This product includes software licensed under the Academic Free License (http://www.opensource.org/licenses/afl-3.0.php), the Common Development and Distribution License (http://www.opensource.org/licenses/cddl1.php) the Common Public License (http://www.opensource.org/licenses/cpl1.0.php), the Sun Binary Code License Agreement Supplemental License Terms, the BSD License (http://www.opensource.org/licenses/bsd-license.php), the new BSD License (http://opensource.org/licenses/BSD-3-Clause), the MIT License (http://www.opensource.org/licenses/mit-license.php), the Artistic License (http://www.opensource.org/licenses/artistic-license-1.0) and the Initial Developer's Public License Version 1.0 (http://www.firebirdsql.org/en/initial-developer-s-public-license-version-1-0/).

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

This product includes software Copyright (c) 2013 Frank Balluffi and Markus Moeller. All rights reserved. Permissions and limitations regarding this software are subject to terms of the MIT license.

This Software is protected by U.S. Patent Numbers 5,794,246; 6,014,670; 6,016,501; 6,029,178; 6,032,158; 6,035,307; 6,044,374; 6,092,086; 6,208,990; 6,339,775; 6,482,288; 6,789,096; 6,823,373; 6,850,947; 6,895,471; 7,117,215; 7,162,643; 7,243,110; 7,254,590; 7,281,001; 7,421,458; 7,496,588; 7,523,121; 7,584,422; 7,676,516; 7,720,842; 7,721,270; 7,774,791; 8,065,266; 8,150,803; 8,166,048; 8,166,071; 8,200,622; 8,224,973; 8,271,477; 8,327,419; 8,386,435; 8,392,440; 8,453,159; 8,458,230; and RE44,478, International Patents and other Patents Pending.

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

NOTICES

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

1. THE DATADIRECT DRIVERS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NON-INFRINGEMENT.

2. IN NO EVENT WILL DATADIRECT OR ITS THIRD PARTY SUPPLIERS BE LIABLE TO THE END-USER CUSTOMER FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, CONSEQUENTIAL OR OTHER DAMAGES ARISING OUT OF THE USE OF THE ODBC DRIVERS, WHETHER OR NOT INFORMED OF THE POSSIBILITIES OF DAMAGES IN ADVANCE. THESE LIMITATIONS APPLY TO ALL CAUSES OF ACTION, INCLUDING, WITHOUT LIMITATION, BREACH OF CONTRACT, BREACH OF WARRANTY, NEGLIGENCE, STRICT LIABILITY, MISREPRESENTATION AND OTHER TORTS.
# Table of Contents

**Preface** ................................................. x

Informatica Resources. ........................................ x
  Informatica My Support Portal. .......................... x
  Informatica Documentation. .............................. x
  Informatica Web Site. .................................... x
  Informatica How-To Library. ......................... x
  Informatica Knowledge Base. ........................ xi
  Informatica Support YouTube Channel. ............... xi
  Informatica Marketplace. ................................ xi
  Informatica Velocity. ................................... xi
  Informatica Global Customer Support ................ xi

**Chapter 1: Introduction to Test Data Management** ................................. 1

Test Data Management Overview. ............................................ 1

TDM Architecture. .............................................. 2
  TDM Tools. ................................................ 3
  TDM Server. ............................................... 3
  Application Services. ........................................ 3
  TDM Databases. ............................................ 4

TDM Connections. ............................................... 4
  Service Connections. ....................................... 4
  Repository Connections. ................................... 5
  Database Connections. ...................................... 5

TDM Process. ................................................................ 5
  Create a Data Masking Policy. .............................. 6
  Create a Project and Import Metadata. .................. 6
  Discover Source Information. .............................. 6
  Define Data Masking and Data Subset Operations. .... 7
  Define Data Generation Operation. ....................... 7
  Create a Plan for Data Masking and Data Subset. .... 7
  Create a Plan for Data Generation. ....................... 8
  Monitor the Workflow. ...................................... 8

TDM Example. .................................................... 8

**Chapter 2: Test Data Manager** .................................................. 10

Test Data Manager Overview. ........................................... 10

Test Data Manager User Interface. ................................. 11
  Views. ....................................................... 11
  Search Field. .............................................. 12
  Quick Links Menu. ......................................... 12
<table>
<thead>
<tr>
<th>Chapter 5: Data Discovery</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Discovery Overview</td>
<td>41</td>
</tr>
<tr>
<td>Data Discovery Sources</td>
<td>42</td>
</tr>
<tr>
<td>Rules and Guidelines for Data Discovery Sources</td>
<td>42</td>
</tr>
<tr>
<td>Discover View</td>
<td>42</td>
</tr>
<tr>
<td>Column Properties</td>
<td>43</td>
</tr>
<tr>
<td>Sensitive Columns</td>
<td>44</td>
</tr>
<tr>
<td>Value Cascades</td>
<td>44</td>
</tr>
<tr>
<td>Auto Cascades</td>
<td>45</td>
</tr>
<tr>
<td>Data Discovery Task Flow</td>
<td>46</td>
</tr>
<tr>
<td>Primary Key Discovery</td>
<td>46</td>
</tr>
<tr>
<td>Primary Key Profile Options</td>
<td>46</td>
</tr>
<tr>
<td>Entity Discovery</td>
<td>47</td>
</tr>
<tr>
<td>Entity Profile Options</td>
<td>47</td>
</tr>
<tr>
<td>Data Domain Discovery</td>
<td>48</td>
</tr>
<tr>
<td>Data Domain Profile Sampling Options</td>
<td>48</td>
</tr>
<tr>
<td>Assigning a Data Domain to Multiple Columns</td>
<td>49</td>
</tr>
<tr>
<td>Manually Updating the Column Data Domain</td>
<td>49</td>
</tr>
<tr>
<td>Profile Management</td>
<td>50</td>
</tr>
</tbody>
</table>
Table of Contents

Creating a Primary Key Profile ......................................................... 50
Creating an Entity Profile ............................................................... 50
Creating and Running a Data Domain Profile .................................... 51
Importing a Profile ........................................................................... 52
Editing a Profile ................................................................................. 52
Deleting a Profile ............................................................................... 53
Apply the Results .............................................................................. 53
  Primary Key Results ........................................................................ 53
  Entity Discovery Results .................................................................. 54
  Data Domain Discovery Results ..................................................... 55
Project Tables .................................................................................... 56
  Table Classification ......................................................................... 56
  Constraints ...................................................................................... 57
Manually Add Keys to Tables in a Project ......................................... 58
Creating a Logical Primary Key ......................................................... 58
Creating a Logical Relationship Between Tables .................................. 59

Chapter 6: Data Subset ..................................................................... 60
Data Subset Overview ......................................................................... 60
Data Subset Task Flow ....................................................................... 61
Entities ............................................................................................. 61
  Entity Example ................................................................................. 61
  Entities View ................................................................................... 62
  Entity Map View ............................................................................. 62
  Tables View .................................................................................... 63
  Relationships View ......................................................................... 63
  Review Changes ............................................................................. 65
Entity Management ............................................................................ 66
  Creating an Entity .......................................................................... 66
  Editing an Entity ............................................................................ 66
  Exporting an Entity ....................................................................... 66
  Importing an Entity ....................................................................... 67
  Deleting an Entity .......................................................................... 67
Groups .............................................................................................. 67
  Creating a Group .......................................................................... 68
  Editing a Group ............................................................................. 68
  Exporting a Group ....................................................................... 68
  Importing a Group ....................................................................... 69
  Deleting a Group .......................................................................... 69
Templates .......................................................................................... 69
  Template Example ......................................................................... 69
  Creating a Template ...................................................................... 70
  Editing a Template ........................................................................ 70
# Table of Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deleting a Template.</td>
<td>70</td>
</tr>
<tr>
<td>Data Subset Plans and Workflows.</td>
<td>71</td>
</tr>
<tr>
<td><strong>Chapter 7: Data Masking.</strong></td>
<td>72</td>
</tr>
<tr>
<td>Data Masking Overview.</td>
<td>72</td>
</tr>
<tr>
<td>Data Masking Components.</td>
<td>72</td>
</tr>
<tr>
<td>Data Masking Task Flow.</td>
<td>73</td>
</tr>
<tr>
<td>Masking Rule Assignments.</td>
<td>74</td>
</tr>
<tr>
<td>Available Data Masking Rules.</td>
<td>74</td>
</tr>
<tr>
<td>Assigning Masking Rules to Columns.</td>
<td>75</td>
</tr>
<tr>
<td>Overriding a Masking Rule.</td>
<td>75</td>
</tr>
<tr>
<td>Assigning Multiple Masking Rules to a Column.</td>
<td>76</td>
</tr>
<tr>
<td>Assigning Default Values to Multiple Columns.</td>
<td>76</td>
</tr>
<tr>
<td>Assigning Custom Masking Rules to Columns.</td>
<td>76</td>
</tr>
<tr>
<td>Assigning Advanced Masking Rules to Columns.</td>
<td>77</td>
</tr>
<tr>
<td>Deleting Masking Rule Assignments.</td>
<td>78</td>
</tr>
<tr>
<td>Data Masking Plans and Workflows.</td>
<td>78</td>
</tr>
<tr>
<td>Masking Components in PowerCenter.</td>
<td>78</td>
</tr>
<tr>
<td>Mapplet Export.</td>
<td>78</td>
</tr>
<tr>
<td><strong>Chapter 8: Data Masking Techniques and Parameters.</strong></td>
<td>80</td>
</tr>
<tr>
<td>Data Masking Techniques and Parameters Overview.</td>
<td>80</td>
</tr>
<tr>
<td>Data Masking Techniques.</td>
<td>81</td>
</tr>
<tr>
<td>Data Masking Parameters.</td>
<td>82</td>
</tr>
<tr>
<td>Repeatable Output.</td>
<td>83</td>
</tr>
<tr>
<td>Exception Handling.</td>
<td>84</td>
</tr>
<tr>
<td>Custom Masking.</td>
<td>85</td>
</tr>
<tr>
<td>Custom Masking Parameters.</td>
<td>85</td>
</tr>
<tr>
<td>Advanced Masking.</td>
<td>85</td>
</tr>
<tr>
<td>Advanced Masking Parameters.</td>
<td>86</td>
</tr>
<tr>
<td>Advanced Masking Example.</td>
<td>87</td>
</tr>
<tr>
<td>Credit Card Masking.</td>
<td>87</td>
</tr>
<tr>
<td>Credit Card Masking Parameters.</td>
<td>88</td>
</tr>
<tr>
<td>Email Masking.</td>
<td>88</td>
</tr>
<tr>
<td>Email Masking Parameters.</td>
<td>89</td>
</tr>
<tr>
<td>Encryption Masking.</td>
<td>90</td>
</tr>
<tr>
<td>Expression Masking.</td>
<td>91</td>
</tr>
<tr>
<td>Expression Masking Parameters.</td>
<td>91</td>
</tr>
<tr>
<td>Rules and Guidelines for Expression Masking.</td>
<td>91</td>
</tr>
<tr>
<td>IP Address Masking.</td>
<td>92</td>
</tr>
<tr>
<td>Key Masking.</td>
<td>92</td>
</tr>
<tr>
<td>Mask Format.</td>
<td>92</td>
</tr>
<tr>
<td>Source String Characters.</td>
<td>93</td>
</tr>
</tbody>
</table>
Chapter 9: Data Generation ................................................. 116
Data Generation Overview. ........................................... 116
Data Generation Components. ...................................... 117
Data Generation Task Flow. .......................................... 117
Data Generation Rule Types. .......................................... 117
Standard Generation Rules. ........................................... 118
Creating a Standard Generation Rule. ............................. 118
Editing a Generation Rule. ............................................. 119
Overriding a Generation Rule. ........................................ 119
Deleting a Generation Rule. ........................................... 119
Custom Generation Rules. ............................................. 120
Creating a Custom Generation Rule. ............................... 120
Ad Hoc Generation Rules. ............................................. 120
## Table of Contents

- Creating an Ad Hoc Generation Rule. ........................................ 121
- Editing an Ad Hoc Generation Rule. ........................................ 121
- Data Generation Rule Assignments. ...................................... 121
- Available Data Generation Rules. ......................................... 122
- Assigning Standard Generation Rules to Columns. .................... 123
- Assigning Custom Generation Rules to Columns. ....................... 123
- Deleting Generation Rule Assignments. .................................. 123
- Data Generation Plans and Workflows. ................................... 124

### Chapter 10: Data Generation Techniques and Parameters ........... 125

- Data Generation Techniques and Parameters Overview. ............... 125
- Data Generation Techniques. ............................................... 126
- Data Generation Parameters. ............................................... 126
- Exception Test Data. ....................................................... 127
- Dictionary Generation. ..................................................... 128
- Dictionary Generation Parameters. ....................................... 129
- Effective Dates Generation. ............................................... 130
- Effective Dates Generation Parameters. .................................. 131
- Effective Dates Generation Example. .................................... 132
- Expression Generation. ..................................................... 132
- Expression Generation Parameters. ....................................... 133
- Random Generation. ....................................................... 133
- Date Random Generation Parameters. .................................... 134
- String Random Generation Parameters. .................................... 135
- Numeric Random Generation Parameters. ................................ 136
- Data Patterns for Random Generation. ................................... 137
- Reference Lookup Generation. ............................................ 140
- Reference Lookup Generation Parameters. ............................... 141
- Sequence Generation. ..................................................... 142
- Date Sequence Generation Parameters. ................................... 142
- Numeric Sequence Generation Parameters. .............................. 143
- Set of Values Generation. .................................................. 144
- Set of Values Generation Parameters. ..................................... 144

### Chapter 11: Plans and Workflows ...................................... 146

- Plans and Workflows Overview. .......................................... 146
- Plans and Workflows Task List. .......................................... 147
- Workflow Connections. ..................................................... 147
- Plan Components. ........................................................... 147
- Plan Settings. ................................................................. 148
- Connection Options. ....................................................... 148
- Target Options. .............................................................. 150
- Update Strategy Options. .................................................. 151
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Generation Properties</td>
<td>151</td>
</tr>
<tr>
<td>Error and Recovery Options</td>
<td>152</td>
</tr>
<tr>
<td>Advanced Options</td>
<td>153</td>
</tr>
<tr>
<td>Masking Components</td>
<td>154</td>
</tr>
<tr>
<td>Subset Components</td>
<td>155</td>
</tr>
<tr>
<td>Generation Components</td>
<td>155</td>
</tr>
<tr>
<td>Component Criteria</td>
<td>155</td>
</tr>
<tr>
<td>Filtering Data Subset Components</td>
<td>156</td>
</tr>
<tr>
<td>Disabling Masking for a Column</td>
<td>156</td>
</tr>
<tr>
<td>Source Settings</td>
<td>156</td>
</tr>
<tr>
<td>Connection Properties</td>
<td>157</td>
</tr>
<tr>
<td>Target Properties</td>
<td>158</td>
</tr>
<tr>
<td>Update Strategy Properties</td>
<td>158</td>
</tr>
<tr>
<td>Error and Recovery Properties</td>
<td>159</td>
</tr>
<tr>
<td>Source and Target Properties</td>
<td>159</td>
</tr>
<tr>
<td>Advanced Properties</td>
<td>160</td>
</tr>
<tr>
<td>Partitioning Properties</td>
<td>161</td>
</tr>
<tr>
<td>Plan Management</td>
<td>161</td>
</tr>
<tr>
<td>Creating a Data Masking and Data Subset Plan</td>
<td>161</td>
</tr>
<tr>
<td>Creating a Data Generation Plan</td>
<td>162</td>
</tr>
<tr>
<td>Creating an Ad Hoc Generation Plan</td>
<td>162</td>
</tr>
<tr>
<td>Copying a Plan</td>
<td>163</td>
</tr>
<tr>
<td>Exporting a Plan</td>
<td>163</td>
</tr>
<tr>
<td>Importing a Plan</td>
<td>163</td>
</tr>
<tr>
<td>Deleting a Plan</td>
<td>164</td>
</tr>
<tr>
<td>Workflow Generation</td>
<td>164</td>
</tr>
<tr>
<td>Generating a Workflow</td>
<td>164</td>
</tr>
<tr>
<td>Workflow View</td>
<td>164</td>
</tr>
<tr>
<td>Executing a Workflow</td>
<td>165</td>
</tr>
<tr>
<td>Workflow Executions View</td>
<td>166</td>
</tr>
<tr>
<td>Workflow Tasks</td>
<td>166</td>
</tr>
<tr>
<td>Workflow Properties Panel</td>
<td>167</td>
</tr>
<tr>
<td>Chapter 12: Monitor</td>
<td>169</td>
</tr>
<tr>
<td>Monitor Overview</td>
<td>169</td>
</tr>
<tr>
<td>Jobs</td>
<td>170</td>
</tr>
<tr>
<td>Job Details</td>
<td>170</td>
</tr>
<tr>
<td>Monitor Tasks</td>
<td>171</td>
</tr>
<tr>
<td>Logs</td>
<td>171</td>
</tr>
<tr>
<td>Chapter 13: Reports</td>
<td>172</td>
</tr>
<tr>
<td>Reports Overview</td>
<td>172</td>
</tr>
<tr>
<td>Data Masking Report</td>
<td>173</td>
</tr>
<tr>
<td>Section</td>
<td>Page</td>
</tr>
<tr>
<td>----------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Running the Data Masking Report</td>
<td>173</td>
</tr>
<tr>
<td>Plan Audit Report</td>
<td>173</td>
</tr>
<tr>
<td>Running a Plan Audit Report</td>
<td>173</td>
</tr>
<tr>
<td>Plan Detail Report</td>
<td>174</td>
</tr>
<tr>
<td>Running the Plan Detail Report</td>
<td>174</td>
</tr>
<tr>
<td>Row Count Report</td>
<td>175</td>
</tr>
<tr>
<td>Running the Row Count Report</td>
<td>175</td>
</tr>
<tr>
<td><strong>Chapter 14: ilmcmd</strong></td>
<td>176</td>
</tr>
<tr>
<td>ilmcmd Overview</td>
<td>176</td>
</tr>
<tr>
<td>Configuring ilmcmd</td>
<td>177</td>
</tr>
<tr>
<td>Running ilmcmd</td>
<td>177</td>
</tr>
<tr>
<td>Entering Options and Arguments</td>
<td>177</td>
</tr>
<tr>
<td>Syntax Notation</td>
<td>178</td>
</tr>
<tr>
<td>Delete</td>
<td>178</td>
</tr>
<tr>
<td>Delete Examples</td>
<td>179</td>
</tr>
<tr>
<td>Export</td>
<td>179</td>
</tr>
<tr>
<td>Export Examples</td>
<td>180</td>
</tr>
<tr>
<td>Import</td>
<td>181</td>
</tr>
<tr>
<td>Import Examples</td>
<td>181</td>
</tr>
<tr>
<td>Search</td>
<td>182</td>
</tr>
<tr>
<td>Search Examples</td>
<td>183</td>
</tr>
<tr>
<td>Workflow</td>
<td>183</td>
</tr>
<tr>
<td>Workflow Examples</td>
<td>184</td>
</tr>
<tr>
<td><strong>Appendix A: Glossary</strong></td>
<td>185</td>
</tr>
<tr>
<td><strong>Index</strong></td>
<td>191</td>
</tr>
</tbody>
</table>
Preface

The Informatica Test Data Management User Guide describes how to protect sensitive data and create lean nonproduction systems for test and development. It shows you how to implement data subset, data masking, data generation, and data discovery operations. This guide is written for users who use Test Data Manager with PowerCenter. It assumes knowledge of operating systems, database engines, flat files, and PowerCenter.

Informatica Resources

Informatica My Support Portal


The site contains product information, user group information, newsletters, access to the Informatica customer support case management system (ATLAS), the Informatica How-To Library, the Informatica Knowledge Base, Informatica Product Documentation, and access to the Informatica user community.

Informatica Documentation

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

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

Informatica Web Site

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

Informatica How-To Library

As an Informatica customer, you can access the Informatica How-To Library at http://mysupport.informatica.com. The How-To Library is a collection of resources to help you learn more
about Informatica products and features. It includes articles and interactive demonstrations that provide solutions to common problems, compare features and behaviors, and guide you through performing specific real-world tasks.

**Informatica Knowledge Base**

As an Informatica customer, you can access the Informatica Knowledge Base at [http://mysupport.informatica.com](http://mysupport.informatica.com). Use the Knowledge Base to search for documented solutions to known technical issues about Informatica products. You can also find answers to frequently asked questions, technical white papers, and technical tips. If you have questions, comments, or ideas about the Knowledge Base, contact the Informatica Knowledge Base team through email at [KB_Feedback@informatica.com](mailto:KB_Feedback@informatica.com).

**Informatica Support YouTube Channel**

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

**Informatica Marketplace**

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

**Informatica Velocity**

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

**Informatica Global Customer Support**

You can contact a Customer Support Center by telephone or through the Online Support. Online Support requires a user name and password. You can request a user name and password at [http://mysupport.informatica.com](http://mysupport.informatica.com).

CHAPTER 1

Introduction to Test Data Management

This chapter includes the following topics:

- Test Data Management Overview, 1
- TDM Architecture, 2
- TDM Connections, 4
- TDM Process, 5
- TDM Example, 8

Test Data Management Overview

Test Data Management (TDM) integrates with PowerCenter, PowerExchange, and Informatica applications to provide a solution 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. An organization can also create test data that does not contain sensitive data from the production database.

An organization creates multiple copies of application data to use for testing and development. Often, the organization maintains strict controls on production systems, but data security in nonproduction instances 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, data masking, and data generation in Test Data Manager.

Data discovery

Use data discovery to run sensitive field profiles to discover which columns 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 same data domain. You can run primary and foreign key profiles to discover potential primary key-foreign key constraints to define relationships between parent and child tables.
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 generation

Use data generation to create a testing environment that does not use data from the production database. Create data generation rules to define the type of data you want to generate. TDM generates data in a schema that you can use for testing.

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. To perform data generation operations, you can generate and run data generation plans in Test Data Manager. TDM integrates with PowerCenter 9.6.0.

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

TDM users have roles and privileges that determine which tasks 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.

TDM Architecture

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

The following image shows the components of TDM:

The TDM architecture consists of the tools that you use to modify the data. The application services include Model Repository Service, Data Integration Service, PowerCenter Repository Service, PowerCenter Integration Service, and Test Data Manager Service. The Model Repository Service and Data Integration Service access the Model repository and profiling warehouse. The Test Data Manager accesses the Test Data Manager Service and uses the database content from the TDM repository associated with the service.
PowerCenter services access the PowerCenter repository. The application services access the domain configuration repository that maintains connections and metadata for the domain.

**TDM Tools**

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

**Test Data Manager**
A web-based client application that you can use to configure data masking, data subset, data generation, and profiles for data discovery. You can also configure connections, manage security, and manage users.

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

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

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

**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:

**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 Profiling warehouse that you want to store data from data discovery profiles.

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

**PowerCenter Integration Service**
An application service that runs data subset, data generation, and data masking workflows. When you configure workflow options in Test Data Manager, you select the PowerCenter Integration Service that runs the workflow.

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

**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 PowerCenter 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 about data sources.

Model repository
A relational database that stores table metadata for data discovery profiles.

PowerCenter repository
A relational database that stores metadata for PowerCenter sources and targets. The PowerCenter repository also stores metadata for the workflows that you generate from plans in Test Data Manager.

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.

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

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.

PowerCenter Integration Service
A data masking, data generation, or data subset workflow requires a connection to a PowerCenter Integration Service. When you generate a workflow from a plan, select the name of the PowerCenter Integration Service to use.

PowerCenter Repository Service
When you import metadata into Test Data Manager from a PowerCenter source, the TDM Server sends a request to the PowerCenter Repository Service to extract source metadata from the PowerCenter repository and load it to the TDM repository.
You create a TDM Service on the machine that hosts the Informatica services. During the TDM Service creation, you select the PowerCenter Repository Service. The TDM Server connects to the PowerCenter repository and creates the TDM repository tables.

Repository Connections

TDM requires connections to repositories.

TDM accesses the following repositories:

- **TDM repository**
  - The PowerCenter Integration Service stores TDM components in the TDM repository.
  - A data masking, data generation, 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.

- **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.

- **PowerCenter repository**
  - When you import metadata into Test Data Manager from a PowerCenter source, the TDM Server sends a request to the PowerCenter Repository Service to copy source metadata from the PowerCenter repository into the TDM repository.

Database Connections

TDM requires a connection to a database source to perform data discovery operations and a database target to perform data generation 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. When you generate workflows from data generation plans, you must select connections for 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 PowerCenter 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. Define source and target connections in the project. Import data sources and create an empty target schema.

3. Optionally, discover information about the source data. Run profiles for data and metadata discovery to discover primary keys, entities, and 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.

   Define data generation operations. Define the tables that you want to include and assign data generation rules to columns in the target table.

5. Generate and run the workflow for data masking, data subset, or data generation.

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 PowerCenter.

**Create a Project and Import Metadata**

Create a project to organize the components for data discovery, masking, subset and generation 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. Import target metadata to perform data generation operations. You can import metadata from a PowerCenter folder or an external database source.

When you import PowerCenter source metadata, the TDM Server sends a request to the PowerCenter Repository Service to extract source metadata from the PowerCenter repository. The PowerCenter 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 primary and foreign key data, entity relationships, and data domains in source tables.

When a data source has no keys, you can run a primary key profile to identify possible primary keys. When the project contains multiple sources, you can run an entity profile to discover possible relationships between tables. Select the primary keys and the entities from the profile results to define the subset data structure.

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 entities, groups, and templates to define the tables that you want to copy to the subset database. An entity defines a set of tables that are related based on physical or logical constraints. A group defines a set of unrelated tables. A template is an optional component that contains the entities and groups.
2. Assign data masking rules to columns in the data source.
3. Create a data subset plan, and add the entities, groups, and templates to it. For each column in a parent table, you can define criteria to filter the data.
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, entities, groups, templates, and plans in the TDM repository. When you generate and run workflows from plans, the PowerCenter Integration Service runs the workflows and loads the data into the target database.

Define Data Generation Operation

To perform data generation operation, create a data generation plan. Add tables and entities to the plan.

Perform the following tasks in Test Data Manager to define the data generation operation:

1. Create entities that you want to add to the generation plan.
2. Create data generation rules and assign the rules to the columns in the target table.
3. Create a data generation plan, and add the entities and tables to the plan. Assign default data generation rules to the columns that do not have rule assignments.

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 entities, groups, and templates to it. You can define additional criteria to filter the data.
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 PowerCenter Integration Service runs the workflows and loads the data into the target database.
Create a Plan for Data Generation

Create a plan to perform data generation operations. You cannot combine a data generation operation with a data masking or a data subset operation in the same plan. You must create a separate plan for data generation.

Perform the following tasks when you want to create a data generation plan:

1. Create a data generation plan and add the tables and entities. You can define additional criteria to filter the data.
2. Enter the number of records that you want to generate.
3. Generate a workflow from the plan.
4. Run the workflow.

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.

TDM Example

An organization wants to enforce a policy to mask sensitive employee stock data in a large data processing environment.

The IT department needs test data for a new employee stock plan in an organization. The organization must ensure that the sensitive data is not compromised in the test data. The test database must contain representative data from the various application environments, including employee personal data, salary data, stock purchases, and job information. The organization uses TDM to establish and enforce a policy for creating the data in the test environment.

The organization completes the following steps:

1. Create a policy. The compliance officer determines the type of employee data that should be masked. The compliance officer creates an Employee_Stock policy.
2. Define data domains. The compliance officer defines data domains to group similar fields for data masking. For example, the data contains columns called Employee_Salary, Yearly_Salary, and Salary_History. All columns that contain "Salary" in the name belong to the same data domain. All columns in the same data domain can receive the same data masking rules.
3. Define data masking rules. The compliance officer creates data masking rules to mask the employee data. For example, the compliance officer masks employee names with substitution masking from a dictionary. The compliance officer applies random masking to the salary columns. He applies Social Security masking to Social Security numbers.
4. Define a project. A project developer defines an Employee_Stock project and imports the data sources to the project. The project developer performs all the data subset, data profiling, and data masking configuration in the project.
5. Run a profile for data discovery. The project developer runs a profile for data discovery. The profile identifies sensitive columns in the source tables and it populates the data domains that the compliance officer defined in the policy.

6. Create table relationships. The database does not contain primary and foreign keys. The project developer runs a profile for primary keys and entities to find relationships between tables. The project developer examines the primary key profile results and the entity profile results to create relationships. The project developer creates logical primary and foreign keys in the tables. In some cases, the project developer selects an entity to use from the profile results.

7. Create entities and groups for data subset. With the constraints in place, the project developer can create entities in an Employee_Stock project. An entity defines a set of related source tables based on constraints. The project includes the Employee, JobHistory, Salary, and Employee_Stock tables. The project developer also creates a group in the project. A group defines unrelated tables to include in the test database. The group includes a table called Stock_History.

8. Approve or reject profile job results. The compliance officer reviews the results and approves or rejects the column assignments to the data domains.

9. Verify all sensitive fields are masked. The compliance officer reviews reports that describe what source data is masked in the project.

10. Create a plan to run data subset and data masking. The project developer creates one plan to run the data masking and subset operations in a workflow. The project developer adds the entities and groups to the plan to define which data to copy to the subset database. The project developer adds the Employee_Stock policy to the plan to define how to mask the data. When the project developer runs a workflow from the plan, the PowerCenter Integration Service runs the workflow and loads the masked data into the subset database.

11. The compliance officer validates the results in the subset database.
Test Data Manager

This chapter includes the following topics:

- Test Data Manager Overview, 10
- Test Data Manager User Interface, 11
- Overview View, 13
- Policies View, 15
- Projects View, 15
- Monitor View, 16
- Administrator View, 16
- Expression Builder, 17
- Logging In to the Test Data Manager, 18

Test Data Manager Overview

Test Data Manager is a web-based user interface that you use to complete data discovery, subset, masking, and generation 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, data generation rules, and data domains. The compliance officer assigns the data masking rules to data domains.

A developer uses Test Data Manager Projects view to define a project and import the data sources to the project. The developer runs profiles to discover primary keys, entities, and data domain assignments. The developer creates entities for data subset operations and assigns rules to columns for data masking and data generation 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, data generation, 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:

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 data masking policies that you can apply to projects.

**Projects**

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

**Monitor**

View the status of jobs that import sources or perform data subset, data masking, data profiling, or data generation 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, rules, and entities that the project contains. When you search for a 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.

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, masking, and generation. You can use the Quick Links menu to access shortcuts to create a generation rule, 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.
Overview View

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

The following image shows the Overview view:

Dashboards

The Overview view shows the TDM dashboards. The dashboards are summary reports about projects, data masking objects, data generation 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 entities, groups, and templates 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, masking rules, and generation rules, in the last 10 accessed projects. The Project Objects tab displays the number of subset objects, including entities, groups, and templates, 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 nine 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.

Plan Execution

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

You can select the plan component, masking, subset, or generation, 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, generation 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, masking, and generation rule assignments in the repository in a pie chart. Masking rules and generation rules are displayed separately.

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 and generation 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.

The following image shows the Policies view:

![Policies View Image]

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, data subset, and data generation 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. Run data generation operations to generate 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 PowerCenter workflow from within a project.
The following image shows the Projects view:

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, data subset, and data generation operations. You can review the TDM Server job log to investigate problems. You can also stop jobs and recover workflows in the Monitor view.

The following image shows 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 generation rule settings and 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 and set default staging and storage connections and create storage tables.
- Create connections to application services and source or target databases.
- Add and manage dictionaries.

The following image shows the Administrator view:

---

**Expression Builder**

Use the Expression Builder to build an expression using PowerCenter 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.
- Configure expressions in an ad hoc generation rule.
- Configure advanced expressions in plans.

The Expression Builder shows a list of PowerCenter 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:
The following image shows the **Expression Builder** where you configure expressions:

![Expression Builder Image]

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

---

**Logging In to the Test Data Manager**

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

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

1. In the address bar of a web browser, enter the Test Data Manager URL.
   - Use the following format if Transport Layer Security is enabled:
     
     ```plaintext
     https://hostname:portnumber/tdm/
     ```
   - Use the following format if Transport Layer Security is not enabled:
     
     ```plaintext
     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:

   ```plaintext
   http://TXW1779:6643/tdm/
   ```
The Login dialog box of the Test Data Manager appears.

2. Enter the user name and password.
   Select the security domain. If the Informatica domain is configured to use LDAP authentication, the default security domain Native.

3. Click Login.
   The Test Data Manager appears.
To log out of the Test Data Manager, click Logout.
Projects Overview

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

The following image shows a project in Test Data Manager:

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, data masking, and data generation operations. You can import a source from the PowerCenter 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 primary keys, the table relationships, and the data domains in the source data.

Define

Define data masking, data subset, and data generation 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 entities, groups, and templates. When you need to perform data generation, assign the generation rules to the columns in the tables of the project.

Execute

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

Monitor

View the status of data source import jobs, profiling jobs, workflow generation jobs, data subset jobs, data masking jobs, and data generation jobs.

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:

<table>
<thead>
<tr>
<th>Component</th>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Source</td>
<td>Data subset</td>
<td>Defines the input data that you want to transform.</td>
</tr>
<tr>
<td></td>
<td>Data discovery</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data masking</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data generation</td>
<td></td>
</tr>
<tr>
<td>Entity</td>
<td>Data subset</td>
<td>Defines a set of related tables based on physical or logical constraints.</td>
</tr>
<tr>
<td></td>
<td>Data generation</td>
<td>Define an entity for a data subset or a data generation operation. Create an entity in the Define view of a project.</td>
</tr>
<tr>
<td>Group</td>
<td>Data subset</td>
<td>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.</td>
</tr>
<tr>
<td>Plan</td>
<td>Data masking</td>
<td>Defines data subset, data masking, or data generation operations. Generate a PowerCenter workflow from a plan. Define a plan in the Execute view of a project.</td>
</tr>
<tr>
<td></td>
<td>Data subset</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Data generation</td>
<td></td>
</tr>
<tr>
<td>Policy assignments</td>
<td>Data masking</td>
<td>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.</td>
</tr>
<tr>
<td>Profile</td>
<td>Data discovery</td>
<td>Suggests the primary keys, relationships, and data domains in a source based on source data and metadata. Define a profile in the Discover view of a project.</td>
</tr>
</tbody>
</table>
## Component | Operation | Description
--- | --- | ---
Rule assignments | Data masking | Defines which data masking or generation rules apply to a source column.
 | Data generation | 
Template | Data subset | A reusable component that contains the entities and groups that you want to subset based on common parameters. Create a template in the Define view of a project.

### 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.

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

You can export a projects 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, masking, and generation 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the project.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the project.</td>
</tr>
<tr>
<td>PowerCenter Repository</td>
<td>The name of the PowerCenter repository to store the repository folder.</td>
</tr>
<tr>
<td>Folder</td>
<td>The name of the project folder in the repository. Default is the project name. You can choose another folder in the repository.</td>
</tr>
<tr>
<td>Owner</td>
<td>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.</td>
</tr>
</tbody>
</table>

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 entities or groups in the project.
4. In the Edit dialog box, change project options. The following table describes the project options that you can change:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the project.</td>
</tr>
<tr>
<td>Description</td>
<td>The description of the project.</td>
</tr>
<tr>
<td>Folder</td>
<td>The name of the project folder in the repository. Default is the project name. You can choose another folder in the repository.</td>
</tr>
<tr>
<td>Owner</td>
<td>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.</td>
</tr>
</tbody>
</table>

5. Click Save.

Deleting a Project

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

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.

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 PowerCenter repository or from a source database. To perform data generation operations, you must import target metadata into the TDM repository.

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 PowerCenter Integration Service passes the row and the column to the target without applying the condition.

### TDM Source Definitions

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

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

### PowerExchange Nonrelational Sources

You can perform TDM operations on nonrelational sources that you define in PowerExchange.

You first need to define datamaps in PowerExchange and import them into PowerCenter. Create the policies, projects, and operations in TDM. Import the PowerCenter sources into a project in TDM.

You can perform TDM operations on the following PowerExchange nonrelational sources:

- Adabas
Importing Data Sources

Import data sources from PowerCenter 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:
   • PowerCenter Repository. Import metadata from PowerCenter.
   • 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. 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.
6. Click Next.
7. Choose one of the following options:
   • If you selected PowerCenter Repository, select the PowerCenter 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.
8. Click Next.
9. Select the tables to import. You can filter the tables by data source, table name, or table description.
10. Click Next.
11. 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.
12. Click Finish.
   If the import is running, view the progress of the import job in the Monitor view. After the job finishes, access the imported metadata through the Data Sources details view.
Chapter 4

Policies

This chapter includes the following topics:

- Policies Overview, 26
- Policies View, 27
- Policies Task Flow, 27
- Data Masking Rules, 27
- Data Domains, 31
- Policy Packs, 34
- Import and Export, 37
- Linking Business Glossary Terms to Global Objects, 38
- Policy Management, 39

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.

The following image shows the Policies view in Test Data Manager:

![Policies View](image)

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.

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 can define different dictionary files for substitution masking rules. Dictionary files contain the sample data for substitution. 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.

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.

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.
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. On the Configure Source Details page, to use source data from a connection:
   a. Select the connection and owner from the lists.
   b. To select a table, type the table name, or click Browse to select the table from a list.
   c. 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.
   d. Click OK to return to the Rule Simulator tab.
6. To enter sample data:
   a. Enter the data in the data fields in the Configure Source Details window. Use the buttons to add or delete rows. You can enter a maximum of 100 rows.
   b. Click OK to return to the Rule Simulator tab.
7. To use default sample data:
   a. Click Copy default data.
   b. Click OK to return to the Rule Simulator tab.
8. On the Rule Simulator tab, click Go to start the simulation.
    View the original source values and the masked values in the Rule Simulator tab.

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 85

**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** area 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.
   - Do not enter masking properties for input columns.
8. To enter more input columns, click **Add Input Column** in the **New Masking Rule** dialog box.
9. Add temporary variable columns
   - You must apply an expression, masking rule, or dependent column for each variable column you define.
   - If you apply a masking rule, configure the input column to create the variable column from.

**PowerCenter Mapplet Rules**

You can create rules from a PowerCenter 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 data source.

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

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 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.

**Creating a Rule From a PowerCenter Mapplet**

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

Export the mapplet to an XML file from PowerCenter 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 datatype to mask.
5. Select the Mapplet masking type.
6. Browse for the XML file that contains the 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.

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 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 \geq 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](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]{3}-[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. Click **Next** to add regular expressions that filter columns by column name. 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 to add.

12. Click **OK**.

13. Enable one rule as the default 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**.

### 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 available in the TDM/Content/Policies folder. Import the TDM policy packs in Test Data Manager. 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.

### 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:

<table>
<thead>
<tr>
<th>Data Domain Name</th>
<th>Default Rule</th>
</tr>
</thead>
<tbody>
<tr>
<td>Age Domain</td>
<td>Age Rule</td>
</tr>
<tr>
<td>Birth Date Domain</td>
<td>Birth Date Rule</td>
</tr>
<tr>
<td>Birth Place Domain</td>
<td>Birth Place Rule</td>
</tr>
<tr>
<td>Country Domain</td>
<td>Country Rule</td>
</tr>
<tr>
<td>Credit Card Domain</td>
<td>Credit Card Numbers Rule</td>
</tr>
<tr>
<td>Drivers License Number Domain</td>
<td>Drivers License Rule</td>
</tr>
<tr>
<td>Email Address Domain</td>
<td>Email Address Rule</td>
</tr>
<tr>
<td>First Name Domain</td>
<td>First Name Rule</td>
</tr>
<tr>
<td>Full Address UPPER Domain</td>
<td>Full Address UPPER Rule</td>
</tr>
<tr>
<td>Full Name Domain</td>
<td>Full Name Rule</td>
</tr>
<tr>
<td>Gender Domain</td>
<td>Gender Rule</td>
</tr>
<tr>
<td>Grades Domain</td>
<td>Grades Rule</td>
</tr>
<tr>
<td>IP Address Domain</td>
<td>IP Address Rule</td>
</tr>
<tr>
<td>Job Position Domain</td>
<td>Job Position Rule</td>
</tr>
<tr>
<td>Last Name Domain</td>
<td>Last Name Rule</td>
</tr>
<tr>
<td>Organization Name Domain</td>
<td>Organization Name Rule</td>
</tr>
<tr>
<td>Passport Domain</td>
<td>Passport Rule</td>
</tr>
<tr>
<td>Phone Domain</td>
<td>Phone Number Rule</td>
</tr>
<tr>
<td>Salary Domain</td>
<td>Salary Rule</td>
</tr>
<tr>
<td>School Name Domain</td>
<td>School Name Rule</td>
</tr>
<tr>
<td>Social Security Number Domain</td>
<td>Social Security Number Rule</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Policy Pack</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Number Domain</td>
<td>Account Number Rule</td>
</tr>
<tr>
<td>Birth Date Domain</td>
<td>Birth Date Rule</td>
</tr>
<tr>
<td>Certificate License Number Domain</td>
<td>Certificate License Number Rule</td>
</tr>
<tr>
<td>Device Identifier Serial Number Domain</td>
<td>Device Identifier Serial Number Rule</td>
</tr>
<tr>
<td>Email Address Domain</td>
<td>Email Address Rule</td>
</tr>
<tr>
<td>First Name Domain</td>
<td>First Name Rule</td>
</tr>
<tr>
<td>Health Plan Beneficiary Number Domain</td>
<td>Health Plan Beneficiary Number Rule</td>
</tr>
<tr>
<td>IP Address Domain</td>
<td>IP Address Rule</td>
</tr>
<tr>
<td>Last Name Domain</td>
<td>Last Name Rule</td>
</tr>
<tr>
<td>Medical Record Number Domain</td>
<td>Medical Record Number Rule</td>
</tr>
<tr>
<td>Phone Domain</td>
<td>Phone Number Rule</td>
</tr>
<tr>
<td>Primary Account Number Domain</td>
<td>Primary Account Number Rule</td>
</tr>
<tr>
<td>Social Security Number Domain</td>
<td>Social Security Number Rule</td>
</tr>
<tr>
<td>State Domain</td>
<td>State Rule</td>
</tr>
<tr>
<td>Unique Identifier Domain</td>
<td>Unique Identifying Number Rule</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Policy Pack</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Account Number Domain</td>
<td>Account Number Rule</td>
</tr>
<tr>
<td>Birth Date Domain</td>
<td>Birth Date Rule</td>
</tr>
<tr>
<td>Credit Card Domain</td>
<td>Credit Card Rule</td>
</tr>
<tr>
<td>Expiration Date Domain</td>
<td>Expiration Date Rule</td>
</tr>
<tr>
<td>First Name Domain</td>
<td>First Name Rule</td>
</tr>
<tr>
<td>Last Name Domain</td>
<td>Last Name Rule</td>
</tr>
</tbody>
</table>

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.

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 for 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. Click **Finish** to import the file.

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

**Linking Business Glossary Terms to Global Objects**

You can link business terms in the Informatica Business Glossary to TDM global objects such as policies, data domains, masking rules, and generation rules. A business term includes the business definition and examples of how to use the business term. Use this to store information that users can look up from one central location. When you link a business term to a TDM object, you can view the TDM object when you select the business term in the Analyst tool.

View business terms linked to objects from the **Policies** view. You must configure an Analyst Service in the TDM Service properties to view or link business terms with TDM objects.

When you select an object from the **Policies** view, a list of business terms linked to the object appears in the Linked Business Terms tab. The list displays the term, the name of the glossary that the term belongs to, and a description.

You can add and remove business term links to objects from the **Policies** view. Click the business term link in the Linked Business Terms tab to view the business term in the Analyst tool.

If you delete a TDM object that has a business term linked to it, you delete the link to the business term. If you edit the name of a linked TDM object, the edit reflects in the glossary. If you delete a business term from a glossary, you delete the object link to the business term.

**Linking a Business Term to an Object**

You can link business terms to an object on which you have Manage privileges.

Perform the following steps to link a business term to a TDM global object.

1. To access the **Policies** view, click **Policies**.
2. Select the policy, data domain, masking rule, or generation rule that you want to link a business term to.
3. Click **Add** on the **Linked Business Terms** tab. The **Link Asset to Business Term** page opens.
4. Select the glossary from which you want to select a business term. The list of business terms in the glossary appears.
5. Select the terms that you want to link to the object and then click **OK**. The business terms appear in the list of linked business terms.
Removing a Business Term Link to an Object

You can remove a business term link to objects on which you have Manage privileges. You can remove the link from Test Data Manager or from the Business Glossary UI.

Perform the following steps to remove a business term link to a TDM global object from Test Data Manager.

1. To access the Policies view, click Policies.
2. Select the policy, data domain, masking rule, or generation rule that you want to edit. The list of business terms linked to the object appears in the Linked Business Terms tab.
3. Select the business term that you want to remove.
5. Click Yes to confirm the deletion. The application deletes the link to the object and updates the Linked Business Terms tab.

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.

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 you want to open.
3. On the Policy tab, click 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 you want to add to the policy.
8. Click OK to select the data domains.
9. Click Save.

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 created 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.
4. Confirm that you want to delete the policy.
    Test Data Manager deletes the policy.

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.
This chapter includes the following topics:

- **Data Discovery Overview, 41**
- **Data Discovery Sources, 42**
- **Discover View, 42**
- **Column Properties, 43**
- **Data Discovery Task Flow, 46**
- **Primary Key Discovery, 46**
- **Entity Discovery, 47**
- **Data Domain Discovery, 48**
- **Profile Management, 50**
- **Apply the Results, 53**
- **Project Tables, 56**

**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 and the entities for the data subset database. You can apply profiling results instead of manually configuring data subset entities or manually applying a data masking rule to one column at a time.

You can run the following types of profiles:

**Entity profile**

Determines relationships between tables. When you run the profile, the Data Integration Service evaluates the structure of the source tables and identifies potential table relationships. Select entities from the profile results. TDM adds the logical constraints in the TDM repository, but does not apply these constraints in the source tables. When you generate a workflow from a data subset plan, TDM uses the constraints to create subset data with referential integrity.

**Primary key profile**

When you run the profile, the Data Integration Service evaluates the structure of the source tables and identifies potential primary keys. The primary key profile includes the percentage of unique values in a
column or combination of columns. Select primary keys from the profile results. TDM adds the primary keys in the TDM repository, but does not add them to the source tables.

**Data domain profile**

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 run data domain profiles and primary key profiles at the same 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
- IBM DB2
- Sybase
- Teradata

Use ODBC to connect to Sybase and Teradata 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 create profiles, run the profiles, and export the results. You can import the results to the TDM repository with Test Data Manager.

- Any table that you import to the repository from a connection 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, manage primary key-foreign key relationships, configure profiles, and update columns for masking in the Discover view.
The following image shows the Discover view in a Project:

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 disable physical primary key columns when you do not need the composite primary keys or entity relationships. 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 and possible primary key-foreign key relationships between tables.

**Columns**

View the results of a data domain profile. Maintain data domain assignments 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:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Column name.</td>
</tr>
<tr>
<td>Description</td>
<td>Optional port description.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>------------------------</td>
<td>---------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Datatype</td>
<td>Column datatype.</td>
</tr>
<tr>
<td>Domain</td>
<td>Name of the data domain that the column belongs to.</td>
</tr>
<tr>
<td>Restricted</td>
<td>The column in the database is a restricted column. The PowerCenter Integration Service does not write the value of a restricted column to a target.</td>
</tr>
<tr>
<td>Sensitive</td>
<td>Indicates whether a column contains sensitive data.</td>
</tr>
<tr>
<td>Similar Value Columns</td>
<td>Shows the number of columns that are related to this column in a value cascade.</td>
</tr>
</tbody>
</table>

### 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.

### 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 profiles for data discovery to find primary keys and entity relationships between tables in the source data. 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 type of profiling you want to perform. You can choose to run a primary key profile, an entity profile, or a 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 primary key profile results, the entity profile results, and data domain profile results.
8. Select and approve the results that you want to use for data masking and data subset operations.

Primary Key Discovery

To discover possible keys in source tables, create a primary key profile.

To identify potential data quality issues, create a primary key profile that discovers non-unique rows. A primary key discovery profile is also useful when the primary key is a combination of columns that uniquely defines a row. Non-conforming rows might contain duplicate information. After you discover potential keys in a table, you can add the keys to the source table in a project.

For example, an Employee table contains an EmplID column. The value of the column is 100% unique. A primary key profile identifies the column as a possible primary key that you can add to the table in the project.

Primary Key Profile Options

When you run a primary key profile, the Data Integration Service infers primary key candidates from the source tables. To limit the candidate results, configure inference options.

A primary key profile identifies the columns and combinations of columns that meet a specific conformance level. You can edit the maximum number of columns to combine for unique key identification.
The following table describes the options for primary key profiles:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Primary Key Rows</td>
<td>The maximum number of columns that the profile can combine to form a primary key. Default is 3.</td>
</tr>
<tr>
<td>Max Rows</td>
<td>The maximum number of rows to profile. Default is 1,000.</td>
</tr>
<tr>
<td>Conformance Criteria</td>
<td>The minimum percentage or maximum number of key violations that the profile allows. Default is a minimum percentage of 99.</td>
</tr>
</tbody>
</table>

**Entity Discovery**

To discover data relationships between source tables, create an entity profile. Entity discovery shows the tables that are potential entities.

The entity profile discovers entity relationships in the source data. An entity consists of related parent and child tables. The entity profile returns a list of possible entities by comparing column names in the source tables. You can select entities from the profile results to add to the project. When you run a data subset operation, the subset database structure and the data in the subset is determined by the entities that you define for the subset.

For example, the Employee table, the Employee_Detail table, and an Employee_Address table contain an EmpID column. The entity profile suggests that the Employee table could be an entity with two direct relationships. The Employee table has two child tables. When you view the profile results, you select the Employee table as an entity. When you run a data subset operation, you can create a filter on EmpID to retrieve the data for employees with an EmpID less than 2000. The data subset operation filters data for each table in the entity relationship.

**Entity Profile Options**

TDM infers foreign key candidates from the source tables. To limit the candidate results, specify inference options.

Define options to configure how the entity profile matches the datatypes of the primary keys and the foreign keys of the entity table.
The following table describes the entity profile options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datatypes used in comparisons</td>
<td>The datatype to use for comparisons. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Inferred Datatypes. Compares the primary key and the foreign key based on</td>
</tr>
<tr>
<td></td>
<td>the datatypes inferred by TDM.</td>
</tr>
<tr>
<td></td>
<td>- Documented Datatypes. Compares the primary key and the foreign key based on</td>
</tr>
<tr>
<td></td>
<td>the datatypes defined for the source column.</td>
</tr>
<tr>
<td>Comparison case sensitivity</td>
<td>Performs case-sensitive comparisons.</td>
</tr>
<tr>
<td>Trim spaces before comparison</td>
<td>Trims spaces before comparing values. Select one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- None. Does not trim leading or trailing spaces.</td>
</tr>
<tr>
<td></td>
<td>- Both. Trims both leading and trailing spaces.</td>
</tr>
<tr>
<td></td>
<td>- Leading. Trims leading spaces.</td>
</tr>
<tr>
<td></td>
<td>- Trailing. Trims trailing spaces.</td>
</tr>
<tr>
<td>Maximum foreign keys returned</td>
<td>Maximum number of matched foreign key results that you want to return.</td>
</tr>
<tr>
<td>Minimum confidence percent</td>
<td>Minimum percentage of foreign key values that match the inferred datatype.</td>
</tr>
<tr>
<td>Regenerate signature</td>
<td>Determines whether the data profiling warehouse should clear the existing</td>
</tr>
<tr>
<td></td>
<td>object information before the Data Integration Service runs the profile.</td>
</tr>
</tbody>
</table>

**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 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:

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

### 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.
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 Primary Key Profile

Create a primary key profile to show a list of possible primary keys that you can add to tables that do not have keys. Create and run primary key profile in the Discover view.

Create a primary key profile if the source data does not contain primary and foreign key relationships.

1. Open the project and click the Discover view.
2. Click the Profile view.
3. Click Actions > New Profile to create a profile.
4. In the New Profile dialog box, enter the profile name and description. Choose to create a primary key profile.
5. Select the tables to profile and click OK.
6. Click Next.
7. In the Select Sampling Options pane, click Override default inference options to change the primary key options.
8. Click Save.
9. Click Actions > Run Profile.

Creating an Entity Profile

Create an entity profile if the source data does not contain primary and foreign key relationships. The entity profile creates a list of possible parent tables that you can define for data subset operations. Create and run entity profiles in the Discover view.

1. Open the project and click the Discover view.
2. Click the Profile view.
3. Click Actions > New Profile to create a profile.
4. Select the tables to profile and click OK.
5. Click Next.
6. In the Select Sampling Options pane, click Override default inference options to change the entity options.
The following table describes the entity options that you can update:

<table>
<thead>
<tr>
<th>Entity Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datatypes used in comparison</td>
<td>Determines how Test Data Manager derives datatypes to compare in the tables. Choose one of the following options: - Inferred datatypes. Datatype derived from the values in the column. - Metadata. The datatype described in the schema.</td>
</tr>
<tr>
<td>Comparison case sensitivity</td>
<td>Indicates if the foreign key comparison is case-sensitive or not. If you choose case-sensitivity, profile job performance might be affected.</td>
</tr>
<tr>
<td>Trim spaces before comparison</td>
<td>Trims leading spaces, trailing spaces, or leading and trailing spaces.</td>
</tr>
<tr>
<td>Minimum confidence percent</td>
<td>Minimum percentage of rows without key violations that the profile allows for foreign key discovery.</td>
</tr>
<tr>
<td>Regenerate Signature</td>
<td>Determines whether the data profiling warehouse should clear the existing object information before the Data Integration Service runs the profile.</td>
</tr>
</tbody>
</table>

7. Click **Save**.

8. Click **Actions > Execute** to run the profile.

---

**Creating and Running 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**.

**Importing a Profile**

You can create a profile in Informatica Developer, run the profile, and export the profile results to an XML file. Import the XML file into the TDM repository from Test Data Manager. The profile results appear in Test Data Manager. You can also import a profile from another TDM repository.

1. Open the project and click the **Discover | Profile** view.
2. Click **Actions > Import Profile**.
   
   The **Import Profiles** dialog box appears.
3. Enter a profile name and description, and select a profile type.
4. Click **Choose File** to navigate to the XML file exported from Informatica Developer and click **OK**.
5. Click **Finish**.
   
   The imported profile appears in the **Discover** view.

**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. When you run primary key profiles and entity profiles, you can change the options.

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 **Relationship Sampling** tab or the **Sampling** tab depending on the profile type.
4. On the **General** tab, you can edit the name and description of the profile.
   
   You cannot change the profile type.
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 **Relationship Sampling** tab, select to override default inference options. Edit the options for finding primary key and foreign keys in the source data.
7. 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.
8. 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.

Apply the Results

After the profile run, you must close the profile in the Test Data Manager and open it again to view the profile results. Click separate views for the primary key results, the entity profile results, and the data domain profile results.

Primary Key Results

The primary key results is a list of possible primary keys that you can add to tables that do not have keys. Use the unique percentage value in the profile results to determine which column or combination of columns that you might add as a primary key for a table.

The following image shows sample primary key profile results:

![Primary Key Profile Results](image)

To view primary key results, close the profile and open it again to display the results. Click the **Profile | Primary Key** view. Examine the primary key profile results. The primary key candidates include the percentage of unique values in a column or combination of columns.

The following table describes the fields in the primary key results:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table Name</td>
<td>The name of the table that contains the column with the primary key candidate.</td>
</tr>
<tr>
<td>Column Name</td>
<td>The name of the column or combination of columns that the Data Integration Service infers to be a primary or unique key.</td>
</tr>
</tbody>
</table>
### Apply the Primary Key Profile Results

You can choose primary key candidates from the profile results and add the keys to tables in the project.

You can configure a status that indicates whether you want to accept the primary key candidate or reject the primary key candidate. When you approve a primary key, the Test Data Manager adds the key to the table in the TDM repository.

Select a column and change the status from the Actions menu. Select Verify, Approve, or Reject.

When you approve the primary key candidate, the Test Data Manager adds the primary key to the table. When you reject the primary key candidate, it does not affect the subset operation.

Use the Verified column to track results as you finish the review. The Verified column does not affect data subset operations.

### Entity Discovery Results

The Entity view shows the list of possible entity relationships in the source data. You can select a column in the profile results and see other columns that might contain the same data. You can select entities from the profile results to add to a data subset operation.

To view entity discovery results after you run the profile, close the profile and open it again. Click the Profile | Entity view.

Select an entity to view the constraints in the Constraints view. Test Data Manager shows the parent columns, the child columns, and the relationships. Approve the constraints to add them to an entity.

You can approve each entity that you want to add to a project.
The following image shows the results of an entity discovery profile:

![Entity Discovery Profile](image)

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

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

**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 entities or 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 know which tables are master tables when you determine which tables are driving tables for entities. You might want to configure more filter criteria for tables that contain transactions or logs.

Constraints

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. To perform data generation operation, you can use select conditional constraints. You can configure the expression to apply constraints at the table level.

When you create a data entity, you select one table for the entity. The table that you select is the entity driving table. Test Data Manager adds other tables to the entity based on the constraints for the driving table.

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 in an entity:

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.

RELATED TOPICS:

• "Entities" on page 61

Major and Minor Constraint Severity

When you define a constraint, configure the constraint severity as major or minor. The severity determines the scope of the data that a data subset receives based on the constraints. You can change the severity to change the final output of a subset operation. The constraint severity level does not affect the structure of an entity. You can change the severity level for a constraint after you create an entity.

Major constraints are constraints that move in both directions – child to parent and parent to child. When you want the parent table to select additional child records that are related to the parent, assign a major constraint between two tables.

When you define a constraint, you define it on the child table. When you define a constraint with a major severity level, all of the children of the parent are included in the data subset. For example, you define a constraint between Customer and Leads. Customer is the parent table. You create a filter on Leads that returns only high quality leads. The subset receives the customer for a high quality lead, but it also receives all the leads for the customer.
When you define a major constraint, some child rows might not conform to the filter criteria. In the Customer and Leads example, the data subset receives all of the leads for the customer instead of just the high quality leads.

Minor constraints are equivalent to foreign key constraints. They are unidirectional constraints, from child to parent. Minor constraints do not move in the reverse direction from parent to child. When you do not want the parent table to select additional child table records that are related to the parent, assign a minor constraint between two tables.

When you define a constraint with a minor severity level, the parent row is included in the data subset, but the data subset does not receive all of the children of the parent row. The data subset receives the high quality leads and the customers for the high quality leads.

If the schema has constraints that form cycles, or if the schema has objects that have more than one major parent constraint, TDM performs recursive evaluation to calculate the subset. TDM needs a staging connection to perform recursive evaluation. You must therefore create a staging connection to perform subset operations on tables with cyclic or tending to cyclic constraints.

The child record selected for the subset operation pulls in related parent records from both parent tables. The parent records select all the related child records from the child table. As more records are selected in the child table, the related parent records are selected again. This is because the records selected by one parent in the child table might refer to parent records in another parent that is not a part of the subset. This process continues in a loop until there are no more related records to select.

**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 manually add keys, you do not need to run a primary key discovery profile.

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

You can add the following types of keys:

**Logical Primary Key**

You can add one column or column combination as the primary key for a table. Use the unique percentage value in the profile results to determine which column to add as the primary key. A unique percentage of 100 indicates that all values in the column are unique.

**Constraint**

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.

**Creating a Logical Primary Key**

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 Columns.
   The columns in the select table appear.
4. Select the checkbox for the column that you want to update.
5. Click Create Logical Primary Key.
   Test Data Manager adds a logical manual primary key to the column.
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:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Constraint identifier.</td>
</tr>
<tr>
<td>Constraint Type</td>
<td>Select the constraint type as Foreign Key. You can select conditional constraint type and enter an expression to perform data generation operation.</td>
</tr>
<tr>
<td>Severity Level</td>
<td>Choose one of the following values:</td>
</tr>
<tr>
<td></td>
<td>- Major. The relationship includes the parent table and it includes the other children of the parent.</td>
</tr>
<tr>
<td></td>
<td>- Minor. The relationship includes the parent table but it does not include the other children of the parent.</td>
</tr>
<tr>
<td>Parent Table</td>
<td>Choose the parent table to establish the foreign key with.</td>
</tr>
<tr>
<td>Enable Constraint</td>
<td>Enable the foreign key relationship.</td>
</tr>
</tbody>
</table>

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.
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.

To create a data subset, you define one or more of the following components:

**Entity**

Defines a set of tables that are related based on physical or logical constraints. Create an entity when you need to copy data from related tables to a subset database. When you create an entity, select parameters to filter data in columns that you want to include in the subset database. You can manually create an entity or use an entity from profile results.

**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.

**Template**

Contains entities and groups. A template must contain at least one entity. Create a template when you want to reuse a set of entities and groups.

After you define the data subset components, add them to a plan. You can generate a workflow from the plan. The workflow loads a subset of the source data into a target database.
Data Subset Task Flow

To define a data subset, add entities, groups, and templates to a project. Create a data subset plan and add the entities, groups, and templates from the project to the plan. To load data into the subset database, generate and start a workflow from the plan.

Complete the following high-level tasks to create the components for the data subset plan:

1. Create a project that contains one or more sources.
2. Create an entity to select related parent and child tables to include in the subset database.
3. Create filter criteria that defines what data to include in the subset database.
4. Optionally, create a group to select unrelated tables to include in the subset database.
5. Optionally, create a template that contains the entities and groups.
6. Create a data subset plan and add the entities, groups, and templates to it. Define parameters to filter the data in the columns. For example, you can define a parameter to filter a month column to contain only October.
7. Generate a workflow from the plan.
8. Start the workflow.
9. Monitor the progress of the workflow.
10. In the target database, verify that the correct data was copied.

Entities

An entity defines a structure for copying related data to the subset database.

An entity consists of a driving table and related tables. A driving table is the starting point for defining relationships between tables in the entity. Test Data Manager defines the tables that relate to the driving table based on physical and logical constraints. You can add tables that have cyclic or circular relationships within the table or between tables. You must add a staging connection before you can add these tables to an entity.

When you create an entity, you select a table as the driving table. Test Data Manager retrieves all related tables to the driving table based on the constraints in the driving table. Test Data Manager adds the related tables to the entity by default.

Define filter parameters for columns of tables in the entity. The filter parameters define the criteria for the subset of the data.

Related Topics:

- "Constraints" on page 57

Entity Example

An employee has an ID number of 10 and you want information about EmpID 10 from multiple tables. A table called Employee contains an EmpID column. The Employee_History and Employee_Order tables have constraints that relate them to the Employee table that contain the EmpID column.

Create an entity and define Employee as the driving table. Test Data Manager adds the Employee_History and Employee_Order tables to the entity. Define a criteria in which the EmpID value must be equal to 10.
Create a plan, generate the workflow, and start the workflow. The target database contains all the data for EmpID 10. The Employee table has one row with EmpID 10, the Employee_History table has multiple rows of history for EmpID 10, and Employee_Order has all the orders for EmpID 10.

Entities View

Create, edit, and delete entities in the Define | Data Subset view of a project.

When you create an entity, you can configure it with the following views:

Entity Map

The Entity Map view is a graphical representation of the tables in the entity and the relationships between the tables. You can disable and enable relationships, and edit filter criteria for each table in the Entity Map view. To select multiple rows in the Relationships view, you can click the required links in the graphical representation of the entity.

Tables

The Tables view contains a list of the tables in the entity. You can edit filter criteria for each table in the Tables view.

Relationships

The Relationships view shows a list of relationships in the entity. For each constraint, the view shows the parent table and the child table in the relationship. You can disable and enable relationships between tables in the Relationships view. To select specific links in the Entity Map view, you can select the corresponding rows in the Relationships view.

Entity Map View

The entity map is a graphical representation of the tables in an entity and the relationships between the tables.

When you create an entity, you define a driving table for the entity. The driving table is a starting point for determining the structure of the entity. Test Data Manager determines the rest of the tables in the entity based on the constraints that include the driving table.

If you drag the entity map panel to the left, a Table Properties panel appears that includes information about a selected table or constraint.

You can select any table in the map to edit the filter criteria. You can click a link to edit the constraint between the tables. You can disable relationships between tables from the map. When you select links in the Entity Map view, the corresponding rows are selected in the Relationships view.

Entity Map Example

You might define the driving table as Employee. Employee has a primary key, EmpID. Employee has three child tables that have the EmpID foreign key. The EmployeeBankDetails child table has an AccountNum constraint that relates EmployeeBankDetails to a BankAccounts table. When you create an entity with Employee as the driving table, Test Data Manager might create an entity map similar to the following map:

![Entity Map Example Diagram]

The driving table icon contains a star in the entity map.
To add a filter criteria for Employee, click the Employee table in the entity map. Edit the filter criteria from the Table Properties panel.

To edit the constraint between the EmployeeDetails and Employee table, click the link between the tables.

When you click the links between the Employee, EmployeeBankDetails, and BankAccounts table, the corresponding rows are selected in the Relationships view.

Choose Tables to View from the Entity Map

You can change which tables you view in the entity map. You can view all the tables or you can view the tables by their relationships with a selected table. By default the entity map shows enabled relationships. You can choose to show disabled relationships.

Select one of the following views:

Show immediate relationships
  Shows the tables that have constraint relationships with a selected table. You can view one level of parent and child tables from the selected table.

Show all parents
  Shows all the parent tables of a selected table.

Show all children
  Shows all the child tables of a selected table.

Show all
  Show all tables in the entity.

Show excluded
  Includes disabled relationships in the entity map.

Tables View

The Tables view contains a list of the tables in the entity. You can edit filter criteria for each table from the Tables view.

The Tables view shows the table name, description, and the name of the database that contains the table. The status field indicates whether the entity includes or excludes the table. When you disable a relationship in the entity and the table is not related to any table in the entity map, the table is excluded from the data subset database.

Relationships View

The Relationships view shows a list of the relationships between tables in the entity. You can disable and enable relationships from the Relationships view.

The Relationships view shows the parent table, the child table, and the database names for each table. The view shows the name of the constraint that defines the relationship between the tables. When a table has more than one relationship in an entity, the table appears more than once in the view.

The Status field indicates whether the relationship is enabled or disabled.

When you select a row in the Relationships view, the corresponding link is selected in the Entity Map view.
Disabling and Enabling Relationships in the Relationships View

You can enable or disable a relationship in the Relationships view.

1. In Test Data Manager, open a project.
2. Navigate to the Define | Data Subset view.
3. Select the entity you need to change.
4. Click the Relationships view.
5. Select the relationship to change.
6. Click Enable Relationship or Disable Relationship.
   The status changes to disabled or enabled.

Enabling a Relationship in the Entity Map

You can enable a disabled relationship from the entity map.

1. In the Entity Map view, select Show Excluded Tables.
   Test Data Manager includes disabled relationships in the entity map. The entity map shows a disabled relationship with broken lines between the objects in the relationship.
2. Click the broken line between the tables in the relationship you want to enable.
3. Click Enable Relationship.
   The broken line becomes a solid line when Test Data Manager enables the relationship.

Disabling a Relationship in the Entity Map

You can disable relationships between entity objects.

1. In Test Data Manager, open a project.
2. Navigate to the Define | Data Subset view.
3. Click the entity to open the Entity Map view.
4. Click the link between the parent and child icons in the map to select the link.
   Test Data Manager highlights the link.
5. Click Disable Relationship.
   The link appears as a dotted line.
6. Click the **Relationships** view.

The relationship status is disabled.

**Review Changes**

When you modify a constraint that affects a table in an entity, you must update the entity to reflect the change.

For example, you create an entity that contains a table called EMP. After you create the entity, you add a constraint relationship between EMP and another table. When you add the constraint, the additional table does not appear in the entity unless you update the entity relationships.

When you view the list of entities in a project, Test Data Manager flags the entity as having a change impact. Open the entity and click **Review Changes** to see the relationship changes that impacted the entity. You can select a relationship and accept the change to update the entity.

**Note:** After you review and update the entity, you must indicate that you are finished reviewing the changes. In the **Entity** view, click **Actions > Mark Changes Review as Complete**. The Review Changes notification no longer appears for the entity.

The following image shows the **Review Changes** dialog box:
Entity Management

Manage entities to perform data subset operations.

You can create, edit, and delete entities in the Define | Data Subset view. You can export and import entities.

Creating an Entity

When you create an entity, you select the driving table. Test Data Manager retrieves the related tables based on the constraints. You can add filter criteria to the driving table to filter source data when you create the data subset database. You can disable relationships between tables in the entity.

Before you create an entity, perform data discovery to identify constraints. Add constraints to the source to define the child tables.

1. Open a Project. To access the Define view, click Define.
2. In the Define view, click Data Subset.
3. Click Actions > New > Entities.
4. In the New Entity dialog box, enter a name and optional description for the entity.
5. Click Select Table to select a driving table from the list.
6. Click Save.

Test Data Manager shows a diagram of the relationship between the tables in the entity. You can view a list of the tables or a list that shows the relationships between the tables.

7. To create a subset of the data based on filter criteria, click Edit in the Criteria pane.
   The Edit Criteria dialog box displays a list of the columns in the entity.
8. To filter data, select the check box for each column you want to filter.
9. If you want to define the filter expression in the entity, select an operator from the list to filter the data.
   You can indicate a filter condition in the entity, but define the expression in the plan.
10. If you enter a filter condition, enter the value to complete the filter expression.
11. Click Save for each filter criteria that you create. If you define multiple filters in an entity, the filter conditions act as "AND."
12. Select the rows and click Save to exit the Edit Criteria dialog box.

Editing an Entity

You can edit an entity to change the general properties, relationships, and filter criteria for tables.

1. On the Entity tab, click Edit.
2. On the General tab, edit the entity name, description, and status.
3. On the Entity Criteria tab, select columns and enter filter criteria.
4. Click Save.

Exporting an Entity

You can export an entity 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 entity to export.
3. Click the **Define** view.
   The list of entities, groups, and templates in the project appear.
4. Select an entity to export.
   Use the checkbox to select the entity.
5. Click **Actions > Export**.
6. Choose the name and path of the XML file to create.
   The default name is a string that contains "Entity_" and the current date and the time.
7. Click **OK**.

**Importing an Entity**

You can import an entity 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 entity into.
   Do not open the project.
3. Click **Actions > Import**.
4. Browse for the XML file that contains the entity to import.
   The XML file has a default name similar to `Entity_130315081854.xml`.
5. To import the entity, click **Finish**.

**Deleting an Entity**

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

To delete an entity, first remove it from all plans and templates.

1. In the **Define | Data Subset** view in the project, click an entity name to select it.
   Do not open the project.
2. Click the **Actions > Delete**.
3. In the **Delete Confirmation** message box, click **OK**.

**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 or entity. 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.

### 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. Click **Save**.

### Editing a Group

You can edit a group to change its description, tables, and tables.

1. In a project, click **Define | Data Subset**.
   - The **Groups** panel shows a list of the groups in the project.
2. Click a group name to select a group.
   - Do not open the group.
3. Click **Edit**.
   - The **Edit** dialog box appears.
4. On the group **General** tab, edit the group name or the description.
5. On the **Tables** tab, change the tables in the group.
6. Click **Save**.

### Exporting a Group

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 entities, groups, and templates in the project appear.
4. Select group to export.
   - Use the checkbox to select the entity.
5. Click **Actions > Export**.
6. Choose the name and path of the XML file to create.
   - The default name is a string that contains "Group_" and the current date and the time.
7. Click **OK**.
Importing a Group

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 similar to Group_130315081854.xml.
5. To import the group, click Finish.

Deleting a Group

If you do not use a group in a project, you can delete it. You cannot delete a group that you assigned to a plan or template.

To delete a group, first remove it from all assigned plans and templates.

1. In a project, click Define | Data Subset.
   The Groups panel shows a list of the groups in the project.
2. Click a group name to select a group.
   Do not open the group.
3. Click Actions > Delete.
   The Edit dialog box appears.
4. In the Delete Confirmation message box, click Yes.

Templates

A template is a container for data subset groups and entities. A template must contain at least one entity.

Use a template to add multiple groups and entities to a plan or to add groups and entities to multiple plans.
You can add groups and entities to more than one template.

You create, edit, and delete templates in the Define | Data Subset view in an project.

Template Example

You want information from multiple tables about different employees. You create a template that gets information about an employee. The template contains an entity that uses multiple tables to gather information about an employee with ID number 10. The template also contains a group that contains unrelated configuration tables.

After you create the template, you can change it to get information about a different employee. For example, you can change the employee to ID number 25 and get the same information about employee ID 25.
Creating a Template

Create a template to contain the entities and groups to add to a data subset plan. You can map template parameters to the entity parameters.

1. On the Define | Data Subset view in the project, click Actions > New > Templates.
2. In the Create Template dialog box, enter a name for the group and optional description for the group.
3. In the Create Template dialog box, enter a name for the template and optional description for the template. Click Next.
4. Click Add Entities to add entities to the template.
5. On the Add Entities dialog box select the checkbox of the entities you want to add. Click Next.
6. Click Next.
   The template parameter dialog box appears. The right panel of the dialog box shows the parameters for entities.
7. To add a template parameter, click Add.
   The template parameters dialog box appears.
8. Enter the template parameter name, description, and datatype.
   a. Add a parameter for each of the column datatypes in the main table. For each parameter that you want to add, enter a name, description, and the datatype.
   Each parameter that you add appears in the list.
   b. To assign parameters to columns in the main table, link the template parameter to the entity parameter. Select the template parameter, select an entity parameter, and click the link button.
9. Click Next to add Groups to the template or click Finish to exit the template wizard.
10. If you chose to enter groups, select one or more groups, and click Finish.

Editing a Template

Edit a template to add or remove entities, groups, and parameters.

1. On the Define | Data Subset view in the project, click a template to open it.
2. Click Edit.
   The Edit template dialog box displays the General, Entities, Parameter Map, or Group.
3. Click a tab, update the template and click Save.

Deleting a Template

You can delete template if it is not assigned to a plan.

To delete a template, first remove it from the assigned plan.

1. On the Define | Data Subset view in the project, select a template.
2. Click Delete.
3. In the Delete Confirmation message box, click Yes.
Data Subset Plans and Workflows

A data subset plan contains the entities, groups, and templates that represent the tables and the data that you want to load to the subset database. Generate a workflow from the plan to create PowerCenter mappings to run the subset operation.

When you start a workflow for the plan, the data subset operation starts.
Data Masking

This chapter includes the following topics:

- Data Masking Overview, 72
- Data Masking Components, 72
- Data Masking Task Flow, 73
- Masking Rule Assignments, 74
- Data Masking Plans and Workflows, 78
- Masking Components in PowerCenter, 78

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 datatype 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. When you select a rule in a data masking plan, you assign a column to the rule. 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 PowerCenter Integration Service performs the masking operation.

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:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>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.</td>
</tr>
<tr>
<td>Column sensitivity</td>
<td>A sensitive column contains sensitive data. Configure column sensitivity to mark columns that you want to mask.</td>
</tr>
<tr>
<td>Data domain</td>
<td>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.</td>
</tr>
<tr>
<td>Plan</td>
<td>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.</td>
</tr>
<tr>
<td>Policy</td>
<td>Defines the data masking rules, the data to mask, and the masking parameters for a source.</td>
</tr>
<tr>
<td>Rule</td>
<td>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.</td>
</tr>
<tr>
<td>Value cascade</td>
<td>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.</td>
</tr>
</tbody>
</table>

**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.

Complete the following high-level steps to create the components that you need in a data masking plan:

1. Create rules, data domains, and policies to define the masking techniques and masking parameters.
2. Create a project that contains one or more data sources.
3. Add policies to the project. When you add a policy to a project, the project receives the data domains and the rules that the policy contains.
4. Add additional rules to the project. When you add additional rules to a project, the project receives the masking rules.
5. Assign rules to columns in the source. You can assign rules from a policy or data domain. You can assign the default rules from data domains to multiple columns at a time. Manually assign advanced rules and mapplet rules to columns.
6. Create a plan and add data masking components to the plan. Generate a workflow from the plan. Monitor the progress of the workflow in the **Monitor** view or in the **Plan** view.
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 image shows the Define | Data Masking view in a project:

![Define | Data Masking view in a project](image)

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

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the table.</td>
</tr>
<tr>
<td>Owner</td>
<td>Name of the database schema.</td>
</tr>
<tr>
<td>Columns</td>
<td>Name of the column to mask.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Data type of the column to mask.</td>
</tr>
<tr>
<td>Domain</td>
<td>Name of the domain that you assigned to the column either from a data domain discovery or a manual assignment.</td>
</tr>
<tr>
<td>Sensitive</td>
<td>Indicates if the column is a sensitive column. Value is Yes or No.</td>
</tr>
<tr>
<td>Similar Value Columns</td>
<td>Indicates that the column is configured with other columns in a cascade. The column shows the number of other columns in the cascade.</td>
</tr>
<tr>
<td>Policy</td>
<td>Policy name that the column is assigned to.</td>
</tr>
<tr>
<td>Masking Rule</td>
<td>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.</td>
</tr>
<tr>
<td>Override</td>
<td>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.</td>
</tr>
</tbody>
</table>

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 PowerCenter 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.

**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.

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.

### Assigning Masking Rules to Columns

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.

### Overriding a Masking Rule

You can override rule parameters when 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 Multiple Masking Rules to a Column

You can assign multiple masking rules to a column. To assign more than one rule, create a duplicate row in the **Define | Data Masking** view and update the rule.

1. Open the **Define | Data Masking** view for a project.
2. Select a column that you want to assign multiple rules to.
   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.

### Assigning Default Values to Multiple Columns

You can assign the default data domain rules to multiple columns at the same time.

1. Open the **Define | Data Masking** view for a project.
2. Select the columns that you want to assign default values to.
3. Click **Rule Assignment**.
   The **Rule Assignments** dialog box appears.
4. 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.
5. Click **Default Assignments**.
   The Test Data Manager updates each column with the default rule.
6. Click **Save**.

### Assigning Custom Masking Rules to Columns

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.
   **Note:** 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.

### Assigning Advanced Masking Rules to Columns

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.
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 Delete.
   Test Data Manager prompts you to confirm deleting the assignment.
3. Click OK to delete the rule assignment from the column.

Data Masking Plans and Workflows

A data masking plan contains columns and policies that represent the tables and data that you want to mask. The plan also contains the rules to use to mask the data. A data masking plan contains at least one port with an assigned rule or one policy. You generate a workflow from a plan to perform the masking operations.

When you start a workflow for the plan, the data masking operation starts.

RELATED TOPICS:

- "Creating a Data Masking and Data Subset Plan" on page 161
- "Generating a Workflow" on page 164

Masking Components in PowerCenter

When you generate and run a workflow from a data masking plan, the PowerCenter Integration Service creates the masking components that are required to run a data masking workflow. The PowerCenter Integration Service stores these components in the PowerCenter repository.

The PowerCenter Integration Service creates the following components to perform the data masking operations:

- Mappings
- Transformations
- Mapplets
- Sessions and workflows

Important: If you edit the components that are generated from Test Data Manager in the PowerCenter Client, the session might generate unexpected results.

Mapplet Export

To implement PowerCenter masking techniques that are not available in Test Data Manager, you can create and export masking mapplets from PowerCenter to an XML file. You can import mapplets as rules into Test Data Manager.

Include one or more Data Masking transformations in the mapplet. The mapplet can be a single or multiple port mapplet. The transformations must be passive transformations. For a single input and output port
mapplet, you can configure the same port as both the input and output port, and you can complete this action for multiple ports at a time when you assign a mapplet that you imported as a rule to columns in a project.

When you assign a mapplet to a columns, assign the source and target columns to the mapplet. The TDM Server applies the PowerCenter transformation within the mapplet to the columns. You can also export rules as PowerCenter mapplets.
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 PowerCenter 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Masking Technique</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Advanced</td>
<td>Applies masking techniques to multiple input and output columns. You can create an expression to combine multiple columns. You can mask all datatypes.</td>
</tr>
<tr>
<td>Credit card</td>
<td>Applies a built-in mask format to disguise credit card numbers. You can mask the string datatype.</td>
</tr>
<tr>
<td>Email address</td>
<td>Applies a built-in mask format to disguise email addresses. You can mask the string datatype.</td>
</tr>
<tr>
<td>Encryption</td>
<td>Applies encryption to the source data. You can assign an encryption method to mask the data with. You can encrypt string data types.</td>
</tr>
<tr>
<td>Expression</td>
<td>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.</td>
</tr>
<tr>
<td>IP address</td>
<td>Applies a built-in mask format to disguise IP addresses. You can mask the string datatype.</td>
</tr>
<tr>
<td>Key</td>
<td>Produces deterministic results for the same source data, masking rules, and seed value. You can mask date, numeric, and string datatypes.</td>
</tr>
<tr>
<td>Mapplet</td>
<td>Applies masking rules from a PowerCenter 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.</td>
</tr>
<tr>
<td>Nullification</td>
<td>Replaces a column of data with a null value. You can mask all datatypes.</td>
</tr>
<tr>
<td>Phone</td>
<td>Applies a built-in mask format to disguise phone numbers. You can mask the string datatype.</td>
</tr>
</tbody>
</table>
Masking Technique | Description
--- | ---
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 datatype that you need to mask. Some masking techniques are not available for all datatypes.

When you configure masking parameters, the parameters for the masking technique appear in the left panel of the wizard. You can configure exception handling parameters in the right panel of the screen.
The following image shows data masking parameters that appear when you configure a Substitution data masking rule:

![Data Masking Parameters Image]

**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 PowerCenter 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, if you have the same Cust_ID column in four tables, and you want all of them to output the same masked values, apply the same seed value when you mask each column.
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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Preprocessing Expression</td>
<td>Optional. Expression to define changes to make to the data before masking. Click <strong>Edit</strong> to configure the preprocessing expression.</td>
</tr>
<tr>
<td>Post processing Expression</td>
<td>Optional. Expression to define changes to make to the masked data before saving the data to the target. Click <strong>Edit</strong> to configure the post processing expression.</td>
</tr>
<tr>
<td>Null and Empty Spaces</td>
<td>The default behavior for handling null values or empty columns in the source data. Choose one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Constant. Mask the data with a constant value. Enter the value to use.</td>
</tr>
<tr>
<td></td>
<td>- Default server value. Mask the source column with the default value from the defaultValue.xml file.</td>
</tr>
<tr>
<td></td>
<td>- Log error and continue. Log an error in the PowerCenter session log and continue processing.</td>
</tr>
<tr>
<td></td>
<td>- Treat as value. Treat null values or spaces as a valid source value. Mask the space or null value with a valid value.</td>
</tr>
<tr>
<td></td>
<td>- Ignore. Do not mask the null or empty space.</td>
</tr>
<tr>
<td>Error Handling</td>
<td>The default behavior for handling errors in the source data. Choose one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Constant. Mask the data with a constant value. Enter the value to use.</td>
</tr>
<tr>
<td></td>
<td>- Default server value. Mask the source column with the default value from the defaultValue.xml file.</td>
</tr>
<tr>
<td></td>
<td>- Log exception and continue. Log an exception in the PowerCenter session log and continue processing.</td>
</tr>
<tr>
<td></td>
<td>- Ignore and continue. Do not mask the null or empty space.</td>
</tr>
<tr>
<td></td>
<td>- Error. Log an error in the PowerCenter session log and stop processing.</td>
</tr>
<tr>
<td>Trim Leading or Trailing Spaces</td>
<td>Trims the leading and trailing spaces from source data. When you enable this option the following source fields are the same: &quot;Jones&quot;, &quot;Jones&quot;, &quot;Jones&quot;.</td>
</tr>
</tbody>
</table>

84 Chapter 8: Data Masking Techniques and Parameters
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 PowerCenter 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 rule.

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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Name</td>
<td>The name of an input, output, or a variable column within a mapplet. 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.</td>
</tr>
</tbody>
</table>
| Column Type | The column type. The mapplet can contain the following types of columns:  
- Input. Receives the source data.  
- Variable. A temporary column that contains intermediate values.  
- 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 column in the source. 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 29

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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Column Name</td>
<td>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.</td>
</tr>
<tr>
<td>Column Type</td>
<td>The column type. You can configure the following types of columns:</td>
</tr>
<tr>
<td></td>
<td>- Input. Receives the source data.</td>
</tr>
<tr>
<td></td>
<td>- 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.</td>
</tr>
<tr>
<td></td>
<td>- 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.</td>
</tr>
<tr>
<td>Datatype</td>
<td>The datatype of the column.</td>
</tr>
<tr>
<td>Precision</td>
<td>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.</td>
</tr>
<tr>
<td>Scale</td>
<td>Number of digits to the right of the decimal point in a number.</td>
</tr>
<tr>
<td>Mandatory</td>
<td>Indicates if you must assign the column to a column in the source. Applies to input and output columns.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression</td>
<td>An expression to apply to the variable column. You can create the expression in the Expression Builder.</td>
</tr>
<tr>
<td>Masking Rule</td>
<td>Applies a masking rule to the input column and writes the results in the variable column. You can enter the following parameters.</td>
</tr>
<tr>
<td></td>
<td>- Condition. Defines whether an input column should be masked or not. If the condition is true, the PowerCenter Integration Service masks the column.</td>
</tr>
<tr>
<td></td>
<td>- Rule. The data masking rule to apply to the input column.</td>
</tr>
<tr>
<td></td>
<td>- Override properties. You can change the data masking rule parameters if the rule owner enabled the rule to be overridden.</td>
</tr>
<tr>
<td></td>
<td>- 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.</td>
</tr>
</tbody>
</table>
### 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, variable columns, and output 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:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Column Type</th>
<th>Masking Technique</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>FIRST_NAME</td>
<td>INPUT</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>LAST_NAME</td>
<td>INPUT</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>FIRST_MASKED</td>
<td>VARIABLE</td>
<td>Substitution on FIRST_NAME</td>
<td>-</td>
</tr>
<tr>
<td>LAST_MASKED</td>
<td>VARIABLE</td>
<td>Substitution on LAST_NAME</td>
<td>-</td>
</tr>
<tr>
<td>FULL_NAME</td>
<td>OUTPUT</td>
<td>Expression</td>
<td>FIRST_MASKED</td>
</tr>
</tbody>
</table>

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 datatype.

### Credit Card Masking

Credit card masking applies a built-in mask format to disguise credit card numbers. The PowerCenter 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 PowerCenter 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 PowerCenter Integration Service applies a default credit card number mask when the source data is not valid.

The PowerCenter Integration Service does not mask the six-digit Bank Identification Number (BIN) at the beginning of the number. For example, the PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatable</td>
<td>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.</td>
</tr>
<tr>
<td>Seed</td>
<td>Starting point for creating repeatable output. Enter a number between 1 and 999. Default seed value is 1.</td>
</tr>
<tr>
<td>Keep card issuer</td>
<td>The PowerCenter 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.</td>
</tr>
<tr>
<td>Replace card issuer</td>
<td>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.</td>
</tr>
</tbody>
</table>

### 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatable output</td>
<td>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.</td>
</tr>
<tr>
<td>Seed</td>
<td>Starting point for creating repeatable output. Enter a number between 1 and 999. Default is 1.</td>
</tr>
<tr>
<td>Standard</td>
<td>Replaces an email string with characters. The standard email masking technique does not create realistic domain or user names.</td>
</tr>
<tr>
<td>Advanced</td>
<td>Replaces an email string with name and domain values from a dictionary or mapping.</td>
</tr>
</tbody>
</table>
| 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.  
  - 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dictionary for columns</td>
<td>The dictionary to use when you choose to form an email from a dictionary column. Choose a dictionary that contains first and last names.</td>
</tr>
<tr>
<td>Expression</td>
<td>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.</td>
</tr>
<tr>
<td>First name column</td>
<td>The dictionary column that contains the first name of the email address.</td>
</tr>
</tbody>
</table>
### Domain Name Masking Parameters

The following table describes the parameters that define how to mask the domain name:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Constant</td>
<td>A constant value to mask the email address with. Each email address receives the same domain name.</td>
</tr>
<tr>
<td>Random</td>
<td>Indicates whether to use a flat file or relational dictionary.</td>
</tr>
<tr>
<td>Domain lookup dictionary</td>
<td>The dictionary file to use from the imported sources.</td>
</tr>
<tr>
<td>Domain name column</td>
<td>The column that contains a dictionary name in the domain lookup dictionary.</td>
</tr>
</tbody>
</table>

### 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 create or change data. Mask all datatypes 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.

You can concatenate data from multiple columns to create a value for another column. For example, you need to create a login name. The source has first name and last name columns. Mask the first and last name from lookup files. In the rule, associate another column called Login. In the Login column, configure an expression to concatenate the first letter of the first name with the last name:

```
SUBSTR(FIRSTNM,1,1) || LASTNM
```

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 datatype. The PowerCenter 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.

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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatable Output</td>
<td>Determines if the masked value should persist for a given column value. You must enter a seed value when you enable repeatable output.</td>
</tr>
<tr>
<td>Alphanumeric Seed</td>
<td>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.</td>
</tr>
<tr>
<td>Expression</td>
<td>Accepts PowerCenter expressions and performs a calculation based on values within a single row.</td>
</tr>
</tbody>
</table>

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, and the storage table does not exist, the Integration Service substitutes the source data with default values.
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 PowerCenter Integration Service masks the network number within the network range.

The PowerCenter Integration Service does not mask the class and private network address. The PowerCenter 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seed</td>
<td>A start number that enables the PowerCenter Integration Service to return deterministic data. You can mask the date, numeric, and string datatypes.</td>
</tr>
<tr>
<td>Mask Format</td>
<td>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.</td>
</tr>
<tr>
<td>Source String Characters</td>
<td>The characters in the source string that you want to mask. You can mask the string datatype.</td>
</tr>
<tr>
<td>Result String Replacement Characters</td>
<td>Substitutes the characters in the target string. You can mask the string datatype.</td>
</tr>
</tbody>
</table>

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 PowerCenter Integration Service replaces each source character with any character. If the mask format is longer than the input string, the PowerCenter Integration Service ignores the extra characters in the mask format. If the mask format is shorter than the source string, the PowerCenter Integration Service does not mask the characters at the end of the source string.

When you choose any mask format character except "R" you must configure at least the Source String Characters option or the Result String Replacement Characters option. Otherwise the mask format is not valid.
**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:

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Alphabetical characters. For example, ASCII characters a to z and A to Z.</td>
</tr>
<tr>
<td>D</td>
<td>Digits. From 0 through 9.</td>
</tr>
<tr>
<td>N</td>
<td>Alphanumeric characters. For example, ASCII characters a to z, A to Z, and 0-9.</td>
</tr>
<tr>
<td>X</td>
<td>Any character. For example, alphanumeric or symbol.</td>
</tr>
<tr>
<td>+</td>
<td>No masking.</td>
</tr>
<tr>
<td>R</td>
<td>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.</td>
</tr>
</tbody>
</table>

**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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mask Only</td>
<td>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.</td>
</tr>
<tr>
<td>Mask all except</td>
<td>Masks all characters in the source except for source string characters. For example, if you enter &quot;-&quot; as the source string character, every character except for &quot;+&quot; will change.</td>
</tr>
</tbody>
</table>

**Result String Replacement Characters**

Configure result string replacement characters to specify masking output.

The PowerCenter 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use only</td>
<td>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 &quot;horse&quot; might be replaced with BAcBA.</td>
</tr>
<tr>
<td>Use all except</td>
<td>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.</td>
</tr>
</tbody>
</table>

**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 PowerCenter Integration Service can mask dates between 1753 and 2400 with key masking. The PowerCenter Integration Service always generates valid dates. If the source year is in a leap year, the PowerCenter Integration Service returns a year that is also a leap year. If the source month contains 31 days, the PowerCenter Integration Service returns a month that has 31 days. If the source month is February, the PowerCenter 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 PowerCenter 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Seed</td>
<td>A start number that enables the PowerCenter 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.</td>
</tr>
<tr>
<td>Mask Format</td>
<td>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.</td>
</tr>
</tbody>
</table>
### Parameter | Description
--- | ---
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 PowerCenter Integration Service does not always return unique data if the number of source string characters is less than the number of result string characters.

| Parameter | Description |
--- | --- |
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: A B C D E F |

### 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. Mask date, numeric, and string datatypes with random masking.

The results of random masking are non-deterministic. Use random masking to mask date, numeric, or string datatypes.
The following table describes the options that you can configure for random masking:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>A range of output values. The PowerCenter Integration Service returns data between the minimum and maximum values. You can configure a range for date, numeric and string datatypes.</td>
</tr>
<tr>
<td>Blurring</td>
<td>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 datatypes.</td>
</tr>
<tr>
<td>Mask Format</td>
<td>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 datatype.</td>
</tr>
<tr>
<td>Source String Characters</td>
<td>The characters in the source string that you want to mask. You can configure source string characters for the string datatype.</td>
</tr>
<tr>
<td>Result String Replacement Characters</td>
<td>Substitutes the characters in the target string. You can configure replacement characters for the string datatype.</td>
</tr>
</tbody>
</table>

**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 PowerCenter 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Blurring Type</th>
<th>Low</th>
<th>High</th>
<th>Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fixed</td>
<td>0</td>
<td>10</td>
<td>Between 66 and 76</td>
</tr>
<tr>
<td>Fixed</td>
<td>10</td>
<td>0</td>
<td>Between 56 and 66</td>
</tr>
<tr>
<td>Fixed</td>
<td>10</td>
<td>10</td>
<td>Between 56 and 76</td>
</tr>
<tr>
<td>Percent</td>
<td>0</td>
<td>50</td>
<td>Between 66 and 99</td>
</tr>
<tr>
<td>Percent</td>
<td>50</td>
<td>0</td>
<td>Between 33 and 66</td>
</tr>
<tr>
<td>Percent</td>
<td>50</td>
<td>50</td>
<td>Between 33 and 99</td>
</tr>
</tbody>
</table>

## 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:

<table>
<thead>
<tr>
<th>Character</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Alphabetical characters. For example, ASCII characters a to z and A to Z.</td>
</tr>
<tr>
<td>D</td>
<td>Digits. From 0 through 9.</td>
</tr>
<tr>
<td>N</td>
<td>Alphanumeric characters. For example, ASCII characters a to z, A to Z, and 0-9.</td>
</tr>
<tr>
<td>X</td>
<td>Any character. For example, alphanumeric or symbol.</td>
</tr>
<tr>
<td>+</td>
<td>No masking.</td>
</tr>
<tr>
<td>R</td>
<td>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.</td>
</tr>
</tbody>
</table>

If you do not define a mask format, the PowerCenter Integration Service replaces each source character with any character. If the mask format is longer than the input string, the PowerCenter Integration Service ignores the extra characters in the mask format. If the mask format is shorter than the source string, the PowerCenter 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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mask Only</td>
<td>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.</td>
</tr>
<tr>
<td>Mask all except</td>
<td>Masks all characters in the source except for source string characters. For example, if you enter &quot;-&quot; as the source string character, every character except for &quot;-&quot; will change.</td>
</tr>
</tbody>
</table>

**Result String Replacement Characters**

Result string replacement characters are a set of characters that the PowerCenter 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 PowerCenter 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:

```
A
B
C
D
E
F
```

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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use only</td>
<td>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 &quot;horse&quot; might be replaced with BAcBA.</td>
</tr>
<tr>
<td>Use all except</td>
<td>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.</td>
</tr>
</tbody>
</table>

**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 PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>The minimum and maximum values to return for the selected datetime value. The date range is a fixed variance.</td>
</tr>
<tr>
<td>Blurring</td>
<td>Masks a date based on a variance that you apply to a unit of the date. The PowerCenter 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.</td>
</tr>
</tbody>
</table>

**Numeric Random Masking Parameters**

When you mask numeric data, you can configure a range of output values for a column.

The PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>A range of output values. The PowerCenter Integration Service returns numeric data between the minimum and maximum values.</td>
</tr>
<tr>
<td>Blurring Range</td>
<td>A range of output values that are within a fixed variance or a percent variance of the source data. The PowerCenter Integration Service returns numeric data that is close to the value of the source data. You can configure a range and a blurring range.</td>
</tr>
</tbody>
</table>

**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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Range</td>
<td>The minimum and maximum string length. The PowerCenter Integration Service returns a string of random characters between the minimum and maximum string length.</td>
</tr>
<tr>
<td>Mask Format</td>
<td>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.</td>
</tr>
<tr>
<td>Source String Characters</td>
<td>The characters in the source string that you want to mask.</td>
</tr>
<tr>
<td>Result String Replacement Characters</td>
<td>Substitutes the characters in the target string.</td>
</tr>
</tbody>
</table>
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 datatypes 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.

**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 PowerCenter 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:

```
Shuffle technique switches the row values in a table. For example, A,B,C,D in a list might shuffle as B,D,C,A.
```

**Shuffle Type**

- **Random**
- **Representative**

**Seed**

- [ ] Constrained
The following table describes the parameters that you can configure for shuffle masking:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
</table>
| 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 PowerCenter 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 PowerCenter 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 PowerCenter Integration Service accesses the latest High Group List from the following location:

<PowerCenter Installation Directory>\infashared\SrcFiles\highgroup.txt

The PowerCenter 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 PowerCenter Integration Service returns deterministic Social Security numbers with repeatable masking. The PowerCenter Integration Service cannot return all unique Social Security numbers because it cannot return valid Social Security numbers that the Social Security Administration has issued.

### Substitution Masking

Substitution masking replaces a column of data with similar but unrelated data from a dictionary. Mask date, numeric, and string datatypes 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.

When you configure substitution masking, select the relational or flat file dictionary that contains the substitute values. The PowerCenter Integration Service performs a lookup on the dictionary and replaces source data with data from the dictionary. Substitution is an effective way to replace production data with realistic test 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lookup Condition</td>
<td>The column name in the source table you can refer to match with the column in the dictionary. This field is optional.</td>
</tr>
<tr>
<td>Unique Substitution Column</td>
<td>The column name in the source table to substitute with unique data. This field is optional.</td>
</tr>
</tbody>
</table>

You can substitute data with repeatable or non-repeatable values. When you choose repeatable values, the PowerCenter 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 PowerCenter 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.

### Substitution Masking Parameters

You can substitute data with repeatable or non-repeatable values.

When you choose repeatable values, the PowerCenter 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:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repeatable Output</td>
<td>Returns deterministic results between sessions. The PowerCenter Integration Service saves masked values in the storage table.</td>
</tr>
<tr>
<td>Seed</td>
<td>A start number that the PowerCenter Integration Service uses to return deterministic data.</td>
</tr>
<tr>
<td>Unique Substitution Data</td>
<td>Replaces the target column with unique masked values for every unique source column value. If you have more values in the source than in the dictionary file, the remaining source values are not masked. For security, the default is non-unique substitution.</td>
</tr>
<tr>
<td>Dictionary Information</td>
<td>Required. Configuration of the flat file or relational table that contains the substitute data values. Configure the following parameters:</td>
</tr>
<tr>
<td></td>
<td>- Dictionary. Displays the flat file or relational table name that you select.</td>
</tr>
<tr>
<td></td>
<td>- Masked Value. The column returned to the masking rule.</td>
</tr>
<tr>
<td></td>
<td>- Lookup Column. The source data column to use in the lookup.</td>
</tr>
<tr>
<td></td>
<td>- Serial Number Column. The column in the dictionary that contains the serial number.</td>
</tr>
</tbody>
</table>

**URL Masking**

URL masking applies a built-in mask format to change URL data. Mask string datatypes with URL masking.

The PowerCenter 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 PowerCenter Integration Service does not mask the protocol of the URL. For example, if the URL is http://www.yahoo.com, the PowerCenter Integration Service can return http://MgL.aijCa.Vsd/. The PowerCenter Integration Service might generate a URL that is not valid.

**Note:** The PowerCenter 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 the Test Data Manager**

Add a dictionary in the 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.

**Adding a Flat File Dictionary**

When you add a flat file dictionary, you can select a dictionary from the local system.

1. In the Administrator | Dictionaries view, click Actions > New Dictionary.
   The New Dictionary tab appears.
2. Enter a name and optional description for the dictionary. Select Flat File as the dictionary type.
3. Click Choose File to browse to the location of the dictionary. Select the dictionary file, and click Open.
4. Optionally, select to overwrite the file if a dictionary exists in the <TDM installation directory>/TDM/infa_shared/LkpFiles directory.
5. Select the file encoding type.
6. Click **Save**.
   A tab with the dictionary properties opens and the dictionary appears in the **Administrator | Dictionaries** view.

**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:

![Substitution Masking Parameters](image)

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:

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. Select in_Country.
• Input column. Select in_FirstName.

The following image shows the var_FirstName_us column properties:

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 Repeateable Output and Unique Substitution Data options, and click Save. The property appears as Yes (Overridden).
   - Lookup column. Select in_Gender.
   - Unique column. Select 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:

25. Click OK.
The `FirstName` variable column appears in the list of output columns. The following image shows the rule columns:

<table>
<thead>
<tr>
<th>Column Name</th>
<th>Dat...</th>
<th>Pr...</th>
<th>S...</th>
<th>Ma...</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>in_Country</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><code>in_FirstName</code></td>
<td>String</td>
<td>10</td>
<td>10</td>
<td>true</td>
</tr>
<tr>
<td><code>in_Gender</code></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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](image)

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:

   ![Column Properties Image]

   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:
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:

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:

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.
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:

   ![Rule Columns Image]

   ![Properties Image]

2. Click Finish.
   The rule appears in the Policies view.
Data Generation Overview

Use data generation to create realistic test data for test environments. You can define generation rules that define the logic to generate data.

Import metadata into a project to define the type of data that you want to generate. You assign generation rules that you create or default rules to the target columns to generate data based on the datatype of the column. When you create a rule, you can choose a generation technique and configure parameters to create random test data.

To implement data generation, you create a data generation plan and a workflow from the plan.

Data Generation Example

You work for an organization that sells plane tickets. You want to generate data in tables that contain customer information such as identification number, status of membership, and address. You want additional tables to store ticket details such as ticket number and flight number. To generate the data, you can perform the following tasks:

- Create data generation rules that load dictionary values such as names into the tables.
- Create random number strings for ticket numbers.
- Create a numeric sequence for identification numbers.
- Use a reference lookup for values such as airport codes.
- Create projects to import metadata, enable relationships, and create entities.
- Make rule assignments, create a plan, and run the plan to generate the data.
Data Generation Components

To perform data generation operations, assign rules to columns. Import metadata to define the columns in the target database.

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

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assignments</td>
<td>The allocation of rules to a column to generate the column data.</td>
</tr>
<tr>
<td>Plan</td>
<td>Defines data generation operations. You import the target schema metadata into the repository in the plan and configure the settings.</td>
</tr>
<tr>
<td>Rule</td>
<td>Defines the data generation technique and parameters. A generation technique defines the logic to generate the data. Generation parameters define how a generation technique in a rule generates data. You can set an override option in a rule that defines whether users can modify the generation parameters for the rule when they assign the rule to the columns in a target.</td>
</tr>
</tbody>
</table>

Data Generation Task Flow

To implement data generation operations, assign generation rules to target columns. Create a plan and add rules to the plan. Generate a workflow from the plan and run the workflow to generate data in the target database.

Perform the following high-level steps to create the components that you need in a data generation plan:

1. Create generation rules that define the generation techniques and configure the parameters. Define default rules that generate data for columns that you do not assign a generation rule to.
2. Create a project and import metadata into the project.
3. Add generation rules to the project and assign the rules to columns in the project.
4. Create a plan and add data generation components to the plan.
5. Generate and run the plan. View the progress of the workflow in the Monitor view.

Data Generation Rule Types

A data generation rule is a rule that defines how you want to generate data in the target table. When you create a data generation rule, you choose a generation technique and configure generation parameters.

You can create the following data generation rule types:
Standard

A data generation rule that uses predefined generation techniques.

Custom

A data generation rule that generates data based on PowerCenter mapplets logic.

Ad Hoc

A data generation rule that you create within a project for a column in the target table.

Standard rules and custom rules are global rules. Global rules are rules that you create in the Policies view of Test Data Manager. When you create a global generation rule in the Policies view, you specify a rule name. This rule is available for projects that you create. You assign the rule to one or more columns in the project that you want to generate data into.

Ad hoc generation rules are rules that you create in the Define | Data Generation view of a project. Ad hoc rules do not have a name, and you cannot move an ad hoc rule. You can find ad hoc generation rules within the projects. You select the target columns for an ad hoc generation rule when you create the rule. You can select a standard generation technique for an ad hoc generation rule or you can select an ad hoc generation technique. You can use reference lookup, effective dates, and expression generation techniques only in an ad hoc generation rule.

A data generation technique defines the logic that generates the data. When you create a generation rule, the techniques vary based on datatype you select. Generation parameters are options that you configure for a generation technique. For example, you can define different start dates and end dates for random date generation.

You can enable users to override generation parameters for a rule. For example, you create a rule that generates random dates from 2011 to 2012. You set the override option for the rule. When a user assigns this rule to a target column, the user can select Saturday and Sunday as exclude dates so that the rule generates random weekday dates from 2011 to 2012.

Standard Generation Rules

A standard generation rule is a global rule that applies a system-defined generation technique. A standard generation rule has one output column.

When you create a standard generation rule, you select the generation technique. You can define one generation technique in a standard rule and you can apply the rule to one column at a time. You can apply the rule multiple times to different columns.

You can use standard generation techniques to generate string, numeric, date, and binary datatypes. The techniques vary based on the datatype.

For example, you create a standard generation rule with the random date technique to generate birth dates. You can assign the rule multiple times in a project to generate birth dates in more than one column or you can use the rule in multiple projects.

Creating a Standard Generation Rule

To create a standard generation rule, you choose a generation technique based on the datatype and configure the generation parameters to apply the technique.

1. Click Policies to access the Policies view.
2. Click Actions > New > Generation 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 generation rule to.
5. Select the standard generation rule.
6. To enable users to override generation parameters for a rule, select the Override Allowed option.
7. Click Next.
8. Enter the generation parameters.
   Note: The parameters that appear depend on the Generation Technique that you select.
9. Enter the exception test data options. Configure whether to allow null or empty spaces. Configure whether to generate data that is not valid.

Editing a Generation Rule

You can edit a global generation rule.

1. Click Policies to access the Policies view.
2. Click the name of the generation rule that you want to edit.
   The rule opens in a tab.
3. Click Edit.
4. Edit the parameters that you want to change and click OK to save the rule.

Overriding a Generation Rule

After you assign a generation rule to a column, you can override the rule properties for the column. You can override rule properties for global generation rules.

1. Open a project and click Define > Data Generation.
2. Select the column that you need to override the rule properties for.
3. Click Actions > Override.
   The Generation Rule Override window appears.
4. Update the values in the generation rule properties.
5. Click Save for each column that you update.
   The Override column value shows Yes-Overridden for the column that you updated.

Deleting a Generation Rule

You can delete a global generation rule.

1. Click Policies to access the Policies view.
2. Click the box to the left of the generation rule that you want to delete.
3. Click Actions > Delete.
   The Delete Generation Rule window appears. The Delete Generation Rule window contains a warning if the rule has assignments.
4. Click OK to delete the rule.
Custom Generation Rules

A custom generation rule is a global rule that generates data based on PowerCenter mapplet logic. You can import a mapplet to create a custom generation rule. You must mark at least one input and one output column as required. Use custom rules to generate data for string, numeric, and date datatypes.

For example, an EMPLOYEES table contains columns for first name and last name. You create a custom generation rule to combine the first and last name of an employee to generate the full name. You can also create custom rule with mapplets when you want to use PowerCenter transformations, such as an SQL transformation or a Stored Procedure transformation. You can also use custom mapplets when you need to use the logic to generate records or generate output with the user defined procedure.

Creating a Custom Generation Rule

Create a custom rule using PowerCenter mapplet to generate data.

1. To access the Policies view, click Policies.
2. Click Actions > Generation Rule.
   The Rule Wizard appears.
3. Enter a name and optional description for the rule.
4. Select the Custom Rule generation type.
5. To select a mapplet from the local folder, click Browse.
6. Click Next.
   The Configure Mandatory Columns dialog box appears with a list of input columns and output columns that are available in the mapplet.
7. To mark the columns as required fields, select the columns, and click Set Mandatory.
   You must at least select one input column and one output column as required fields.
8. Click Save.
   The custom rule appears as another tab. Review and close the tab.

Ad Hoc Generation Rules

An ad hoc generation rule is a data generation rule that you create within a project. You can create an ad hoc generation rule for a column in the target table.

An ad hoc rule does not have a name. You cannot move an ad hoc rule because you use target columns to define the rule.

An ad hoc rule can have more than one target column, based on the generation technique that you select.

When you create an ad hoc rule, the generation techniques that you can select depend on the datatype of the column. To create an ad hoc rule, you can use a standard or an ad hoc generation technique. The ad hoc generation techniques include expression generation, reference lookup generation, and effective dates generation.

For example, an Airport table contains airport codes. You create a reference lookup rule on a From_City column that generates data from the Airport table and inserts it into the From_City column in the target table.
Creating an Ad Hoc Generation Rule

You can create an ad hoc generation rule in a project. The generation techniques change based on the data type of the column that you selected. The generation parameters change based on the generation technique that you select.

1. In a project, click **Define > Data Generation** to access the **Data Generation** view.
2. Select a column to assign the ad hoc generation rule to.
3. Click inside the **Generation Rule** column to view the list of rules available for the data type of the column.
5. Select the generation technique.
6. Enter the generation parameters.
   **Note:** If a Java heap space exception occurs, contact the Administrator to increase the heap size.
7. Enter the exception test data options. Configure whether to allow null or empty spaces. Configure whether to generate data that is not valid.

Editing an Ad Hoc Generation Rule

You can edit an ad hoc rule assignment to a column.

1. In a project, click **Define > Data Generation** to access the Data Generation view.
2. Select a column to edit the ad hoc rule assignment.
3. Click **Actions > Edit Generation Rule**. The **Edit Generation Rule** window appears.
4. Edit the parameters that you want to change.

Data Generation Rule Assignments

You can assign a global or an ad hoc generation rule to a column within a project.

To assign a rule, choose a rule from the list in the **Generation Rule** column or select **New Generation Rule**. If you do not select a rule, the column receives the default rule that you configure in default settings.
The following image shows the Define | Data Generation view in a project:

The following table describes the fields in the Data Generation view:

<table>
<thead>
<tr>
<th>Column</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Table</td>
<td>Name of the table.</td>
</tr>
<tr>
<td>Column</td>
<td>Name of the column of the generated data.</td>
</tr>
<tr>
<td>Data Type</td>
<td>Data type of the column of the generated data.</td>
</tr>
<tr>
<td>Is Null</td>
<td>Indicates whether the column can contain null values.</td>
</tr>
<tr>
<td>Generation Rule</td>
<td>The generation rule that you want to apply to the column. When you click inside the column, you can choose which rule to apply to the column. The rules change based on the data type of the column. Select New Generation Rule to create an ad hoc generation rule for the column.</td>
</tr>
<tr>
<td>Override</td>
<td>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.</td>
</tr>
</tbody>
</table>

Available Data Generation Rules

The Data Generation view lists the rules that you can assign to each column of the target table.

If no global rules appear for a column when you click Generation Rule, make sure that the project contains global generation rules for the column datatype.

Ad hoc generation rules are available for columns of all datatypes. Click New Generation Rule to create an ad hoc generation rule.

Note: In the Data Generation view, you must click the Save icon each time you assign a rule. If you do not save the rule assignment before you assign the next rule, the Test Data Manager discards the rule assignment.
Assigning Standard Generation Rules to Columns

Assign standard generation rules to one or more target columns to generate data in the columns.

1. In a project, click Define > Data Generation to access the Data Generation view.
2. Select a column to assign a generation rule to.
3. Click inside the Generation Rule column to view the list of rules available for the datatype of the column.
4. Select a rule.
5. Click Save for each column that you update.

Assigning Custom Generation Rules to Columns

Assign custom generation rules to one or more target columns to generate data in the columns.

1. In the Define | Data Generation view, click the Generation 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.
   Note: When you select a source input column, the rule ports with same datatype 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.

Deleting Generation Rule Assignments

You can delete a generation rule assignment to a column.

1. In a project, click Define > Data Generation.
2. Select the column that you want to delete the rule assignment for.
3. Click Actions > Clear Assignment.
   The Clear Rule Assignment window appears.
Data Generation Plans and Workflows

A data generation plan contains columns and entities that represent the tables and data that you want to generate. The plan also contains the rules to generate the data. You generate a workflow from a plan to perform the generation operations.

When you start a workflow for the plan, the data generation operation starts.
CHAPTER 10

Data Generation Techniques and Parameters

This chapter includes the following topics:

- Data Generation Techniques and Parameters Overview, 125
- Data Generation Techniques, 126
- Data Generation Parameters, 126
- Dictionary Generation, 128
- Effective Dates Generation, 130
- Expression Generation, 132
- Random Generation, 133
- Reference Lookup Generation, 140
- Sequence Generation, 142
- Set of Values Generation, 144

Data Generation Techniques and Parameters Overview

A data generation technique defines how you want to generate data in a column. A generation parameter is an option that you configure for the technique.

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

Some generation techniques are available only for ad hoc generation rules. The generation techniques that you can select to create a rule can change when you create an ad hoc generation rule.

You can create generation rules that create random strings, use data from a dictionary or set of values, or that create data based on a sequence that you define.
Data Generation Techniques

You can apply generation techniques based on the target datatype that you configure for a column.

For example, if the column datatype is numeric, you can define generation values that are within a fixed range or values in a sequence. You can select an additional rule to create values that are not valid values within the data set.

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

<table>
<thead>
<tr>
<th>Generation Technique</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom</td>
<td>Standard technique that applies generation rules from a mapplet. The custom mapplet accepts input and uses the logic in the mapplet to generate the output.</td>
</tr>
<tr>
<td>Dictionary</td>
<td>Standard technique that imports dictionary values to generate data.</td>
</tr>
<tr>
<td>Effective Dates</td>
<td>Ad hoc technique that generates related dates for two columns in the target database.</td>
</tr>
<tr>
<td>Expression</td>
<td>Ad hoc technique that accepts expression as the input to generate the output.</td>
</tr>
<tr>
<td>Random</td>
<td>Standard technique that generates random strings, values, and dates.</td>
</tr>
<tr>
<td>Reference Lookup</td>
<td>Ad hoc technique that generates data from a reference lookup table. You can have multiple column assignments within a reference lookup rule.</td>
</tr>
<tr>
<td>Sequence</td>
<td>Standard technique that generates numeric and date values in a sequence.</td>
</tr>
<tr>
<td>Set of Values</td>
<td>Standard technique that uses a finite set of values to use in data generation.</td>
</tr>
</tbody>
</table>

Data Generation Parameters

Configure data generation parameters to define how to apply a data generation technique. The parameters that you configure depend on the data type that you need to generate. Some generation techniques are not available for all data types.
The following image shows data generation parameters that appear when you configure a random string generation rule:

**Exception Test Data**

Configure exception test data parameters to generate null and values that are not valid.

Select a global rule that you previously saved or select default to generate random null values and values that are not valid. Invalid values are data values that do not conform to the generation rule.

For example, you want to create a random date generation rule that generates 85% of the dates between 2012, and 15% of the dates between 2005 and 2010. You create a rule called Dates_2005 that generates dates from 2005 to 2010. You save the Dates_2005 rule. You create a rule called Recent_Dates that generates dates from 2012 to 2013. When you create the Recent_Dates rule, you select Dates_2005 in the invalid values parameter and enter 15 in the percent parameter.
Exception Test Data Parameters

You can configure exception test data parameters based on the data generation rules that you create. You can use exception test data parameters to generate null values and values that are not valid.

The following table describes the exception test data parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null Values</td>
<td>A list of global rules that you previously created or the default rule to generate null values. Percentage defines the percent of rows to have the null values.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules that you previously created or the default rule to generate values that are not valid. Percentage defines the percent of rows to have the values that are not valid.</td>
</tr>
</tbody>
</table>

Dictionary Generation

A dictionary generation rule is a global or ad hoc generation rule that loads dictionary values into the target test database.

Before you create a dictionary generation rule, you must add a dictionary to TDM in the Administrator | Dictionaries view. Users that have permissions to create generation rules have permissions to add dictionaries.

Dictionary generation can use the string, numeric, date, and binary data types. You can create a rule based on a dictionary column. The binary data type contains a serial number column that increases performance. Generate dictionary values for columns that have a finite number of values, such as states or countries.

You can create dictionary generation rule either from a flat file or relational database. You can generate data for a binary data type from relational databases but not from flat files.

Dictionary generation rules can be global rules or ad hoc rules.

The following image shows the dictionary generation parameters:

```
<table>
<thead>
<tr>
<th>Generates string data from dictionary</th>
</tr>
</thead>
<tbody>
<tr>
<td>*Dictionary Name</td>
</tr>
<tr>
<td>*Dictionary Column</td>
</tr>
<tr>
<td>Generate Unique Values</td>
</tr>
</tbody>
</table>

Exception Test Data

Null Values
Default 0%

Invalid Values
Default 0%
```
Dictionary Generation Example

You want to load customer names into the target database. You import a dictionary that has first and last name columns. You create two dictionary rules with the string data type. When you create the first name rule, you select the dictionary that contains the names and choose the first name dictionary column. In the last name rule, you select the same dictionary and choose the last name dictionary column.

Dictionary Generation Parameters

Create a dictionary generation rule to load dictionary values into target databases. Configure dictionary generation parameters for string, date, numeric, and binary data types.

The following table describes the parameters that you can configure to generate dictionary data:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dictionary Name</td>
<td>The name of the dictionary. You need to import a dictionary in TDM.</td>
</tr>
<tr>
<td>Dictionary Column</td>
<td>The column that you want to get the data from. If the values in the dictionary are larger than the size of the column then it truncates the values.</td>
</tr>
<tr>
<td>Dictionary Lookup Column</td>
<td>The lookup column in the dictionary that you can match with the table column to generate data. When you assign a standard generation rule to a column or create an ad hoc generation rule, you can specify the lookup column. This field is optional. If you enter double data type value in the lookup column and table column, the process generates null values in the target table.</td>
</tr>
<tr>
<td>Table Column</td>
<td>The column in the target table based on which you can match the lookup column in the dictionary. When you assign a standard generation rule to a column or create an ad hoc generation rule, you can specify the table column. This field is optional. If you enter double data type value in the lookup column and table column, the process generates null values in the target table.</td>
</tr>
<tr>
<td>Serial Number Columns</td>
<td>A serial number to find the location of the data. You can configure serial number for relational databases and not for flat files. You enter the serial number column for binary data types.</td>
</tr>
<tr>
<td>Generate Unique Values</td>
<td>Generates unique data values. You cannot generate unique values for flat file dictionaries and binary data types.</td>
</tr>
</tbody>
</table>
### Effective Dates Generation

An effective dates generation rule is an ad hoc rule that generates related date data for two columns. The effective dates are sequences of dates that consist of start dates and end dates. You can create effective dates rules with the date data type.

An effective dates generation rule generates a random start date within a range. The rule creates an end date that is after the start date and another start date that immediately follows the end date. The number of rows in the grouping column that have the same value determines the number of dates in the effective dates sequence.

The column in which you create the rule must be either the effective start date or the effective end date. If you create the rule on the effective start date column, you must select a different column as the effective end date column. If you create the rule on the effective end date column, you must select a different column as the effective start date column. The rule generates date values for both columns. You select the grouping column to generate sequences of dates in multiple rows.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null Values</td>
<td>A list of global rules or a default rule that is available for the data type you select. A null value is a value that a default rule or a global rule generates in the target for the columns based on the data type you select.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or a default rule that is available for the data type you select. Any value that does not exist in the dictionary is not valid. You cannot generate values that are not valid for binary data types.</td>
</tr>
</tbody>
</table>
The following image shows the effective dates generation parameters:

![Effective Dates Generation Parameters](image)

### Effective Dates Generation Parameters

Create effective dates generation rule for date datatypes.

The following table describes the parameters that you can configure to generate data for effective dates:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Date/Time</td>
<td>The date and time at which you can start to generate dates.</td>
</tr>
<tr>
<td>End Date/Time</td>
<td>The date and time at which you can stop to generate dates.</td>
</tr>
<tr>
<td>Exclude</td>
<td>One or two days of a week that you do not want to include in the effective dates generation rule.</td>
</tr>
<tr>
<td>Effective Start Date</td>
<td>The column name in the target table where you want to put the generated effective start date. Default is the column name in which you create the effective dates generation rule. You can change it to any other available column name.</td>
</tr>
<tr>
<td>Effective End Date</td>
<td>The column name in the target table where you want to put the generated effective end date. You can select any column other than the effective start date.</td>
</tr>
<tr>
<td>Grouping Column</td>
<td>The column that determines how you group the data to generate effective dates in the target column.</td>
</tr>
</tbody>
</table>
### Parameter Description

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generate Unique Values</td>
<td>Generates unique data values.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>Any value with effective end date greater than the effective start date. If effective start date is less than effective end date of the previous record for the grouping column, the values are not valid. This parameter has default rule selected and you cannot change the rule. You enter the percent for values that are not valid.</td>
</tr>
</tbody>
</table>

### Effective Dates Generation Example

You want to generate start dates and end dates for customer addresses.

The target table has two customers. Each customer has one identification number. The table has three address rows for each customer. You choose a date range to generate a random effective start date for each customer. You select an effective start date column to put the start date into. You select an effective end date column to put an end date into. You select the customer identification number as the grouping column.

The rule generates a random start date for the first address row for each customer. The rule generates a random end date that occurs after the start date. The rule generates a start date for the next row that is after the end date of the previous row.

The following table shows a sample table with the generated data:

<table>
<thead>
<tr>
<th>Customer ID</th>
<th>Address</th>
<th>Start Date</th>
<th>End Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1111</td>
<td>55 Crestwood Dr</td>
<td>11/20/2001</td>
<td>06/17/2003</td>
</tr>
<tr>
<td>1111</td>
<td>200 Larkin St</td>
<td>06/18/2003</td>
<td>04/16/2004</td>
</tr>
<tr>
<td>1111</td>
<td>1000 Cooke Ln</td>
<td>04/17/2004</td>
<td>08/16/2012</td>
</tr>
<tr>
<td>2222</td>
<td>12708 Danbrooke Ct</td>
<td>03/06/2008</td>
<td>05/19/2010</td>
</tr>
<tr>
<td>2222</td>
<td>1200 S Sixth Ave</td>
<td>05/20/2010</td>
<td>07/06/2011</td>
</tr>
<tr>
<td>2222</td>
<td>2243 Creeksie Ct</td>
<td>07/07/2011</td>
<td>11/24/2013</td>
</tr>
</tbody>
</table>

### Expression Generation

An expression generation rule is an ad hoc rule that generates data from an expression. You can create expression rules with the string, numeric, and date datatypes.

When you create an expression generation rule, you use the **Expression Builder** to create an expression. In the **Expression Builder**, you can choose from a list of columns and functions to include in the expression. To enter a join condition, you can choose columns and operators. Click the **Add** arrow to include the column in the expression.
The following image shows the expression generation parameters:

Expression Generation Example

You want to create a Name column that combines the data from the FirstName and LastName columns. You use the following expression to create the rule:

CONCAT (FirstName+'/'+LastName)

Expression Generation Parameters

You can configure expressions to generate data for string, date, and numeric datatypes. The following table describes the parameters that you can configure to generate expressions:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expression</td>
<td>An expression that you can configure to generate data.</td>
</tr>
<tr>
<td>Null Values</td>
<td>A list of global rules or default rule that are available for the datatype you select. A null value is value that a default rule or a global rule generates in the target column based on the datatype you select.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or default rule that are available for the datatype you select. Any value that does not follow the specified expression is not valid.</td>
</tr>
</tbody>
</table>

Random Generation

A random generation rule is a global or ad hoc rule that creates random string, numeric, and date values. You can use random generation to generate data such as salary and birth dates.

When you configure random generation, you can specify a minimum and maximum length for data strings and a minimum and maximum value for numeric data. You can include a regular expression to create a fixed
percentage of the generated data. The percentages of the pattern distributions must total 100. When you generate dates, you can select a start date and end date to create random dates within a range.

Random generation rules can be global rules or ad hoc rules.

The following image shows the random generation parameters:

![Random Generation Parameters](image)

**Random Generation Example**

You want to generate addresses for customers. You create a rule that uses the random technique and the string data type. You define a regular expression in the rule. The regular expression creates rows with a four digit number, a random string of 12 characters, and "St." or "Ave." You set the pattern percentage to 100 to generate all the rows with the regular expression.

**Date Random Generation Parameters**

Create a random data generation rule for date datatypes.

The following table describes the parameters that you can configure to generate random data for date datatypes:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Date/Time</td>
<td>The date and time at which you can start to generate random dates.</td>
</tr>
<tr>
<td>End Date/Time</td>
<td>The date and time at which you can stop to generate random dates.</td>
</tr>
<tr>
<td>Exclude</td>
<td>One or two days of a week that you do not want to include in the effective date generation rule. You must enter separate days in the exclude list.</td>
</tr>
</tbody>
</table>
String Random Generation Parameters

Create a random data generation rule for string datatypes.

The following table describes the parameters that you can configure to generate random data for string datatype:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>A range provided to generate random data. You can select Random to specify the range of the string for data generation.</td>
</tr>
<tr>
<td>Minimum Length</td>
<td>The minimum length for the string datatype. Default is 1.</td>
</tr>
<tr>
<td>Maximum Length</td>
<td>The maximum length for the string datatype. Default is the maximum value that the column permits.</td>
</tr>
<tr>
<td>From Regular Expression</td>
<td>A regular expression to generation random data. You can select From Regular Expression to create fixed percentage of data for generation.</td>
</tr>
<tr>
<td>Pattern</td>
<td>A regular expression that describes the format of generated data.</td>
</tr>
<tr>
<td>Distribution</td>
<td>The percentage of rows with the regular expression data. If you want to use more expressions, you can add more rows.</td>
</tr>
<tr>
<td>Generate Unique Values</td>
<td>Generates unique data values. You cannot generate unique values when you select From Regular Expression.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Null Values</td>
<td>A list of global rules or default rule that are available for the string datatype. A null value is a value that a default rule or a global rule generates in the target columns for the string datatype.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or default rule that you can select for the string datatype. Any value that is less than the minimum length or more than maximum length for a specified range is not valid. A string that does not match the specified pattern is not valid. You cannot generate values that are not valid when you select From Regular Expression.</td>
</tr>
</tbody>
</table>

**Numeric Random Generation Parameters**

Create random data generation rule for numeric datatypes.

The following table describes the parameters that you can configure to generate random data for numeric datatype:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Random</td>
<td>A rule to generate random data. You can select Random to specify the range of numeric values for data generation.</td>
</tr>
<tr>
<td>Minimum Value</td>
<td>The minimum value for the numeric datatype. Default is 0.</td>
</tr>
<tr>
<td>Maximum Value</td>
<td>The maximum value for the numeric datatype. Default is the maximum value that the column permits.</td>
</tr>
<tr>
<td>From Regular Expression</td>
<td>A regular expression to generation random data. You can select From Regular Expression to create fixed percentage of data for generation.</td>
</tr>
<tr>
<td>Pattern</td>
<td>A regular expression that describes the format of generated data.</td>
</tr>
<tr>
<td>Distribution</td>
<td>The percentage of rows with the regular expression data. If you want to use more expressions, you can add more rows.</td>
</tr>
<tr>
<td>Generate Unique Values</td>
<td>Generates unique data values. You cannot generate unique values when you select From Regular Expression.</td>
</tr>
</tbody>
</table>
null values
A list of global rules or default rule that are available for
the numeric datatype. A null value is a value that a
default rule or a global rule generates in the target for
the numeric datatype.

invalid values
A list of global rules or default rule that you can select
for the numeric datatype. Any value that is less than
the minimum or more than the maximum for a specified
range is not valid. A value that does not match the
specified pattern is not valid.
You cannot generate values that are not valid when you
select From Regular Expression.

Data Patterns for Random Generation
You can enter data patterns from regular expressions to generate string and numeric data.

To generate numbers that contain special characters or any other operators, you use random string data
generation technique. You can use the following operators to generate string data patterns: . , \d , \w,
( opt1| opt2|...), { }, [ ].

To generate numbers that do not contain special characters or any other operators, you use random numeric
data generation technique. To generate numeric data, you can combine the following patterns: \d,
alternates (1|2|3|...), and [0-9]. You cannot nest the alternates.

When you enter data patterns to generate the credit card number, Social Security number, and Social
Insurance numbers, the generated data might not be valid. These numbers follow certain algorithms and you
cannot use data patterns to generate valid numbers.

String Data Pattern Examples
You want to generate email addresses, credit card numbers, phone numbers, Social Insurance numbers,
Social Security numbers, dates, IP addresses, and employee IDs from regular expressions.

Email Address
To generate email addresses, you must enter the pattern in the following order:
1. Enter the first eight characters as lowercase alphabet characters [a-z].
2. Enter the sign (@).
3. Enter hotmail, yahoo, gmail, aol, or Comcast.
4. Enter .com, co.uk, or .net.
To generate an email address, use the following pattern: [a-z][@][0\s\w]+(\s\w+)\.([a-z][\w-][\w-].\w+|\w\.\w+). To generate a data pattern for uppercase and lower alphabet characters, you can enter
([a-z]|([a-z])\[0\s\w]+(\s\w+)\.([a-z][\w-][\w-].\w+|\w\.\w+).

Credit Card Number
To generate a credit card number, you enter the first character and the pattern.
The following table shows the output for sample patterns that generate credit card numbers:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>4\d{3}-\d{4}-\d{4}-\d{4}</td>
<td>4312-0036-5436-0876</td>
</tr>
<tr>
<td>3\d{4} \d{4} \d{6}</td>
<td>3852 0000 023237</td>
</tr>
</tbody>
</table>

Phone Number

The following table shows the output for sample patterns that generate phone numbers:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>\d{3}.\d{3}.\d{4}</td>
<td>617.555.1212</td>
</tr>
<tr>
<td>\d{3}-\d{3}-\d{4}</td>
<td>617-555-1212</td>
</tr>
<tr>
<td>(\d{3}) \d{3}-\d{4}</td>
<td>(617) 555-1212</td>
</tr>
<tr>
<td>+\d{3}.\d{3}.\d{4}</td>
<td>+1.617.555.1212</td>
</tr>
<tr>
<td>\d{3} \d{3} \d{4}</td>
<td>1.617.555.1212</td>
</tr>
</tbody>
</table>

Social Insurance Number

The following table shows the output for sample patterns that generate Social Insurance numbers:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>\d{3}-\d{3}-\d{3}</td>
<td>289-645-236</td>
</tr>
<tr>
<td>\d{3}\d{3}\d{3}</td>
<td>289.645.236</td>
</tr>
<tr>
<td>\d{3} \d{3} \d{3}</td>
<td>289 645 236</td>
</tr>
</tbody>
</table>

Social Security Number

The following table shows the output for a sample pattern that generates a Social Security number:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>\d{3}-\d{3}-\d{4}</td>
<td>999-99-9999</td>
</tr>
<tr>
<td>^\d{3}-\d{2}-\d{4}$</td>
<td>333-22-4444</td>
</tr>
<tr>
<td>(^?!000)\d{3}) ([- ]?) ((?!00)\d{2}) ([- ]?) ((?!0000)\d{4})</td>
<td>123-45-6789</td>
</tr>
</tbody>
</table>
Date

The following table shows the output for sample patterns that generate dates:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1</td>
<td>3</td>
</tr>
<tr>
<td>(4</td>
<td>6</td>
</tr>
<tr>
<td>(0</td>
<td>1-9)[10-2][3]-([0</td>
</tr>
</tbody>
</table>

IP Address

The following table shows the output for sample patterns that generate IP addresses:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>([0-9][1-9][0-9][1-9][0-9][0-9][0-9][0-9][0-9][0-9]/.([0-9][1-9][0-9][1-9][0-9][0-9][0-9][0-9][0-9][0-9]</td>
<td>255.255.0.0</td>
</tr>
<tr>
<td>([0-9][1-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9][0-9]/.([0-9][1-9][0-9][1-9][0-9][0-9][0-9][0-9][0-9][0-9]</td>
<td>192.168.0.1</td>
</tr>
</tbody>
</table>

Employee ID

The following table shows the output for sample patterns that generate employee IDs:

<table>
<thead>
<tr>
<th>Pattern</th>
<th>Sample Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>([A-Z][a-z][2]._ld+</td>
<td>ldd_1111</td>
</tr>
<tr>
<td>([A-Z][A-Z]-ld'd'd'd'd)</td>
<td>ID-4321</td>
</tr>
</tbody>
</table>

Numeric Data Pattern Examples

You want to generate credit card numbers, phone numbers, Social Insurance numbers, and Social Security numbers from regular expressions.

To generate a 10-digit phone number, consider the following scenarios:

- A phone number that starts with 3 or 4, you enter the data pattern as (3|4)[d][9].
- A phone number that starts with 6 and consists of digits such as 2,4,6, and 8, you enter the data pattern as 6(2|4|6|8)[0-9].
- A phone number that starts with 6 and consists of digits between 3-7, you enter the data pattern 6 [3-7][9].
The following table shows the sample output for the patterns that you can enter:

<table>
<thead>
<tr>
<th>Example</th>
<th>Pattern</th>
</tr>
</thead>
<tbody>
<tr>
<td>Credit card number</td>
<td>You enter the pattern as 5\d{15} to generate a sample output as 5312003654360876.</td>
</tr>
<tr>
<td>Phone number</td>
<td>You enter the pattern as \d{10} to generate a sample output as 6175551212.</td>
</tr>
<tr>
<td>Social Insurance number</td>
<td>You enter the pattern as \d{9} to generate a sample output as 289645236.</td>
</tr>
<tr>
<td>Social Security number</td>
<td>You enter the pattern as \d{9} to generate a sample output as 999999999.</td>
</tr>
</tbody>
</table>

Reference Lookup Generation

A reference lookup generation rule is an ad hoc rule that generates data from a reference table. You can create reference lookup rules with the string, numeric, and date data types.

Create a reference lookup generation rule to generate column data using data from an existing table. Specify the parent table and assign the parent columns to child columns in the target database.

You must assign a parent table and parent column to the target column that you create the rule for. When you create the rule you can also assign parent columns from the same parent table to other target child columns.
The following image shows the reference lookup generation parameters:

Reference Lookup Generation Example

You want to generate customer identification numbers in a column. You want to use the same customer identification numbers that exist in a different table. You create a reference lookup generation rule that references the column in the other table and generates the values from that table.

Reference Lookup Generation Parameters

Create reference lookup rules for string, numeric, and date datatypes.

The following table describes the parameters that you can configure to generate data from a reference lookup table:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minimum Value</td>
<td>The minimum number of child records for a parent record. Default is 1.</td>
</tr>
<tr>
<td>Maximum Value</td>
<td>The maximum number of child records for a parent record. Default is 3.</td>
</tr>
<tr>
<td>Data Source</td>
<td>The connection to the parent table.</td>
</tr>
<tr>
<td>Parent Table</td>
<td>The parent table that you can get the data from.</td>
</tr>
<tr>
<td>Child Column</td>
<td>The column name on which you create the rule. You cannot remove the first child column. To enter additional child columns, you can add more rows.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-------------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Parent Column</td>
<td>The parent column for the corresponding child column. You add more rows to enter additional child and parent columns. You can also assign parent columns from the same parent table to the other target child columns.</td>
</tr>
<tr>
<td>Lookup Condition</td>
<td>A conditional expression that you enter to extract the data. The data you extract from the parent table must conform to this condition.</td>
</tr>
<tr>
<td>Parent records without children</td>
<td>The percent of parent records without a child.</td>
</tr>
<tr>
<td>Child records without a parent</td>
<td>The percent of child records without a parent.</td>
</tr>
</tbody>
</table>

**Sequence Generation**

A sequence generation rule is a global or ad hoc rule that generates numeric and date values in a sequence.

You select a beginning value and an increment value when you create a sequence generation rule.

Sequence generation rules can be global rules or ad hoc rules.

The following image shows the sequence generation parameters:

```
<table>
<thead>
<tr>
<th>Generates numeric data in a sequence</th>
</tr>
</thead>
<tbody>
<tr>
<td>* Start Value</td>
</tr>
<tr>
<td>* Increment Value</td>
</tr>
<tr>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Exception Test Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Null Values</td>
</tr>
<tr>
<td>Default</td>
</tr>
<tr>
<td>%</td>
</tr>
<tr>
<td>Invalid Values</td>
</tr>
<tr>
<td>Default</td>
</tr>
<tr>
<td>%</td>
</tr>
</tbody>
</table>
```

**Sequence Generation Example**

You want to generate ticket numbers in ascending order. The target table has a TICKET_NUMBER column. You create a sequence generation rule with the numeric data type. You enter a start value of 1000 and an increment value of one (1).

**Date Sequence Generation Parameters**

Create sequence generation rule for date datatypes.
The following table describes the parameters that you can configure to generate a sequence of data for date datatype:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Date</td>
<td>The date at which you can start generation of sequence data for date datatypes.</td>
</tr>
<tr>
<td>Increment by</td>
<td>The value to increment after the start date. You can enter a number and select year, month, day, hour, minutes, or seconds.</td>
</tr>
<tr>
<td>Null Values</td>
<td>A list of global rules or default rule that are available for the date datatype. A null value is a value that a default rule or a global rule generates in the target column for the date datatypes.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or default rule that are available for the date datatype. Any value that is less than the sequence start value for a positive increment or more than the sequence start value for a negative increment is not valid.</td>
</tr>
</tbody>
</table>

**Numeric Sequence Generation Parameters**

Create sequence generation rule for numeric datatypes.

The following table describes the parameters that you can configure to generate a sequence of data for numeric datatype:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Value</td>
<td>The minimum value to start generation of a sequence. You can enter a number to generate the sequence data.</td>
</tr>
<tr>
<td>Increment Value</td>
<td>The value to increment after the start value during generation of the sequence data. You can enter a number to increment the value.</td>
</tr>
<tr>
<td>Null Values</td>
<td>A list of global rules or default rule that are available for the numeric datatype. A null value is value that a default rule or a global rule generates in the target column for the numeric datatype.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or default rule that are available for the numeric datatype. Any value that is less than the sequence start value for a positive increment or more than the sequence start value for a negative increment is not valid.</td>
</tr>
</tbody>
</table>
Set of Values Generation

A set of values generation rule is a global or ad hoc rule that writes a small data set to the target table. You can list the data values that you want to write to the table when you create a set of values generation rule.

You can use the string, numeric, and date data types in a set of values generation rule. You add data values in the rule and specify what percent of the rows in the target table you want to have the value. You can use set of values generation for columns that have a small number of values, such as membership status.

Set of values rules can be global rules or ad hoc rules.

The following image shows the set of values parameters:

![Set of Values Generation Parameters](image)

Set of Values Generation Example

You want to generate payment information. The target database has a PAYMENT_TYPE column with Visa, MasterCard, and Paypal values. You want 60% of the rows to have the Visa value, 30% of the rows to have the MasterCard value, and 10% of the rows to have the Paypal value.

You create a set of values rule with the string data type. You add Visa in the value field and enter 60 in the distribution percent field. You click the add button twice to add two additional values to the rule. In the second row, you enter MasterCard in the value field and 30 in the distribution percent field. In the third row, you enter Paypal in the value field and 10 in the distribution field.

Set of Values Generation Parameters

Configure set of values parameters to generate data for string, date, and numeric datatypes.

The following table describes the parameters that you can configure to generate a set of values:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Value</td>
<td>The exact data value that you want to write to a target table. You can enter exact values for the string and numeric datatypes. You enter a date for date datatype. If you want to use more than one value, you can add more rows.</td>
</tr>
<tr>
<td>Distribution%</td>
<td>The percent of rows for the specified value. The sum of the distribution of the values have to be 100%.</td>
</tr>
<tr>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Null Values</td>
<td>A list of global rules or default rule that are available for the datatype you select. A null value is a value that a default rule or a global rule generates in the target columns based on the datatype you select.</td>
</tr>
<tr>
<td>Invalid Values</td>
<td>A list of global rules or default rule that are available for the datatype you select. Any number, date, or string that is not a part of the set of values is not valid.</td>
</tr>
</tbody>
</table>
This chapter includes the following topics:

- Plans and Workflows Overview, 146
- Workflow Connections, 147
- Plan Components, 147
- Plan Settings, 148
- Masking Components, 154
- Subset Components, 155
- Generation Components, 155
- Component Criteria, 155
- Source Settings, 156
- Plan Management, 161
- Workflow Generation, 164
- Executing a Workflow, 165
- Workflow Executions View, 166

Plans and Workflows Overview

A plan defines a data subset, data masking, or data generation operation. It includes the components that you need to generate a workflow.

When you create a plan, you add data masking, data subset components, or data generation components based on the operation that you need to perform. You cannot add data masking or data subset components to a plan when you want to add data generation components. You can add policies and rules to run data masking operations. Add entities, groups, and templates to run data subset operations. Add entities and tables to run the data generation operations.

Define the workflow connections in the plan. Define the source and target connections. If required for the plan, edit staging and storage connections and define lookup and dictionary connections.

You can create multiple workflows from one plan. Define workflow properties in the plan, such as commit properties, update strategies, and recovery strategies.

View the sessions that run from each workflow. View statistics that describe the number of rows that the target receives. The Session Details tab shows the affected rows, the rejected rows, and the number of rows that the session wrote to the target.

When you start a workflow, the PowerCenter Integration Service completes the plan operations.
Plans and Workflows Task List

Complete 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 entities, groups, and templates.
4. Add data generation entities and tables to the plan. When you enter data generation components, you cannot enter data masking or data subset components.
5. Switch masking rules off or on.
6. Optionally, change the criteria of data subset components.
7. Configure plan settings including the plan properties, error and recovery strategies, and advanced settings.
8. Optionally, configure an override strategy for a data source or a table.
9. Generate a workflow.
10. Run the workflow.
11. Monitor the workflow.

Workflow Connections

Workflows contain connections to the PowerCenter Integration Service, the TDM repository, and one or more connection objects.

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.
- Staging connection. A connection to a database for staging data. The default connection entered in the Preferences view appears. You can change the connection for a plan.
- Lookup connection. A connection to a database that contains lookup tables.
- Storage connection. A connection to a database that contains storage tables. The default connection entered in the Preferences view appears. You can change the connection for a plan.
- Teradata connection. An ODBC connection to a Teradata database.
- Dictionary connection. A connection to a database that contains a dictionary table for substitution, email masking, and data generation.

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. You can also perform data generation operations. When you perform data generation operations, you cannot perform data subset or data masking operations.

The following table describes the components that you can add to a plan for each type of operation:

<table>
<thead>
<tr>
<th>Component</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rule</td>
<td>Data masking rules. For 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.</td>
</tr>
<tr>
<td>Policy</td>
<td>A Policy is a set of data domains. Each data domain can have multiple rules. You can add policies to a data masking plan.</td>
</tr>
<tr>
<td>Entity</td>
<td>Defines a set of tables that are related based on physical or logical constraints. An entity can contain parent tables and child tables. An entity maintains relational hierarchy in the tables. Define filter parameters for ports in the entity to extract the columns from the tables in an entity. When you run a workflow generated from a data subset plan, the PowerCenter Integration Service extracts source data based on the filter parameters defined in the entity and loads the data to the subset tables. If an entity is part of a template that is in the plan, you cannot add the same entity to the plan. Remove the template from the plan, or remove the entity from the template to continue. You can add entities to a data subset or a data generation plan.</td>
</tr>
<tr>
<td>Group</td>
<td>Defines a set of tables to copy to a target subset database. You can add groups to a data subset plan.</td>
</tr>
<tr>
<td>Template</td>
<td>A reusable component that contains the entities and groups of a specific module that you want to subset based on common parameters. You can add templates to a data subset plan.</td>
</tr>
<tr>
<td>Table</td>
<td>Defines the target table columns that contain generation rule assignments. You can add tables to a data generation plan.</td>
</tr>
</tbody>
</table>

**Plan Settings**

Configure plan settings that apply to all data sources in the plan.

The plan settings include PowerCenter options, connection information, recovery strategies, and source and target properties.

**Connection Options**

Enter source and target connections.
The following table describes connection options:

<table>
<thead>
<tr>
<th>Connection Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Use Connection Variables</td>
<td>Use a connection variable instead of selecting a connection. When you select this, all connections appear with connection variable names.</td>
</tr>
<tr>
<td>Use Source Connection as Target Connection</td>
<td>Use the same connection for source and targets. Use with inplace masking.</td>
</tr>
<tr>
<td>Source Connection</td>
<td>A connection to the source database or the source file. Select a source connection from the list.</td>
</tr>
<tr>
<td>Source Connection Directory Path</td>
<td>Displays if flat file sources are present. Enter the path to the flat file source directory.</td>
</tr>
<tr>
<td>Target Connection</td>
<td>Choose relational or flat file from the list. Select a target connection from the list.</td>
</tr>
<tr>
<td>Parameter File Name</td>
<td>Appears if you select Use Connection Variables. The name of the parameter file that contains the variable names for source and target connections. Enter the name of the parameter file.</td>
</tr>
<tr>
<td>Parameter File Path</td>
<td>Appears if you select Use Connection Variables. The path to the file that contains the variable names for source and target connections. Enter the path to the parameter file.</td>
</tr>
<tr>
<td>Source File Name</td>
<td>The name of the IMS source file or VSAM source file. If the source is a Hierarchical IMS file, enter the name of the source unload file. If the source is a Hierarchical VSAM/SEQ file, enter the name of the source file. Required if the source connection is PXW_NRDB Batch.</td>
</tr>
<tr>
<td>Target File Name</td>
<td>The name of the IMS target file or VSAM target file. If the target is a Hierarchical IMS file, enter the name of the target unload file. If the target is a Hierarchical VSAM/SEQ file, enter the name of the target file. The target file name must be different from the source file name. Required if the target connection is PXW_NRDB Batch.</td>
</tr>
<tr>
<td>Teradata ODBC Connection</td>
<td>Name of the Teradata ODBC connection. Required if the target connection is Teradata PT. Select a connection from the list.</td>
</tr>
<tr>
<td>Dictionary Lookup Connection</td>
<td>Connection to a database that contains lookup tables. Required if the plan has a dictionary generation rule assignment or a mapplet rule that includes a transformation. Choose relational or flat file from the list. Select a dictionary lookup connection from the list.</td>
</tr>
<tr>
<td>Dictionary Connection</td>
<td>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.</td>
</tr>
</tbody>
</table>
Connection Options | Description
---|---
Email Dictionary Connection | A connection to the database that contains the email dictionary table. The email dictionary connection option does not appear unless the project contains a masking rule that requires a relational email dictionary.

Number of Records to Generate | Required. The number of records that you want to generate in the target table when you create a generation plan.

Reference Lookup Connection | A connection to the database that contains parent tables. Required if the generation plan has reference lookup rule assignment. Choose relational or flat file from the list. Select a dictionary lookup connection from the list.

---

Dictionary Connections

A dictionary is a flat file or 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.

When you define a substitution masking rule with a flat file dictionary, you can browse to a dictionary from Test Data Manager. Test Data Manager imports the structure of the dictionary file to enable you to define the serial number column and the lookup column.

By default, when you run a workflow, the PowerCenter Integration Service finds flat file dictionaries in the following location: `<Informatica installation directory>\server\infra_shared\lkpfiles` You cannot change this location. The TDM installation installs the built-in dictionaries in the lkpfiles location. When you create a custom dictionary file, Test Data Manager copies the file to the lkpfiles directory.

**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:

<table>
<thead>
<tr>
<th>Other Property Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncate Table</td>
<td>Truncates the table before loading it. By default, this is not selected.</td>
</tr>
<tr>
<td>Disable Indexes</td>
<td>Disables indexes for faster loading.</td>
</tr>
<tr>
<td>Disable Constraints</td>
<td>Disables constraints in the database before loading the data to the target and enables the constraints after loading.</td>
</tr>
</tbody>
</table>
Update Strategy Options

Configure how the PowerCenter Integration Service updates the target in the plan update strategy options. Configure whether the source rows are insert or update rows. Determine how the PowerCenter Integration Service updates the target based on whether the rows exist in the target.

The following table describes the options for update strategy:

<table>
<thead>
<tr>
<th>Update Strategy Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treat source rows as</td>
<td>Indicates how the PowerCenter Integration Service treats source rows. Choose Insert or Update. If an imported PowerCenter mapplet contains a transformation that is configured to set the update strategy, the default option is Insert.</td>
</tr>
<tr>
<td>Update as</td>
<td>Determines whether the PowerCenter Integration Service updates or inserts rows. Choose one of the following options: - Update. The PowerCenter Integration Service updates all rows flagged for update. - Insert. The PowerCenter Integration Service inserts all rows flagged for update. - Update else Insert. The PowerCenter Integration Service updates rows flagged for update if they exist in the target, and inserts remaining rows marked for insert.</td>
</tr>
</tbody>
</table>

Data Generation Properties

Before you create a plan to generate data, you can define default settings in the Administrator | Preferences view. You can update the default settings when you create a data generation plan.

The default rules are not populated if the rules are not part of a particular project. If the default rules are blank, then you need to select the default rules that are applicable to the plan.

The following table describes the properties to update default settings for data generation:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Default Numeric Generation Rule</td>
<td>The default rule you have set to generate data for numeric datatype. You can select a separate default rule for numeric datatype.</td>
</tr>
<tr>
<td>Default String Generation Rule</td>
<td>The default rule you have set to generate data for string datatype. You can select a separate default rule for string datatype.</td>
</tr>
<tr>
<td>Default Date Generation Rule</td>
<td>The default rule you have set to generate data for date datatype. You can select a separate default rule for date datatype.</td>
</tr>
<tr>
<td>Default Binary Generation Rule</td>
<td>The default rule you have set to generate data for binary datatype. You can select a separate default rule for date datatype.</td>
</tr>
<tr>
<td>Apply Default Rule to</td>
<td>The default rule you have set for all the columns without generation rules. You can also apply the default rule to the columns that do not have null values.</td>
</tr>
<tr>
<td>Property</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Minimum</td>
<td>The minimum number of child records for a parent record. Default is 1. You can enter a minimum numeric value.</td>
</tr>
<tr>
<td>Maximum</td>
<td>The maximum number of child record for a parent record. Default is 1. You can enter a maximum numeric value.</td>
</tr>
</tbody>
</table>

### Error and Recovery Options

Configure properties for committing data and error recovery in the **Error and Recovery** fields. The section is minimized by default. The options are populated with default data if available.

The following table describes the error and recovery options:

<table>
<thead>
<tr>
<th>Error and Recovery Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop on Error</td>
<td>Indicates how many non fatal errors the PowerCenter Integration Service encounters before it stops the session. If you enter zero, the session does not stop for non fatal errors. Default is zero.</td>
</tr>
<tr>
<td>Suspend on Error</td>
<td>Suspend workflow processing when an error occurs. You can fix errors and then restart the workflow. The default is No.</td>
</tr>
<tr>
<td>Rollback Transactions on Error</td>
<td>Rolls back the transaction at the next commit point when it encounters a non fatal write error. The default is Yes.</td>
</tr>
</tbody>
</table>
| Recover Strategy                 | Strategy for recovering a workflow when errors occur. Choose one of the following recovery strategies:  
- Resume from last checkpoint. The PowerCenter Integration Service saves the session state of operation and maintains target recovery tables.  
- Restart task. The PowerCenter Integration Service runs the task again when it recovers the workflow.  
- Fail task and continue workflow. Default. The PowerCenter Integration Service cannot recover the task, but it continues the workflow. |
| Enable High Availability Recovery| Select Yes to enable workflow recovery. High availability recovery is not available for web service workflows. The default is No. |
| Commit Type                      | Choose a source-based or target-based commit. By default, the PowerCenter Data Integration Service performs a target-based commit. |
| Commit Interval                  | The number of rows to process before a commit. Default is 10,000. |
The Advanced Options include options to set the locale, change the staging connection and storage connection, and set the number of tables to process in a workflow. You can also configure an email address to send notifications to. This section is minimized by default. The options are populated with default data if available.

The following table describes advanced options:

<table>
<thead>
<tr>
<th>Advanced Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Staging Connection</td>
<td>A connection to a database for creating temporary staging tables. The default set in the Workflow Generation section appears. You can change the connection used in a plan. Use this connection for scenarios that involve Cyclic Subset, Tending to Cyclic subset, Inplace Batch Updates, Non Relational Shuffle, and Hierarchical PWX.</td>
</tr>
<tr>
<td>Storage Connection</td>
<td>A connection to a database that contains storage tables. The Data Masking transformation maintains the storage tables for repeatable substitution and expression masking between sessions. A storage table row contains the source column and a masked value pair. The storage tables for substitution and expression masking are separate tables, IDM_SUBSTITUTION_STORAGE and IDM_EXPRESSION_STORAGE.</td>
</tr>
<tr>
<td>Datetime Format String</td>
<td>Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds.</td>
</tr>
<tr>
<td></td>
<td>- Seconds. MM/DD/YYYY HH24:MI:SS</td>
</tr>
<tr>
<td></td>
<td>- Milliseconds. MM/DD/YYYY HH24:MI:SS.MS</td>
</tr>
<tr>
<td></td>
<td>- Microseconds. MM/DD/YYYY HH24:MI:SS.US</td>
</tr>
<tr>
<td></td>
<td>- Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS</td>
</tr>
<tr>
<td></td>
<td>Default is microseconds.</td>
</tr>
<tr>
<td>Enable High Precision</td>
<td>Processes the Decimal datatype to a precision of 28. Default is No.</td>
</tr>
<tr>
<td>Tables Per Workflow</td>
<td>Sets the number of tables to process in a workflow. Default is 10.</td>
</tr>
<tr>
<td>Mapplet</td>
<td>Use to Reuse or Replace the mapplets in PowerCenter.</td>
</tr>
<tr>
<td>Max Parallel Sessions</td>
<td>The maximum number of sessions that can run at the same time. Applicable if you disable indexes and constraints in the target by selecting the options under Target options.</td>
</tr>
<tr>
<td>Locale</td>
<td>Sets the locale for data masking.</td>
</tr>
<tr>
<td>Email</td>
<td>An email address to send notifications to when the plan execution completes. Click settings to configure the email address, subject, and text of the email.</td>
</tr>
<tr>
<td>Advanced Options</td>
<td>Description</td>
</tr>
<tr>
<td>----------------------------------</td>
<td>--------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Target Load Type</td>
<td>Choose Normal or Bulk. If you select Normal, the PowerCenter Integration Service loads targets normally. Choose Normal mode if the mapping contains an Update Strategy transformation. Choose Bulk to improve session performance. Bulk load limits the ability to recover because no database logging occurs. Default is Normal.</td>
</tr>
<tr>
<td>Target Schema Name</td>
<td>Table name prefix. Specify the target schema name if the target is in a different schema.</td>
</tr>
<tr>
<td>Source Schema Name</td>
<td>Table owner name. Specify the source schema name if the source is in a different schema.</td>
</tr>
<tr>
<td>Batch Update</td>
<td>Enables batch update to update target rows in a batch mode. Improves performance. Use with inplace masking.</td>
</tr>
<tr>
<td>Batch Size</td>
<td>The number of target rows that you want to move at a time. This option is available when you select the Batch Update as Yes. Use with inplace masking.</td>
</tr>
<tr>
<td>Use Oracle ROWID in Source</td>
<td>Determines whether to use the ROWID from Oracle sources as a column. The ROWID indicates the physical location of a row in the database. Enabled when the Oracle source and target connections are the same.</td>
</tr>
<tr>
<td>Join parent and child tables using</td>
<td>Choose how to join the parent and the child tables in the SQL. Choose one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- IN clause. At least one value in the parent table must match a value in the child table.</td>
</tr>
<tr>
<td></td>
<td>- EXISTS clause. The join must return at least one row.</td>
</tr>
<tr>
<td>Enable special characters in metadata</td>
<td>Select whether to allow special characters in plan metadata. Choose one of the following options:</td>
</tr>
<tr>
<td></td>
<td>- Yes. Select yes to allow special characters in metadata.</td>
</tr>
<tr>
<td></td>
<td>- No. Select no to disable special characters in metadata.</td>
</tr>
<tr>
<td></td>
<td>Default is No.</td>
</tr>
</tbody>
</table>

**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 PowerCenter 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 entities, groups, and templates 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, you can restrict the subset operation to some of the entities instead of performing the data subset operation on all the entities in the project. You can choose the entities, groups, and templates that you want to test.

The **Add Subset Components** dialog box shows the entities, groups, and templates in the project. Select the components that you want to apply in the data subset operation. The PowerCenter Integration Service performs the data subset operation with the components that you select.

**Generation Components**

Generation components are the entities and tables in a data generation operation. When you create a plan for a data generation operation, select the data generation components that you want to include in the workflow.

To run data generation, you can restrict the generation operation to some of the generation rules instead of applying all the generation rules. You can choose the generation rules that you want to test. When you configure the plan criteria, you can enter the number of records that you want to generate.

The **Generation Components** dialog box shows the entities and tables in the project. Select the components that you want to apply in the data generation operation.

**Component Criteria**

You can define filter criteria for data subset components and for data masking components in a plan. Define the filter criteria for each column you need to filter.

You can define expressions to limit data values for specific columns in a data subset operation. You can also limit the value to a percentage of rows, an interval of rows, or to an absolute value. You can disable and enable data masking for a column.

Define filter criteria for a column in the **Plan Components** panel. Choose the data masking or data subset component to filter.
Filtering Data Subset Components

You can define filter criteria for data subset components at the column level.

1. To create a data subset filter for a column, click the data subset component in the Plan Components panel.

2. Select a column from the Criteria panel to apply the filter to.

   Test Data Manager shows existing filter criteria. You can change the filter criteria or add new criteria.

3. Choose a method to filter the column data:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Custom</td>
<td>Create a basic expression or an advanced expression to filter the data.</td>
</tr>
<tr>
<td>Limit</td>
<td>Limit the subset results by percentage, by absolute value, or by an interval of rows to write to the target.</td>
</tr>
</tbody>
</table>

4. If you choose the Custom method, select one of the following options to define a filter:

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic</td>
<td>Define a simple expression that includes just an operator and operand.</td>
</tr>
<tr>
<td>Advanced</td>
<td>Define an expression in the Expression Editor. You can include multiple columns and functions in the expression.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Connection Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source Connection</td>
<td>A connection to the source database or the source file. Select a source connection from the list.</td>
</tr>
<tr>
<td>Source Connection Directory Path</td>
<td>The path to the flat file source directory. Appears if flat file sources are present.</td>
</tr>
<tr>
<td>Source Filename</td>
<td>The name of the source file. Required if the source is a flat file. Default is <code>&lt;name of selected source&gt;.dat</code>.</td>
</tr>
<tr>
<td>Target Connection</td>
<td>A connection to the target database or the target file. Choose relational or flat file from the list. Select a target connection from the list.</td>
</tr>
<tr>
<td>Output Filename</td>
<td>The name of the target file. Required if the target is a flat file. Default is <code>&lt;name of selected source&gt;.out</code>.</td>
</tr>
<tr>
<td>Column Delimiter</td>
<td>A character which separates columns from each other in the CSV file. Required if the source is a flat file. Default is a comma (,).</td>
</tr>
<tr>
<td>Row Separator</td>
<td>A character which separates columns from each other in the CSV file. Required if the source is a flat file. Default is a pair of double quotes (&quot;`).</td>
</tr>
<tr>
<td>Source File Name</td>
<td>The name of the IMS source file or VSAM source file. If the source is a Hierarchical IMS file, enter the name of the source unload file. If the source is a Hierarchical VSAM/SEQ file, enter the name of the source file. Required if the source connection is PXW_NRDB Batch.</td>
</tr>
<tr>
<td>Target File Name</td>
<td>The name of the IMS target file or VSAM target file. If the target is a Hierarchical IMS file, enter the name of the target unload file. If the target is a Hierarchical VSAM/SEQ file, enter the name of the target file. The target file name must be different from the source file name. Required if the target connection is PXW_NRDB Batch.</td>
</tr>
<tr>
<td>Teradata ODBC Connection</td>
<td>Name of the Teradata ODBC connection. Required if the target connection is Teradata PT. Select a connection from the list.</td>
</tr>
<tr>
<td>Lookup Connection</td>
<td>Connection to a database that contains lookup tables. Required if the plan has a mapplet rule which includes a transformation. Choose relational or flat file from the list. Select a lookup connection from the list.</td>
</tr>
</tbody>
</table>
### Connection Options

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dictionary Connection</td>
</tr>
<tr>
<td>Email Dictionary Connection</td>
</tr>
</tbody>
</table>

### Target Properties

The following table describes target options that you can configure at the data source level and at the source level:

<table>
<thead>
<tr>
<th>Other Property Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Truncate Table</td>
<td>Truncates the table before loading it. Default is disabled.</td>
</tr>
</tbody>
</table>

### Update Strategy Properties

The following table describes strategy options that you can configure at the data source level:

<table>
<thead>
<tr>
<th>Update Strategy Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treat source rows as</td>
<td>Indicates how the PowerCenter Integration Service treats source rows. Choose Insert or Update. If an imported PowerCenter mapplet contains a transformation that is configured to set the update strategy, the default option is Insert.</td>
</tr>
</tbody>
</table>
| Update as | Determines whether the PowerCenter Integration Service updates or inserts rows. Choose one of the following options:  
- Update. The PowerCenter Integration Service updates all rows flagged for update.  
- Insert. The PowerCenter Integration Service inserts all rows flagged for update.  
- Update else Insert. The PowerCenter Integration Service updates rows flagged for update if they exist in the target, and inserts remaining rows marked for insert. |
Error and Recovery Properties

The following table describes connection options that you can configure at the data source level:

<table>
<thead>
<tr>
<th>Error and Recovery Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Commit Type</td>
<td>Choose a source-based or target-based commit. By default, the PowerCenter Data Integration Service performs a target-based commit.</td>
</tr>
<tr>
<td>Commit Interval</td>
<td>The number of rows to process before a commit. Default is 10,000.</td>
</tr>
</tbody>
</table>

Source and Target Properties

You can override the name of the table owner if you want to select a table from a different schema.

The following table describes source and target options that you can configure at the data source level:

<table>
<thead>
<tr>
<th>Source and Target Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Imported Source Owner Name</td>
<td>The default name of the source owner.</td>
</tr>
<tr>
<td>Runtime Source Owner Name</td>
<td>The name of the table owner from which you want the source data. Specify the owner name of the source table if you need the table from a different schema.</td>
</tr>
<tr>
<td>Target Prefix Name</td>
<td>The name prefix of the target table. Specify the target schema name if the target is in a different schema.</td>
</tr>
</tbody>
</table>
### Advanced Properties

The following table describes advanced options that you can override at the data source level and at the table level:

<table>
<thead>
<tr>
<th>Advanced Options</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Datetime Format String</td>
<td>Date-time format defined in the session properties. You can enter seconds, milliseconds, microseconds, or nanoseconds.</td>
</tr>
<tr>
<td></td>
<td>- Seconds. MM/DD/YYYY HH24:MI:SS</td>
</tr>
<tr>
<td></td>
<td>- Milliseconds. MM/DD/YYYY HH24:MI:SS.MS</td>
</tr>
<tr>
<td></td>
<td>- Microseconds. MM/DD/YYYY HH24:MI:SS.US</td>
</tr>
<tr>
<td></td>
<td>- Nanoseconds. MM/DD/YYYY HH24:MI:SS.NS</td>
</tr>
<tr>
<td></td>
<td>Default is microseconds.</td>
</tr>
<tr>
<td>Target Load Type</td>
<td>Choose Normal or Bulk. If you select Normal, the PowerCenter Integration Service loads targets normally. Choose Normal mode if the mapping contains an Update Strategy transformation. Choose Bulk to improve session performance. Bulk load limits the ability to recover because no database logging occurs. Default is Normal.</td>
</tr>
<tr>
<td>Target Schema Name</td>
<td>Table name prefix. Specify the target schema name if the target is in a different schema.</td>
</tr>
<tr>
<td>Source Schema Name</td>
<td>Table owner name. Specify the source schema name if the source is in a different schema.</td>
</tr>
<tr>
<td>Source Pre SQL</td>
<td>SQL query executed before the plan is run. Required if you need to perform an operation before you run the plan. Enter the complete SQL statement.</td>
</tr>
<tr>
<td>Source Post SQL</td>
<td>SQL query executed after the plan is run. Required if you need to perform an operation after you run the plan. Enter the complete SQL statement.</td>
</tr>
<tr>
<td>Batch Size</td>
<td>The number of target rows that you want to move at a time. This option is available when you select the Batch Update as Yes. Use with inplace masking.</td>
</tr>
<tr>
<td>Use Oracle ROWID in Source</td>
<td>Determines whether to use the ROWID from Oracle sources as a column. The ROWID indicates the physical location of a row in the database. Enabled when the Oracle source and target connections are the same.</td>
</tr>
<tr>
<td>Performance Hint</td>
<td>Optimizes the SQL query after the plan is run. Enter a performance hint value only for Oracle databases.</td>
</tr>
</tbody>
</table>
Partitioning Properties

You must specify partitioning details if the tables in the source database contain partitions.

The following table describes partitioning options that you can configure at the table level:

<table>
<thead>
<tr>
<th>Partitioning Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enable Partitioning</td>
<td>Select to enable partitioning of output tables.</td>
</tr>
<tr>
<td>Partition Type</td>
<td>Select the type of partitioning:</td>
</tr>
<tr>
<td></td>
<td>- Pass Through. Use pass through partitioning to increase data throughput without increasing the number of partitions.</td>
</tr>
<tr>
<td></td>
<td>- Key Range. Use key range when the source data tables are partitioned by key range.</td>
</tr>
<tr>
<td></td>
<td>- Database Partitioning. Use database partitioning for Oracle or IBM DB2 sources and targets.</td>
</tr>
<tr>
<td>Number of partitions</td>
<td>The number of partitions in the source data. Enter the number of partitions and click Select to enter partition information for the target data.</td>
</tr>
</tbody>
</table>

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. To perform a data generation operation, you must create a separate plan. Add entities, groups, and templates 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 entities, groups, and templates to add to the plan. Click Next.
   You can assign criteria for masking columns and generating subsets.
8. To skip adding a data generation component, click Next.
9. To skip masking a rule, select the check box for the rule and click Off.
10. To filter subset components, select the component and choose to enter a basic or advanced expression.
    Test Data Manager shows any existing filter criteria. You can change it.
11. To limit the subset results click Limit and choose to limit by percentage, by absolute value, or defining an interval of rows to create. Click Next.

12. Configure the connections and other properties. Click Next.

13. To override plan settings for a data source, select the data source and click Override Plan Settings and enter the properties.

14. To override data source settings for a table, select the table and click Override Data Source Settings and enter the properties.

15. Click Finish. The plan appears in the project.

**RELATED TOPICS:**
- "Generating a Workflow“ on page 164
- "Data Masking Plans and Workflows“ on page 78

**Creating a Data Generation Plan**

To perform data generation operation, you create a plan. When you add data generation components, you cannot add data subset or data masking components.

1. Open a project and click Execute.
2. Click New Plan.
3. In the New Plan dialog box, enter a name and optional description for the plan.
4. Click Next.
5. To skip adding the masking components, click Next.
6. To skip adding the subset components, click Next.
7. To add data generation operation to the plan, click Add Generation Components.
8. Select the entities and the tables to add to the plan.
9. Click OK. The entities and tables appear in the list.
10. Click Next.
11. To view or filter the table names, select the table.
12. To specify generation criteria for an entity, select the entity and enter the values.
13. Click Next.
14. To configure generation plan settings, enter the number of records that you want to generate.
15. Configure connections and target properties.
16. To make changes to data generation, error recovery, and advanced settings, you can edit the settings.
17. To override plan settings, click Override Plan Settings and enter the properties.
18. To override table settings, click Override Data Source Settings and enter the properties.
19. Click Finish. The generation plan appears in the project.

**Creating an Ad Hoc Generation Plan**

You can create an ad hoc generation plan for a column.
1. Open a project and click **Define > Data Generation**.
2. Select a column from the table for which you want to create an ad hoc generation plan.
3. Click **Actions > Generate and Execute**.
   The **Generate and Execute Plan** dialog box appears.
4. Enter a name for the plan or keep the default plan name that appears on the screen.
5. Select the Integration Service.
6. Enter the number of records that you want to generate.
7. Configure connections and target properties.
8. To make changes to data generation, error and recovery, and advanced settings, edit the settings.
9. Click **OK**.
   A **Create and Execute** dialog box appears.
10. To run the plan, click **Yes**.

### 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 project 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, data masking, and data generation operations.

You can generate and start a workflow in a single step.

When you generate a workflow, the PowerCenter Repository Service generates mappings for the workflow. When you start a workflow, the PowerCenter Integration Service performs the data masking, data subset, and data generation tasks defined in the plan. You can select the PowerCenter 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 PowerCenter Repository 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 161
- "Data Masking Plans and Workflows" on page 78

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:

<table>
<thead>
<tr>
<th>Property</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Workflow Name</td>
<td>A string that includes the project name, plan name, connection, and table name.</td>
</tr>
<tr>
<td>User</td>
<td>The name of the user that generated the workflow.</td>
</tr>
<tr>
<td>Generation</td>
<td>Date and time that the workflow was generated.</td>
</tr>
</tbody>
</table>

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 execute the workflow to run sessions from the PowerCenter mappings. If you generated multiple workflows for the same plan, you can execute each workflow separately.

You can start a workflow from the Plan | Properties view or the Plan | Workflow view.

1. If the plan you selected contains one workflow, click Actions > Execute Workflow in either view.
2. 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.
3. View 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 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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job ID</td>
<td>The job number that identifies the job. If you click the Job ID, you can view the workflow log.</td>
</tr>
<tr>
<td>Name</td>
<td>The workflow string that includes the project name, plan name, connection, and table name.</td>
</tr>
<tr>
<td>Description</td>
<td>Describes the type of job that the workflow performs. The job type can be profiling, import, and workflow operations.</td>
</tr>
<tr>
<td>Status</td>
<td>The current status of the workflow.</td>
</tr>
<tr>
<td>Start Date/Time</td>
<td>The date and time that the workflow started.</td>
</tr>
<tr>
<td>End Date/Time</td>
<td>The date and time that the workflow ended.</td>
</tr>
<tr>
<td>User</td>
<td>The name of the user that started the workflow.</td>
</tr>
</tbody>
</table>

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 Recover

Recover a stopped workflow. If high availability recovery is enabled, the PowerCenter Integration Service restarts the workflow from the last saved checkpoint. Otherwise, the PowerCenter Integration Service recovers the workflow according to the recovery strategy that you set in the plan.

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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Job ID</td>
<td>The job number that identifies the session. If you click the Job ID, you can view the session log.</td>
</tr>
<tr>
<td>Session Name</td>
<td>The session name is the same value as the workflow name, except the session name starts with &quot;S&quot; and ends with a sequential number. If you click the session name, you can view the session details.</td>
</tr>
<tr>
<td>Status</td>
<td>The current status of the session.</td>
</tr>
<tr>
<td></td>
<td>- In Queue. The PowerCenter Integration Service is waiting for resources before it starts the session.</td>
</tr>
<tr>
<td></td>
<td>- Running. The session is running.</td>
</tr>
<tr>
<td></td>
<td>- Success. The session finished successfully.</td>
</tr>
<tr>
<td></td>
<td>- Error. The session did not complete due to errors.</td>
</tr>
<tr>
<td>Source Rows Succeeded</td>
<td>The number of valid source rows read.</td>
</tr>
<tr>
<td>Source Rows Failed</td>
<td>The number of source rows in error.</td>
</tr>
<tr>
<td>Target Rows Succeeded</td>
<td>The number of rows written to the target.</td>
</tr>
<tr>
<td>Target Rows Failed</td>
<td>The number of rows that were not written to the target.</td>
</tr>
<tr>
<td>Start Date/Time</td>
<td>The date and time that the session started.</td>
</tr>
<tr>
<td>End Date/Time</td>
<td>The date and time that the session ended.</td>
</tr>
<tr>
<td>User</td>
<td>The name of the user that started the workflow.</td>
</tr>
</tbody>
</table>

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:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tables</td>
<td>The names of the tables that the session processed.</td>
</tr>
<tr>
<td>Affected Rows</td>
<td>The number of rows that the session processed from the table.</td>
</tr>
<tr>
<td>Rejected Rows</td>
<td>The number of rows that the PowerCenter Integration Service did not process due to errors.</td>
</tr>
<tr>
<td>Throughput</td>
<td>The number of affected rows that the PowerCenter Integration Service did not process due to errors.</td>
</tr>
<tr>
<td>Last Error Message</td>
<td>The text of the last error message in the session. Click the link to view the complete message.</td>
</tr>
</tbody>
</table>
Chapter 12

Monitor

This chapter includes the following topics:

- Monitor Overview, 169
- Jobs, 170
- Monitor Tasks, 171
- Logs, 171

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 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.

The following image shows the Monitor view:
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.

**Import from PowerCenter**
Imports a source from the PowerCenter repository.

**Profiling**
Performs data discovery for primary keys, entities, and data domains.

**Generate Workflow**
Generates a workflow from a plan.

**Execute Workflow**
Runs a workflow for data subset, data masking, or data generation operations after you generate the workflow.

**Workflow**
Runs the data subset, data masking, or data generation operation. The execute workflow job might run multiple workflows.

**Session**
Performs a task within the workflow. A workflow might have multiple sessions. Select a workflow to view the session details 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.

**Name**
The name or the type of job. You can view import, profiling, workflow, and session jobs.

**Description**
The name of the plan, profile, PowerCenter 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.

Workflow Recover
   Recovers an aborted or stopped workflow. If high availability recovery is enabled, the PowerCenter Integration Service restarts the workflow from the last saved checkpoint. Otherwise, the PowerCenter Integration Service clears the state of operation and the recovery information before it restarts the workflow.

Logs

You can view log files to troubleshoot jobs. To view the logs, click the job ID.

The TDM server generates a log file that you can view to debug problems when a TDM job fails. If you are viewing a workflow or a session, you can view the session and workflow logs. The PowerCenter Integration Service generates the session and workflow logs.
Reports Overview

You can run a report from Test Data Manager to see detailed information about a plan or a policy. Run the type of report that contains the information you want to see.

Test Data Manager contains 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:

<table>
<thead>
<tr>
<th>Report</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Masking Report</td>
<td>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.</td>
</tr>
<tr>
<td>Plan Audit Report</td>
<td>A report that you run on a plan that contains detailed information about the policy assignments in the plan.</td>
</tr>
<tr>
<td>Plan Detail Report</td>
<td>A report that you run on a plan that lists policy assignments, entity details, groups, templates, and template components that are part of the plan.</td>
</tr>
<tr>
<td>Row Count</td>
<td>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.</td>
</tr>
</tbody>
</table>
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:

<table>
<thead>
<tr>
<th>Component</th>
<th>Properties</th>
</tr>
</thead>
<tbody>
<tr>
<td>Policy Assignment</td>
<td>You can view the following policy assignment details:</td>
</tr>
<tr>
<td></td>
<td>- Data Source</td>
</tr>
<tr>
<td></td>
<td>- Table</td>
</tr>
<tr>
<td></td>
<td>- Column</td>
</tr>
<tr>
<td></td>
<td>- Data Type</td>
</tr>
<tr>
<td></td>
<td>- Data Domain</td>
</tr>
<tr>
<td></td>
<td>- Policy</td>
</tr>
<tr>
<td></td>
<td>- Rule</td>
</tr>
<tr>
<td></td>
<td>- Rule Type</td>
</tr>
<tr>
<td></td>
<td>- Additional Details</td>
</tr>
<tr>
<td></td>
<td>- Masking Properties</td>
</tr>
<tr>
<td></td>
<td>- Exception Handling</td>
</tr>
</tbody>
</table>

| 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:

<table>
<thead>
<tr>
<th>Component</th>
<th>Detail</th>
</tr>
</thead>
<tbody>
<tr>
<td>Masking Assignment</td>
<td>You can view the following policy assignment details:</td>
</tr>
<tr>
<td></td>
<td>- Data Source</td>
</tr>
<tr>
<td></td>
<td>- Table</td>
</tr>
<tr>
<td></td>
<td>- Column</td>
</tr>
<tr>
<td></td>
<td>- Data Type</td>
</tr>
<tr>
<td></td>
<td>- Data Domain</td>
</tr>
<tr>
<td></td>
<td>- Policy</td>
</tr>
<tr>
<td></td>
<td>- Rule</td>
</tr>
<tr>
<td></td>
<td>- Masking Properties</td>
</tr>
<tr>
<td>Entity</td>
<td>You can view the following entity details:</td>
</tr>
<tr>
<td></td>
<td>- Name</td>
</tr>
<tr>
<td></td>
<td>- Description</td>
</tr>
<tr>
<td></td>
<td>- Filter Criteria</td>
</tr>
<tr>
<td>Group</td>
<td>You can view the following group details:</td>
</tr>
<tr>
<td></td>
<td>- Name</td>
</tr>
<tr>
<td></td>
<td>- Description</td>
</tr>
<tr>
<td>Template</td>
<td>You can view the following template details:</td>
</tr>
<tr>
<td></td>
<td>- Name</td>
</tr>
<tr>
<td></td>
<td>- Description</td>
</tr>
<tr>
<td></td>
<td>- Filter Criteria</td>
</tr>
<tr>
<td>Template Component</td>
<td>You can view the following template component details:</td>
</tr>
<tr>
<td></td>
<td>- Name</td>
</tr>
<tr>
<td></td>
<td>- Type</td>
</tr>
<tr>
<td></td>
<td>- Description</td>
</tr>
<tr>
<td></td>
<td>- Component Name</td>
</tr>
</tbody>
</table>

### 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.

Row Count Report

A row count report contains a list of the tables that took part in a plan execution and the number of rows that the plan affected.

Running the Row Count 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 jobs, click Monitor.
4. Click a job that has an Execute Workflow job type.
5. Click Actions > Row Count Report.
The row count report opens in a new window.
This chapter includes the following topics:

- **ilmcmd Overview, 176**
- **Configuring ilmcmd, 177**
- **Running ilmcmd, 177**
- **Entering Options and Arguments, 177**
- **Syntax Notation, 178**
- **Delete, 178**
- **Export, 179**
- **Import, 181**
- **Search, 182**
- **Workflow, 183**

**ilmcmd Overview**

ilmcmd is a command line program that you use to complete TDM tasks. You can use ilmcmd to complete a subset of the tasks that you can perform with Test Data Manager.

Use ilmcmd to complete the following tasks:

- Import XML files into the TDM repository.
- Export data subset and masking objects to XML files.
- Search for data subset and masking objects.
- Validate policies, plans, and data subset objects.
- Delete data subset and masking objects.
- Generate and run workflows.
- Display the status of workflows.
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.kts

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 Windows, ilmcmd is a batch file with a .bat extension. 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:

<table>
<thead>
<tr>
<th>Convention</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-x</td>
<td>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.</td>
</tr>
<tr>
<td>&lt;x&gt;</td>
<td>Required option. If you omit a required option, the command line program returns an error message.</td>
</tr>
<tr>
<td>&lt;x</td>
<td>y&gt;</td>
</tr>
<tr>
<td>[x]</td>
<td>Optional option. The command runs whether or not you enter these options.</td>
</tr>
</tbody>
</table>
| [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 ILM repository. To delete an object, specify the name of the object or the name of an XML file that lists the object details.

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
• Rule
• Policy
• Project
• Plan

The ilmcmd delete command uses the following syntax:

ilmcmd
  <-Delete | -d>
The following table describes ilmcmd delete options and arguments:

<table>
<thead>
<tr>
<th>Option</th>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ObjectType</td>
<td>-ot</td>
<td>Required to delete an object by name. The type of object to export. For example, you can enter &quot;Group.&quot;</td>
</tr>
<tr>
<td>-Project</td>
<td>-app</td>
<td>Required to delete an object by name. The name of the project that contains the object to delete.</td>
</tr>
<tr>
<td>-Name</td>
<td>-n</td>
<td>Required to delete an object by name. The name of the object that you want to delete.</td>
</tr>
<tr>
<td>-Help</td>
<td>-h</td>
<td>Optional. Displays help for the command.</td>
</tr>
</tbody>
</table>

**Delete Examples**

**Deleting a Rule**

The following sample command deletes the *Credit/Card/Mask* rule:

```
./ilmcmd -d -ot Rule -n Credit/Card/Mask
```

**Deleting a Policy**

The following sample command deletes the *Personal policy*:

```
./ilmcmd -d -ot Policy -n Personal
```

**Export**

Exports objects from the ILM Test Data Manager to an XML file.

When you export an object from the ILM repository, you specify the object type and the location of the object in the repository. You can export the following object types:

- Rule
- Policy
- Data Domain
- Entity
- Group
- Template
- Profile
- Project
• DataSource
• 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ObjectType</td>
<td>object_type</td>
<td>Required. The type of object to export. For example, you can enter &quot;Group&quot; or &quot;RuleAssignment.&quot; If the object type contains a space in it, remove the space when you enter the argument.</td>
</tr>
<tr>
<td>-Name</td>
<td>object_name</td>
<td>Required. The name of the object you want to export.</td>
</tr>
<tr>
<td>-Project</td>
<td>project_name</td>
<td>Required to export a plan, entity, group, or template. The name of the project that contains the plan or port assignment to export.</td>
</tr>
<tr>
<td>-File</td>
<td>XML_file_location</td>
<td>Required. The path and file name of the XML file to which you export the object.</td>
</tr>
<tr>
<td>-DataSourceName</td>
<td>schema_name</td>
<td>Required to export a port assignment. The name of the schema that contains the port assignment to export.</td>
</tr>
<tr>
<td>-Help</td>
<td>n/a</td>
<td>Optional. Displays help for the command.</td>
</tr>
</tbody>
</table>

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 Rule

The following sample command exports the Credit_Card_Mask rule to the CCR.xml file:

```
ilmcmd -e -ot Rule -n Credit_Card_Mask -f /home/infal/Desktop/CCR.xml
```

Import

Imports objects from an XML file to the ILM Test Data Manager.

You can import the following object types:

- Rules
- Policies
- Data domains
- Entities
- Groups
- Templates
- Profile
- Project
- Data Source
- Source Definition
- Plans

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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-File</td>
<td>-f</td>
<td>Required. The path and file name of the import file.</td>
</tr>
<tr>
<td>-Help</td>
<td>-h</td>
<td>Optional. Displays help for the command.</td>
</tr>
</tbody>
</table>

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 ilmserv -port 6002
```

Search

Searches for source definitions, rules, policies, and plans.

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

- Rule
- Policy
- Data Domain
- Entity
- Group
- Template
- 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:

<table>
<thead>
<tr>
<th>Option</th>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-ObjectType</td>
<td>object_type</td>
<td>Required. The type of object to export. For example, you can enter &quot;Group&quot; or &quot;Project.&quot; If the object type contains a space in it, remove the space when you enter the argument.</td>
</tr>
<tr>
<td>-Project</td>
<td>folder_name</td>
<td>Required to search for a source definition, plan, entity, group, or template. The name of the project that contains the object to search for.</td>
</tr>
<tr>
<td>-DataSourceName</td>
<td>schema_name</td>
<td>Required to search for a source definition. The name of the schema that contains the source definition.</td>
</tr>
<tr>
<td>Option</td>
<td>Argument</td>
<td>Description</td>
</tr>
<tr>
<td>------------</td>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>-NamePattern -np</td>
<td>name_pattern</td>
<td>Required. The name pattern. Use the asterisk (<em>) character as a wild card. For example, the name pattern Rule_C</em> returns the following rules: - Rule_Customer - Rule_CustID</td>
</tr>
<tr>
<td>-Help -h</td>
<td>n/a</td>
<td>Optional. Displays help for the command.</td>
</tr>
</tbody>
</table>

**Search Examples**

**Searching for a 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 | -f> folder_name
<-PlanName | -p> plan_name
<-WorkflowName | -wfn> Workflow_name
<-IntegrationService | -is> integration_service_name
[-Help | -h]
```
The following table describes the ilmcmd workflow options and arguments:

<table>
<thead>
<tr>
<th>Option</th>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-Generate</td>
<td>-g</td>
<td>n/a</td>
</tr>
<tr>
<td>-Execute</td>
<td>-ex</td>
<td>n/a</td>
</tr>
<tr>
<td>-GetStatus</td>
<td>-gs</td>
<td>n/a</td>
</tr>
<tr>
<td>-Project</td>
<td>-f</td>
<td>project_name</td>
</tr>
<tr>
<td>-PlanName</td>
<td>-pn</td>
<td>plan_name</td>
</tr>
<tr>
<td>-WorkflowName</td>
<td>-wfn</td>
<td>workflow_name</td>
</tr>
<tr>
<td>-IntegrationService</td>
<td>-is</td>
<td>integration_service_name</td>
</tr>
<tr>
<td>-Help</td>
<td>-h</td>
<td>n/a</td>
</tr>
</tbody>
</table>

**Workflow Examples**

**Generating a Workflow**

The following sample command generates a workflow for the Plan_NameMasking plan:

ilmcmd -wf -g -Project Infra_Project -pn Plan_NameMasking

**Monitoring a Workflow**

The following sample command monitors the status of a workflow for the Plan_NameMasking plan:

ilmcmd -wf -ex -Project Infra_Project -pn Plan_NameMasking -is Integration_Service

**Running a Workflow**

The following sample command runs a workflow for the Plan_NameMasking plan:

ilmcmd -wf -ex -fn TDM -app App_TDM -pn Plan_NameMasking -is PCInteg -un Administrator -pd Administrator -hn ilmsv -port 6002
Glossary

accelerator
An Informatica solution that contains predefined data subset and data masking components for major business applications.

ad hoc generation rule
A data generation rule that you can create within a project. You can use ad hoc generation techniques within an ad hoc generation rule.

advanced masking
A data masking technique that can include other masking techniques and multiple input and output ports.

auto cascade
A cascade type that masks the primary key in the parent table and the related foreign keys in the child tables with the same value.

blurring
A masking parameter for numeric and date values that creates an output value within a fixed or percent variance from the source data value. Configure blurring to return a random value that is close to the original value.

calculated masking
A masking technique that uses an expression to combine the values of multiple ports.

cascade
A data masking component that masks columns across tables based on column value. Define similar columns in Test Data Manager in order to mask them with the same rules.

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

custom generation rule
A data generation rule that uses PowerCenter mapplets to generate data.
data discovery
The process of discovering the metadata of source systems that includes content, such as data values and frequencies, and structure such as primary keys, foreign keys, and functional dependencies.

data generation
The process to generate realistic test data for the testing environment without using the production data.

data masking
The process of replacing sensitive columns of source data with realistic test data.

data subset
A small, targeted, and referentially intact copy of production data.

dependent masking
A masking technique that replaces the values of one source column based on the values of another source column.

dictionary
A flat file or relational table of data that substitution masking uses to replace a column of data with similar but unrelated data.

dictionary generation
A standard data generation technique that imports dictionary values into the target column.

effective dates generation
An ad hoc generation technique that generates related dates for two columns in the target database.

encryption masking
A data masking technique that applies encryption algorithms to mask source data. You can choose the algorithm to encrypt the data.

entity
Defines a set of tables that are related based on physical or logical constraints. An entity can contain parent tables and child tables. An entity maintains relational hierarchy in the tables. Define filter parameters for ports in the entity to extract the columns from the tables in an entity. When you run a workflow generated from a data subset plan, the PowerCenter Integration Service extracts source data based on the filter parameters defined in the entity and loads the data to the subset tables.

Expression Builder
A wizard in Test Data Manager that includes Informatica functions, variables, and operators to build expressions from the input and the output column values.

expression generation
An ad hoc generation technique that generates data based on the expression.
expression masking
A data masking technique that applies an expression to a port to change or create data.

foreign key profile
A type of data analysis that finds column values in a data source that match the primary key column values in another data source.

generation rule
A rule that you create to define how you want to generate data in the target table.

group
Defines a set of tables to copy to a target subset database.

ilmcmd
A command line program that you can use to complete TDM tasks from any machine. ilmcmd allows you to complete tasks that you can complete in Test Data Manager.

key masking
A data masking technique that produces deterministic results for the same source data, masking rules, and seed value.

mask format
A masking parameter that limits each character in the output column to an alphabetic, numeric, or alphanumeric character.

nullification masking
A data masking technique that replaces column data with null values.

pattern
A data masking component that uses a regular expression to filter the ports that you want to mask when you assign a rule to columns. Use a pattern to mask sensitive data such as IDs, telephone numbers, postal codes, and state names.

plan
Defines data subset, data masking, or data generation operations. You can add entities, groups, templates, policies, rules, and tables to a plan. When you generate and run workflows from a plan, the PowerCenter Integration Service generates and runs the workflows.

policy
Defines the data masking rules, the data to mask, and masking parameters for a source.

PowerCenter
An Informatica solution that can extract data from multiple sources, transform the data according to business logic you build in the client application, and load the transformed data into file and relational targets for data warehousing.
profile
Used in data discovery.
Defines the tables, fields, and keys to discover. You can create profiles to discover the following items in tables:
- Sensitive fields
- Foreign keys
- Primary keys

project
A container component for entities, groups, templates, and one or more sources that you want to use in data discovery, subset, and masking operations. When you create a project, you add one or more sources to the project. Any source that you add to a project is not available to other projects.

random generation
A standard data generation technique that generates random data for string, numeric, and date datatypes.

random masking
A masking technique that produces random, non-repeatable results for the source data and masking rules.

range
A data masking parameter that defines the minimum and maximum values to mask the numeric and date datatypes, and the minimum and maximum values for string lengths of string datatype.

reference lookup generation
An ad hoc generation technique that generates data from a reference table.

residual table
In a data subset plan, a table that is missing from entities and groups in the plan but which is part of the project. You can add residual tables to a data subset plan.

result string replacement characters
A data masking parameter that defines the substitute characters for masked data.

rule
Defines the data masking technique, an optional rule qualifier, and masking parameters.

rule qualifier
A rule component that filters the records that you want to mask.

seed
A start number that key masking uses to return deterministic data.

sequence generation
A standard data generation technique that generates numeric and date values in a sequence.
**set of values generation**
A standard data generation technique that defines a finite set of values to write to the target table.

**shuffle masking**
A data masking technique that masks the data in a column with data from the same column from another row of the table.

**simulation masking**
A masking type that generates a simulation report without masking any data. The report contains policies, rules, masking parameters, qualified ports, and column values that should be masked and their corresponding masked values.

**source string characters**
A data masking technique that masks the selected characters. You can choose to mask the selected source string characters or mask all characters except the source string characters.

**standard generation rule**
A data generation rule that applies a predefined data generation technique.

**substitution masking**
A data masking technique that replaces a column of data with similar but unrelated data.

**template**
Used in data subset.

A reusable component that contains the entities and groups of a specific module that you want to subset based on common parameters.

**Test Data Management (TDM)**
The Informatica solution that bundles Data Subset, Data Generation, and Data Masking to protect sensitive data and create lean nonproduction systems for test and development purposes.

**TDM repository**
A relational database that stores the components that you define in Test Data Manager, such as policies, projects, entities, and data masking rules. The TDM repository stores metadata that you import into Test Data Manager from a source database or from the PowerCenter repository. The TDM repository stores constraints that define relationships between the source tables in a project.

**TDM Server**
Server that runs Test Data Manager and integrates withInformatica application services to perform data subset, data masking, and data discovery operations.

**Test Data Manager**
The web-based user interface that you use to configure and run data subset, data masking, and discovery operations.
value cascade

A cascade type that masks the column values in different tables with the same value.
INDEX

A
abort workflows 166
Actions menu
Test Data Manager 11
ad hoc
generation plan 163
ad hoc generation rules
creating 121
description 120
Administrator view
description 17
advanced masking
description 85
parameters 86
advanced options
plan settings 153
advanced rules
assigning rule to columns 75
creating 30
description 29
application services
TDM 3
architecture
TDM 2
asset linking
create link 38
delete link 39
assign
advanced 77
advanced rule
assignment 77
custom generation 123
assigning
rules in policies to columns in an project source 75
assignments
deleting 78
multiple rules to a column 76
overriding a rule 75
rules to columns 74, 121
auto cascades
overview 45

B
blurring
random masking 96

C
calculated masking
description 87

cascades
auto 45
value 44
CLI
ilmcmd description 176
columns
cascades 44
sensitivity 44
commit interval
workflow properties 152
commit type
workflow properties 150, 152
components
data generation 117
data masking 73
conditional
constraints 59
connections
Administrator view 17
configuring workflows 147
plan settings 149
TDM 4
to databases 5
to repositories 6
to services 4
constraint
major and minor severity 57
constraint-based loading
workflow properties 150, 152
constraints
adding 59
reviewing changes 65
content panel
Test Data Manager 11
copying plans 163
creating
entities 66
credit card masking
description 88
criteria
subset and masking 155
custom generation
assign 123
custom generation rules
description 120
custom mapplet
masking rule 85
parameters 85
custom masking
parameters 85
dashboards
Test Data Manager 13
data discovery
nonrelational sources 42
overview 41
rules and guidelines 42
task flow 46
TDM process 6
data discovery profiles
creating 51
deleting 53
ing
editing 52
data domain
default masking rules 31
data domain definitions
data patterns 33
metadata patterns 33
data domain profile
sampling options in 49
data domains
assigning to multiple columns 49
copying 34
creating 34
discovering 48
exporting and importing 37
manually adding to columns 49
options 33
overview 31
status 33
data expressions
patterns 33
data generation
expression generation 133
random generation 136
sequence generation 143
set of values 144
ad hoc rules 120
components 117
creating ad hoc rules 121
creating standard rules 118
custom rules 120
deleting rules 119
dictionary generation 120
editing rules 119
effective dates generation parameters 131
exception test data 127
overview 116
parameters 127
plans 124
rules 125
standard rules 118
task flow 117
techniques 125, 126
data generation parameters
random generation 134, 135
reference lookup 141
data masking
assigning multiple rules to a column 76
assigning rules in policies to columns 75
components 73
creating advanced rules 30
creating rules from a mapplet 30
creating standard rules 28
disabling at the plan level 156
masking techniques list 81
data masking (continued)
metadata and data patterns 32
overview 72
plans 146
repeatable output 83
rules 81
rules overview 27
standard rules 28
task flow 73
techniques 81
data masking plans
creating 161
data patterns
data domain definitions 33
TDM data domains 32
data subset
creating groups 68
deleting groups 68
deleting templates 70
entities 66
overview 60
plans 71, 78, 146
process 7
task flow 61
templates 70
data subset plans
creating 161
databases
connections to 5
TDM 4
date format string
workflow properties 152
date sequence generation
parameters 143
date values
key masking 94
random masking 98
default rule
TDM data domains 31
Define view
Test Data Manager 20
definitions
TDM repository 24
delete
ilmcmd command 178
deleting 164
details panel
Test Data Manager 11
dictionary connections
plan settings 149
relational and flat file 150
dictionary generation
description 128
parameters 129
dictionary location
flat files 150
discover
discover
columns 43
profiles 43
tables 43
Discover view
Test Data Manager 20
discovering
data domains 48
primary key 46
domains
- assigning to multiple columns 49
- exporting and importing 37
- duplicate data domains 34
- policies 40

E
- editing entities 66
- effective dates generation
description 130
- email masking
description 88
- parameters 89
- encryption masking
description 90
- entities
  - circular relationships 61
  - creating 66
  - deleting 67
  - editing 66
  - exporting TDM 67
  - importing TDM 67
- overview 61
- entity
  - TDM projects 21
- entity discovery
description 41
- entity profiles
  - inference options 47
- examples
  - TDM 8
- export
  - ilmcmd command 179
  - exporting entities 67
  - groups 68
  - policy 37
  - TDM projects 23
- expression
cascade 77
- Expression Builder
  - overview 17
- expression generation
  - parameters 133
  - description 132
- expression masking
  - description 91
  - parameters 91
- rules and guidelines for 91

F
- filter criteria
  - disabling masking by column 156
- filtering data
  - Test Data Manager 12
- foreign key constraints 59

G
- generate staging DDL
  - workflow properties 152
- generating
  - PowerCenter mappings 78
  - workflows 78, 124
- generation rules
  - assigning 123
  - deleting 119
  - description 117
  - editing 119
- generation techniques
dictionary 128
- effective dates 130
- expression 132
- random 133
- reference lookup 140
- sequence 142
- set of values 144
- groups
  - creating 68
  - deleting 69
  - editing 68
  - exporting TDM 68
  - importing TDM 69
- overview 67

H
- high availability recovery
  - workflow properties 152
- high precision
  - workflow properties 152

I
- ilmcmd
  - command line help 177
- ilmcmd commands
  - delete 178
  - entering options and arguments 177
  - export 179
  - import 181
  - overview 176
  - search 182
  - syntax for 178
  - workflow 183
- import
  - ilmcmd command 181
- importing
  - entities 67
  - groups 69
  - metadata from PowerCenter into TDM 25
  - plans 163
  - profiles 52
  - TDM projects 24
- importing metadata
  - from PowerCenter into Test Data Manager 25
- index
  - disabling in plan 153
  - inference options
    - entity profiles 47
    - integers
    - masking large integers 75
IP address masking
description 92

classic masking
description 52

date masking
description 94

K
key masking
date values 94
description 92
masking numeric values 94
masking string values 94
result string replacement characters 93
source string characters 93

L
limit
data subset plan 155
locale
plan settings 153
logging in
Test Data Manager 18
logical primary keys
creating 58

M
major severity
configuring constraints 57
mandatory columns
description 30
mapplets
assigning rule to columns 75
creating rules from 30
TDM 78
mark changes review complete
entity changes 65
mask format
description 92
masking
disabling by column 155
large integers 75
masking options
blurring 96
mask format 92
range 96
result string replacement characters 93
seed 83
source string characters 93
masking parameters
advanced masking 96
date key masking 94
date random masking 96
email masking 89
expression masking 91
numeric key masking 94
numeric random masking 99
random strings 99
string key masking 94
substitution 102
masking rule
custom mapplet 85
masking techniques (continued)
email 88
encryption 90
expression 91
IP address 92
key masking 92
lists 81
nullification 95
phone 95
random 96
Social Security number 101
substitution 102
URL 103
maximum confidence percent
ingference options 47
maximum foreign keys returned
ingference options 47
metadata patterns
data domain definitions 33
TDM data domains 32
minimum confidence percent
ingference options 47
minor severity
configuring constraints 57
Monitor
details 170
jobs 170, 171
logs 171
tasks 171
Monitor overview
description 169
Test Data Manager 16

N
nullification masking
description 95
numeric random generation
parameters 136
numeric sequence generation
parameters 143
numeric values
key masking 94
random masking 99

O
overriding a rule
data masking 75
overview
data discovery 41
data generation 116
data masking 72
groups 87
TDM 1
Overview view
Test Data Manager 13

P
parameters
data generation 125, 127
data masking 81
partitioning
properties 161
rule assignment
multiple rules to a column 76
rule assignments
description 74, 121
overriding the rule 75
rule simulation
preview rule output 29
simulator 28
standard rule 28
rules
ad hoc generation rules 120
adding to policies 30
advanced 29
assigning multiple rules to a column 76
assigning to columns 74, 75, 121
creating from a mapplet 30
custom generation rules 120
deleting column assignments 76
exporting and importing 37
from PowerCenter mapplets 30
not available for assignment 75, 122
overview 27
standard generation rules 118
standard masking rules 28
rules and guidelines
expression masking 91

S
sampling options
data domain profile 49
search
ilmcmd command 182
seed 83
sensitivity
for ports 44
sensitivity levels
data domains 33
sequence generation
description 142
numeric 143
services
connections to 4
session
details 168
sessions
tab 167
set of values generation
parameters 144
description 144
severity
configuring constraints 57
similar value columns
deleting the group 45
description 44
editing 45
setting 44
SIN numbers
masking Social Insurance numbers 101
source settings
plan settings 156
source string characters
key masking 93
random masking 93
sources
TDM 24
special format masking
Social Insurance numbers 101
special masking techniques
credit card 88
e-mail 88
IP address 92
phone 95
Social Security number 101
URL 103
SSN masking
description 101
standard generation rules
creating 118
description 118
standard masking rules
description 28
standard rules
creating 28
start digit
Social Insurance numbers 101
status
data domains 33
stop on error
workflow properties 152
string values
key masking 94
random data masking 99
substitution masking
description 102
parameters 102
syntax
ilmcmd commands 178

T
table classification
description 56
tables
classification 56
tables per mapping
description 153
targets
TDM 24
task flow
data subset 61
TDM
application services 3
architecture 2
connections 4
database connections 5
databases 4
description 15
example 8
overview 1
process overview 5
repository connections 5
service connections 4
sources 24
targets 24
tools 3
TDM processes
data discovery 6
TDM projects
components 21
managing 22
owner 22
TDM repository
definitions 24
techniques
  data generation 125, 126
  data masking 81
templates
  creating 70
  deleting 70
  description 69
  editing 70
  TDM projects 21
Test Data Manager
  Administrator view 17
  filtering data 12
  logging in to 18
  Monitor view 16
  Overview view 13
  Policies view 15
  Projects view 15
  reports 172
  user interface overview 10
  views 11
thinkmaps
  for policy assignments 172
  for rule assignments 172
tools
  TDM 3
trim spaces before comparison
  inference options 47

U
URL masking
  description 103

V
  value cascades
    deleting 45
    description 44
    editing 45
    setting 44

W
  workflow
    execute 165
  workflow executions
    properties panel 167
  Workflow Executions view
    plan settings 166
    Sessions tab 167
    tasks 166
  Workflow view
    plan settings 165
    workflows
      aborting 166
      connections for 147
      generating 78, 124
      generating data subset 71
      generating from plans 164
      ilmcmd command 183
      properties in plan settings 160, 152
      recovering 166