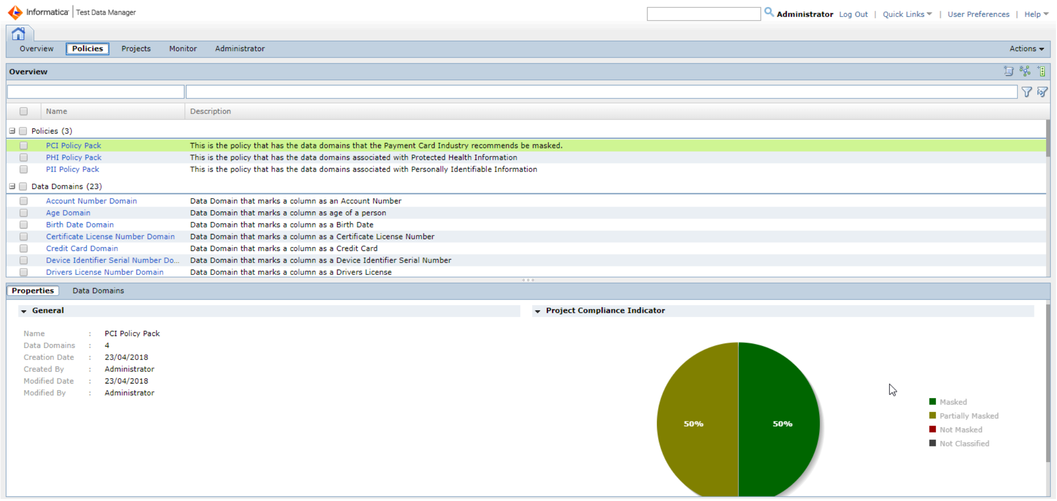
The developer opens the **Monitor** view to check the status of jobs that perform data masking, data subset, and other operations.

An administrator uses the **Administrator** view to create connections, configure workflow options, perform server management, and restrict user access to TDM components.

Test Data Manager User Interface

Test Data Manager contains options to view and edit TDM components.

The following image shows a view in Test Data Manager:



Name	Description
Policies (3)	
PCI Policy Pack	This is the policy that has the data domains that the Payment Card Industry recommends be masked.
PHI Policy Pack	This is the policy that has the data domains associated with Protected Health Information
PII Policy Pack	This is the policy that has the data domains associated with Personally Identifiable Information
Data Domains (23)	
Account Number Domain	Data Domain that marks a column as an Account Number
Age Domain	Data Domain that marks a column as age of a person
Birth Date Domain	Data Domain that marks a column as a Birth Date
Certificate License Number Domain	Data Domain that marks a column as a Certificate License Number
Credit Card Domain	Data Domain that marks a column as a Credit Card
Device Identifier Serial Number Do...	Data Domain that marks a column as a Device Identifier Serial Number
Drivers License Number Domain	Data Domain that marks a column as a Drivers License

Name : PCI Policy Pack
Data Domains : 4
Creation Date : 23/04/2018
Created By : Administrator
Modified Date : 23/04/2018
Modified By : Administrator

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.

Views

Access Test Data Manager views to perform tasks such as defining data masking policies or configuring projects.

Test Data Manager contains the following views:

Overview

View dashboard reports about projects in the TDM repository.

Policies

Define policies and masking rules that you can add to projects.

Projects

Define a project that contains source data and the data subset, data masking, or data profiling operations for the data.

Monitor

View the status of jobs that import sources or perform data subset, data masking, or data profiling operations. Stop or abort jobs.

Administrator

Manage connections, dictionaries, and workflow options.

Note: By default, an administrator can access the **Administrator** view of Test Data Manager. A user must have privileges to access the other views in Test Data Manager.

Search Field

Use the Search field to search for objects within the TDM repository. Search for objects such as projects, plans, and assignments. You cannot search for connections.

Enter the name or part of the name of the object you want to find. A search tab opens where you can filter the types of objects to include in the search. A search count displays the number of objects that match the search criteria. The search results return all objects and assignments contained within it and objects that contain it. For example, when you include projects within the search objects, the results also list objects such as policies and rules that the project contains. When you search for a masking rule, the results include the rule, assignments, objects that contain the rule, and dictionaries in the rule.

You can search for objects other than connections globally from the Search field. You cannot search for objects when indexing tables.

Advanced Search Text Filter

In Test Data Manager, you can enter a text instead of the wild characters in the filter criteria to search for columns.

Based on the type of filter operators, TDM searches and displays the results.

The following table lists the sample formats that the filter supports:

Filter Type	Sample Text	Operator Type
Exact match	"EMPLOYEE_FIRSTNAME"	Equality operator
Starts with	EMPLOYEE*	Like operator and type is STARTS_WITH
Ends with	*SALARY	Like operator and type is ENDS_WITH
Contains	BONUS	Like operator and type is CONTAINS

The following table lists the Test Data Manager views and pages in which you can use the advanced filter:

Test Data Manager Page	Path in Test Data Manager	Advanced Filter Supported Columns
Discover | Column	Project -> Discover -> Columns	Table Name, Column, Owner, Data Type, and Domain
Define | Data Masking	Project -> Define -> Data Masking	Name, Owner, Columns, Data Type, Domain, Policy, and Masking Rule

Quick Links Menu

You can quickly navigate to content tabs in Test Data Manager from the **Quick Links** menu.

From the **Quick Links** menu, you can open tabs to manage data subset, and masking. You can use the **Quick Links** menu to access shortcuts to create a masking rule, policy, data domain, or project.

The items that you can access from the **Quick Links** menu change based on user permissions.

User Preferences Menu

The **User Preferences** menu contains options that you can use to modify Test Data Manager.

Click **User Preferences** to view the **Edit Preferences** dialog box. The **Edit Preferences** dialog box contains the following tabs:

General

Displays general Test Data Manager preferences. Select **Show Start Screen** to display the **Informatica Test Data Manager** start screen when you log in to Test Data Manager.

Projects

Displays a list of projects that are visible in the dashboards in the **Overview** view. Disable projects in the **Projects** tab that you do not want to appear in the **Overview** view.

Actions Menu

Click the **Actions** menu to modify data that you select in the Content panel or to perform tasks such as importing and exporting objects.

You can choose different options in the Actions menu based on the Test Data Manager view.

Data Filter

You can filter data that appears in the **Content** and **Details** panel of Test Data Manager views.

Test Data Manager **Content** and **Details** panels might contain multiple rows of data. For example, the Policies view might show 25 data domain names. You might want to limit the data domain names to names that include "Emp." To apply the filter, type "Emp" in the filter field for the data domain name. Click the **Filter** icon. The **Policies** view shows data domains such as Employee_Domain or Emp_Number.

To reset the filter results, click the **Reset Filter** icon.

You can apply filters to different columns of data in the **Contents** panel based on Test Data Manager view.

Overview View

View dashboard reports about projects in the **Overview** view. The dashboard reports are graphs that show the distribution of data masking, and data subset objects in projects. You can use links in the **Overview** view to access project and the policy objects.

Dashboards

The **Overview** view shows the TDM dashboards. The dashboards are summary reports about projects, data masking objects, and administration objects.

You can view the distribution of data by sensitivity level in projects. You can view the distribution of data masking, data domains, policies, and rules in projects. View the distribution of groups in data subset projects.

You can view the number of data masking policies, the number of data domains, and the number of rules in the TDM repository.

Project Risk Analysis

The **Project Risk Analysis** dashboard shows the percentage of columns that are in each domain sensitivity level. It also shows the percentage of columns that do not belong to domains.

When you create a data domain, you select the sensitivity level for all columns in the domain. You can select a sensitivity level from the levels configured by the administrator.

The **Project Risk Analysis** dashboard shows the percentage of columns that belong to domains at each sensitivity level. You can view the projects that contain the columns. Move the pointer over the section of the pie chart that you want to view. The dashboard shows the number of domains in that sensitivity level, and the number of projects that contains columns in the domains. Click the **Number of Projects** link to view a list of the projects.

Project Assignments and Project Objects

The Project Assignments and Project Objects dashboard shows graphs of the number of objects and assignments in up to 10 recently accessed projects.

The Project Assignment tab displays by default. This graph displays the number of assignments, including data domains, policies, and masking rules, in the last 10 accessed projects. The Project Objects tab displays the number of subset objects, including groups, in up to 10 recently accessed projects.

Recent Project Sensitive Data Distribution

The **Recent Project Sensitive Data Distribution** dashboard shows a graph of the sensitive domain information in the last 10 accessed projects.

You must have set data domain sensitivity levels to view the distribution information. If you do not set data domain sensitivity levels, no graph appears.

Recent Activities

The **Recent Activities** panel shows the last 10 Test Data Manager components that changed.

The **Recent Activities** panel shows the name of the TDM component that changed, the type of component, and who made the change. The change actions are created, updated, and deleted. The panel shows the date of the change.

You can view and download an Audit Trail report from the **Recent Activities** tab.

Plan Execution

The **Plan Execution** dashboard displays plan execution data for a particular date.

You can select the plan component, masking, or subset, for which to view data. You can view data for plans in progress, completed, or both. The default display is for the current date, all plan components, and both completed and in progress status. You can browse the calendar to choose a different date, or enter the specific date. Use the arrows to move the date forward or backward.

Global Components

The **Global Components** dashboard shows the proportion of policies, masking rules, and data domains, out of the total number of components available in the repository.

Use the **Global Components** dashboard to view the distribution of policies, data domains, and masking assignments in the repository in a pie chart.

Recent Projects

The **Recent Projects** dashboard lists up to the last 10 projects you modified. You cannot view information on projects created or modified by other users.

The **Recent Projects** dashboard shows the name of the modified project, the project description, the user name, and the date of modification. Click on the project name to open the project.

Policies View

Maintain policies, the data domains, and the data masking rules in the **Policies** view. Create a policy and add the data domains to the policy. You can apply the policy to multiple data masking projects in Test Data Manager.

The **Policies** view lists the policies, the data domains, and the rules in the TDM repository. Select a policy, a data domain, or a rule to change it. Or, choose to add a new policy, data domain, or rule, from the **Actions** menu.

Projects View

Maintain projects in the **Projects** view. A project is the top-level container that you use to organize the components for data discovery, data masking, and data subset operations.

From the **Projects** view, you can create a project and you can import source data for the project. Assign policies and rules to columns in the source data. Run profiles for data discovery against the data. Run data masking and data subset operations on the source data.

The **Projects** view shows a list of projects. You can select a project to view the data sources and recent activities.

When you select a project, you can update it. You can define profiles for data discovery, data masking, and data subset operations. You can generate a workflow from within a project.

Monitor View

In the **Monitor** view, you can review the status of jobs that perform tasks such as import data, run profiling, or run data masking, and data subset operations. You can review the TDM Server job log to investigate problems. You can also stop jobs and recover workflows in the **Monitor** view.

Select a job in the **Monitor** view and choose options from the **Actions** menu.

Administrator View

An administrator can set default settings, create connections, add and manage dictionaries, and perform TDM server tasks in the **Administrator** view.

An administrator can perform the following tasks in the **Administrator** view:

- Configure default settings including child and parent record settings.
- Configure data domain sensitivity levels for tracking the sensitive data that users need to mask.
- Add and modify default project configuration fields.
- Configure workflow properties.
- Create connections to source or target databases.
- Add and manage dictionaries.
- View and search application level logs.
- Set or modify the log levels to debug the logs.
- Configure connection and mapping details for a Hadoop connection.

Expression Builder

Use the **Expression Builder** to build an expression using functions.

Click **Expression** to open the **Expression Builder** when you perform any of the following tasks:

- Create masking rules with Expression masking type.
- Add pre-processing or post-processing expressions to a standard masking rule.
- Create advanced masking rules.
- Assign rules to a column.

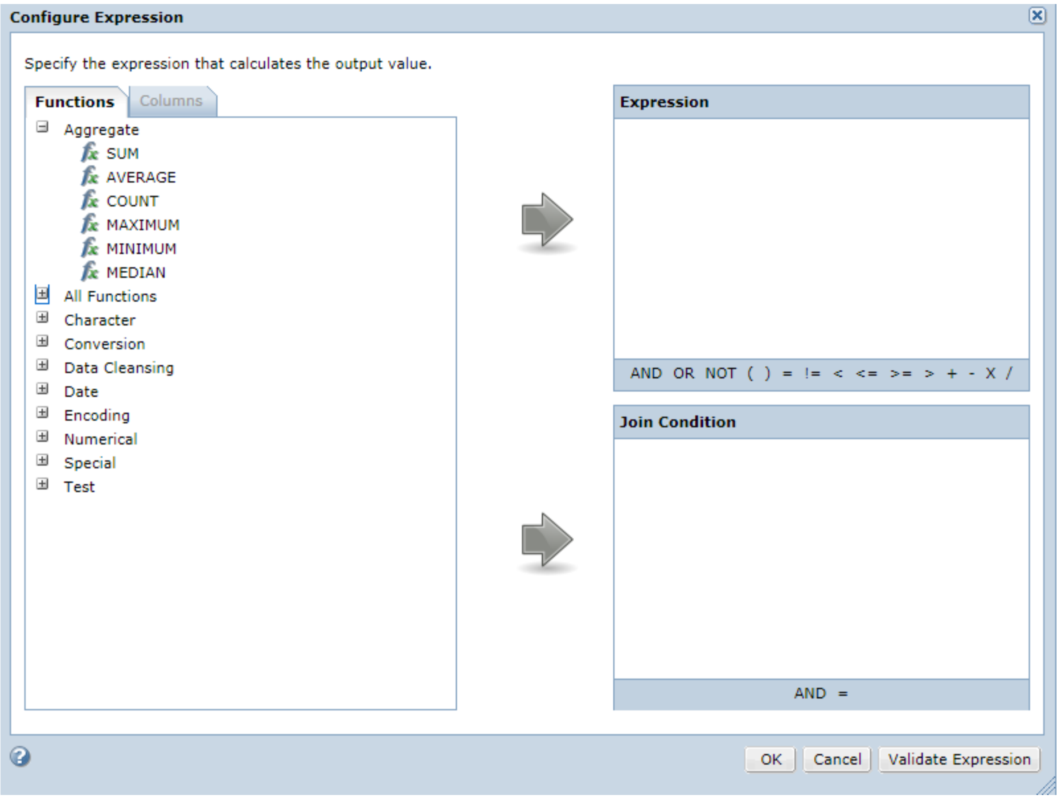
The **Expression Builder** shows a list of functions by category. To view the complete list, expand the **All Functions** list.

You can choose from a list of available columns to include in an expression. Select a function and click the **Add** arrow to add the function to an expression. You can choose columns and operators to enter join conditions.

For example, you might create an advanced masking rule that contains a first name and a last name input port. The masking rule has two variable ports that receive masked values for the first name and the last name. You might create an expression for the output port that concatenates the masked names in the variable ports and includes a space between them:

```
CONCAT( CONCAT( VAR1, ' ' ), VAR2 )
```

The following image shows the **Expression Builder** where you configure expressions:



Configure Expression

Specify the expression that calculates the output value.

For more information about the function syntax, see the *Informatica Transformation Language Reference*.

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**.

CHAPTER 3

Projects

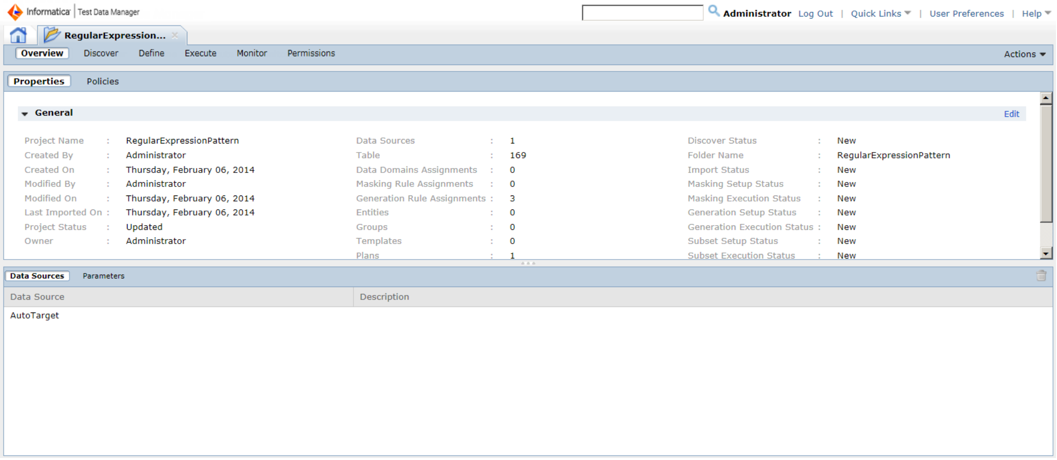
This chapter includes the following topics:

- Projects Overview, 31
- Project Components, 32
- Project Logs, 33
- Project Management, 33
- Data Sources, 36
- Delete a Table, 40
- Project Permission and Security, 40

Projects Overview

A project is the top-level container that you use to organize the components for data discovery, masking, and subset operations.

The following image shows a project in Test Data Manager:



Project Name : RegularExpressionPattern
Created By : Administrator
Created On : Thursday, February 06, 2014
Modified By : Administrator
Modified On : Thursday, February 06, 2014
Last Imported On : Thursday, February 06, 2014
Project Status : Updated
Owner : Administrator

Data Sources : 1
Table : 169
Data Domains Assignments : 0
Masking Rule Assignments : 0
Generation Rule Assignments : 3
Entities : 0
Groups : 0
Templates : 0
Plans : 1

Discover Status : New
Folder Name : RegularExpressionPattern
Import Status : New
Masking Setup Status : New
Masking Execution Status : New
Generation Setup Status : New
Generation Execution Status : New
Subset Setup Status : New
Subset Execution Status : New

Data Source	Description
AutoTarget	

A project contains the following views:

Overview

Edit the project general properties in the **Properties** view. Add policies and rules to the project in the **Policies** view. In the **Data Sources** view, import data sources to a project for data discovery, data subset, and data masking operations. You can import a source from the Model repository, or you can import a source from a database. You can import multiple types of sources to the same project and define relationships between them.

Discover

Discover the data domains in the source data.

Define

Define data masking and data subset operations. When you define a data masking operation, assign the rules and policies to columns in the project source. When you need to create a data subset, define groups.

Execute

Define a data subset or data masking plan. Generate and run a workflow from the plan.

Monitor

View the status of data source import jobs, profiling jobs, workflow generation jobs, data subset jobs, and data masking jobs. You can refresh and abort a job. You can abort, stop, and recover a workflow. You can remove a job that is scheduled to run at a later time.

Permissions

Apply the user group and the user permission to projects. When you create a project, you are the project owner and have access to the project. If other users need access to the project, you can add the users in the **Permissions** view. Apart from permission on a specific project, users need the minimum required privileges to access or perform any tasks on the project.

Project Components

A project contains one or more data sources. Other components that you add to a project depend on the operations that you need to perform on the data.

The following table describes the components that you can create in a project:

Component	Operation	Description
Data Source	Data subset Data discovery Data masking	Defines the input data that you want to transform.
Group	Data subset	Defines a set of unrelated tables to copy to a target subset database with data subset. Create a Group in the **Define** view of a project.
Plan	Data masking Data subset	Defines data subset or data masking operations. Generate a workflow from a plan. Define a plan in the **Execute** view of a project.
Policy assignments	Data masking	Assigns policies to the project. When you assign a policy to a project, the project receives the data domains and the data masking rules in the policy.

Component	Operation	Description
Profile	Data discovery	Suggests the data domains in a source based on source data and metadata. Define a profile in the **Discover** view of a project.
Rule assignments	Data masking	Defines which data masking rules apply to a source column.

Project Logs

You can configure the location where you want to save the reject files for a project. If you do not specify a location, Data Integration Service saves the reject files in the default location set in the Data Integration Service properties.

The Data Integration Service stores the reject files in the following default location:

- Reject file directory: `$PMBadFileDir\`

You can configure the location where you want to save reject files from the **Edit Project** dialog box. You can specify the location in an existing project. You cannot specify the location when you create the project.

You can specify the type of error log files to generate error logs and the location to store error logs for a specific plan when you create the plan.

Project Management

When you create a project, you add one or more sources to the project. When you edit a project, you can update the name, description, folder location, and associated sources. You can also edit the location where you want to store the log files of the project.

When you create a project, you are the project owner. You have access to the project folder by default.

You can export a project and project components to an XML file. You can import the data from the XML file into Test Data Manager. You might export and import a project to back it up, or you might need to move a project from a development to a production system.

Creating a Project

Create a project folder to contain the data discovery, subset, and masking components to apply to a data source.

1. In Test Data Manager, click **Projects** to access the projects.

   A list of projects appears.

2. Click **Actions** > **New**.

3. In the **Create Project** dialog box, enter project properties. The following table describes the project properties:

Option	Description
Name	The name of the project. The project name must not contains spaces if you want to perform TDM operations on Hadoop data sources.
Description	The description of the project.
Folder	The name of the project folder in the repository. Default is the project name. You can choose another folder in the repository.
Owner	The name of the user that owns the folder. The folder owner has all permissions on the folder. The default is the name of the user that created the folder. You can select another user as the folder owner.

4. Click **OK**.

   The properties for the project appear in Test Data Manager.

Editing a Project

You can edit a project to change its name, description, or associated sources.

1. To access the projects, click **Projects**.

   A list of projects appears.
2. Click on the project that you want to edit.

   The project opens in a separate tab with project properties and data sources details.
3. To change the project properties, click **Edit**.

   You cannot remove a source that is used by groups in the project.
4. In the **Edit** dialog box, change project options. The following table describes the project options that you can change:

Option	Description
Name	The name of the project.
Description	The description of the project.
Folder	The name of the project folder in the repository. Default is the project name. You can choose another folder in the repository.
Owner	The name of the user that owns the folder. The folder owner has all permissions on the folder. The default is the name of the user that created the folder. You can select another user as the folder owner.

5. Click **Save**.

Copying a Project

You can create a project by copying a project. When you copy a project, Test Data Manager creates a copy of the project but does not import the metadata or objects of the project.

1. To access the **Projects** view, click **Projects**.
2. Click a project description to select the project.

   Do not open the project.
3. Click **Actions** > **Duplicate**.

   The **Copy <Project Name>** dialog box appears.
4. Change the name and description of the project. Click **Save**.

Deleting a Project

When you delete a project, all sources that are associated with the project are deleted.

1. To access the projects, click **Projects**.

   The list of projects appears.
2. To select a project without opening it, click the Description or Created On column of the project.
3. Click **Actions** > **Delete**.
4. In the **Confirm Delete** dialog box, click **Yes**.

Configuring Project Log Locations

You can configure the location where you want to save the reject files that the Data Integration Service generates for project workflows.

1. Open the project and click **Actions** > **Edit**.

   The **Edit Project** dialog box opens.
2. Click the **Default Log File** tab.
3. Perform the following tasks to change the location of the reject files:
   - In the **Reject File Directory** field, enter the location where you want to store reject files.

   You can enter a fully defined path. You must add a delimiter with a slash or backslash at the end of the file path.

   Note: The directory path must exist. TDM does not create the directory.
4. Click **Save**.

Exporting a Project

You can export a project to an XML file and import the XML file to another TDM repository. When you export a project, the XML file contains all the project components, such as the source information, the connections, the constraints, the data domains, and the assignments.

1. To access the projects, click **Projects**.

   A list of projects appears.
2. Click a project to view the project properties and the data sources.
3. Click **Actions** > **Export**.

4. Choose the name and path of the XML file to create.
5. Click **OK**.

Importing a Project

You can import a project from an XML file that was exported from another TDM repository.

Before you import a project into Test Data Manager, you must import all the global components that are assigned in the project.

1. To access the projects, click **Projects**.
2. Click **Actions** > **Import**.
3. Browse for the XML file that contains the project to import. Click **Finish**.

   Test Data Manager imports the project.

Data Sources

To perform data subset, masking, and discovery operations, you must import source metadata into the TDM repository. You can import sources from the Model repository or from a source database.

When you create a project, add one or more sources to the project. You can add more than one type of source to the project. For example, you can add a flat file source and a relational source to the project. You can create constraints to create relationships between the sources and apply filter criteria for data subset and data masking.

Note: If the source data has a numeric column with precision greater than 28, you cannot apply a data masking or data subset condition to that column. The Data Integration Service passes the row and the column to the target without applying the condition.

TDM Source Definitions

Import source metadata from the Model repository or directly from the source database to the TDM repository. The TDM project and the Model repository folders share the same name to ensure that the Data Integration Service stores workflow information in the correct Model repository folder.

You can import source metadata directly from the source if the source is a relational database. If the source contains many tables, you can increase the performance if you import the metadata directly from the source instead of from Model repository.

When you import source definitions from the Model repository into the TDM repository, you can import all the definitions supported by the Model repository. When you import source metadata from the Model repository, you import the Model repository folder that contains source metadata. Test Data Manager creates a copy of the folder in the TDM repository.

To use flat file sources, you must import flat files from the Model repository. You cannot create flat file connections in Test Data Manager.

Note: Rowid is a reserved keyword. You cannot import Oracle sources that have rowid as a column name or a table name.

Hive and HDFS Data Sources

You can perform data movement, data domain discovery, and data masking operations on Hive and Hadoop Distributed File System (HDFS) data sources.

You can use Hive and HDFS connections in a Hadoop plan.

You can create Hive and HDFS connections in Test Data Manager, and import the Hadoop data sources in to a project. In a Hadoop plan, you can select Hive and HDFS connections as source, target, or both.

You must configure a cluster configuration in the Administrator tool before you perform TDM operations on Hive and HDFS sources. A cluster configuration is an object that contains configuration information about the Hadoop cluster. The cluster configuration enables the Data Integration Service to push mapping logic to the Hadoop environment.

The Hive database schema might contain temporary junk tables that are created when you run a mapping. The following sample formats are the junk tables in a Hive database schema:

```
w1413372528_infa_generatedsource_1_alpha_check
```

```
w1413372528_write_employee1_group_cast_alpha_check
```

Ensure that you do not select any temporary tables when you import data sources.

You can create a Hadoop plan to move data from Hive, HDFS, flat files, or relational databases such as Oracle, DB2, ODBC-Sybase, and ODBC-Microsoft SQL Server into Hive or HDFS targets. You can also create a Hadoop plan when you want to move data between Hive and HDFS sources and targets. If the source is HDFS, you can move data to a Hive or an HDFS target. If the source is Hive, you can move data to a Hive or an HDFS target. You can extract data from Hive and HDFS to a flat file in a Hadoop plan.

To run a Hadoop plan, TDM uses Data Integration Service that is configured for pushdown optimization. When you generate and run the Hadoop plan, TDM generates the mappings and the Data Integration Service pushes the mappings to the Hadoop cluster to improve the performance. You can use a Blaze or a Hive execution engine to run Hadoop mappings. When you select an HDFS target connection, you can use Avro or Parquet resource formats to mask data.

Hive Inplace Masking

You can perform an inplace masking operation on Hive data sources. Use a Hive or a Spark execution engine to run the mappings in the cluster. You can use all the data masking techniques while you perform an inplace masking on Hive data sources with a Hive execution engine. When you use a Spark engine, you cannot perform shuffle and substitution masking.

Before you perform an inplace masking operation on Hive data sources, you must take a backup of source tables. If the data movement from staging to source tables fails, TDM truncates source tables and there might be loss of data.

Avro and Parquet Data Sources

When you select an HDFS target connection, use Avro or Parquet resource formats to mask data and to move data in groups.

Avro and Parquet are semi-structured data sources. Apache Avro is a data serialization system in binary or other data formats and the Avro data is in a format that might not be directly human-readable. Apache Parquet is a columnar storage format that can be processed in a Hadoop environment and uses a record shredding and assembly algorithm. Use Avro and Parquet sources for single-level hierarchy files.

You can move data into the target with Avro and Parquet resource formats when you use a Hive, a Blaze, or a Spark engine.

If you use Parquet format, you cannot use null or repeated constraints. The table must not contain any null value in a column or a row. If there is any such column, you must restrict the column before data ingestion. You cannot run profiles on Avro and Parquet source formats.

Execution Engines

Use a Blaze, a Spark, or a Hive engine to run the Hadoop mappings in a workflow.

The Data Integration Service generates the Blaze, Spark, or Hive engine script based on the mapping logic, a unique identifier for the script, and the tasks that the script depends on.

You can select the execution engine at the plan level. If you select the Hive execution engine, Hive assigns the mapping job to MapReduce. If you select the Blaze execution engine, the processing is faster because Blaze uses an internal workflow compiler to run the mapping. Use a Blaze engine to improve the speed and performance of the task.

If you do not use Kerberos authentication, you can use a Blaze engine for complex file targets. In Hive inplace masking, you can use Hive or Spark execution engines.

If you use a Hive or a Blaze engine, you can use the following transformations in a mapplet rule:

- Expression
- Data Masking
- Case Converter
- Comparison
- Decision
- Labeler
- Merge
- Parser
- Weighted Average
- Standardizer
- Java Passive

If you use a Spark engine, you can use the following transformations in a mapplet rule:

- Expression
- Data Masking
- Java Passive

You cannot use a Blaze engine for the following options:

- ODBC sources and ODBC dictionaries
- Complex file target if you use Kerberos authentication
- Truncate target table
- Source is Hive and target is HDFS
- Hive inplace masking

The Spark engine has the following limitations:

- You cannot use a Spark engine when the sources are relational databases such as Oracle, Sybase, Microsoft SQL Server, and DB2 for Linux, UNIX, and Windows.
- You cannot perform shuffle and substitution masking with a Spark engine.
- With the Spark engine, you cannot perform data masking operations on the Binary data type in Hive.

Importing Data Sources

Import data sources from the Model repository or from an external database to a project in Test Data Manager.

1. To view a list of the projects in Test Data Manager, click **Projects**.
2. Click a project in the list to open it.
3. Click **Actions** > **Import Metadata**.

   The **Import Metadata** window appears.
4. Choose one of the following options:
   - Model Repository. Import metadata from the Model Repository.
   - Datasource Connection. Import metadata from a database connection.
5. Choose whether you want to review the metadata changes before you import the data sources. Test Data Manager displays the information before the import runs. You can choose to skip the import option.

   Test Data Manager shows you the metadata you are importing and the impact on rules and domain assignments or entities and groups in Test Data Manager.

   Test Data Manager shows you the metadata you are importing and the impact on domain assignments or groups in Test Data Manager.
6. If you select an Oracle database connection, you can choose to ignore the tables that do not contain any data.

   Test Data Manager excludes all the empty tables in the Oracle database and lists the tables that contain data.
7. Click **Next**.
8. Choose one of the following options:
   - If you selected **Model Repository**, select the Model Repository folder that contains the data source you want to import. You can filter folders by the folder name or the description.
   - If you selected **Datasource Connection**, select the schema to import. You can filter schemas by schema name.
9. Click **Next**.
10. Select the files or tables that you want to import. You can filter the tables by data source, table name, or table description. If you choose a Hive database and if there are temporary junk tables present in the schema, ensure that you do not select those tables.
11. Click **Next**.
12. Choose when to import the sources. Choose one of the following options:
    - Import Now. Import the data source immediately. To run the import in the background, select **Run import in the background**.
    - Schedule Later. Schedule the import to occur at a specific date and time. Click the calendar to select a date. Use the hour and minute sliders to set the time.
13. Click **Finish**.

    The **Import Progress** dialog box appears with the import progress bar.
14. View the progress of the import job in the **Monitor** view.

    If you chose to import review changes, the **Impact Review** dialog box appears with the metadata, assignments, and project objects.

15. Review the impact, and approve or reject the changes. If you approve the changes, TDM imports the metadata successfully.

    After the job finishes, access the imported metadata through the **Data Sources** details view.

Delete a Table

You can delete a table from a project if you added too many tables or some tables are not necessary after a period of time.

You can directly delete an orphan table that is not related to another tables and does not contain any assignments. If you delete a table that is a part of profile results, a warning appears and TDM deletes the table from the profile results as well.

You can delete a table that is a part of a group, or a plan. If the table is part of a group or a masking plan, the **Impacted Objects** dialog box appears with the list of affected objects.

You can delete one table at a time.

Deleting a Table

You can delete a table that you added in a data source if you do not use them in a TDM operation.

1. Open a project.
2. Click **Discover** > **Tables**.
3. Select a table that you want to delete.
4. Click **Actions** > **Delete Table**.

   A confirmation message appears if the table does not contain any assignments.
   If the table is part of a group or a masking plan, the **Impacted Objects** dialog box appears with the list of affected objects.
5. To confirm and delete the table, click **Continue**.

Project Permission and Security

Use project permissions to control access to projects.

When you create a project, you become the owner of the project. As the project owner, you can add users and user groups and assign the required levels of permission. The domain administrator can also add and edit project permissions.

You can access projects based on the permissions that you have. For example, if you do not have any permissions for a project, it does not appear in the list of projects. You cannot view projects that appear in search results if you do not have the required permissions.

Projects have the following levels of permission:

- Read
- Write

- Execute

Project Permissions

You must have the required project permissions to access and perform tasks in a project.

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

Updating User and Group Security

When you create a project, you can assign read, write, and execute permissions to users and user groups. Edit the project permissions assigned to users and user groups from the **Permissions** tab of the project. Changes to permissions take effect from the subsequent login.

1. Open a project and click **Permissions**.

   A list of the users and user groups with permissions for the project appears.
2. Click **Edit** on the **Users** or **User Groups** tab.

   The **Edit Project Permissions** dialog box opens.
3. 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.
4. To delete a user or user group, select the user or user group from the list and click **Delete**.
5. 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.

6. Click **OK**.

CHAPTER 4

Policies

This chapter includes the following topics:

- Policies Overview, 43
- Policies View, 43
- Policies Task Flow, 44
- Rules, 44
- Data Domains, 44
- Policy Packs, 48
- Import and Export, 51
- Policy Management, 53

Policies Overview

A policy is a data masking component that describes the methods to maintain the privacy of specific types of source data.

A policy contains data domains. A data domain describes the functional meaning of a column based on the column data or the column name. For example, a Social_Security data domain contains all database columns with numbers in the following format: 999-99-9999. A Salary data domain might include the Salary, Bonus, and Commission columns in a database.

A data domain contains data masking rules. A data masking rule is a data masking technique to mask a specific type of data. For example, you might configure the Substitution data masking technique for First Names and Last Name columns. You configure two Substitution masking rules because each rule contains different parameters.

You can configure data domains, rules, and policies separately. Apply the rules to data domains and add the data domains to a policy. After you define the policy, you can assign the policy to a data source in a project. You can apply a policy to multiple projects.

Policies View

The **Policies** view shows the policies, data domains, and rules in the TDM repository.

Policies Task Flow

You can create policies, rules, and data domains in any order.

Complete the following high-level steps to define policies:

- Create data masking rules.
- Define data domains to describe which columns can receive the same masking rules.
- Assign the data masking rules to the data domains.
- Create a policy.
- Assign the data domains to the policy.

Rules

A rule defines a logic to mask sensitive data.

Create a data masking rule and configure the rule parameters. After you add rules to a project, you must assign the rules to columns based on the data types.

You can create rules for string, numeric, and date data types.

You can add the rules to data domains, and use the data domains in policies and plans.

Data Domains

A data domain is an object that represents the functional meaning of a column based on the column data or the column name. Configure data domains to group data source columns for data masking. You can assign a masking rule to a data domain and all the columns in the data domain are masked with the same rule.

Create data domains to describe the columns you need to mask with the same masking rules. Assign at least one masking rule to each data domain.

For example, you might need to mask all the instances of Social Security number with the same masking rule. You can create a data domain that describes the Social Security data that occurs in the different columns. A database might have a Social Security number in a column called SSN. The database also has a column called SOCIAL_SECURITY in a different table. A Social Security number might also appear in a COMMENTS column.

When you create the data domain, you create a data expression that describes the data format for Social Security numbers. A Social Security number has this format: `999-99-9999`. You can also create multiple metadata expressions that describe possible column names for Social Security numbers. Social Security column names might include `SSN` or `Social`.

After you define a data domain, you can add the data domain to a policy. You can run profiles for data discovery against data sources in a project. Run profiles to find the columns for data domains. For example, the profile job can find all the Social Security numbers in the source data based on how you defined the data domain. The profile assigns data domains to columns.

Note: If you do not have Data Discovery, you can still use data domains to aggregate data. However, you must manually associate source columns with the data domains.

Apply Masking Rules to a Data Domain

You can assign one or more data masking rules to the data domain. When you assign a masking rule to a data domain, the columns in the domain receive the data masking rule when you configure data masking.

When you assign data masking rules to the data domain, the rules are called preferred rules. If you assign multiple rules to the data domain, you enable one of the rules to be the default rule. The default rule is applied to all columns in the data domain. You can manually change the masking rule for a column to a different preferred rule. You can also apply more than one masking rule to a column.

For example, an organization has a data domain called Last_Name. The Last_Name data domain describes columns that contain last names in company databases. The company can use a shuffle masking rule to mask the last names of customers in a database. The shuffle masking rule is the default rule. The organization applies a substitution masking technique to mask the last names of customers in a different table. The substitution masking rule is a different preferred masking rule in the data domain.

Metadata and Data Patterns for Data Domains

A data pattern and a metadata pattern are regular expressions that you configure to group columns into a data domain. Use regular expressions to find sensitive data such as IDs, telephone numbers, postal codes, and Social Security numbers in the source data.

A regular expression is a text string that describes a search pattern. A regular expression provides a way to match strings of text or patterns of characters in the source data.

A data domain expression can contain data expressions and metadata expressions. A data expression identifies data values in a source. A metadata expression identifies column names in a source. When a data domain contains multiple expressions, any column name or column value that matches an expression in the pattern appear in the search results.

Regular Expression Syntax

A regular expression contains characters that represent source character types, source character sets, and string or word boundaries in the source columns. A regular expression can also contain quantifiers that determine how many times characters can occur in the source data. Regular expressions are case sensitive.

The following special characters are examples of characters that you can include in a regular expression:

Any character except [\^$.|?*+()

All characters except the listed special characters match a single instance of themselves. For example, `abc` always matches `abc`.

\ (backslash) followed by any of the following special characters: `[\^$.|?*+(){}`

A backslash escapes any special character in a regular expression, so the character loses the special meaning.

*** (asterisk)**

Matches the preceding token zero or more times.

[(left bracket)

Marks the beginning of specifications for one character that you want to match.

- (hyphen)

Specifies a range of characters. For example, [a-zA-Z0-9] matches any letter or digit.

] (right bracket)

Marks the end of the specifications for one character.

? (question mark)

Makes the preceding item optional.

{n} where n is an integer > = 1

Repeats the previous item n times.

For information about creating regular expressions, see tutorials and documentation for regular expressions on the internet such as http://www.regular-expressions.info/tutorial.html.

Data Patterns

Data patterns are regular expressions that describe the format of source data in a data domain.

A data pattern can contain multiple data expressions. If any of the expressions match patterns of the data for a column, then the column belongs in the data domain. You can configure detailed regular expressions to identify data in columns.

For example, a Social Security number contains numbers in the following pattern:

```
999-99-9999
```

The following regular expression shows a data pattern that describes the format of a Social Security number:

```
[0-9]{3}-[0-9]{2}-[0-9]{4}
```

Metadata Patterns

A metadata pattern is a regular expression that identifies column names in a source. A metadata pattern can contain multiple metadata expressions.

A metadata expression can be a column name or part of a column name. For example, if you configure `.*Name*` as a metadata expression, column names such as Name, Employee_Name, and Organization_Name in the source appear in the search result.

A column name that matches any metadata expression in the pattern appears in the search results.

A Social Security number might have different column names. The following regular expressions are metadata expression to find Social Security numbers by column name:

```
.*SSN*
.*SOCIAL*
.*SECURITY*
```

Data Domain Options

When you create a data domain you configure options that describe the data domain.

Configure the following options to describe a data domain:

Name

Data domain name.

Sensitivity level

The sensitivity level for all columns in the data domain. The Administrator defines the sensitivity levels that you can choose from when you apply the sensitivity level option.

Description

Description of the data domain.

Status

Data domain status is enabled or disabled. When the data domain is enabled, a profile for data discovery includes the data domain. Default is enabled.

Creating a Data Domain

When you create a data domain, you can enter regular expressions that describe the data that you want to include in the data domain. You can also enter regular expressions that describe the names of database columns to include.

1. To access the policies, click **Policies**.

   The **Policies** view shows a list of the policies, data domains, and rules in the TDM repository.
2. Click **Actions** > **New** > **Data Domain**.
3. Enter the name, sensitivity level, and description for the data domain. Click **Next**.
4. Click **Next**.
5. Optionally, enter a regular expression to filter columns by data pattern.
6. To add more expressions for data patterns, click the **+** icon.
7. To add regular expressions that filter columns by column name, click **Next**. Or, click **Finish** to skip entering any more data domain information.

   You can add multiple expressions.
8. Enter regular expressions to filter columns by column name.
9. Click **Next** if you want to apply preferred masking rules to the data domain. Or, click **Finish** to finish configuring the data domain.
10. To add preferred masking rules to the data domain, click **Add Rules**.

    The **Add Rules** dialog box appears.
11. Select the data masking rules that you want to add.
12. Click **OK**.
13. Enable a default masking rule.
14. Click **Finish**.

Copying a Data Domain

You can create a data domain by copying a data domain.

1. To access the policies, click **Policies**.
2. Click a data domain description to select the data domain.

   Do not open the data domain.
3. Click **Actions** > **Duplicate**.

   The **Copy <Data Domain Name>** dialog box appears.
4. Change the name and description of the data domain. Click **Save**.

Editing a Data Domain

You can edit a data domain to update the rules, data patterns, and metadata patterns.

1. To access the policies, click **Policies**.

   The **Policies** view shows a list of the policies, data domains, and rules in the TDM repository.
2. Click the name of the data domain that you want to edit.

   The data domain opens in a tab.
3. Click **Actions** > **Edit**.

   The **Edit** dialog box appears.
4. To add or edit expressions for data patterns, click the **Data Patterns** tab.
5. To add or edit expressions for metadata patterns, click the **Metadata Patterns** tab.
6. To add or edit masking rules, click the **Preferred Rules** tab. Click **Save**.

   If you delete a rule that contains assignments, the **Impacted Objects** dialog box appears with the list of affected columns and plans.
7. To download the list of affected columns and plans, click **Export**, and save the .csv file.
8. To save the changes, click **Continue**.

   To update the changes in a plan, you must generate and run the plan again.

Deleting a Data Domain

When you delete a data domain, you delete the data domain assignments. When you delete a data domain, you do not delete the rules that you add to the data domain.

1. To access the policies, click **Policies**.

   The **Policies** view shows a list of the policies, data domains, and rules in the TDM repository.
2. Select the name of the data domain that you want to delete.
3. Click **Actions** > **Delete**.

   The **Delete Data Domain** dialog box appears. If you delete a data domain that contains assignments, the **Impacted Objects** dialog box appears with the list of affected columns and plans.
4. To delete the data domain that has no assignments, click **OK**.
5. To delete the data domain that contains assignments, click **Continue**. To download the list of affected objects, click **Export**, and save the .csv file.

   To update the changes in a plan, you must generate and run the plan again.

Policy Packs

A policy pack contains rules that mask common types of sensitive data in the business applications.

TDM includes the following three policy packs:

- PII
- PHI

- PCI

After you install TDM, the policy packs are visible in the **Policies** view. The policy packs include the data and metadata patterns of the data domains. You can view the regular expression strings that define the search patterns in TDM policy packs.

To view and work with policies, you must have the Test Data Manager Service View Policies and Manage Policies privileges. You cannot view the policies if you do not have at least the View Policies privilege.

PII Policy Pack

The Personally Identifiable Information (PII) pack contains data masking rules and policies specific to masking personal information.

The following table describes the data domains and the corresponding default rules available in PII policy pack:

Data Domain Name	Default Rule
Age Domain	Age Rule
Birth Date Domain	Birth Date Rule
Birth Place Domain	Birth Place Rule
Country Domain	Country Rule
Credit Card Domain	Credit Card Numbers Rule
Drivers License Number Domain	Drivers License Rule
Email Address Domain	Email Address Rule
First Name Domain	First Name Rule
Full Address UPPER Domain	Full Address UPPER Rule
Full Name Domain	Full Name Rule
Gender Domain	Gender Rule
Grades Domain	Grades Rule
IP Address Domain	IP Address Rule
Job Position Domain	Job Position Rule
Last Name Domain	Last Name Rule
Organization Name Domain	Organization Name Rule
Passport Domain	Passport Rule
Phone Domain	Phone Number Rule
Salary Domain	Salary Rule

Data Domain Name	Default Rule
School Name Domain	School Name Rule
Social Security Number Domain	Social Security Number Rule
State Domain	State Rule
Street Name Domain	Address Street Name Rule
UK National Identifier Domain	National Identifier Rule
Vehicle Registration Number Domain	Vehicle Registration Rule
ZIP Domain	ZIP Rule

PHI Policy Pack

The Protected Health Information (PHI) pack contains data masking rules and policies specific to the healthcare and the pharmaceutical industries.

The following table describes the data domains and the corresponding default rules available in PHI policy pack:

Policy Pack	Description
Account Number Domain	Account Number Rule
Birth Date Domain	Birth Date Rule
Certificate License Number Domain	Certificate License Number Rule
Device Identifier Serial Number Domain	Device Identifier Serial Number Rule
Email Address Domain	Email Address Rule
First Name Domain	First Name Rule
Health Plan Beneficiary Number Domain	Health Plan Beneficiary Number Rule
IP Address Domain	IP Address Rule
Last Name Domain	Last Name Rule
Medical Record Number Domain	Medical Record Number Rule
Phone Domain	Phone Number Rule
Primary Account Number Domain	Primary Account Number Rule
Social Security Number Domain	Social Security Number Rule
State Domain	State Rule

Policy Pack	Description
Unique Identifier Domain	Unique Identifying Number Rule
Web URL Domain	Web URL Rule
ZIP Domain	ZIP Rule

PCI Policy Pack

The Payment Card Industry (PCI) pack contains data masking rules and policies specific to the banking and the finance industries.

The following table describes the data domains and the corresponding default rules available in PCI policy pack:

Policy Pack	Description
Account Number Domain	Account Number Rule
Birth Date Domain	Birth Date Rule
Credit Card Domain	Credit Card Rule
Expiration Date Domain	Expiration Date Rule
First Name Domain	First Name Rule
Last Name Domain	Last Name Rule

Import and Export

You can export policies, domains, and rules to XML files and import the XML files to another TDM repository.

When you export a policy, Test Data Manager exports references to the data domains in the policy. When you export data domains, Test Data Manager exports references to the rules in the repository. However, Test Data Manager does not export the data domains or rules unless you choose to export them.

If a rule or a policy in TDM contains assignments, you cannot import the file. You must either clear the assignment or change the name of the file.

Test Data Manager exports the policies, data domains, and rules to separate XML files. Each file has a default name and number that includes the date. For example: `Policy_121112141309.xml` or `Domain_121112141309.xml`. You can rename the files before you save them.

Exporting Policy Components

Export policies, domains, and rules to separate XML files.

1. In the **Policies** view, choose the policies, data domains, and rules to export.

   To choose an object, select the check box next to the object name.
2. Click **Actions** > **Export**.
3. Click **Open** to open the XML file in a text editor or click **Save** to save the XML file on the system.

   Test Data Manager prompts you for each type of XML file you create. It downloads separate files file policies, data domains, and rules.

Importing Policy Components

You can import policies, data domains, and rules from XML files that have been exported from a TDM repository.

1. In the **Policies** view, click **Actions** > **Import**.
2. Browse for the XML file to import.
3. To import the file, click **Finish** .

   If a policy, data domain, or rule already exists in the TDM repository, Test Data Manager overwrites the original file.

   Note: If a rule or a policy in TDM contains assignments, you cannot import the file. You must either clear the assignment or change the name of the file.

Data Domain Import

Before you import a profile, verify that TDM contains the objects in the profile. So before you import a data domain profile, you must import the data domains in the profile. You can import data domains from the Model repository into TDM. You can edit the data domain in Test Data Manager after you import it.

If you edit a data domain in the Developer tool after you import it, you can import it again to update the data domain in TDM.

You can view data domains that you import in the list of data domains in the **Policies** view.

When you import a data domain from the Model repository, TDM imports the following information in the data domain:

- Name
- Description
- Links to business terms created in the Analyst tool

TDM does not import the following information in the data domain:

- Data pattern
- Metadata pattern
- Preferred rule

If a data domain with the same name exists in TDM, you can choose to overwrite it. If you continue, TDM imports and updates the description and linked business terms of the data domain. It does not overwrite the data patterns, metadata patterns, and preferred rules that you configured within Test Data Manager.

When you import a data domain, TDM merges the linked business terms and retains existing linked business terms. If you re-import a data domain after you delete a linked term from it in the Developer tool, TDM retains the linked term that you deleted. You can manually delete the linked business term from Test Data Manager.

If you re-import a data domain after you change the name of the data domain in the Developer tool or in Test Data Manager, TDM updates the name. You cannot re-import the data domain if a data domain in TDM has the same name.

You can edit data domains that you import in Test Data Manager. You can edit the description, or add a data pattern, metadata pattern, or preferred rule. You can add or delete links to business terms.

Importing a Data Domain

You can import a data domain from the Model repository into the TDM repository. Import data domains before you import a data domain profile.

TDM does not import the data patterns, metadata patterns, or the preferred rules of the data domain.

1. Click the **Policies** view.
2. Click **Actions** > **Import Data Domains**.

   The **Import Data Domains** window opens with a list of data domains in the Model repository.
3. Select the required data domains.
4. Assign a sensitivity level to each data domain that you want to import. Click the **Sensitivity Level** column and choose the required level from the list. Click **Save**.
5. Click **Import**.

   If a data domain with the same name exists, you are prompted to overwrite the data domain or cancel the import. TDM imports the data domain with the description and any linked business terms. You can view the data domain and its properties in the list of data domains.

Policy Management

The **Policies** view shows the policies in the TDM repository. You can create and edit a policy from the **Policies** view.

You can create a policy before or after you define data masking rules and data domains. After you create a policy, you can edit the policy to add data domains and rules. You can copy a policy if you need to create a policy with similar rules or data domains. The duplicate policy contains the rules and data domains from the original policy.

You can export a policy from a TDM repository to an XML file. Import the XML file in another TDM repository.

Creating a Policy

Create a policy in the **Policies** view. You can create the policy before you create data domains and rules. You can add the data domains and rules to the policy at any time.

1. In the **Policies** view, click **Actions** > **New** > **Policy**.

   The **New Policy** dialog box appears.
2. Enter a name and optional description for the policy, and click **Next**.
3. To add data domains to the policy, click **Add Data Domains**.

4. Select the data domains from the list.
5. Click **Finish**.

   The policy appears in the **Policies** view.

Copying a Policy

You can create a policy by copying a policy. When you copy a policy, Test Data Manager copies the data domains and rules in the original policy to the new policy.

1. To access the **Policies** view, click **Policies**.
2. Click a policy description to select the policy.

   Do not open the policy.
3. Click **Actions** > **Duplicate**.

   The **Copy <Policy Name>** dialog box appears.
4. Change the name and description of the policy. Click **Save**.

Editing a Policy

You can edit the policy properties and update the data domains in a policy.

1. To access the **Policies** view, click **Policies**.
2. Click the name of the policy that you want to edit.

   The policy opens in a tab.
3. Click **Actions** > **Edit**.

   The **Edit Policy** dialog box appears.
4. You can change the policy name, description, or status.
5. Click the **Data Domains** tab to edit data domains in the policy.
6. Click **Add** to add data domains to the policy.

   A list of data domains appears.
7. Select the data domains that you want to add to the policy.
8. Click **OK** to select the data domains.
9. Click **Save**.

   If you delete a data domain from the policy and if the data domain contains rule assignments, the **Impacted Objects** dialog box appears with the list of affected columns and plans.
10. To download the list of affected columns and plans, click **Export**, and save the .csv file.
11. To save the changes, click **Continue**.

    To update the changes in a plan, you must generate and run the plan again.

Deleting a Policy

When you delete a policy, you do not delete the data domains or rules that you added to the policy. When you delete a policy, you remove all the assignments that you create with the policy.

1. To access the **Policies** view, click **Policies**.

2. Click to open the policy you want to delete.
3. Click **Actions** > **Delete**.

   The **Delete Policies** dialog box appears.
4. Confirm that you want to delete the policy.

   If the policy contains assignments, the **Impacted Objects** dialog box appears with the list of affected columns and plans.
5. To download the list of affected objects, click **Export**, and save the .csv file.
6. To delete the policy, click **Continue**.

   Test Data Manager deletes the policy. To update the changes in a plan, generate and run the plan again.

CHAPTER 5

Data Discovery

This chapter includes the following topics:

- Data Discovery Overview, 56
- Data Discovery Sources, 57
- Discover View, 57
- Column Properties, 58
- Data Discovery Task Flow, 61
- Data Domain Discovery, 61
- Profile Management, 64
- Profile Import, 65
- Apply the Results, 66
- Project Tables, 68

Data Discovery Overview

Run profiles to discover source data for data masking and data subset operations.

Before you run data masking, you can discover which columns to mask with the same masking rules. Before you create a data subset, you can discover relationships between tables. You can apply profiling results instead of manually applying a data masking rule to one column at a time.

You can run the following type of profile:

Data domain

Identifies the columns that belong in a data domain based on the data value of the column or the column name. Use the results when you apply data masking rules. You can apply a rule to all columns that belong to the data domain instead of manually applying the rule to one column at a time.

You can also import and run profiles that you create in Informatica Developer.

Data Discovery Sources

You can run profiles for data discovery on relational sources.

You can run profiles for data discovery on the following sources:

- Oracle
- Microsoft SQL Server
- Sybase

Use ODBC to connect to Sybase source.

You can run data domain profiles on Hive and HDFS sources.

Rules and Guidelines for Data Discovery Sources

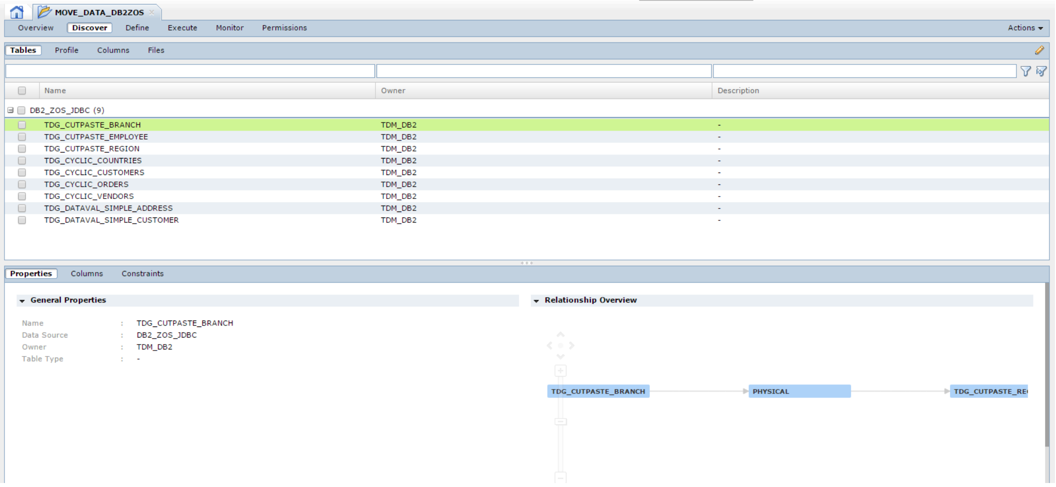
Use the following rules and guidelines when you configure the data discovery sources:

- You cannot run a profile for data discovery on nonrelational sources. However, you can use profiling in Informatica Data Quality to create and run profiles and then export the results. You can import the results to the TDM repository with Test Data Manager. For information on how to run profiles in Informatica Data Quality and export them, see the Informatica *Profile Guide*.
- You can import the results of profiles that use specific options in Informatica Data Quality.
  - You can view the results of profiles that use the Enterprise Discovery Profile and Profile options.
  - To view the profile results for Hadoop, you import the profile results from the Developer client with the Enterprise Discovery Profile option.
  - You can import the results of data discovery performed with mapplets created using simple regular expressions. You cannot import results if the mapplets use labeler, tokenizer, content set, or reference tables.
  - You can import data domains from the Model repository into TDM. TDM does not import the data pattern, metadata pattern, and preferred rules.
  - You can import and view domain discovery results of profiles run in the project. You can import the results of profiles created in folders within the project, but you cannot view the results in TDM.
- A table that you import to the repository must have the same connection when you use it in a profile. If you use a different connection for the profile than you used to import the data source, you might receive unexpected results.
- You cannot run a profile that contains two tables with the same name. For example, a project might have more than one EMPLOYEE table. Each EMPLOYEE table might have a different owner. You must create a separate profile for each EMPLOYEE table.

Discover View

View tables, configure profiles, and update columns for masking in the **Discover** view.

The following image shows the **Discover** view in a Project:



□	Name	Owner	Description
□	DB2_ZOS_JDBC (9)		
□	TDG_CUTPASTE_BRANCH	TDM_DB2	-
□	TDG_CUTPASTE_EMPLOYEE	TDM_DB2	-
□	TDG_CUTPASTE_REGION	TDM_DB2	-
□	TDG_CYCLIC_COUNTRIES	TDM_DB2	-
□	TDG_CYCLIC_CUSTOMERS	TDM_DB2	-
□	TDG_CYCLIC_ORDERS	TDM_DB2	-
□	TDG_CYCLIC_VENDORS	TDM_DB2	-
□	TDG_DATAVAL_SIMPLE_ADDRESS	TDM_DB2	-
□	TDG_DATAVAL_SIMPLE_CUSTOMER	TDM_DB2	-

Name : TDG_CUTPASTE_BRANCH
Data Source : DB2_ZOS_JDBC
Owner : TDM_DB2
Table Type : -

The **Discover** view contains the following views:

Tables

View the tables in the data source. Select a table and view the relationship between the table and other tables in the project. View the columns in a table. Define primary key and foreign keys in tables. You can delete a table from the metadata.

You can disable physical primary key columns when you do not need the composite primary keys. In the **Tables** | **Columns** view, you select a row with a physical primary key and enable or disable the primary key from the **Actions** menu. You can also edit the precision of a source table column based on the values that you need in the target column.

Profile

Define a profile for data discovery. Configure the profile to find columns for data domains.

Columns

Maintain data domain assignments. Mark columns as restricted, or sensitive, assign bulk data domains, and define similar column values for value cascades.

Column Properties

You can configure column properties for a project source.

The following table describes the properties listed on the **Discover** | **Columns** view:

Property	Description
Table Name	Name of the table.
Column	Name of the column.
Owner	Name of the database schema.
Data Type	Column data type.

Property	Description
Domain	Name of the data domain that the column belongs to.
Restricted	The column in the database is a restricted column. The Data Integration Service does not write the value of a restricted column to a target.
Computed	Indicates whether a column contains data that is computed from other columns. You cannot assign rules to a computed column.
Sensitive	Indicates whether a column contains sensitive data.
Similar Value Columns	Shows the number of columns that are related to this column in a value cascade.

Sensitive Columns

When you mark a column as sensitive, you indicate that the column contains sensitive data. Sensitive columns are for reference only and they do not affect processing.

When you assign a data domain to a column, Test Data Manager marks the column as sensitive.

Computed Columns

Computed columns are columns that contain data determined by data in other columns.

For example, an *Employee_Salary* table contains columns *Emp_ID*, *Basic_Salary*, *Allowances*, *Tax_Deduction*, and *Net_Pay*.

Net_Pay is a computed column that is calculated as `Basic_Salary + Allowances - Tax_Deduction`. The database populates the data in *Net_Pay*. If the values in any of the three columns change, the database updates the value in the *Net_Pay* column.

You can include tables that contain computed columns in a TDM operation. Because the values are calculated based on data in other columns, the TDM operation does not copy data into computed columns in the target. The database populates the values based on the data in other columns.

You cannot apply data masking rules on computed columns. You cannot create a value cascade or an auto cascade on computed columns.

Value Cascades

Use a value cascade to mask similar columns in different tables with the same masking rules. Use a value cascade when the columns that you want to mask are not related by key constraints. Data masking returns the same values into each column. Configure a group of similar value columns to define a value cascade.

Configure value cascades as similar value columns and define one of the columns as the driving column. If you assign a rule to the driving column, Test Data Manager assigns the rule to all the columns in the group. For example, you configure employee_num column in one table and EmpID in another table as similar value columns. Define employee_num as the driving table. When you assign a data masking rule to employee_num, EmpID receives the same masking rule. The columns in a value cascade must be the same type.

When you view the columns in a project, a driving table shows a number in the **Similar Value Columns** column. The number indicates how many other columns are in the value cascade. The similar value columns do not show a number in the **Similar Value Columns** column.

You can view a list of the columns in a value cascade. Select the driving column in the content panel. Click the **Similar Value Columns** view in the details panel. A list of the similar value columns appears.

Creating a Value Cascade

Configure a value cascade as a group of similar value columns and define one of the columns as the driving column.

1. In the **Discover | Columns** view, select multiple columns that you need to define as similar columns.
2. Click **Actions** > **Set as similar value columns**.

   A list of the columns with similar values appears.
3. Choose one column as the driving column.

   To select the column, click the driving table column checkbox to change the value from No to Yes.

Changing a Value Cascade

You can change a value cascade configuration. You can change the driving column, delete a column, or add similar value columns.

1. In the **Discover | Columns** view, select the driving table of the value cascade you want to change.

   A driving table has a numeric value in the **Similar Value Columns** column.
2. Click the **Similar Value Columns** view in the **Details** panel.

   A list of the columns in the value cascade appears.
3. Click **Edit Similar Value Columns**.

   The **Edit Similar Value Columns** dialog box appears.
4. To change the driving column to a different column, change the **Driving Column** value to Yes for the column that you want.

   You do not have to disable the previous driving table column. The value changes to No by default.
5. To add similar value columns to the group, click **Add**.

   Select the columns from the **Add Similar Value Columns** dialog box.
6. To remove columns from the group, select the columns on the **Edit Similar Values** dialog box and then click **Delete**.

Deleting a Value Cascade Group

You can delete a value cascade.

1. In the **Discover | Columns** view, select the driving table for the value cascade.

   A driving table has a numeric value in the **Similar Value Columns** column.
2. Click the **Similar Value Columns** view in the **Details** panel.

   A list of the similar value columns in the value cascade appears.
3. Select any column in the group and click **Delete.**
4. Confirm that you want to delete the similar values group.

   Test Data Manager removes the value cascade.

Auto Cascades

An auto cascade masks the primary key in a parent table and related foreign keys in child tables with the same value.

Tables with key constraints have auto cascades applied by default. You can manually disable or enable auto cascades. For example, you need to mask the employee ID in an employee master table and two child tables. The employee ID is a primary key in the master table. When you mask the employee ID in the employee table, the employee ID in the child tables receives the same masked value.

To use auto cascades, key constraints must exist between the parent and the child tables.

Data Discovery Task Flow

You can run a data domain profile to search for columns to assign to data domains for data masking.

Before you can run profiles, the administrator must configure a connection to the source database for data discovery. The administrator must also configure connections to the Data Integration Service and the Model Repository Service.

Complete the following high-level steps to perform data discovery:

1. Create a profile.
2. Select the data domain discovery profile.
3. If you choose to run a data domain discovery profile, choose the data domains to search for.
4. Choose the sampling size for the profile.
5. Run the profile and monitor the job.
6. After the job completes, open the profile again.
7. Review the data domain profile results.
8. Select and approve the results that you want to use for data masking and data subset operations.

Data Domain Discovery

Data domain discovery finds the source columns that contain similar data. The profile assigns the same data domain name to each column that contains similar data. You can assign the same data masking rules to all the columns in a data domain at the same time.

Create a data domain to describe the columns you need to mask with the same data masking rules. When you create a data domain, you configure regular expressions that define patterns in the data or patterns in the column names.

Run the data domain discovery profile to find the columns that match the criteria in the data domain regular expressions. When you configure a profile for data domain discovery, select the tables to search in the data domain discovery operation. Select which data domains to search for in the tables. You can select policies that contain data domains instead of selecting each data domain to search for.

After you run the profile for data discovery you can view the profile results. The profile results assign source columns to data domains. You can choose which profile results to use for data masking.

Data Domain Profiling on Hive and HDFS Sources

You can run data domain profiles on Hive and HDFS data sources to identify sensitive data.

You can run a data domain profile on Hive data sources.

You can run a profile on an HDFS source from the Developer tool. You can then import the profile results to the TDM repository with Test Data Manager. After you import the profile results, you must run the profiles to view the data domain profile results in Test Data Manager.

Data Domain Profile Sampling Options

When you run a profile for data domain discovery, configure sampling options to limit the number of rows to search or limit the regular expressions to search with.

The following table describes the sampling options that you can select in a profile for data discovery:

Option	Description
Data	Search for patterns in the data only.
Column Name	Search for patterns in the column name only.
Data and Column Name	Search for patterns in the data and in the column name.
Maximum Rows to Profile	Limit the number of rows to profile. Default is 1000.
Minimum Conformance Percent	Minimum percentage of rows where the column data or metadata matches the data domain.

Assigning a Data Domain to Multiple Columns

You can manually assign a data domain to multiple columns at a time. You can also remove the data domain assignment from multiple columns at a time.

1. Open a project.
2. Navigate to the **Discover** | **Columns** view.

   A list of all the columns in the project appears.
3. Select the columns that you want to assign the data domain to.
4. Click **Actions** > **Edit Assignments**.

   The **Edit Data Domain Assignment** dialog box appears.
5. Choose the data domain to assign to the columns.

   You can choose a blank data domain to remove the previous data domain assignment.
6. Click **Save**.

   The data domain assignments appear in the **Columns** view.

Manually Updating the Column Data Domain

You can manually update a data domain for a column. When you add a data domain to a column, Test Data Manager marks the column as a sensitive column.

1. Open the project and click **Discover** | **Columns**.
2. Click the **Domain** column for the column you want to update.
   A list of data domains appears.
3. Select the data domain to add to the column.

Note: You can run a profile for data domain discovery to update the data domain for columns.

Data Domain Assignments Import and Export

You can import a data domain assignments file into Test Data Manager and export data domain assignments from Test Data Manager.

You can export a data domain assignments file from Secure@Source and import the data domain assignments file into Test Data Manager.

Import a data domain assignments file in CSV format and view the data domain assignments in Test Data Manager. If a data domain assignment with the same name exists, TDM overwrites the existing data domain assignment with the imported data domain assignment. Any error during import is logged in the log file.

When you import a data domain assignments file, the records from the file are imported sequentially. If TDM fails to import a record, an error is logged and the import resumes from the next record. You can view the errors in the **Administrator** | **Application Logs** tab.

The CSV file that you import must contain columns in the following order:

- Data Source Name
- Owner Name
- Table Name
- Column Name
- Is Sensitive
- Domain Name

TDM considers the first row of the file as the header and does not import the first row. If the CSV file does not contain a header, you lose the first row of data.

Export data domain assignments as CSV files. You can import the CSV file into Test Data Manager.

To import or export a data domain assignments file, you must have read and write permissions on the project and Discover Project privilege.

Exporting a Data Domain Assignments File

1. Open the required project and click the **Discover** | **Columns** view.
2. Click **Actions** > **Export Data Domain Assignments**.
3. If you configure the browser to prompt for a download location, you must enter a file path and file name.

   Click **Save** to save the file.

Importing a Data Domain Assignments File

1. Open the required project and click the **Discover | Columns** view.
2. Click **Actions** > **Import Data Domain Assignments**.

   The **Import a Data Domain Assignments File** dialog box appears.
3. Select the CSV file that you want to import and click **Finish**.

   The imported data domain column assignments appears in the **Discover | Columns** view.

Profile Management

You can create and run profiles, add primary and unique keys to the source, and verify and add constraints to the source. Alternatively, you can import profiles that you create and run in Informatica Developer.

You can run a profile multiple times, and you can edit a profile between runs.

Creating a Data Domain Profile

Create and run data domain profiles in the **Discover** view.

A project must contain policies before you create a data domain profile. The policies contain the data domains that you can use in a profile for data discovery.

1. Open the project and click the **Discover** view.
2. Click the **Profile** view.

   The **Profile** view shows a list of the profiles in the project.
3. Click **Actions** > **New Profile** to create a new profile.
4. In the **New Profile** dialog box, enter the profile name and description. Choose to create a data domain profile.
5. Select the tables to profile and click **OK**.
6. Click **Next.**
7. In the **Select Sampling Options** pane, choose whether to add policies or data domains to the profile. When you select a policy, Test Data Manager includes all data domains in the policy.

   Test Data Manager returns a list of policies or data domains in the pane.
8. Select the policies or the data domains to profile.
9. In the **Sampling** panel, select whether to run data discovery on the source data, the column name, or the data and the column name.

   You can run a profile for column metadata and then run it again for the source data.
10. Enter the maximum number of rows to profile.
11. Enter the minimum conformance percent.

    All rows might not conform to the data domain expression pattern. You can enter a minimum percentage of the profiled rows that must conform.
12. Click **Save.**
13. Click **Actions** > **Execute**.

Editing a Profile

You can edit a profile and run it multiple times. When you run the profile, the Data Integration Service overwrites the original profile results.

When you edit a profile, you can change the profile name and description. You can add or remove tables. When you run data domain profiles, you can change the policies and data domains in the profile.

1. Open the project and click the **Discover** | **Profile** view.
2. In the **Discover** view, click a profile name to open it.
3. Click **Edit**.

   The **Edit Profile** dialog box appears. The dialog box has the **General** tab, the **Tables** tab, and the **Sampling** tab depending on the profile type.
4. On the **General** tab, you can edit the name and description of the profile.
5. On the **Tables** tab, add or remove tables that you want to profile.

   To add a table, click **Add** and select the tables you want to add. To delete a table, select the check box for the table you want to delete. Click **Delete**.
6. On the **Sampling** tab, you can change the data domains or policies in the profile. You can choose to run the profile against the source data, the column name, or both data and column name.
7. Click **Actions** > **Execute** to run the profile.

Deleting a Profile

You can delete profiles.

1. Open the project and click the **Discover** view.
2. In the **Discover** pane, select the profile row, but do not open the profile.
3. Click **Actions** > **Delete** to delete the profile.
4. Click **Yes** to confirm the delete.

Profile Import

You can import a profile from the Model repository and view the profile in Test Data Manager.

For example, you can run a profile on a flat file source in the Developer tool. You can import the profile into TDM. You can then use the profile information to apply masking rules to mask source data or to create a data subset in TDM.

Import the profile from within a project.

Before you import a profile, import the metadata and the objects in the profile into the TDM repository. For example, before you import a data domain profile, the TDM repository must contain all the data domains that appear in the profile. You can import the data domains from the Model repository. You cannot view the profile after import if data domains with the same name do not exist in the TDM repository.

You can import profiles with table names that contain alphanumeric characters or the special character _.

Importing a Profile

You can import profiles from the Model repository into TDM. You can then view the profile in Test Data Manager.

Verify that the source metadata and the data domains are in the TDM repository before you import the profile. Import the metadata into the project from a connection by the same name as the connection that you used to run the profile

If you delete the metadata from the project after you import a profile, the profile does not display results. If you import the metadata again, and import the profile again with a different name, the results appear in the first profile as well.

1. Open the project and click the **Discover | Profile** view.
2. Click **Actions** > **Import Profile**.

   The **Import Profiles** dialog box appears.
3. Select the profile that you want to import from the Model repository. Click **Next**.

   You cannot import a profile that contains a table name with special characters other than _.
4. Optional. TDM imports the profile with the same name and description. Edit the profile name and description if required.
5. Optional. Choose to import the curation information with the profile.

   When you import the curation information, TDM imports the saved review information and performs tasks based on the review information. For example, TDM assigns a data domain to a column if the inferred data domain is approved in a data domain profile.
6. Select the Data Domain profile type.
7. Click **Finish**.

   The imported profile appears in the **Discover** view.

Apply the Results

After the profile run, you must close the profile in Test Data Manager and open it again to view the profile results.

Data Domain Discovery Results

The data domain profile results show a list of source columns and possible data domains to assign to the columns. You can select which data domain candidates to use for data masking from the profile results.

To view data domain discovery results, close the profile and open it again. Click the **Profile | Data Domain** view.

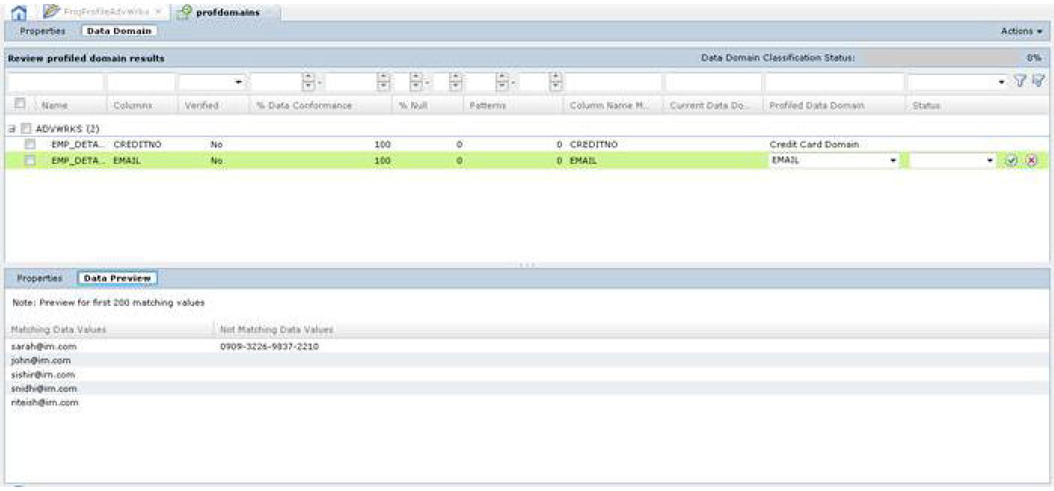
Select a column and click on the **Data Preview** tab to view the source data of the selected column. The data viewer displays the first 200 records of columns returned in the data domain profile.

You can select rows and approve the data domain for each column. When you approve the suggested data domain for the column, you can assign the rules in the domain to each column in the data domain. You can assign the rules in the **Define | Data Masking** view.

After you are finished working on the data domain for a column, you can verify the data domain for each column in the **Data Domain** view. The **Verify** column is for tracking. It does not affect the data domain profile operation.

When you finish approving the data domains, you can mark the data domain classification as completed. Use this method to verify that you reviewed all the results. Completing the data domain classification does not affect any process.

The following image shows the data domain discovery results:



Review profiled domain results

Data Domain Classification Status: 0%

	Name	Columns	Verified	% Data Conformance	% Null	Patterns	Column Name M...	Current Data Do...	Profiled Data Domain	Status
□	ADVWRKS (2)									
□	EMP_DETA...	CREDITNO	No	100	0	0	CREDITNO		Credit Card Domain	
□	EMP_DETA...	EMAIL	No	100	0	0	EMAIL		EMAIL	

Note: Preview for first 200 matching values

Matching Data Values	Not Matching Data Values
sarah@im.com	0909-3226-9837-2210
john@im.com	
sishir@im.com	
snidhi@im.com	
niteish@im.com	

The following table describes the columns in the data discovery results:

Column Name	Description
Table	Name of the table.
Source	Name of the column to mask.
% Data Conformance	The percentage of rows that contain metadata or data patterns that match the data domain.
% Null	The number of rows that contain NULL values in the source column.
Patterns	The number of data domain data or metadata patterns that match the column.
Column Name Match	Indicates if the column name is the matching data domain pattern.
Current Data Domain	Shows the previous data domain when you run the data domain profile more than once.
Profiled data domain	The data domain name returned from the latest profile run.
Status	Shows whether the data domain is verified or approved.

Project Tables

You can view the source tables in a project. You can view the primary keys and foreign keys in each table.

Click the **Discover | Tables** view to see the data source tables in the project. You can filter the list of tables by the table name or by the description.

Click the table name to view the table properties in the **General Properties** pane. View the columns in the table in the **Columns** view. Test Data Manager also shows the child tables in the **Relationship Overview** pane.

Table Classification

You can assign a classification to source tables to identify what kind of data the table contains. The classification does not affect profile operations. You can filter tables by this classification when you create plans.

In the **Discover | Tables** view, select the tables that you want to assign a classification to. Choose one of the following table classifications:

- Temporary
- Master
- Configuration
- Transactional
- Log
- Seed

For example, you might want to configure more filter criteria for tables that contain transactions or logs.

Constraints

Foreign key constraints define parent-child relationships between the source tables. Use constraints to determine the tables to include in a data subset. You can also limit the values that you want to store in the data subset table columns.

Use data discovery to find relationships between tables. When you identify the relationships that you want to add to the TDM repository, create and edit constraints in Test Data Manager.

The following types of constraints define relationships between tables:

Primary Key

A column or combination of columns that uniquely identifies a row in a data source. A table can have one primary key.

Logical Manual

A parent-child relationship between tables based on columns that are not keys. You can create the following logical constraints:

- Logical constraints that you accept from data discovery profiles. You can delete these logical constraints.
- Logical constraints that you define in Test Data Manager. You can delete these logical constraints.

Manually Add Keys to Tables in a Project

You can manually add primary keys and foreign keys to tables to establish relationships between tables for data subset operations.

When you add keys, you define constraints for data subset, and data masking operations in the project. You do not update the source database.

You can add the following types of keys constraints:

Primary Key

You can add one column or column combination as the primary key for a table. A primary key column cannot contain null or duplicate values. You cannot add more than one primary key constraint.

Foreign Key

Add a column or column combination as a foreign key in a table. When you define a constraint, you define a foreign key in a table and relate it to a column in a parent table.

Unique Key

Add a column or set of columns as a constraint to define a unique key in a table. A unique key column can contain null values. You can create more than one unique key in table.

Creating a Primary Key Constraint

You can add a logical primary key to a table to create table relationships for data subset.

1. In a project, click the **Discover | Tables** view.
2. Click a table name to select the table.
3. Click **Constraints**.
4. Click **Create New Constraint**.

   The **New Constraint** dialog box appears.
5. Select **Primary Key**.
6. Click **Next**.
7. To add the columns, click **Add**. Select a column for which you want to add a primary key constraint.
8. Click **Finish**.

Creating a Logical Relationship Between Tables

You can add a logical foreign key to a table to create a table relationship for data subset. Choose a column from a parent table to establish a key relationship between the tables.

1. In a project, click **Discover** > **Tables**.
2. Click the table in which you want to create the foreign key.
3. Click the **Constraints** tab.
4. Click **Create New Constraint**.
5. Enter the constraint properties.

The following table describes the constraint properties:

Property	Description
Name	Constraint identifier.
Constraint Type	Select the constraint type as Foreign Key.
Parent Table	Choose the parent table to establish the foreign key with.
Enable Constraint	Enable the foreign key relationship.

6. Click **Next**.

   A list of the columns in the table appears in the left panel. A list of the columns in the parent table appears in the right pane.
7. Click a child column from the left pane. Click a parent column from the right pane. Click the **Link** icon to map the parent-child relationship.
8. Click **Finish**.

Creating a Unique Key Constraint

You can add a unique key to a table to create table relationships for data subset.

1. In a project, click the **Discover | Tables** view.
2. Click a table name to select the table.
3. Click **Constraints**.
4. Click **Create New Constraint**.

   The **New Constraint** dialog box appears.
5. Select **Unique Key**.
6. Click **Next**.
7. To add the columns, click **Add**. Select the column for which you want to add a unique key constraint.
8. Click **Finish**.

CHAPTER 6

Creating a Data Subset

This chapter includes the following topics:

- Data Subset Overview, 71
- Data Subset Process Flow, 71
- Data Subset Components, 72
- Creating a Group, 73
- Editing a Data Subset Component, 73
- Exporting a Data Subset Component, 73
- Importing a Data Subset Component, 74
- Copying a Data Subset Component, 74
- Deleting a Data Subset Component, 74
- Creating a Data Subset, 75

Data Subset Overview

You can create a subset of production data if you need a small, targeted, and referentially intact copy of the production data to use in a nonproduction environment. A nonproduction environment might include development, test, or training environments.

For example, you might create a subset of financial data for a specific region or time period.

You can create a data subset from relational or flat file sources.

To create a data subset, you define data subset components and add them to a plan. The components that you create depend on the type of data source and the data subset that you require.

You can generate a workflow from the plan. The workflow loads a subset of the source data into a target database.

Data Subset Process Flow

Create a data subset to create a subset of production data that is referentially intact.

You can perform the following tasks related to data subset in Test Data Manager:

Create data subset components

You can create groups. Create a group to add one or more specific tables to a data subset.

Edit data subset components

You can edit groups that you create or import and use the edited groups to create a data subset.

Export or import data subset components

You can export groups that you create in Test Data Manager. You can import the groups and use them to create a data subset with a different Test Data Manager Service.

Run a plan to create a data subset

You can add relational or flat file sources to a plan to create a data subset based on subset components that you add to the plan.

Data Subset Components

You can create a data subset from relational and flat file sources.

To create a data subset, you define the following component:

Group

Defines a set of unrelated tables. Create a group when you need to copy data from one or more unrelated tables to a subset database. Applicable for relational and flat file sources.

Groups

A group defines one or more unrelated tables that you want to copy to a subset database. Create a group to add unrelated tables to a plan, or to copy unfiltered data to a target.

When you add tables to a group, you can include residual tables in the group. A residual table is a table that has not been added to a group. Select residual tables when you want to include all the tables in a source in a data subset plan.

When you edit a group, you can add or remove tables from the group.

You create, edit, and delete groups in the **Define** view of an application.

Group Example

The test data contains multiple tables with organization information and employee information.

Some of the tables contain foreign keys and are related tables. Some of the tables that contain employee information are not related to tables that contain organization information. You want to create a data subset that contains employee skill information and organization location information. To ensure that you include all tables with data that you require in the data subset, create a group and add the tables that contain the data.

Create and run a plan that includes the group. The data subset that you create contains tables with the data that you require.

Creating a Group

To create a group, select the tables that you want to add to a data subset plan.

1. On the **Define** | **Data Subset** view in the project, click **Actions** > **New** > **Groups**.
2. In the **Create Group** dialog box, enter a name for the group and optional description for the group. Click **Next**.
3. To select one or more tables for the group, click **Add Tables**.
4. Optionally, you can filter the list of tables to search for.
5. Select the required tables and click **OK**.
6. Click **Save**.

Editing a Data Subset Component

You can edit a group in a project. Edit a group to change the general properties and tables in the group.

1. In a project, click **Define** | **Data Subset**.

   The **Data Subset** tab shows a list of the groups in the project.
2. Click the required group name to open it.
3. Click **Actions** > **Edit**.

   The **Edit** dialog box appears.
4. On the **General** tab, edit the name, description, and status.
5. To edit a group, on the **Tables** tab, change the tables in the group.
6. Click **Save**.

   A list of plans that include the group appears. To update the changes in the plans, generate and run the plans again.
7. Optional. To download the list of plans in a .csv file, click **Export**.

   The file contains a list of the plans with the description and the project information.
8. Click **Continue**.

Exporting a Data Subset Component

You can export a group to an XML file and import the XML file to another TDM repository.

1. Click **Projects** to open the **Projects** view.

   A list of projects appears.
2. Open the project that contains the group to export.
3. Click the **Define** view.

   The list of groups in the project appears.
4. Select a group to export.

Use the checkbox to select the required component.

5. Click **Actions** > **Export**.
6. Choose the name and path of the XML file to create.

   The default name is a string that contains "<component type>_" and the current date and the time.
7. Click **OK**.

Importing a Data Subset Component

You can import a group from an XML file that was exported from another TDM repository.

1. To open the **Projects** view, click **Projects**.
2. Click the project description to select a project to import the group into.

   Do not open the project.
3. Click **Actions** > **Import**.
4. Browse for the XML file that contains the group to import.

   The XML file has a default name with a string that contains "<component type>_" and the date and time of export.
5. To import the group, click **Finish**.

Copying a Data Subset Component

You can create a data subset component by copying a data subset component. Test Data Manager copies the tables in the original group into the new group.

1. Open a project and click **Define** > **Data Subset** to view a list of subset components in the project.
2. Click the description to select the required subset component.

   Do not open the group.
3. Click **Actions** > **Duplicate**.

   The **Duplicate <component name>** dialog box appears.
4. Change the name and description. Click **Save**.

Deleting a Data Subset Component

If you do not use a subset component, you can delete the component. If you delete a subset component that is assigned to a plan, the plan is not valid.

1. In the **Define | Data Subset** view in the project, click to select the required data subset components.

   Do not open the subset component.
2. Click the **Actions** > **Delete**.

3. In the **Delete Objects** message box, click **OK**.

   A list of plans that the subset components are included in appears. To update the changes in the plans, generate and run the plans again.

4. Optional. To download the list of plans in a .csv file, click **Export**.

   The file contains a list of the plans with the description and the project information.

5. Click **Continue**.

Creating a Data Subset

Create a data subset to create a referentially intact subset of production data.

1. Create a project and add the required data sources to the project.
2. Perform the following tasks based on the type of data source:
   - Relational database. Create the required data subset components for relational sources.
   - Flat files. Create the required data subset components for relational sources.
3. Create a plan and add the required data subset components and data sources to the plan.
4. Run the workflow.
5. Monitor the progress of the workflow.

Related Topics:

- “Creating a Project” on page 33
- “Importing Data Sources” on page 39
- “Plan Settings” on page 127
- “Creating a Data Masking and Data Subset Plan” on page 135
- “Workflow Generation” on page 137
- “Executing a Workflow” on page 139
- “Viewing the Log Messages” on page 145

CHAPTER 7

Performing a Data Masking Operation

This chapter includes the following topics:

- Data Masking Overview, 76
- Data Masking Task Flow, 77
- Data Masking Rules, 77
- Creating and Assigning Data Masking Rules, 79
- Modifying Data Masking Rules and Assignments, 84
- Performing a Data Masking Operation, 88
- Data Masking Components, 88

Data Masking Overview

Use data masking to replace source data in sensitive columns with realistic test data for nonproduction environments. When you create data masking rules, you define the logic to replace sensitive data. To configure the sensitive columns that you want to mask, assign data masking rules to source columns, data domains, and policies.

A policy defines the data masking rules, the data to mask, and the masking parameters for a source. When you assign data masking rules to policies, you can assign multiple source columns to the data masking rules. You can also assign a rule directly to a source column. You can assign data masking rules based on the data type of the source columns.

To implement data masking, create a data masking plan and generate a workflow from the plan. A data masking plan can contain policies and rules. In a data masking plan, select a rule that is assigned to a column. Policies and rules define how to mask sensitive and confidential data in a target database. A data masking plan contains at least one rule or one policy.

When you start the workflow, the Data Integration Service performs the masking operation.

Data Masking Task Flow

To implement data masking operations, assign masking rules to columns in a source. Create a plan and add policies and rules to the plan. Generate a workflow from the plan to mask data in a target database.

You can perform the following tasks related to data masking in Test Data Manager:

Create data masking rules

You can create a data masking rule and add the masking rule to a project. You can also add a masking rule to a data domain and add the data domain to a policy. Add the masking rule to a data domain, and add the data domain to policies.

Edit data masking rules

You can edit a data masking rule to change the rule parameters.

Assign data masking rules to columns

Use the masking rules within a project to assign to target columns. You can assign a masking rule or a policy to a target column.

Run a plan to mask sensitive data

Create a plan and add data masking components. To mask sensitive data, you must generate and run the plan.

Data Masking Rules

A data masking rule is a data masking technique to mask a specific type of data. You can create a standard rule, advanced rule, or a rule that you import as a mapplet.

A data masking technique defines the logic that masks the data. Masking parameters are options that you configure for a masking technique. For example, you might blur output results by different percentages for different columns. Most masking techniques have associated masking parameters.

You can enable users to override masking parameters for a rule. For example, you create a rule with the substitution masking technique to mask column data based on a flat file substitution source. You set the override option for the rule. When a developer assigns this rule to columns in a source, the developer can select a relational database as a substitution source rather than a flat file.

You can assign rules to source columns, data domains, policies, and plans.

Standard Masking Rules

A standard masking rule is a data masking rule that applies a built-in masking technique. A standard masking rule has one input column and one output column.

When you create a standard masking rule, you select the masking technique from a list. You can define one masking technique in a standard masking rule and you can apply the rule to one column.

Test Data Manager has masking techniques that you can select to create masking rules. You can use standard masking techniques based on the source datatype and masking type that you configure for a column. You can restrict the characters in a string to replace and the characters to apply in the mask. When you mask numbers and dates, you can provide a range of numbers for the masked data. You can configure a range that is a fixed or percentage variance from the original number.

Rule Simulation

You can simulate the output of a standard rule to preview the output before you assign the rule to a column.

Use the Rule Simulator to view the output of a standard rule before you assign it to a column or add it to a plan. View the output of the rule and change the rule properties if required before assigning it to a column. You can choose to include data from a connection in the simulation. Alternatively, you can use default sample data or enter up to 100 rows of sample data on which to simulate the rule output. View the original data values and the masked values in the Rule Simulator tab.

The latest simulation configuration details are stored in the browser cache. You can edit the properties of a rule after viewing the simulation results, and run the simulation again on the same data with the updated rule. Clearing the cache deletes the configuration information.

Mapplet Rules

You can create rules from a mapplet. The mapplet contains the logic to mask the input columns and return data to the target columns. When you create a rule from a mapplet, you assign the mapplet column names to input and output columns when you assign the rule to a column in the data source.

Import a mapplet from an XML file that you exported from the Model repository. The mapplet can contain any passive transformations.

To mask Hadoop data, you can import the mapplets that you create in the Developer tool. You cannot use sequence generator, lookup, and classifier transformations when you import mapplet to mask Hadoop data.

A mapplet can contain multiple input and multiple output columns. All columns might not be available in all projects. You must configure one input column and one output column as required columns. The mandatory columns must have source and target assignments when you assign the rule to a column in the data source. Test Data Manager has an interface to assign multiple columns to a rule from a mapplet.

The TDM repository stores the mapplet logic when you import the mapplet. You cannot change the mapplet in Test Data Manager.

Advanced Masking Rules

An advanced masking rule is a combination of masking techniques that mask multiple source columns or a target column based on values of more than one input column.

For example, you can create a full masked name by masking the first name and last name input columns. Define variable columns to contain the masked names. Add an output column that contains a result of an expression that combines the first name and last name variable columns.

Create the following types of columns in an advanced rule:

Input

The source column that you want to mask.

Variable

A column that contains intermediate values in a calculation. The variable column receives a value from an expression or a masking technique. You can configure multiple variable columns in order to combine multiple masking techniques.

Output

The target column that receives the masked value. The output column type contains a masking technique and masking parameters.

Related Topics:

- "Advanced Masking" on page 95
- "Creating a Data Masking and Data Subset Plan" on page 135
- "Generating a Workflow" on page 138

Masking Rule Assignments

You can assign a data masking rule to a column in the **Define | Data Masking** view. Choose a rule from a list in the **Masking Rule** column in the view. The data domain default rule appears at the top of the list when you click the **Masking Rule** column.

The following table describes the fields in the **Data Masking** view:

Column	Description
Name	Name of the table.
Owner	Name of the database schema.
Columns	Name of the column to mask.
Data Type	Data type of the column to mask.
Domain	Name of the domain that you assigned to the column either from a data domain discovery or a manual assignment.
Sensitive	Indicates if the column is a sensitive column. Value is Yes or No.
Similar Value Columns	Indicates that the column is configured with other columns in a cascade. The column shows the number of other columns in the cascade.
Policy	Policy name that the column is assigned to.
Masking Rule	The rules to apply to the column. When you click inside the Masking Rule column, you can choose which rule to apply to the column. A preferred rule from a data domain has an asterisk (*) before the name.
Override	Shows the override property status for a rule. If the property is Yes, you can override the rule properties when you assign the rule to a column. When you override the rule parameters for a column the **Override** column value is Yes-Overridden.

Creating and Assigning Data Masking Rules

Create and assign data masking rules to perform data masking operations. Update the masking rule properties, and change the masking rule assignments.

The **Policies** view shows the masking rules in the TDM repository. After you create a masking rule, you can edit and delete a masking rule from the **Policies** view.

Creating a Standard Masking Rule

Create a rule to define a masking technique, the datatype to mask, and masking parameters that define how to apply the technique.

1. To access the **Policies** view, click **Policies**.
2. Click **Actions** > **New** > **Masking Rule**.

   The **Rule Wizard** appears.
3. Enter a name and optional description for the rule.
4. Select the datatype of the column to apply the masking rule to.
5. Select the Standard masking rule.
6. To enable users to override masking parameters for a rule, select the **Override Allowed** option.
7. Click **Next**.

   Note: The **Masking Parameters** dialog box changes based on the **Masking Technique** you select.
8. Enter the masking parameters.
9. Enter the exception handling options. Configure how to handle null or empty spaces. Configure whether to continue processing on error.
10. Click **Finish**.

Previewing Rule Output

Use the Rule Simulator to view the output of a standard masking rule on selected data. You can use data from a connection, use default sample data, or enter sample data on which to view the rule output.

1. To access the **Policies** view, click **Policies**.
2. Click a masking rule name to open the **Rule Properties** page of the masking rule.
3. Click **Rule Simulator** to open the **Rule Simulator** configuration tab.
4. Select the type of configuration from the Source Details list and click **Edit**. Select Default to use default sample data, My Test Data to enter sample data, or Connection to use data from a connection. The **Configure Source Details** page opens.
5. Optional. To use source data from a connection, on the **Configure Source Details** page:
   a. Open the **Connection** tab.
   b. Select the connection and owner from the lists.
   c. Click **Browse** to select the table from a list.
   d. Select the required column from the list of columns.

      By default, up to 10 random columns appear in the list. You can enter the name of a column that you require if it does not appear in the list.
   e. Enter the number of rows to include in the simulation. The default value is 20. The maximum number of rows you can include is 100.
   f. Click **OK**.
6. Optional. To enter sample data:
   a. Enter the data in the data fields on the **Sample Data** tab. Use the buttons to add or delete rows. You can enter a maximum of 100 rows.
   b. Click **OK**.

7. Optional. To use default sample data:
   a. Click the **Copy default data** button on the **Sample Data** tab.
   b. Click **OK**.
8. On the **Rule Simulator** tab, click **Go** to start the simulation.

   The original source values and the masked values appear on the **Rule Simulator** tab.

Creating a Mapplet Masking Rule

You can create a data masking rule from a mapplet. The mapplet contains the logic to mask the source fields.

Export the mapplet to an XML file from the Model repository before you import it to Test Data Manager.

1. To access the **Policies** view, click **Policies**.
2. Click **Actions** > **New** > **Masking Rule**.

   The **Rule Wizard** appears.
3. Enter a name and optional description for the rule.
4. Choose the data type to mask.
5. Select the Mapplet masking type.
6. To mask Hadoop data, select **Hadoop** and browse to the XML file that contains the Informatica Developer mapplet to import.
7. Click **Next**.
8. Select at least one input column and output column as mandatory columns.

   Select the column and click the mandatory column to change the value from `No` to `Yes`.
9. Click **Finish**.

Creating an Advanced Masking Rule

Create an advanced masking rule to combine more than one masking technique or to mask multiple columns.

1. To access the **Policies** view, click **Policies**.
2. Click **Actions** > **New** > **Masking Rule**.

   The **Rule Wizard** appears.
3. Enter a name and optional description for the rule.
4. Select the Advanced masking rule.
5. Click **Next**.
6. In the **Input Columns** section, click **Add New**.

   The **Add Column** dialog box appears.
7. Enter the column properties, such as the name, datatype, precision, and scale. Select whether the column is mandatory to assign to a data source in all projects.

   You cannot enter masking properties for input columns.
8. Click **OK**.
9. To enter more input columns, click **Create Input Column** in the **New Masking Rule** dialog box.
10. In the **Variable Columns** section, click **Add New**.

11. Enter an expression, a masking rule, or a dependent column for each variable column that you define. If you apply a masking rule, configure the input column to create the variable column from.
12. Click **OK**.
13. In the **Output Columns** section, click **Add New**.
14. Enter an expression, a masking rule, or a dependent column for the output column.
15. Click **OK**.

Adding Data Masking Rules to a Project

After you create the data masking rules, you must add the rules to a project. Use the masking rules within the project to mask target columns.

1. Click **Projects**.

   You can see a list of projects.
2. Open the project in which you want to use the rule.

   The project window opens in another tab.
3. Click **Overview** > **Policies**.
4. Click **Actions** > **Add Additional Rules**.

   The **Add Additional Rules** dialog box appears.
5. Select the masking rules that you created.
6. Click **OK**.

   The masking rules appear under the **Additional Rules** list.

Assigning a Standard Masking Rule

You can assign a standard masking rule to a target column. Assign masking rules from data domains or policies to one or more columns in the project source that you want to mask.

1. In the project, click **Define | Data Masking** to access the **Data Masking** view.
2. Select a column to assign a masking rule to.
3. If the **Domain** is blank for the column, click the **Policy** column and choose a policy that contains the data masking rule that you want to assign.
4. Click inside the **Rule** column to view the list of available rules.

   The data domain preferred rules appear at the top of the list. The other rules in the policy appear at the bottom of the list.
5. Select a masking rule. If you choose substitution masking rule, you can specify the rule assignment parameters. If you choose mapplet or advanced rule, you can assign the rule columns to the columns in the project.
6. Click **Save** for each column that you update.
7. If you want to update a masking rule assignment, select another masking rule.

   If the masking rule contains assignments, the **Impacted Objects** dialog box appears with the list of affected plans, columns, and data domains.
8. To download the list of affected columns and plans, click **Export**, and save the .csv file.
9. To save the changes, click **Continue**.

   To update the changes in a plan, you must generate and run the plan again.

Assigning a Custom Masking Rule

Mapplet rules can require values from multiple ports. You must map each source input column to a rule input and output port that you configure in the mapplet rule.

1. In the **Define | Data Masking** view, click the **Masking Rule** column for the column that requires a mapplet rule.
2. Select the mapplet rule from the list.

   The **Custom Rule Assignment** dialog box appears with the list of columns in the table.
3. Select a source input column in the left panel and a rule input port in the right panel.

   When you select a source input column, the rule ports with same data type become available for mapping.
4. Click the **Link** icon to create a mapping.

   An arrow appears that links the source port to the rule port.

   Note: To create a mapping, you can also click **Show Simple** and map the ports.
5. If you need to remove the link, select the source input port and the rule input port. Click the **Unlink** icon.

   The arrow between the ports disappears.
6. After you map the rule input ports, click **Next**.
7. Select a rule output port in the left panel and a source input column in the right panel.
8. Click the **Link** icon to create a mapping. To remove a mapping, click the **Unlink** icon.
9. Click **Save**.

   Note: If the source database is Sybase and the target database is Hadoop, you must limit the custom rule assignment to a maximum of 20 columns. The workflow generation fails if you assign a custom rule to more than 20 columns.

Assigning an Advanced Masking Rule

Assign an advanced masking rule to a column and map source input columns to the rule input and output ports. To perform expression cascade, you can add another table, select columns from the table, and map the input columns. An expression cascade joins the two tables to generate a combined masked output.

1. In the **Define | Data Masking** view, click the **Masking Rule** column for the column that requires an advanced rule.
2. Select the advanced rule from the list.

   The **Advanced Rule Assignment** dialog box appears with the list of columns in the table.
3. To select columns from another table, click the **Add Associated Table** icon.

   The **Add Associated Table** dialog box appears.
4. Select an associated table and click **OK**.

   The table name appears with a list of column.
5. Select a source input column in the left panel and a rule input port in the right panel.

   Note: When you select a source input column, the rule ports with same data type become available for mapping.
6. Click the **Link** icon to create a mapping.

   An arrow appears that links the source port to the rule port.

   Note: To create a mapping, you can also click **Show Simple** and map the ports.

7. If you need to remove the link, select the source input port and the rule input port. Click the **Unlink** icon.

   The arrow between the ports disappears.
8. After you map the rule input ports, click **Next**.
9. Provide a join condition for the two source input tables to perform expression cascade.
10. Click **Next**.
11. Select a rule output port in the left panel and a source input column in the right panel.
12. Click the **Link** icon to create a mapping. To remove a mapping, click the **Unlink** icon.
13. Click **Save**.

Modifying Data Masking Rules and Assignments

You can update or delete masking rules after you assign the rules to the target columns.

When you update or delete a masking rule, a warning message appears that contains the list of columns, plans, and data domains that contain the rule assignment. You can export the list of affected objects and save them in a .csv file.

If you update a masking rule, you change the properties within the masking rule. The changes do not take effect in the plans that contain the rules. You must fix the changes in the plan, and generate and run the plan again. If you make changes in the name or type of the field of a custom masking rule, the assignments do not contain the changes. You must import the mapplet again.

If you delete masking rules, TDM does not delete the masking rule from the columns, data domains, and plans that contain the rule assignment. To update the changes in a plan, you must generate and run the plan again.

Available Data Masking Rules

The **Data Masking** view lists the rules that you can assign to each column of the data source. You can assign one or more rules for a column if the masking rules are available for the column.

If no rules appear for a column when you click **Masking Rule**, check for the following situations:

- The project has no rules for the column datatype.
- The project does not contain policies with masking rules.
- A data domain that has a preferred rule that you need to use is not in a policy that you assigned to the project.

The Data Integration Service does not mask integers greater than 28 characters.

If a column has a data domain assignment, the preferred rules for the data domain appear at the top of the list in the **Masking Rule** column. If you forget to add a rule to a data domain, you can assign rules from policy preferred rules. The policy preferred rules appear below the data domain rules in the rules list.

You can apply the default data domain rules to multiple columns at a time. Select multiple columns and click **Rule Assignment**. You can choose to assign the default rules in the **Rule Assignment** dialog box.

You can apply a single masking rule and a policy to multiple columns of similar data type. Select the rule from the **Single Rule Assignment** dialog box. For example, you can search for an ID column, select all the ID columns, and assign a single rule, a policy, or both to the selected columns.

When you select mapplet rules, advanced rules, or any rule that uses multiple columns, you must configure a mapping between the source ports and the rule ports. The Test Data Manager notifies you when you need to configure the source to rule port mapping.

Note: In the **Data Masking** view, you must click the **Save** icon after each time you assign a rule. If you do not save the rule assignment before you assign another rule, the Test Data Manager discards the rule assignment.

Editing a Masking Rule

You can edit a masking rule and update the masking rule properties that you want.

1. To access the **Policies** view, click **Policies**.
2. Click the name of the masking rule that you want to edit.

   The rule opens in a tab.
3. Click **Actions** > **Edit**.
4. Edit the parameters that you want to change and click **OK** to save the rule.

   If the masking rule has assignments, the **Impacted Objects** dialog box appears with the list of affected columns, plans, and data domains.
5. To download the list of affected columns and plans, click **Export**, and save the .csv file.
6. To save the changes, click **Continue** .

   To update the changes in a plan, you must generate and run the plan again.

Copying a Masking Rule

You can create a masking rule by copying a masking rule. When you copy a masking rule, Test Data Manager copies the masking rule properties from the original masking rule to the new masking rule.

1. To access the **Policies** view, click **Policies**.
2. Click a masking rule description to select the masking rule.

   Do not open the masking rule.
3. Click **Actions** > **Duplicate**.

   The **Copy <Masking Rule Name>** dialog box appears.
4. Change the name and description of the masking rule. Click **Save**.

Deleting a Masking Rule

You can delete a masking rule before or after you assign the masking rule to a column.

1. To access the Policies view, click **Policies**.
2. Click the masking rule that you want to delete.
3. Click **Actions** > **Delete**.

   The **Delete Masking Rule** dialog box appears with a warning message. If the masking rule has assignments, the **Impacted Objects** dialog box appears with the list of affected columns, plans, and data domains.
4. To delete the masking rule that do not have assignments, click **OK**.

5. To delete the masking rule that have assignments, click **Continue**. To download the list of affected objects, click **Export**, and save the .csv file.

   To update the changes in a plan, you must generate and run the plan again.

Overriding a Masking Rule

You can override rule parameters after you assign the masking rule to a column. The rule must have the override property enabled.

1. In the **Define | Data Masking** view, select a column that has a masking rule with the override property enabled.

   The **Override** column value is Yes.
2. Click **Actions** > **Override**.

   The **Edit <Rule Type>** dialog box appears and shows the rule parameters.
3. Change the rule parameters and the exception parameters as required.
4. Click **Save**.

   Test Data Manager adds an Overridden flag to the **Override** column when you override the rule.

Assigning Single and Multiple Masking Rules

You can assign multiple masking rules to a column. To assign more than one rule to a column, create a duplicate row and update the rule. You can assign the default data domain rules to multiple columns at the same time. Assign a single masking rule or a policy or both to multiple columns of similar data type.

1. Open the **Define | Data Masking** view for a project.
2. Select a column that you want to assign the rule to. If you want to assign a rule to more than one column, you can select multiple columns.

   Use the checkbox to select the column.
3. Click **Rule Assignment**.

   The **Rule Assignments** dialog box appears.
4. Select the column that you want to assign multiple policies to.

   Use the checkbox to select the column.
5. Click **Copy Rule Assignment**.

   The Test Data Manager creates a row that is a duplicate of the row you selected.
6. Change the policy to view more preferred masking rules.
7. In the **Masking Rule** column, select a rule from the list.
8. Click **Save**.

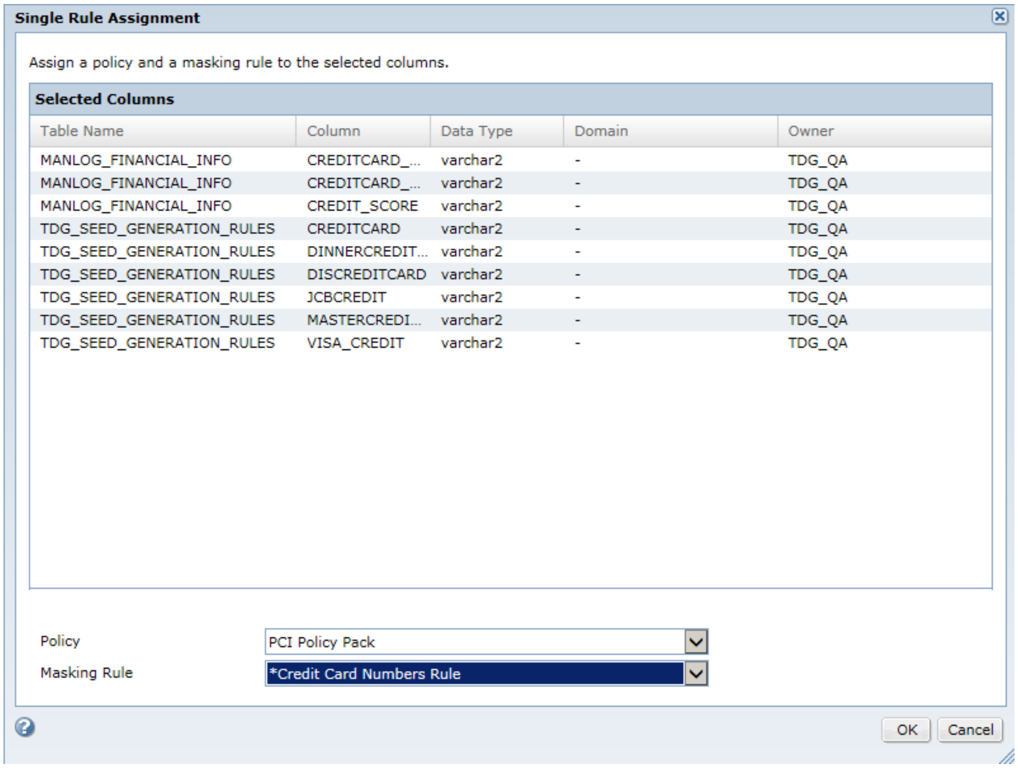
   The **Define | Data Masking** view shows two rows for the same column. Each column has a different rule assignment.
9. To assign the default rule to multiple columns, select which columns you want to update with the default values. You can select the checkbox at the top of the dialog box to select all rows.
10. Click **Default Assignments**.

    The Test Data Manager updates each column with the default rule.
11. Click **Save**.

12. To assign a single rule to multiple columns, search for the similar data type columns.
13. Select all the columns that you want to assign a masking rule to.
14. Click **Single Rule Assignment**.

    The **Single Rule Assignment** dialog box appears.
15. Select a policy and a masking rule.

    The following image shows the single masking rule assignment to multiple columns:



Single Rule Assignment

Assign a policy and a masking rule to the selected columns.

Selected Columns

Table Name	Column	Data Type	Domain	Owner
MANLOG_FINANCIAL_INFO	CREDITCARD_...	varchar2	-	TDG_QA
MANLOG_FINANCIAL_INFO	CREDITCARD_...	varchar2	-	TDG_QA
MANLOG_FINANCIAL_INFO	CREDIT_SCORE	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	CREDITCARD	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	DINNERCREDIT...	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	DISCREDITCARD	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	JCBCREDIT	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	MASTERCREDI...	varchar2	-	TDG_QA
TDG_SEED_GENERATION_RULES	VISA_CREDIT	varchar2	-	TDG_QA

Policy: PCI Policy Pack

Masking Rule: *Credit Card Numbers Rule

16. Click **OK**.

    The Test Data Manager updates each column with the assigned masking rule.

Deleting Masking Rule Assignments

You can delete a masking rule assignment to a column.

1. In the **Define | Data Masking** view, select the column that you need to delete the rule assignments for.
2. Click **Clear Assignment**.

   Test Data Manager prompts you to confirm deleting the assignment.
3. To delete the rule assignment from the column, click **OK**.

   If a plan contains the masking rule assignment, the **Impacted Objects** dialog box appears with the list of affected columns, plans, and data domains.
4. To delete the masking rule assignment, click **Continue**. To download the list of affected objects, click **Export**, and save the .csv file.

   To update the changes in a plan, you must generate and run the plan again.

Performing a Data Masking Operation

Perform a data masking operation in Test Data Manager to mask sensitive data.

1. Create a data masking rule.
2. Create a project and import metadata into the project.

   Create different projects to work with XSD sources and relational sources.
3. Add the data masking rule to the project.
4. Assign the masking rule to a target column. If the target is an XML file, apply the masking rule to an XML element or an attribute.
5. Create a plan, add the data masking rule, and configure the source and target connections.
6. Generate and run the workflow.
7. Monitor the progress of the workflow.

Related Topics:

- "Creating a Project" on page 33
- "Importing Data Sources" on page 39
- "Plan Settings" on page 127
- "Creating a Data Masking and Data Subset Plan" on page 135
- "Workflow Generation" on page 137
- "Executing a Workflow" on page 139
- "Viewing the Log Messages" on page 145

Data Masking Components

To perform data masking operations, assign rules to data domains, policies, and columns. Use data domains and data discovery to find columns that you want to mask. Create cascades to mask similar columns.

The following table describes the components that you create to implement data masking operations:

Component	Description
Assignments	The allocation of rules to a column to mask the column data. You assign a rule to a column through either a column assignment or a data domain assignment. A column assignment assigns a rule directly to a column in a source. A data domain assignment assigns one or more rules in a data domain to columns in a source.
Column sensitivity	A sensitive column contains sensitive data. Configure column sensitivity to mark columns that you want to mask.
Data domain	An object that represents the functional meaning of a column based on the column data or the column name. Use a data domain to filter the ports that you want to mask when you assign a rule to columns. Define patterns in the data or patterns in the column names when you configure a data domain.

Component	Description
Plan	Defines data masking operations. A data masking plan indicates whether to mask data in place in the source database or in stream in a target database.
Policy	Defines the data masking rules, the data to mask, and the masking parameters for a source.
Rule	Defines the data masking technique, an optional rule qualifier, and masking parameters. A masking technique defines the logic that is used to mask the data. Masking parameters define how a masking technique in a rule masks source data. You can set an override option in a rule that defines whether users can modify the masking parameters for the rule when they assign the rule to columns in a source.
Value cascade	Masks similar columns across tables. You can identify similar columns in a project and configure them to cascade masking rules. Use cascades when some fields are denormalized across multiple tables.

CHAPTER 8

Data Masking Techniques and Parameters

This chapter includes the following topics:

- Data Masking Techniques and Parameters Overview, 91
- Data Masking Techniques, 91
- Data Masking Parameters, 92
- Custom Masking, 94
- Advanced Masking, 95
- Credit Card Masking, 97
- Email Masking, 98
- Encryption Masking, 100
- Expression Masking, 100
- IP Address Masking, 101
- Key Masking, 102
- Nullification Masking, 104
- Phone Masking, 105
- Random Masking, 105
- Shuffle Masking, 109
- SIN Masking, 110
- SSN Masking, 111
- Substitution Masking, 111
- URL Masking, 112
- Name Substitution Example , 113
- Shuffle Address Example, 118

Data Masking Techniques and Parameters Overview

A data masking technique is the type of data masking to apply to a selected column. The masking parameters are the options that you configure for the technique.

The type of masking technique that you can apply depends on the datatype of the column that you need to mask. When you choose a masking technique, Test Data Manager displays parameters for the masking technique.

You can restrict the characters in a string to replace and the characters to apply in the mask. You can provide a range of numbers to mask numbers and dates. You can configure a range that is a fixed or percentage variance from the original number.

You can configure different masking parameters for a masking technique and save each configuration as a masking rule. The Data Integration Service modifies source data based on masking rules that you assign to each column. You can maintain data relationships in the masked data and maintain referential integrity between database tables.

Data Masking Techniques

You can apply masking techniques based on the source datatype that you configure for a column. For example, if the column datatype is numeric, you can define a masked value that is within a fixed or percent variance from the original value.

The Data Integration Service does not mask integers greater than 28 characters.

The following table describes the masking techniques that you can choose when you define a rule:

Masking Technique	Description
Advanced	Applies masking techniques to multiple input and output columns. You can create an expression to combine multiple columns. You can mask all datatypes.
Credit card	Applies a built-in mask format to disguise credit card numbers. You can mask the string datatype.
Email address	Applies a built-in mask format to disguise email addresses. You can mask the string datatype.
Encryption	Applies encryption to the source data. You can assign an encryption method to mask the data with. You can encrypt string data types.
Expression	Applies an expression to a column to create or disguise data. Use expression masking if the expression references one column. If the expression includes multiple columns, use the calculated masking technique with advanced masking. You can mask all datatypes.
IP address	Applies a built-in mask format to disguise IP addresses. You can mask the string datatype.
Key	Produces deterministic results for the same source data, masking rules, and seed value. You can mask date, numeric, and string datatypes.
Mapplet	Applies masking rules from a mapplet. The mapplet contains the logic to mask the input columns and return data to the target. A mapplet can have multiple input and output columns.

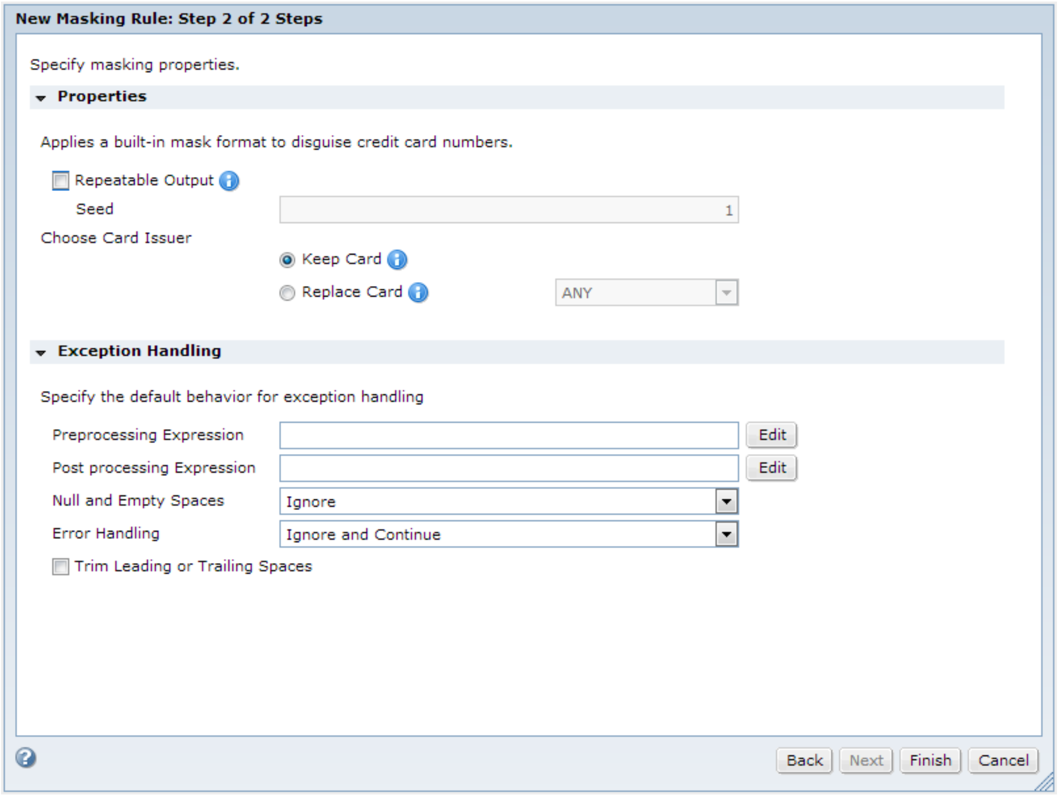
Masking Technique	Description
Nullification	Replaces a column of data with a null value. You can mask all datatypes.
Phone	Applies a built-in mask format to disguise phone numbers. You can mask the string datatype.
Random	Produces random, non-repeatable results for the same source data and masking rules. You can mask date, numeric, and string datatypes.
Shuffle	Applies sensitive column values from one row to another row in the same table. You can restrict which rows to shuffle data from. You can mask date, numeric, and string datatypes.
SIN	Applies a mask to Social Insurance numbers. You can mask a string datatype.
SSN	Applies a built-in mask format to disguise Social Security numbers. You can mask the string datatype.
Substitution	Replaces a column of data with similar but unrelated data from a dictionary. You can mask date, numeric, and string datatypes.
URL	Applies a built-in mask format to disguise URL data. You can mask a string datatype.

Data Masking Parameters

Configure data masking parameters to define how to apply a data masking technique. The parameters that you configure depend on the data type that you need to mask. Some masking techniques are not available for all data types.

You can configure the data masking properties and the exception handling parameters in the **New Masking Rule** wizard.

The following image shows data masking parameters that appear when you configure a Credit Card data masking rule:



New Masking Rule: Step 2 of 2 Steps

Specify masking properties.

Properties

Applies a built-in mask format to disguise credit card numbers.

□ Repeatable Output

Seed: 1

Choose Card Issuer

● Keep Card

○ Replace Card: ANY

Exception Handling

Specify the default behavior for exception handling

Preprocessing Expression:

Post processing Expression:

Null and Empty Spaces: Ignore

Error Handling: Ignore and Continue

□ Trim Leading or Trailing Spaces

Repeatable Output

Repeatable data masking output returns deterministic values. Use repeatable masking when you generate a data masking workflow more than once and you need to return the same masked values each time it runs.

Configure repeatable output if you have the same value in multiple source tables and you want to return the masked value in all of the target tables. The tables in the target database receive consistent masked values.

For example, customer John Smith has two account numbers, 1234 and 5678. The account numbers are in multiple tables. The Data Integration Service masks John Smith as Frank Martinez in all the tables. It always masks account number 1234 as 6549 and account number 5678 as 3214.

You can enter a seed value when you configure repeatable output. You can configure a dictionary file with replacement data values for substitution masking. When you configure repeatable output, TDM returns the same value from the dictionary whenever a specific value appears in the source data.

Seed

Apply a seed value to create repeatable output for data masking output. The seed value is a starting point for generating masked values.

You can define a seed value from 1 through 999. The default seed value is 1. Apply the same seed value to a column to return the same masked data values in different source data. For example, you have the same `Cust_ID` column in four tables. If you want all of them to output the same masked values, you can apply the same seed value when you mask each column.

You can create a variable for the seed value in a parameter file. When you configure repeatable output in a masking rule, enter the name of the variable instead of a seed value. You can then change the seed value in the parameter file if you need to change the seed value, instead of editing the masking rule.

The TDM administrator can also set a central seed value for all data masking components with repeatable output. The central seed value overrides any other seed value.

Exception Handling

Data masking exception handling defines options for handling nulls, blanks, empty strings, and errors in source data. You can configure exception handling for each data masking rule that you create. You can specify preprocessing and post processing expression parameters to apply changes before and after masking the data.

The following table describes the exception handling options:

Option	Description
Preprocessing Expression	Optional. Expression to define changes to make to the data before masking. Click **Edit** to configure the preprocessing expression.
Post processing Expression	Optional. Expression to define changes to make to the masked data before saving the data to the target. Click **Edit** to configure the post processing expression.
Null and Empty Spaces	The default behavior for handling null values or empty columns in the source data. Choose one of the following options: - Constant. Mask the data with a constant value. Enter the value to use. - Log error and continue. Log an error in the session log and continue processing. - Treat as value. Treat null values or spaces as a valid source value. Mask the space or null value with a valid value. - Ignore. Do not mask the null or empty space. **Note:** For columns with dependent masking, there are no separate rules to handle null and empty spaces. TDM masks the dependent column based on the rules and values of the column that it depends on.
Error Handling	The default behavior for handling errors in the source data. Choose one of the following options: - Constant. Mask the data with a constant value. Enter the value to use. - Log exception and continue. Log an exception in the session log and continue processing. - Ignore and continue. Do not mask the null or empty space. - Error. Log an error in the session log and stop processing.
Trim Leading or Trailing Spaces	Trims the leading and trailing spaces from source data. When you enable this option the following source fields are the same: " Jones", "Jones", "Jones " .

Custom Masking

Use custom masking when you want to apply multiple masking techniques to a column or when you want to configure multiple input and output columns in the same masking rule. You can mask all data types.

To create a custom masking rule, you must import the mapplets. Set the mapplet input columns and output columns as required. The mapplet must contain at least one input and one output column.

When you assign the custom masking rule to columns, you must map the source columns to the columns in the custom masking rule.

Custom Masking Parameters

Configure input and output columns that you import from a mapplet when you create a custom masking rule.

The following table describes the general properties that you can configure for input and output columns:

Parameter	Description
Column Name	The name of an input or an output column within a mapplet. When you assign the rule to a column, you map the column names in the rule to column names in the database.
Column Type	The column type. The mapplet can contain the following types of columns: - Input. Receives the input data. - Output. Returns the output data.
Data Type	The data type of the column.
Precision	The precision for the column. The maximum number of digits or the maximum number of characters that the column can accommodate. For example, 874.560 has a precision of 6.
Mandatory	Indicates if you must assign the column to a table column in the project. Applies to input and output columns. You must set at least one input and one output column as required.
Group Name	Specifies the group to which the columns belong. The group name can be input, output, or any other name that you provide when you create a mapplet.

Advanced Masking

Use advanced masking when you want to apply multiple masking techniques to a column or when you want to configure multiple input and output columns in the same masking rule. You can mask all datatypes.

When you configure advanced masking, you configure the input columns, output columns, and variable columns. Variable columns are work fields that you can define to temporarily store data.

When you create the columns in the masking rule, the column names do not need to be the same as the column names in the source. When you assign the masking rule to columns, you must map the source columns to columns in the advanced masking rule.

Related Topics:

- “Advanced Masking Rules” on page 78
- “Creating a Data Masking and Data Subset Plan” on page 135
- “Generating a Workflow” on page 138

Advanced Masking Parameters

Configure parameters for each column that you create in an advanced masking rule.

The following table describes the general properties that you can configure for input, output, and variable columns:

Parameter	Description
Column Name	The name of an input, output, or variable column. Enter any name. The name does not have to match the name of a column in the source. When you assign the rule to source data in a project, you map the column names in the rule to column names in the database.
Column Type	The column type. You can configure the following types of columns: - Input. Receives the source data. - Variable. A temporary column that contains intermediate values. You can apply masking rules to variable column values in order to mask data before returning data to output columns. - Output. Returns the output data. You can apply an expression or a masking rule to variable column data and return the data in the output column.
Datatype	The datatype of the column.
Precision	The precision for the column. The maximum number of digits or the maximum number of characters that the column can accommodate. For example, 798.650 has a precision of 6.
Scale	Number of digits to the right of the decimal point in a number.
Mandatory	Indicates if you must assign the column to a column in the source. Applies to input and output columns.

The following table describes the masking properties that you can configure for variable columns and output columns:

Parameter	Description
Expression	An expression to apply to the variable column. You can create the expression in the Expression Builder.
Masking Rule	Applies a masking rule to the input column and writes the results in the variable column. You can enter the following parameters. - Condition. Defines whether an input column should be masked or not. If the condition is true, the Data Integration Service masks the column. - Rule. The data masking rule to apply to the input column. - Override properties. You can change the data masking rule parameters if the rule owner enabled the rule to be overridden. - Input column. The name of the input column to apply the masking rule to. Select an input column from the columns that you added to the rule.
Condition Input	Applies an expression to the Output column only. Select a condition from the list. If you applied conditions on the input column, you can use it for the output column.
Dependent	Applies dependent masking. Dependent masking replaces the value of a column based on the values returned from a dictionary row for another column. You must define substitution masking for another column before configuring a column for dependent masking. Enter the following parameter: - Input column. The name of the input column that you configured for substitution masking. - Dictionary column. Choose the dictionary column to replace the dependent column with.

Advanced Masking Example

You can create an expression to combine multiple columns in an advanced masking rule.

Create an expression in the **Expression Builder**. Select columns, functions, variables, and operators to build expressions. The expression can reference input columns and variable columns.

For example, you have a CUSTOMERS table that contains first and last names. You want to mask the names and combine the masked values into a masked full name.

The following table shows the columns that you create in an the advanced masking rule to combine the masked names:

Column Name	Column Type	Masking Technique	Expression
FIRST_NAME	INPUT	–	–
LAST_NAME	INPUT	–	–
FIRST_MASKED	VARIABLE	Substitution on FIRST_NAME	_
LAST_MASKED	VARIABLE	Substitution on LAST_NAME	_
FULL_NAME	OUTPUT	Expression	FIRST_MASKED || ' ' || LAST_MASKED

Mask the FIRST_NAME and LAST_NAME with Substitution. Return FULL_NAME using an expression that combines the FIRST_MASKED and LAST_MASKED columns.

When you create expressions in the Expression Builder, use the point-and-click interface to minimize errors. Verify that the expression returns a value that matches the output column data type.

Credit Card Masking

Credit card masking applies a built-in mask format to disguise credit card numbers. The Data Integration Service creates a masked number that has a valid checksum. You can choose from multiple credit card number formats. Mask the string datatype with credit card masking.

The Data Integration Service generates a logically valid credit card number when it masks a credit card number. The length of the source credit card number must be between 13 to 19 digits. The input credit card number must have a valid checksum based on credit card industry rules.

You can apply repeatable masking to credit card numbers. Each credit card type has a different format. You can choose to keep the same credit card type in the masked credit card number, or you can change the credit card type. For example, if the source credit card number is a Visa credit card number, you can configure the rule to return a Visa credit card number. Or, you can configure the rule to return a credit card number that is a different credit card type.

The source credit card number can contain numbers, spaces, and hyphens. If the credit card has incorrect characters or is the wrong length, the Integration Service writes an error to the session log. The Data Integration Service applies a default credit card number mask when the source data is not valid.

The Data Integration Service does not mask the six-digit Bank Identification Number (BIN) at the beginning of the number. For example, the Data Integration Service might mask the credit card number `4539 1596 8210 2773` as `4539 1516 0556 7067`.

Credit Card Masking Parameters

The following table describes the parameters that you can configure for credit card masking:

Parameter	Description
Repeatable	Returns the same masked value when you generate a workflow multiple times, or when you generate masked values for a column that is in multiple tables.
Seed	Starting point for creating repeatable output. Enter a number between 1 and 999. Default seed value is 1.
Keep card issuer	The Data Integration Service returns the same credit card type for the masked credit card. For example, if the source credit card is a Visa card, generate a masked credit card number that is the Visa format.
Replace card issuer	Replaces the source credit card type with another credit card type. When you enable replace card issuer, select which type of credit card to replace it with. You can choose credit cards such as AMEX, VISA, JCB, and MASTERCARD. Default is ANY.

Email Masking

Email masking generates realistic email addresses. You can apply a mask from the masked values of a first and last name. You can apply a constant domain name or create an expression to define the domain. Mask the string datatype with email masking.

You can configure email masking to mask email addresses the same way each time the same user appears in the source data. You can configure an email address from the masked first and last name columns. You can configure email masking as repeatable between workflows, or you can configure email masking to be repeatable in one workflow.

Note: The Data Integration Service always returns ASCII characters for an email address.

Email Masking Parameters

When you configure email masking, you can configure parameters to mask the user name and the domain name in the email address. You can specify dictionary files that contain the user names and the domain names. You can define an expression that combines names to create realistic email user names.

Email Masking General Parameters

The following table describes the parameters that define how to mask an email address:

Parameter	Description
Repeatable output	Returns the same masked value when you generate a workflow multiple times, or when you generate masked values for a column that is in multiple tables.
Seed	Starting point for creating repeatable output. Enter a number between 1 and 999. Default is 1.

Parameter	Description
Standard	Replaces an email string with characters. The standard email masking technique does not create realistic domain or user names.
Advanced	Replaces an email string with name and domain values from a dictionary or mapping.
Columns to form email	Indicates where to retrieve the columns to form the email address. Choose one of the following options: - From mapping. Do not use a dictionary for user names. Assign the column names when you assign the rule to source columns. - From dictionary. Mask user names from a dictionary file.

User Name Masking Parameters

The following table describes the parameters that define how to mask the user name in the email address:

Parameter	Description
Dictionary for columns	The dictionary to use when you choose to form an email from a dictionary column. Choose a dictionary that contains first and last names.
Expression	Defines an expression when you select to create an email from a mapping. You can create an expression to define the user name from parts of the first and last names. The column names in the expression do not have to match the source column names. Assign columns to the expression when you assign rules to columns.
First name column	The dictionary column that contains the first name of the email address.
First name length	Length for the first name. For example, you might choose the first letter of the name, or the first 4 letters of the name. Default is 10.
Delimiter	The delimiter between the first and last names of the email address. Choose a period, underscore, or hyphen.
Last name column	Dictionary column that contains the last name of the email address.
Last name length	The length for the last name. For example, you might choose the first 6 characters of the last name. Default is 10.

Domain Name Masking Parameters

The following table describes the parameters that define how to mask the domain name:

Parameter	Description
Constant	A constant value to mask the email address with. Each email address receives the same domain name.
Random	Indicates whether to use a flat file or relational dictionary.

Parameter	Description
Domain lookup dictionary	The dictionary file to use from the imported sources.
Domain name column	The column that contains a dictionary name in the domain lookup dictionary.

Encryption Masking

Encryption masking applies encryption algorithms to mask source data. You can choose the algorithm to encrypt the data. Mask string datatypes with encryption masking.

Select from the following encryption algorithms:

AES

Advanced Encryption Standard with 128-bit encoding.

CRC

Cyclic Redundancy Check. Finds data transmission errors or verifies that data is not modified. Computes a checksum.

MD5

MD5 Message-Digest Algorithm. One-way cryptographic hash function with a 128-bit hash value.

To configure encryption masking, enter an encryption key that is 16 characters or less.

Expression Masking

Expression masking applies an expression to a column to mask or change data. Mask all data types with expression masking.

To configure expression masking, you create an expression in the **Expression Builder**.

Select column, functions, variables, and operators to build expressions. The expression can reference input columns and output columns.

In expression masking, you can append additional data to the source column data. For example, you want to mask the first names. The source has the FirstName column and you want to concatenate the values in the FirstName column with the string ABC. When you configure the data masking rule, select the FirstName column and enter the following expression in the expression editor.

```
CONCAT(FirstName,'ABC')
```

When you configure expression masking for a column, the column name appears as the expression by default.

Select functions, columns, variables, and operators from the point-and-click interface to minimize errors when you build expressions.

When you create an expression, verify that the expression returns a value that matches the column data type. The Data Integration Service returns zero if the return value does not match a numeric column. It returns NULL if the return value does not match a string column.

You cannot perform expression masking with repeatable output for Hadoop data sources.

For information about expression syntax, see the *Informatica Transformation Language Reference*.

Expression Masking Parameters

You can configure parameters for the expression masking technique.

The following table describes the parameters that you can configure for expression masking:

Parameter	Description
Repeatable Output	Determines if the masked value should persist for a given column value. You must enter a seed value when you enable repeatable output. You cannot perform expression masking with repeatable output for Hadoop data sources.
Alphanumeric Seed	The alphanumeric seed is a key that allows multiple data masking rules to generate the same masked values from the same source values. Define the same seed in each data masking rule that requires the same results for a column. The seed can be any text.
Expression	Accepts expressions and performs a calculation based on values within a single row.

Rules and Guidelines for Expression Masking

Use the following rules and guidelines for expression masking:

- You cannot use the output from an expression as input to another expression. If you manually add the output column name to the expression, you might get unexpected results.
- Select functions, columns, variables, and operators from the point-and-click interface to minimize errors when you build expressions.
- If the data masking rule is configured for repeatable masking, specify the storage table in the data masking plan. If the storage table does not exist, the data masking operation fails.

IP Address Masking

IP address masking applies a built-in mask format to change IP addresses. Mask string datatypes with IP address masking.

Use IP address masking to mask data with the string datatype.

IP masking splits an IP address into four numbers, separated by periods. The first number is the network. The Data Integration Service masks the network number within the network range.

The Data Integration Service does not mask the class and private network address. The Data Integration Service masks a Class A IP address as a Class A IP Address and a 10.x.x.x address as a 10.x.x.x address.

For example, the Data Integration Service can mask `11.12.23.34` as `75.32.42.52`, and `10.23.24.32` as `10.61.74.84`.

Key Masking

Key masking produces deterministic results for the same source data, masking rules, and seed value. Mask date, numeric, and string datatypes with key masking.

The following table describes the parameters that you can configure for key masking:

Parameter	Description
Seed	A start number that enables the Data Integration Service to return deterministic data. You can mask the date, numeric, and string datatypes.
Mask Format	The type of character to substitute for each character in the input data. You can limit each character to an alphabetic, numeric, or alphanumeric character type. You can mask the string datatype.
Source String Characters	The characters in the source string that you want to mask. You can mask the string datatype.
Result String Replacement Characters	Substitutes the characters in the target string. You can mask the string datatype.

Mask Format

Configure a mask format to limit each character in the output column to an alphabetic, numeric, or alphanumeric character.

If you do not define a mask format, the Data Integration Service replaces each source character with any character. If the mask format is longer than the input string, the Data Integration Service ignores the extra characters in the mask format. If the mask format is shorter than the source string, the Data Integration Service does not mask the characters at the end of the source string.

Note: The mask format contains uppercase characters. When you enter a lowercase mask character, Test Data Manager converts the character to uppercase.

The following table describes mask format characters:

Character	Description
A	Alphabetical characters. For example, ASCII characters a to z and A to Z.
D	Digits. From 0 through 9.
N	Alphanumeric characters. For example, ASCII characters a to z, A to Z, and 0-9.
X	Any character. For example, alphanumeric or symbol.
+	No masking.
R	Remaining characters. R specifies that the remaining characters in the string can be any character type. R must appear as the last character of the mask.

Source String Characters

Configure source string characters to choose the characters that you want to mask.

For example, if you set the number sign (#) as a source string character, it is masked every time it occurs in the input data. The position of the characters in the source string does not matter, and you can configure any number of characters. If you do not configure source string characters, the masking replaces all the source characters in the column.

The source characters are case sensitive. The Data Integration Service does not always return unique data if the number of source string characters is fewer than the number of result string characters.

The following table describes the options that you can configure for source string characters:

Option	Description
Mask Only	Masks characters in the source that you configure as source string characters. For example, if you enter A and b as source string characters, every instance of A and b in the source data will change. A source character that is not an A or b will not change.
Mask all except	Masks all characters in the source except for source string characters. For example, if you enter "-" as the source string character, every character except for "-" will change.

Result String Replacement Characters

Configure result string replacement characters to specify masking output.

The Data Integration Service replaces characters in the source string with the result string replacement characters. For example, enter the following characters to configure each mask to contain uppercase alphabetic characters A through F:

```
ABCDEF
```

To avoid generating the same output for different input values, configure a wide range of substitute characters, or mask only a few source characters. The position of each character in the string does not matter.

The following table describes the options for result string replacement characters:

Option	Description
Use only	Masks the source with only the characters you define as result string replacement characters. For example, if you enter the characters A, B, and c, the masking replaces every character in the source column with an A, B, or c. The word "horse" might be replaced with `BAcBA`.
Use all except	Masks the source with any characters except the characters you define as result string replacement characters. For example, if you enter A, B, and c result string replacement characters, the masked data never has the characters A, B, or c.

Date Key Masking

You can configure key masking with dates to generate deterministic output.

You can change the seed to match the seed value for another column to return repeatable datetime values between the columns.

The Data Integration Service can mask dates between 1753 and 2400 with key masking. The Data Integration Service always generates valid dates. If the source year is in a leap year, the Data Integration Service returns a year that is also a leap year. If the source month contains 31 days, the Data Integration Service returns a month that has 31 days. If the source month is February, the Data Integration Service returns "February."

Numeric Key Masking Parameters

You can configure key masking for numeric values and generate deterministic output.

When you configure a column for numeric key masking, you can select a seed value for the column. When the Data Integration Service masks the source data, it applies a masking algorithm that requires the seed.

You can change the seed value for a column to produce repeatable results if the same source value occurs in a different column. Configure repeatable results when you want to maintain a primary key-foreign key relationship between two tables. In each rule, enter the same seed value for the primary-key column as the seed value for the foreign-key column. The Data Integration Service produces deterministic results for the same numeric values. The referential integrity is maintained between the tables.

String Key Masking Parameters

Configure string key masking to mask all or part of a string. To limit the masking output to certain characters, specify a mask format and result string replacement characters. If you need repeatable output, specify a seed value.

The following table describes the masking parameters that you can configure for key masking string values:

Parameter	Description
Seed	A start number that enables the Data Integration Service to return deterministic data. Select a seed value between 1 and 1,000. Apply the same seed value to a column to return the same masked data values in different source data.
Mask Format	The type of character to substitute for each character in the input data. You can limit each character to an alphabetic, numeric, or alphanumeric character type.
Source String Characters	The characters in the source string that you want to mask. For example, mask the number sign (#) character whenever it occurs in the input data. Leave this field blank to mask all the input characters. The Data Integration Service does not always return unique data if the number of source string characters is less than the number of result string characters.
Result String Replacement Characters	Substitutes the characters in the target string. For example, enter the following characters to configure each mask to contain uppercase alphabetic characters A through F: `ABCDEF`

Nullification Masking

Nullification masking replaces a column of data with a null value. Use nullification masking to mask binary, date, numeric, or string data.

Nullification masking has no parameters.

Phone Masking

Phone masking applies a built-in masking format to change phone number data. Mask string datatypes with phone masking.

Phone masking does not change the format of the original phone number. For example, phone masking can mask the phone number `(408)382 0658` as `(408)256 3106`.

The source data can contain numbers, spaces, hyphens, and parentheses. Phone masking does not mask alphabetic or special characters.

Random Masking

Random masking produces random, non-repeatable results for the same source data and masking rules.

Random masking does not require a seed value. The results of random masking are non-deterministic. Use random masking to mask date, numeric, and string data types.

The following table describes the options that you can configure for random masking:

Option	Description
Range	A range of output values. The Data Integration Service returns data between the minimum and maximum values. You can configure a range for date, numeric and string data types.
Blurring	A range of output values with a fixed or percent variance from the source data. Returns data that is close to the value of the source data. You can configure blurring for date and numeric data types.
Mask Format	The type of character to substitute for each character in the input data. You can limit each character to an alphabetic, numeric, or alphanumeric character type. You can configure a mask format for the string data type.
Source String Characters	The characters in the source string that you want to mask. You can configure source string characters for the string data type.
Result String Replacement Characters	Substitutes the characters in the target string. You can configure replacement characters for the string data type.

Range Masking

Configure a range to define an output range for numeric, date, or string data.

When you define a range for numeric or date values, the Data Integration Service masks the source data with a value between the minimum and maximum values. When you configure a range for a string, you configure a range of string lengths.

Note: When you configure date random masking, the maximum datetime must be later than the minimum datetime.

Blurring

Configure blurring to return a random value that is close to the original value. For random masking of datetime or numeric data, blurring creates an output value within a fixed or percent variance from the source data value.

Date Blurring

To blur a datetime source value, select a unit of time to blur, a high bound, and a low bound. You can select year, month, day, or hour as the unit of time. By default, the blur unit is year.

For example, to restrict the masked date to a date within two years of the source date, select year as the unit. Enter two as the low and high bound. If a source date is 02 February, 2006, the Data Integration Service returns a date between 02 February, 2004 and 02 February, 2008.

Numeric Blurring

To blur a numeric source value, select a fixed or percent variance, a high bound, and a low bound. The high and low bounds must be greater than or equal to zero.

The following table lists the masking results for blurring range values when the input source value is 66:

Blurring Type	Low	High	Result
Fixed	0	10	Between 66 and 76
Fixed	10	0	Between 56 and 66
Fixed	10	10	Between 56 and 76
Percent	0	50	Between 66 and 99
Percent	50	0	Between 33 and 66
Percent	50	50	Between 33 and 99

Mask Format

Configure a mask format to limit each character in the output column to an alphabetic, numeric, or alphanumeric character.

Note: The mask format contains uppercase characters. When you enter a lowercase mask character, Test Data Manager converts the character to uppercase.

The following table describes mask format characters:

Character	Description
A	Alphabetical characters. For example, ASCII characters a to z and A to Z.
D	Digits. From 0 through 9.
N	Alphanumeric characters. For example, ASCII characters a to z, A to Z, and 0-9.
X	Any character. For example, alphanumeric or symbol.

Character	Description
+	No masking.
R	Remaining characters. R specifies that the remaining characters in the string can be any character type. R must appear as the last character of the mask.

If you do not define a mask format, the Data Integration Service replaces each source character with any character. If the mask format is longer than the input string, the Data Integration Service ignores the extra characters in the mask format. If the mask format is shorter than the source string, the Data Integration Service does not mask the characters at the end of the source string.

Source String Characters

Source string characters are characters that you want to mask in the source. Configure source string characters if you want to mask a few of the characters in the input string.

For example, if you set the number sign (#) as a source string character, it is masked every time it occurs in the input data. The position of the characters in the source string does not matter, and you can configure any number of characters. If you do not configure source string characters, the masking replaces all the source characters in the column.

The source characters are case sensitive. The Data Integration Service does not always return unique data if the number of source string characters is fewer than the number of result string characters.

The following table describes the options that you can configure for source string characters:

Option	Description
Mask Only	Masks characters in the source that you configure as source string characters. For example, if you enter A and b as source string characters, every instance of A and b in the source data will change. A source character that is not an A or b will not change.
Mask all except	Masks all characters in the source except for source string characters. For example, if you enter "-" as the source string character, every character except for "-" will change.

Result String Replacement Characters

Result string replacement characters are a set of characters that the Data Integration Service can use to mask to the source data. You can configure the masking rule to mask the source only from the set of characters, or you can configure the masking rule to mask the source with any character except the result string replacement characters.

The Data Integration Service replaces characters in the source string with the result string replacement characters. For example, enter the following characters to configure each mask to contain uppercase alphabetic characters A through F:

```
ABCDEF
```

To avoid generating the same output for different input values, configure a wide range of substitute characters, or mask only a few source characters. The position of each character in the string does not matter.

The following table describes the options for result string replacement characters:

Option	Description
Use only	Masks the source with only the characters you define as result string replacement characters. For example, if you enter the characters A, B, and c, the masking replaces every character in the source column with an A, B, or c. The word "horse" might be replaced with `BAcBA`.
Use all except	Masks the source with any characters except the characters you define as result string replacement characters. For example, if you enter A, B, and c result string replacement characters, the masked data never has the characters A, B, or c.

Date Random Masking Parameters

To mask datetime values with random masking, either configure a range of output dates or choose a variance.

When you configure a variance, choose a part of the date to blur. Choose the year, month, day, hour, minute, or second. The Data Integration Service returns a date that is within the range you configure.

The following table describes the parameters that you can configure for random masking of datetime values:

Parameter	Description
Range	The minimum and maximum values to return for the selected datetime value. The date range is a fixed variance.
Blurring	Masks a date based on a variance that you apply to a unit of the date. The Data Integration Service returns a date that is within the variance. You can blur the year, month, day, or hour. Choose a low and high variance to apply.

Numeric Random Masking Parameters

When you mask numeric data, you can configure a range of output values for a column.

The Data Integration Service returns a value between the minimum and maximum values of the range depending on column precision. To define the range, configure the minimum and maximum ranges or a blurring range based on a variance from the original source value.

The following table describes the parameters that you can configure for random masking of numeric data:

Parameter	Description
Range	A range of output values. The Data Integration Service returns numeric data between the minimum and maximum values.
Blurring Range	A range of output values that are within a fixed variance or a percent variance of the source data. The Data Integration Service returns numeric data that is close to the value of the source data. You can configure a range and a blurring range.

String Random Masking Parameters

Configure random masking to generate random output for string columns.

To configure limitations for each character in the output string, configure a mask format. Configure filter characters to define which source characters to mask and the characters to mask them with.

The following table describes the parameters that you can configure for random masking of string columns:

Parameter	Description
Range	The minimum and maximum string length. The Data Integration Service returns a string of random characters between the minimum and maximum string length.
Mask Format	The type of character to substitute for each character in the input data. You can limit each character to an alphabetic, numeric, or alphanumeric character type.
Source String Characters	The characters in the source string that you want to mask.
Result String Replacement Characters	Substitutes the characters in the target string.

Shuffle Masking

Shuffle masking masks the data in a column with data from the same column in another row of the table. Shuffle masking switches all the values for a column in a file or database table. You can restrict which values to shuffle based on a lookup condition or a constraint. Mask date, numeric, and string data types with shuffle masking.

For example, you might want to switch the first name values from one customer to another customer in a table. The table includes the following rows:

```
100 Tom Bender
101 Sue Slade
102 Bob Bold
103 Eli Jones
```

When you apply shuffle masking, the rows contain the following data:

```
100 Bob Bender
101 Eli Slade
102 Tom Bold
103 Sue Jones
```

You can configure shuffle masking to shuffle data randomly or you can configure shuffle masking to return repeatable results.

For Hive and HDFS data sources, you can use shuffle masking only when the source is a relational database and the target is Hive or HDFS.

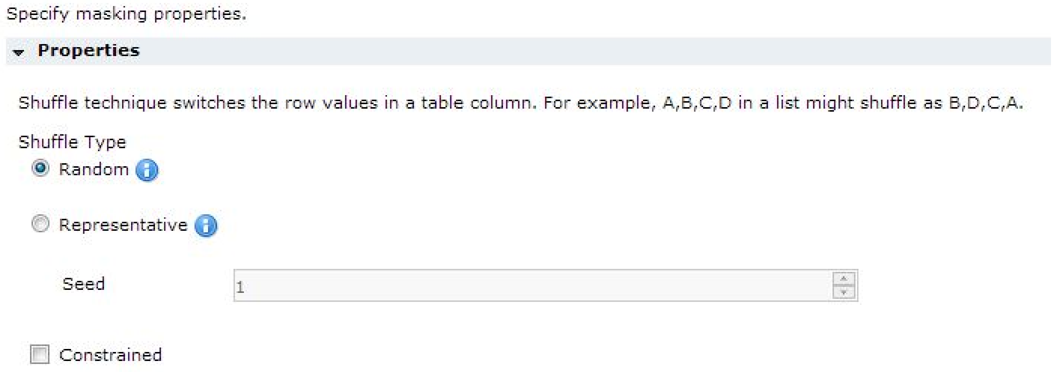
You cannot use shuffle masking when both the source and the target use Hadoop HDFS connections.

Note: If the source file might have empty strings in the shuffle column, set the **Null and Empty Spaces** option to Treat as Value in the rule exception handling. When you set the option to Treat as Value, the Data Integration Service masks the space or the null value with a valid value. The default is to skip masking the empty column.

Shuffle Masking Parameters

You can configure masking parameters to determine if shuffle masking is repeatable, the masking is repeatable for one workflow run, or the masking is random. You can also configure a lookup to ensure that replacement values originate from rows that contain specific values.

The following image shows Data Masking parameters that appear when you configure a Shuffle data masking rule:



Specify masking properties.

▾ Properties

Shuffle technique switches the row values in a table column. For example, A,B,C,D in a list might shuffle as B,D,C,A.

Shuffle Type

● Random

○ Representative

Seed: 1

□ Constrained

The following table describes the parameters that you can configure for shuffle masking:

Parameter	Description
Shuffle Type	Select random or representative shuffling: - Random. Shuffle values from one row to another without checking if the target values are unique for each source value. For example, the Integration Service masks 12345 with 65432 in a row. The Integration Service can also replace 33333 with 12345 in another row. - Representative. All source rows with the same value receive the same shuffle value. When the Integration Service replaces 12345 with 65432, then it can use 65432 as a mask value for any row with a 12345 source value. Representative masking does not save values between workflow runs. Use repeatable masking to return the same values between workflow runs.
Seed	Starting point for creating repeatable output. Enter a number between 1 and 999. Default is 1. Enabled when Representative Shuffle Type is selected.
Constrained	Restricts applying shuffle masking to rows that are constrained by another column. For example, shuffle employee names based on gender. Or, shuffle addresses within the same city. Choose the constraint column when you assign the rule to columns in a project.

SIN Masking

You can mask a Social Insurance number (SIN) that is nine digits. The digits can be delimited by any set of characters. Mask string datatypes with social insurance masking.

If the number contains no delimiters, the masked number contains no delimiters. Otherwise the masked number has the following format: `xxx-xxx-xxx`.

You can define the first digit of the masked SIN.

Enable **Start Digit** and enter the digit. The Data Integration Service creates masked Social Insurance numbers that start with the number that you enter.

You can configure repeatable masking for Social Insurance numbers. To configure repeatable masking for SIN numbers, click Repeatable Output and enter a Seed Value.

SSN Masking

SSN masking applies a built-in mask format to change Social Security numbers. Mask string datatypes with SSN masking.

The Data Integration Service generates valid Social Security numbers. To avoid generating numbers that the Social Security Administration has already issued, you can download the latest version of the High Group List at the following location:

```
http://www.ssa.gov/employer/highgroup.txt
```

The Data Integration Service accesses the latest High Group List from the following location:

```
<PowerCenter Installation Directory>\infa_shared\SrcFiles\highgroup.txt
```

The Data Integration Service generates Social Security numbers that are not on the High Group List.

The SSN masking accepts any SSN format that contains nine digits. You can delimit the digits with any set of characters. For example, the SSN masking accepts the following format:

```
+=54-*9944$#789-,*()”
```

You can configure repeatable masking for Social Security numbers. Select the **Repeatable Output** option, select the **Seed** option, and enter a value.

The Data Integration Service returns deterministic Social Security numbers with repeatable masking. The Data Integration Service cannot return all unique Social Security numbers because it cannot return valid Social Security numbers that the Social Security Administration has issued.

SSN Randomization

The Social Security Administration developed a method to randomize the nine-digit SSN. This method eliminates the geographical significance of the first three digits of SSN and protects the integrity of the Social Security numbers.

When you configure SSN masking parameters, you can use the SSN randomization technique to mask a Social Security number.

Substitution Masking

Substitution masking replaces a column of data with similar but unrelated data from a dictionary. Mask date, numeric, and string data types with substitution masking.

Use substitution masking to mask string data with realistic output. For example, if you want to mask address data, you specify a dictionary file that contains addresses. If you want to mask a Social Security number, you can specify the InvalidSSN dictionary file that contains Social Security numbers that are not valid.

Substitution is an effective way to replace production data with realistic test data. When you configure substitution masking, select the relational dictionary that contains the substitute values. The Data Integration Service performs a lookup on the dictionary and replaces source data with data from the dictionary. You can use relational dictionary to mask Hadoop data.

When you assign a substitution masking rule to a column, you can specify the rule assignment parameters.

The following table describes the rule assignment parameters that you can configure:

Parameter	Description
Lookup Condition	The column name in the source table you can refer to match with the column in the dictionary. This field is optional.

You can substitute data with repeatable or non-repeatable values. When you choose repeatable values, the Data Integration Service produces deterministic results for the same source data and seed value. You must configure a seed value to substitute data with deterministic results. The Data Integration Service maintains a storage table of source and masked values for repeatable masking. You can specify the storage table you want to use when you generate a workflow.

You cannot use flat file dictionaries and unique substitution masking to mask Hadoop data.

Substitution Masking Parameters

You can substitute data with repeatable or non-repeatable values.

When you choose repeatable values, the Data Integration Service produces deterministic results for the same source data and seed value. You must configure a seed value to substitute data with deterministic results.

You can configure the following substitution masking parameters:

Parameter	Description
Repeatable Output	Returns deterministic results between sessions. The Data Integration Service saves masked values in the storage table.
Seed	A start number that the Data Integration Service uses to return deterministic data.
Dictionary Information	Required. Configuration of the relational table that contains the substitute data values. Configure the following parameters: - Dictionary. Displays the relational table name that you select. - Masked Value. The column returned to the masking rule. - Lookup Column. The source data column to use in the lookup. - Serial Number Column. The column in the dictionary that contains the serial number.

URL Masking

URL masking applies a built-in mask format to change URL data. Mask string datatypes with URL masking.

The Data Integration Service parses a URL by searching for the `://` string and parsing the substring to the right of it. The source URL must contain the `://` string. The source URL can contain numbers and alphabetic characters.

The Data Integration Service does not mask the protocol of the URL. For example, if the URL is `http://www.yahoo.com`, the Data Integration Service can return `http://MgL.aHjCa.VsD/`. The Data Integration Service might generate a URL that is not valid.

Note: The Data Integration Service always returns ASCII characters for a URL.

Name Substitution Example

You want to mask employee names and you want to preserve the gender and nationality of the names in the masked data.

You create substitution masking rules to mask first names based on gender and nationality. Substitution masking replaces a column of data with similar but unrelated data from a dictionary. You use substitution masking to mask string data with realistic output. You use advanced masking to apply multiple masking techniques to a column.

Complete the following steps to create rules to mask names based on gender and nationality:

1. Add a dictionary in Test Data Manager.
2. Create a substitution rule that substitutes first names based on gender.
3. Create an advanced masking rule to substitute first names based on nationality.

Add a Dictionary in Test Data Manager

Add a dictionary in Test Data Manager to use for substitution masking.

The dictionary must contain country, gender, and first name columns. You can use a flat file or a relational dictionary. Test Data Manager uses the dictionary to substitute data.

The following text is a sample from the flat file dictionary that you use to mask the employee names:

```
SNO, COUNTRY, GENDER, FIRSTNAME
1, US, M, Adam
2, US, F, Sarah
3, JP, M, Mahito
4, JP, F, Kyoko
```

Adding a Relational Dictionary

When you add a relational dictionary, you 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.

Creating the Substitution Rule

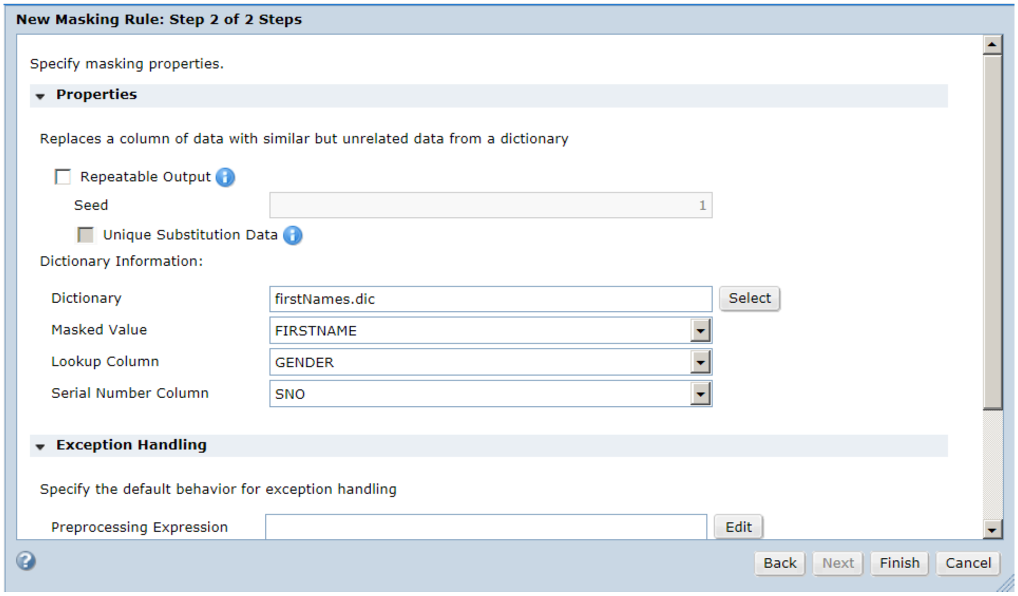
Create a substitution rule that substitutes first name based on gender.

1. In the **Policies** view, click **Actions** > **New** > **Masking Rule**.

   The **New Rule** window appears.

2. Enter a name and optional description for the rule.
3. Select the string datatype and standard substitution masking type. Select override allowed.
4. Click **Next**.
5. Enter the following substitution masking parameters:
   - Dictionary. Select the dictionary to use for the masking rule.
   - Masked Value. Select FIRSTNAME.
   - Lookup column. Select GENDER.
   - Serial number column. Select the serial number column.

   The following image shows the substitution masking parameters:



New Masking Rule: Step 2 of 2 Steps

Specify masking properties.

Properties

Replaces a column of data with similar but unrelated data from a dictionary

□ Repeatable Output

Seed: 1

□ Unique Substitution Data

Dictionary Information:

Dictionary: firstNames.dic

Masked Value: FIRSTNAME

Lookup Column: GENDER

Serial Number Column: SNO

Exception Handling

Specify the default behavior for exception handling

Preprocessing Expression:

6. Click **Finish**.

   The rule appears in the **Policies** view.

Creating the Advanced Masking Rule

Create an advanced rule that substitutes first names based on nationality with a lookup condition on gender.

1. In the **Policies** view, click **Actions** > **New** > **Masking Rule**.

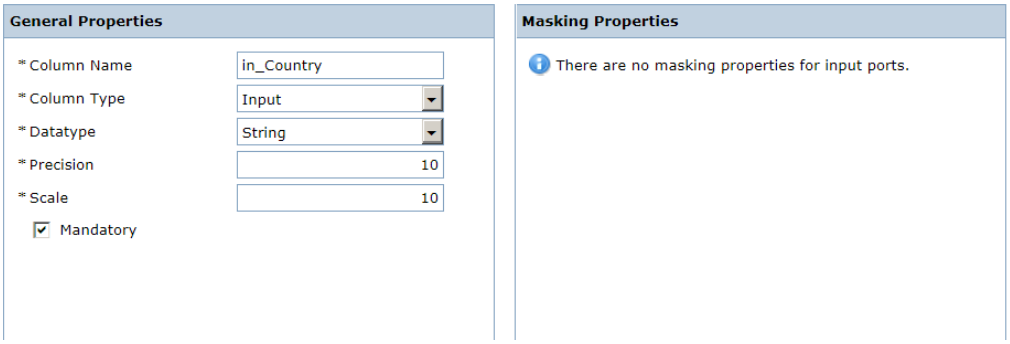
   The **New Masking Rule** window appears.

2. Enter a name and optional description for the rule.
3. Select the advanced masking type.
4. Click **Next**.
5. Click **Create Input Column** in the **Specify Masking Properties** window.

   The **Add Column** window appears.

6. Enter the following general properties:
   - Column name. Enter in_Country.
   - Column type. Select input.
   - Datatype. Select string.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.

   The following image shows the in_Country column properties:



General Properties

* Column Name: in_Country
* Column Type: Input
* Datatype: String
* Precision: 10
* Scale: 10

☑ Mandatory

Masking Properties

There are no masking properties for input ports.

7. Click **OK**.

   The in_Country input column appears in the list of input columns.
8. Click **Create Input Column**.

   The **Add Column** window appears.
9. Enter the following general properties:
   - Column name. Enter in_FirstName.
   - Column type. Select input.
   - Datatype. Select string.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.
10. Click **OK**.

    The in_FirstName input column appears in the list of input columns.
11. Click **Create Input Column**.

    The **Add Column** window appears.
12. Enter the following general properties:
    - Column name. Enter in_Gender.
    - Column type. Select input.
    - Datatype. Select string.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.
13. Click **OK**.

The in_Gender input column appears in the list of input columns.

14. Click **Create Input Column**.

    The **Add Column** window appears.

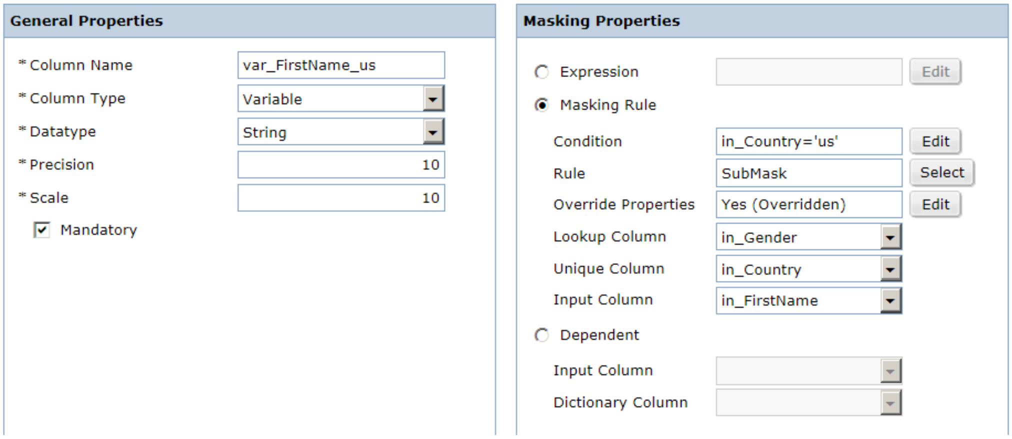
15. Enter the following general properties:

    - Column name. Enter var_FirstName_us.
    - Column type. Select variable.
    - Datatype. Select string.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.

16. Select the masking rule masking property. Configure the following masking properties:

    - Condition. Enter in_Country='us'.
    - Rule. Select the substitution rule you created in Step 2.
    - Override Properties. Click **Edit**. Enable **Repeatable Output** and **Unique Substitution Data** options, and click **Save**. The property appears as Yes (Overridden).
    - Lookup column. Select in_Gender.
    - Unique column. Selct in_Country.
    - Input column. Select in_FirstName.

    The following image shows the var_FirstName_us column properties:



General Properties

* Column Name	var_FirstName_us
* Column Type	Variable
* Datatype	String
* Precision	10
* Scale	10
☑ Mandatory	

Masking Properties

○ Expression	
● Masking Rule	
Condition	in_Country='us'
Rule	SubMask
Override Properties	Yes (Overridden)
Lookup Column	in_Gender
Unique Column	in_Country
Input Column	in_FirstName
○ Dependent	
Input Column	
Dictionary Column	

17. Click **OK**.

    The var_FirstName_us variable column appears in the list of variable columns.

18. Click **Create Input Column**.

    The **Add Column** window appears.

19. Enter the following general properties:

    - Column name. Enter var_FirstName_jp.
    - Column type. Select variable.
    - Datatype. Select string.
    - Precision. Select 10.

- Scale. Select 10.
- Mandatory. Select the check box.

20. Select the masking rule masking property. Configure the following masking properties:
    - Condition. Enter in_Country='jp'.
    - Rule. Select the substitution rule you created in Step 2.
    - Override Properties. Click **Edit**. Enable **Repeatable Output** and **Unique Substitution Data** options, and click **Save**. The property appears as Yes (Overridden).
    - Lookup column. Select in_Gender.
    - Unique column. Selct in_Country.
    - Input column. Select in_FirstName.

21. Click **OK**.

    The var_FirstName_jp variable column appears in the list of variable columns.

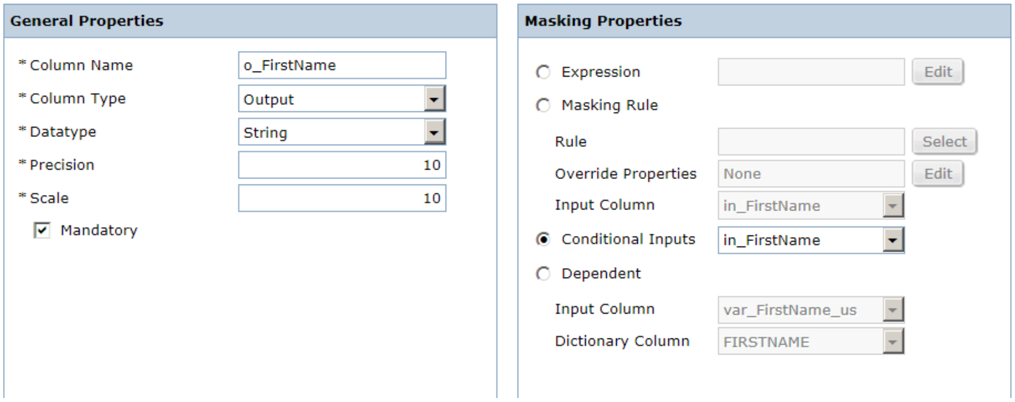
22. Click **Create Input Column**.

    The **Add Column** window appears.

23. Enter the following general properties:
    - Column name. Enter o_FirstName.
    - Column type. Select output.
    - Datatype. Select string.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.

24. Select the conditional inputs masking property. Configure the following masking property:
    - Conditional input. Select in_FirstName.

    The following image shows the o_FirstNames column properties:



General Properties

* Column Name	o_FirstName
* Column Type	Output
* Datatype	String
* Precision	10
* Scale	10
☑ Mandatory	

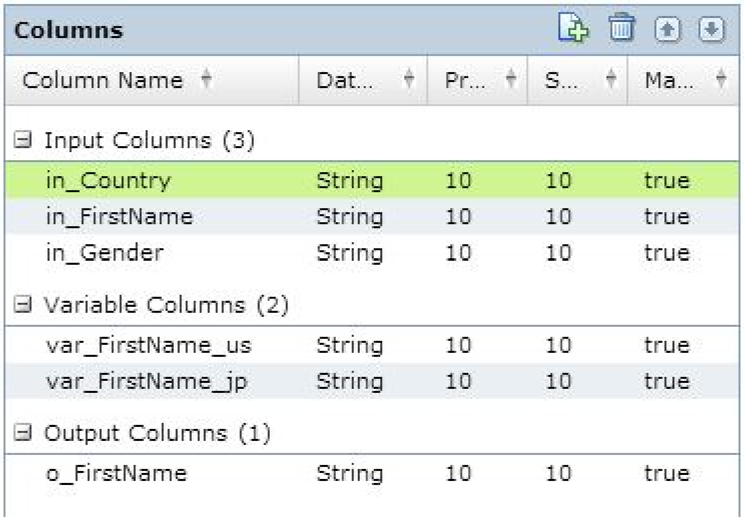
Masking Properties

○ Expression		Edit
○ Masking Rule		
Rule		Select
Override Properties	None	Edit
Input Column	in_FirstName	
● Conditional Inputs	in_FirstName	
○ Dependent		
Input Column	var_FirstName_us	
Dictionary Column	FIRSTNAME	

25. Click **OK**.

    The o_FirstName variable column appears in the list of output columns.

The following image shows the rule columns:



Columns

Column Name	Dat...	Pr...	S...	Ma...
⊟ Input Columns (3)				
in_Country	String	10	10	true
in_FirstName	String	10	10	true
in_Gender	String	10	10	true
⊟ Variable Columns (2)				
var_FirstName_us	String	10	10	true
var_FirstName_jp	String	10	10	true
⊟ Output Columns (1)				
o_FirstName	String	10	10	true

26. Click **Next**.

    Review the rule map that appears.

27. Click **Finish**.

    The rule appears in the **Policies** view.

Shuffle Address Example

You want to mask employee addresses and keep the ZIP code unmasked.

Create a shuffle rule and an advanced masking rule to shuffle addresses and leave the ZIP code unmasked. Shuffle masking masks the data in a column with data from the same column in another row of the table. Shuffle masking switches all the values for a column in a file or database table. You can restrict which values to shuffle based on a lookup condition or a constraint.

Create shuffle masking rules with a dictionary that contains three address columns and a ZIP code column. The ZIP code remains unmasked. The three address columns shuffle, but remain consistent.

The following text shows a sample flat file with the required columns:

```
SNO,AddressLine1,AddressLine2,AddressLine3,ZIP
1,3290 Apple Lane,Chillicothe,IL,61523
2,7760 Ash Street,Dallas,TX,75240
3,2229 Ash Street,Moscow,TN,38057
4,6698 Caldwell Road,Rochester,NY,14620
```

Complete the following steps to create rules to shuffle addresses with the ZIP code unmasked in Test Data Manager:

1. Create a shuffle rule.
2. Create an advanced masking rule that keeps the ZIP code column unmasked and shuffles the address columns together.

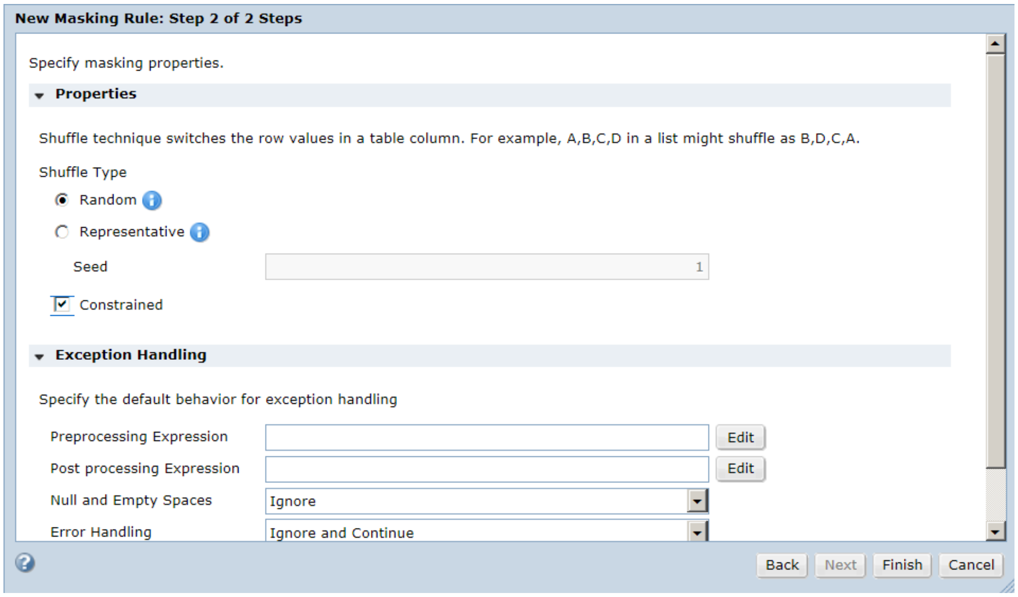
Creating the Shuffle Rule

Create a shuffle rule in Test Data Manager.

1. In the **Policies** view, click **Actions** > **New** > **Masking Rule**.

   The **New Rule** window appears.
2. Enter a name and optional description for the rule.
3. Select the string datatype and standard shuffle masking type. Select override allowed.
4. Click **Next**.
5. Enter the following shuffle masking parameters:
   - Shuffle type. Select Random.
   - Constrained. Select Constrained.

   The following image shows the shuffle masking parameters:



New Masking Rule: Step 2 of 2 Steps

Specify masking properties.

Properties

Shuffle technique switches the row values in a table column. For example, A,B,C,D in a list might shuffle as B,D,C,A.

Shuffle Type

● Random

○ Representative

Seed: 1

☑ Constrained

Exception Handling

Specify the default behavior for exception handling

Preprocessing Expression: Edit

Post processing Expression: Edit

Null and Empty Spaces: Ignore

Error Handling: Ignore and Continue

6. Click **Finish**.

   The rule appears in the **Policies** view.

Create the Advanced Masking Rule

Create an advanced masking rule that shuffles address lines.

The advanced rule shuffles three address line columns together and keeps the ZIP code column unmasked. The rule uses shuffle masking with a lookup on AddressLine3 and dependent masking on AddressLine1 and AddressLine2.

In the rule editor, you create input columns, variable columns, and output columns.

Setting Up the Advanced Masking Rule

Set up the advanced masking rule.

1. In the **Policies** view, click **Actions** > **New** > **Masking Rule**.

   The **New Masking Rule** window appears.

2. Enter a name and optional description for the rule.
3. Select the advanced masking type.
4. Click **Next**.

Creating the Input Columns

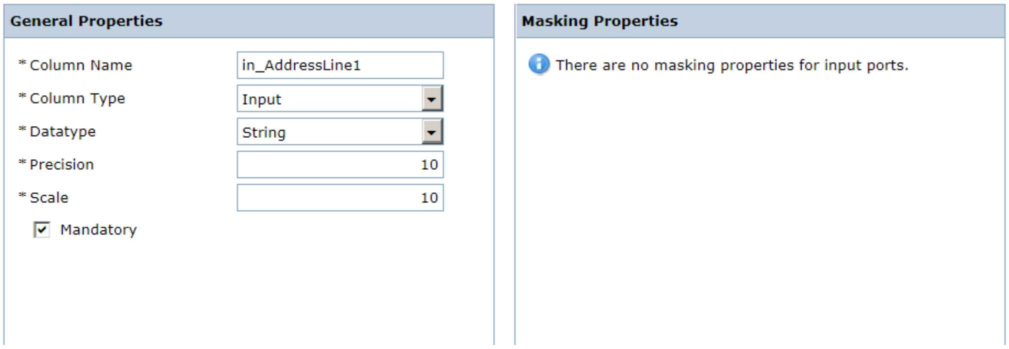
Create input columns in the advanced masking rule.

1. Create an input column. Click **Create Input Column** in the **Specify Masking Properties** window.

   The **Add Column** window appears.

2. Enter the following general properties:
   - Column name. Enter i_AddressLine1.
   - Column type. Select Input.
   - Datatype. Select String.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.

   The following image shows the i_AddressLine1 column properties:



General Properties

* Column Name	in_AddressLine1
* Column Type	Input
* Datatype	String
* Precision	10
* Scale	10

☑ Mandatory

Masking Properties

There are no masking properties for input ports.

3. Click **OK**.

   The i_AddressLine1 input column appears in the list of input columns.

4. Create an input column with the following general properties:
   - Column name. Enter i_AddressLine2.
   - Column type. Select Input.
   - Datatype. Select String.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.

5. Create an input column with the following general properties:
   - Column name. Enter i_AddressLine3.
   - Column type. Select Input.
   - Datatype. Select String.
   - Precision. Select 10.

- Scale. Select 10.
- Mandatory. Select the check box.

6. Create an input column with the following general properties:
   - Column name. Enter i_ZIP.
   - Column type. Select Input.
   - Datatype. Select String.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.

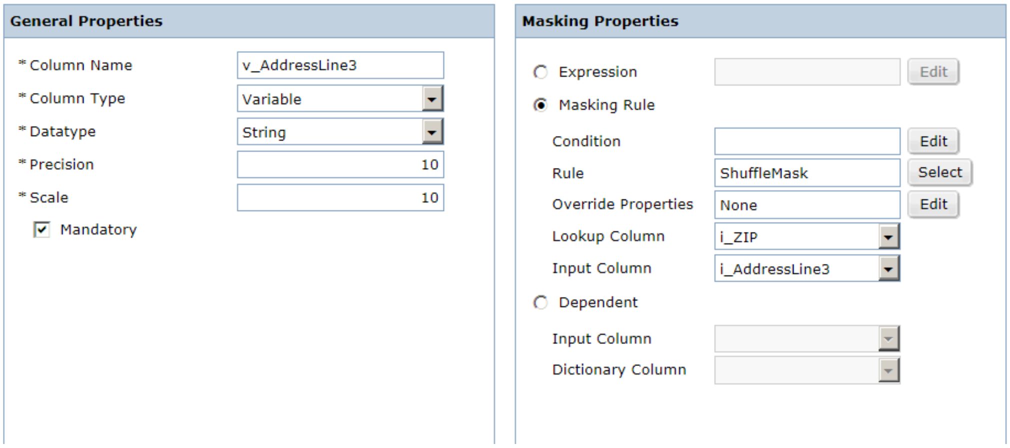
Creating the Variable Columns

Create variable columns in the advanced masking rule.

1. Create a variable column to shuffle AddressLine3 based on ZIP. Click **Create Input Column**.

   The **Add Column** window appears.
2. Enter the following general properties:
   - Column name. Enter v_AddressLine3.
   - Column type. Select Variable.
   - Datatype. Select String.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.
3. Select the masking rule masking property. Configure the following masking properties:
   - Condition. Leave blank.
   - Rule. Select the shuffle rule that you created in Step 2.
   - Override Properties. Select None.
   - Lookup column. Select i_ZIP.
   - Input column. Select i_AddressLine3.

   The following image shows the v_AddressLine3 column properties:



General Properties

* Column Name	v_AddressLine3
* Column Type	Variable
* Datatype	String
* Precision	10
* Scale	10
☑ Mandatory	

Masking Properties

○ Expression	
● Masking Rule	
Condition	
Rule	ShuffleMask
Override Properties	None
Lookup Column	i_ZIP
Input Column	i_AddressLine3
○ Dependent	
Input Column	
Dictionary Column	

4. Click **OK**.

   The v_AddressLine3 variable column appears in the list of variable columns.

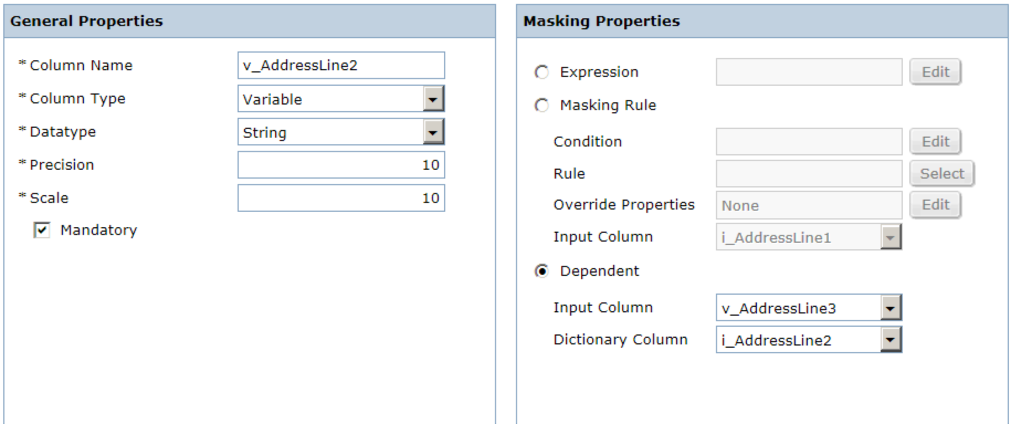
5. Create a variable column to mask AddressLine2 based on the masked value of AddressLine3. Click **Create Input Column**.

   The **Add Column** window appears.

6. Enter the following general properties:
   - Column name. Enter v_AddressLine2.
   - Column type. Select Variable.
   - Datatype. Select String.
   - Precision. Select 10.
   - Scale. Select 10.
   - Mandatory. Select the check box.

7. Select the dependent masking property. Configure the following masking properties:
   - Input column. Select v_AddressLine3.
   - Dictionary column. Select i_AddressLine2.

   The following image shows the v_AddressLine2 column properties:



General Properties

* Column Name	v_AddressLine2
* Column Type	Variable
* Datatype	String
* Precision	10
* Scale	10
☑ Mandatory	

Masking Properties

○ Expression	
○ Masking Rule	
Condition	
Rule	
Override Properties	None
Input Column	i_AddressLine1
● Dependent	
Input Column	v_AddressLine3
Dictionary Column	i_AddressLine2

8. Click **OK**.

   The v_AddressLine2 variable column appears in the list of variable columns.

9. Create a variable column to mask AddressLine1 based on the masked value of AddressLine3. Click **Create Input Column**.

   The **Add Column** window appears.

10. Enter the following general properties:
    - Column name. Enter v_AddressLine1.
    - Column type. Select Variable.
    - Datatype. Select String.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.

11. Select the dependent masking property. Configure the following masking properties:
    - Input column. Select v_AddressLine3.
    - Dictionary column. Select i_AddressLine1.
12. Click **OK**.

    The v_AddressLine1 variable column appears in the list of variable columns.

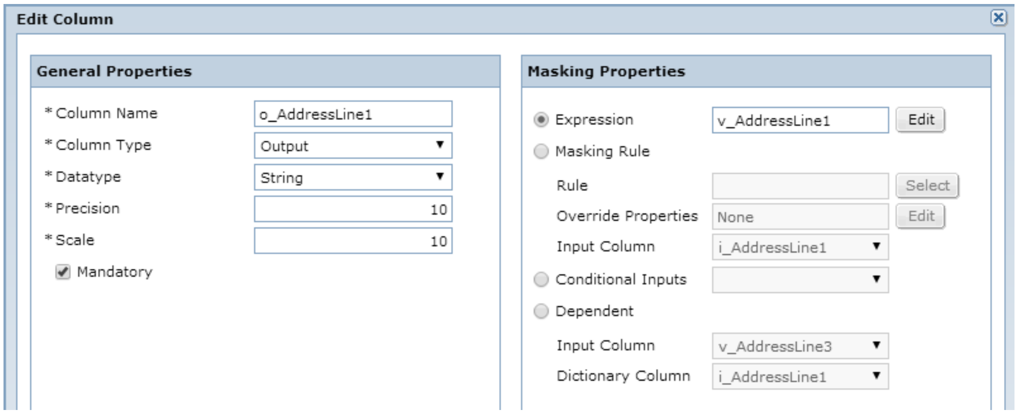
Creating the Output Columns

Create output columns in the advanced masking rule.

1. Click **Create Input Column**.

    The **Add Column** window appears.
2. Enter the following general properties:
    - Column name. Enter o_AddressLine1.
    - Column type. Select Output.
    - Datatype. Select String.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.
3. Select the dependent masking property. Configure the following dependent properties:
    - Input column. Select v_AddressLine3.
    - Dictionary column. Select i_AddressLine1.

    The following image shows the o_AddressLine1 column properties:



Edit Column

General Properties

* Column Name	o_AddressLine1
* Column Type	Output ▼
* Datatype	String ▼
* Precision	10
* Scale	10
☑ Mandatory	

Masking Properties

● Expression	v_AddressLine1	Edit
○ Masking Rule		
Rule		Select
Override Properties	None	Edit
Input Column	i_AddressLine1 ▼	
○ Conditional Inputs	▼	
○ Dependent		
Input Column	v_AddressLine3 ▼	
Dictionary Column	i_AddressLine1 ▼	

4. Click **OK**.

    The o_AddressLine1 variable column appears in the list of variable columns.
5. Click **Create Input Column**.

    The **Add Column** window appears.
6. Enter the following general properties:
    - Column name. Enter o_AddressLine2.
    - Column type. Select Output.
    - Datatype. Select String.

- Precision. Select 10.
- Scale. Select 10.
- Mandatory. Select the check box.

7. Select the dependent masking property. Configure the following dependent properties:
   - Input column. Select v_AddressLine3.
   - Dictionary column. Select i_AddressLine2.
8. Click **OK**.

   The o_AddressLine2 variable column appears in the list of variable columns.
9. Click **Create Input Column**.

   The **Add Column** window appears.
10. Enter the following general properties:
    - Column name. Enter o_AddressLine3.
    - Column type. Select Output.
    - Datatype. Select String.
    - Precision. Select 10.
    - Scale. Select 10.
    - Mandatory. Select the check box.
11. Select the dependent masking property. Configure the following dependent properties:
    - Input column. Select v_AddressLine3.
    - Dictionary column. Select i_AddressLine3.
12. Click **OK**.

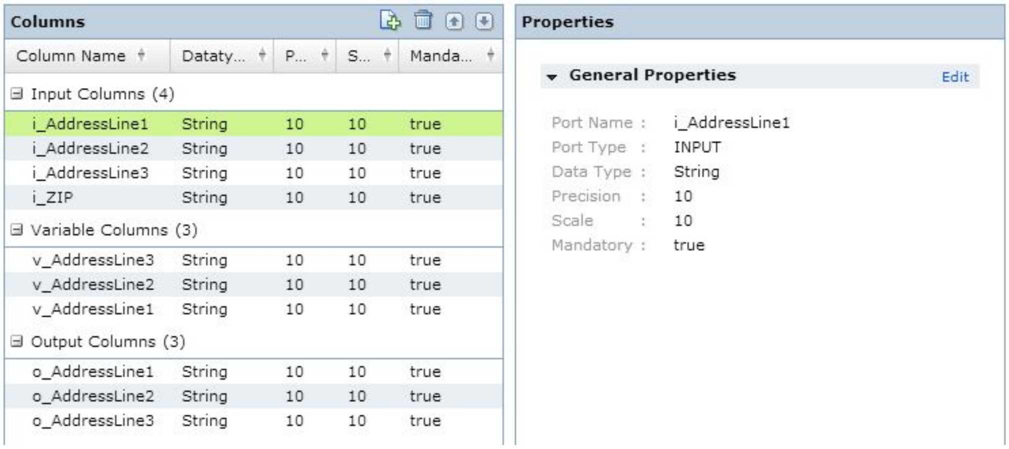
    The o_AddressLine3 variable column appears in the list of variable columns.

Saving the Advanced Masking Rule

Save the advanced masking rule.

1. Verify that all the rule columns that you created are visible in the columns list.

   The following image shows the rule columns:



Columns

Column Name	Dataty...	P...	S...	Manda...
⊟ Input Columns (4)				
i_AddressLine1	String	10	10	true
i_AddressLine2	String	10	10	true
i_AddressLine3	String	10	10	true
i_ZIP	String	10	10	true
⊟ Variable Columns (3)				
v_AddressLine3	String	10	10	true
v_AddressLine2	String	10	10	true
v_AddressLine1	String	10	10	true
⊟ Output Columns (3)				
o_AddressLine1	String	10	10	true
o_AddressLine2	String	10	10	true
o_AddressLine3	String	10	10	true

Properties

General Properties Edit

Port Name : i_AddressLine1
Port Type : INPUT
Data Type : String
Precision : 10
Scale : 10
Mandatory : true

2. Click **Finish**.

   The rule appears in the **Policies** view.

CHAPTER 9

Plans and Workflows

This chapter includes the following topics:

- Plans and Workflows Overview, 125
- Workflow Connections, 126
- Plan Components, 126
- Persist Mapping, 127
- Plan Settings, 127
- Masking Components, 130
- Subset Components, 130
- Hadoop Components, 131
- Component Criteria, 131
- Source Settings, 131
- Plan Management, 135
- Workflow Generation, 137
- Executing a Workflow, 139
- Workflow Executions View, 139

Plans and Workflows Overview

A plan defines a data subset or data masking operation. It includes the components that you need to generate a workflow.

When you create a plan, you add data masking or data subset components based on the operation that you need to perform. You can add policies and rules to run data masking operations. Add entities, groups, and templates to run data subset operations on relational or flat file sources.

If you add both subset and masking components to a plan, the subset component gets priority. TDM performs the subset operation first. TDM then applies masking rules to columns in the subset data that have masking assignments. TDM does not apply masking assignments to source data that is not included in the subset, even if the data is part of a cascade.

Define the workflow connections in the plan. Define the source and target connections. If required for the plan, define lookup and dictionary connections.

When you update or delete a plan component, you must generate and run the plan again to update the changes.

You can create multiple workflows from one plan. Define workflow properties in the plan, such as commit properties, update strategies, and recovery strategies.

When you start a workflow, the Data Integration Service completes the plan operations.

Plans and Workflows Task List

Perform the following steps to create the plan and run the workflow:

1. Create a plan. Enter a plan name and description.
2. Add data masking rules and policies to the plan.
3. Add data subset groups.
4. Switch masking rules off or on.
5. Configure plan settings including the plan properties and advanced settings.
6. Optionally, configure an override strategy for a data source or a table.
7. Generate a workflow. You can choose to generate and run a workflow in a single step.
8. Run the workflow.
9. Monitor the workflow.

Workflow Connections

Workflows contain connections to the Data Integration Service, the TDM repository, and one or more connection objects.

An offline job requires three connections to the TDM repository. When you run a plan with multiple workflows, each workflow requires one connection. You can run a maximum of eight workflows at a time. When you run eight workflows, you need 11 connections. Additional workflows move into the queued state.

Each workflow can use different connections to relational sources and paths to flat files. Choose the connections in the plan.

When you create a plan, you can select the following connections:

- Source. The connection that is used to connect to the source.
- Target. The connection that is used to connect to the target.
- Lookup connection. A connection to a database that contains lookup tables.
- Dictionary connection. A connection to a database that contains a dictionary table for substitution and email masking.

Plan Components

When you create a plan, you add components to the plan based on the operations that you want to perform.

You can perform data masking operations, data subset operations, or both.

The following table describes the components that you can add to a plan for each type of operation:

Component	Description
Rule	Data masking rules. For a data masking operation, you can add rules assigned to a policy and add rules that are not assigned to a policy. You can add rules to a data masking plan.
Policy	A Policy is a set of data domains. Each data domain can have multiple rules. You can add policies to a data masking plan.
Group	Defines a set of tables to copy to a target subset database. You can add groups to a data subset plan.

Persist Mapping

You can store the mappings in the Model repository when you run a Hadoop plan.

When you create a Hadoop plan, you can enable or disable the **Persist Mapping** option in the plan settings. Default is what the TDM administrator configures. You can choose to 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.

You can connect to the Informatica Developer Tool, create mappings in the folder, and store the mappings in the Model repository. When you choose to persist mappings in Model repository and run the Hadoop plan, the TDM generated mappings overwrite the mappings in the Model repository if the folder name in the Informatica Developer Tool is same as the plan name in TDM. If you do not want TDM mappings to overwrite the mappings in the Model repository, you must not create a folder name in the Informatica Developer Tool in the following format: `Plan_XY, where XY is the plan ID`

Plan Settings

Configure plan settings that apply to all data sources in the plan.

Connection Options

Enter source and target connections.

The following table describes connection options:

Connection Options	Description
Use Source Connection as Target Connection	Use the same connection for source and targets. Use with inplace masking.
Source Connection	A connection to the source database. Choose a variable, relational, or Hadoop HDFS connection from the list. Select the variable name or select a source connection from the list. You cannot use a Hadoop HDFS connection as a variable when you configure a source. In a Hadoop plan, you can select a Hive or an HDFS connection.
Source Connection Directory Path	Displays if flat file sources are present. Enter the path to the flat file source directory.
Target Connection	Choose a relational or flat file connection from the list. Select a target connection from the list. In a Hadoop plan, you can select a Hive or an HDFS connection.
Dictionary Lookup Connection	Connection to a database that contains lookup tables. Required if the plan has a mapplet rule that includes a transformation. Choose relational from the list. Select a dictionary lookup connection from the list.
Dictionary Connection	A connection to the database that contains the dictionary table. The dictionary connection option does not appear unless the project contains a rule that requires a relational dictionary.

Dictionary Connections

A dictionary is a relational table that contains substitute data. When you define a relational dictionary, you can define the connection to the dictionary in the project plan and workflow.

The **Dictionary Connection** field appears in the **Plan Settings** if the plan requires a connection.

Note: To access a dictionary on Microsoft SQL Server, create an ODBC connection using the DataDirect SQL Server Wire Protocol driver.

Target Options

Configure properties for committing data and error recovery in the **Target** fields.

The following table describes target options:

Other Property Options	Description
Truncate Table	Truncates the table before loading it. By default, this is not selected.

Advanced Options

The Advanced Options include options to set the locale and the date and time format in a workflow. This section is minimized by default. The options are populated with default data if available.

The following table describes advanced options:

Advanced Options	Description
Datetime Format String	Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds. - Seconds. MM/DD/YYYY HH24:MI:SS - Milliseconds. MM/DD/YYYY HH24:MI:SS.MS - Microseconds. MM/DD/YYYY HH24:MI:SS.US - Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS Default is microseconds.
Locale	Sets the locale. A locale properties file corresponding to the locale you select must exist in the `<Informatica installation directory>/TDM/lang` location.
Source Schema Name	Table owner name. Specify the source schema name if the source is in a different schema. Choose to enter a value or parameter. Enter the schema name in uppercase unless the schema name is case sensitive in the database. If the schema name is case sensitive in the database, enter the name as in the database. If you choose to enter a parameter, select the parameter name from the list.

Hadoop Plan Settings

Enter source and target connections for the Hadoop plan.

The following table describes connection options:

Connection Options	Description
Source Connection	Required. A connection to the source database. Select a source connection from the list. When you create a Hadoop plan, you can select Oracle, DB2, Sybase, Microsoft SQL Server, Hive, flat file, or HDFS connections.
Target Connection	Required. When you create a Hadoop plan, you can select a relational or an HDFS target connection from the list. When you select a relational target connection type, you can select the Hive connection.
Resource Format	Required if you select the target connection as HDFS. The format of the target file. You can select the following file formats: - None. The target contains the HDFS file format. - AVRO. A data serialization system. A complex file data object for Avro data sources in the local system. The target contains the Avro file format. - Parquet. A complex file data object for Parquet data sources in the local system. The target contains the Parquet file format.
Truncate Tables	Truncates the table before loading it. By default, this option is selected. You can truncate the tables for Hive connections. You cannot truncate tables if you use an HDFS connection or a Blaze execution engine.

Connection Options	Description
Date-time Format String	Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds. - Seconds. MM/DD/YYYY HH24:MI:SS - Milliseconds. MM/DD/YYYY HH24:MI:SS.MS - Microseconds. MM/DD/YYYY HH24:MI:SS.US - Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS Default is microseconds.
Locale	Sets the locale for data movement and data masking operations.
Persist Mapping	Optional. Stores the mappings in the Model repository for future use.
Execution Engine	The Hadoop environment that runs the mapping. Select Blaze, Spark, or Hive.

Masking Components

Masking components are the policies and masking rules in a data masking operation. When you create a plan for a data masking operation, select the data masking components that you want to include in the workflow.

When you run a data masking operation, you can restrict the masking operation to some of the masking rules instead of applying all of the masking rules. You can choose the policies and masking rules that you want to test.

The **Masking Components** dialog box shows the policies and rules in the project. Select the policies and rules that you want to apply in the masking operation. The Data Integration Service performs masking with the components that you select. You can disable and enable masking for specific columns in a workflow run when you configure the plan criteria.

Subset Components

Subset components are the groups in a data subset operation. When you create a plan for a data subset operation, select the subset components that you want to include in the workflow.

When you run a data subset operation on a relational or flat file source, you can restrict the subset operation to specific entities in the project. You can choose the groups that you want to include.

The **Add Subset Components** dialog box shows the groups in the project. Select the components that you want to apply in the data subset operation. The Data Integration Service performs the data subset operation with the components that you select.

Hadoop Components

You can add policies, rules, and groups to a Hadoop plan.

In a Hadoop plan, you can move data from source to target with or without masking. When you create a Hadoop plan, you can add groups and masking components such as policies and rules that you want to include in the workflow. If you want to move the master data to a Hadoop target without masking, you can create a group and add the group to a Hadoop plan.

In Test Data Manager, you can click the **Add Masking Components** dialog box to add the policies and rules to a plan. You can click the **Add Groups** dialog box to add the groups to a plan. The Data Integration Service performs the data masking or data movement operation when you run the Hadoop plan.

Component Criteria

You can disable and enable data masking for a column.

Disabling Masking for a Column

You can disable or enable masking for a column in the plan.

1. Click a masking component in the **Plan Components** panel.
2. Select a column from the **Criteria** panel.
3. To disable masking for the column, click **OFF**.
4. To enable masking for the column, click **ON**.

Source Settings

The source level properties are a subset of plan settings. You can change the settings for all the tables in the source or you can change the settings for each table in the source.

If a property is not available at the data source level, the property value at the plan level takes precedence. A lookup connection appears in the settings if there is a mapplet rule that is associated with a column in the data source.

To change settings at the source or table level, choose the **Source Settings** tab when you edit the plan.

Connection Properties

The following table describes connection options that you can configure at the data source level and the table level:

Connection Options	Description
Source Connection	A connection to the source database. Choose variable or relational from the list. Select the variable name or select a source connection from the list. You cannot use a Hadoop HDFS connection as a variable when you configure a source or a target. For Hadoop sources, you can select a Hive or an HDFS connection.
Source Connection Directory Path	The path to the flat file source directory. Appears if flat file sources are present.
Source Filename	The name of the source file. Required if the source is a flat file. Default is <name of selected source>.dat.
Target Connection	A connection to the target database or the target file. Choose variable, relational, or flat file from the list. Select a target connection or variable name from the list. For Hadoop targets, you can select a Hive or an HDFS connection. If you enter the target connection as a variable, ensure that you define the scope of the variable as **Global** or **Integration Service** in the parameter file. In a multinode setup, if you use parameters to enter the connection information, the parameter file must be present on the node on which the Test Data Manager Service runs. The file must be in the same directory path on both nodes.
Output Filename	The name of the target file. Required if the target is a flat file. Default is <name of selected source>.out.
File Encoding	The file encoding type. Required if the target is a flat file. If the source is a flat file, the default type is what the flat file contains. If the source is relational, default is MS Windows Latin 1 (ANSI), superset of Latin1. You can select the type of flat file encoding that you want in the target.
File Format	The format of a flat file. Required if the target is a flat file. You can choose one of the following options: - Fixed Width. The width of the columns is fixed. You cannot specify a column delimiter. - Delimited. You can limit the width of the columns. Specify a column delimiter. Default is Delimited.
Column Delimiter	A character that separates columns of data. Required if the source is a flat file and if the file format is delimited. Default is a comma (,).
Row Separator	A character that separates rows of data. Required if the source is a flat file. Default is \012 LF (\n).

Connection Options	Description

Optional Quotes	Select No Quotes, Single Quote, or Double Quotes. If you select a quote character, the Integration Service ignores delimiter characters within the quote characters. Therefore, the Integration Service uses quote characters to escape the delimiter.
	For example, a source file uses a comma as a delimiter and contains the following row:
	`342-3849, ‘Smith, Jenna’, ‘Rockville, MD’, 6.`
	If you select the optional single quote character, the Integration Service ignores the commas within the quotes and reads the row as four fields.
	If you do not select the optional single quote, the Integration Service reads six separate fields.
	When the Integration Service reads two optional quote characters within a quoted string, it treats them as one quote character. For example, the Integration Service reads the following quoted string as
	`I’m going tomorrow:`
	`2353, ‘I’’m going tomorrow’, MD`
	Additionally, if you select an optional quote character, the Integration Service reads a string as a quoted string if the quote character is the first character of the field.
	Note: You can improve session performance if the source file does not contain quotes or escape characters.

Include Headers	Optional. You can choose to include the headers in the target flat file. Default is Yes.

Target Properties

The following table describes target options that you can configure at the data source level and at the source level:

Other Property Options	Description
Truncate Table	Truncates the table before loading it. Default is disabled.

Advanced Properties

The following table describes advanced options that you can override at the data source level and at the table level:

Advanced Options	Description
Datetime Format String	Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds. - Seconds. MM/DD/YYYY HH24:MI:SS - Milliseconds. MM/DD/YYYY HH24:MI:SS.MS - Microseconds. MM/DD/YYYY HH24:MI:SS.US - Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS Default is microseconds.
Source Schema Name	Table owner name. Specify the source schema name if the source is in a different schema. Choose to enter a value or parameter. Enter the schema name in uppercase unless the schema name is case sensitive in the database. If the schema name is case sensitive in the database, enter the name as in the database. If you choose to enter a parameter, select the parameter name from the list. If the source is Cassandra, you must specify the source schema name. If the source is MongoDB, you must not specify the source schema name.

.

Hadoop Data Source Settings

Enter source and target connections for the Hadoop plan.

The following table describes connection options:

Connection Options	Description
Source Connection	Required. A connection to the source database. Select a source connection from the list. When you create a Hadoop plan, you can select Oracle, DB2, Sybase, Microsoft SQL Server, Hive, flat file, or HDFS connections.
Target Connection	Required. When you create a Hadoop plan, you can select a relational or an HDFS target connection from the list. When you select a relational target connection type, you can select the Hive connection.
Output Filename	The name of the target file. Required if the target is HDFS. Default extension is .csv.
Column Delimiter	A character that separates columns from each other in the .csv file. Required if the source is a flat file. Default is a comma (,).
Row Separator	A character that separates columns from each other in the .csv file. Required if the source is a flat file. Default is a new line.
Truncate Tables	Truncates the table before loading it. By default, this option is selected. You can truncate the tables for Hive connections. You cannot truncate tables for HDFS connections.

Connection Options	Description
Date-time Format String	Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds. - Seconds. MM/DD/YYYY HH24:MI:SS - Milliseconds. MM/DD/YYYY HH24:MI:SS.MS - Microseconds. MM/DD/YYYY HH24:MI:SS.US - Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS Default is microseconds.
Max Parallel Sessions	The maximum number of mappings that can run at the same time. Default number of mappings is 5.
Locale	Sets the locale for data movement and data masking operations.

Plan Management

After you create a plan you can edit it. You can copy the plan, export it to an XML file, or delete the plan. You can import a plan that you created in another TDM repository and exported.

Creating a Data Masking and Data Subset Plan

When you create a plan, add components to it to define its operations. You can combine a data subset and a data masking operation in the same plan, or create separate plans. Add groups to complete data subset operations. Add rules and policies to plans to perform data masking operations.

1. Open a project and click **Execute** to view the project plans.
2. Click **Actions** > **New**.
3. In the **New Plan** dialog box, enter a name and optional description for the plan.
4. To add a data masking operation to the plan, click **Add Masking Components**.
5. Select the policies and rules to add to the plan. Click **Next**.
6. To add a data subset operation to the plan, click **Add Subset Components**.
7. Select the groups to add to the plan. Click **Next**.
8. To skip masking a rule, select the check box for the rule and click **Off**.
9. To filter subset components, select the component and choose to enter a basic or advanced expression.
10. To limit the subset results, click **Limit** and choose to limit by percentage, by absolute value, or by defining an interval of rows to create. Click **Next**.
11. Configure the connections and other properties. Click **Next**.
12. To override plan settings for a data source, select the data source and click **Override Plan Settings** and enter the properties.
13. To override data source settings for a table, select the table and click **Override Data Source Settings** and enter the properties.
14. Click **Finish**.

    The plan appears in the project.

Related Topics:

- “Generating a Workflow” on page 138
- “Advanced Masking Rules” on page 78
- “Advanced Masking” on page 95

Creating a Hadoop Plan

To perform data movement and data masking operations for Hadoop connections, you can create a Hadoop plan. Add groups and data masking components to a Hadoop plan. You cannot perform data subset operations for Hadoop sources and targets.

1. Open a project and click **Execute**.
2. Click **Actions** > **New**.
3. In the **New Plan** dialog box, enter a name and optional description for the plan.
4. Select **Hadoop** plan type.
5. Click **Next**.
6. To add a data masking operation to the plan, click **Add Masking Components**.
7. Select the policies and rules to add to the plan. Click **OK**.
8. Click **Next**.
9. To add groups to the plan, click **Add Groups**. You can add groups to a plan to move data from a source to a target.
10. Select the groups to add to the plan. Click **OK**.
11. Click **Next**.
12. Review all the masking components and groups.

    You cannot edit the groups.
13. Click **Next**.
14. Configure source and target connections.
15. If you select an HDFS target connection, you can choose to select the resource format. Select Avro or Parquet. Default is None.
16. Configure target properties and error and recovery settings.
17. Configure advanced settings. You can choose to persist mapping to store mappings for future use. You can select Hive or Blaze execution engine.
18. Click **Next**.
19. To override plan settings, click **Override Plan Settings** and enter the properties.
20. To override table settings, click **Override Data Source Settings** and enter the properties.
21. Click **Finish**.

Copying a Plan

Copy a plan to create another plan with similar components. Create a copy of a plan and edit the copy.

1. Open a project and click **Execute** to view the project plans.
2. Click the plan **Description** or **Status** field to select a plan.

   Do not open the plan.

3. Click **Actions** > **Duplicate**.
4. Optionally, enter a plan name and a description.

   The default name is `Copy of <original name>`.

Exporting a Plan

You can export a group to an XML file and import the XML file to another TDM repository.

1. Open a project and click **Execute** to view the project plans.
2. Click the plan **Description** or **Status** field to select a plan.

   Do not open the plan.
3. Click **Actions** > **Export**.
4. Choose to save the file.
5. Enter the XML file name and the path of the file.

   The default name is a string that contains "Plan_" and the current date and the time.

Importing a Plan

You can import a plan from an XML file that was exported from another TDM repository.

1. To open the **Projects** view, click **Projects**.
2. Click the project description to select a project to import the plan into.

   Do not open the project.
3. Click **Actions** > **Import**.
4. Browse for the XML file that contains the group to import.

   The XML file has a default name similar to `Plan_130315081854.xml`.
5. Click **Finish** to import the plan.

Deleting a Plan

You can delete plans. When you delete the plan, you delete the workflow for the plan.

1. Open a project and click **Execute** to view the project plans.
2. Click the plan **Description** or **Status** field to select a plan.

   Do not open the plan.
3. Click **Actions** > **Delete**.
4. Click **Yes** to confirm the delete.

Workflow Generation

Generate a workflow from a plan, and then run the workflow to perform data subset and data masking operations.

You can generate and start a workflow in a single step.

When you generate a workflow, the Data Integration Service generates mappings for the workflow. When you start a workflow, the Data Integration Service performs the data masking and data subset tasks defined in the plan. You can select the Data Integration Service to run the workflow.

You can view the status of generated workflows or the load configuration on the **Monitor** view.

Generating a Workflow

After you create a plan, generate a workflow. The Data Integration Service generates the mappings to include in the workflow.

1. In a project click **Execute** to access the plans in the project.
2. Select a plan from the list.
3. Click **Actions** > **Generate Workflow**.
4. Choose whether to run it immediately or to generate the workflow later.

Related Topics:

- "Creating a Data Masking and Data Subset Plan" on page 135
- "Advanced Masking Rules" on page 78
- "Advanced Masking" on page 95

Workflow View

The Workflow view contains a list of the workflows that you generate from a plan. You can view details about the workflows and sessions that you generated from the plan.

The Workflow Details panel contains the list of workflows in the plan and the properties panel contains the workflow properties and session names.

The **Workflow** view has a **Workflow Details** panel and a **Properties** panel. The **Workflow Details** panel contains the list of workflows in the plan. The Properties panel contains properties for a workflow and the session names in the workflow.

Properties Panel

The **Properties** panel in the **Workflow** view contains a list of the workflows generated from a plan.

You can view the following information about generated workflows:

Property	Description
Workflow Name	A string that includes the project name, plan name, connection, and table name.
User	The name of the user that generated the workflow.
Generation	Date and time that the workflow was generated.

Details Panel

The **Details** panel shows the **Workflow** details and the **Session** details for a workflow that you select in the **Workflows** view.

The **Workflow** details shows the summary information about a workflow. When you select a workflow in the **Workflow** view, the workflow details appear. You can view the workflow name, the user, and the project name.

The **Session** details panel appears next to the **Workflow** details panel. The panel shows each session name and the number of tables that the session processes for a workflow.

Executing a Workflow

After you generate a workflow, you can run the workflow to run sessions from the mappings. If you generated multiple workflows for the same plan, you can run each workflow separately.

You can start a workflow from the **Plan | Properties** view or the **Plan | Workflow** view.

1. To generate and run a workflow in a single step, in the **Plan | Properties** page, click **Actions** > **Generate and Execute**.
2. You can generate and run a workflow in separate steps. If the plan you selected contains one workflow, click **Actions** > **Execute Workflow** in either view.
3. If the plan you selected contains multiple workflows, click the **Workflow** view.
   a. Select the workflow that you want to start.
   b. Click **Actions** > **Execute Workflow**.
4. Click the **Generate and Execute** button to run the workflow.
5. View the workflow status in the **Workflow Execution** view.

Workflow Executions View

The **Workflow Executions** view shows the current workflow that runs for the plan. You can also view the status of all the previous workflows. The list can contain workflow runs for workflows that no longer exist in the plan. The order of the workflow is based on the start date and time. To copy flat files to an integrated test tool, the workflow runs a separate job.

To view the workflow log, you can click **Job ID**. To view the session log, you can click **Session ID** in the **Sessions** tab.

The following table describes the fields in the **Workflow Executions** view for each workflow run:

Field	Description
Job ID	The job number that identifies the job. If you click the Job ID, you can view the workflow log.
Name	The workflow string that includes the project name, plan name, connection, and table name.

Field	Description
Description	Describes the type of job that the workflow performs. The job type can be profiling, import, and workflow operations.
Status	The current status of the workflow. - In Queue. The Data Integration Service is waiting for resources before it starts the workflow. - Running. The workflow is running. - Success. The workflow finished successfully. - Error. The workflow did not complete due to errors.
Start Date/Time	The date and time that the workflow started.
End Date/Time	The date and time that the workflow ended.
User	The name of the user that started the workflow.

Workflow Tasks

You can stop, abort, and recover the workflow in the Workflow Executions view.

You can perform the following tasks on the Workflow Executions view:

Auto Refresh

Causes the view to refresh every ten seconds automatically. You can turn auto refresh on and off.

Abort

Stops a job immediately without waiting for a commit. You can abort all jobs, such as profiling and import. You cannot abort workflows.

Workflow Stop

Stops a workflow after the next commit occurs.

Workflow Abort

Stops a workflow immediately without waiting for a commit.

Workflow Properties Panel

The **Properties** panel shows summary information about a workflow that you select in the **Workflow Executions** view.

The **Properties** panel shows the same information as the workflow you select in the list. The properties also include generation date, the elapsed time, and the project name.

Workflow Sessions Tab

The **Sessions** tab lists the sessions that ran in the workflow that you select in the **Workflow Executions** view.

The following table describes the fields for each session in the **Sessions** tab:

Field	Description
Job ID	The job number that identifies the session. If you click the Job ID, you can view the session log.
Session Name	The session name is the same value as the workflow name, except the session name starts with "S" and ends with a sequential number. If you click the session name, you can view the session details.
Status	The current status of the session. - In Queue. The Data Integration Service is waiting for resources before it starts the session. - Running. The session is running. - Success. The session finished successfully. - Error. The session did not complete due to errors.
Start Date/Time	The date and time that the session started.
End Date/Time	The date and time that the session ended.
User	The name of the user that started the workflow.

Session Details

The **Session Details** dialog box contains a list of tables that a session processed. The **Session Details** dialog box appears when you click the session name in the **Workflow Execution Sessions** tab.

If the **Workflow Executions** view has refresh enabled, the session details refresh when the workflows refresh.

The following table describes the fields in the **Session Details** dialog box:

Field	Description
Tables	The names of the tables that the session processed.

CHAPTER 10

Monitor

This chapter includes the following topics:

- Monitor Overview, 142
- Jobs, 142
- Monitor Tasks, 143
- Logs, 144
- Monitoring for Hadoop, 145

Monitor Overview

In the **Monitor** view, you can monitor the status of jobs that you start in Test Data Manager. You can stop jobs from running and view job and session logs.

You can monitor jobs for all projects in the **Monitor** view. To monitor jobs for a single project, open the project and click **Monitor**.

You can sort, filter, and perform tasks on jobs in the **Monitor** view. Select a job in the **Monitor** view to view the job details in the **Properties** pane. You can also view the workflow and session logs for a job.

Jobs

Check the status of a job and view the job details in the **Monitor** view.

You can view the following types of jobs in the **Monitor** view:

Import from Source

Imports source data from a source file.

Profiling

Performs data discovery for data domains.

Generate Workflow

Generates a workflow from a plan.

Execute Workflow

Runs a workflow for data subset or data masking operations after you generate the workflow.

Workflow

Runs the data subset or data masking operation. The execute workflow job might run multiple workflows.

Session

Performs a task within the workflow. A workflow might have multiple sessions. Click on a workflow job ID to view the session details in another tab in the **Session** pane.

Job Details

You can sort and filter jobs by job details.

The **Monitor** view contains the following job details:

Job ID

The job ID number. TDM creates consecutive job ID numbers for each job. When you click on an Execute Workflow job, the workflow details open in a separate tab.

Name

The name or the type of job. You can view import, profiling, and workflow jobs.

Description

The name of the plan, profile, Model repository folder, or connection associated with the job.

Status

The status of the job. A job can have the following statuses:

- Error. The job did not run successfully. Click the job ID to view the job log file.
- In Queue. The job is in the queue to run.
- Running. The job is running.
- Success. The job ran successfully.
- Terminated. The job was terminated.

Project

The name of the project that contains the job. Project details are not visible in the **Monitor** view within a project.

Start Date / Time

The date and time the job started.

End Date / Time

The date and time the job ended.

User

The user that ran the job.

Monitor Tasks

You can perform tasks in the Monitor view based on the job that you view.

You can perform the following tasks in the Monitor view:

Auto Refresh

Refreshes the view every ten seconds. If you disable the auto refresh, click **Actions** > **Refresh** to manually refresh.

Abort

Stops a job immediately. You can abort all jobs except workflows.

Workflow Stop

Stops a workflow after the next commit occurs.

Workflow Abort

Stops a workflow immediately.

Logs

You can view logs to troubleshoot jobs. To view the logs, select a job and click the **Logs** tab. You can view the session, workflow, and console logs.

When a TDM job runs, the TDM server generates logs. The Data Integration Service generates the session and workflow logs. For Hadoop operations, the Data Integration Service generates the session and workflow logs. When a TDM job fails, you can view the logs to debug problems.

When a job is triggered, TDM always generates console logs. If the job logs are not available, you can view the console logs to check the messages.

When you click a job ID, you can view the following log details:

Date/ Time

The date and time the job ended.

Severity

The severity level of the log messages.

Description

The detailed description of the error message, the cause of the problem, and the solution.

Severity Levels

You can view the severity level of a log message and get a clear understanding of the problem level.

The log messages include the following severity levels:

Error

Indicates that the TDM server failed to perform an operation or respond to a request from a client application.

Warning

Indicates that the TDM server is performing an operation that might cause an error.

Info

Indicates that the TDM server is performing an operation that does not cause errors or problems.

Debug

Indicates TDM Server operations at a detailed level. The debug messages generally record the success or failure of server operations.

Trace

Indicates TDM Server operations at a more specific level than the debug logs. Trace messages are generally trace code paths.

Viewing the Log Messages

View the log messages to troubleshoot the problems when a TDM job fails. You can search and filter the log messages based on the dates, severity levels, and keywords in the error description.

1. Click **Monitor**.
2. Click a job ID and click the **Logs** tab.

   A list of log messages appears.
3. Search and filter the logs from the list of log messages.
4. Select the log message and download the log file.
5. To download the log file, click **Download**.
6. To view console logs for each job, select a job ID and click **Actions** > **View Console Logs**.

Monitoring for Hadoop

You can view the mapping logs for a Hadoop plan.

You can click the Job ID to view the mapping logs. You cannot view the source and target row count details because Hadoop mappings do not return row count details for any job.

You cannot view the workflow logs for a Hadoop plan.

CHAPTER 11

Reports

This chapter includes the following topics:

- Reports Overview, 146
- Audit Trail Report, 147
- Data Masking Report, 147
- Plan Audit Report, 147
- Plan Detail Report, 149

Reports Overview

You can run a report from Test Data Manager to see detailed information about recent activities, a plan or a policy. Run the type of report that contains the information you want to see.

Test Data Manager contains activity, data masking, plan audit, plan detail, and row count reports. You can run a data masking report on a project. You can run a plan audit, plan detail, or a row count report on a plan.

When you run a report in Test Data Manager, the report opens in a new window. Make sure you do not have a pop-up blocker enabled so that the report window can open.

The following table describes the reports that you can run in Test Data Manager:

Report	Description
Audit Trail Report	A report that you run from the Overview view that lists all recent activities performed in Test Data Manager.
Data Masking Report	A report that you run on a project that lists all the table columns and the rules and policies that are assigned to the columns.
Plan Audit Report	A report that you run on a plan that contains detailed information about the policy assignments in the plan.
Plan Detail Report	A report that you run on a plan that lists policy assignments and groups that are part of the plan.
Row Count	A report that you run on a plan that lists the tables in the plan and the number of rows in each table that the plan affects.

Audit Trail Report

An audit trail report lists the recent activities performed in Test Data Manager.

The report contains information on the type of activity, type of object, name of object, the user that performed the activity, and the date and time. You can apply filters to view specific activity records or to view activities for a specific date range.

An audit trail report does not consider permissions and privileges. A report that you generate includes all recent activities.

You can download the report in `.csv` file format.

Running an Audit Trail Report

Run an audit trail report to view all recent activities performed in Test Data Manager. You can download the report in CSV file format.

1. Log in to Test Data Manager.
2. Click the **Audit Trail Report** button in the **Recent Activities** tab.

   The Audit Trail Report window opens. The report lists all recent activities.
3. Optional. You can filter and view records that match specific requirements.
4. Optional. Click the **Download** button to download the report as a CSV file `Recent Activities.csv`.

Data Masking Report

A data masking report lists all the columns within a project and the rules and policies that you assign to the project.

Running the Data Masking Report

1. Click **Projects** to view a list of the projects in Test Data Manager.
2. Click a project in the list to open it.
3. Click **Define** > **Data Masking** > **Print Masking Report** to view the project plans.

   The data masking report opens in a new window.

Plan Audit Report

Run a plan audit report to view detailed information about the data masking components that took part in a plan execution.

You can generate a plan audit report to present to auditors or administrators that need comprehensive information about the policies and rules that took part in a plan execution.

You can create a plan audit report for a plan that has successfully completed at least one execution.

The following table lists the properties in a plan audit report:

Component	Properties
Policy Assignment	You can view the following policy assignment details: - Data Source - Table - Column - Data Type - Data Domain - Policy - Rule - Rule Type - Additional Details - Masking Properties - Exception Handling
Plan Execution	You can view the following plan execution details: - Source Connection - Target Connection - Created Date - Plan Start Date - Plan End Date - Total Rows Processed - Status of the Plan

Running a Plan Audit Report

1. Click **Projects** to view a list of the projects in Test Data Manager.
2. Click a project in the list to open it.
3. Click **Execute** to view the project jobs.
4. Click a plan.
5. Click **Actions** > **Plan Audit Report**.

   The row count report opens in a new window.

Plan Detail Report

A plan detail report lists the data subset and data masking components within a plan.

The following table lists the properties in a plan detail report:

Component	Detail
Masking Assignment	You can view the following policy assignment details: - Data Source - Table - Column - Data Type - Data Domain - Policy - Rule - Masking Properties
Group	You can view the following group details: - Name - Description

Running the Plan Detail Report

1. To view a list of the projects, click **Projects**.
2. Click a project in the list to open it.
3. To view the project plans, click **Execute**.
4. Click a plan.
5. Click **Actions** > **Plan Detail Report**.

   The plan detail report opens in a new window.

CHAPTER 12

ilmcmd

This chapter includes the following topics:

- ilmcmd Overview, 150
- Configuring ilmcmd, 151
- Running ilmcmd, 151
- Entering Options and Arguments, 151
- Syntax Notation, 152
- Delete, 152
- Export, 153
- Import, 155
- Search, 156
- Workflow, 157

ilmcmd Overview

ilmcmd is a command line program that you use to perform TDM tasks. You can use ilmcmd to complete a subset of the tasks that you can perform with Test Data Manager.

Use ilmcmd to perform the following tasks:

- Import XML files into the TDM repository.
- Export data subset and data masking objects to XML files.
- Search for data subset and data masking objects.
- Validate policies, plans, and data subset objects.
- Delete data subset and data masking objects.
- Generate and run workflows.
- Display the status of workflows.

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

Configuring ilmcmd

When Test Data Manager runs on HTTPS, you must configure the ilmcmd before you can run ilmcmd commands.

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

```
<Install directory>/TDM/utilities/ilmcli/conf
```

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

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

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

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

   ```
   <Installation directory>/TDM/utilities/ilmcli/conf/userConfig.ilm
   ```

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

Running ilmcmd

Invoke ilmcmd from the command line. You can issue commands directly or from a script, batch file, or other program. On UNIX, ilmcmd is a shell script with no extension.

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

   By default, ilmcmd is installed in the following directory:

   ```
   <installation directory>\utilities\ilmcli\bin
   ```

2. Enter `ilmcmd` followed by the command name and its required options and arguments.

   For example:

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

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

```
ilmcmd -h
```

Entering Options and Arguments

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

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

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

Syntax Notation

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

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

Convention	Description
-x	Option placed before an argument. This designates the parameter you enter. For example, to enter the user name for ilmcmd, type -un or -UserName followed by the user name.
<x>	Required option. If you omit a required option, the command line program returns an error message.
<x | y>	Select between required options. If you omit a required option, the command line program returns an error message. If a pipe symbol (|) separates options, specify exactly one option. If options are not separated by pipe symbols, specify all the options.
[x]	Optional option. The command runs whether or not you enter these options.
[x | y]	Select between optional options. For example, you can display help for all ilmcmd commands by using the -h or -help option. `[-h|-help]` The command runs whether or not you enter the optional parameter.
< <x | y> | <a | b> **>**	When a set contains subsets, the superset is indicated with bold brackets (**< >**). Subsets are separated with the bold pipe symbol (**|**).

Delete

Deletes objects from the TDM repository. To delete an object, specify the name of the object.

When you delete an object by name, you specify the object type and the location of the object in the repository. You can delete the following object types:

- Data domain
- Masking rule
- Policy
- Project
- Plan

The ilmcmd delete command uses the following syntax:

```
ilmcmd
<-Delete | -d>
<-ObjectType | -ot> object_type
<-Project> project_name
<-Name | -n> object_name
[-Help | -h]
```

The following table describes ilmcmd delete options and arguments:

Option	Argument	Description
-ObjectType -ot	object_type	Required to delete an object by name. The type of object to delete. For example, you can enter "Project."
-Project	project_name	Required to delete an object by name. The name of the project that contains the object to delete.
-Name -n	object_name	Required to delete an object by name. The name of the object that you want to delete.
-Help -h	n/a	Optional. Displays help for the command.

Delete Examples

Deleting a Masking Rule

The following sample command deletes the `Credit_Card_Mask` rule on UNIX:

```
./ilmcmd -d -ot MaskingRule -n Credit_Card_Mask
```

Deleting a Policy

The following sample command deletes the `Personal` policy on UNIX:

```
./ilmcmd -d -ot Policy -n Personal
```

Deleting a Project

The following sample command deletes the `Customer_DataGeneration` project on UNIX:

```
ilmcmd -d -ot Project -n Customer_DataGeneration
```

Export

Exports objects from the Test Data Manager to an XML file.

When you export an object from the TDM repository, you specify the object type and the location of the object in the repository. You can export the following object types:

- Masking rule
- Policy
- Data domain
- Group
- Profile
- Project
- Data source
- Source definition
- Connections

- Plan

The ilmcmd export command uses the following syntax:

```
ilmcmd
<-Export | -e>
<-ObjectType | -ot> object_type
<-Name | -n> object_name
<-Project> project_name
<-File | -f> XML_file_location
<-DataSourceName | -dsn> schema_name
[-Help | -h]
```

The following table describes the ilmcmd export options and arguments:

Option	Argument	Description
-ObjectType -ot	object_type	Required. The type of object to export. For example, you can enter "Group" or "RuleAssignment." If the object type contains a space in it, remove the space when you enter the argument.
-Name -n	object_name	Required. The name of the object you want to export.
-Project -p	project_name	Required to export a plan or group. The name of the project that contains the plan or port assignment to export.
-File -f	XML_file_location	Required. The path and file name of the XML file to which you export the object.
-DataSourceName -dsn	schema_name	Required to export a port assignment. The name of the schema that contains the port assignment to export.
-Help -h	n/a	Optional. Displays help for the command.

Export Examples

Exporting a Policy

The following sample command exports the `Policy_Customer` policy to the `Policy_Customer.xml` file:

```
ilmcmd -e -ot Policy -n Policy_Customer -f C:\Informatica\ILMServer\CLI
\Policy_Customer.xml
```

Exporting a Masking Rule

The following sample command exports the Credit_Card_Mask rule to the `CCR.xml` file:

```
ilmcmd -e -ot MaskingRule -n Credit_Card_Mask -f /home/infa1/Desktop/CCR.xml
```

Exporting a Project

The following sample command exports the CustRecord_Data project to the `CustRecord.xml` file:

```
ilmcmd -e -ot Project -n CustRecord_Data -f E:\project\CustRecord.xml
```

Import

Imports objects from an XML file to the Test Data Manager.

You can import the following object types:

- Masking rule
- Policy
- Data domain
- Group
- Profile
- Project
- Data source
- Source definition
- Plan

The ilmcmd import command uses the following syntax:

```
ilmcmd
<-Import | -i>
<-File | -f> XML_file_location
[-Help | -h]
```

The following table describes the ilmcmd import options and arguments:

Option	Argument	Description
-File -f	XML_file_location	Required. The path and file name of the import file.
-Help -h	n/a	Optional. Displays help for the command.

Import Examples

Importing a Plan

The following sample command imports the plan listed in the `Plan_Customer.xml` file to the repository:

```
ilmcmd -i -f C:\Informatica\ILMServer\CLI\Plan_Customer.xml -un Administrator -pd
Administrator -hn ilmserver -port 6002
```

Importing a Source Definition

The following sample command imports the source definition listed in the `SrcDef_CustomerDB.xml` file to the repository:

```
ilmcmd -i -f C:\Informatica\ILMServer\CLI\SrcDef_CustomerDB.xml -un Administrator -pd
Administrator -hn ilmserver -port 6002
```

Search

Searches for source definitions, masking rules, policies, and plans.

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

- Masking rule
- Policy
- Data domain
- Group
- Profile
- Project
- Data source
- Source definition
- Connection
- Plan

The ilmcmd search command uses the following syntax:

```
ilmcmd
<-Search | -s>
<-ObjectType | -ot> object_type
<-Project | -p> project_name
<-DataSourceName | -dsn> schema_name
<-NamePattern | -np> name_pattern
[-Help | -h]
```

The following table describes the ilmcmd search options and arguments:

Option	Argument	Description
-ObjectType -ot	object_type	Required. The type of object to export. For example, you can enter "Group" or "Project." If the object type contains a space in it, remove the space when you enter the argument.
-Project -p	folder_name	Required to search for a source definition, plan, or group. The name of the project that contains the object to search for.
-DataSourceName -dsn	schema_name	Required to search for a source definition. The name of the schema that contains the source definition.
-NamePattern -np	name_pattern	Required. The name pattern. Use the asterisk (*) character as a wild card. For example, the name pattern `Rule_C*` returns the following rules: - `Rule_Customer` - `Rule_CustID`
-Help -h	n/a	Optional. Displays help for the command.

Search Examples

Searching for a Masking Rule

The following sample command searches for rules that match the name pattern `Rule*`:

```
ilmcmd -s -ot Rule -np Rule.*
```

The sample command returns the following output:

```
Rule_Customer
Rule_Ticket
```

Searching for a Policy

The following sample command searches for policies that match the name pattern `CUST*`:

```
ilmcmd -s -ot Policy -np CUST.*
```

The sample command returns the following output:

```
CUSTOMER
CUST_COUPONS
```

Workflow

Generates a workflow, displays the status of a workflow run, or runs a workflow.

The ilmcmd workflow command uses the following syntax:

```
ilmcmd
<-WorkFlow | -wf>
<<-Generate | -g> |  <-Execute | -ex>  | <-GetStatus | -gs>>
<-Project> folder_name
<-PlanName | -pn> plan_name
<-ParamFile | -pf paramfile
<-WorkFlowName | -wfn> Workflow_name
<-IntegrationService | -is> integration_service_name
[-Help | -h]
```

The following table describes the ilmcmd workflow options and arguments:

Option	Argument	Description
-Generate -g	n/a	Generates a workflow.
-Execute -ex	n/a	Runs the workflow.
-GetStatus -gs	n/a	Gets the status of a workflow.
-GetWorkFlows -gwf	n/a	Gets workflows for a plan.
-Project	Project	Required. The name of the folder that contains the plan.

Option	Argument	Description
-PlanName -pn	PlanName	Required. The plan name.
-paramfile -pf	ParamFile	Required if the plan contains parameters and you want to use parameter values from a specific parameter file. If you do not enter this option, the workflow uses the parameter file specified in the plan.
-WorkFlowName -wfn	WorkflowName	Required to generate a workflow. The location of the workflow properties file.
-IntegrationService -is	IntegrationService	Required to run a workflow. The name of the Data Integration Service. Required when you run the workflow.
-JobId	JobId	Required. The job ID of the workflow run.
-Help -h	n/a	Optional. Displays help for the command.

Workflow Examples

Generating a Workflow

The following sample command generates a workflow for the `Plan_NameMasking` plan:

```
ilmcmd -wf -g -Project Infa_Project -pn Plan_NameMasking
```

Monitoring a Workflow

The following sample command monitors the status of a workflow for a plan:

```
ilmcmd -wf -gs -JobId 360
```

Running a Workflow

The following sample command runs a workflow for the `Plan_NameMasking` plan:

```
ilmcmd -wf -ex -Project TDM -pn Plan_NameMasking -is PCInteg
```

APPENDIX A

Data Type Reference

This appendix includes the following topics:

- Data Type Reference Overview, 159
- Oracle, 159
- Microsoft SQL Server, 162
- DB2 for Linux, UNIX, and Windows, 163
- Sybase ASE, 165
- HDFS, 167
- Hive, 167
- Flat File, 168

Data Type Reference Overview

You can pass different data types from sources to targets and perform TDM operations on the data types.

In TDM, you can perform data masking, data subset, and data discovery operations on the source databases for supported data types. You cannot perform the operations on data types that TDM does not support. For some data types, you might need to configure the precision and scale to a specific range of values.

Oracle

The following table describes the operations that you can perform on data types in Oracle database:

Data Types	Data Movement	Data Masking	Profiling	Comments
Binary_Double	Yes	Yes	Yes	
Binary_Float	Yes	Yes	Yes	
Blob	Yes	Yes	No	You can perform nullification masking.
Char	Yes	Yes	Yes	

Data Types	Data Movement	Data Masking	Profiling	Comments
Clob	Yes	Yes	No	
Date	Yes	Yes	Yes	You cannot run data domain profiles.
Dburitype	No	No	No	
Decimal(P,S)	Yes	Yes	Yes	
Float	Yes	Yes	Yes	
Httpuritype	No	No	No	
Integer	Yes	Yes	Yes	
Interval day to second	Yes	No	No	For the Interval day(9) to second(9) data type, you must set the precision greater than 18.
Interval year to month	Yes	No	No	For the Interval year(9) to month data type, you must set the precision greater than 11.
Long	Yes	Yes	No	
Longraw	Yes	Yes	No	You can perform nullification masking.
Longvarchar	Yes	Yes	No	
Nchar	Yes	Yes	Yes	
Nclob	Yes	Yes	No	
Number	Yes	Yes	Yes	
Nvarchar	Yes	Yes	Yes	
Nvarchar2(N)	Yes	Yes	Yes	
Raw	Yes	Yes	No	You can perform nullification masking.
Real	Yes	Yes	Yes	
Rowid	Yes	No	No	
Smallint	Yes	Yes	Yes	
Timestamp	Yes	Yes	Yes	You cannot run data domain profiles.

Data Types	Data Movement	Data Masking	Profiling	Comments
Timestamp with time zone	Yes	No	No	
Urowid	Yes	No	No	
Varchar	Yes	Yes	Yes	
Varchar2(N)	Yes	Yes	Yes	
XML	Yes	No	No	
Xdburitype	No	No	No	

You cannot perform data masking, data subset, and profiling operations on the following Oracle data types:

- Anydata
- Anydataset
- Anytype
- Bfile
- Ordaudio
- Orddicom
- Orddoc
- Ordimage
- Ordimagesignature
- Ordvideo
- Sdo_Geometry
- Sdo_Georaster
- Sdo_Topo_Geometry
- SI_Averagecolor
- SI_Color
- SI_Colorhistogram
- SI_Featurelist
- SI_Positionalcolor
- SI_Stillimage
- SI_Texture
- Timestamp with local time zone
- Uritype

Microsoft SQL Server

The following table describes the operations that you can perform on data types in Microsoft SQL Server database:

Data Types	Data Movement	Data Masking	Profiling	Comments
Bigint	Yes	Yes	Yes	
Binary	Yes	Yes	No	Use nullification data masking rule to mask the data.
Bit	Yes	Yes	Yes	If you enter data patterns, you must provide T and F values instead of 1 and 0.
Char	Yes	Yes	Yes	
Date	Yes	Yes	Yes	
Datetime	Yes	No	Yes	You cannot run data domain profiles.
Datetime2	Yes	Yes	No	
Datetimeoffset	Yes	No	No	
Decimal	Yes	Yes	Yes	You can enter a precision value up to 28.
Float	Yes	Yes	Yes	To perform a data masking operation, you must set the precision to less than 28.
Image	Yes	Yes	No	
Int	Yes	Yes	Yes	
Money	Yes	Yes	Yes	
Nchar	Yes	Yes	Yes	
Ntext	Yes	Yes	No	
Numeric	Yes	Yes	Yes	You can enter a precision value up to 28.
Nvarchar	Yes	Yes	Yes	
Real	Yes	Yes	Yes	
Smalldatetime	Yes	No	Yes	You cannot run data domain profiles.

Data Types	Data Movement	Data Masking	Profiling	Comments
Smallint	Yes	Yes	Yes	
Smallmoney	Yes	Yes	Yes	
Sql_variant	Yes	No	No	You must increase the precision value after you import metadata.
Text	Yes	Yes	No	
Time	Yes	Yes	Yes	When you enter values in milliseconds, TDM truncates the data.
Timestamp	n/a	n/a	No	
Tinyint	Yes	Yes	Yes	
Uniqueidentifier	Yes	No	Yes	You cannot run data domain profiles.
Varbinary	Yes	Yes	No	Use nullification data masking rule to mask the data.
Varchar	Yes	Yes	Yes	
XML	Yes	No	No	

You cannot perform data masking, data subset, and profiling operations on the following Microsoft SQL Server data types:

- Hierarchyid
- Geography
- Geometry

DB2 for Linux, UNIX, and Windows

The following table describes the operations that you can perform on data types in DB2 database for Linux, UNIX, and Windows operating systems:

Data Types	Data Movement	Data Masking	Data Discovery	Comments
Bigint	Yes	Yes	Yes	
Blob (N)	Yes	Yes	No	You can perform nullification masking.

Data Types	Data Movement	Data Masking	Data Discovery	Comments
Char for bit data	Yes	Yes	No	
Char varying (Length)	Yes	Yes	Yes	To run a profile, you must keep the field length less than or equal to 255.
Character (N)	Yes	Yes	Yes	
Clob (N)	Yes	Yes	No	You can perform nullification masking.
Date	Yes	Yes	Yes	
Dbclob (N)	Yes	Yes	No	You can perform nullification masking.
Dec	Yes	Yes	Yes	
Decfloat (16)	Yes	Yes	No	
Decfloat (34)	Yes	Yes	No	
Decimal	Yes	Yes	Yes	
Double	Yes	Yes	Yes	
Double Precision	Yes	Yes	Yes	
Float	Yes	Yes	Yes	You can enter a precision value up to 28.
Graphic (N)	Yes	Yes	Yes	
Integer	Yes	Yes	Yes	
Long Varchar	Yes	Yes	No	
Long Vargraphic	Yes	Yes	No	
Num	Yes	Yes	Yes	
Numeric	Yes	Yes	Yes	
Real	Yes	Yes	Yes	
Smallint	Yes	Yes	Yes	
Time	Yes	Yes	Yes	
Timestamp	Yes	Yes	Yes	
Varchar for Bit Data	Yes	Yes	No	

Data Types	Data Movement	Data Masking	Data Discovery	Comments
Varchar (N)	Yes	Yes	Yes	To run a profile, you must keep the field length less than or equal to 255.
Vargraphic (N)	Yes	Yes	Yes	
XML	Yes	No	No	

Sybase ASE

The following table describes the operations that you can perform on data types in Sybase ASE database:

Data Types	Data Movement	Data Masking	Profiling	Comments
Bigdatetime	Yes	Yes	Yes	
Bigint	Yes	Yes	Yes	
Bigtime	Yes	Yes	Yes	
Bit	Yes	Yes	Yes	
Binary (n)	Yes	Yes	No	You can perform nullification masking.
Char (n)	Yes	Yes	Yes	You can enter characters up to 2000. To run a profile, you must enter the field length less than or equal to 255.
Date	Yes	Yes	Yes	
Datetime	Yes	Yes	Yes	
Decimal (P,S)	Yes	Yes	Yes	You can enter a precision value up to 28.
Double Precision	Yes	Yes	Yes	You can perform data masking for precision values up to 28.
Float (P)	Yes	Yes	Yes	You can perform data masking for precision values up to 28.
Image	Yes	Yes	No	You can perform nullification masking.

Data Types	Data Movement	Data Masking	Profiling	Comments
Int	Yes	Yes	Yes	
Longsysname	Yes	Yes	Yes	
Money	Yes	Yes	Yes	
Nchar (N)	Yes	Yes	No	
Numeric (P,S)	Yes	Yes	Yes	You can enter a precision value up to 28.
Nvarchar (N)	Yes	Yes	No	
Real	Yes	Yes	Yes	
Smalldatetime	Yes	Yes	Yes	
Smallint	Yes	Yes	Yes	
Smallmoney	Yes	Yes	Yes	
Sysname	Yes	Yes	Yes	
Text	Yes	Yes	No	
Time	Yes	Yes	Yes	
Timestamp	Yes	Yes	No	You can perform nullification masking. Timestamp values are system generated in Sybase database. You can restrict the data type from Test Data Manager to generate data.
Tinyint	Yes	Yes	Yes	
Unichar (N)	Yes	Yes	Yes	
Unitext	Yes	Yes	No	
Univarchar (N)	Yes	Yes	Yes	
Unsigned bigint	Yes	Yes	Yes	You cannot provide negative values.
Unsigned int	Yes	Yes	Yes	You cannot provide negative values.
Unsigned smallint	Yes	Yes	Yes	You cannot provide negative values.

Data Types	Data Movement	Data Masking	Profiling	Comments
Unsigned tinyint	Yes	Yes	Yes	
Varbinary (N)	Yes	Yes	No	You can perform nullification masking.
Varchar (N)	Yes	Yes	Yes	To run a profile, you must enter the field length less than or equal to 255.

HDFS

The following table describes the operations that you can perform on data types in HDFS:

Data Types	Data Movement	Data Masking	Comments
Datetime	Yes	Yes	When you create a Hadoop plan, you must provide the date time format in the source.
Number	Yes	Yes	
String	Yes	Yes	

Hive

The following table describes the operations that you can perform on data types in Hive database:

Data Types	Data Movement	Data Masking
Bigint	Yes	Yes
Binary	Yes	Yes
Boolean	Yes	Yes
Decimal	Yes	Yes
Double	Yes	Yes
Float	Yes	Yes
Int	Yes	Yes

Data Types	Data Movement	Data Masking
Smallint	Yes	Yes
String	Yes	Yes
Tinyint	Yes	Yes

You cannot perform a data movement and data masking operation on the following Hive data types:

- Array
- Char
- Date
- Maps
- Struct
- Timestamp
- Varchar

Flat File

The following table describes the operations that you can perform on data types in flat files:

Data Types	Data Movement	Data Masking	Profiling	Comments
Bigint	Yes	Yes	Yes	
Datetime	Yes	Yes	Yes	
Double	Yes	Yes	Yes	
Int	Yes	Yes	Yes	
Nstring	Yes	Yes	Yes	
Number	Yes	Yes	Yes	
String	Yes	Yes	Yes	

APPENDIX B

Data Type Reference for Hadoop

This appendix includes the following topics:

- Data Type Reference for Hadoop Overview, 169
- Oracle, 170
- Microsoft SQL Server, 172
- DB2 for Linux, UNIX, and Windows, 173
- Sybase ASE, 174
- Hive, 176
- HDFS, 177
- Flat File, 177

Data Type Reference for Hadoop Overview

You can perform data movement, data domain discovery, and data masking operations on Hadoop data sources.

Use Hive and HDFS connections in a Hadoop plan to perform data movement, data domain discovery, and data masking operations. When you generate and run the Hadoop plan, TDM generates the mappings and the Data Integration Service pushes the mappings to the Hadoop cluster to improve the performance.

When the target is Hive, or HDFS, TDM supports the data types for the following source connections:

- Oracle
- Microsoft SQL Server
- DB2 for Linux, UNIX, and Windows
- Sybase ASE
- Hive
- HDFS
- Flat File

Oracle

The following table describes the supported Oracle data types when the target is Hive or HDFS:

Data Types	Hive	HDFS	Comments
Blob	Yes	No	
Char	Yes	Yes	
Clob	Yes	Yes	
Date	No	No	
Decimal(P,S)	Yes	Yes	You can enter a precision value up to 28.
Float	Yes	Yes	You can enter a precision value up to 28.
Integer	Yes	Yes	
Long	Yes	Yes	
Longraw	Yes	No	
Longvarchar	Yes	Yes	
Nchar	Yes	Yes	
Nclob	Yes	Yes	
Number	Yes	Yes	You can enter a precision value up to 28 and scale value up to 27.
Nvarchar2(N)	Yes	Yes	
Raw	Yes	No	
Real	Yes	Yes	
Smallint	Yes	Yes	
Timestamp	Yes	Yes	When you move data from Oracle to HDFS, TDM truncates nanoseconds value and displays zeroes for the data types Timestamp (1) to Timestamp (9).
Varchar2(N)	Yes	Yes	

When the target is Hive or HDFS, TDM does not support the following Oracle data types:

- Anydata

- Anydataset
- Anytype
- Bfile
- Binary_Double
- Binary_Float
- Dburitype
- Httpuritype
- Interval day to second and Interval day(9) to second
- Interval year to month and Interval year(9) to month
- Ordaudio
- Orddicom
- Orddoc
- Ordimage
- Ordimagesignature
- Ordvideo
- Rowid
- Sdo_Geometry
- Sdo_Georaster
- Sdo_Topo_Geometry
- SI_Averagecolor
- SI_Color
- SI_Colorhistogram
- SI_Featurelist
- SI_Positionalcolor
- SI_Stillimage
- SI_Texture
- Timestamp with local time zone, Timestamp (6) with local time zone, and Timestamp (9) with local time zone
- Timestamp with time zone, Timestamp (6) with time zone and Timestamp (9) with time zone
- Uritype
- Urowid
- XML
- Xdburitype

Microsoft SQL Server

The following table describes the supported Microsoft SQL Server data types when the target is Hive or HDFS:

Data Types	HIVE	HDFS	Comments
Bigint	Yes	Yes	
Binary	Yes	No	
Bit	Yes	Yes	
Char	Yes	Yes	
Date	No	No	
Datetime	Yes	Yes	
Datetime2	Yes	Yes	
Datetimeoffset	Yes	Yes	
Decimal	Yes	Yes	You can enter a precision value up to 28.
Float	Yes	Yes	You can enter a precision value up to 28.
Image	Yes	No	
Int	Yes	Yes	
Money	Yes	Yes	
Nchar	Yes	Yes	
Ntext	Yes	Yes	
Numeric	Yes	Yes	You can enter a precision value up to 28.
Nvarchar	Yes	Yes	
Real	Yes	Yes	You can enter a precision value up to 28.
Smalldatetime	No	No	
Smallint	Yes	Yes	
Smallmoney	Yes	Yes	
Sql_variant	No	No	You must increase the precision value after you import metadata.

Data Types	HIVE	HDFS	Comments
Text	Yes	Yes	
Time	No	No	TDM moves the time along with the current date into the target.
Timestamp	No	No	TDM moves the timestamp along with the date into the target.
Tinyint	Yes	Yes	
Varbinary	Yes	No	
Varchar	Yes	Yes	

When the target is Hive or HDFS, TDM does not support the following Microsoft SQL Server data types:

- Geography
- Geometry
- HierarchyID
- Uniqueidentifier
- XML

DB2 for Linux, UNIX, and Windows

The following table describes the supported DB2 data types when the target is Hive or HDFS:

Data Types	Hive	HDFS	Comments
Bigint	Yes	Yes	
Blob (N)	Yes	No	You can perform nullification data masking operations.
Char for bit data	Yes	No	You can perform nullification data masking operations.
Char varying (Length)	Yes	Yes	
Character (N)	Yes	Yes	
Clob (N)	Yes	Yes	
Date	No	No	
Dbclob	Yes	Yes	
Decfloat (16)	No	No	

Data Types	Hive	HDFS	Comments
Decfloat (34)	No	No	
Decimal	Yes	Yes	
Double	Yes	Yes	
Double Precision	Yes	Yes	
Float	Yes	Yes	
Graphic (N)	Yes	Yes	
Integer	Yes	Yes	
Long Varchar	Yes	Yes	
Long Vargraphic	Yes	Yes	
Num	Yes	Yes	
Numeric	Yes	Yes	
Real	Yes	Yes	
Smallint	Yes	Yes	
Time	No	No	
Timestamp	No	No	
Varchar (N)	Yes	Yes	
Varchar for Bit Data	Yes	No	You can perform nullification data masking operations.
Vargraphic (N)	Yes	Yes	
XML	No	No	

Sybase ASE

The following table describes the supported Sybase ASE data types when the target is Hive or HDFS:

Data Types	Hive	HDFS	Comments
Bigdatetime	Yes	Yes	
Bigint	Yes	Yes	

Data Types	Hive	HDFS	Comments
Bigtime	Yes	Yes	TDM moves the time along with the current date into the target.
Bit	Yes	Yes	
Binary (n)	Yes	No	
Char (n)	Yes	Yes	
Datetime	Yes	Yes	
Decimal (P,S)	Yes	Yes	
Double Precision	Yes	Yes	
Float (P)	Yes	Yes	
Image	Yes	No	
Int	Yes	Yes	
Longsysname	Yes	Yes	
Money	Yes	Yes	
Nchar (N)	Yes	Yes	
Numeric (P,S)	Yes	Yes	
Nvarchar (N)	Yes	Yes	
Real	Yes	Yes	
Smallint	Yes	Yes	
Smallmoney	Yes	Yes	
Sysname	Yes	Yes	
Text	Yes	Yes	
Tinyint	Yes	Yes	
Unichar (N)	Yes	Yes	
Unitext	Yes	Yes	
Univarchar (N)	Yes	Yes	
Unsigned bigint	Yes	Yes	
Unsigned int	Yes	Yes	
Unsigned smallint	Yes	Yes	

Data Types	Hive	HDFS	Comments
Unsigned tinyint	Yes	Yes	
Varbinary (N)	Yes	No	
Varchar (N)	Yes	Yes	
Date	No	No	
Smalldatetime	No	No	
Time	No	No	TDM moves the time along with the current date into the target.

When the target is Hive or HDFS, TDM does not support the following Sybase ASE data type:

- Timestamp

Hive

The following table describes the supported Hive data types when the target is Hive or HDFS:

Data Types	Hive	HDFS	Comments
Bigint	Yes	Yes	
Binary	Yes	No	You cannot perform data masking operations on Binary data types if you use a Spark execution engine to run a Hadoop plan.
Boolean	Yes	Yes	If the source contains True or False values, TDM populates the target with values 1 or 0.
Decimal	Yes	Yes	
Double	Yes	Yes	
Float	Yes	Yes	There is a data mismatch if the source contains the value 3.3 and the target contains the value 3.299999952316280.
Int	Yes	Yes	
Smallint	Yes	Yes	

Data Types	Hive	HDFS	Comments
String	Yes	Yes	
Tinyint	Yes	Yes	

When the target is Hive or HDFS, TDM does not support the following Hive data types:

- Array
- Char
- Date
- Maps
- Struct
- Timestamp
- Varchar

HDFS

The following table describes the supported HDFS data types when the target is Hive or HDFS:

Data Types	Hive	HDFS
Number	Yes	Yes
String	Yes	Yes

When the target is Hive or HDFS, TDM does not support the following HDFS data type:

- Datetime

When you create a Hadoop plan, you must provide the date time format in the source.

Flat File

The following table describes the supported flat file data types when the target is Hive or HDFS:

Data Types	Hive	HDFS
Number	Yes	Yes
String	Yes	Yes
Datetime	No	No

INDEX

A

abort
workflows 140
Actions menu
Test Data Manager 23
Administrator view
description 28
advanced masking
description 95
parameters 96
advanced options
plan settings 128
advanced rules
assigning rule to columns 82
creating 81
description 78
advanced search
text match 24
application services
TDM 17
architecture
TDM 16
assign
advanced 83
advanced rule
assignment 83
mapplet rule 83
assigning
rules in policies to columns in an project source 82
assignments
deleting 87
multiple rules to a column 86
overriding a rule 86
rules to columns 79
auto cascades
overview 61

B

blurring
random masking 106

C

calculated masking
description 97
cascades
auto 61
value 59
CLI
ilmcmd description 150
columns
cascades 59
columns *(continued)*
computed 59
sensitivity 59
commit type
workflow properties 128
components
data masking 88
connections
Administrator view 28
configuring workflows 126
plan settings 128
TDM 18
to databases 19
to repositories 19
to services 18
constraint-based loading
workflow properties 128
constraints
adding 69
content panel
Test Data Manager 23
copying
plans 136
credit card masking
description 97
criteria
subset and masking 131
custom mapplet
masking rule 94
custom masking
parameters 95

D

dashboards
Test Data Manager 25
data discovery
nonrelational sources 57
overview 56
rules and guidelines 57
task flow 61
TDM process 20
data discovery profiles
creating 64
deleting 65
editing 65
data domain
default masking rules 45
delete 48
edit metadata 48
data domain assignments
exporting and importing 63, 64
exporting from Secure@Source 63
data domain definitions
data patterns 46

data domain definitions *(continued)*
metadata patterns 46
data domain profile
sampling options in 62
data domains
assigning to multiple columns 62
copying 47
creating 47
discovering 61
exportingand importing 51
manually adding to columns 63
options 46
overview 44
status 46
data expressions
patterns 46
data masking
assigning multiple rules to a column 86
assigning rules in policies to columns 82
components 88
creating advanced rules 81
creating rules from a mapplet 81
creating standard rules 80
delete rules 85
disabling at the plan level 131
editing rules 85
masking techniques list 91
metadata and data patterns 45
overview 76
plan 88
plans 125
repeatable output 93
rules 91
rules overview 77
standard rules 77
task flow 77
techniques 91
data masking plans
creating 135
data patterns
data domain definitions 46
TDM data domains 45
data sources
Hadoop 37
data subset
create 75
creating groups 73
editing groups 73
overview 71
plans 125
process 20
data subset plans
creating 135
databases
connections to 19
TDM 18
date values
key masking 103
random masking 108
default rule
TDM data domains 45
Define view
Test Data Manager 31
definitions
TDM repository 36
delete
ilmcmd command 152
delete table
impacted objects 40
project 40
deleting 137
deleting a table
project 40
details panel
Test Data Manager 23
dictionary connections
plan settings 128
relational and flat file 128
discover
columns 58
discover
profiles 57
tables 57
Discover view)
Test Data Manager 31
discovering
data domains 61
domains
assigning to multiple columns 62
exporting and importing 51
duplicate
data domains 47
groups 74
masking rules 85
policies 54
projects 35

E

email masking
description 98
parameters 98
encryption masking
description 100
execution engine
blaze 38
hive 38
export
ilmcmd command 153
exporting
groups 73
policy 52
TDM projects 35
expression
cascade 83
Expression Builder
overview 28
expression masking
description 100
parameters 101
rules and guidelines for 101

F

filter criteria
disabling masking by column 131
filtering data
Test Data Manager 25
foreign key
constraints 69

G

groups
- copying 74
- creating 73
- deleting 74
- editing 73
- exporting TDM 73
- importing TDM 74
- overview 72

H

Hadoop plan
- groups 136
- policies 136
- rules 136

Hadoop sources
- HDFS 37
- Hive 37

I

ilmcmd
- command line help 151

ilmcmd commands
- delete 152
- entering options and arguments 151
- export 153
- import 155
- overview 150
- search 156
- syntax for 152
- workflow 157

import
- ilmcmd command 155
- policy 52
- rule 52

importing
- groups 74
- metadata from Model repository into TDM 39
- plans 137
- profiles 66
- TDM projects 36

importing metadata
- from Model repository into Test Data Manager 39

index
- disabling in plan 128

integers
- masking large integers 84

IP address masking
- description 101

K

key masking
- date values 103
- description 102
- masking numeric values 104
- masking string values 104
- result string replacement characters 103
- source string characters 103

L

limit
- data subset plan 131

locale
- plan settings 128

logging in
- Test Data Manager 29

logs
- project 33

M

manage logs
- project 35

mandatory columns
- description 81

mapplet
- parameters 95

mapplets
- assigning rule to columns 82
- creating rules from 78

mask format
- description 102

masking
- disabling by column 131
- large integers 84

masking component
- policies 88
- rules 88

masking options
- blurring 106
- mask format 102
- range 105
- result string replacement characters 103
- seed 93
- source string characters 103

masking parameters
- advanced masking 96
- date key masking 103
- date random masking 108
- email masking 98
- expression masking 101
- numeric key masking 104
- numeric random masking 108
- random strings 109
- string key masking 104
- substitution 112

masking rule
- copying 85
- custom mapplet 94
- mapplet 83

masking rules
- delete 84, 85
- editing 85
- update 84

masking techniques
- advanced 95
- calculated 97
- credit card 97
- email 98
- encryption 100
- expression 100
- IP address 101
- key masking 102
- lists 91
- nullification 104

masking techniques *(continued)*
phone 105
random 105
Social Security number 111
substitution 111
URL 112
metadata patterns
data domain definitions 46
TDM data domains 45
Model repository
importing metadata from 39
Monitor
details 143
jobs 142–144
logs 144
tasks 143
Monitor view
description 142
Test Data Manager 27

N

nullification masking
description 104
numeric values
key masking 104
random masking 108

O

overriding a rule
data masking 86
overview
data discovery 56
data masking 76
groups 72
TDM 14
Overview view
Test Data Manager 25

P

parameters
data masking 91
patterns
data domain definitions 46
TDM data domains 45
permission view
projects 41
Persistent Data Masking
masking Social Insurance numbers 110
phone masking
description 105
plan criteria
configuring 131
plan settings
advanced options 128
connections 128
description 127
Hadoop 129
source settings 131
Workflow Executions view 139
Workflow view 138
plans
creating 135

plans *(continued)*
deleting 137
disabling masking for columns 131
importing TDM 137
properties 126
TDM projects 32
TDM workflows 125
policies
copying 54
creating 53
editing 54
exporting and importing 51
managing 53
overview 43
task flow 44
Policies view
Test Data Manager 27
policy
delete 54
export 52
preferred masking rules
overview 45
primary key
create constraint 69
processes
data subset 20
profile
TDM projects 32
profiles
data domains 61
deleting 65
editing 65
importing 66
project
list of tables 68
project risk analysis
description 26
project security
maintaining in TDM 41
projects
copying 35
creating in TDM 33
deleting in TDM 35
editing in TDM 34
exporting TDM 35
importing TDM 36
TDM 31
TDM components 32
Projects view
Test Data Manager 27
properties
plans 126
properties panel
workflow executions 140

Q

quick links
Overview view 25

R

random masking
blurring 106
description 105
masking date values 108

random masking *(continued)*
masking string values 109
numeric values 108
range 105
result string replacement characters 103
source string characters 103
range
random masking 105
regular expressions
syntax 45
repeatable output
data masking 93
reports
audit trail 147
Test Data Manager 146
repositories
connections to 19
resource format
avro 37
parquet 37
result string replacement characters
key masking 103
random masking 103
rule assignment
multiple rules to a column 86
rule assignments
description 79
overriding the rule 86
rule simulation
preview rule output 80
simulator 78
standard rule 78
rules
adding to policies 53
advanced 78
assigning multiple rules to a column 86
assigning to columns 79, 82
creating from a mapplet 81
data masking 44
deleting column assignments 87
exporting and importing 51
not available for assignment 84
overview 77
standard masking rules 77
rules and guidelines
expression masking 101

S

sampling options
data domain profile 62
search
ilmcmd command 156
seed 93
sensitivity
for ports 59
sensitivity levels
data domains 46
services
connections to 18
session
details 141
sessions
tab 141
similar value columns
deleting the group 60
description 59
similar value columns *(continued)*
editing 60
setting 60
SIN numbers
masking Social Insurance numbers 110
source settings
Hadoop 134
plan settings 131
source string characters
key masking 103
random masking 103
sources
TDM 36
special format masking
Social Insurance numbers 110
special masking techniques
credit card 97
email 98
IP address 101
phone 105
Social Security number 111
URL 112
SSN masking
description 111
standard masking rules
description 77
standard rules
creating 80
start digit
Social Insurance numbers 110
status
data domains 46
string values
key masking 104
random data masking 109
substitution masking
description 111
parameters 112
syntax
ilmcmd commands 152

T

table classification
description 68
tables
classification 68
tables per mapping
description 128
targets
TDM 36
TDM
application services 17
architecture 16
connections 18
database connections 19
databases 18
overview 14
process overview 19
repository connections 19
service connections 18
sources 36
targets 36
tools 17
TDM processes
data discovery 20

TDM projects
components 32
managing 33
owner 33
TDM repository
definitions 36
techniques
data masking 91
templates
TDM projects 32
Test Data Manager
Administrator view 28
filtering data 25
logging in to 29
Monitor view 27
Overview view 25
Policies view 27
Projects view 27
reports 146
user interface overview 22
views 23
thinkmaps
for policy assignments 146
for rule assignments 146
tools
TDM 17

U

unique key
create constraint 70

URL masking
description 112

V

value cascades
deleting 60
description 59
editing 60
setting 60

W

workflow
execute 139
workflow executions
properties panel 140
Workflow Executions view
plan settings 139
Sessions tab 141
tasks 140
Workflow view
plan settings 138
workflows
aborting 140
connections for 126
generating from plans 138
ilmcmd command 157
properties in plan settings 128
recovering 140