



---

# Reference Data Management




## Server Guide

Release 4.8.0  
B035-0000-9711  
December 2022

# Copyrights or Trademarks

All copyrights and trademarks used in Teradata documentation are the property of their respective owners. For more information, see [Section : “Teradata Trademark and Trademark Attributions.”](#)

## Product Safety

Safety Type	Description
	Indicates a situation which, if not avoided, could result in damage to property, such as to equipment or data, but not related to personal injury.
	Indicates a hazardous situation which, if not avoided, could result in minor or moderate personal injury.
	Indicates a hazardous situation which, if not avoided, could result in death or serious personal injury.

## Warranty Disclaimer

**Except as may be provided in a separate written agreement with Teradata or required by applicable law, the information available from the Teradata Documentation website or contained in Teradata information products is provided on an "as-is" basis, without warranty of any kind, either express or implied, including the implied warranties of merchantability, fitness for a particular purpose, or noninfringement.**

The information available from the Teradata Documentation website or contained in Teradata information products may contain references or cross-references to features, functions, products, or services that are not announced or available in your country. Such references do not imply that Teradata Corporation intends to announce such features, functions, products, or services in your country. Please consult your local Teradata Corporation representative for those features, functions, products, or services available in your country.

The information available from the Teradata Documentation website or contained in Teradata information products may be changed or updated by Teradata at any time without notice. Teradata may also make changes in the products or services described in this information at any time without notice.



## Feedback

To maintain the quality of our products and services, e-mail your comments on the accuracy, clarity, organization, and value of this document to: [docs@teradata.com](mailto:docs@teradata.com).

Any comments or materials (collectively referred to as "Feedback") sent to Teradata Corporation will be deemed nonconfidential. Without any payment or other obligation of any kind and without any restriction of any kind, Teradata and its affiliates are hereby free to (1) reproduce, distribute, provide access to, publish, transmit, publicly display, publicly perform, and create derivative works of, the Feedback, (2) use any ideas, concepts, know-how, and techniques contained in such Feedback for any purpose whatsoever, including developing, manufacturing, and marketing products and services incorporating the Feedback, and (3) authorize others to do any or all of the above.

## Teradata Trademark and Trademark Attributions

Teradata, BYNET, Claraview, Covalent, DecisionCast, IntelliBase, IntelliCloud, IntelliFlex, IntelliSphere, nPath, QueryGrid, SQL-MapReduce, Stacki, "Teradata" logo, Teradata Analytics Platform, Teradata Decision Experts, "Teradata Labs" logo, Teradata ServiceConnect, and Teradata Vantage are trademarks or registered trademarks of Teradata Corporation or its affiliates in the United States and other countries.

Adaptec and SCSISelect are trademarks or registered trademarks of Adaptec, Inc.

Amazon Web Services, AWS, Amazon Elastic Compute Cloud, Amazon EC2, Amazon Simple Storage Service, Amazon S3, AWS CloudFormation, and AWS Marketplace are trademarks of Amazon.com, Inc. or its affiliates in the United States and/or other countries.

AMD Opteron and Opteron are trademarks of Advanced Micro Devices, Inc.

Apache, Apache Avro, Apache Hadoop, Apache Hive, Hadoop, and the yellow elephant logo are either registered trademarks or trademarks of the Apache Software Foundation in the United States and/or other countries.

Apple, Mac, and OS X all are registered trademarks of Apple Inc.

Axeda is a registered trademark of Axeda Corporation. Axeda Agents, Axeda Applications, Axeda Policy Manager, Axeda Enterprise, Axeda Access, Axeda Software Management, Axeda Service, Axeda ServiceLink, and Firewall-Friendly are trademarks and Maximum Results and Maximum Support are servicemarks of Axeda Corporation.

CENTOS is a trademark of Red Hat, Inc., registered in the U.S. and other countries.

Cloudera and CDH are trademarks or registered trademarks of Cloudera Inc. in the United States, and in jurisdictions throughout the world.

Data Domain, EMC, PowerPath, SRDF, and Symmetrix are either registered trademarks or trademarks of EMC Corporation in the United States and/or other countries.

GoldenGate is a trademark of Oracle.

Hewlett-Packard and HP are registered trademarks of Hewlett-Packard Company.

Hortonworks, the Hortonworks logo and other Hortonworks trademarks are trademarks of Hortonworks Inc. in the United States and other countries.

Intel, Pentium, and XEON are registered trademarks of Intel Corporation.

IBM, CICS, RACF, Tivoli, IBM Spectrum Protect, and z/OS are trademarks or registered trademarks of International Business Machines Corporation.

Linux is a registered trademark of Linus Torvalds.

LSI is a registered trademark of LSI Corporation.

Microsoft, Azure, Active Directory, Windows, Windows NT, and Windows Server are registered trademarks of Microsoft Corporation in the United States and other countries.

NetVault is a trademark of Quest Software, Inc.

Novell and SUSE are registered trademarks of Novell, Inc., in the United States and other countries.

Oracle, OpenJDK, Java, and Solaris are trademarks or registered trademarks of Oracle and/or its affiliates.

QLogic and SANbox are trademarks or registered trademarks of QLogic Corporation.

Quantum and the Quantum logo are trademarks of Quantum Corporation, registered in the U.S.A. and other countries.

Red Hat is a trademark of Red Hat, Inc., registered in the U.S. and other countries. Used under license.

SAP is the trademark or registered trademark of SAP AG in Germany and in several other countries.

SAS and SAS/C are trademarks or registered trademarks of SAS Institute Inc.

Sentinel® is a registered trademark of SafeNet, Inc.

Simba, the Simba logo, SimbaEngine, SimbaEngine C/S, SimbaExpress and SimbaLib are registered trademarks of Simba Technologies Inc.

SPARC is a registered trademark of SPARC International, Inc.

Unicode and the Unicode logo are registered trademarks of Unicode, Inc. in the United States and other countries.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Veritas, the Veritas Logo and NetBackup are trademarks or registered trademarks of Veritas Technologies LLC or its affiliates in the U.S. and other countries.

Other product and company names mentioned herein may be the trademarks of their respective owners.

## Purpose

Welcome to Teradata's Reference Data Management (RDM). Reference Data Management represents a relatively recent variation from the traditional Teradata's Master Data Management (MDM). You can use the RDM user guide to work with Reference Data Management capabilities and its features such as code set management, authorization, hierarchy management and publication of data.

Topics:

- [Reference Data Management](#)
- [About This Book](#)
- [Related Documentation](#)
- [Customer Support](#)
- [Documentation Feedback](#)

## Reference Data Management

Reference Data Management is defined in various ways across industries, analysts, and customers. The “wider” definitions of RDM includes not only Code Set Management; but also, Flat Table Maintenance, Excel Integration with rule validations, Workflow and most importantly Hierarchy Management. Teradata believes that as the industry evolves, Reference Data Management will have a greater emphasis on this wider approach.

Some of the common definitions of Reference Data Management includes the following:

Reference data is data that defines the set of permissible values to be used by other data fields. Reference data gains in value when it is widely re-used and widely referenced. Typically, it does not change overly much in terms of definition (apart from occasional revisions). Reference data often is defined by standards organizations (such as country codes as defined in ISO 3166-1).

Typical examples of reference data includes the following:

- Units of measure
- Country codes
- Corporate codes
- Fixed conversion rates (e.g., weight, temperature, and length)
- Calendar (structure and constraints)

Reference data should be distinguished from master data, which represents key business entities such as customers and materials in all the necessary detail (e.g., for customers: number, name, address and date of account creation). In contrast, reference data usually consists only of a list of permissible values and attached textual descriptions.

[http://en.wikipedia.org/wiki/Reference\\_data](http://en.wikipedia.org/wiki/Reference_data)

For another similar definition, Aaron Zornes of the MDM Institute, one of the leading analysts in the RDM space, explains that:

Reference data equals coded, semantically stable, relatively static data sets shared by multiple constituencies. RDM provides the processes and technologies for recognizing, harmonizing and sharing coded, relatively static data sets for “reference” by multiple constituencies (people, systems, and other master data domains). These systems provide governance, process, security, and audit control around the mastering of reference data. RDM is needed in both operational & analytical MDM use cases where capability often used to provide attributes, hierarchies and KPIs.

<http://0046c64.netsolhost.com/mdm/aboutMDMinstitute.html>

Furthermore, Reference Data Management use cases obviously lend direct support to Analytical or EDW based implementations. In fact many times, the RDM Solution becomes the Enterprise's central hub for all types of reference data. So, Teradata believes that reference data include code sets with and without relationships, flat tables of data (many times jointly managed via Excel), hierarchy management, cross reference management and the supporting workflow and collaboration activities as RDM.

Teradata MDM Reference Data Management user interface is designed for Business users to manage reference data and provides the following key features:

- **Reference Data Management (RDM):** The RDM helps you add, modify, and delete the valid values of an editable code type. You can also modify code type values, add new values to the selected code type, and/or delete existing ones from the Code Master page. In addition to this, you can maintain source values and its mappings to valid values.
- **Model Builder:** Model Builder provides ability to model tables and views using web UI. A RDM based table can be modeled and generated at runtime without server restart which can take part in all RDM processes.
- **Configurable UI:** Configurable UI feature enables to create User Interface on RDM tables at runtime. All kind of data manipulation actions (Create, Edit, Delete, Mass Update and Excel Upload) can be performed using configurable UI.
- **Data Authorization:** Data Authorization allows users of different user groups to have access to different subsets of data.
- **Hierarchy Management:** a hierarchy is defined as any logical grouping of objects/components in the system. Instances of any arbitrary object/component in the system can be grouped together to form a tree structure, thus constituting a hierarchy.

# About This Book

This document describes in detail the primary usage of Teradata's RDM along with detailed description of the user interface, features, and procedures associated with specific business processes.

## Target Audience

This document is intended for System Administrators and Technical Support personnel responsible for configuring RDM and its application users.

## What You Should Know

This document assumes that you have prior experience in working on Unix platforms: Solaris/Linux/AIX, or the Windows 2008 server platforms, as applicable. It also assumes that you have a basic knowledge about creating code values and data load workflow in RDM.

## Document Structure

This book contains the following chapters and Appendixes:

- [Chapter 1: "Introduction."](#) - provides overview of primary use cases of RDM Server
- [Chapter 2: "Enabling Features of RDM."](#) - provides an overview of RDM workflow collaboration, alerts and notifications.
- [Chapter 3: "hRDM Table Maintenance."](#) - provides detailed information on CRUD operations on web UI.
- [Chapter 4: "Code Set Management."](#) - provides detailed description on code set management.
- [Chapter 5: "Hierarchy Management."](#) - provides detailed description on hierarchy management.
- [Appendix A: "Landing Page and Favorites."](#) - provides information about RDM landing page and Manage favorites.
- [Appendix B: "Model Builder."](#) - provides detailed information on using Model Builder UI.
- [Appendix C: "Configurable UI."](#) - provides detailed information on Configurable UI.
- [Appendix E: "Extended Attributes."](#) - provides information on extended attributes feature.
- [Appendix D: "Data Publication and Integration."](#) - provides information on RDM data publication.
- [Appendix F: "Glossary."](#) - provides a list of technical and business terminologies used in RDM

## Changes to This Book

The following changes were made to this book in support of the current release. For a complete list of changes to the product, refer *Master Data Management Release Definition* associated with this release.

Date and Release	Description
2018, 4.2.0	First New Release.
April 2019, 4.3.0	<p>Included code set compare version in <a href="#">Chapter 4: “Code Set Management,”</a></p> <p>Included flattened view hierarchy, copy hierarchy version in <a href="#">Chapter 5: “Hierarchy Management,”</a></p> <p>Included approval group requests and E- signature in <a href="#">Chapter 3: “hRDM Table Maintenance,”</a></p> <p>Included Model builder dashboard, updated View ISG log sections in <a href="#">Appendix B: “Model Builder.”</a></p>
November 2019, 4.4.0	<p>Included Handling of unassigned hierarchy nodes details in <a href="#">Chapter 5: “Hierarchy Management,”</a></p> <p>Included Support of non MDM table for MDM operations details in <a href="#">Appendix B: “Model Builder.”</a></p>
June 2020, 4.5.0	<p>Model Builder UI enhancements updated in <a href="#">Appendix B: “Model Builder.”</a></p> <p>Included Support of non MDM table for MDM operations details in <a href="#">Appendix B: “Model Builder.”</a></p>
March 2021, 4.6.0	<p>Model Builder UI enhancements updated in <a href="#">Appendix B: “Model Builder.”</a></p> <p>Replaced all the screenshots in the guide as per the new UI changes.</p>
March 2022, 4.7.0	<p>Manage Approval UI Enhancements</p> <p>Model Builder UI Enhancements</p> <p>Replaced all the screenshots in the guide as per the new UI changes.</p>
November 2022, 4.8.0	Release Version Updated

## Related Documentation

For more information on RDM, refer the following documents:

- *Master Data Management Release Definition*  
(Master Data Management 4.8.0 Release Definition.pdf)
- *Master Data Management Server Guide*  
(Master Data Management 4.8.0 Server Guide.pdf)

The above Teradata documents are available at: <https://docs.teradata.com>

## To Read The Documentation

To read the .pdf files, you must have Adobe Acrobat Reader, version 4.0 or higher. If you do not have Acrobat Reader on your machine, you can download it from Adobe's Web site at <http://www.adobe.com>.

## Customer Support

Customer support is available at the Teradata customer support Web site (<https://access.teradata.com>), where you can:

- Request shipment of software.
- Download software documentation.
- Submit new issues or cases.
- Track the status of current issues or cases.

## Documentation Feedback

Please share your thoughts and ideas:

- Send feedback to [docs@teradata.com](mailto:docs@teradata.com).
- Navigate to <https://teradata-documentation.ideas.aha.io/ideas/new> and provide your ideas.

# Table of Contents

Purpose .....	i
Reference Data Management .....	i
About This Book .....	iii
Target Audience .....	iii
What You Should Know .....	iii
Document Structure .....	iii
Changes to This Book .....	iv
Related Documentation .....	iv
To Read The Documentation .....	iv
Customer Support .....	iv
Documentation Feedback .....	v

---

## **Chapter 1: Introduction .....**

Managing Code Sets .....	1
UI Table Maintenance & Rules Driven Excel Uploads .....	2
Hierarchy Management .....	3

---

## **Chapter 2: Enabling Features of RDM .....**

Workflow and Collaboration .....	7
Alerts and Notification .....	7
Web Services .....	8

---

## **Chapter 3: RDM Table Maintenance .....**

CRUD Data Maintenance from Web UI .....	9
Configurable UI Sample Screen .....	10
Configurable UI—CRUD Operations .....	11
Excel Upload & Validations .....	19



---

## **Chapter 4: Code Set Management** ..... 61

Instantiate RDM Tables .....	61
Create New Code Set .....	64
Add Code Values .....	68
Edit Code Value .....	71
Set As Default .....	72
Create New Source Systems .....	73
Create Source Mapping .....	75
Load from Excel .....	81
Auto Matching Logic .....	87
Code Set Groups .....	90
Add New Group .....	91
Assign Members (Code Sets) to Group .....	93
Remove Members (Code Sets) Assigned to Group .....	95
Grouping of Code Value Attributes .....	97
Source System Association .....	106
Generate RDM Source Tables and Populate Data .....	107
Create Code Set .....	108
Create Source Table Association .....	110
Create Code Values .....	111
Map Code Values to Source .....	113
Manage Version .....	116
Create Code Set Version .....	117
View Code Set Version .....	119
Manage Authorization .....	120
Enable Authorization for Code Sets .....	120
Manage Authorization for Individual Code Set .....	121
Manage Authorization for Code Set Group .....	122

---

## **Chapter 5: Hierarchy Management** ..... 125

Data Model .....	125
Sample Hierarchy Structure .....	126
Create Hierarchy Tables Through Model Builder .....	126
Generate and Validate Hierarchy Tables .....	127
Hierarchy Management Functionalities .....	131
Manage Hierarchy Objects .....	132
Manage Relationship .....	147

Hierarchy Manager .....	156
Create Hierarchy .....	157
Hierarchy on Custom Tables .....	160
Edit Hierarchy .....	166
Create Version .....	168
View Version .....	168
Promote Hierarchy .....	170
Delete Hierarchy and Hierarchy Version .....	176
Manage Custom Actions .....	177
View Hierarchy .....	179
Compare Hierarchy .....	179
Hierarchy Viewer .....	183
Export Hierarchy .....	195
Import Hierarchy .....	197
Merge Hierarchy .....	201
Hierarchy Logs .....	207
Manage Rollup Attributes .....	207
Hierarchy Statistics .....	216
Manage Cross Reference .....	218

---

## **Appendix A: Landing Page and Favorites .....**

Managing RDM Landing Page .....	223
Create Page .....	223
Assign Landing Page .....	224
User Favorites .....	227
Create Favorite at User Level .....	227
Create Favorite at Role Level .....	228
Manage Favorites .....	229

---

## **Appendix B: Model Builder .....**

Introduction .....	233
User Security Access to Metadata of Generated Tables .....	235
Model Builder UI .....	235
Create or Edit Model .....	236
Create RDM Based View .....	258
Auto Generate View .....	260
Validate Models .....	262
Run ISG on Validated Model .....	264

View ISG Logs . . . . .	264
API to Validate and Generate ISG . . . . .	266
Modify Table Primary Key . . . . .	268
Delete Model and Delete Model Metadata . . . . .	271
Export Model . . . . .	275
Import from X-Document . . . . .	276
Manage Dictionaries . . . . .	278
Manage Templates . . . . .	280
Manage Metadata Import . . . . .	284
<hr/>	
<b>Appendix C: Configurable UI . . . . .</b>	<b>295</b>
Generating Configurable UI . . . . .	295
Generate Simple Configurable UI . . . . .	295
Advanced Configurable UI . . . . .	297
<hr/>	
<b>Appendix D: Extended Attributes . . . . .</b>	<b>307</b>
Manage Extended Attributes . . . . .	307
Step 1: Studio or Model Builder Configuration . . . . .	307
Step 2: RDM UI Configurations: Manage Extended Attributes . . . . .	309
<hr/>	
<b>Appendix E: Data Publication and Integration . . . . .</b>	<b>315</b>
Data Publishing and Integration . . . . .	315
Add Publication Objects . . . . .	315
Publish Table . . . . .	317
View Published Data . . . . .	318
<hr/>	
<b>Appendix F: Glossary . . . . .</b>	<b>321</b>
Technical & Business Glossary . . . . .	321

# List of Figures

Figure 1: Local to Global Mapping Management . . . . .	2
Figure 2: Configurable UI—Account Table . . . . .	3
Figure 3: Hierarchy Viewer . . . . .	4
Figure 4: Hierarchy Viewer—Version . . . . .	5
Figure 5: Compare Hierarchy . . . . .	6
Figure 6: RDM Alerts . . . . .	8
Figure 7: Table Activity Search . . . . .	10
Figure 8: Configurable UI—Sample Account Table . . . . .	11
Figure 9: Create Record . . . . .	11
Figure 10: Create Record . . . . .	12
Figure 11: Create Record . . . . .	12
Figure 12: Configurable UI—Record Created . . . . .	13
Figure 13: Copy Record . . . . .	13
Figure 14: Configurable UI—Record Copied . . . . .	14
Figure 15: Edit Record . . . . .	15
Figure 16: Edit Record . . . . .	15
Figure 17: Configurable UI—Mass Update . . . . .	16
Figure 18: Confirmation Webpage Dialog . . . . .	16
Figure 19: Mass Update . . . . .	17
Figure 20: Configurable UI . . . . .	17
Figure 21: Configurable UI—Delete Record . . . . .	18
Figure 22: Confirmation Webpage Dialog . . . . .	18
Figure 23: Configurable UI—Record(s) Deleted . . . . .	19
Figure 24: Configurable UI . . . . .	19
Figure 25: Upload File . . . . .	20
Figure 26: Scheduler Details . . . . .	20
Figure 27: Upload File—Sample Excel File . . . . .	21
Figure 28: View Reports . . . . .	21
Figure 29: Configurable UI—Record(s) Uploaded . . . . .	22
Figure 30: Manage Business Rule . . . . .	24
Figure 31: Create Business Rule . . . . .	25
Figure 32: Rule Editor Wizard—Error Data . . . . .	26

Figure 33: Rule Editor Wizard—From Context . . . . .	27
Figure 34: Rule Editor Wizard—Join Context . . . . .	28
Figure 35: Rule Editor Wizard—Search Context . . . . .	29
Figure 36: Rule Editor Wizard—Search Context Dynamic Expression . . . . .	30
Figure 37: Rule Editor Wizard—Action Type . . . . .	31
Figure 38: Summary Rule Expression . . . . .	32
Figure 39: Rule Editor Wizard—Custom Expression . . . . .	33
Figure 40: Manage Business Rules . . . . .	34
Figure 41: Approval Details . . . . .	35
Figure 42: Manage Approvals . . . . .	37
Figure 43: Approval Details . . . . .	38
Figure 44: Confirmation Webpage Dialog . . . . .	38
Figure 45: Manage Approvals . . . . .	40
Figure 46: Column Level Approval . . . . .	41
Figure 47: Create Approval Rules . . . . .	42
Figure 48: Assignment Approval Rules . . . . .	43
Figure 49: Assignment Approval Rules . . . . .	44
Figure 50: Edit Record . . . . .	44
Figure 51: Configurable UI . . . . .	45
Figure 52: Edit Record . . . . .	45
Figure 53: Configurable UI . . . . .	46
Figure 54: Create Channel Table Record . . . . .	46
Figure 55: Approval Monitor . . . . .	47
Figure 56: Create Record . . . . .	47
Figure 57: Configurable UI . . . . .	48
Figure 58: Configurable UI—Account Table . . . . .	49
Figure 59: Upload File . . . . .	49
Figure 60: Upload File—Sample Excel File . . . . .	50
Figure 61: View Reports . . . . .	50
Figure 62: Upload Status . . . . .	51
Figure 63: Record Edit Form . . . . .	52
Figure 64: Record Edit Form . . . . .	52
Figure 65: Upload Status . . . . .	53
Figure 66: View Reports . . . . .	54
Figure 67: Approval Inbox . . . . .	54
Figure 68: Approval Inbox . . . . .	54

Figure 69: View Records . . . . .	55
Figure 70: Email Notification. . . . .	56
Figure 71: Approval Inbox . . . . .	57
Figure 72: Configurable UI—Account Table. . . . .	57
Figure 73: Approval Inbox Data. . . . .	58
Figure 74: Makers Inbox . . . . .	59
Figure 75: Model Builder . . . . .	62
Figure 76: Manage Template . . . . .	63
Figure 77: Manage Template . . . . .	63
Figure 78: Model Builder ISG Logs. . . . .	64
Figure 79: Model Builder . . . . .	64
Figure 80: Reference Data Dashboard . . . . .	65
Figure 81: Manage Lookup Data . . . . .	66
Figure 82: Add Code Set . . . . .	67
Figure 83: Manage Lookup Data . . . . .	68
Figure 84: Manage Code Values . . . . .	69
Figure 85: Add Code Value . . . . .	70
Figure 86: Manage Code Values . . . . .	71
Figure 87: Edit Code Value . . . . .	72
Figure 88: Manage Code Values . . . . .	73
Figure 89: Manage Lookup Data . . . . .	73
Figure 90: Manage Code Values . . . . .	74
Figure 91: Add Code Value . . . . .	75
Figure 92: Manage Lookup Data . . . . .	76
Figure 93: Confirmation Pop-up. . . . .	76
Figure 94: Manage Lookup Data . . . . .	77
Figure 95: Manage Mapping . . . . .	78
Figure 96: Multi Edit Mapping. . . . .	79
Figure 97: Manage Mapping . . . . .	80
Figure 98: Manage Mapping . . . . .	81
Figure 99: Manage Lookup Data . . . . .	82
Figure 100: Manage Mapping . . . . .	83
Figure 101: Upload Mapping Values . . . . .	84
Figure 102: Sample Excel File—Brand Master Table . . . . .	85
Figure 103: View Reports. . . . .	86
Figure 104: Manage Lookup Data . . . . .	86

Figure 105: Matching Profiles . . . . .	87
Figure 106: Matching Dashboard . . . . .	88
Figure 107: Survivorship Workbench . . . . .	89
Figure 108: Merge Match Records . . . . .	89
Figure 109: Manage Lookup Data . . . . .	90
Figure 110: Manage Lookup Data . . . . .	91
Figure 111: Manage Groups . . . . .	92
Figure 112: Reference Data Dashboard . . . . .	93
Figure 113: Manage Lookup Data . . . . .	94
Figure 114: Manage Group . . . . .	94
Figure 115: Manage Lookup Data . . . . .	95
Figure 116: Manage Group . . . . .	96
Figure 117: Manage Group . . . . .	97
Figure 118: Confirmation—Webpage Dialog . . . . .	97
Figure 119: Attribute Groups Dashboard . . . . .	98
Figure 120: Manage Attribute Groups . . . . .	98
Figure 121: Manage Attribute Groups . . . . .	99
Figure 122: Manage Attribute Groups . . . . .	100
Figure 123: Attribute Groups Dashboard . . . . .	101
Figure 124: Manage Attribute Groups . . . . .	101
Figure 125: Manage Attribute Groups . . . . .	102
Figure 126: Attribute Groups Dashboard . . . . .	103
Figure 127: Add Code Value . . . . .	104
Figure 128: Edit Code Value . . . . .	105
Figure 129: Manage Code Values—View Details . . . . .	106
Figure 130: Manage Template . . . . .	107
Figure 131: Create Source Map . . . . .	108
Figure 132: Source Map . . . . .	108
Figure 133: Add Code Set . . . . .	109
Figure 134: Manage Lookup Data . . . . .	109
Figure 135: Associated Tables for Code Set . . . . .	110
Figure 136: Manage Code Values . . . . .	111
Figure 137: Edit Code Value . . . . .	112
Figure 138: Edit Code Value—Add Source Data . . . . .	112
Figure 139: Edit Code Value—Edit Source Data . . . . .	113
Figure 140: Add Mapping . . . . .	114

Figure 141: Source Data Lookup .....	115
Figure 142: Manage Code Values .....	116
Figure 143: Manage Version .....	117
Figure 144: Create Version.....	118
Figure 145: Manage Version .....	119
Figure 146: Manage Lookup Data .....	119
Figure 147: Manage Authorization.....	121
Figure 148: Manage Authorization—Code Set .....	122
Figure 149: Manage Authorization—Code Set Group.....	123
Figure 150: Hierarchy Nodes and Node Relations for Three Level Hierarchy .....	126
Figure 151: Model Builder.....	127
Figure 152: Manage Template .....	128
Figure 153: Model Builder ISG Logs.....	128
Figure 154: Model Builder ISG Log Details .....	129
Figure 155: Model Builder ISG Logs.....	129
Figure 156: Model Builder—Generated Models .....	130
Figure 157: Email Notification—Model Builder ISG .....	130
Figure 158: Configurable UI .....	131
Figure 159: Manage Hierarchy Objects .....	133
Figure 160: Create Hierarchy Objects .....	134
Figure 161: Manage Hierarchy Objects .....	134
Figure 162: Configurable UI .....	135
Figure 163: Email Notification-Rename Workflow.....	135
Figure 164: Load Hierarchy Data.....	136
Figure 165: Hierarchy Objects—Excel Upload .....	136
Figure 166: Manage Hierarchy Objects .....	137
Figure 167: Edit Hierarchy Object .....	138
Figure 168: Customize Icon .....	140
Figure 169: Hierarchy Viewer .....	140
Figure 170: Manage Extended Attributes.....	142
Figure 171: Add Extended Attribute .....	144
Figure 172: Process Flow of Create Extended Attribute .....	144
Figure 173: Configurable UI .....	145
Figure 174: Hierarchy Manager .....	146
Figure 175: Manage Relationships.....	146
Figure 176: Relationship Details .....	146



Figure 177: Manage Relationships . . . . .	147
Figure 178: Create Relationship . . . . .	148
Figure 179: Manage Relationships . . . . .	149
Figure 180: Load Hierarchy Relationship Data . . . . .	149
Figure 181: Hierarchy Relations. . . . .	150
Figure 182: Hierarchy Relationship—Excel Upload . . . . .	150
Figure 183: Hierarchy Relationship . . . . .	151
Figure 184: Manage Relationships . . . . .	151
Figure 185: Select Parent Data . . . . .	152
Figure 186: Select Child Data. . . . .	153
Figure 187: Select Parent Data . . . . .	154
Figure 188: Selected Parent Object Data . . . . .	155
Figure 189: Manage Relationships-Delete Option . . . . .	156
Figure 190: Hierarchy Manager . . . . .	158
Figure 191: Create Hierarchy . . . . .	158
Figure 192: Manage Publication Objects . . . . .	159
Figure 193: Create Hierarchy—Manage Relationships . . . . .	160
Figure 194: Hierarchy Manager . . . . .	160
Figure 195: Manage Hierarchy Objects . . . . .	161
Figure 196: Create Hierarchy Objects . . . . .	162
Figure 197: Manage Hierarchy Objects . . . . .	163
Figure 198: Create Relationship . . . . .	164
Figure 199: Manage Relationships . . . . .	164
Figure 200: Create Hierarchy . . . . .	165
Figure 201: Create Hierarchy—Manage Relationships . . . . .	165
Figure 202: Hierarchy Viewer . . . . .	166
Figure 203: Edit Hierarchy . . . . .	167
Figure 204: Create Version. . . . .	168
Figure 205: Manage Versions. . . . .	169
Figure 206: Manage Versions. . . . .	169
Figure 207: Hierarchy Version Viewer:<version>. . . . .	170
Figure 208: Approval Details . . . . .	172
Figure 209: Manage Versions. . . . .	173
Figure 210: Hierarchy Manager . . . . .	173
Figure 211: Manage Versions. . . . .	174
Figure 212: Approval Inbox . . . . .	174

Figure 213: Approval Inbox—View Record . . . . .	175
Figure 214: Manage Scheduler . . . . .	175
Figure 215: Manage Versions . . . . .	176
Figure 216: Scheduler to Delete Hierarchy Version . . . . .	177
Figure 217: Manage Custom Actions . . . . .	177
Figure 218: Create Custom Action . . . . .	178
Figure 219: Hierarchy Viewer—Manage Custom Actions Option . . . . .	179
Figure 220: Compare Hierarchy . . . . .	180
Figure 221: Compare Hierarchy—Compare Metadata . . . . .	181
Figure 222: Compare Hierarchy on Viewer . . . . .	182
Figure 223: Compare Details . . . . .	183
Figure 224: Hierarchy Viewer . . . . .	184
Figure 225: Hierarchy Viewer—Show All Peers . . . . .	187
Figure 226: Hierarchy Viewer—Show Siblings . . . . .	188
Figure 227: Hierarchy Viewer—Show Children . . . . .	189
Figure 228: Hierarchy Viewer—Edit Option . . . . .	190
Figure 229: Hierarchy Viewer—Add Child . . . . .	191
Figure 230: Hierarchy Viewer—Add Sibling . . . . .	192
Figure 231: Hierarchy Viewer—Menu Options . . . . .	193
Figure 232: Hierarchy Viewer—Search . . . . .	194
Figure 233: Hierarchy Viewer—Drag and Drop Wrong Icon . . . . .	195
Figure 234: Export Hierarchy . . . . .	196
Figure 235: Export Hierarchy . . . . .	196
Figure 236: Hierarchy Manager—Import Hierarchy Option . . . . .	197
Figure 237: Import Hierarchy . . . . .	197
Figure 238: View Reports . . . . .	199
Figure 239: Import Hierarchies—Document Template . . . . .	199
Figure 240: Template File—Node Template . . . . .	200
Figure 241: Template File—Relation Template . . . . .	200
Figure 242: Merge Hierarchy . . . . .	202
Figure 243: Select Level . . . . .	202
Figure 244: Merge Hierarchy . . . . .	203
Figure 245: Merge Hierarchy . . . . .	204
Figure 246: Example—Hierarchy Structure Before Merge Process . . . . .	206
Figure 247: Example—Hierarchy Structure After Merge Process . . . . .	206
Figure 248: Email Notification . . . . .	207

Figure 249: Manage Rollup Attributes . . . . .	208
Figure 250: Add Hierarchy Attributes . . . . .	208
Figure 251: Manage Rollup Attributes . . . . .	209
Figure 252: Edit Hierarchy Attribute . . . . .	210
Figure 253: Manage Rollup Attributes . . . . .	210
Figure 254: Hierarchy Manager—Rollup Attributes . . . . .	211
Figure 255: Hierarchy Attributes Details . . . . .	212
Figure 256: Hierarchy Viewer—Rollup Attribute . . . . .	213
Figure 257: Hierarchy Viewer—Rollup Attribute—Drag and Drop . . . . .	214
Figure 258: Hierarchy Viewer—Rollup Attribute—Drag and Drop . . . . .	215
Figure 259: Hierarchy Viewer—Rollup Attribute—Drag and Drop . . . . .	216
Figure 260: Stats Manager . . . . .	217
Figure 261: Hierarchy Statistics . . . . .	217
Figure 262: Manage Cross Reference. . . . .	218
Figure 263: Select Parent Data . . . . .	219
Figure 264: Selected Parent Object Data—Linked Child Object Data . . . . .	220
Figure 265: View Additional Attributes Value—Cross Reference . . . . .	221
Figure 266: Manage Pages . . . . .	223
Figure 267: Add Page . . . . .	224
Figure 268: User Details—Landing Page. . . . .	225
Figure 269: User Details—Landing Page. . . . .	226
Figure 270: Role Details—Landing Page. . . . .	227
Figure 271: Create Favorite . . . . .	227
Figure 272: Favorites List. . . . .	228
Figure 273: Create Favorite . . . . .	229
Figure 274: Manage Favorites . . . . .	230
Figure 275: Organize Favorites . . . . .	231
Figure 276: Model Builder Stages . . . . .	234
Figure 277: Activity Details—User Security . . . . .	235
Figure 278: Model Builder . . . . .	236
Figure 279: Model Builder . . . . .	237
Figure 280: Object Details—Advanced Setting-View Details. . . . .	238
Figure 281: Object Details—Advanced Setting-Staging Details . . . . .	239
Figure 282: Model Builder—Advanced Setting-Configuration Details. . . . .	240
Figure 283: Model Builder—Advanced Setting-Configuration Details. . . . .	241
Figure 284: Model Builder—Advanced Setting-Configuration Details. . . . .	242

Figure 285: Model Builder—Columns Tab . . . . .	243
Figure 286: Dictionary Data Type . . . . .	244
Figure 287: Constraint . . . . .	245
Figure 288: Valid Value—Constant . . . . .	246
Figure 289: Valid Value—Lookup . . . . .	246
Figure 290: Valid Value—Reference . . . . .	247
Figure 291: Model Builder—Keys Tab . . . . .	249
Figure 292: Model Builder—Links Tab . . . . .	249
Figure 293: Model Builder . . . . .	250
Figure 294: Upload File . . . . .	251
Figure 295: Upload File—Error Message . . . . .	251
Figure 296: Model Builder Template—Simple Template . . . . .	252
Figure 297: Model Builder Template—Advanced Template . . . . .	252
Figure 298: Model Builder Template—Non MDM Template . . . . .	253
Figure 299: Model Builder—View SQL . . . . .	259
Figure 300: Auto Generate Views UI . . . . .	260
Figure 301: Auto Generate Views UI—View Generation Logs . . . . .	261
Figure 302: Auto Generate Views UI—View Generation Logs . . . . .	262
Figure 303: Model Builder . . . . .	263
Figure 304: Model Builder . . . . .	263
Figure 305: Model Builder . . . . .	264
Figure 306: Model Builder ISG Logs . . . . .	264
Figure 307: Model Builder ISG Log Details . . . . .	265
Figure 308: Model Builder ISG Logs . . . . .	265
Figure 309: Model Builder—Deployed Table . . . . .	266
Figure 310: Model Details—Deployed Table Configurable UI . . . . .	266
Figure 311: Model Builder . . . . .	269
Figure 312: Model Builder . . . . .	269
Figure 313: Model Builder . . . . .	270
Figure 314: Model Builder . . . . .	272
Figure 315: Model Builder . . . . .	273
Figure 316: Model Builder . . . . .	274
Figure 317: Metadata Deletion Logs . . . . .	274
Figure 318: Upload X-Doc File . . . . .	276
Figure 319: Import Models . . . . .	277
Figure 320: Model Builder . . . . .	278

Figure 321: Model Builder—Manage Dictionary Option	279
Figure 322: Manage Dictionary	279
Figure 323: Manage Dictionary—Dictionary Details	279
Figure 324: Manage Dictionary—Dictionary Properties	280
Figure 325: Model Builder—Manage Template Option	280
Figure 326: Manage Template	281
Figure 327: Manage Template—Table Count	282
Figure 328: Model Builder—Generated Models	282
Figure 329: Model Builder—Object Details	283
Figure 330: Model Builder—Columns Tab	283
Figure 331: Model Builder	284
Figure 332: Manage Metadata Import	284
Figure 333: Import Metadata Objects	285
Figure 334: Import Metadata Objects	286
Figure 335: Import Metadata Objects	286
Figure 336: Import Metadata Objects	287
Figure 337: Web Profiler	289
Figure 338: Web Profiler—Values Results	289
Figure 339: Web Profiler—Values Results—Graphs	290
Figure 340: Advanced Options—Primary Key Column(s)	291
Figure 341: Advanced Options—Macro Mapping	292
Figure 342: View and Edit Object Details	293
Figure 343: Manage Metadata Import	294
Figure 344: Metadata Import Log	294
Figure 345: Generate Configurable UI	295
Figure 346: Generate Configurable UI	296
Figure 347: Sample of Simple Configurable UI—Account Table	297
Figure 348: Configurable UI	298
Figure 349: Configurable UIs—Define New UI	298
Figure 350: Configurable UIs—Define Search Columns	299
Figure 351: Configurable UI—Configure Advanced Filter	300
Figure 352: Configurable UIs—Configure Search Columns	301
Figure 353: Configurable UI—Define Result Columns	302
Figure 354: Configurable UIs—Define Result Columns	303
Figure 355: Configurable Uis—Configure Result Columns	304
Figure 356: Configurable UI	305

Figure 357: Preview of Advanced Configurable UI. . . . .	306
Figure 358: Extended Attributes—MDM Studio . . . . .	308
Figure 359: Model Builder—Extended Attribute. . . . .	309
Figure 360: Manage Extended Attributes. . . . .	310
Figure 361: Add Extended Attribute . . . . .	310
Figure 362: Manage Extended Attributes. . . . .	312
Figure 363: Extended Attribute—Configurable UI . . . . .	312
Figure 364: Manage Publication Objects . . . . .	316
Figure 365: Add Publication Object. . . . .	316
Figure 366: Manage Publication Objects . . . . .	317
Figure 367: Publish Table. . . . .	318
Figure 368: Import Metadata Objects. . . . .	319
Figure 369: Generate Configurable UI. . . . .	319
Figure 370: Configurable UI—PUB_Account . . . . .	320

# List of Tables

Table 1: Sample Business Rule Creation Details . . . . .	23
Table 2: Hierarchy Node . . . . .	125
Table 3: Hierarchy Relation . . . . .	125
Table 4: Activities Involved in Create and Manage Hierarchies . . . . .	131
Table 5: Manage Hierarchy Objects UI Fields . . . . .	133
Table 6: SYS_EXT_ATTRBS_MAP Table . . . . .	142
Table 7: SYS_EXT_ATTRBS . . . . .	143
Table 8: Manage Relationship UI Fields . . . . .	147
Table 9: Hierarchy Table Details . . . . .	156
Table 10: Hierarchy Viewer Panels . . . . .	184
Table 11: Hierarchy Viewer—Node Detail Panel Buttons . . . . .	185
Table 12: Hierarchy Rollup Attributes . . . . .	209
Table 13: Temporal Data Type Description . . . . .	242
Table 14: Delete and Delete Metadata Use Cases . . . . .	275
Table 15: Add Extended Attribute Fields . . . . .	311

---

## What's In This Chapter

This chapter provides overview of usage of RDM features.

Topics include:

- [Managing Code Sets](#)
- [UI Table Maintenance & Rules Driven Excel Uploads](#)
- [Hierarchy Management](#)

## Managing Code Sets

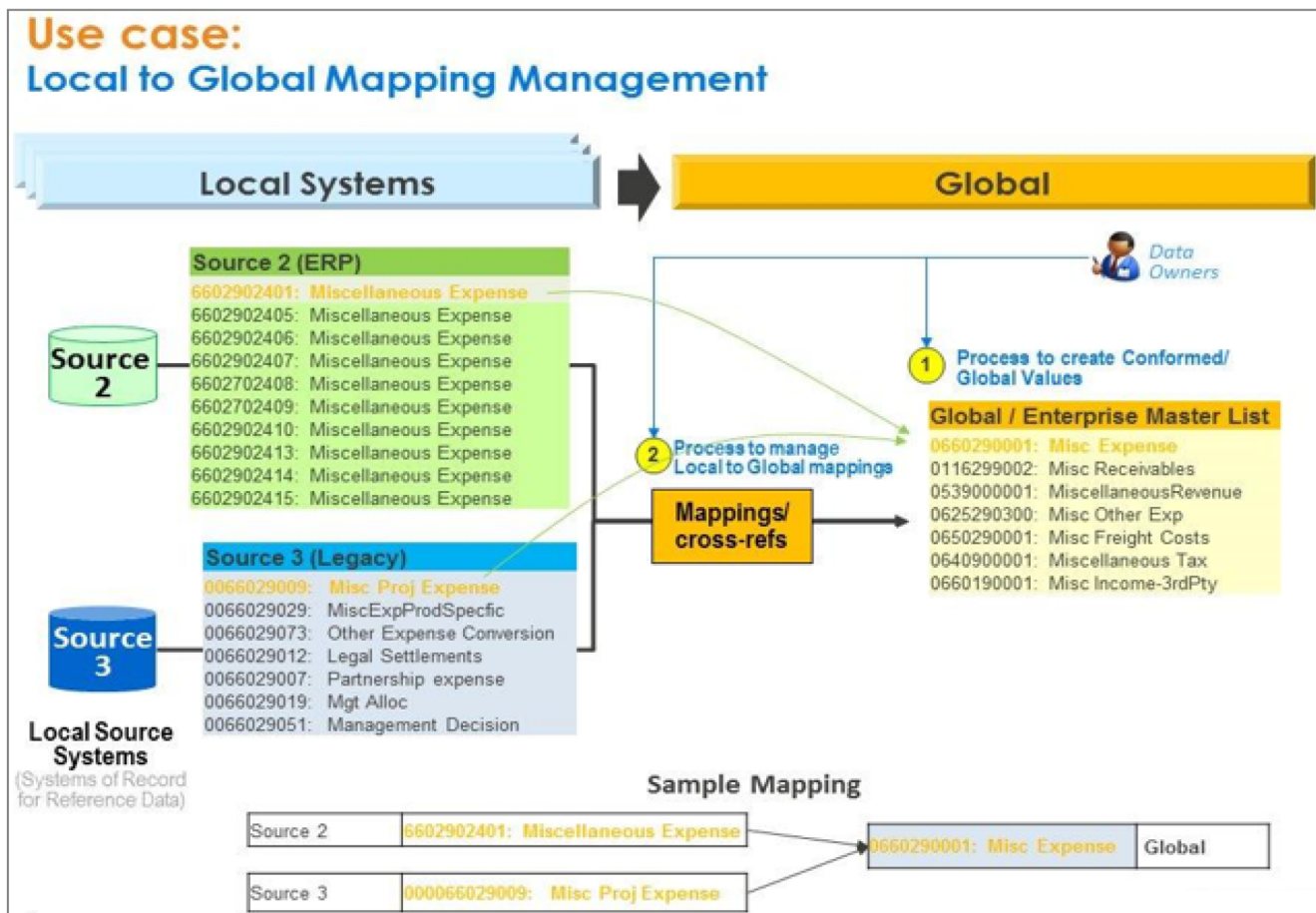
Teradata implementations most commonly involve some aspect of managing code sets (name: value pairs) to enable reference data. Code Sets include valid value lists represented by “lookup tables” in the Data Warehouse. Code Sets could be ISO codes, UOMs, Status Codes, Geo, General Ledger Codes, Reason Codes, and so on. Normally, these codes are maintained in many disparate systems (ERP, CRM, PLM, etc). Managing them at an enterprise level leads to correct rollups and analytics.

At the simplest level, some customers use Teradata RDM's Code Set Management feature to enable business users to maintain and manage the code sets. In these cases, RDM is used simply as a “valid value master”.

More commonly, once values for code sets have been defined at the enterprise level; then other remote or local system values need to be mapped to the “Enterprise Code Values” at the global level. Once identified, Code Sets are harmonized from the multitude of source systems to maintain for Local to Global mappings. Business users need to be the final owners of the code values and their corresponding values. So, the process needs to be simplified and enabled via a web browser Dashboard to manage Code Sets and their Mappings.



Figure 1: Local to Global Mapping Management



Business users are most knowledgeable of these mappings and therefore should own the mappings. However, prior to the advent of RDM solutions, there was no simple, standard way to manage these important tasks. Most commonly, Excel and IT managed home grown solutions were used.

In previous releases, Teradata referred to this use case as “Lookup Reference Data Management” or “LRDM”.

## UI Table Maintenance & Rules Driven Excel Uploads

The second major use case that enables business users to easily maintain tables via a role based web interface with Excel integration. These tables could be RDM, EDW, sandbox or anywhere on the Teradata Platform. Additionally, many times the Tables include attributes that are valid value code sets that were defined in the first use case. Teradata RDM enables Create, Read, Update, Delete (referred to as CRUD) operations for Table Maintenance via any combination of Web UI, Excel Upload/Download and web services. A common framework of role based access control, application/business rules, approvals, notifications

and alerts via dashboards, email, etc., and so on is applied to any of these forms of maintenance. Many times, Excel was the common method to manage, but the data still had to get updated to Teradata through many processes. So, the web interface allows users to either—use the UI Maintenance, use Excel, or both.

Figure 2: Configurable UI—Account Table

Account				SEARCH	UPLOAD EXCEL	↑↓	☰	≡	CREATE	⋮	♥	?
<input type="checkbox"/>	AccountId	AccountName	Amount	Department								
<input type="checkbox"/>	6	Back Office Account	90000	Back Office								
<input type="checkbox"/>	7	Branch Account	90000	Branch								
<input type="checkbox"/>	9	Departmental Account	90000	Departmental								

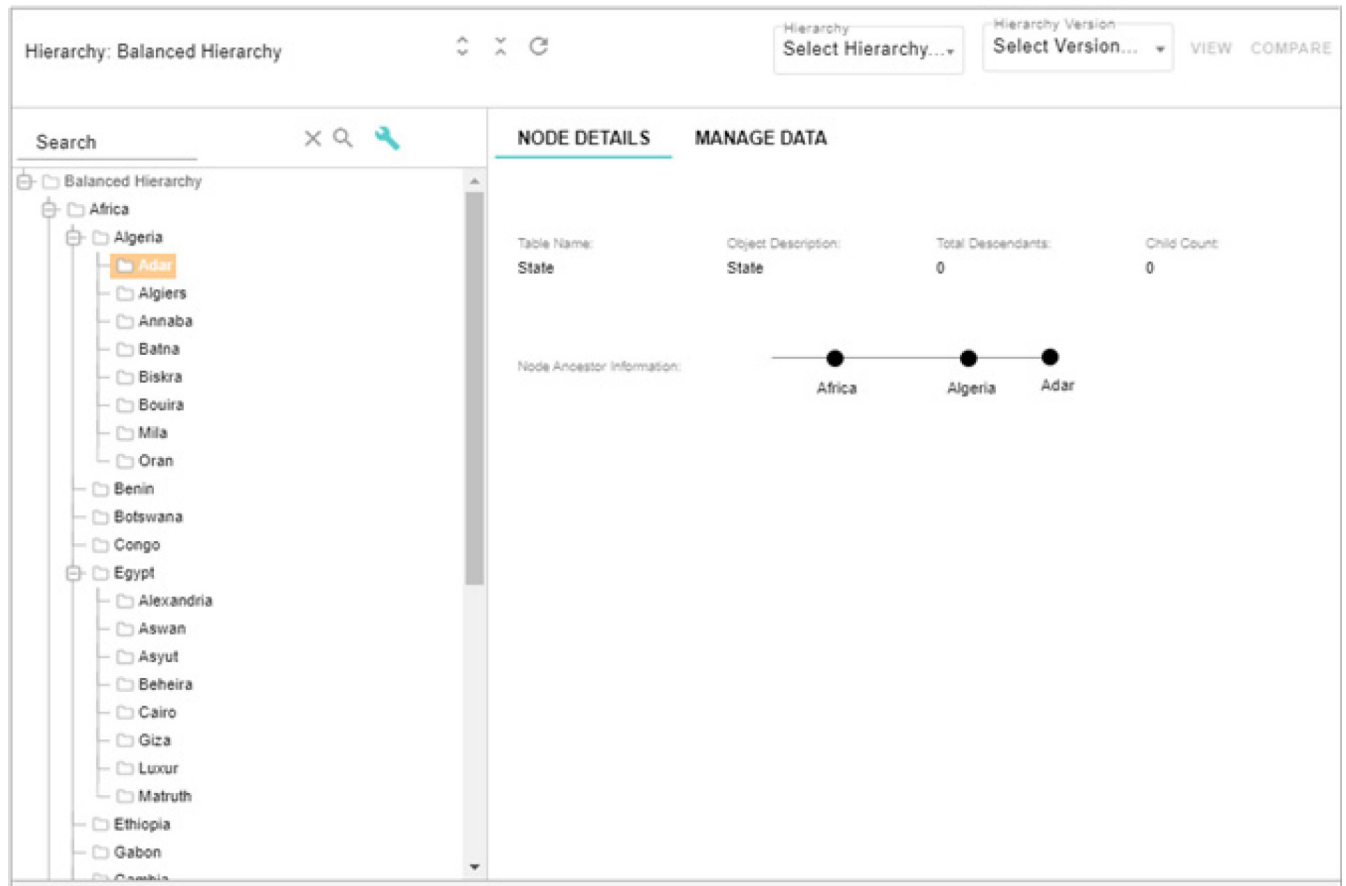
1 - 3 of 3 |< < > >|

## Hierarchy Management

The third major usage involves Hierarchy Management, Versioning, Visualization, Mass Maintenance, etc. Hierarchy Management leverages the Teradata RDM Hierarchy model as the underlying data.

Teradata RDM provides Hierarchy Viewer and Manager to allow users to visualize, search, manage data and relationships and show rollup attributes. [Figure 3](#) displays an example hierarchy and how the tool displays the visualization.

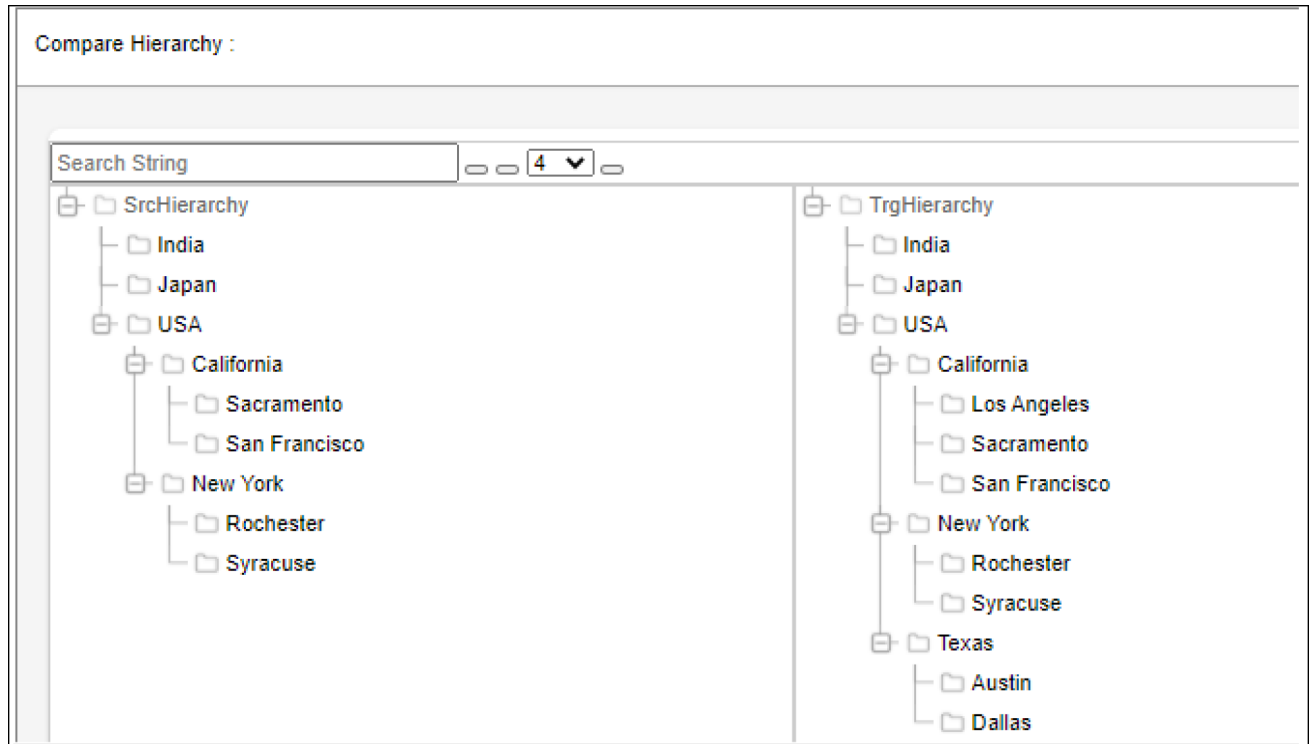
Figure 3: Hierarchy Viewer



Teradata RDM has the ability to create a hierarchy version based on a current hierarchy that has already been created. Creating the version will allow users to make edits to the hierarchy without altering the original hierarchy. In [Figure 3](#), select the hierarchy and hierarchy version from the corresponding dropdown and click View to view the hierarchy version.

Teradata RDM also has the ability to compare hierarchies using the ‘Compare Hierarchy’ function in Hierarchy Manager. This can be done to compare two completely different hierarchies, a version to an original hierarchy, or two versions. [Figure 4](#) displays an example of comparing one hierarchy to an alternate version with the value China, Iran and Japan moved to a different Continent.

Figure 4: Compare Hierarchy



## CHAPTER 2 **Enabling Features of RDM**

---

### **What's In This Chapter**

This chapter provides an overview of RDM workflow collaboration, alerts and notifications.

Topics include:

- [Workflow and Collaboration](#)
- [Alerts and Notification](#)
- [Web Services](#)

## **Workflow and Collaboration**

One of the strong differentiating feature of Teradata RDM is the robust workflow engine. Teradata's solution supports powerful collaboration capabilities. Approvals are the most basic form of workflow and collaboration. Basic approval and escalation workflows are easily configured by business users from the run-time web UI. The solution also includes a workflow engine to build complex workflows. The activities performed within a workflow can range from data manipulation, to data validation, to business rule invocation, etc.

RDM provides capability to define new workflows and/or modify existing workflows with an easy-to-use MDM Studio and configuration web-based screens so that these workflows could be used to mimic or improve existing business processes. Each workflow can be configured to use different roles, different parts of the business entities to be read/edited with a role specific UI and different approval routing, etc. Workflows can be created from an extensible template library that is provided or from scratch, using built-in constructs called nodes like script, stored procedure, decision, branch, macro, event and x-rule among others. Data services can be plugged into these workflows as either stand-alone workflows or incorporated into an overall workflow to orchestrate a business process.

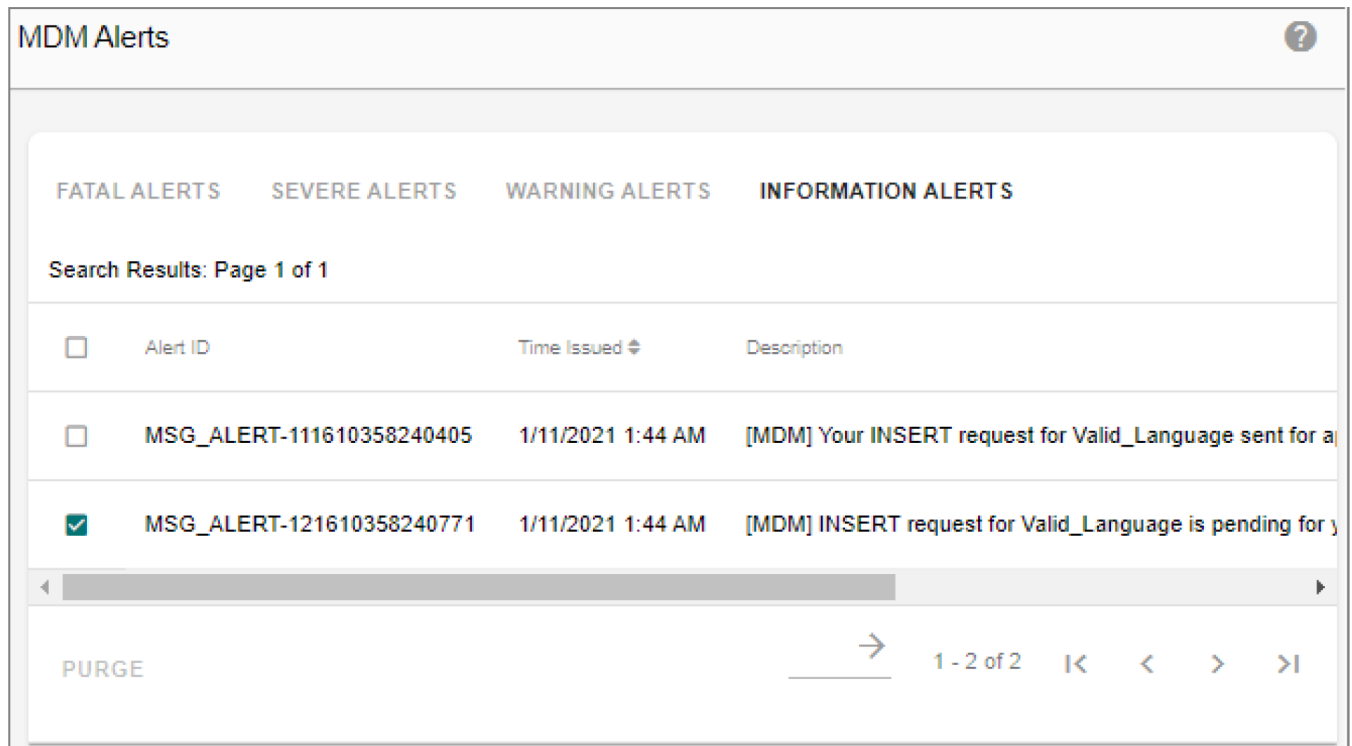
## **Alerts and Notification**

Alerts are a mechanism through which notifications are published to users upon the occurrence of events that they are subscribed to. Alerts can be custom created for any desired RDM system activity. Alerts can be configured to be web only, email or both. Common sources of alerts are RDM's data load workflow, error correction workflows for excel, approval notifications, etc.

RDM supports four categories of alerts:

- Information Alerts - a message indicating that some activity has occurred.
- Warning Alerts - processing continues but there is some issue, which should be checked.
- Severe Alerts - a problem which prevents some action from completing its processing.
- Fatal Alerts - Fatal issue, all processing stopped.

Figure 5: RDM Alerts



**MDM Alerts**

FATAL ALERTS   SEVERE ALERTS   WARNING ALERTS   **INFORMATION ALERTS**

Search Results: Page 1 of 1

<input type="checkbox"/>	Alert ID	Time Issued	Description
<input type="checkbox"/>	MSG_ALERT-111610358240405	1/11/2021 1:44 AM	[MDM] Your INSERT request for Valid_Language sent for a
<input checked="" type="checkbox"/>	MSG_ALERT-121610358240771	1/11/2021 1:44 AM	[MDM] INSERT request for Valid_Language is pending for y

PURGE   →   1 - 2 of 2   < >

## Web Services

Teradata RDM has an open, Service Oriented Architecture (SOA) that enables secure integration into a complex internal and external ecosystem. As such, any RDM service can be enabled for external access via a Web Service. Calling out to external Web Services is easily configured within the Teradata RDM workflow, using a Web Services node configuration. In addition, any RDM Master Table can be automatically enabled for Web Services access at the attribute level for any Create, Read, Update or Delete (CRUD) operation. All Web Services code is automatically generated by the product.

Teradata RDM provides an extensive set of APIs that can be useful to programmatically provide changes to data, modify data relationships, provide automated user administration, and perform other functions within Teradata RDM. All Teradata RDM functions that are exposed as Web Services have API documentation that serve as contracts for these services. This API documentation can be generated in PDF and also included as part of WSDL generation that is shared with the consuming applications/systems. This provides extensive

documentation about the input, output specifications for each of these Web Services and is applicable for both standard Web Services and configured Web Services.

Teradata RDM also supports REST based web services. REST based web service supports CRUD operations on tables, executing APIs and workflows.

## CHAPTER 3 **hRDM Table Maintenance**

---

### What's In This Chapter

This chapter provides detailed information on CRUD operations on web UI.

Topics include:

- [CRUD Data Maintenance from Web UI](#)

## **CRUD Data Maintenance from Web UI**

Teradata RDM allows the use of out of the box (OOTB) Configurable Web UIs to perform all kind of data manipulation actions (Create, Edit, Delete, Mass Update and Excel Upload) on database tables. The CRUD action buttons on the configurable UIs are displayed as per the access given to the users on the Table Editor Activity Search UI as in [Figure 6](#). The logged user will be able to perform CRUD activity on records in the table only if the user have been assigned permission for corresponding activities.

For the walk-through of configurable UI, Account table is taken as example. Configurable Web UIs can be auto created or can be user-defined. Web UI supports Filters, Sorts, Saved Favorites and other ease of user utilities. Web UI supports Excel Templates for Upload (enforcing business rules and Approvals).



Figure 6: Table Activity Search

Activity Details?

Role Name  
Admin

Role Id  
ADMIN

WORKFLOW ACTIVITY SEARCH

TABLE EDITOR ACTIVITY SEARCH

^ Search Results: Page 1 of 10

<input type="checkbox"/>	Activity Name	Create	Edit	Delete	Meta Data	Mass Update	Excel Up
<input checked="" type="checkbox"/>	*All Tables(MDM_INPUT_STAGING)	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Edit	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Meta Data	<input checked="" type="checkbox"/> Mass Update	<input type="checkbox"/>
<input checked="" type="checkbox"/>	*All Tables(MDM_MASTER)	<input checked="" type="checkbox"/> Create	<input checked="" type="checkbox"/> Edit	<input checked="" type="checkbox"/> Delete	<input checked="" type="checkbox"/> Meta Data	<input checked="" type="checkbox"/> Mass Update	<input checked="" type="checkbox"/>
<input type="checkbox"/>	*All Tables(MDM_NET_CHANGE)	<input type="checkbox"/> Create	<input type="checkbox"/> Edit	<input type="checkbox"/> Delete	<input type="checkbox"/> Meta Data	<input type="checkbox"/> Mass Update	<input type="checkbox"/>
<input type="checkbox"/>	*All Tables(MDM_OUTPUT_STAGING)	<input type="checkbox"/> Create	<input type="checkbox"/> Edit	<input type="checkbox"/> Delete	<input type="checkbox"/> Meta Data	<input type="checkbox"/> Mass Update	<input type="checkbox"/>

→

1 - 20 of 188

|<

<

>

>|

CANCEL

SAVE

SAVE AND RETURN

### Configurable UI Sample Screen

The [Figure 7](#) displays the sample Configurable UI generated through the RDM UI for the Account Table.

Figure 7: Configurable UI—Sample Account Table

Account				
<div> <input type="text"/> <span>UPLOAD EXCEL</span> <span>↕</span> <span>☰</span> <span>☰</span> <span>CREATE</span> <span>⋮</span> <span>♥</span> <span>?</span> </div>				
<input type="checkbox"/>	Accountid ⚙	AccountName ⚙	Amount ⚙	Department ⚙
<input type="checkbox"/>	6	Back Office Account	90000	Back Office ⋮
<input type="checkbox"/>	7	Branch Account	90000	Branch ⋮
<input type="checkbox"/>	9	Departmental Account	90000	Departmental ⋮
<div> <span>→</span> <span>1 - 3 of 3</span> <span>⏪</span> <span>&lt;</span> <span>&gt;</span> <span>⏩</span> </div>				

## Configurable UI—CRUD Operations

Teradata Configurable UI Utility provides a number of CRUD (Create, Read/Search, Update/Edit, Delete) table maintenance actions. CRUD maintenance actions also enforce rules and approvals. The following section provides the CRUD action examples.

### Create Record

To create a record in a table, perform the following steps:

- 1 On the configurable UI (Figure 7), click Create.
- 2 On the **Create** UI (Figure 8), enter the required details and click **Save** to persist the data to the underlying table.



Figure 10: Create Record

Create Account

Accountid \*

30

Key

AccountName \*

Sample Account

Amount

10000

Department

Back Office

Storeid

2

TemporalValidTi...

TemporalValidTi...

BACK

CREATE

- Click **Create**.
- The record is now added to the Account table as in [Figure 11](#).

Figure 11: Configurable UI—Record Created

Account

UPLOAD EXCEL

CREATE

Record created successfully.

	Accountid	AccountName	Amount	Department	Storeid
<input type="checkbox"/>	6	Back Office Account	90000	Back Office	2
<input type="checkbox"/>	7	Branch Account	90000	Branch	1
<input type="checkbox"/>	9	Departmental Account	90000	Departmental	1
<input type="checkbox"/>	30	Sample Account	100000	Back Office	2

1 - 4 of 4

### Copy Record

The Configurable UI allows copying of existing records. To copy a record, perform the following steps:

- 1 On the configurable UI (Figure 7), click on the Menu icon corresponding to the record that is to be copied and click **Copy**.
- 2 On the **Copy** UI, enter new primary key value and modify any details if required and click **Create** as in Figure 12.

Figure 12: Copy Record

**Copy Account**

Accountid \* 301

AccountName \* Sample Account Copy

Amount 100000

Department Back Office

StoreId 2

TemporalValidT... 3/24/2021

TemporalValidT... 3/25/2021

BACK CREATE

The copied record is added to the Account table as in Figure 13.

Figure 13: Configurable UI—Record Copied

**Account**

Record created successfully.

Accountid	AccountName	Amount	Department	StoreId
6	Back Office Account	90000	Back Office	2
7	Branch Account	90000	Branch	1
9	Departmental Account	90000	Departmental	1
30	Sample Account	100000	Back Office	2
301	Sample Account Copy	100000	Back Office	2

1 - 5 of 5

## Edit Record

The Configurable UI allows editing of table records. Configurable UI provides two types of edit— Simple Edit and Mass update.

Perform the following steps to perform simple edit:

- 1 On the configurable UI, select any one record that is to be edited and click **Edit**.
- 2 On the Edit Record UI ([Figure 14](#)), modify the necessary details and click **Save**.

Figure 14: Edit Record

The screenshot shows a web form titled "Edit Account". It contains the following fields and values:

- AccountId:** 301
- AccountName:** Sample Account Copy
- Amount:** 100000
- Department:** Back Office
- StoreId:** 2
- Temporal/ValidTime:** 3/24/2021 - 3/25/2021

At the bottom right, there are two buttons: **BACK** and **SAVE**.

The record is now edited.

## Mass Update

Configurable UI also supports multiple edit records simultaneously. Perform the following steps to perform multiple edit records:

- 1 On the configurable UI, select multiple records to be edited and click **Mass Update**.
- 2 On the Confirmation Webpage Dialog, click **Yes**.

Figure 15: Confirmation Webpage Dialog

The screenshot shows a confirmation dialog box with the title "Message -- Webpage Dialog". The message inside says: "Are you sure you want to update the selected record?". At the bottom right, there are two buttons: **YES** and **NO**.

- 3 On the **Mass Update** UI (Figure 16), select the columns you want to update and enter the updated values and click **Save**.

Figure 16: Mass Update

The screenshot shows a web application window titled "Mass Update Account". Inside the window, there is a list of four fields, each with a checkbox to its left. The first field, "AccountName", has its checkbox checked. The second field, "Amount", has its checkbox unchecked. The third field, "Department", has its checkbox checked and a dropdown menu below it showing "Select...". The fourth field, "StoreId", has its checkbox unchecked. At the bottom right of the window, there are two buttons: "BACK" and "SAVE".

The records are now updated and the message “Record(s) updated successfully” is displayed.

### Delete Record

The Configurable UI allows deletion of records. Note that through this delete maintenance, it will only set the Status to Inactive. It is not a hard delete from the database. Perform the following steps to delete a record:

- 1 On the configurable UI, select the record(s) to be deleted and click **Delete**.
- 2 On the Confirmation Webpage Dialog, click **Yes**.

The record is now deleted (Status = inactive) and the message: “Record(s) deleted successfully” is displayed.

## Excel Upload & Validations

The Configurable UI allows Upload of Data through Excel. This reads the Excel data file and imports the specified data. This feature also enforces Business Rules and Approvals specified for the Table.

### Simple Excel Upload without Rules and Approvals

To perform basic upload for Account Table through Excel, perform the following steps:

- 1 Access Account table configurable UI from left navigation links ( **Tables** -> **Master Tables**).
- 2 On the Account table configurable UI, click **Upload Excel** as in [Figure 17](#).

Figure 17: Configurable UI

AccountId	AccountName	Amount	Department	StoreId
6	Back Office Account	90000	Back Office	2
7	Branch Account	90000	Branch	1
9	Departmental Account	90000	Departmental	1
30	Sample Account	100000	Back Office	2

- 3 On the **Upload File** UI, click **Choose File** and select the excel file that is to be uploaded and Click **Upload** as in [Figure 18](#).

Figure 18: Upload File

File \*

Choose File AccountTemplate.xlsx

BACK SCHEDULE UPLOAD UPLOAD

**Schedule Upload:** Schedule Upload button on the Upload Documents page provide an option to schedule the excel upload for the selected service, table and the specified file. The **Scheduler Details** page ([Figure 19](#)) is displayed.



Figure 19: Scheduler Details

### Scheduler Details

Scheduler Parameters

Name \*

DataUpload\_RT

Server Date and Time

24/03/2021 07:35:20

Start At (Server Time): \*

3/24/2021

00

00

☐ Recurring

☐ Dependent

Workflow Parameters

File Path \*

Service Name \*

Master

Table Name \*

Account

BACK

SAVE

On the **Scheduler Details** page (Figure 19), by default, the **DataUpload\_RT** workflow is selected and in **Workflow Parameters** pane, the service name and the table name will be

RDM Server Guide

18

selected. Specify the other details and click Save. The upload process will be triggered on the scheduled date and time.





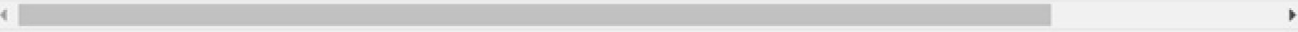

Figure 20 displays a sample Excel file for the above scenario.

Figure 20: Upload File—Sample Excel File

AccountId	AccountName	Amount	Department	StoreId
8	Customer Services Account	90000	Customer Services	2
6	Bank Office Account	90000	Back Office	2
4	Acquisitions Account	90000	Acquisitions	2

Once the **Upload** button is clicked, the **View Reports** page displays the upload report and its status as in Figure 21.

Figure 21: View Reports

View Reports									
REFRESH  									
<div>  Search Reports         </div>									
<div>  Reports: Page 1 of 1         </div>									
<input type="checkbox"/>	Report ID #	Template Name #	Upload Type #	Report Time #	Progress #	Total Records	Errors / Rejected	Pending Approv	
<input type="checkbox"/>	11	EntityAccountTemplate	Master	3/24/2021 7:38 AM	Completed	2	0	0	
<div>  </div>									
<div>  1 - 1 of 1  &lt; &lt; &gt; &gt;          </div>									

For this scenario, since no rules or approvals were created, all the eight records that were uploaded have been immediately created in the database after processing. This is now viewable in the Account Table UI as in Figure 22.

Figure 22: Configurable UI—Record(s) Uploaded

Account					
<input type="checkbox"/>	Accountid ↕	AccountName ↕	Amount ↕	Department ↕	Storeid ↕
<input type="checkbox"/>	6	Back Office Account	90000	Back Office	2
<input type="checkbox"/>	7	Branch Account	90000	Branch	1
<input type="checkbox"/>	8	Customer Services Account	90000	Branch	2
<input type="checkbox"/>	9	Departmental Account	90000	Departmental	1
<input type="checkbox"/>	12	Help Desk Account	90000	Branch	2
<input type="checkbox"/>	30	Sample Account	100000	Back Office	2
					1 - 6 of 6

The following section describes the Excel Upload process with business rules and approvals.

### Business Rules Management from Web UI

Business Rules allow the users to specify restrictions on table actions based on certain criteria.

RDM supports the following types of business rules:

- **Data Acquisition Rules:** the data acquisition rules are created on input staging tables to filter out records that satisfy the defined business rules or conditions. Examples of acquisition rules include filtering raw transaction data like banking data entry details that are not transformed and cleansed.
- **Data Cleansing Rules:** the data cleansing rules identify and correct invalid or inaccurate data records and provide a clean and consistent data set. Examples of cleansing rules include normalize spaces, remove extra trailing or leading spaces in the data columns, correct invalid characters and null values, provide default values etc.
- **Data Standardization Rules:** the data standardization rules establish a common data format across the organization. The data standardization rules identify and update data with certain standard formats. Examples of standardization rules include replace known values to preferred values using lookup code sets, example: Street to ST, change uppercase to lowercase, standardize city name based on the postal code etc.



- Cleansing and Standardization (C&S) rules will be executed as part of Do DB Persist request.
  - The records updated or enriched by C&S rules can be reviewed by approvers.
  - The C&S rules will be executed as part of Configurable UI & Excel Upload process.
- 
- **Data Enrichment Rules:** the data enrichment rules allow data enrichment on the input data that is change data using calculations, Teradata functions etc., which can be of Insert and Update types. Example may include address correction, enrichment rules can be used to correct address by standardizing street, city, and state fields, and adding the last four digits of the zip code etc.
  - **Data Transformation Rules:** the data transformation rules can be categorized as a set of rules defined to covert the data format of a source data system into the data format of a destination. Examples include value conversions or translation functions, normalizing numeric values to conform to minimum and maximum values.
  - **Data Validation Rules:** the data validation rules define validation criteria or checks on any master table as per business scenario by which erroneous data can be identified. Examples include data size checks like data value must consist of 10 characters only, format checks to see that data conform to specific format like first 9 characters must be digits from 0 through 9 and 10th must be digit or X etc.
  - **Default Validation Rules:** RDM checks for database constraints as part of validation rules while persisting data into Master table through DO\_DB\_PERSIST framework. Common validation rules are provided OOTB and get executed during execution of DO\_DB\_PERSIST requests.  
The different default validation rule includes (Uniqueness violation, Referential integrity, Not Null and Column check constraints).

### **Add Business Rules**

For this scenario, will create two business rules for the Account Table.

First Rule: Account - Amount Rule

This rule sends out an error for all account rows table action with Amount greater than 100000.

Table 1: Sample Business Rule Creation Details

Basic Information	
Name	Account-Amount_Rule
Repository and Table Name	Master-Account
Rule Type	Data Validation

Table 1: Sample Business Rule Creation Details

Status	ACTIVE
<b>Error Data</b>	
Error Code	Amount should be less than 100000
Error Severity	SEVERE_ERROR
<b>From Context</b>	
Selected tables	Account (from BCM_MASTER)
<b>Join Context</b>	
Tables - Linked Document	Account - Store
Selected Links	N/A
<b>Search Context</b>	
Regular Expression	Account with Amount > 100000
<b>Action Type</b>	
Selected	All

To create a new rule, perform the following steps:

1. Navigate to **Manage Business Rules** UI from the left navigation pane (**Data Harmonization** - > **Business Rules** and click **Manage Business Rules**).

Figure 23: Manage Business Rule

Manage Business Rules

GENERATE RULES CREATE

Business Rules Search

Repository: Master Rule Type: Select... ☐ Auto Generated Rule

Object Type: Tables Table: Select... Rule Name:

RULE SEQUENCE CLEAR SEARCH

Search Results: No Records found

Rule Name #	Rule Type #
-------------	-------------

1 - 0 of 0

- 2 On the **Manage Business Rules** UI (Figure 23), click **Create**.
- 3 On the **Create Business Rule** UI (Figure 24), enter the required details about the business rule and click **Next**.

Figure 24: Create Business Rule

**Create Business Rule** ?

\* denotes required field

Rule Name\*  
Account\_Amount Rule

Repository\*  
Master

Description  
Account\_Amount Rule

Object\*  
Tables

Start Date

Tables\*  
Account

End Date

Rule Type\*  
Data Validation

Status  
ACTIVE

CANCEL NEXT

- **Rule Name:** unique name of the business rule.
- **Description:** short meaningful description about the rule. This description will be displayed to the business user when a rule violation occurs.
- **Start and End Date:** validity of the rule.
- **Repositories:** repository on which the business rule is being created. Repository can be (Input Staging, Master or Non MDM).
- **Objects:** objects dropdown lists the valid values as Tables, Code Sets and Hierarchy Objects. On selection of the object in Object dropdown, the corresponding selected object dropdown is displayed. If the object is selected as Tables, the tables dropdown is displayed.
- **Tables:** name of the table in the selected repository on which the business rule is being created.
- **Rule Type:** type of business rule being created.
- **Status:** Active or Inactive

4 On the **Rule Editor Wizard** UI, set the required details about the rule.

- **Error Data**

From the **Select Step** section, click **Error Data**. The Error Data section defines the Error Code, which refers to the error message that will be displayed when this rule is violated and Error Severity, which refers to the error criticality.

For this example, define the Error Code as “Amount should be less than 100000” and the Error Severity as Severe\_Error as in [Figure 25](#).

Figure 25: Rule Editor Wizard—Error Data

The screenshot shows the 'Rule Editor Wizard' window. The 'Select Step' section on the left has 'Error Data' selected. The 'Error Data' section on the right contains the following fields:

- Error Code:** A text input field containing 'Amount should be less than 100000'.
- Error Severity:** A dropdown menu with 'Severe\_Error' selected.

Below these fields is a 'Dynamic Expression Viewer' section. It displays the following text:

```
Data Validation BCM_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE_ERROR]
From BCM_MASTER.Account
Where
Action Type :Insert,Update,Activate,Deactivate,Delete,All
```

At the bottom right of the window are three buttons: 'CANCEL', 'BACK', and 'VALIDATE'.

- **From Context**

From the **Select Step** section, click **From Context**. The From Context section defines the tables involved in the rule. For this example, retain this as Account as in [Figure 26](#).



Figure 26: Rule Editor Wizard—From Context

The screenshot shows the 'Rule Editor Wizard' window, specifically the 'From Context' step. On the left, a sidebar lists steps: 'Error Data', 'From Context' (highlighted), 'Join Context', 'Search Context', and 'Action Type'. The main area is divided into 'Repositories' and 'Available Tables'. The 'Repositories' dropdown is set to 'BCM\_MASTER'. Below it, 'Available Tables' shows 'Select... Account' and 'Selected Tables' shows 'BCM\_MASTER.Account'. A 'Dynamic Expression Viewer' section contains the following text: 'Data Validation BCM\_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE\_ERROR]', 'From BCM\_MASTER.Account', 'Where', and 'Action Type :Insert,Update,Activate,Deactivate,Delete,All'. At the bottom right are 'CANCEL', 'BACK', and 'VALIDATE' buttons.

- **Join Context** (optional)  
From the **Select Step** section, click **Join Context**. The Join Context section defines if there are any joins necessary for this rule. For this example, retain this as blank as in [Figure 27](#).

Figure 27: Rule Editor Wizard—Join Context

**Rule Editor Wizard**

**Select Step**      **Join Context**

Error Data

From Context

Join Context

Search Context

Action Type

Tables: BCM\_MASTER.Account

Linked Document

Available Links

Selected Links

Dynamic Expression Viewer

Data Validation BCM\_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE\_ERROR]  
From BCM\_MASTER.Account  
Where  
Action Type :Insert,Update,Activate,Deactivate,Delete,All

CANCEL BACK VALIDATE

- **Search Context**

From the **Select Step** section, click **Search Context** to define Rule. This screen defines most of the actual business rule.

- In the **Search Context** section, right-click **Context** and select **Regular Expression** as in Figure 28.
- In the regular expression editor, define the rule as Amount > 100000.

Figure 28: Rule Editor Wizard—Search Context

**Rule Editor Wizard**

Select Step      Search Context

Error Data

From Context

Join Context

Search Context

Action Type

Dynamic Expression Viewer

Data Validation BCM\_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE\_ERROR]  
From BCM\_MASTER.Account  
Where  
Action Type :Insert,Update,Activate,Deactivate,Delete,All

CANCEL    BACK    VALIDATE

The changes made here are also reflected in the **Dynamic Expression Viewer** section as in [Figure 29](#).

Figure 29: Rule Editor Wizard—Search Context Dynamic Expression

The screenshot displays the 'Rule Editor Wizard' window, specifically the 'Search Context' step. The interface is divided into several sections:

- Select Step:** A sidebar on the left lists the steps: Error Data, From Context, Join Context, Search Context (highlighted), and Action Type.
- Search Context:** A tree view shows the context structure: Context > Regular Expression (highlighted).
- Context Details:** A panel on the right showing details for the selected context:
  - Property:** Amount
  - Condition:** >
  - Value Type:** Radio buttons for Value (selected), Field, and Code Set.
  - Function:** Select...
  - Value:** 100000
- Dynamic Expression Viewer:** A section at the bottom showing the generated dynamic expression:

```
Data Validation BCM_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE_ERROR]
From BCM_MASTER.Account
Where
[BCM_MASTER.Account]Amount>100000
Action Type :Insert,Update,Activate,Deactivate,Delete,All
```
- Buttons:** At the bottom right, there are buttons for CANCEL, BACK, and a green VALIDATE button.

- Action Type

From the **Select Step** section, click **Action Type**. The **Action Type** section defines what action is involved with this rule.  
 For this example, retain as All as in [Figure 30](#).

Figure 30: Rule Editor Wizard—Action Type

The screenshot shows the 'Rule Editor Wizard' window. It has a title bar with a question mark icon. The main content area is divided into two sections: 'Select Step' and 'Action Type'. The 'Select Step' section lists several options: 'Error Data', 'From Context', 'Join Context', 'Search Context', and 'Action Type'. The 'Action Type' section is currently selected and shows a list of actions: 'All', 'Insert', 'Update', 'Activate', 'Deactivate', and 'Delete'. Each action has a checkbox next to it, and all are checked. Below this, there is a 'Dynamic Expression Viewer' section with a text area containing the following text: 'Data Validation BCM\_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE\_ERROR] From BCM\_MASTER.Account Where [BCM\_MASTER.Account]Amount>100000 Action Type :Insert,Update,Activate,Deactivate,Delete,All'. At the bottom right, there are three buttons: 'CANCEL', 'BACK', and 'VALIDATE'.

Select Step	Action Type
Error Data	<input checked="" type="checkbox"/> All
From Context	<input checked="" type="checkbox"/> Insert
Join Context	<input checked="" type="checkbox"/> Update
Search Context	<input checked="" type="checkbox"/> Activate
<u>Action Type</u>	<input checked="" type="checkbox"/> Deactivate
	<input checked="" type="checkbox"/> Delete

Dynamic Expression Viewer

```
Data Validation BCM_MASTER.Account [Error Code : Amount should be less than 100000 , Error Severity : SEVERE_ERROR]
From BCM_MASTER.Account
Where
[BCM_MASTER.Account]Amount>100000
Action Type :Insert,Update,Activate,Deactivate,Delete,All
```

CANCEL BACK VALIDATE

5 Click **Validate** once done to validate the rule for any errors as in [Figure 30](#).

Figure 31: Summary Rule Expression

Summary Rule Expression

Rule Name

Account\_Amount Rule

Table Name

Account

Rule Type

Data Validation

```

update T1 from MST_ACCOUNT T1
SET SYS_ERR_CODE = (COALESCE(T1.SYS_ERR_CODE,"") || "Amount should be less than 100000;") ,
SYS_ERR_SVRTY = 'SEVERE_ERROR'
where T1.AMOUNT > 100000
        
```

EXECUTE

CHECK SYNTAX

BACK

SAVE

6 On the **Summary Rule Expression** UI (Figure 31), click **Save** to save the business rule.

Message: “Business Rule created successfully” is displayed.

Do the same steps for a second rule - Account Code Set rule.

This rule sends out an error for all account rows table action with Store that does not exist in the Store table.

Figure 32: Rule Editor Wizard—Custom Expression

The screenshot shows the 'Rule Editor Wizard' window. On the left, under 'Select Step', the steps are: Error Data, From Context, Join Context, Search Context (which is highlighted), and Action Type. In the center, under 'Search Context', a tree view shows 'Context' with a sub-item 'Custom Expression' highlighted in orange. On the right, under 'Custom Expression', a text area contains the SQL: `T1.department not in (SELECT VALUE_ID from LK_ AND T2.STATUS='ACTIVE')`. Below this is a 'Dynamic Expression Viewer' section containing a preview of the rule configuration:   
Data Validation BCM\_MASTER.Account [Error Code : Should be one from lookup value(Department) , Error Severity : SEVERE\_ERROR]  
From BCM\_MASTER.Account  
Where  
T1.department not in (SELECT VALUE\_ID from LK\_CODE\_MST\_VAL T2 where T2.TYPE\_ID='Department' AND T2.STATUS='ACTIVE')  
Action Type :Insert,Update,Activate,Deactivate,Delete,All  
At the bottom right are three buttons: CANCEL, BACK, and a green VALIDATE button.

- The only notable difference is that the second rule uses a different Search Context using Custom Expression as in [Figure 32](#). Custom Expressions allow the use of any valid SQL based rule. Instead of building and configuring the rule from the Expression Builder, the SQL is either typed or pasted into the UI.
- The two rules should both be active once done.

### Modify Business Rules

Apart from Business rules creation, the other actions available on the Manage Business Rules UI includes the following and as displayed in [Figure 33](#).

- Activate—activates the selected business rule.
- Deactivate—deactivates the selected business rule.
- Delete—removes selected business rule.
- Edit— allow you to change the details of the rule. The Edit UI is same as the Create Rule UI, but allows you to modify a previously created rule.
- Copy—copies the current selected business rule to a new rule.

Figure 33: Manage Business Rules

**Manage Business Rules** GENERATE RULES CREATE ♥ ?

**Business Rules Search**

Repository: **Master** Rule Type: **Select...** ☐ Auto Generated Rule

Object Type: **Tables** Table: **Select...** Rule Name:

RULE SEQUENCE CLEAR SEARCH

Search Results: Page 1 of 1

Rule Name	Rule Type	Table	Status	Description
Account_Amount Rule	Data Validation	Account	ACTIVE	Account_Amount Rule
Account Code Set Rule	Data Validation	Account	ACTIVE	Account Code Set Rule

1 - 2 of 2 < > << >>

### Approval Management from Web UI

Approvals are the most simple and common type of workflow in RDM. Therefore, creating and modifying Approvals must be easy for a Business User to configure from the Web UI.



Approvals allow the users to set user-based or user-role based reviews on table actions performed on specific tables. This is performed in the “Manage Approvals” page. In this UI, you will be able to create, delete, and modify current approvals in place.

### Create Approval

For this scenario, create approval process for Account Tables following the below steps:

- 1 Navigate to **Manage Approvals** UI from **Data Harmonization -> Approvals -> Manage Approvals**
- 2 On **Manage Approvals** UI, click **Create**.
- 3 On the **Approval Details** UI, enter the necessary details for the approval as in [Figure 34](#).

Figure 34: Approval Details

Approval Details

BACK SAVE ?

Approval Name \*

Approval Description \*

Approval Object Type \*  
Tables

Tables \*  
Select...

Approval Type \*  
Serial

Action Type \*  
ALL

Approval User Type \*  
Select

☒ Advanced Settings

Notification Type  
Select...

Reprocess Time (in days)  
15

☒ Interested Parties

Emails

Available User Group

RDM\_ADMIN  
RDM\_USER  
SU\_UGP  
TAS\_USER

Assigned User Group

- q **Approval Name:** unique name of the approval process.
- q **Approval Description:** brief description about the purpose of approval process being created.
- q **Approval Type:** (serial or parallel)
  - Serial Approval Type**
    - w In serial approval type, the approval process happens in serial manner. The record or document is sent to the first approver, if the approval process with the first approver is completed and approved then the approval request is sent to the next available approver if any.
    - w The underlying record will be persisted to MASTER table if all of the approvers approve the record.
    - w The underlying record will not be persisted to MASTER table if any of the approvers reject the record and will be persisted to the Error table with appropriate message and comments.



The serial approval will happen in the same order of approvers as in the assigned approvers list.

---

#### **Parallel Approval Type**

- w In parallel approval type, the approval process happens in parallel manner. The record or document is sent to all approvers simultaneously.
- w The underlying record will be persisted to Master table if all the approvers approve the request.
- w The underlying record will not be persisted to Master table if any of the approvers reject the record and will be persisted to the Error table with appropriate message and comments.
- q **Approval Object Type:** the object type on which the approval is created. Object type can be Tables, Business Entity, Business Rules, Views, Model, Code Sets, Hierarchy or Hierarchy Objects. On selection of the object in Approval Object Type drop-down, the corresponding selected object drop-down is displayed. If the object is selected as Tables, the tables drop-down is displayed.
  - w **Table:** approval can be created on any MDM master tables.
  - w **Business Entity:** approval can be created on any business entity comprising a set of MDM tables. Approval will be enabled on all the tables within the business entity.
    - **Business Rules:** approval can be created on business rules with all action types. For example, for action type activate and deactivate, when a business rule with action type as activate is submitted for approval and the business rule approval request is approved, then standard validations like Business rule validation, SQL validation will be executed and the status of the rule will be Active. When a business rule with action type as activate is submitted for approval and if the business rule approval request is rejected, then the status will be changed to

Inactive.

- w **View:** manage approval honors all DocViews created from MDM Studio. The Save API for the views needs to be configured for approval to work on views. For more details on Save API for views, refer to *Appendix: Multiple Table Updates Using DocView in Master Data Management Reference Guide.pdf*.
  - **Code Sets:** approval can be created on any code sets.
  - **Model:** approval can be defined on models created through Model Builder UI.
  - **Hierarchy Objects:** approval can be created on any hierarchy objects.
  - **Hierarchy:** approval can be created on any hierarchy enabled for promotion process.
- q **Approval User Type:** User Based or User Group Based option.
  - w **User Based:** if User Based option is selected, the **Available Approvers** list displays all the available users.
 

In user based approval, the record is passed from one user to another user as per the order of users in the Available Approvers list.
  - w **User Group Based:** if User Group Based option is selected, the **Available Approvers** list displays all the available users groups.
 

In user group based approval, the record is passed from one user group to another user group. In User Group based approval, for any user group selected as part of Approval, either all users of the group or any one user of the group will be able to participate in Approval.



If the user group name and user name are same, preference will be given to user name approval type.

---

- q **Action Type:** the action type can be (insert, update, delete or all).
 

For approval on models, the action type by default is Update. For action type insert, update or delete, corresponding action will be taken on the tables using DO\_DB\_PERSIST execution. Direct SQL actions will not be honored. For details on DO\_DB\_PERSIST execution, refer to [Section : “DataPersist Specification”](#) in [Chapter 2: “Admin Tasks.”](#)



- w An approval for a document with different approval type and same action type cannot be created. The below message is displayed.
 

Message: Approval already exists on <approval object type> for < selected action type>.

For example: Code Set with Approval type as serial and action type as insert. Code Set with Approval type as parallel and action type as insert cannot be created. Message is displayed as “Approval already exists on Code Set for INSERT Action type”.

- 
- q **Advanced Settings:** select this checkbox to update the notification type, Reprocess Time and Interested Parties settings as below.
  - q **Notification Type:** the notification type can be (Email, Web Alerts or both).
  - q **Interested Parties:** enable the Interested Parties checkbox to send notification email to all interested parties when the record is approved by all the users or when a record is rejected by any user. Interested parties will be in CC list of the email sent. Interested parties can be specified by entering the email Ids and/or user groups.
    - **Emails:** enter the email addresses of all the interested parties separated by commas.
    - **User Group:** select the interested user group(s) from **Available User Group** list and move it to **Assigned User Group** list. All the users of the selected user groups in the Assigned user group(s) list will receive the notification email.
  - q **Reprocess Time (in days):** enter the number of days until which the record can be reprocessed. If an approval rejected record is not reprocessed within the specified reprocess time, it will be considered as invalid and cannot be reprocessed further.
  - q **Available Approvers** list displays all the available approvers (based on user or user group type). The **Assigned Approvers** list displays the list of assigned approvers.
- 4 Click **Save**.

### **Modify Approval**

Apart from creating approvals, the Manage Approvals Page also enables the modification of existing approvals and also includes the following features:

- On the Manage Approvals UI, select the approval to be modified and click any of the following action buttons:
  - **Edit**—allows the user to modify the selected approval. The edit page is similar to the create approvals page.
  - **Delete**—deletes the selected approval.
  - **Activate**—activates any selected inactive approvals.
  - **Deactivate**—deactivates any selected approvals.
  - **User Groups**—add and manage the selected approval for user group.
- **Multi-user approvals**  
Multi-user approvals can be either serial or parallel.
  - a For this example, modify Account Approval to have rdm\_user as its third approval as in [Figure 35](#).
  - b Click **Update**.

Figure 35: Approval Details

Approval Details

BACK RULES UPDATE ?

Approval Name

AttributeSetApproval

Approval Description \*

AttributeSetApproval

Approval Object Type \*

Tables

Tables \*

AttributeSet

Approval Type \*

Serial

Action Type \*

ALL

Approval User Type \*

User Based

Available Approvers

admin  
ciuser  
rdm\_user  
tasuser

>  
<

Assigned Approvers

rdm\_admin

UP  
DOWN

☒ Advanced Settings

Notification Type

ALL

Reprocess Time (in days)

15

☒ Interested Parties

Emails

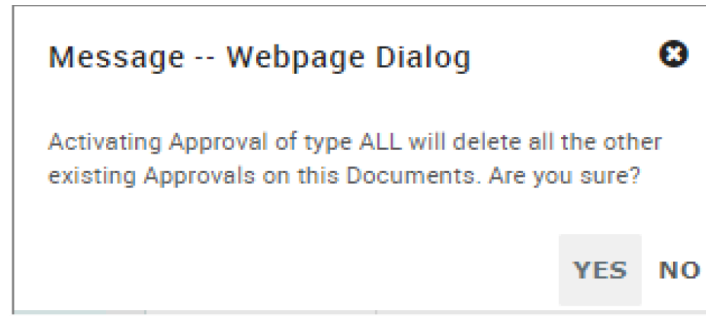
Available User Group

ADMIN\_UGP  
CI\_UGP  
RDM\_ADMIN  
RDM\_USER

>  
<

Assigned User Group

Figure 36: Confirmation Webpage Dialog



- c On the Webpage Dialog, click **Yes**.  
The approval is now updated.

### ***Advanced Approval Options (Rule Based Approval)***

Apart from the Table Action based Approval process, Teradata RDM also allows more options to implement complex approval scenarios. Rule Based Approval allows approval for specific rows of a table. Specific rows of a table can be sent for approval based on rules created.

The Rule Based approval involves the following approval types:

- **Auto Approval**—The Auto Approval also known as Skip Approval allows rows of a table to skip the approval process based on certain rule criteria. Applicable for all table actions. Only one rule can be set as Auto Approval rule.
- **Assignment Approval**—The Assignment Approval rule allows you to assign specific approvers to approve specific rows of a table based on certain rule criteria.
- **Column Level Approval**—The Column Level Approval allows you to specify important columns for approval. The Column Level approval will be triggered on the incoming rows only if the specified column value changes.

Any or all the above approval type can be defined on the specific rows of a table. The rule based approval is defined on tables on which table based approval is already defined.

For description of advanced approval options (rule based approval), the following example is used:

Table: Channel table

- Channel Id
- Channel Name
- Channel Type

#### **1 Column Level Approval**

- For Updates, only rows that have changes in Channel Type will be considered.

#### **2 Assignment Approval**

- If the row's Channel Type = Marketing then Approver = Marketing User
- If the row's Channel Type = Billing then Approver = Billing User

- If the row's Channel Type = Accounting then Approver = Accounting User
- 3 Auto Approval
- If Channel Type is not equal to any of the three scenarios above, Record will ignore approval process

### Create Rule Based Approvals

Perform the following steps to create rule based approvals:

- 1 Navigate to **Manage Approvals** UI and click on the Menu icon corresponding to the required approval and click Edit as in [Figure 37](#).

Figure 37: Manage Approvals

The screenshot shows the 'Manage Approvals' web interface. At the top, there's a header with the title 'Manage Approvals', a green 'CREATE' button, a heart icon, and a help icon. Below the header is a table with the following columns: 'Approval Object', 'Approval Object Type', 'Approval Type', 'Is Active', and 'Action Type'. The table contains two rows: 'Account' and 'Channel'. Each row has a 'Select All' button with a search icon. A context menu is open over the 'Channel' row, displaying options: 'Edit', 'Delete', 'Activate', 'Deactivate', and 'User Groups'. The table also includes a search bar at the top left, a sort icon at the top right, and pagination controls at the bottom right showing '1 - 2 of 2'.

Approval Object	Approval Object Type	Approval Type	Is Active	Action Type
Account	Table	Serial	True	ALL
Channel	Table	Serial	True	ALL

Figure 38: Manage Approvals

**Approval Details** BACK RULES UPDATE ?

Approval Name  
AttributeSetApproval

Approval Description \*  
AttributeSetApproval

Approval Object Type \*  
Tables

Tables \*  
AttributeSet

Approval Type \*  
Serial

Action Type \*  
ALL

Approval User Type \*  
User Based

Available Approvers

admin  
ciuser  
rdm\_user  
tasuser

Assigned Approvers

\*  
rdm\_admin

UP  
DOWN

☒ Advanced Settings

Notification Type  
ALL

Reprocess Time (in days)  
15

☒ Interested Parties

Emails

Available User Group

ADMIN\_UGP  
CI\_UGP  
RDM\_ADMIN  
RDM\_USER

Assigned User Group

The **Manage Approval Rules** UI with **Column Level Approval** UI section is displayed if the selected table level approval is of Action type UPDATE or ALL else if the selected table level approval is of Insert Action type, the **Manage Approval Rules** UI without the **Column Level Approval** section is displayed.

- 2 On the **Column Level Approval** UI, you can select the important columns on which the rules will be defined from the **Available Columns** list and move it to **Selected Columns** list



and click **Next** to navigate to **Create Approval Rules** UI or click **Save and Return** as in [Figure 39](#).

Figure 39: Column Level Approval

**Manage Approval Rules** BACK DEFINE RULES SAVE AND RETURN ?

^ Approval Details

Approval Table  
**AttributeSet**

Approval Type  
**Serial**

Action Type  
**ALL**

Approval UserType  
**User Based**

^ Column Level Approval

Available Columns

Selected Columns

SYS\_AUTH\_ID  
SOURCE  
CREATED\_BY  
CREATION\_DATE  
ENTITY\_STATE  
LAST\_MODIFIED\_BY  
LAST\_MODIFIED\_DATE  
SYS\_TYPE  
SYS\_ERR\_CODE  
SYS\_ERR\_SEVERITY  
SYS\_APPROVAL\_ID

>>  
>  
<  
<<

Rules Defined

Search Results: No Rules found

Rule Id	Expression	Approval Type	Rule created from	Priority	Approvers	User Type	Date Valid From	Date Valid Till
---------	------------	---------------	-------------------	----------	-----------	-----------	-----------------	-----------------

3 On the **Manage Approval Rules** UI ([Figure 40](#)), click **Define Rules**

Figure 40: Create Approval Rules

**Create Approval Rules**

\* denotes required field

Approval Table  
**LastSyncUp**

Property Name  
ApplicationName

Operation  
=

Select:  
☒ Value 
 ☐ Reference 
 ☐ Code Set

☐ Compare with Old Value

Property Value  
MDM

Rule Valid

Actions ( ) AND OR

CLEAR RULE UPDATE RULE SAVE RULE

Approval Rules: No Rules found

Rule Id	Expression	Approval Type	Date Valid From

BACK CANCEL NEXT

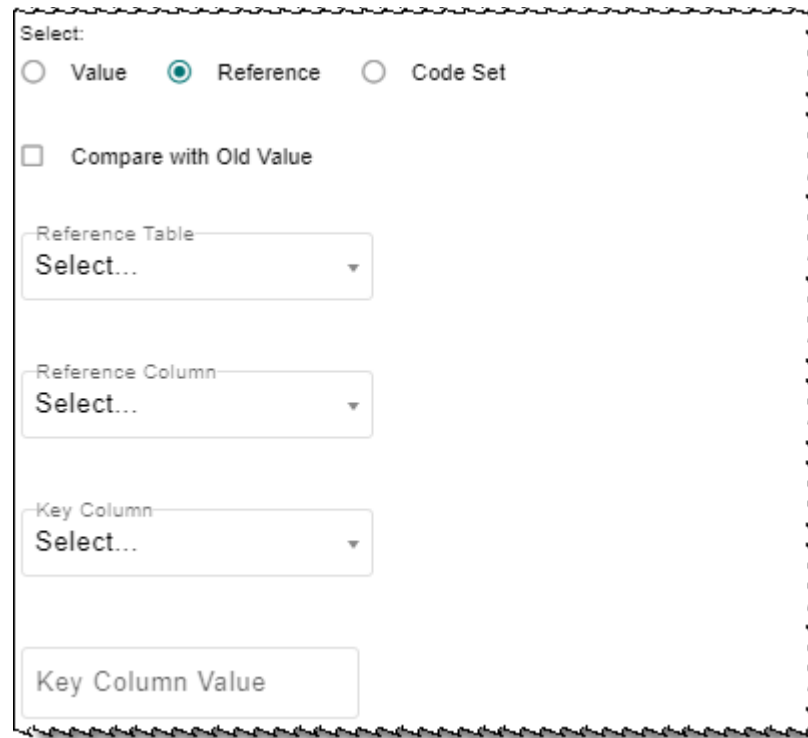
1 - 0 of -1

The **Create Approval Rules** UI is displayed. Create a rule by entering the following

- **Property Name:** lists all the columns of the selected table. Select the column on which the rule is being created.
- **Operation:** select the operation parameter (=, <, >, <=, >=, <>, is null, and is not null)
- **Value** or **Reference** or **Code Set** radio button.

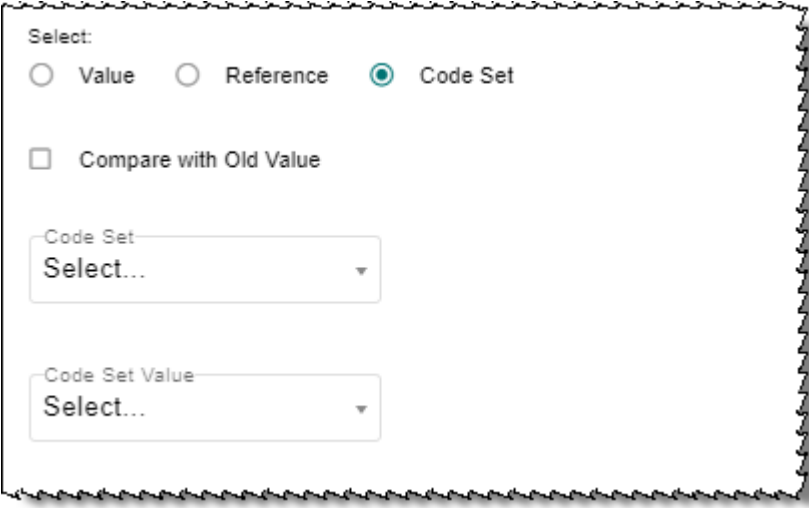
If **Value** option is selected, enter the value in the Property Value field.

If **Reference** option is selected, additional fields are displayed as below. Select the reference table, reference column, key column and enter the key column value.



- Default operator supported for Key column value is '='
- The datatype of the column in the approval table and the datatype in reference column should be same.
- The key column and its values should be such that only single row is fetched. Preferably the key column should be the PK of the table.
- For Reference option, if the key column is selected as date type with timestamp, the date value needs to be entered in database format (YYYY-MM-DD HH:MI:SS) and not in the locale format.

If **Code Set** option is selected, additional fields are displayed as below. Select the code set and code set value.



Select:

☐ Value ☐ Reference ☒ Code Set

☐ Compare with Old Value

Code Set  
Select...

Code Set Value  
Select...

- **Compare with Old Value:** select this checkbox and click on Add Condition, a rule will be generated in the form “Table1.Col1 [Operator] #<OldValue>”. This rule compares the column values with the existing values and if there is any change in the column values, approval will be triggered.
- **Rule Valid From:** specify the From and To date. The rule being created will be valid only for the specified duration of days.
- **Actions:** you can optionally use the AND, OR, ( and ) buttons to build complex rule expressions.
- **Add Condition:** click Add Condition button to add the rule. The rule expression is displayed in the **Expression Editor** field.
- **Clear Rule:** click Clear Rule to delete the rule expression.
- **Save Rule:** click Save Rule to save the rule being created.

Figure 41: Create Approval Rules

### Create Approval Rules

denotes required field

✓ Rule was created successfully.

Approval Table  
**LastSyncUp**

Property Name  
ApplicationName

Operation  
=

Select:  
☒ Value
 ☐ Reference
 ☐ Code Set

☐ Compare with Old Value

Property Value

Rule Valid

Actions ( ) AND OR

CLEAR RULE UPDATE RULE SAVE RULE

Approval Rules

Rule Id	Expression	Approval Type	Date Valid From	
1	LastSyncUp ApplicationName = 'MDM&a...	None	1/20/2021	Delete Edit Set Auto Approval

BACK CANCEL NEXT

1 - 1 of -1

- **Update Rule:** to update a rule, on the Approval Rules section, click on the Menu icon corresponding to the rule and click **Edit**. The rule expression is displayed in the Expression Editor and the **Update Rule** button will be enabled. You can modify the expression and click **Update Rule**.
- 4 To set the Auto Approval rule, select the Rule and Click **Set Auto Approval** and click **Next**.

Once the auto approval is enabled on the rule, the activation message is displayed and the rule type will be changed to Auto Approval as displayed in the **Approval Type** column in [Figure 42](#).

Figure 42: Auto Approval Rules

Create Approval Rules

?

denotes required field

✓

Auto approval activation on the selected rule was successful.

Approval Table

LastSyncUp

Property Name

ApplicationName

Operation

=

Select:

☒ Value

☐ Reference

☐ Code Set

☐ Compare with Old Value

Property Value

Expression Editor

CLEAR RULE

UPDATE RULE

SAVE RULE

Rule Valid

Actions

( ) AND OR

Approval Rules

Rule Id	Expression	Approval Type	Date Valid From
=	x	x	x
1	LastSyncUp.ApplicationName = 'MDM&a...	Auto Approval	1/20/2021

BACK

CANCEL

NEXT

→

1 - 1 of -1

|<

<

>

>|

- 5 Click **Next** to define assignment.
- The **Assignment Approval Rules** UI is displayed.

Figure 43: Assignment Approval Rules

Rule Id	Expression	Rule created from	User Type
2	LastSyncUp.ApplicationName = 'MDM1&...	UI	
1	LastSyncUp.ApplicationName = 'MDM&a...	UI	

Assign Approvers

BACK SET PRIORITY GENERATE ADDITIONAL RULES DONE 1 - 2 of 2

- 6 On the **Assignment Approval Rules** (Figure 42), click on the Menu icon corresponding to the required rule and click **Assign Approvers** to assign approves for the selected rule.  
The **Assign Approvers** pane is displayed. The **Available Approvers** list displays all the available approvers. On **Assign Approvers** pane, select the User Type. Based on user type (User based, User Group based), the Available Approvers list will be populated.
- 7 On the **Assign Approvers** pane, select the approvers from **Available Approvers** list and move it to **Selected Approvers** list as in Figure 42 and click **Save Users**.
- 8 Similarly assign approvers for all the rules.  
The Assigned Users column displays all the users assigned for approval.

Figure 44: Assignment Approval Rules

### Assignment Approval Rules

Approval Rules: Page 1 of 1

Rule Id	Expression	User Type	Assigned
= <input type="text"/> X <input type="text"/>	X <input type="text"/>	X <input type="text"/>	X <input type="text"/>
3	LastSyncUp.ApplicationName = 'MDM2&...		
2	LastSyncUp.ApplicationName = 'MDM1&...		

BACK SET PRIORITY GENERATE ADDITIONAL RULES DONE 1 - 2 of 2

### Assign Approvers

User Type  
User Based

Available Approvers

Assigned Approvers

rdm\_user

tasuser

>>

>

<

<<

rdm\_admin

admin

SAVE USERS

Message: Assignment approval saved successfully is displayed and the **Assigned Users** column displays all the users assigned for approval.



Figure 45: Assignment Approval Rules

Assignment Approval Rules ?

Assignment approval was saved successfully.

Approval Rules: Page 1 of 1

Rule Id	Expression	Rule created from	User Type	Assigned Users
<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>	<div><div></div><div></div><div></div></div>
3	LastSyncUp.InstanceID = 'MDM4'...	UI	UserBased	rdm_admin,ar
2	LastSyncUp.ApplicationName = 'MDM2&...	UI	UserBased	rdm_user
1	LastSyncUp.ApplicationName = 'MDM1&...	UI	UserBased	rdm_admin,rd

BACK

SET PRIORITY

GENERATE ADDITIONAL RULES

DONE

1 - 3 of 3

### Set Priority

Perform the following to set priority for execution of rules.

- 1 On the **Assignment Approvals Rules** UI (Figure 46), click on the Menu icon corresponding to the required rule and click **Set Priority** to set priority for execution of the assignment rules.

Figure 46: Assignment Approval Rules

Assignment Approval Rules <span>?</span>									
Approval Rules: Page 1 of 1									
Rule Id #	Expression #		User Type #		Assigned Users #				
<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>	<input type="text" value=""/>
3	LastSyncUp.ApplicationName = 'MDM2&...'		UserBased		rdm_admin,admin				
2	LastSyncUp.ApplicationName = 'MDM1&...'								

The **Rule Priority** page (Figure 47) is displayed.

Figure 47: Assignment Approval Rules—Rule Priority

Assignment Approval Rules <span>?</span>			
Approval Rules: Page 1 of 1			
Rule Id ☺	Expression ☺	User Type ☺	Assigned Users ☺
<input type="text" value="="/> <input type="text" value="v"/> <input type="text" value="x"/> <input type="text" value="Q"/>	<input type="text" value="x"/> <input type="text" value="Q"/>	<input type="text" value="x"/> <input type="text" value="Q"/>	<input type="text" value="x"/>
3	LastSyncUp.ApplicationName = 'MDM2&...	UserBased	rdm_admin,admin <input type="text" value="⋮"/>
2	LastSyncUp.ApplicationName = 'MDM1&...		<input type="text" value="⋮"/>
<div> <div>BACK</div> <div>SET PRIORITY</div> <div>DONE</div> <div> <input type="text" value="→"/> 1 - 2 of 2 <div> <input type="text" value=" &lt;"/> <input type="text" value="&lt;"/> <input type="text" value="&gt;"/> <input type="text" value="&gt; "/> </div> </div> </div>			

- On the **Rule Priority** page (Figure 47), click on the Menu icon corresponding to the required rule and click **Up** or **Down** button to sequence the order of rule execution and then click **Done**.

### Triggering Advanced Approvals

The below section describes some examples of the Channel table approval process as per the use cases above.

#### 1 Sample Process—Column Level Approval

- Navigate to Channel table UI from left navigation pane ( **Tables -> Master Tables**) and select a record and click **Edit**.
- On the **Edit Channel** UI, modify only the ChannelName field and click **Save** as in Figure 48.

Figure 48: Edit Record

Edit Channel

Channel\_Id

2

Key

Channel\_Name\*

Marketing\_Channel

Channel\_Type\*

Marketing

BACK

SAVE

For Updates, Channel Type must be changed to enable approval process.

Note that the Approval is by-passed and the record is updated as the column level approval is enabled only on ChannelType field.

Figure 49: Configurable UI

Channel				
<div> <div> <div>🔍</div> <div>UPLOAD EXCEL</div> <div>↑↓</div> <div>☰</div> <div>☰</div> <div>CREATE</div> <div>⋮</div> <div>❤️</div> <div>?</div> </div> </div>				
<input type="checkbox"/>	Channel_Id ↕	Channel_Name ↕	Channel_Type ↕	Target Id ↕
<input type="checkbox"/>	2	Marketing	Marketing_Channel	⋮
<input type="checkbox"/>	11	Channel13	Basic	⋮

- Navigate back to **Edit Channel** UI and modify the ChannelType field and click **Save** as in [Figure 50](#).

Figure 50: Edit Record

Edit Channel

Channel\_Id

2

Channel\_Name \*

Marketing

Channel\_Type \*

Marketing\_Channel\_Type

BACK

SAVE

Note that the record is now submitted for Approval as in [Figure 51](#).

Figure 51: Configurable UI

Channel				
<div> <div> <div>🔍</div> <div>UPLOAD EXCEL</div> <div>↑↓</div> <div>☰</div> <div>☰</div> <div>CREATE</div> <div>⋮</div> <div>❤️</div> <div>?</div> </div> </div>				
<div> <div>🟢</div> <div>Record(s) submitted for approval.</div> </div>				
<input type="checkbox"/>	Channel_Id ↕	Channel_Name ↕	Channel_Type ↕	Target Id ↕
<input type="checkbox"/>	2	Marketing	Marketing	⋮

## 2 Sample Process - Assignment Approval

- Navigate to Create Record UI, example: Create Channel table UI from left navigation pane ( **Tables** -> **Master Tables**) and enter the record details as in [Figure 52](#).  
Insert with Channel Type = Accounting

Figure 52: Create Channel Table Record

The screenshot shows a web form titled "Create Channel". It has three input fields: "Channel\_Id" with the value "3", "Channel\_Name" with the value "Billing", and "Channel\_Type" with the value "Billing". Each field has a red asterisk indicating it is required. A key icon is visible next to the "Channel\_Id" field. In the top right corner, there is a heart icon. At the bottom right, there are two buttons: "BACK" and "CREATE".

The record is now submitted for approval to the Accounting user.

Similarly insert records with Channel Type = Marketing and Channel Type = Billing. Note that both the records are sent for approval (sent to Marketing user and Billing user respectively).

Now the approvers can login to Approval Inbox and approve the records.

### 3 Sample Process - Auto Approval

- Navigate to Create Record UI, example: Create Channel table UI from left navigation pane ( **Tables** -> **Master Tables**) and enter the record details as in [Figure 53](#).  
For Auto Approval Rule: Channel Type should not be Marketing, Billing or Accounting.

Figure 53: Create Record

The screenshot shows a web form titled "Create Channel". It has three input fields: "Channel\_Id" with the value "11", "Channel\_Name" with the value "Channel13", and "Channel\_Type" with the value "Basic". Each field has a red asterisk indicating it is required. A key icon is visible next to the "Channel\_Id" field. In the top right corner, there is a heart icon. At the bottom right, there are two buttons: "BACK" and "CREATE".

Now the record is created and the ChannelType is not Marketing, Billing or Accounting as in [Figure 54](#).

Figure 54: Configurable UI

Channel_Id	Channel_Name	Channel_Type	Target Id	Authorization
2	Marketing	Marketing		
11	Channel13	Basic		

### Excel Upload with Rules and Approvals

For excel upload with rules and approval scenario, consider the following rules and approvals defined above.

Summary of Approvals enforced against the Account Table:

- Account Approval: Multi-user approval on All actions
  - The Type of Approval is Serial and User- Based.
  - The Table actions involved in the approvals are “All” actions (Insert, Update, Delete).
  - The User is notified through Web and Email.
  - The Approvers are admin and lrdm\_admin user (in that order).

Summary of Rules enforced against Account Table:

- Rule 1: Referential Integrity Validation
  - The StoreId column from Account table is foreign key to StoreId column from Store Table.
  - If the entered StoreId value is not present in Store Table the RDM application returns error.
  - Error Message: Data not found in table (MST\_STORE)
- Rule 2: Amount Validation
  - If the entered amount is greater than 100,000, the RDM application returns error.
  - Error Message: AMOUNT SHOULD BE LESS THAN 100000
- Rule 3: Code Set Validation
  - The Department column is derived from code sets. If the input value is not matching the Department code set; returns error.
  - Error Message: SHOULD BE ONE FROM LOOKUP VALUE(DEPARTMENT).

To perform the upload through Excel, follow the below steps:

- Access Account Table Configurable UI from left navigation links (**Tables ->Master Tables**).
- On the Account table configurable UI, click **Upload Excel** as in [Figure 55](#).

Figure 55: Configurable UI—Account Table

Account

UPLOAD EXCEL

CREATE

<input type="checkbox"/>	Accountid	AccountName	Amount	Department	StoreId	
<input type="checkbox"/>	6	Back Office Account	90000	Back Office	2	
<input type="checkbox"/>	7	Branch Account	90000	Branch	1	
<input type="checkbox"/>	9	Departmental Account	90000	Departmental	1	
<input type="checkbox"/>	30	Sample Account	100000	Back Office	2	

→ 1 - 4 of 4 |< < > >|

- On the **Upload File** UI, click **Choose File** and select the excel file that is to be uploaded and Click **Upload** as in [Figure 56](#).

Figure 56: Upload File

Upload File

File \*

Choose File Account1\_Rules.xlsx

BACK

SCHEDULE UPLOAD

UPLOAD

[Figure 57](#) displays the sample Excel file for this scenario:  
 The highlighted yellow rows are the records expected to have violations.  
 Rule Violations: Amount greater than 100000, Store Ids not in Store List.  
 Code Set Violations: Department not existing in valid values of code set.

Figure 57: Upload File—Sample Excel File

AccountId	AccountName	Department	Amount	StoreId
8	Customer Services Account	Customer Services	90000	2
6	Bank Office Account	Back Office	90000	2
4	Acquisitions Account	Acquisitions	90000	2
12	Help Desk Account	Help Desk	90000	2
10	Desk Account	Desk	90000	2
7	Branch Account	Branch	90000	2
16	Nerve Center Account	Nerve Center	330000	22
9	Departmental Account	Departmental	90000	2
2	Salary Account	Salary Department	990000	11
5	Arm Account	ArmsLegs	2200000	32
11	Food Chain Account	Food Department	90000	2
13	HR Account	HR Department	500000	22
14	Interdepartmental Account	Interdepartmental Account	500000	22
15	Mailroom Account	Mailroom	500000	22
17	Personnel Account	Personnel	500000	22
18	R & D Account	R & D Department	500000	22
19	Sales Account	Sales Department	500000	22
20	Secretariat Account	Secretariat	500000	22
21	Security Account	Security Department	500000	22

The View Reports UI displays the upload excel processes and its status. Given the Sample data, 16 error records and 3 correct records (pending approval) are displayed as in [Figure 58](#).

Figure 58: View Reports

View Reports						
<div>REFRESH  </div>						
<div>Search Reports</div>						
<div>Reports: Page 1 of 1</div>						
<input type="checkbox"/>	Report ID ↕	Template Name ↕	Upload Type ↕	Report Time ▼	Progress ↕	Total Records      Errors / Rejected
<input type="checkbox"/>	101	EntityAccountTemplate	Master	4/5/2021 7:03 AM	Completed with gaps : sent for approval	19      16

- On the **View Reports** UI, click on the Report Id to drill down to view the details of error records as in [Figure 59](#).



Figure 59: Upload Status

Upload Status						
Error Details						
error_Account: Page 1 of 1						
<input type="checkbox"/>	AccountId	AccountName	Amount	Department	StoreId	Lo
	x	x	x	x Select All	x	x
<input type="checkbox"/>	12	Help Desk Account	90000	Help Desk	2	
<input type="checkbox"/>	10	Desk Account	90000	Desk	2	
<input type="checkbox"/>	8	Customer Services Account	90000	Customer Services	2	
<input type="checkbox"/>	17	Personnel Account	500000	Personnel	22	
<input type="checkbox"/>	4	Acquisitions Account	90000	Acquisitions	2	
<input type="checkbox"/>	14	Interdepartmental Account	500000	Interdepartmental Account	22	
<input type="checkbox"/>	20	Secretariat Account	500000	Secretariat	22	
<input type="checkbox"/>	15	Mailroom Account	500000	Mailroom	22	
<input type="checkbox"/>	18	R & D Account	500000	R & D Department	22	
<input type="checkbox"/>	5	Arm Account	2200000	ArmsLegs	32	

- 5 On the **Upload Status** UI (Figure 59), select any record and click **Edit** to reprocess the record.
- 6 On the **Record Edit Form**, modify the record details and click **Save and Reload**.  
In this example, change amount to 9000 from 330000 and StoreId to 2 from 22 as in Figure 60.

Figure 60: Record Edit Form

Record Edit Form

Error Message  
Should be one from lookup  
value(Department);Amount  
should be less than  
100000;Should be one from  
lookup value(Department);

AccountId \*

20

key

[number]

AccountName \*

Secretariat Account

[text]

Amount

500000

[number]

Department

Select...

StoreId

22

[number]

Locale

Select...

TemporalValidTime

04/05/2021 - 12/31/9999

[PeriodDate]

BACK

SAVE

SAVE AND RELOAD

Message: “Record reloaded successfully” is displayed as in [Figure 61](#) and the error record count will come down to 15.

Figure 61: Upload Status

Upload Details

BACK?

Upload Status

✓

Record reloaded successfully, sent for approval.

101

Start Time  
4/5/2021 7:03 AM

Process Time  
4/5/2021 7:04 AM

Progress  
Completed with gaps : sent for approval

Error Details

error\_Account: Page 1 of 1

1

<input type="checkbox"/>	AccountId	AccountName	Amount	Department	StoreId	Lc
	x	x	x	x Select All	x	x
<input type="checkbox"/>	12	Help Desk Account	90000	Help Desk	2	
<input type="checkbox"/>	10	Desk Account	90000	Desk	2	
<input type="checkbox"/>	8	Customer Services Account	90000	Customer Services	2	
<input type="checkbox"/>	17	Personnel Account	500000	Personnel	22	
<input type="checkbox"/>	4	Acquisitions Account	90000	Acquisitions	2	

→

1 - 15 of 15

|< < > >|

- 7 On the **Upload Details** UI, click **Back**.

The changes are now reflected in the View Reports page. 15 Errors and 5 pending Approval as in [Figure 62](#).

Figure 62: View Reports

Report ID	Template Name	Upload Type	Report Time	Progress	Total Records	Errors / Rejected	Pending A
101	EntityAccountTemplate	Master	4/5/2021 7:03 AM	Completed with gaps : sent for approval	19	15	4

- Now navigate to Approval Inbox UI and approve the 8 records. First, login as admin user and access the Approval Inbox on the upper right corner as in [Figure 63](#).

Figure 63: Approval Inbox

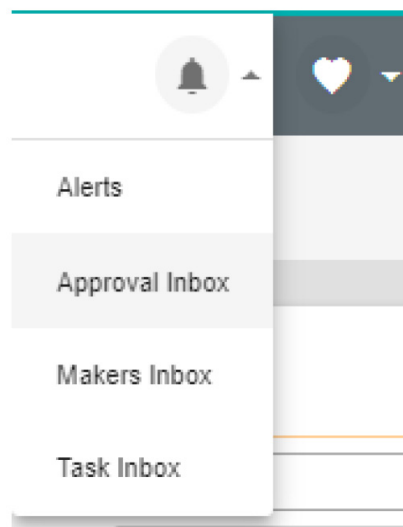
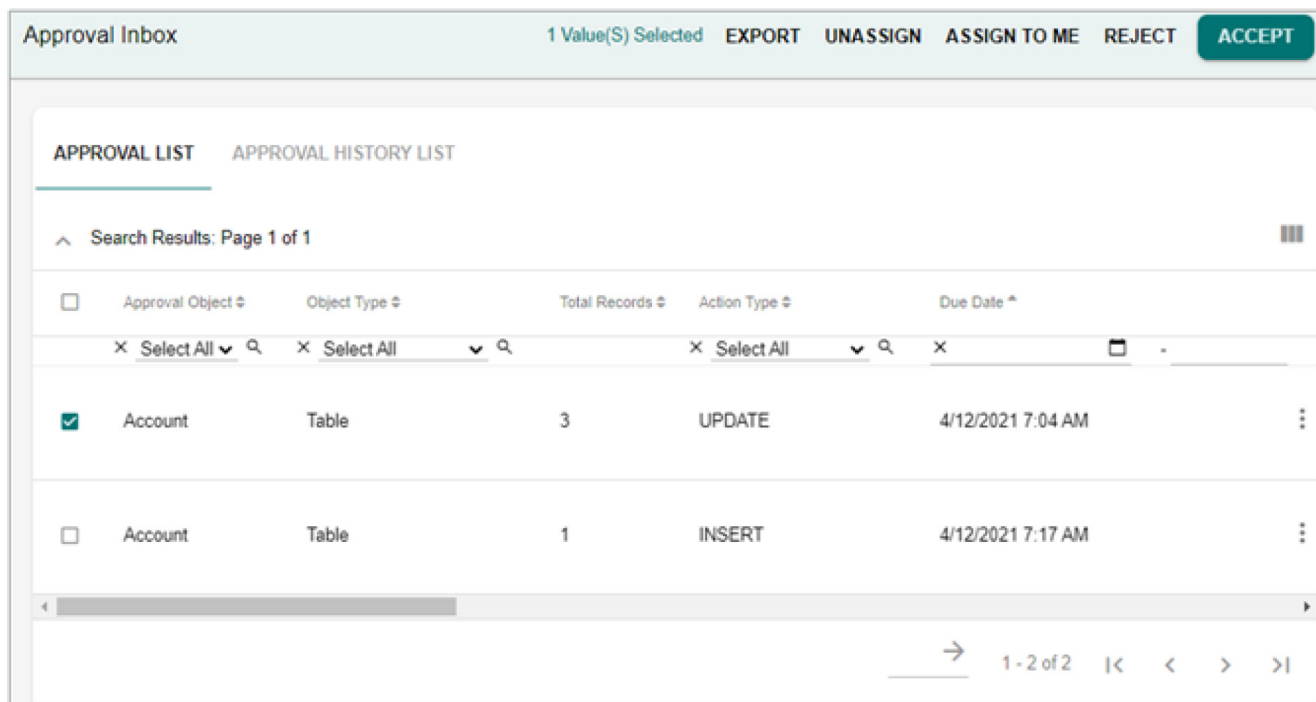


Figure 64: Approval Inbox



- 9 In the **Approval Inbox** UI, select the pending approval and click **Accept** to accept the approval.

The Reject option is also available, but not used for this scenario.

Optionally you can review the records before approving the records using the View Records option. For this scenario, review the approval with 4 records. The View Record(s) UI displays the record details as in [Figure 65](#).

Figure 65: View Records



- 10 On the **View Records** UI, click **Action** and then click **Accept**.

Message: “Record(s) approved successfully” is displayed.

### Approval Notifications

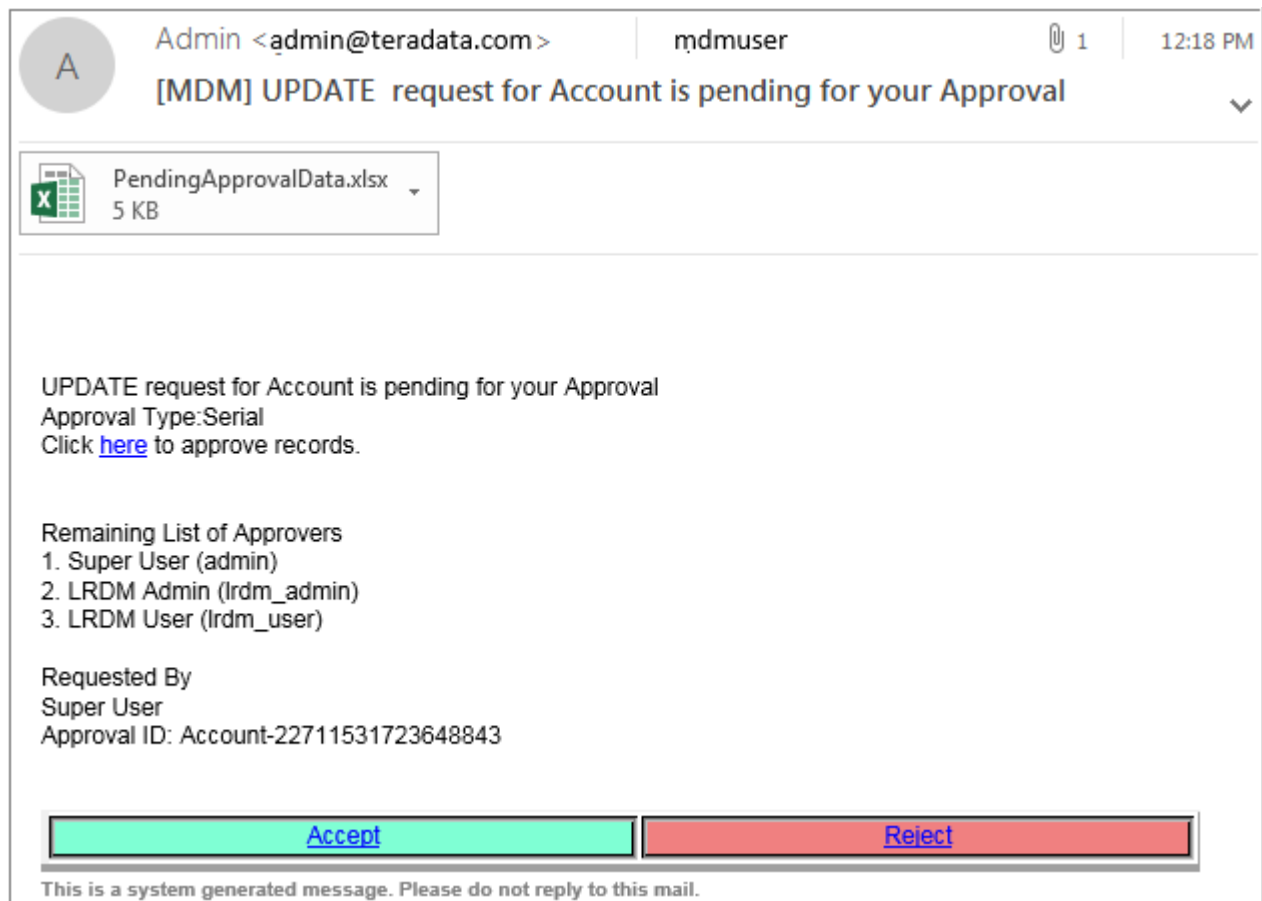
Approval notification will be sent to the users either by Email, Web Alerts or both.

Email notifications will be sent to approvers during approval process (both when an approval request is generated and when an approval request is approved or rejected).

The email notification will have the following details as in [Figure 66](#).

- Contain a link to RDM login page (Click Here). On click of that, you will be directed to the login page. When you login, you will be redirected to Approval Inbox of the logged in user.
- The email also provides an option to approver to Accept or Reject a record. Approver can approve/reject records by clicking on the Accept/Reject links without logging on to the application.
- In case of bulk records, the email notification will have all the above details along with an excel sheet containing all the record details.
- In case of user group approval, users of user group will get initial email for the approval request intimation where they will get link to Assign the approval request to themselves. The user who assigns the approval request to self will get a second email to accept/reject the approval request.

Figure 66: Email Notification



- 11 Now login as second approver (rdm\_admin) and approve the records.  
Approve both the 3 records (initial load) and 1 record (re-edited record).

After the last approval has gone through, the records will be reflected on the Account Master Table as in [Figure 67](#).

Figure 67: Configurable UI—Account Table

Account				
		UPLOAD EXCEL		CREATE
<input type="checkbox"/>	AccountId	AccountName	Amount	Department
<input type="checkbox"/>	6	Back Office Account	90000	Back Off
<input type="checkbox"/>	7	Branch Account	90000	Branch
<input type="checkbox"/>	8	Customer Services Account	90000	Branch
<input type="checkbox"/>	9	Departmental Account	90000	Departm
<input type="checkbox"/>	12	Help Desk Account	90000	Branch
<input type="checkbox"/>	20	Secretariat Account	90000	Branch
<input type="checkbox"/>	30	Sample Account	100000	Back Off
<div> <div>1 - 7 of 7</div> <div> <div>&lt;</div> <div>&gt;</div> </div> </div>				

### Export Approval Requests

The list of approval requests on Approval Inbox UI can be exported to Excel for data analysis using the Export button. The exported excel sheet will have the following details as in [Figure 68](#):

Figure 68: Approval Inbox Data

Approval Details										
ID	ApprovalID	LoadID	Requester	ObjectType	TotalRecords	ActionType	Status	CreatedDate	DueDate	PendingSince
Account-2271153	Account	102	USR_1	Table	7	UPDATE	Active	07/16/201	07/23/201	26 Minute Se
Approval Record Details										
Attribute	Amount	AccountName	Department	AccountID	StoreID	Target ID	Authorization	Source	Created By	Creation Date
Current Value	90000	Branch Ac	Branch	7	2				USR_1	
New Value	90000	Branch Ac	Branch	7	2				USR_1	
Current Value	90000	Customer	Customer	8	2				USR_1	
New Value	90000	Customer	Customer	8	2				USR_1	
Current Value	90000	Acquisiti	Acquisiti	4	2				USR_1	
New Value	90000	Acquisiti	Acquisiti	4	2				USR_1	
Current Value	90000	Desk Acc	Desk	10	2				USR_1	
New Value	90000	Desk Acc	Desk	10	2				USR_1	
Current Value	90000	Departme	Departme	9	2				USR_1	
New Value	90000	Departme	Departme	9	2				USR_1	

- Approval Details: displays the Approval related details like: Approval Id, Approval Object, Object Type, Total Records, Action Type, Status, Created Date, Due Date, Pending Status and Approval Type.
- Approval Record Details: displays the record details, the current value and the new value of each field. For updated records, the changed data will be highlighted with different colors.

### Makers Inbox

Any RDM users can review the approval requests created by them on the Makers Inbox UI. Navigate to Makers Inbox from Tools icon on RDM UI.

On the **Makers Inbox** UI (Figure 68), you can perform the following tasks:

- Enter comments in the **Comments** text field and click **Withdraw** to withdraw any record sent for approval.
- Select any record and click **View Record/s** to view record details.
- You can also edit and reprocess the rejected records on view records UI if the user has Enrichment Access activity assigned and Create or Edit access on the table.
- With the **Approval History** tab selected, you can select the rejected record or withdrawn record and click on **Reprocess** to navigate to **Error Details** UI from where you can edit the record(s) and click **Reload** to reprocess the record.



Figure 69: Makers Inbox

Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date	ID
AttributeSet	Table	6	UPDATE	Feb 21, 2022,...	1 month(s), 0 ...	USR_1	Serial	2	Feb 14, 2022,...	AttributeSet-2...

rdm\_admin  
Approver name

Waiting for Approval  
Inbox status

View records  
Withdraw

Rows per page 10 1 of 1

## Approval Group Requests

Many times for data integrity issues and based on business requirements, there is a need to group and approve related approval requests on different tables (related or independent) together. MDM provides an option on the Approval Monitor UI to select a set of individual approval requests and group it under a single parent approval request. Once the parent request is created, the Approval Inbox and Approval Monitor UIs display only the parent request and using the View button, the approver can view the child request details. The approvers can approve or reject only the parent request and all the linked child approval requests will be auto-approved or rejected accordingly.

Grouping of individual approval request must satisfy the following conditions else grouping cannot be done and valid error message will be displayed on the UI.

- The selected requests for grouping must be of the same approval type (all the selected requests must be either serial or parallel approval type).
- The selected requests for grouping must be of the same user type (all the selected requests must be either User based or User group based).
- If any of the individual approval request under the grouped request is partially approved, grouping cannot be done.
- Grouping is supported only on Table type approval object.
- Unique approvers from all requests grouped will be picked as approver of grouped request.

Grouping requests involves the following:

- [Create Group Request](#)
- [Group Request on Approval Inbox](#)
- [Group Request on Approval Monitor](#)
- [Group Requests on Makers Inbox](#)
- [Group Requests on Approval History Tab](#)

### Create Group Request

Perform the following steps to create group requests:

- 1 On the **Approval Monitor** UI, select the approval requests to be grouped and click **Group Requests** as in [Figure 70](#).

Figure 70: Approval Monitor

Approval monitor

1 value(s) selected

EXPORT

GROUP REQUESTS

<input checked="" type="checkbox"/>	Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date	ID
<input checked="" type="checkbox"/>	AttributeSet	Table	6	UPDATE	Feb 21, 2...	1 month(...	USR_1	Serial	2	Feb 14, 2...	Attribute... <div></div>

Rows per page

10

1 of 1

- 2 On the **Grouping Details** UI, enter the following details as in [Figure 71](#).
  - **Group Name:** unique name of the group being created.
  - **Group Description:** brief description about the group being created.
  - **Approvers:** the list of approvers for the grouped request in case of serial approval. Drag and drop the approvers to change the approval type serial.

Figure 71: Grouping Details

The screenshot shows a 'Group requests' dialog box with a close button (X). It contains three input fields: 'Group name \*', 'Group description', and 'Approvers'. The 'Approvers' field shows a list of users, with 'RDM\_ADMIN' selected. At the bottom are 'CANCEL' and 'SAVE' buttons.

Group name \*

Group description

Approvers

RDM\_ADMIN

CANCEL

SAVE

- 3 Click **Save**.

Message: “Requests are grouped successfully” is displayed. An email will be sent to the approvers about the grouped request, involved approval requests and the tables involved. The email will not contain the records details and there will be no option to accept/reject from email.

The grouped requests will be displayed on the Approval Inbox, Approval Monitor, Approval History and on the Maker’s Inbox UIs.



- Once grouping is done, individual approval requests for each approval grouped will be removed from approvers Inbox/Monitor UI and instead group request will be populated.
- Once requests are grouped, you cannot remove any request from the group.

### Group Request on Approval Inbox

On the **Approval Inbox** UI, the grouped request is displayed as single request and will have the Action Type as “GROUPED” and Object Type as “ApprovalGroup” as in [Figure 72](#).

Figure 72: Approval Inbox

Approval inbox										
1 value(s) selected EXPORT UNASSIGN ASSIGN TO ME REJECT ACCEPT										
<input checked="" type="checkbox"/>	Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date ↓ ID
<input checked="" type="checkbox"/>	Group Test 1	ApprovalGroup	2	GROUPED	Mar 23, 2022...	7Hour(s), 56 ...	USR_1	Serial		Mar 16, 2022... Group Test 1...
Rows per page 10 1 of 1 < >										

The approver can use the **View records** option to view the details of the grouped requests. The **View Record(s)** UI displays all the individual approval requests and the approver can view the individual approval request details by selecting the request and clicking on **View** as in [Figure 73](#).

On the **Approval Inbox** UI, approver can take action on a single group request at a time. The approver can select the group request and perform all the activities available on the Approval Inbox UI.



If any business rule validation error occurs on any one of the record under the grouped request, then on Accept by the last approver none of the records would get persisted but the error record will appear in the Error table. The process would rollback such that the Approval Request appears back on the UI with the status “Waiting for Approval” for the last approver. An email would be sent to the current approver and the person who grouped the requests notifying the Validation error with the details.

Figure 73: Approval Inbox—View Record(s)—Group Request

Grouped approval records					Q	BACK
ID	Approval object	Action	Total records	Creation date ↓		
AttributeSet-21131647430209053	AttributeSet	INSERT	1	Mar 16, 2022, 4:30:09 AM		
AttributeSet-21121647429808170	AttributeSet	UPDATE	1	Mar 16, 2022, 4:23:29 A	View records	
					Rows per page 10	2 of 2 < >

Figure 74: Records for Group Request

View records						Sort & search on New Value		Q	↓	BACK
	Attribute_Set_Id 🔑	Attribute_Set_Name	Attribute_Set_Description	Group_Class_Id	Uptd_Priority					
New Value	2345	Group recrd		1						
						Rows per page 10	1 of 1	<	>	

### **Group Request on Approval Monitor**

On the **Approval Monitor** UI, the details of group request are displayed in the same way as on the **Approval Inbox** UI. The grouped request is displayed as a single request and will have the Action Type as “GROUPED” and Object Type as “ApprovalGroup”.

Figure 75: Approval Monitor

Approval monitor

1 value(s) selectedEXPORTGROUP REQUESTS

<input checked="" type="checkbox"/>	Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date	ID
<input checked="" type="checkbox"/>	Group Te...	Approval...	2	GROUPED	Mar 23, 2...	8Hour(s),...	USR_1	Serial		Mar 16, 2...	Group Te...

Rows per page101 of

View records

Take action

In addition to the above details, the **Approval Monitor** UI also displays the list of approvers for each grouped request and their inbox status as in [Figure 75](#).

Figure 76: Approval Monitor

Approval monitor

1 value(s) selectedEXPORTGROUP REQUESTS

<input checked="" type="checkbox"/>	Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date	ID
<input checked="" type="checkbox"/>	Group Te...	Approval...	2	GROUPED	Mar 23, 2...	8Hour(s),...	USR_1	Serial		Mar 16, 2...	Group Te...

RDM\_ADMIN  
Approver name

Waiting for Approval  
Inbox status

Rows per page101 of 1

<>

On the **Approval Monitor** UI, click on the Menu icon corresponding to the group request and click **View** to view and edit individual requests of the selected group.

**Group Requests on Makers Inbox**

On the **Makers Inbox** UI as well the grouped request is displayed as a single request and will have the Action Type as “GROUPED” and Object Type as “ApprovalGroup”. The Makers Inbox UI also displays the list of approvers for each grouped request and their inbox status as in [Figure 77](#).

Figure 77: Makers Inbox—Group Request

Approval object	Object type	Total records	Action	Due date	Pending since	Requestor	Approval type	Load ID	Creation date	ID
Group Test 1	ApprovalGroup	2	GROUPED	Mar 23, 2022,...	8Hour(s), 11 ...	USR_1	Serial		Mar 16, 2022,...	Group Test 1-...
<div> <div>RDM_ADMIN Approver name</div> <div>Waiting for Approval Inbox status</div> <div> View records Withdraw </div> </div>										

On the **Makers Inbox** UI, the approver can view the records and on ‘Withdraw’ will withdraw (reject) the grouped request.

### **Group Requests on Approval History Tab**

The **Approval History** tab on the Approval Inbox UI, Approval Monitor UI and Makers Inbox UI displays the actions performed on the group request. On the History tab, for group request, you cannot perform Reprocess activity.

### **Approval Re-authentication (E-Signature) in MDM**

MDM provides an option to enable re-authentication security support on accept/reject approval requests in Approval Inbox UI. By default, the re-authentication feature is disabled on Approval Inbox UI. To enable re-authentication on approval request, navigate to xserver.xml and xserverweb.xml available at: `<MDM_Install_Directory>\cfg\properties` and set the “RE\_AUTH\_ENABLED” service parameter value to “true”. Restart the server for the changes to be effective.

Once the re-authentication is enabled, each time you try to Accept/Reject an approval request on Approval Inbox UI, the password entry option is displayed. On successful authentication with correct password entry, you can perform Accept/Reject actions.

On the password entry popup, without entering password, you can click on **Cancel** button and proceed with other operations on the UI like Export, Unassign. Assign to Me and View Records. But on the **View Records** UI, you cannot accept or reject record without password authentication.

On the password entry pop-up, if wrong password is entered, error message as “Invalid Password” is displayed.

By default, three password failed attempts are allowed. After three wrong entry of password, you will be logged out of the application. The user account will get locked. Login as admin and unlock the user. Once the user account is unlocked, you can again login to application with correct password.

The default value for password failed attempt is configurable. To configure, navigate to xserver.xml and xserverweb.xml available at: `<MDM_Install_Directory>\cfg\properties` and set the value for “RE\_AUTH\_PWD\_RETRY\_COUNT” service parameter.

## CHAPTER 4 Code Set Management

---

### What's In This Chapter

This chapter provides detailed description on code set management.

Topics include:

- [Instantiate RDM Tables](#)
- [Create New Code Set](#)
- [Add Code Values](#)
- [Create New Source Systems](#)
- [Create Source Mapping](#)
- [Load from Excel](#)
- [Auto Matching Logic](#)
- [Code Set Groups](#)
- [Source System Association](#)
- [Manage Version](#)
- [Manage Authorization](#)

## Instantiate RDM Tables

Teradata RDM manages Code Set data using the tables—Reference Code, Standard Map and Reference Description Tables. By default, only one generic set of these tables are available. To generate additional RDM Tables, perform the following steps:

- 1 Navigate to **Model Builder** UI from left navigation pane: **Model Setup -> Model Builder** and select **Template** option from **Actions** dropdown.
- 2 On the **Manage Template** UI, select the RDM template type and enter the number of tables to be generated in the “Enter number of tables to be generated” text field and click **Generate Model** to initiate the ISG or schema generation as in [Figure 78](#).

When RDM Template option is selected, option to create Source Map table is provided in addition to the system tables of RDM. To create Source map tables, select the SourceMap Template checkbox and enter number of tables to be generated.

Figure 78: Manage Template

Manage Template

3 Value(S) Selected VALIDATE TEMPLATE GENERATE MODEL

Template Type: RDM Template

☐ SourceMap Template

Enter number of tables to be generated.: 4

Search Templates: Page 1 of 1

<input checked="" type="checkbox"/>	Template Name #	Folder Name #	Generated Table Count	Modified By #	Modified Date #
<input checked="" type="checkbox"/>	Reference_Code	LRDM	0		
<input checked="" type="checkbox"/>	Reference_Desc	LRDM	0		
<input checked="" type="checkbox"/>	Standard_Map	LRDM	0		

1 - 3 of 3

- 3 On the Confirmation popup, click **Yes**.  
The ISG started message will be displayed on the UI.
- 4 On the **Manage Template** UI, click **Back**.
- 5 On the **Model Builder** UI, click **Logs**.  
Review the generated logs.
- 6 Click **Back** once the Deployment status is changed to successful as in [Figure 79](#).



Figure 79: Model Deployment Logs

Model Deployment Logs

BACK REFRESH ?

Search Model Builder Logs: Page 1 of 1

Execution Id #	Deployment Status	Source Name #	Generated Tables	Start Time #
x	x	x	x	x
13	SUCCESSFUL : 4 Minute(s) 21 Second(s)	RDM	Reference_0001_Code Reference_0001_Desc Reference_0002_Code Reference_0002_Desc Reference_0003_Code Reference_0003_Desc	03/26/2021 11:50:41

→ 1 - 12 of 12 < > >>

- 7 On the **Model Builder** UI, select the **Generated Models** option.  
The RDM tables will be listed in the **Search Models** pane as in [Figure 80](#).

Figure 80: Model Builder

Model Builder

ACTIONS

CREATE

Status  
Generated

Groups  
Select...

Search Models: Page 1 of 5

<input type="checkbox"/>	Object Name #	Object Type #	Description #	Logical Source Name #	Modified By #	Modified Date #
	x	x	x	x	x	x
<input type="checkbox"/>	Reference_0003_Desc	Table	Reference_Desc	Default_DB		
<input type="checkbox"/>	Standard_0001_Map	Table	Standard_Map	Default_DB		
<input type="checkbox"/>	Standard_0002_Map	Table	Standard_Map	Default_DB		
<input type="checkbox"/>	Standard_0004_Map	Table	Standard_Map	Default_DB		

1 - 20 of 83

## Create New Code Set

To add new code set:

- 1 On the **Manage Lookup Data** page, select **Create** from **Actions** dropdown.  
The **Create Code Set** page is displayed.

Figure 81: Create Code Set

**Create Code Set** ?

**Name \***  
Gender

**Description \***  
Gender

**Template table \***  
Default

**Source map table \***  
MST\_STANDARD\_MAP

**Parent Code Set**  
MDM Modules

☒ **Advanced Settings**

**Start Date \***  
01/19/2021

**End Date \***  
12/31/9999

**Editable**  
true

**Source System Id \***  
1:EDW Source System

☒ **Override Code Value Sequence**

**BACK SAVE**

- 2 On the **Create Code Set** page (Figure 81), perform the following:
  - **Name:** enter the code type in the **Name** field (example: Gender, Country, Designation, etc.)



The code set entered would be validated without page refresh and if the entered code set name already exists, then a message would be displayed on the UI as in below figure.

The screenshot shows the 'Create Code Set' dialog box. The 'Name' field contains 'Country'. The 'Description' field is empty. The 'Template table' dropdown is set to 'Default'. The 'Source map table' dropdown is set to 'MST\_STANDARD\_MAP'. A modal message box is displayed in the center with the title 'Message -- Webpage Dialog' and the text 'Code Set with same name already exists.' with an 'OK' button. At the bottom right of the dialog box are 'BACK' and 'SAVE' buttons.

- **Description:** enter the code type description in the **Description** field.
- **Template table:** select the template type.
  - **Default:** if Default type is selected, select the standard map table from the **Standard Map Table** drop-down and if required select the parent code set from the Parent Code Set dropdown.
  - **Multitable:** if Multitable option is selected, select the table from the **Standard Map Table** drop-down.

During MDM installation, if multiple table option selected, the Standard Map Table drop-down would display multiple tables. If single table option selected, the MST\_Standard\_Map Table would be displayed. Refer to *Master Data Management Installation Guide* for details.
- **Customer Table/Views:** On the LRDM UI, the code set values are stored in a single

column in “Reference\_Code” table or in any multi table. LRDM UI provides an option to store the code set values in multiple columns in the selected custom value table or view.

To store code values in custom tables:

- On the **Create Code Set** page, select **Customer Table/View** from **Template** table dropdown.

The **Create Code Set** page refreshes to display Service Name, Table or View Name, Value Column and Description Column dropdowns as in [Figure 82](#).

Figure 82: Create Code Set—Customer Table or View

- Select the service name (Master or Non MDM), table or view, the value column and description column click **Save**.

The value in the Description Column is displayed for the code value selected in Value Column field.



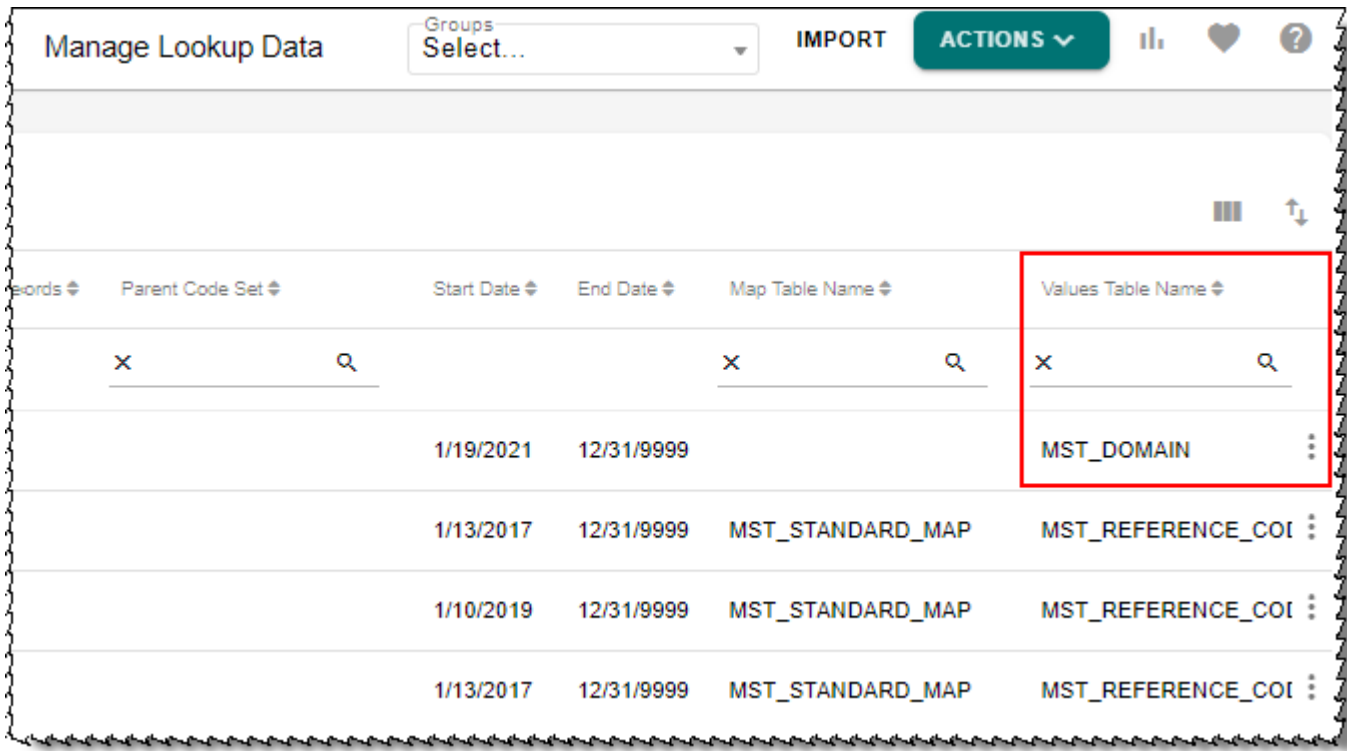
1. During the table creation using Model Builder, for columns in which look-up is defined and if the option to display descriptions of codes is selected, then in the create and edit UI, the lookup drop-down displays the code description instead of code values. For enabling Display Description option in model builder UI, see [Section : “Create or Edit Model”](#) in Chapter Model Builder.

2. Display description is honored on the following modules (Configurable UI, Excel - Import and Export, E2E, Approval Inbox and Approval Monitoring)

The Selected Columns will store all the details of the code values like name, Id, description etc., in # separated format.

When a code set with customer table or view is saved, the **Values Table Name** column on the **Manage Lookup Data** UI displays the Customer Table or View Name as in [Figure 83](#).

Figure 83: Manage Lookup Data—Values Table Name



On the **Manage Lookup Data** UI, for any code set with custom table or view defined, on click of Manage Code Values, it navigates to Configurable UI of the corresponding custom table. On the Configurable UI, you can add and manage code values.

- **Advanced Settings:** click the Advanced Setting checkbox to select the below options:
  - select start and end date for the code type entered.
  - set the **Editable** fields to true/false.By default, Editable fields will be set to true.



If the Editable field is set to False, the following restrictions would be applied on a non-editable Code Set on LRDM UI:

- Code Set cannot be edited or deleted.
  - Code Values cannot be added, edited or deleted.
- 

- enter the source system Id in **Source System Id** field.  
By default, the source system Id field is populated with a value 1.
  - **Override Code Value Sequence:** select the Override Code Value Sequence checkbox to enable overriding the ordering sequencing of the code values on the Manage Code Value UI of the selected code set.  
By default, for all the code value, the Override Code Value Sequence is not enabled and the code values are displayed in alphabetical order on UI. If the Override Code Value Sequence checkbox is enabled, on the Manage Code Values UI, the Manage Sequence button is displayed to override the ordering sequence.
  - Click **Save**.  
Message: “Code Set added successfully” is displayed on the **Manage Lookup Data** page.
- 



- A workflow provided with OOTB MDM to rename the multi tables (Reference Code, Standard Map and Language Description) of LRDM as per the names of Code Set assigned. Any spaces or special characters in the name of code set will be ignored while renaming the LRDM multi tables. The name of the workflow is: RenameDocumentLogicalNameWkf. This workflow can be scheduled using scheduler. This workflow internally looks for all the concerned LRDM and Hierarchy tables that are used but not renamed and renames all. In case of any failure, entire changes are reverted back.
- When RLS is enabled on a Code\_Set table, you will be able to create a code set only when RLS is explicitly assigned to the rows containing CTL\_ID and Categories code sets.
- It is recommended not to create custom code set on Reference\_Code (or on multitable for the same table) as it may display the enterprise record count wrongly.
- If user is adding a custom code set then the count of existing enterprise records will not be updated automatically for the existing enterprise record of the custom code set. User needs to refresh the count through “Refresh Count” button provided on the UI. By default, the business rule "populate\_Stats\_(table\_name)" is inactive. You must first make this business rule active and the record count will be updated automatically.
- Pending approval check is not honored on creating code set or code value with the same name as existing code set name or code value.

## Add Code Values

The Add Code Values functionality created new source code values as well as new enterprise records. To add code values:

- 1 On the **Manage Lookup Data**, click the Menu icon corresponding to the selected code set and click **Code Values**.

The **Manage Code Values** for the selected code set opens.

Figure 84: Manage Code Values

Manage Lookup Data

Groups Select... IMPORT ACTIONS

Search Results: Page 1 of 1

Code Set Id	Name	Description	Associated Tables	Enterprise Records	Mapped Source Records	Unmapped Source Records	Parent Code Set	
250	MDM Modules	List Of MDM Mod...		11	0	0		⋮
220	Entity Type Code	The defined typ...		0	0	0		⋮
216	STATE	States		711			COUNTRY	⋮
215	Status Color Codes	Process status ...		15				⋮
214	COUNTRY	ISO defined cod...		229				⋮
213	Cleansing Tool	Cleansing Tool		2				⋮
212	CRDM Process	CRDM Process		4				⋮
211	DELIVERY_METHOD	Notification Ty...		5				⋮

Code Values  
Deactivate  
Edit

1 - 20 of 20

- 2 Click **Create**.

The **Create Code Values** page opens.

Figure 85: Create Code Values

- 3 On **Create Code Values**, provide following details:
  - **Enterprise Code Value**: name of the code value being created.
  - **Description**: description for the code value.
  - **Start Date**: effective begin date for code value to be valid
  - **End Date**: effective end date for code value to no longer be valid.
  - **Source System ID**: select a valid source system number

Message: “Code Value Added Successfully” displays.

## Edit Code Values

To edit code values:

- 1 On the **Manage Lookup Data**, click the Menu icon corresponding to the selected code set and click **Code Value**.

The **Manage Code Values** for the selected code set opens.

- 2 Under **Manage Code Values**, click the Menu icon for a particular Enterprise data and click **Edit**.



Figure 86: Edit Code Values

Edit Code Value

BACK UPDATE ?

Enterprise Data

CODE VALUE DETAILS CHANGE HISTORY APPROVAL HISTORY

Enterprise Code Value  
Survivorship

Description\*  
Survivorship

Start Date\*  
1/10/2019

End Date\*  
12/31/9999

Source System Id  
1

☒ Advanced Settings

- 3 Modify the required details and click **UPDATE**.

## Set As Default

A code value of a code set can be set as default value. The default value of a code set will be selected by default in create/edit UIs wherever it is referred. Only one code value per code set can be set as default value.

To set code value as default value:

- 1 On the **Manage Lookup Data**, click the Menu icon corresponding to the selected code value and click **Set as Default**.

The selected code value would be displayed by default in the drop-down in the corresponding Create or Edit configurable UIs.

## Create New Source Systems

The Source System Id is an important code set as it includes the source systems. It is the list of valid source systems that sync data with Teradata.

To create a new source system, perform the following steps:

- 1 On the **Manage Lookup Data** UI, search Source Systems code set by entering “Source Systems” in the **Name** column as in [Figure 87](#).

Figure 87: Manage Lookup Data

The screenshot shows the 'Manage Lookup Data' interface. At the top, there's a header with 'Manage Lookup Data', a 'Groups Select...' dropdown, an 'IMPORT' button, and an 'ACTIONS' dropdown. Below the header, a search bar contains 'Source Systems'. The main table displays search results for 'Source Systems' with the following columns: Code Set Id, Name, Description, Associated Tables, Enterprise Records, Mapped Source Records, Unmapped Source Records, Parent Code Set, and Start Date. The table shows one result with Code Set Id '1', Name 'Source Systems', Description 'Source Systems', 1 Enterprise Record, 0 Mapped Source Records, and 0 Unmapped Source Records. The start date is 1/13/20.

Code Set Id	Name	Description	Associated Tables	Enterprise Records	Mapped Source Records	Unmapped Source Records	Parent Code Set	Start Date
1	Source Systems	Source Systems		1	0	0		1/13/20

- 2 Drill into the code values by either clicking Manage Code Values or by clicking on the number of Enterprise Records on Manage Lookup Data UI.

Figure 88: Manage Code Values

The screenshot shows the 'Manage Code Values (Code Set : Source Systems)' interface. At the top, there's a header with 'Manage Code Values (Code Set : Source Systems)', a 'BACK' button, an 'IMPORT' button, and a 'CREATE' button. Below the header, there's a search bar and a table displaying code values. The table has columns: Enterprise Code, Enterprise Code Value, Enterprise Description, Parent Enterprise Code Value, Start Date, End Date, Is Default Value, Created By, Creation Date, Last Modified By, and Last Modified Date. The table shows one result with Enterprise Code '1', Enterprise Code Value '1', Enterprise Description 'EDW Source System', Start Date '1/13/2017', End Date '12/31/9999', Is Default Value 'false', Created By 'BackEnd', and Creation Date '4/26/2022, 3:43 AM'.

Enterprise Code	Enterprise Code Value	Enterprise Description	Parent Enterprise Code Value	Start Date	End Date	Is Default Value	Created By	Creation Date	Last Modified By	Last Modified Date
1	1	EDW Source System		1/13/2017	12/31/9999	false	BackEnd	4/26/2022, 3:43 AM		

On the Manage Code Values UI, the EDW Source System is the primary conformed set. In these examples, the “EDW Source System” is considered as “EDW Target System”. The values under the Source System Id = “1” are the global list of conformed enterprise values.

- 3 On the **Manage Code Values** UI ([Figure 88](#)), click **Create** to add new source system.
- 4 On the **Create Code Value**, enter the required data and click **Save**.  
Message: “Code Value added successfully” is displayed.

## Create Source Mapping

Manage lookup data UI allows you to drill into unmapped source records and manually map to the appropriate enterprise value.

To map unmapped source records manually, perform the following steps:

- 1 On the **Manage Lookup Data** UI, click the count of unmapped records in the “Unmapped Source Records” column of the code set.
- 2 On the **Manage Mapping** UI, click **Create**.
- 3 On the **Add Mapping** UI, enter the details as in [Figure 89](#) and click **SAVE**.

Figure 89: Manage Lookup Data

Add Mapping BACK SAVE ?

Code Set Details

Mapping

Enterprise Code Value  
Select...

Standard Map Details

Source Code \*  Description \*

☐ Advanced Settings

The below confirmation message is displayed.

Figure 90: Confirmation Pop-up

Message -- Webpage Dialog ✕

The Source Code will be saved as unmapped record. Do you want to continue?

YES NO

- 4 On the Confirmation pop-up click **Yes**.  
Message: “Mapping added successfully” is displayed. Similarly add the required mappings.
- 5 On the **Manage Lookup Data** UI, click on the number of unmapped records in the “Unmapped Source Records” column of the code set as in [Figure 91](#).

Figure 91: Manage Lookup Data

Manage Lookup Data

Search Results: Page 1 of 2

Code Set Id #	Name #	Description #	Associated Tables	Enterprise Records #	Mapped Source Records #	Unmapped Source Records #
1201	US States	US States	2	0	1	...
1101	Department	Department	3	0	0	...
1	Source Systems	Source Systems	1	0	0	...
250	MDM Modules	List Of MDM Mod...	11	0	0	...
220	Entity Type Code	The defined typ...	0	0	0	...
216	STATE	States	711			...
215	Table Source	Table Source	2			...

1 - 20 of 22

- 6 On the **Manage Mapping** UI, select the value to be mapped and click **Edit** as in [Figure 92](#).

Figure 92: Manage Mapping

Manage Mapping

BACK DELETE IMPORT EDIT CREATE ?

Code Set Details

Filter Mappings

Status: UnMapped Source System Id: All Enterprise Code Value: Select...

CLEAR SEARCH

Mapped Values: Page 1 of 1

Source System Id #	Source Code #	Source Description #	Enterprise Code Value #	Enterprise Code #
1:EDW Source System	New York	New York	-1	

1 - 1 of 1

- 7 On the **Multi Edit Mapping** UI, select the correct Enterprise Code Value in the dropdown and click **Save** as in [Figure 93](#).

Figure 93: Multi Edit Mapping

Multi Edit Mapping

Code Set Details

Search Results

Source Code

Source Code

New York

Source System Id

1:EDW Source System

Enterprise Code Value

Select...

Description

New York

Start Date

3/26/2021

End Date

12/31/9999

Source Code Value

BACK

SAVE

The system will return to the unmapped source records and display a success message. Also the value will no longer be displayed within the unmapped records as in [Figure 94](#).

Figure 94: Manage Mapping

The screenshot displays the 'Manage Mapping' interface. At the top, there are navigation buttons: BACK, DELETE, IMPORT, EDIT, and a prominent CREATE button. A green checkmark icon and the text 'Mapping edited successfully.' are visible. Below this, there are expandable sections for 'Code Set Details' and 'Filter Mappings'. The 'Filter Mappings' section contains three dropdown menus: 'Status' (set to 'UnMapped'), 'Source System Id' (set to '1:EDW Source System'), and 'Enterprise Code Value' (set to 'Select...'). To the right of these filters is a 'CLEAR' button and a 'SEARCH' button. Below the filters, a message states 'Mapped Values: No Records Found'. At the bottom, there is a table header with columns: 'Source System Id', 'Source Code', 'Source Description', 'Enterprise Code Value', and 'Enterprise Code'. Each column has a search icon and a dropdown arrow. The table body is currently empty, and a pagination bar at the bottom right shows '1 - 0 of 0'.

- 8 On the **Manage Mappings** UI, search the mapped source records with **Show Mapped** checkbox selected to confirm that the New York source code is now correctly mapped to the enterprise code value as in [Figure 95](#).

Figure 95: Manage Mapping

Manage Mapping

BACKDELETEIMPORTEDITCREATE?

Code Set Details

Filter Mappings

Status

Source System Id

Enterprise Code Value

Mapped

1:EDW Source System

Select...

CLEAR

SEARCH

Mapped Values: Page 1 of 1

<input type="checkbox"/>	Source System Id #	Source Code #	Source Description #	Enterprise Code Value #	Enterprise Code #
<input type="checkbox"/>	1:EDW Source System	New York	New York	california	50201

1 - 1 of 1

<>>>

# Load from Excel

Teradata RDM allows you to upload a list of new source records to a code set using Excel.



- 1 On the Manage Lookup Data UI, click on the number of unmapped records in the **Unmapped Source Records** column of the code set as in [Figure 96](#).

Figure 96: Manage Lookup Data

Manage Lookup Data

Groups: Select... **IMPORT** **ACTIONS**

Search Results: Page 1 of 2 1

Code Set Id	Name	Description	Associated Tables	Enterprise Records	Mapped Source Records	Unmapped Source Records	
1301	Brand Master	Brand Master		0	0	0	
1201	US States	US States		2	0	1	
1101	Department	Department		3	0	0	
1	Source Systems	Source Systems		1	0	0	
250	MDM Modules	List Of MDM Mod...		11	0	0	
220	Entity Type Code	The defined typ...		0	0	0	

1 - 20 of 23 |< < > >|

- 2 On the **Manage Mapping** UI, click **Import** as in [Figure 97](#).

Figure 97: Manage Mapping

Manage Mapping

BACKDELETEIMPORTEDITCREATE?

Code Set Details

Filter Mappings

Status

UnMapped

Source System Id

All

Enterprise Code Value

Select...

CLEAR

SEARCH

Mapped Values: No Records Found

Source System Id

Source Code

Source Description

Enterprise Code Value

Enterprise Code

x

x

x

x

=

x

1 - 0 of 0

|<

<

>

>|

- 3 On the **Upload Mapping Values** UI, click **Choose File** to navigate to the excel file that includes the necessary columns to upload the new source codes and then click **Upload** as in [Figure 98](#).

Figure 98: Upload Mapping Values

BACK
UPLOAD
↑↓
?

Code Set Details

Mapping

Enterprise Code Value  
Select...

Upload Mapping Values

File\*  
Choose File Brandmaster1.xlsx

Figure 99 displays sample Excel file that includes 22 new rows of source codes for the BrandMaster table.

Figure 99: Sample Excel File—Brand Master Table

Code_Set	Ctl_Id	EDW_Cod	Source_Code	Description	Domain_Id	End_Date	Process_Id	Process_N	Record_D	Start_Date	Other_Description
1252	1	-1	ALTAR	ALTAR	1	12/31/999	-1	MANUAL	0	3/28/2019	UNKNOWN
1252	1	-1	ALTEWF	ALTEWF	1	12/31/999	-1		0	3/28/2020	UNKNOWN
1252	1	-1	AMBERL	AMBERL	1	12/31/999	-1		0	3/28/2021	UNKNOWN
1252	1	-1	AMCLS	AMCLS	1	12/31/999	-1		0	3/28/2022	UNKNOWN
1252	1	-1	ARCEAS	ARCEAS	1	12/31/999	-1		0	3/28/2023	UNKNOWN
1252	1	-1	ASPLWF	ASPLWF	1	12/31/999	-1		0	3/28/2024	UNKNOWN
1252	1	-1	BARBWG	BARBWG	1	12/31/999	-1		0	3/28/2025	UNKNOWN
1252	1	-1	BARU	BARU	1	12/31/999	-1		0	3/28/2026	UNKNOWN
1252	1	-1	BAYS	BAYS	1	12/31/999	-1		0	3/28/2027	UNKNOWN
1252	1	-1	THATS IT	THATS IT	1	12/31/999	-1		0	3/28/2028	UNKNOWN
1252	1	-1	TIGERS MLK	TIGERS MLK	1	12/31/999	-1		0	3/28/2029	UNKNOWN
1252	1	-1	TIMPONE'S	TIMPONE'S	1	12/31/999	-1		0	3/28/2030	UNKNOWN
1252	1	-1	TINYMIGHTY	TINYMIGHTY	1	12/31/999	-1		0	3/28/2031	UNKNOWN
1252	1	-1	TIPUS CHAI	TIPUS CHAI	1	12/31/999	-1		0	3/28/2032	UNKNOWN
1252	1	-1	TODOS DIRT	TODOS DIRT	1	12/31/999	-1		0	3/28/2033	UNKNOWN
1252	1	-1	TOP SECRET	TOP SECRET	1	12/31/999	-1		0	3/28/2034	UNKNOWN
1252	1	-1	TOSCHI	TOSCHI	1	12/31/999	-1		0	3/28/2035	UNKNOWN
1252	1	-1	TRAINA	TRAINA	1	12/31/999	-1		0	3/28/2036	UNKNOWN
1252	1	-1	TREE HUGGR	TREE HUGGR	1	12/31/999	-1		0	3/28/2037	UNKNOWN
1252	1	-1	TRU COCO	TRU COCO	1	12/31/999	-1		0	3/28/2038	UNKNOWN
1252	1	-1	TRUE LEMON	TRUE LEMON	1	12/31/999	-1		0	3/28/2039	UNKNOWN
1252	1	-1	TRUE LIME	TRUE LIME	1	12/31/999	-1		0	3/28/2040	UNKNOWN

The system redirect you to the View Reports UI, which displays the status of the upload. Initially the status is displayed as “Progress” to indicate that the upload is processing.

- 4 On the **View Reports** UI, click Refresh icon to refresh the 'Progress' status until the process is completed.
- 5 Navigate back to the Manage Lookup Data UI and find the code set (Brand Master) that includes the new source codes. The number of unmapped records has increased by 22 due to the new records in the Excel.

Figure 100: Manage Lookup Data

Name	Description	Associated Tables	Enterprise Records	Mapped Source Records	Unmapped Source Records
Brand Master	Brand Master		0	0	22
US States	US States		2	0	1
Department	Department		3	0	0
Source Systems	Source Systems		1	0	0
MDM Modules	List Of MDM Mod...		11	0	0
Entity Type Code	The defined b...		0	0	0

## Auto Matching Logic

Teradata RDM also provides the functionality to set up auto matching logic for new unmapped source records. Perform the following steps to initialize the functionality:

- 1 Navigate to **Matching profiles (Data Harmonization -> Matching)** and select the correct matching profile for the new unmapped source codes and click **Execute**.

For detailed description on creating matching profiles, refer to *Master Data Management Server Guide.pdf*.

Figure 101: Matching Profiles

Matching profile					
					?
<input type="checkbox"/>	Name	Description	Profile type	Last modified date ↓	
<input type="checkbox"/>	SRC_WEB_US_Customer_Match		Match	Apr 29, 2022, 6:09:52 AM	⋮
<input type="checkbox"/>	SRC_SAPNA_Customer		Match	Apr 29, 2022, 6:09:29 AM	Edit
<input type="checkbox"/>	Manual_SAPNA_Customer_Match		External	Apr 29, 2022, 6:09:23 AM	Execute
<input type="checkbox"/>	Manual_WEB_US_Customer_Match		External	Apr 29, 2022, 6:09:18 AM	Result
<input type="checkbox"/>	SRC_CCENTER_US_Customer_Match		Match	Apr 29, 2022, 6:08:35 AM	Delete
<input type="checkbox"/>	SRC_LEGACY_US_Customer_Match		Match	Apr 29, 2022, 6:08:30 AM	⋮
<input type="checkbox"/>	Manual_CCENTER_US_Customer_Match		External	Apr 29, 2022, 6:08:26 AM	⋮
<input type="checkbox"/>	Manual_LEGACY_US_Customer_Match		External	Apr 29, 2022, 6:08:22 AM	⋮
<input type="checkbox"/>	External_M		External	Apr 26, 2022, 5:37:22 PM	⋮
Items per page: 25 1 – 9 of 9 < >					

- Click on the Menu icon corresponding to the **Matching Profiles** and click **Result** to view the number of matched rows, potential match rows and new rows of the executed match profile.
- Navigate to Survivorship profiles (**Data Harmonization -> Survivorship**) and select the related profile and click **View Match Records**.  
For detailed description on creating survivorship profiles, refer to *Master Data Management Server Guide.pdf*.
- Navigate back to **Manage Lookup Data** UI to see that the number of records under unmapped source records column is decreased and the number of mapped source records is increased.

## Code Set Groups

The below section describes the following:

- [Grouping of Code Sets](#)
- [Grouping of Code Value Attributes](#)

## Grouping of Code Sets

The grouping feature in LRDM allows you to logically group similar code sets. By default, the MDM Code Sets group is available and includes all the OOTB code sets used in different features of MDM.

The following points lists the grouping concepts in LRDM:

- Create a new group and assign members (code sets) to a group.
- View the members (code sets) of the existing groups.
- Search and filter code sets based on group names.
- All the OOTB code sets can be assigned to the existing groups.
- Assign newly created code sets to a group.
- Single code set can be assigned to multiple groups.

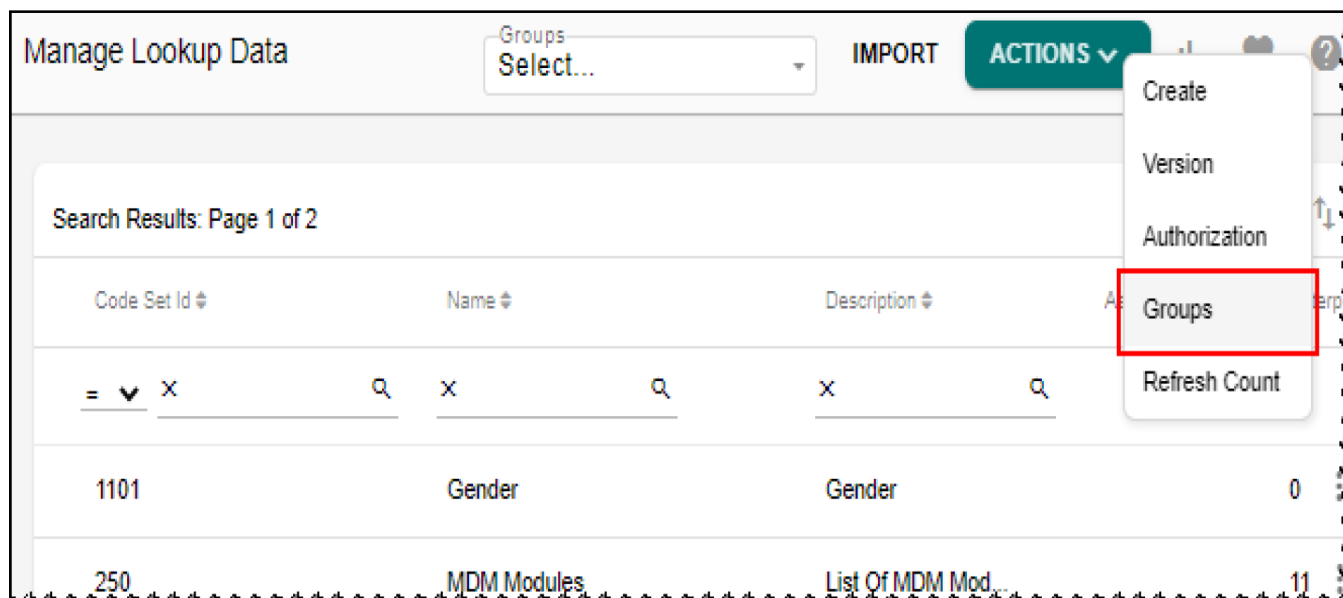
The following sections describes the detailed steps on creating groups, assigning members and remove members of the selected group.

### Adding Groups

To add new group:

- 1 On the navigation pane, navigate to **Manage Lookup Data** UI from **Reference Data** -> **Manage Reference Data**.

Figure 102: Manage Lookup Data



- 2 On the **Manage Lookup Data** UI (Figure 102), select **Groups** from the **Actions** dropdown. The **Manage Lookup Data** UI refreshes to display the **Create Group**, **Assign Members** and **Remove Members** options as in Figure 103.

Figure 103: Manage Lookup Data

Manage Lookup Data

Groups  
Select...

BACK IMPORT ACTIONS

CREATE GROUP ASSIGN MEMBERS REMOVE MEMBERS

Search Results: Page 1 of 2

Code Set Id	Name	Description	Associated Tables	Enterp
1101	Gender	Gender		0

3 On the **Manage Lookup Data** UI, click **Create Group**.

Figure 104: Manage Group

Manage Group

1 Value(S) Selected SAVE

Category Name  
Code Set Categories

Category Description  
Defines Code Set Categories

Group Name  
MDM Modules Group

Group Description  
MDM Modules Group

Code\_Set: Page 1 of 1

Code_Set_Id	Description	Map_Table_Name	Start Date
1	Source Systems	MST_STANDARD_MAP	1/13/2017 12:00 AM
250	List Of MDM Modules	MST_STANDARD_MAP	1/10/2019 12:00 AM

1 - 20 of 20

- 4 On the **Manage Group** page (Figure 104),
- enter the group name in **Group Name** field.
  - enter the description in the **Group Description** field.

- select the code sets that you want to assign to the group and click **Save**.

Message: “New Group created successfully” is displayed on the Manage Lookup Data UI.

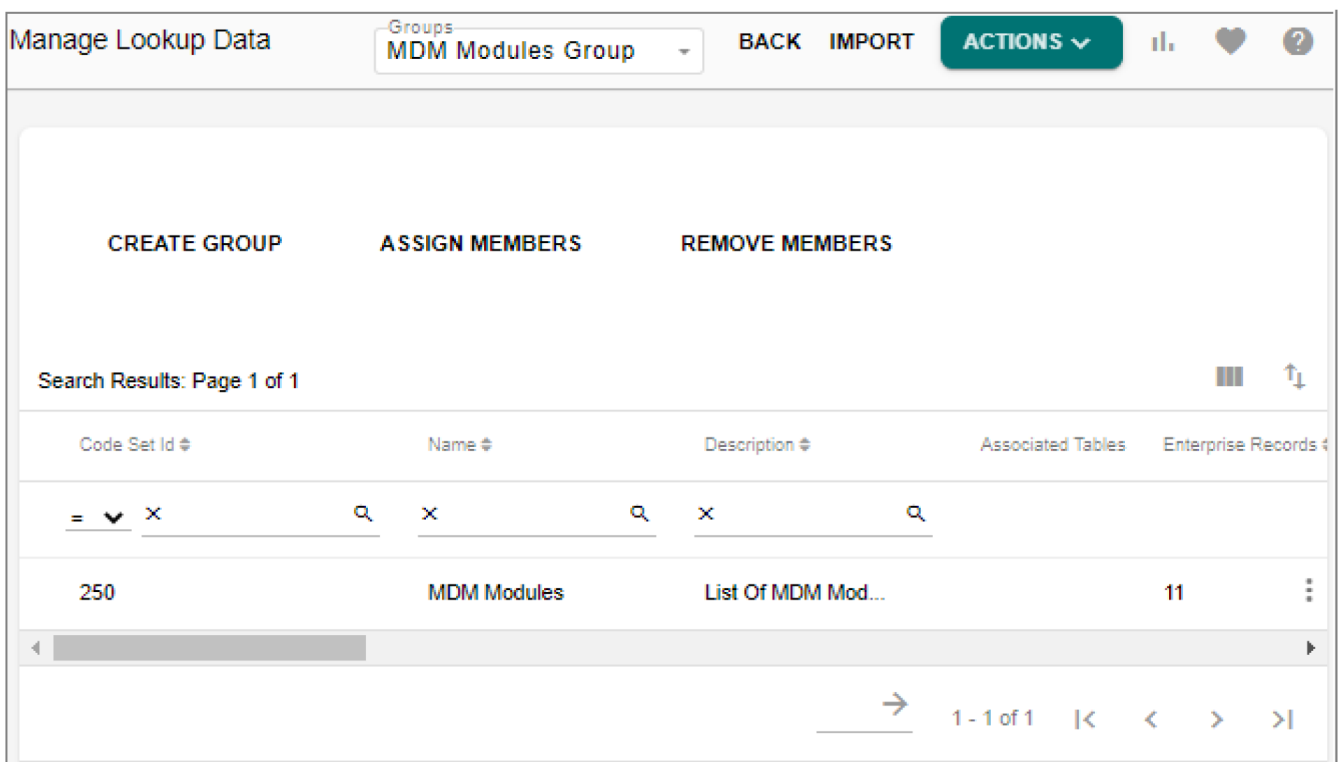
You can also add the members to the created group later as in [Section : “Assign Code Sets \(Members\) to Group”](#).

## Assign Code Sets (Members) to Group

To assign members to the selected group.

- 1 On the **Manage Lookup Data** UI ([Figure 102](#)), select **Groups** from the **Actions** dropdown.  
The **Manage Lookup Data** UI refreshes to display the Group Create, Assign Members and Remove Members options as in [Figure 103](#).
- 2 On the **Manage Lookup Data** UI, select the required group from the **Groups** dropdown.  
The **Manage Lookup Data** UI refreshes to display all the assigned members of the selected group as below.

Figure 105: Manage Lookup Data



- 3 On the **Manage Lookup Data** UI ([Figure 106](#)), click **Assign Members**.  
The **Manage Group** UI ([Figure 106](#)) is displayed. Code sets that are already assigned to the selected group will not be displayed on the Manage Group UI.



Figure 106: Manage Group

Manage Group 1 Value(S) Selected **SAVE**

Category Name  
Code Set Categories

Category Description  
Defines Code Set Categories

Group Name  
MDM Modules Group

Group Description  
MDM Modules Group

Code\_Set: Page 1 of 1

<input type="checkbox"/>	Code_Set_Id	Description	Map_Table_Name	Start Date
<input type="checkbox"/>	214	Matching Tool Params		1/13/2
<input checked="" type="checkbox"/>	213	CRDM Process		1/13/2
<input type="checkbox"/>	212	ISO defined codes for the Countries of Earth		1/13/2

1 - 20 of 20

- On the **Manage Group** UI (Figure 106), select the members (code sets) and click **Save**.  
Message: “Member/Members assigned to the Group successfully” is displayed on the **Manage Lookup Data** UI.

### Remove Code Sets (Members) Assigned to Group

To remove members of the selected group.

- On the **Manage Lookup Data** UI (Figure 102), select **Groups** from the **Actions** dropdown.  
The **Manage Lookup Data** UI refreshes to display the Group Create, Assign Members and Remove Members options as in Figure 103.
- On the **Manage Lookup Data** UI, select the required group from the **Groups** dropdown.  
The **Manage Lookup Data** UI refreshes to display all the assigned members of the selected group as below.

Figure 107: Manage Lookup Data

Manage Lookup Data

GroupsMDM Modules Group

BACKIMPORT

ACTIONS

CREATE GROUP

ASSIGN MEMBERS

REMOVE MEMBERS

Search Results: Page 1 of 1

Code Set Id	Name	Description	Associated Tables	Enterprise Records
=	x	x	x	
250	MDM Modules	List Of MDM Mod...	11	
213	CRDM Process	CRDM Process	4	

1 - 2 of 2

- On the **Manage Lookup Data** page (Figure 107), click **Remove Members**.  
The **Manage Group** page (Figure 106) is displayed.

Figure 108: Manage Group

Manage Group

1 Value(S) Selected DELETE

Category Name

Code Set Categories

Group Name \*

MDM Modules Group

Category Description

Defines Code Set Categories

Group Description

MDM Modules Group


Code\_Set: Page 1 of 1

<input type="checkbox"/>	Code_Set_Id	Description	Map_Table_Name	Start Date	
<input type="checkbox"/>	250	List Of MDM Modules	MST_STANDARD_MAP	1/10/2019 12:00 AM	⋮
<input checked="" type="checkbox"/>	213	CRDM Process		1/13/2017 12:00 AM	⋮

→

1 - 2 of 2

|< < > >|

- On the **Manage Group** page (Figure 106), click on the Menu icon  corresponding to the required code set and click **Delete**.

The confirmation popup is displayed as below.

- On the confirmation popup, click **Yes**.

Message: “Group edited successfully” is displayed on the Manage Loolup Data UI.

## Delete Group



You can delete groups only from the Manage Categories UI.

To delete group from Manage Categories UI:


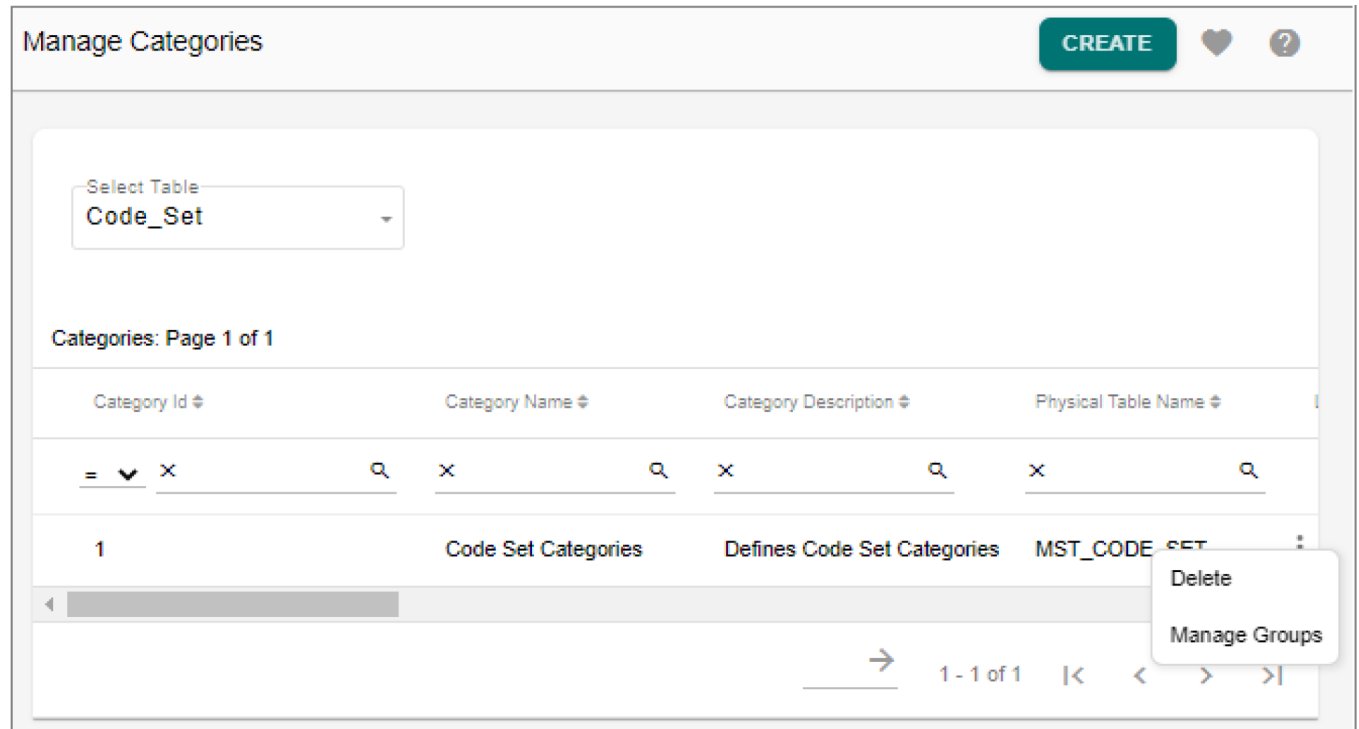
- 1 Navigate to **Manage Categories** UI (**Model Setup -> Categorize Data**) and on the **Manage Categories** UI, select the table as Code Set and click on the Menu icon  corresponding to the Code Set Categories and click **Manage Groups** as in [Figure 109](#)

Figure 109: Manage Categories




- 2 On the **Manage Groups and Attribute Sets** UI, with **Groups** option selected, click on the Menu icon  corresponding to the required group and click **Delete** as in [Figure 110](#).

Figure 110: Manage Groups and Attribute Sets

Manage Groups And Attribute Sets

BACK

CREATE

?

Category Name

Code Set Categories

Category Description

Defines Code Set Categories

☒ Groups

☐ Attribute Sets

Groups: Page 1 of 1

Group Id	Group Name	Group Description
101	MDM Modules Group	MDM Modules Group
1	MDM Code Sets	Group of OOTB MDM Code Sets

→

1 - 2 of 2

|<

<

Delete

View Logs

Assign Members

Assign Values

Apply Values

The confirmation popup is displayed as below.

Message -- Webpage Dialog

Do you want to delete selected group? This will delete associated Group Members also.

YES

NO

- On the confirmation popup, click **Yes**.  
Message: “Group deleted successfully” is displayed.

## Grouping of Code Value Attributes

In case of Multi table LRDM code values, additional attributes can be added through model builder. The added additional attributes for a code set can be grouped into logical groups on the Group Attributes UI.

Pre-requisites for creating groups through Group Attributes UI

- 1 Once the MDM is installed, create Reference\_Code tables using Model Builder through template based table creation option. Add additional columns for the selected table and generate ISG.
- 2 Create code sets by selecting the required standard map table. For details, see [Section : “Create New Code Set.”](#)

Navigate to Attributes Group Dashboard from left navigation pane **Reference Data -> Manage Reference Data** and click graphical view icon. On the **Reference Data Dashboard**, select **Group Attributes** from **Actions** dropdown.

On the Group Attributes UI, you can perform the following tasks:

- [Create Group](#)
- [Edit and Delete Group](#)

### Create Group

To create group:

- 1 On the **Attribute Group Dashboard** UI, click **Create** as in [Figure 111](#) or navigate to tabular view by clicking the tabular icon and on the **Manage Attribute Group** UI, click **Create** as in [Figure 112](#).

Figure 111: Attribute Groups Dashboard

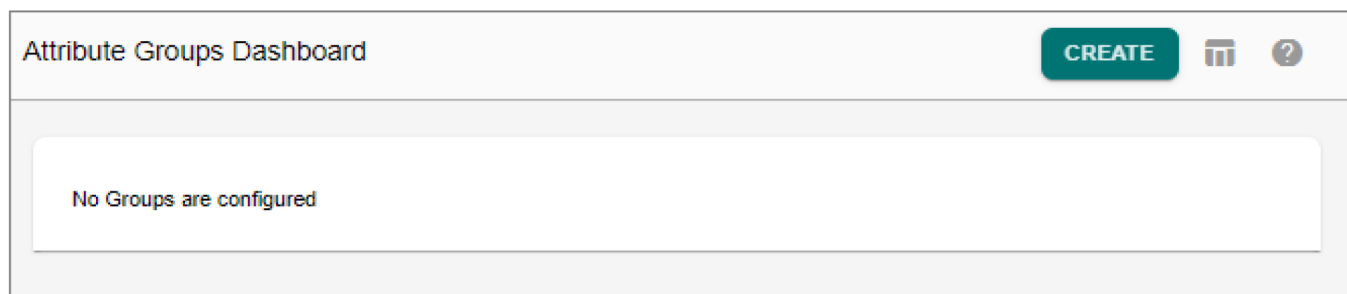


Figure 112: Manage Attribute Groups

Manage Attribute Groups

BACKCREATE?

Attribute Groups: No Grouped Code Values found

Table Name	No. of Groups
x	x

The **Create Attribute Groups** UI refreshes to displays the group addition details.

Figure 113: Create Attribute Groups

Create Attribute Groups

?

Table Name

Product\_Type\_Value

Group1

Group Name\*

Display Name\*

Description

Available Attributes

Billing\_Address  
Billing\_Id  
Billing\_Name  
Contact\_Details  
Contact\_Email\_Id  
Shipping\_Address  
Shipping\_Id

Selected Attributes

>>  
>  
<  
<<

BACKDELETEADDSAVE

- 2 On the **Create Attribute Groups** UI (Figure 113), select the table to which you want to add group and enter the below details:
  - **Group Name:** name of the group being created.
  - **Display Name:** display name of the group being created.
  - **Description:** brief description about the group being created.
  - **Available Attributes:** displays all the additional attributes or column available for the group. Select all the required attributes from the Available Attributes list and move it to the Selected Attributes list.
  - **Selected Attributes:** displays the list of attributes or columns added to the group.
- 3 Click **Add** to add another group.
- 4 You can also select group(s) by selecting the corresponding checkbox and click **Delete** to delete the group.
- 5 Click **Save** to save all the groups details.

Figure 114: Create Attribute Groups

The screenshot displays the 'Create Attribute Groups' web interface. At the top, the title 'Create Attribute Groups' is visible with a help icon. Below the title, a 'Table Name' dropdown menu is set to 'Product\_Type\_Value'. Underneath, a section titled 'Group1' (with an expand/collapse arrow) contains a checkbox and three input fields: 'Group Name' with the value 'Shipping', 'Display Name' with the value 'Shipping', and 'Description' with the value 'Shipping Details'. To the right of these fields are two lists of attributes. The 'Available Attributes' list contains: Billing\_Address, Billing\_Id, Billing\_Name, Contact\_Details, Contact\_Email\_Id, and Shipping\_Id. The 'Selected Attributes' list contains: Shipping\_Address. Between these two lists are four navigation buttons: '>>', '>', '<', and '<<'. At the bottom right of the interface, there are four buttons: 'BACK', 'DELETE', 'ADD', and 'SAVE'.

Message: “Groups created successfully” is displayed on the Attribute Dashboard page as in Figure 115.

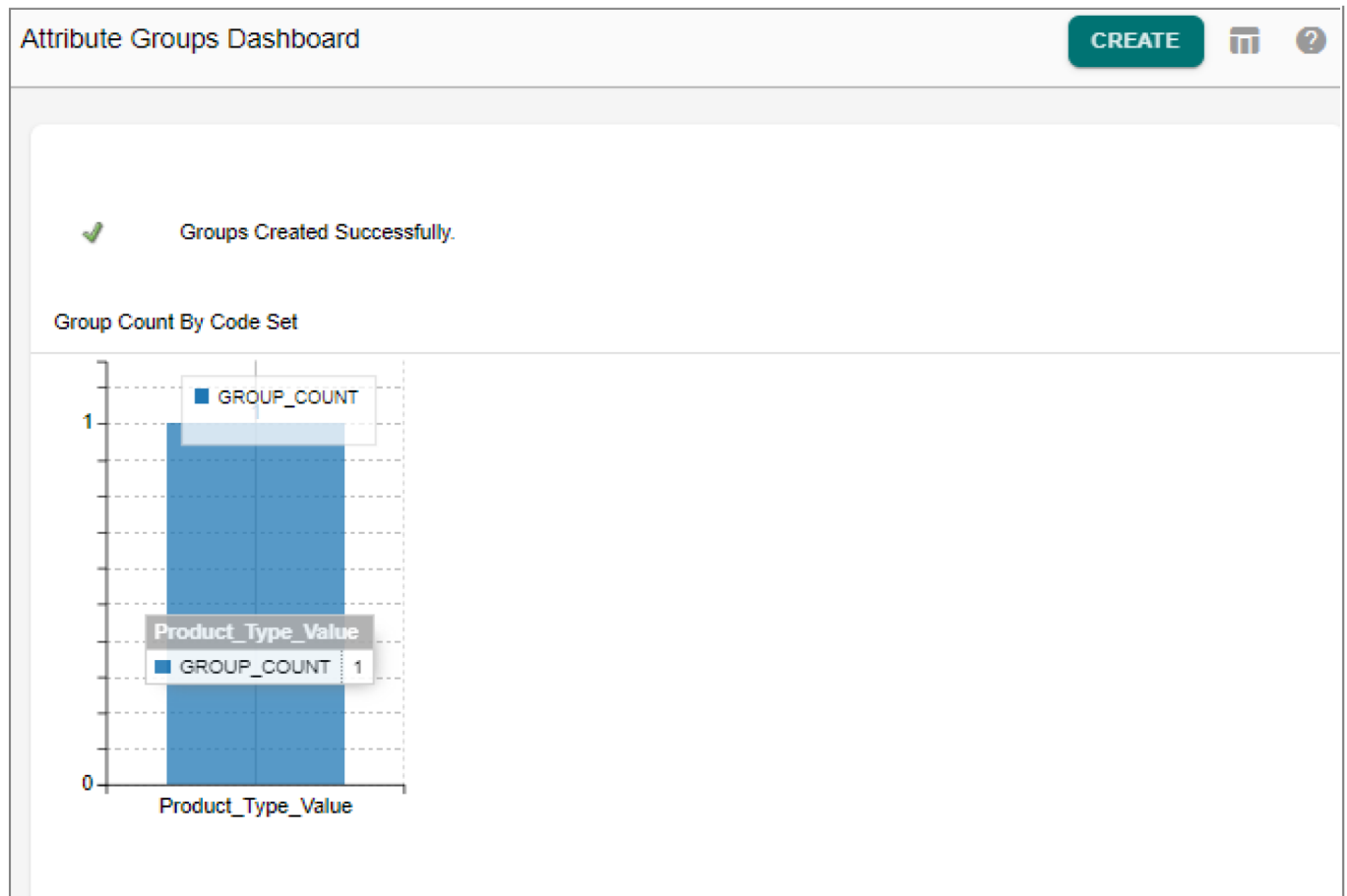
On the **Attributes Group Dashboard**,

- A bar graph displays the number of groups per code set.



- The graphical view can be changed to tabular view by clicking on the Tabular icon on the right corner of the page.
- On click of a code set, the edit page of the corresponding code set groups is displayed.
- Navigate to create group page by clicking the Create button.

Figure 115: Attribute Groups Dashboard



## Edit and Delete Group

To edit group details:



- 1 On the **Attribute Group Dashboard** UI (Figure 115), click on the code set or navigate to tabular view by clicking the tabular icon  and on the **Manage Attribute Group** UI, click on the Menu icon  corresponding to the code set and click **Edit** as in Figure 116.

Figure 116: Manage Attribute Groups

Manage Attribute Groups

BACK CREATE ?

Attribute Groups

Table Name	No. of Groups
Product_Type_Value	1

Delete Edit

- 2 On the **Create Attribute Groups** UI, modify the existing group details or add new groups and click **Update** as in [Figure 117](#).

Figure 117: Create Attribute Groups

Create Attribute Groups ?

Table Name  
Product\_Type\_Value

Shipping

Group Name\*  
Shipping

Display Name\*  
Shipping

Description  
Shipping Details

Available Attributes

Billing\_Address  
Billing\_Id  
Billing\_Name  
Contact\_Details  
Contact\_Email\_Id  
Shipping\_Id

Selected Attributes

Shipping\_Address

BACK DELETE ADD UPDATE

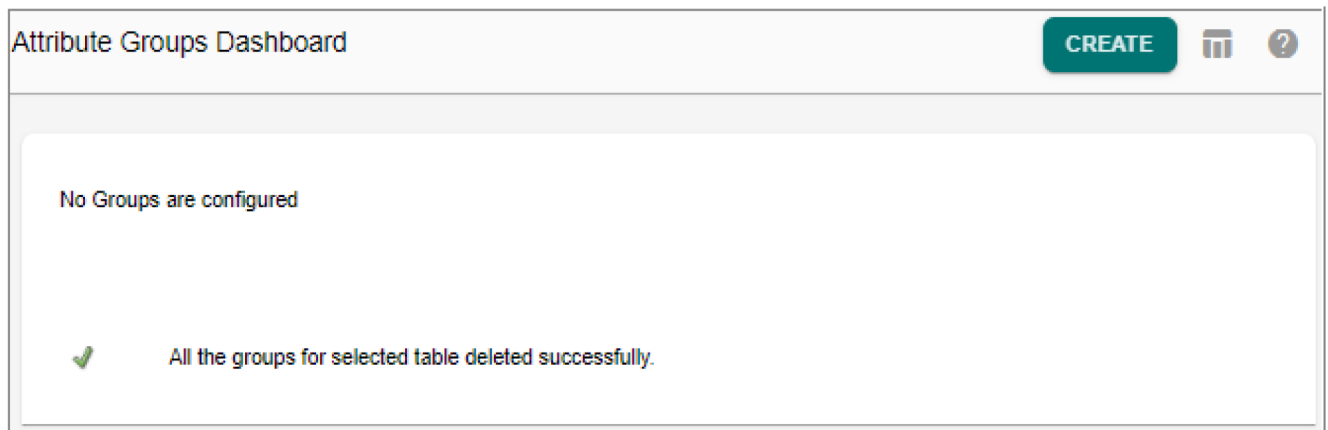
Message: “Groups updated successfully” is displayed.

### To delete group:

You can delete the groups either by using step 1 or 2.

- 1 On the **Attribute Group Dashboard** UI, click on the code set and on the **Manage Attributes Groups** UI select group(s) by selecting the corresponding checkbox and click **Delete**.
- 2 Navigate to tabular view by clicking the tabular icon and on the **Manage Attribute Group** UI, select the code set and click **Delete** to delete all the groups of the selected code set as in [Figure 118](#).

Figure 118: Attribute Groups Dashboard



### Add and Edit Code Values

The Add and Edit Code Values UI displays the added additional attributes under each group in separate containers as in [Figure 119](#) and [Figure 120](#). The container name is the group name. You can add or edit the code values of the required columns and click save to save the values to the corresponding table.

Figure 119: Create Code Value

Create Code Value

Code Set Details

Code Set Id

1101

Standard Map Table

MST\_STANDARD\_0001\_MAP

Name

Product\_Type

Start Date

1/23/2021

Description

Product\_Type

End Date

12/31/9999

Created By

USR\_1

Creation Date

1/23/2021 4:28 AM

Last Modified By

Last Modified Date

Code Value Details

Enterprise Code Value

Office\_Products

Description

Office\_Products

☒ Advanced Settings

Start Date

01/23/2021

End Date

12/31/9999

Source System Id

1:EDW Source System

Shipping

Shipping\_Address

Additional Attributes

Shipping\_Id

Billing\_Name

BACK

SAVE

Figure 120: Edit Code Value

?

Enterprise Data

CODE VALUE DETAILS

CHANGE HISTORY

APPROVALHISTORY

Enterprise Code Value

Office\_Products

Description \*

Office\_Products

☒ Advanced Settings

Start Date \*

1/23/2021

End Date \*

12/31/9999

Source System Id

1

Shipping

Shipping\_Address

Additional Attributes

Shipping\_Id

Billing\_Name

Contact\_Details

Billing\_Address

Contact\_Email\_Id

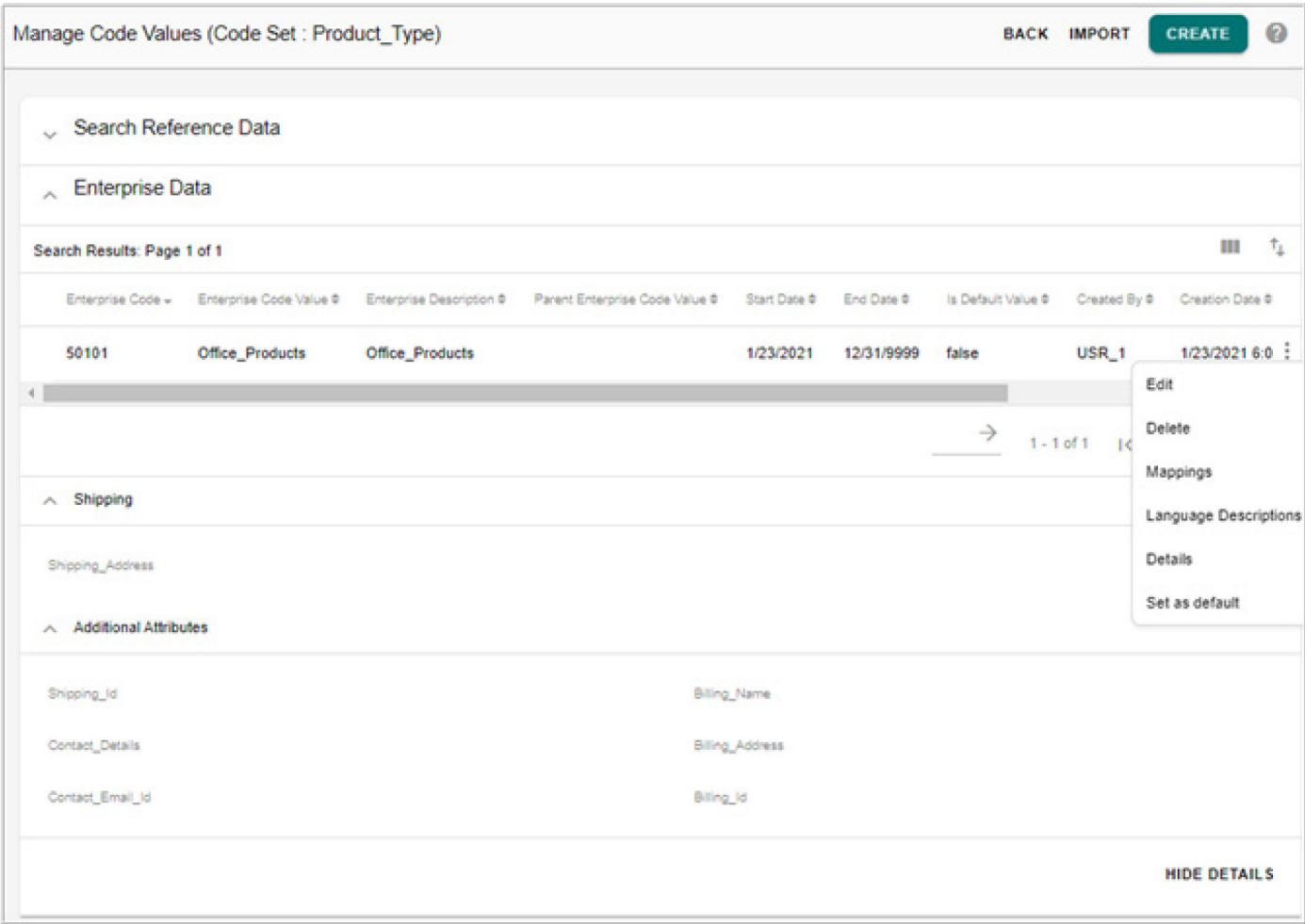
Billing\_Id

BACK

UPDATE

The Manage Code Value—View Details UI (Figure 121) displays all the added additional extra attribute values of the selected code value according to groups. The unassigned attributes will be available under Additional Attributes list.

Figure 121: Manage Code Values—View Details



## Source System Association

The RDM provides an option to create a table for source system association to maintain and view the source data. You can associate multiple source Tables to each Code Set. Each code value can be mapped to a source code and its attributes available in the Source Tables. The source system association provides the following capabilities:

- Source System Association allows to associate source systems for code set mappings.
- For any code set, multiple source systems can be mapped.
- Source systems are master tables of MDM and can be generated through model builder template.
- Source system association is supported on code sets created on multi table set-up and not supported for the out of the box code sets.

The below lists outlines the various steps involved in the source system association process:

- [Generate RDM Source Tables and Populate Data](#)

- [Create Code Set](#)
- [Create Source Table Association](#)
- [Create Code Values](#)
- [Map Code Values to Source](#)

## Generate RDM Source Tables and Populate Data

Perform the following steps to generate RDM source tables through Model Builder UI:

- 1 Navigate to **Model Builder—Manage Template** UI from left navigation pane (**Model Setup** - > **Model Builder**).
- 2 Select the **RDM Template** and **SourceMap Template** option and click **Generate Model** as in [Figure 122](#).

Figure 122: Manage Template

Manage Template

BACK GENERATE MODEL ?

Template Type  
RDM Template

☒ SourceMap Template

Enter number of tables to b...  
1

Search Templates: Page 1 of 1

<input type="checkbox"/>	Template Name	Folder Name	Generated Table Count	Modified By	Modified Date
<input checked="" type="checkbox"/>	Source_Map	LRDM	1		

1 - 1 of 1

Once the source tables are generated, populate the source tables through configurable UI or through Excel Upload.

- 3 Navigate to corresponding configurable UI (source map table) from left navigation pane (**Tables** -> **Master Tables** -> **LRDM**).

- 4 On the Source Map UI, click **Create** and on the Create Source Map UI, enter the details and click **Create** as in [Figure 123](#).

Figure 123: Create Source Map

The screenshot shows the 'Source\_0001\_Map' interface. At the top right, it says '1 Value(S) Selected' and has a 'SAVE' button. Below this is a search bar and a table with columns: Source\_Id, Source\_Code, Description, and End Date. The table contains one row with the following data:

Source_Id	Source_Code	Description	End Date
1	S_Plastics	Plastic_Non Disposable	

The 'Source\_Id' column has a checkbox next to the value '1'. The bottom of the interface shows pagination: '1 - 1 of 1'.

The populated data is displayed on the Source Map UI as in [Figure 124](#).

Figure 124: Source Map

The screenshot shows the 'Source\_0001\_Map' interface with two entries in the table. At the top right, there are buttons for 'UPLOAD EXCEL', '+', a grid icon, a heart icon, and a question mark icon. The table has columns: Source\_Id, Source\_Code, Description, and End Date. The data is as follows:

Source_Id	Source_Code	Description	End Date
1	S_Plastics	Plastic_Non Disposable	
2	S_Plastics2	Plastic_resuable	

The 'Source\_Id' column has checkboxes next to the values '1' and '2'. A 'Copy' button is visible next to the first row. The bottom of the interface shows pagination: '1 - 2 of 2'.

## Create Code Set

Perform the following steps to create code set:

- 1 Navigate to **Manage Lookup Data UI** from the left navigation pane (**Reference Data->Manage Reference Data**).
- 2 On the **Manage Lookup Data** page, select **Create** from the **Actions** dropdown.



- 3 On the **Create Code Set** page, enter the required details as in [Figure 125](#) and click **Save**.

Figure 125: Add Code Set

The screenshot shows a web form titled "Create Code Set" with a help icon in the top right corner. The form contains several input fields: a "Name" field with the value "Materials", a "Description" field with the value "Materials", a "Template table" dropdown menu with "Multitable" selected, a "Source map table" dropdown menu with "MST\_STANDARD\_00..." selected, and a "Parent Code Set" dropdown menu with "Select..." selected. There is also an unchecked checkbox labeled "Advanced Settings". At the bottom right of the form, there are two buttons: "BACK" and "SAVE".

The code set is added and the Associate Table icon is displayed in the Associate Table column as in [Figure 126](#).

Figure 126: Manage Lookup Data

Manage Lookup Data

Groups Select... IMPORT ACTIONS

Code Set added successfully.

Search Results: Page 1 of 2

Code Set Id	Name	Description	Associated Tables	Enterprise Role
1103	Materials	Materials	Associated Tables	0
1102	Gender	Gender	Associated Tables	0
1101	Product_Type	Product_Type	Associated Tables	1
1	Source Systems	Source Systems		1

## Create Source Table Association

To associate source table with code set, perform the following steps:

- 1 On the **Manage Lookup Data** UI, click on the Associate Table icon of the corresponding code set as in [Figure 126](#).
- 2 On the **Associated Tables for Code Set** UI, enter the new table association details and click **Add Association** as in [Figure 127](#).

Figure 127: Associated Tables for Code Set

Associated Tables For Code Set
ADD ASSOCIATION
?

CodeSetDetails

Add New Table Association

Table Name \*
MST\_SOURCE\_0001\_MAP

Table Source \*
Source 1

Source System Id \*
1

Source Description
External Source1

Association Description
Plasrics

Existing Table Association

Search Results: Page 1 of 1

Table Id ^	Table Name ☞	Table Type ☞	Table Source ☞	Source Desc
7	MST_STANDARD_0004_MAP	Map	RDM	Interna ⋮
8	MST_REFERENCE_0004_CODE	Value	RDM	Interna ⋮

1 - 3 of 3
|<
<
>
>|



By Default <Table names separated by comma> tables will be displayed as Associated tables. The Source Description will be marked as Internal Source.

Message: “Association added successfully” is displayed. Similarly create the required associations.

**To edit association:**

- 1 On the **Associated Tables for Code Set** UI, click on the Menu icon corresponding to the required table association and click **Edit**.

The **Associated Tables for Code Set** UI, refreshes to display the **Edit Association** pane as in [Figure 128](#).

- 2 On the **Edit Association** pane, modify the required values and click **Save**.

Message: “Association edited successfully” is displayed.

Figure 128: Associated Tables for Code Set

Associated Tables For Code Set
SAVE
?

CodeSetDetails

Add New Table Association

Existing Table Association

Search Results: Page 1 of 1

Table Id ^	Table Name ☺	Table Type ☺	Table Source ☺	Source Description
x	x	x	x	x
7	MST_STANDARD_0004_MAP	Map	RDM	Internal Sc
8	MST_REFERENCE_0004_CODE	Value	RDM	Internal Sc
9	MST_REFERENCE_0004_DESC	Description	RDM	Internal Sc
10	MST_SOURCE_0001_MAP	Source	Source 1	External S

Delete Association
Edit

Edit Association

Table Name

MST\_SOURCE\_0001\_MA

Table Source \*

Source 1

Source Description

External Source1

Association Description

Plasrics

#### To delete Association:

- 1 On the **Associated Tables for Code Set** UI, click on the Menu icon corresponding to the required table association and click **Delete Association**.
- 2 On the confirmation web page dialog, click **Yes**.

Message: “Association deleted successfully” is displayed.

## Create Code Values

Navigate to **Manage Code Values** UI of the corresponding code set and create the required code values. The added code values will be displayed in the Enterprise Code Values pane as in [Figure 129](#). For details on creating code values, see [Section : “Add Code Values.”](#)

Figure 129: Manage Code Values

Enterprise Code	Enterprise Code Value	Enterprise Description	Parent Enterprise Code Value	Start Date	End Date	Is Default Value
50102	Plastics	Plastics		1/23/2021	12/31/9999	false

## Map Code Values to Source

To add mapping:

- 1 On the **Manage Code Values** UI, in the **Enterprise Data** pane and click **Manage Mappings** and on the **Manage Mappings** UI, click **Add Mapping**.

Figure 130: Add Mapping

**Add Mapping**

Code Set Details

Mapping

Enterprise Code Value  
Plastics:Plastics

Standard Map Details

Source Code:  
S\_Plastics

Description:  
Plastics Non Disposable

☒ Advanced Settings

Start Date:  
01/23/2021

End Date:  
12/31/9999

Source System Id:  
1:EDW Source System

Source Code Value

BACK SAVE

- 2 On the **Add Mapping** UI (Figure 130),
  - In the **Mapping** pane, select the enterprise code value from **Enterprise Code Value** dropdown.
  - In the **Standard Map Details** pane, for **Source Code**, click on corresponding lookup icon and on the **Source Data Lookup** UI, select the source table from **Source Table** drop down, select the source data and click **Select** as in Figure 131.

Figure 131: Source Data Lookup

Source Table

MST\_SOURCE\_0001\_MAF

Source Data: Page 1 of 1

Source_Id	Source_Code	Description	Start Date	End Date	Target Id	Authorization	Source	Created
	X	X						
1	S_Plastics	Plastic_Non Disposable					UI	
2	S_Plastics2	Plastic_resuable					UI	

→

1 - 2 of 2

<

>

- In the **Standard Map Details** pane, enter the source code description and source code value, select the start and end date, source system Id and click **Save**.

Message: “Mapping added successfully” is displayed. The **Source\_Map** pane on the Manage Code Values page displays the added mappings as in [Figure 132](#).

Figure 132: Manage Code Values

Manage Code Values (Code Set : Materials) BACK IMPORT CREATE ?

Search Reference Data

Enterprise Data

Search Results: Page 1 of 1

Enterprise Code	Enterprise Code Value	Enterprise Description	Parent Enterprise Code Value	Start Date	End Date	Is Default Value	Cre
50102	Plastics	Plastics		1/23/2021	12/31/9999	false	

1 - 1 of 1

Source\_0001\_Map

Search Results: Page 1 of 1

Source_Id	Source_Code	Description	End Date	Start Date
1	S_Plastics	Plastic_Non Disposable		

1 - 1 of 1

## Manage Version

Version of one or more code sets along with its code values, mappings, source records and language descriptions can be created. The created version can be used for data reference at any point of time in future.

Navigate to **Manage Versions** UI by selecting **Versions** from **Actions** drop-down on **Manage Lookup Data** UI.



Figure 133: Manage Version

Manage Versions

BACK COMPARE VERSION CREATE VERSION

Code Set Version Details: No Version Found

Version Id	Version Name	Description	Start Date

1 - 0 of 0

On the **Manage Versions** UI, you can perform the following tasks:

- [Create Code Set Version](#)
- [View Code Set Version](#)
- [Compare Code Set Versions](#)

## Create Code Set Version

Versions of code set can be created using create version option. On creation of a version, the entire code set along with code values and mappings are versioned against a version name.

To create code set version:

- 1 On the **Manage Versions** UI ([Figure 133](#)), click **Create Version**.  
The **Create Version** UI ([Figure 134](#)), is displayed.

Figure 134: Create Version

**Create Version** ?

Version Name\*  
Version\_V1

Description\*  
Version\_V1

Effective Start Date(MM/DD/...)\*  
1/27/2021

Effective End Date(MM/DD/...)\*  
12/31/9999

**Select Code Sets**

Available Code Set		Selected Code Set
Country	>	MDM Modules
Entity Type Code		Materials
Gender	<	Product_Type
Source Systems		
State		

BACK SAVE

- 2 On the Create Version UI (Figure 134), enter the following:
  - **Version Name:** enter a name for the code set version being created.
  - **Description:** enter the description for the version being created.
  - **Start Date and End Date:** the start date and end date specifies the start and end date for the version being created.

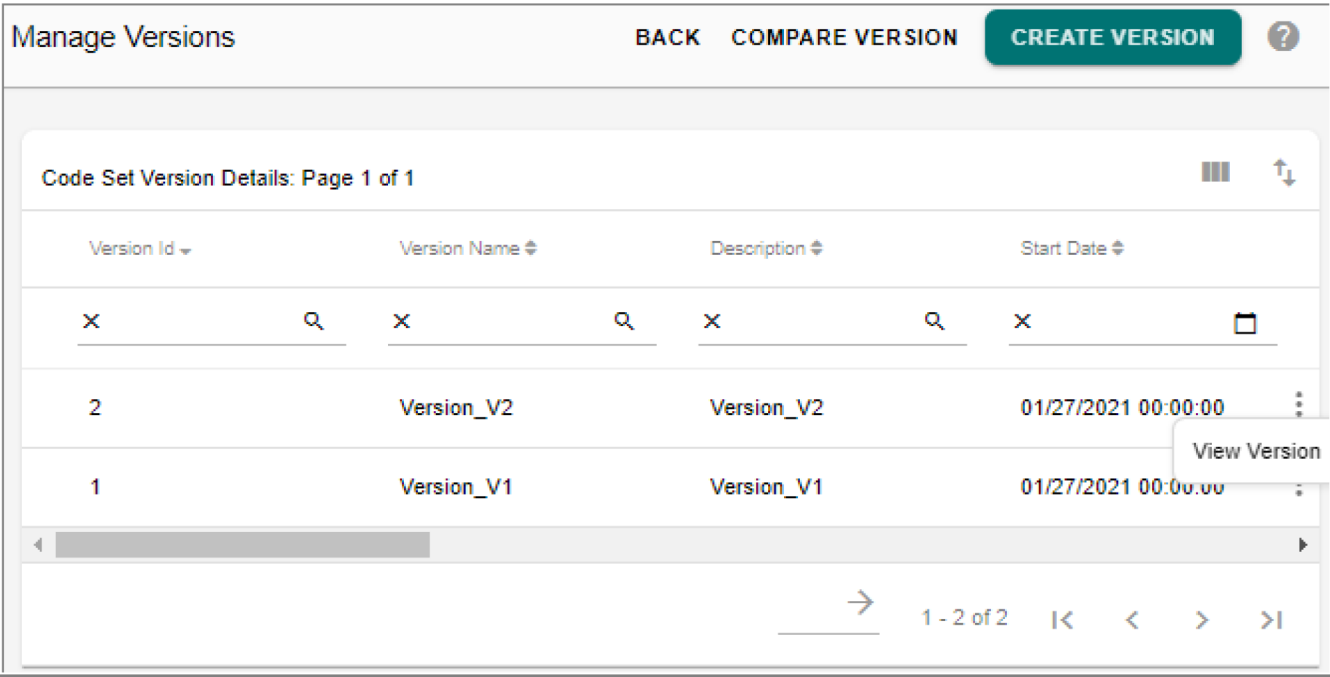
In the **Select Code Sets** pane, the **Available Code Sets** list displays all the available code sets.

- From the **Available Code Sets** list, select the code sets and move it to the **Selected Code Set** list to be included under the version being created.
- 3 Click **Save**.
- Message: “Version created successfully” is displayed.

View Code Set Version

- To view code set version:
- 1 On the **Manage Versions** UI, click on the Menu icon corresponding to the required version and click **View Version** as in [Figure 135](#).

Figure 135: Manage Version



The **Manage Lookup Data** UI displays the list of code sets versioned under the selected version name as in [Figure 136](#).

Figure 136: Manage Lookup Data

Code Set Id	Name	Description	Associated Tables
250	MDM Modules	List Of MDM Mod...	
220	Entity Type Code	The defined typ...	

- 2 On the **Manage Lookup Data** UI (Figure 136), click on the Menu icon corresponding to any code set and click **Code Values** to navigate to the Manage Code Values UI.

## Compare Code Set Versions

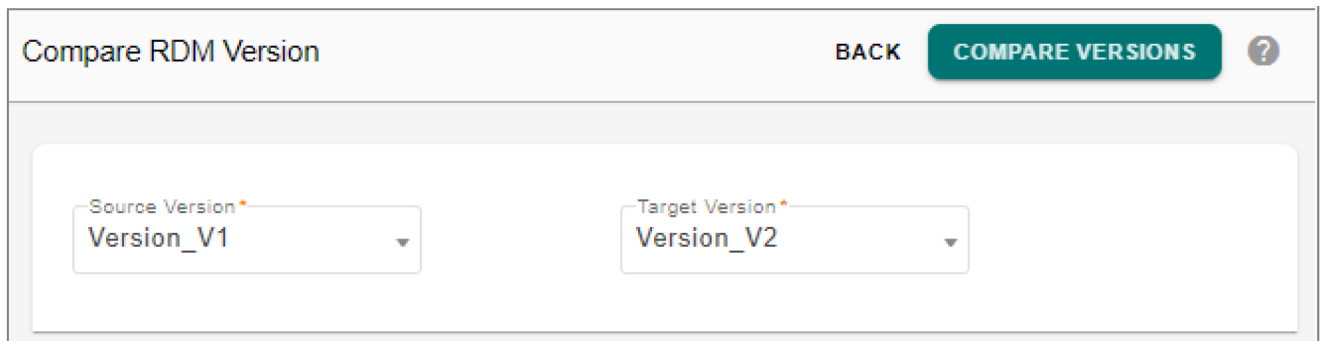
The code set compare version feature enables you to compare two different versions of code sets to identify the following:

- Code sets and its corresponding Values/Mappings that were added/removed/updated between the two versions.
- Specific attribute changes in code values and mappings.

To compare code set versions:

- 1 On the **Manage Versions** UI (Figure 135), click **Compare Version**.
- 2 On the **Compare RDM Version** UI, select the source and target version to compare and click **Compare Versions** as in Figure 137.

Figure 137: Compare RDM Version



The **Compare RDM Version** UI (Figure 138) refreshes to displays the following details:

- The **Source Code Set Details** and **Target Code Set Details** sections display the source code set details and target code set details respectively including the source map and Multi table code set details.
- The **Code Value and Mapping Comparison Summary** section displays the changes in code value and mappings. The **Action Type** column displays the type of operation performed. Click on the hyperlink on the Action Type value to view the details of attribute level changes of code values and mappings as in Figure 139.



If an unmapped source record is updated in any of the code set version, the **Code Value and Mapping Comparison Summary** UI section displays two records: one with Action type as “DELETED” and other one with Action type as “ADDED”.

Figure 138: Compare RDM Version

Compare RDM Version

BACK

COMPARE VERSIONS

Source Version \*  
Version\_V1

Target Version \*  
Version\_V2

Source Code Set Details

Target Code Set Details

Code Sets	Multi Table Code Set	Source Map Tables	Code Sets	Multi Table Code Set
MDM Modules	No		Entity Type Code	No
Product_Type	Yes		MDM Modules	No
Materials	Yes	MST_SOURCE_0001_MAP		

Code Value and Mapping Comparison Summary: Page 1 of 1

Code Set	Code Value	Source Code	Action Type	Comparison Type
x	x	x	x	x
Materials	Plastics		DELETED	Values
Materials	Plastics	S_Plastics	DELETED	Mapping
Product_Type	Office_Products	S_Plastic2	DELETED	Mapping
Product_Type	Office_Products	S_Plastic1	DELETED	Mapping

1 - 6 of 6

- The **Summary of Attribute Changes** UI displays the new values that were added to the code set for action type “ADDED”. For “DELETED” action type, the **Summary of Attribute Changes** UI displays the old values of the code set that were deleted. For “UPDATED” action type, the **Summary of Attribute Changes** UI displays the both the old value and new value of the code set.

Figure 139: Summary of Attribute Changes

Summary Of Attribute Changes

Code Set  
Materials

Code Value  
Plastics

Summary of Attribute changes

	EDW_CODE	Description	Start Date	End Date	CTL_ID
Old Value	50102	Plastics	01/23/2021 00:00:00	12/31/9999 00:00:00	1

- On the **Compare RDM Version** UI (Figure 138), click on the Export to Excel icon to export the comparison details to an excel file. The details will be displayed under two tabs as **Version Comparison Details** and **Code Value and Mapping Summary** as in Figure 140.

Figure 140: Excel File of Compare Code Set Versions

Source Version			Source Version			Target Version		
Version_V1			Version_V1			Version_V2		
Source Code Set Details								
Code Sets Multi Tabl Source Map Tables								
MDM Moc No			Code Value and Mapping Comparison summary					
Product_T Yes			Code Set			Code Valu Source Co Action Ty Comparisi Comparison Table		
Materials Yes			MST_SOURCE_0001_MAP	Product_T Office_PrcS_Plastic2	DELETED	Mapping	VER_STANDARD_0001_MAP	
				Product_T Office_PrcS_Plastic1	DELETED	Mapping	VER_STANDARD_0001_MAP	
				Product_T Office_Products	DELETED	Values	VER_REFERENCE_0001_CODE	
Target Version				Product_T Enterprise_Products	DELETED	Values	VER_REFERENCE_0001_CODE	
Version_V2				Materials Plastics	DELETED	Values	VER_REFERENCE_0004_CODE	
Target Code Set Details				Materials Plastics	S_Plastics	DELETED	Mapping	VER_STANDARD_0004_MAP
Code Sets Multi Tabl Source Map Tables								
Entity Typ No								
MDM Moc No								
Version Comparison Details			Version Comparison Details			Code Value and Mapping summary		

## Manage Authorization

The Manage Authorization option on Manage Lookup Data UI allows you to assign access to individual code set or individual group.

The Manage Authorization feature is available on the LRDM UI only for a user assigned to a role having access to manageCodeSetAuthorization activity. By default, user: Admin will have manageCodeSetAuthorization activity assigned to ADMIN role. For authorization of

code sets to work as per the roles assigned, authorization has to be enabled for Code Sets. When authorization is enabled on code set, the assigned role is populated in SYS\_Auth\_Id of the MST code set table and the same role will be populated in all the linked tables like MST\_Code Values, MST\_Reference Code, MST\_Stanadard Map tables and source tables etc. Any changes in the MST code set for authorization will be reflected in all the linked tables.



- All MDM users will have access to code sets grouped under the group “MDM Code Sets” as these code sets are used in many internal workflows of MDM. A default role is assigned to all user groups of MDM for this purpose. Once the authorization is enabled for code sets, this default role is assigned to the code sets of “MDM Code Sets” group. That is all the out of the box code sets will have default role assigned.
  - Newly added non out of the box code set must be assigned to any role for RLS feature to work as expected.
  - If authorization for code set is already enabled and if the user tries to add any code set, the newly added code set will not be displayed on the Manage Lookup Data UI until the newly added code set is assigned to a role on the Manage Authorization UI.
- 

## Manage Authorization UI

Navigate to **Manage Authorization** UI by selecting **Authorization** option from **Actions** dropdown on **Manage Lookup Data** UI.

The [Figure 141](#) displays the Manage Authorization UI.



Figure 141: Manage Authorization

Manage Authorization

BACK SAVE ?

☐ Enable Authorization for Code Sets

☐ Code Set

☒ Group

Groups  
Select...

Code Set: Page 1 of 2

☒ Assign Role ☐ Remove Role

Role Details

Available Roles

Assigned Roles

Admin  
Default Role  
RDM\_ADMIN  
RDM\_USER  
Super User  
TAS User  
Toolkit User

>  
<

On the Manage Authorization UI:

- **Enable Authorization for Code Sets:** Select this checkbox to enable authorization for code set table. Once Authorization is enabled for code sets and if you disable it by deselecting the checkbox, it will remove all the authorization data of the code set table. (that is SYS\_AUTH\_ID for all code sets will be set to NULL).
- **Code Set:** select this option to assign or remove role for individual code set. On selection of Code Set option, all the available code sets will be displayed in the Code set pane.

- **Group:** select this option and select the group from Groups dropdown to assign or remove role for individual group. On selection of Group option, all the member of selected group will be displayed.

### **Manage Authorization on Individual Code Set**

Perform the following steps to assign roles on individual code set:

- 1 On the **Manage Authorization** UI, select the option **Code Set**.  
All the available code sets will be displayed in the **Code Set** pane.
- 2 In the **Code Set** pane, select the required code sets and select the **Assign Role** option.
- 3 In the **Role Details** pane, select one or more role(s) from **Available Role** list and move it to **Assigned Role** list as in [Figure 142](#).

Figure 142: Manage Authorization

Manage Authorization

BACK

SAVE

?

☐ Enable Authorization for Code Sets
 

☒ Code Set
 ☐ Group

Code Set: Page 1 of 2

Authorization	Code Set Id	Name	Description
<input checked="" type="checkbox"/>	1103	Materials	Mat
<input type="checkbox"/>	1102	Gender	Ger
<input checked="" type="checkbox"/>	1101	Product_Type	Pro

1 - 20 of 23

<<

<

>

>>

☒ Assign Role
 ☐ Remove Role

Role Details

Available Roles

Default Role  
 RDM\_USER  
 Super User  
 TAS User  
 Toolkit User

>

<

Assigned Roles

Admin  
 RDM\_ADMIN

4 Click **Save**.

The selected roles will be assigned to the selected code set(s).

Perform the following steps to remove the assigned roles on individual code set:

- On the **Manage Authorization** UI, select the option **Code Set**.  
All the available code sets will be displayed in the **Code Set** pane.
- In the **Code Set** pane, select the required code sets and select the **Remove Role** option.

- 3 In the **Role Details** pane, select one or more role(s) from **Assigned Role** list and move it to **Available Role** list.
- 4 Click **Save**.

The selected roles will be removed on the selected code set(s).

### **Manage Authorization on Individual Group**

Perform the following steps to assign roles on individual group:

- 1 On the **Manage Authorization** UI, select the option **Group** and select the required group from the **Groups** dropdown.  
All the available code sets of the selected group is displayed.
- 2 Select **Assign Role** option.
- 3 In the **Role Details** pane, select one or more role(s) from **Available Role** list and move it to **Assigned Role** list.
- 4 Click **Save**.

The selected roles will be assigned to all the code set(s) of the group.

Perform the following steps to remove the assigned roles on individual group:

- 1 On the **Manage Authorization** UI, select the option **Group** and select the required group from the **Groups** dropdown.  
All the available code sets of the selected group is displayed.
- 2 Select **Remove Role** option.
- 3 In the **Role Details** pane, select one or more role(s) from **Assigned Role** list and move it to **Available Role** list.
- 4 Click **Save**.

The selected roles will be removed on all the code set(s) of the group.

# CHAPTER 5 Hierarchy Management

## What's In This Chapter

This chapter provides detailed description on hierarchy management.

Topics include:

- [Data Model](#)
- [Sample Hierarchy Structure](#)
- [Create Hierarchy Tables Through Model Builder](#)
- [Hierarchy Management Functionalities](#)
- [Hierarchy Manager](#)

## Data Model

The data model uses generic structures that can house data for any hierarchy. The tables involved are HierarchyNode and HierarchyNodeRelationship. The HierarchyNode table stores the data for one (or more) levels of the hierarchy and the HierarchyNodeRelationship table stores the parent child relationship between the HierarchyNode tables.

Table 2: Hierarchy Node

Table Name	Attribute Name	Attribute Data Type	Other Details
HierarchyNode	Node_Id	String	Unique Identifier
	Node_Name	String	
	Node_Desc	String	
	Node_Ext_Id	String	

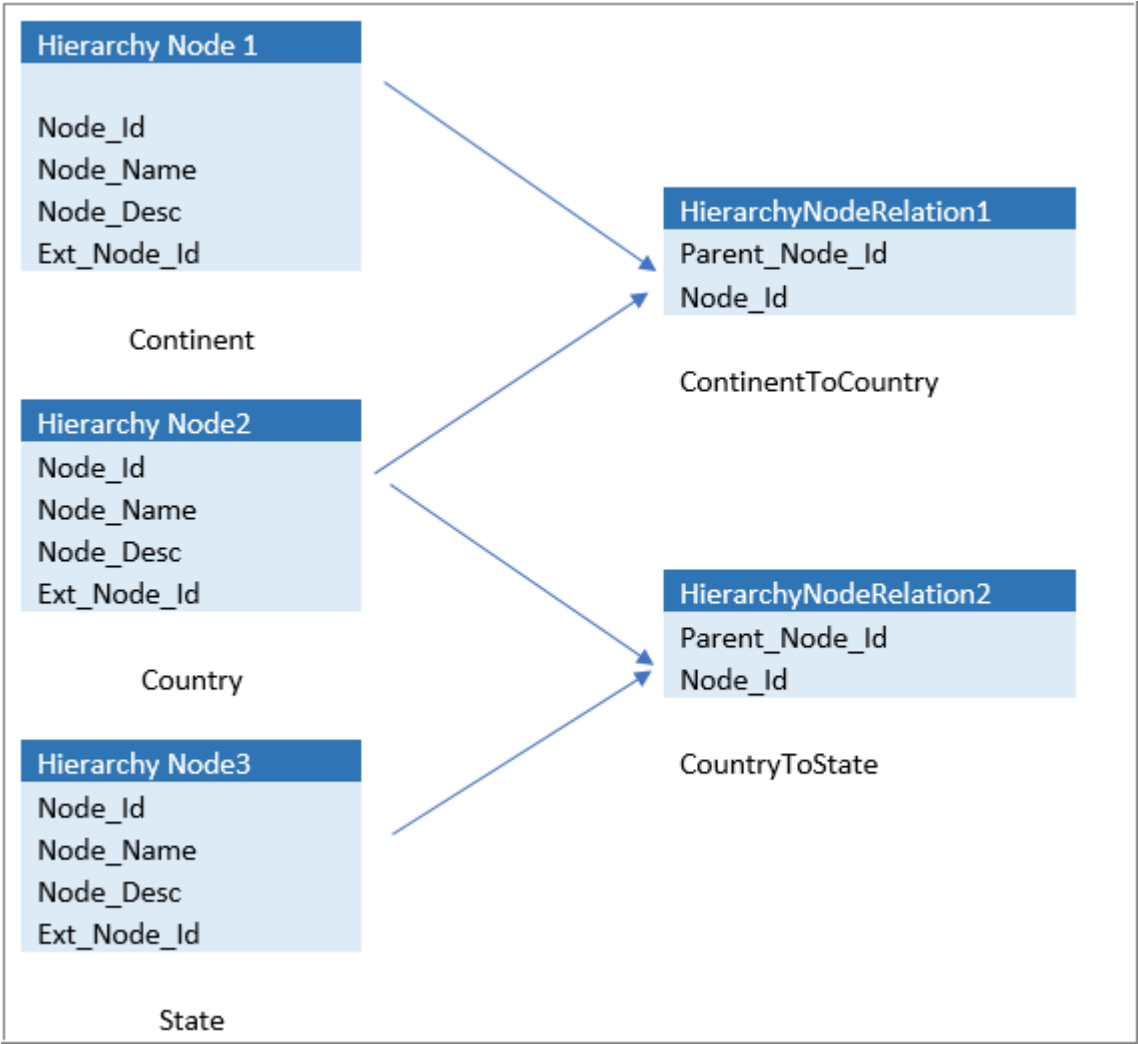
Table 3: Hierarchy Relation

Table Name	Attribute Name	Attribute Data Type	Other Details
HierarchyRelation	Parent_Node_Id	String	Represents the Node ID of the Parent Hierarchy Node
	Node_Id	String	Represents the Node ID of the Child Hierarchy Node

### Sample Hierarchy Structure

In this chapter, three level Geography Hierarchy is used for all the demonstration of hierarchy features. In the example hierarchy, three level includes the following: Continent, Country and State.

Figure 143: Hierarchy Nodes and Node Relations for Three Level Hierarchy



### Create Hierarchy Tables Through Model Builder

Hierarchy tables for nodes and relationships need to be created before creating hierarchy. Model Builder is enhanced to generate these tables from a pre-defined template. Any number of HierarchyNode or HierarchyRelationship tables can be created during runtime through Model Builder—Manage Template UI.

## Generate and Validate Hierarchy Tables

- 1 Navigate to **Model Builder** UI from left navigation pane: **Model Setup -> Model Builder** and select **Template** option from **Actions** dropdown.
- 2 On the **Manage Template** UI, select the **Hierarchy Template** option, enter the number of hierarchy tables to be generated and select the table type (HierarchyNode and/or HierarchyNodeRltn) and click **Generate Models** to initiate the ISG or schema generation.

Figure 144: Manage Template

The screenshot shows the 'Manage Template' interface. At the top, there's a header bar with 'Manage Template' on the left and '2 Value(S) Selected', 'VALIDATE TEMPLATE', and 'GENERATE MODEL' on the right. Below the header, there's a 'Template Type' dropdown menu set to 'Hierarchy Template'. Underneath, a text input field is labeled 'Enter number of tables to be generated.' with the value '4' entered. Below this is a 'Search Templates: Page 1 of 1' section. A table lists the templates:

<input checked="" type="checkbox"/>	Template Name	Folder Name	Generated Table Count	Modified By	Modified Date
<input checked="" type="checkbox"/>	HierarchyNode	Hierarchy Nodes	0		
<input checked="" type="checkbox"/>	HierarchyNodeRltn	Hierarchy Relations	0		

At the bottom right, there's a pagination control showing '1 - 2 of 2' and navigation arrows.

- 3 On the **Model Builder** UI, select **Logs** from **Actions** dropdown to monitor the Schema Generation for the Hierarchy Models.

Figure 145: Model Deployment Logs

Model Deployment Logs

BACK REFRESH ?

Search :Model Builder Logs: Page 1 of 1

Execution Id #	Deployment Status	Source Name #	Generated Tables	Start Time #
X	X	X		X
11	ISG IN PROGRESS	RDM	<div>HierarchyNode_0001   HierarchyNode_0002   HierarchyNode_0003   HierarchyNode_0004   Hierarchy_0001_Node   Hierarchy_0002_Node  </div>	03/25/2021 15:01:4 Log Details

1 - 10 of 10 < > >>

- 4 On the **Model Deployment Logs** (Figure 145), click on the Menu icon corresponding to the required execution Id of the log and click **Log Details** to view the complete deployment log information as in Figure 146.



Figure 146: Model Deployment Log Details

Model Deployment Log Details				BACK	DOWNLOAD LOGS	?
LOG STATUS				PRE/POST ISG COUNT		
Execution Id						
11						
Executed Requests				Executed SQLs		
ISG Log Details				SQL Statements		
Parameter Name	Status	Operation Start Time	Operatic	Execution Time	SQLs	
Final Status	SUCCESSFUL	03/25/2021 08:01:45	03/25/2			
customPreIsqReq	SUCCESSFUL	03/25/2021 08:01:49	03/25/2	03/25/2021 08:01:52	CREATE TABLE RDMUSRTST_MST.MST_HIER_NOI	
Schema Generation	SUCCESSFUL	03/25/2021 08:01:49	03/25/2	03/25/2021 08:01:52	REPLACE VIEW MST_HIER_NODE_0001_RLTN AS	
Create Activity	SUCCESSFUL	03/25/2021 08:02:49	03/25/2	03/25/2021 08:01:52	UPDATE EA_EXT_ATTRBS SET IS_USED=0...	
Populate SysTableMap	SUCCESSFUL	03/25/2021 08:02:49	03/25/2	03/25/2021 08:01:53	UPDATE EA_EXT_ATTRBS SET IS_USED=0...	
Populate FacetInfo	SUCCESSFUL	03/25/2021 08:02:50	03/25/2	03/25/2021 08:01:53	UPDATE EA_EXT_ATTRBS SET IS_USED=0...	
Populate Column Map	SUCCESSFUL	03/25/2021 08:02:51	03/25/2	03/25/2021 08:01:53	UPDATE EA_EXT_ATTRBS SET IS_USED=0...	

Once the ISG is completed successfully as in [Figure 147](#), the **Generated Table Count** column on the **Manage Template** UI displays the number of tables generated and the models will appear in the Generated Models List on the **Model Builder** UI as in [Figure 148](#) and will be available for use.

An email notification displaying model builder ISG status and the list of names of tables generated through model builder ISG is sent to the admin.

Figure 147: Model Deployment Logs

The screenshot shows the 'Model Deployment Logs' interface. At the top, there are 'BACK', 'REFRESH', and a help icon. Below is a search bar with the text 'Search :Model Builder Logs: Page 1 of 1'. The main table has columns: Execution Id #, Deployment Status, Source Name #, Generated Tables, and Start Time #. Each column has a search icon. The table contains one row with Execution Id 11, Status 'SUCCESSFUL : 2 Minute(s) 7 Second(s)', Source Name 'RDM', and Start Time '03/25/2021 15:'. A dropdown menu is open for the 'Generated Tables' column, showing a list of tables: HierarchyNode\_0001\_..., HierarchyNode\_0002\_..., HierarchyNode\_0003\_..., HierarchyNode\_0004\_..., Hierarchy\_0001\_Node..., and Hierarchy\_0002\_Node... The bottom of the interface shows a pagination bar with '1 - 10 of 10' and navigation arrows.

Execution Id #	Deployment Status	Source Name #	Generated Tables	Start Time #
11	SUCCESSFUL : 2 Minute(s) 7 Second(s)	RDM	HierarchyNode_0001_ HierarchyNode_0002_ HierarchyNode_0003_ HierarchyNode_0004_ Hierarchy_0001_Node Hierarchy_0002_Node	03/25/2021 15:

Figure 148: Model Builder—Generated Models

The screenshot shows the 'Model Builder' interface. At the top, there are 'ACTIONS' and 'CREATE' buttons, along with a help icon. Below are two dropdown menus: 'Status' set to 'Generated' and 'Groups' set to 'Select...'. A search bar shows 'Search Models: Page 1 of 4'. The main table has columns: Object Name #, Object Type #, Description #, Logical Source Name #, and Modified By #. Each column has a search icon. The table lists five generated models, all of type 'Table' and located in 'Default\_DB'. The models are: HierarchyNode\_0004\_Rltn, HierarchyNode\_0002\_Rltn, Hierarchy\_0002\_Node, Hierarchy\_0003\_Node, and Hierarchy\_0004\_Node. The bottom of the interface shows a pagination bar with '1 - 20 of 63' and navigation arrows.

Object Name #	Object Type #	Description #	Logical Source Name #	Modified By #
HierarchyNode_0004_Rltn	Table	HierarchyNodeRltn	Default_DB	
HierarchyNode_0002_Rltn	Table	HierarchyNodeRltn	Default_DB	
Hierarchy_0002_Node	Table	HierarchyNode	Default_DB	
Hierarchy_0003_Node	Table	HierarchyNode	Default_DB	
Hierarchy_0004_Node	Table	HierarchyNode	Default_DB	

The Configurable UIs for all the tables are also auto-generated during ISG and can be viewed from left navigation pane (**Tables -> Master Tables**) as in [Figure 149](#).

Figure 149: Configurable UI

teradata	
Master tables	Filter
<ul style="list-style-type: none"> <li>AbcModels</li> <li>Account</li> <li>CDI</li> <li>Cleansing</li> <li>Group</li> <li><b>Hierarchy Nodes</b></li> <li>Hierarchy Relations</li> <li>LRDM</li> <li>ManualMatch</li> <li>MDRM</li> <li>SyncUp</li> </ul>	<div> <div>Name</div> <div>Hierarchy_0001_Node</div> <div>Hierarchy_0002_Node</div> <div>Hierarchy_0003_Node</div> <div>Hierarchy_0004_Node</div> </div>

## Hierarchy Management Functionalities

On the RDM left navigation pane, use **Reference Data-> Hierarchy Management** links to perform various activities related to hierarchy.

Below are the key activities involved while creating and managing hierarchies.

Table 4: Activities Involved in Create and Manage Hierarchies

Activity	Sub-Activity	Purpose	Link
Create New Hierarchy	Create Hierarchy Objects	Tables/Objects that participate in the hierarchy.	-Manage Hierarchy Objects link -Hierarchy Manager (upon creating a hierarchy)
	Manage Extended Attributes	Adds up an additional column that may reference or lookup to another table to provide more details in the table.	-Manage Hierarchy Objects link > Manage Extended Attributes button
	Create Relationships	Holds the parent/child relationship of the selected tables/objects.	-Manage Hierarchy Relationships link -Hierarchy Manager (upon creating a hierarchy)
Create/Manage Rollup Attributes	Create Rollup Attributes	Determines the type of Rollup function (aggregates) that will be used in the hierarchy.	-Manage Hierarchy Rollup Attributes link
	Assign Rollup Attributes	Incorporates the rollup attribute to the selected hierarchy. A view will be generated automatically containing the aggregated values of the objects participating in the hierarchy.	- Hierarchy Manager link > Manage Rollup Attributes button
Create Approval Object	Create Approval Object	Allows the system to determine any changes done in the selected object (for version submissions) which is sent to the approver who accepts or rejects the changes made.	-Data Harmonization > Approvals > Manage Approvals
Create Versions of Hierarchy	Create Versions of Hierarchy	Stores yearly versions that will be analyzed by a business user.	-Hierarchy Manager > Create Version button
View Versions of Hierarchy	View Version	To review a selected version before submitting for approval.	-Hierarchy Manager > View Version button
	Submit For Approval	Sends the selected version to the approver	-Hierarchy Manager > View Version button
Approval Inbox	Compare with Base Hierarchy	Compare the submitted version to the base hierarchy.	-Settings icon (Upper right side) > Approval Inbox
	Assign To Me	Assigns the selected hierarchy to the user for approval.	-Settings icon (Upper right side) > Approval Inbox
	Approve/Reject Hierarchy	Approve/Reject the hierarchy.	-Settings icon (Upper right side) > Approval Inbox

## Manage Hierarchy Objects

Hierarchy Objects represent each level of a Hierarchy. Creating Hierarchy Object is the first step to create a Hierarchy. As soon as a Hierarchy Object is created, the next available Node table gets assigned to the Object. By default, node Id, node name, external node Id and node description attributes are available as part of any node table. Additional attributes can be defined using extended attributes.

Navigate to **Reference Data -> Hierarchy Management -> Manage Hierarchy Objects**.

The **Manage Hierarchy Objects** UI displays all the existing hierarchy objects and relationships created and their corresponding tables and details. The Manage hierarchy Objects UI displays the following fields:

Table 5: Manage Hierarchy Objects UI Fields

UI Field	Purpose
RoId	Uniquely identifies a relational object.
Hierarchy Object Name	User defined hierarchy object name.
Hierarchy Object Description	User defined hierarchy object description.
Logical Table Name	Logical name of the table that the RDM system can read.
URL	This field stores the URL link that is being used to display a configurable workflow. Some URLs contain parameters such as workflowId that is essential in displaying the desired results.
URL Param	URL Parameter is set to NODE_ID by default.

On the Manage Hierarchy Objects UI, you can create new hierarchy objects and manage existing hierarchy objects by performing the following tasks on the Manage Hierarchy Objects UI:

- [Create Hierarchy Object](#)
- [Edit Hierarchy Objects](#)
- [Manage Extended Attributes](#)
- [Configurable UI](#)
- [Show Hierarchy and Show Relationship](#)

### Create Hierarchy Object

To create new hierarchy objects, navigate to **Create Hierarchy Object** UI from **Manage Hierarchy Objects** UI using **Create** button.

Figure 150: Manage Hierarchy Objects

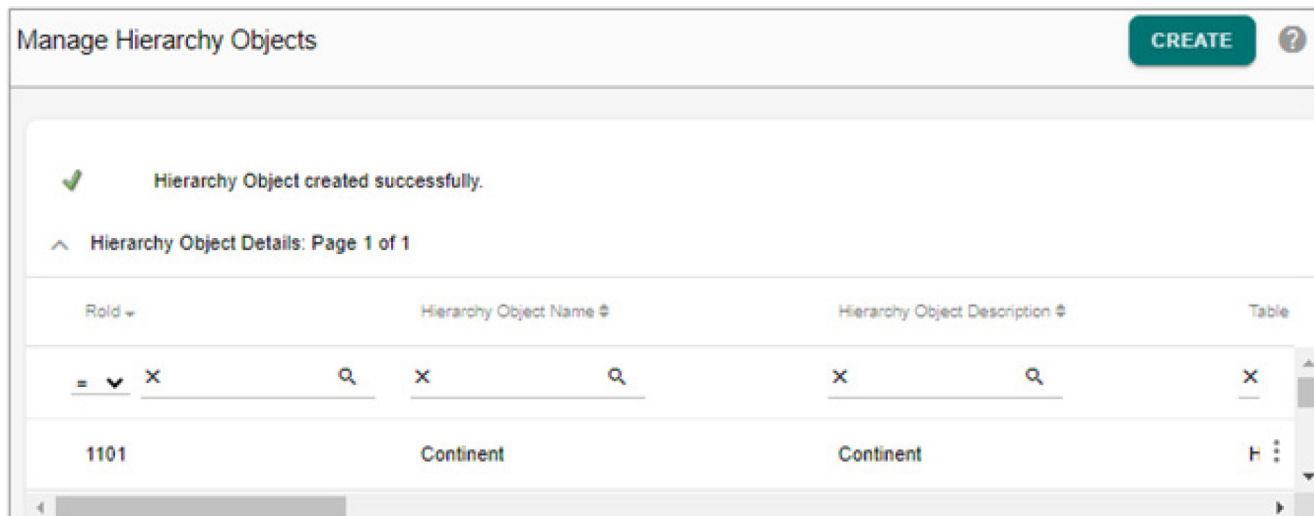
Figure 151: Create Hierarchy Objects

On **Create Hierarchy Object** page ([Figure 151](#)), enter the following and click **Save**.

- **Object Name:** unique name of the object.
- **Object Description:** brief description about the object being create.
- **Create Object on Custom Tables:** select this option as “No” to create objects on OOTB master tables.

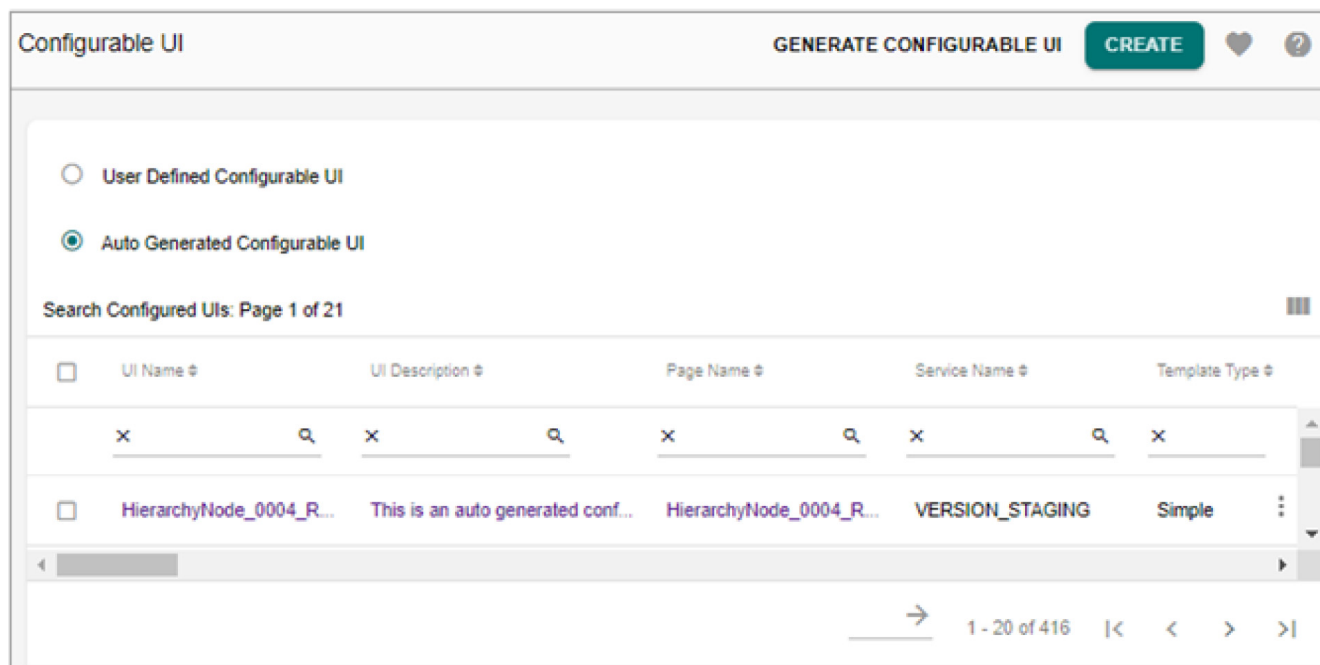
On clicking **Save**, the Hierarchy Object will be created and will be assigned to the next available Hierarchy Node Table. You can check this assignment on the Manage Hierarchy Objects UI as in [Figure 152](#).

Figure 152: Manage Hierarchy Objects



When Hierarchy Object is saved, links to base and version configurable UI will be stored along with it. The configurable UIs for both base and version will be created as part of ISG through model builder as in [Figure 153](#).

Figure 153: Configurable UI



A workflow is provided with OOB RDM to rename the Hierarchy Object and Relationship tables as per the Hierarchy Object name and Relationship name assigned respectively. Any spaces or special characters in the name will be ignored while renaming Hierarchy tables. The name of the workflow is: RenameDocumentLogicalNameWkf. This workflow can be scheduled using scheduler. This workflow internally looks for all the RDM and Hierarchy tables that are used but not renamed and renames all. In case of any failure, entire changes are reverted back.

The RenameDocumentLogicalNameWkf should be executed before creating hierarchy using the corresponding objects and relationships.

An email notification displaying the list of renamed tables will be sent to the admin.

### Load Hierarchy Object Data

Data can be loaded into the hierarchy objects either through backend or through the corresponding configurable UI. You can use the excel upload feature to load hierarchy node data. The load for Continent object is demonstrated in the following section.

First, Prepare the Excel sheet to be loaded to the Hierarchy Node (for example - for Continent table) as in [Figure 154](#). The NODE\_ID is optional, if not provided it will automatically be generated by the system. All computations will be done on Node Name. If Node Name for an uploaded data exists, it will be considered as Update.

Figure 154: Load Hierarchy Data

A	B	C	D
Node Id	Node Name	Node External Id	Description
104	North America		
100	Africa		
102	Europe		
105	South America		
103	Antarctica		
101	Asia		

Now navigate to the Continent configurable UI (on the Manage Hierarchy Objects UI, select Continent and click Manage Base Config UI) and perform the excel upload process.

Figure 155: Hierarchy Objects—Excel Upload

Similarly create other required hierarchy objects (Country and State) as in [Figure 156](#) and load the corresponding data.



Figure 156: Manage Hierarchy Objects

Manage Hierarchy Objects

CREATE ?

^ Hierarchy Object Details: Page 1 of 1

Role	Hierarchy Object Name	Hierarchy Object Description	Table
<div> <div>=</div> <div>▼</div> <div>×</div> <div>Q</div> </div>	<div> <div>×</div> <div>Q</div> </div>	<div> <div>×</div> <div>Q</div> </div>	<div> <div>×</div> </div>
1103	State	State	H ⋮
1102	Country	Country	H ⋮
1101	Continent	Continent	H ⋮

→

1 - 8 of 8

⏪

<

>

⏩

## Edit Hierarchy Objects

You can modify the hierarchy object details using the Edit option.

Navigate to **Reference Data -> Hierarchy Management -> Manage Hierarchy Objects** and click on the Menu icon corresponding to the required hierarchy object and click **Edit** to open the Edit Hierarchy Objects UI.

Figure 157: Edit Hierarchy Object

?

\* denotes required field

Object Name

Country

Object Description

Country

Logical Table Name

Hierarchy\_0002\_Node

Physical Table Name

MST\_HIER\_0002\_NODE

Master URL

/mdm/start.x2ps?  
START\_WORKFLOW=performHierarchyAction&

Version URL

/mdm/start.x2ps?  
START\_WORKFLOW=performHierarchyAction&

URL Parameter

NODE\_ID

☒ Customize icon

Select Icon

Key Columns

No. of Display Columns:  
1

No. of Sort Columns:  
1

☐ Define Filter

Key Column 1

NODE\_ID

Display Column 1:

NODE\_NAME

Sort Column 1:

NODE\_NAME

On **Edit Hierarchy Object** page ([Figure 157](#)), you can modify the following parameters:

- **Object Name:** unique name of the object. Non editable field.
- **Object Description:** brief description about the object being edited.
- **Logical Table Name:** master table logical name. Non editable field.
- **Physical Table Name:** table physical name. Non editable field.

- **Master URL:** URL configured for the hierarchy object. On click of a node in hierarchy viewer, the master URL of page set is rendered in lower bottom panel of hierarchy viewer. It is editable field.
  - **Version URL:** version URL configured for the hierarchy object. On click of a node in hierarchy viewer, the version URL of page set is rendered in lower bottom panel of hierarchy viewer. It is editable field.
  - **URL Parameter:** the URL parameter specifies the Node ID selected automatically from the available Node table.
- 



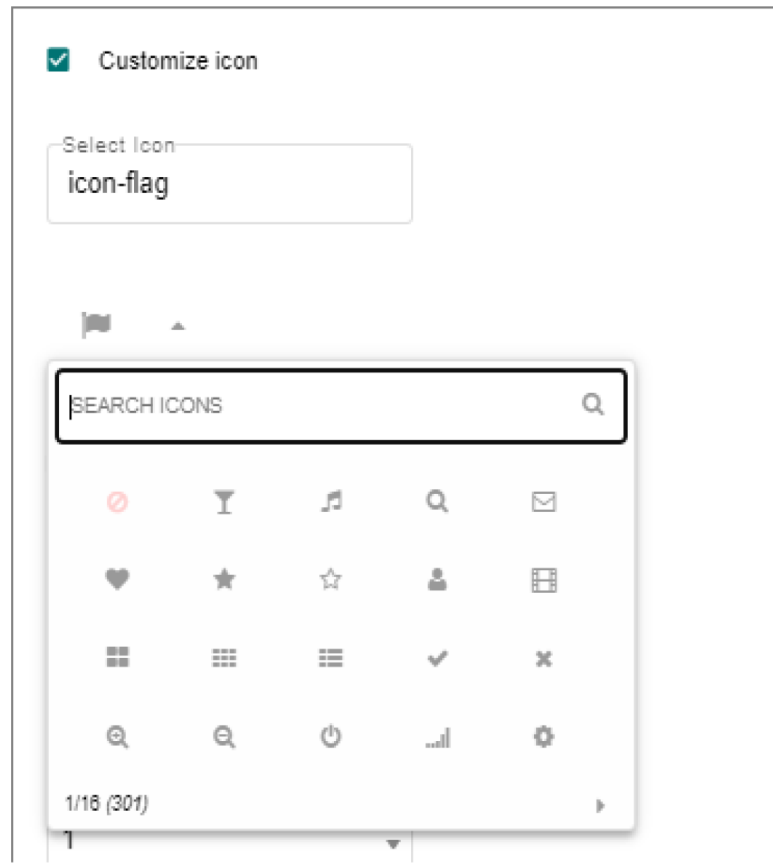
- Master URL and URL parameters are auto populated for master tables.
  - The configurable UI to be shown from the Hierarchy can be specified by adding a parameter “WF\_ID” in the Master URL and Version URL of the Hierarchy Object. The workflowId of the required configurable UI needs to be appended to the Master/Version URL in order for that configurable UI to be used across all hierarchy actions. If no workflowId has been mentioned explicitly, then the latest user-defined configurable UI will be used.
- 
- **Customize Icon:** select the Customize icon checkbox to assign different icons for different hierarchy objects. In the hierarchy tree view, each hierarchy node level displays the selected icon preceding the node name. The icon display helps to identify the level at which the data is present.  
  
When Customize icon checkbox is enabled, the **Select Icon** textbox and a dropdown is displayed as in [Figure 158](#). Click on the dropdown arrow and select the required icon to be assigned to the object. You can also search icons by name.
- 



Customize icon is not supported for relationship objects.

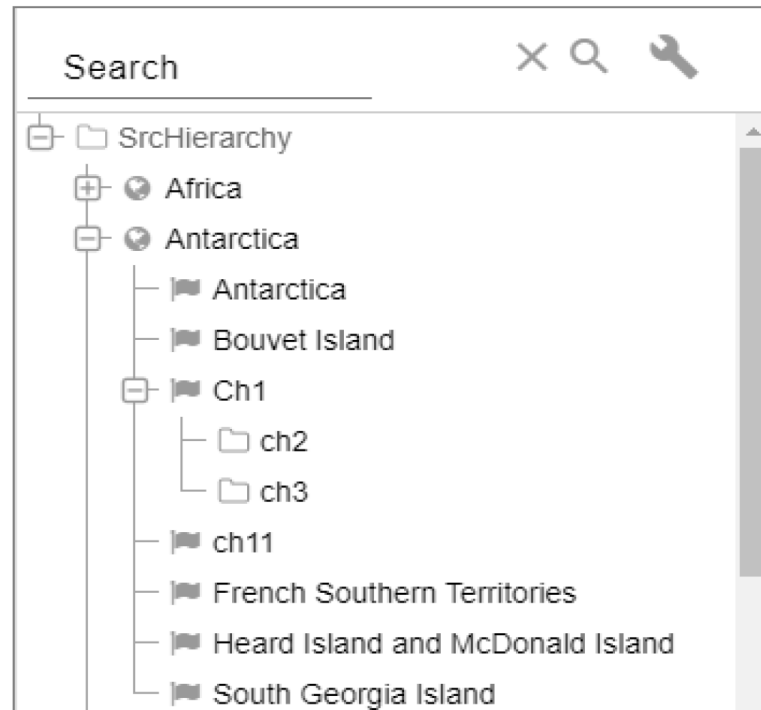
---

Figure 158: Customize Icon



The assigned icons for each object will be displayed on the hierarchy tree view as in [Figure 159](#).

Figure 159: Hierarchy Viewer



- **Key Columns:** it specifies the number of columns used for Key columns. Non editable field.
- **No. of Display Columns:** specifies the number of columns to be displayed on the hierarchy viewer.
- **Display Column:** it specifies the column that will be displayed on the hierarchy viewer.
- **No. of Sort Columns:** it specifies the number of columns used for sorting.
- **Sort Column:** it specifies the column that will be used for sorting nodes in hierarchy viewer.



- You can select maximum five number of display columns and five number of sort columns.
- It is recommended not to use special character '#' in hierarchy display column value as drag and drop may not work on the hierarchy viewer.

- **Define Filter:** You can define filters on hierarchy objects. Filters defined on hierarchy objects allow filtering data of hierarchies in hierarchy viewer. Filter can be of static or dynamic in nature. For static filters, the filter values are provided along with the filter condition. For dynamic filters, the filter values are taken from the user while invoking the hierarchy for viewing.

For detailed description on filter, refer to Master Data Management Server Guide.pdf

You can delete the existing hierarchy object using **Delete** options on Manage Hierarchy Objects UI.

### Manage Extended Attributes

The Manage Extended Attributes button on the Manage Hierarchy Objects UI allows you to reference a column of the selected object or table to another referencing table or a code set. It uses the pre-created extended attributes defined for the tables. The Manage Extended Attributes UI displays the list of extended attributes filtered on the selected table, which are stored in the SYS\_EXT\_ATTRBS\_MAP. You can create extended attributes from this page. Note that this process is assigning business usage at runtime to existing physical attributes in the database. The physical columns for these have already been created for the HierarchyNode tables (20 string, 5 decimal and 2 date types).

Figure 160: Manage Extended Attributes

All underlying node tables of Hierarchy Object contain extended attributes. Additional attributes for Hierarchy Objects can be defined through extended attributes.

Two main tables are involved in the creating extended attributes activity:

- SYS\_EXT\_ATTRBS\_MAP: contains the extended attributes mapped to the table and the details defined for the extended attributes.
- SYS\_EXT\_ATTRBS: contains the list of tables and extended attributes that are both used and unused. [Table 6](#) lists the details for SYS\_EXT\_ATTRBS\_MAP table.

Table 6: SYS\_EXT\_ATTRBS\_MAP Table

UI Field	Purpose	Comment
Table Name	The extended attribute will be assigned to this particular Logical Table Name.	In Manage Extended Attributes Page, this field is filtered out based on the selected table from the previous page.
Attribute Logical Name	Name of the extended attribute.	Required
Description	Description of the extended attribute.	Not Required, but strongly recommend using a description that identifies the extended attribute.
Valid Value Type	There are 2 types of Valid Value Type: 1.Reference - user is asked to choose a reference table and the reference column or property that will be used to for referencing the attribute name. 2.LookUp - makes use of the Code Set table that stores possible values under a specific category. The user will be asked to select the code set name that is created.	This field only shows up when user selects 'string' in data type field and ticks the 'Is Valid Value Type' checkbox.
Code Set Name	The code set that will be chosen by the user will be used for lookups. Only 1 code set can be selected.	This field only shows up when user selects the 'LookUp' value in the 'Valid Value Type' field.
Reference Table	The user is given an option to choose a table in the Master database that will be used as reference.	This field only shows up when user selects the 'Reference' value in the 'Valid Value Type' field.
Default Value	Default Value of the attribute.	Only applicable to string and decimal data types.
Is Required	Determines if the extended attribute is required.	If 'Is Required' is ticked, then value is 1 else 0.
Attribute Physical Name	Physical Name of extended attribute. String— EXT_ATTRB_1 Date— EXT_ATTRB_DATE_1 Decimal— EXT_ATTRB_DEC_1	This corresponds to the total number of string, date and decimal extended attributes defined in each table.
Attribute Data Type	Determines the data type of the extended attribute. The user has options for data type namely: String Decimal Date	Total extended attributes: 27 Breakdown: String - 20 Decimal - 5 Date - 2
Decimal Precision	Determines the decimal precision of the decimal data type.	This field only shows up when user selects 'decimal' in data type field.

[Table 7](#) lists the details for SYS\_EXT\_ATTRBS table. This table is used to mark the availability of the extended attributes.

Table 7: SYS\_EXT\_ATTRBS

UI Field	Purpose
DOCUMENT_NAME	The extended attribute will be assigned to this particular Logical Table Name.
ATTRIBUTE_NAME	Contains the list of all pre-created extended attributes.
ATTRIBUTE_DATA_TYPE	Data type of the pre-created extended attribute.
IS_USED	Determines if extended attribute is used or not. If extended attribute is used, then value = 1, else 0.

### **Create New Extended Attribute**

On the **Add Extended Attribute** UI, the table name field is already selected based on the Table name selected on the Manage Hierarchy Objects UI. You must specify the data type to be assigned to the extended attribute being created and fill in the other fields and click **Save** as in [Figure 161](#).



Figure 161: Add Extended Attribute

**Add Extended Attribute** ?

\* denotes required field

Objects  
Hierarchy Objects

Table Name \*  
Hierarchy\_0003\_Node

Data Type \*  
string

☐ Is Valid Value Type

Extended Attribute Name \*  
GeoType

☐ Is Required

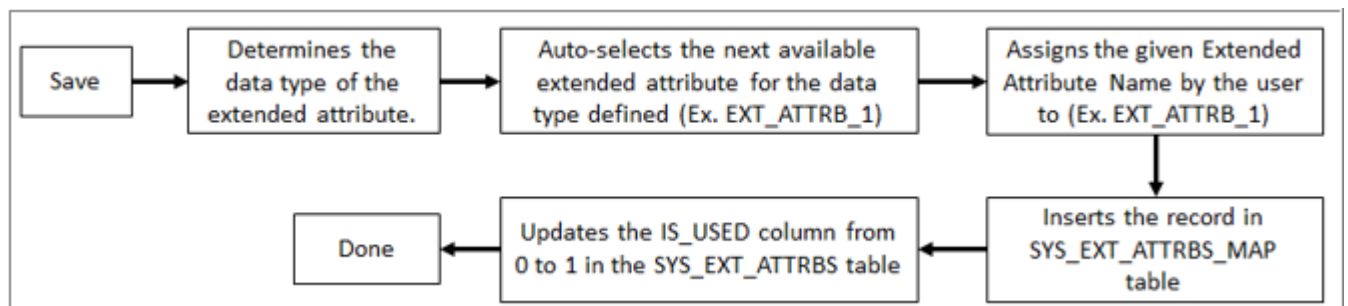
Default Value

Description

CANCEL SAVE

Figure 162 displays the process flow running in the background when saving an extended attribute.

Figure 162: Process Flow of Create Extended Attribute



## Configurable UI

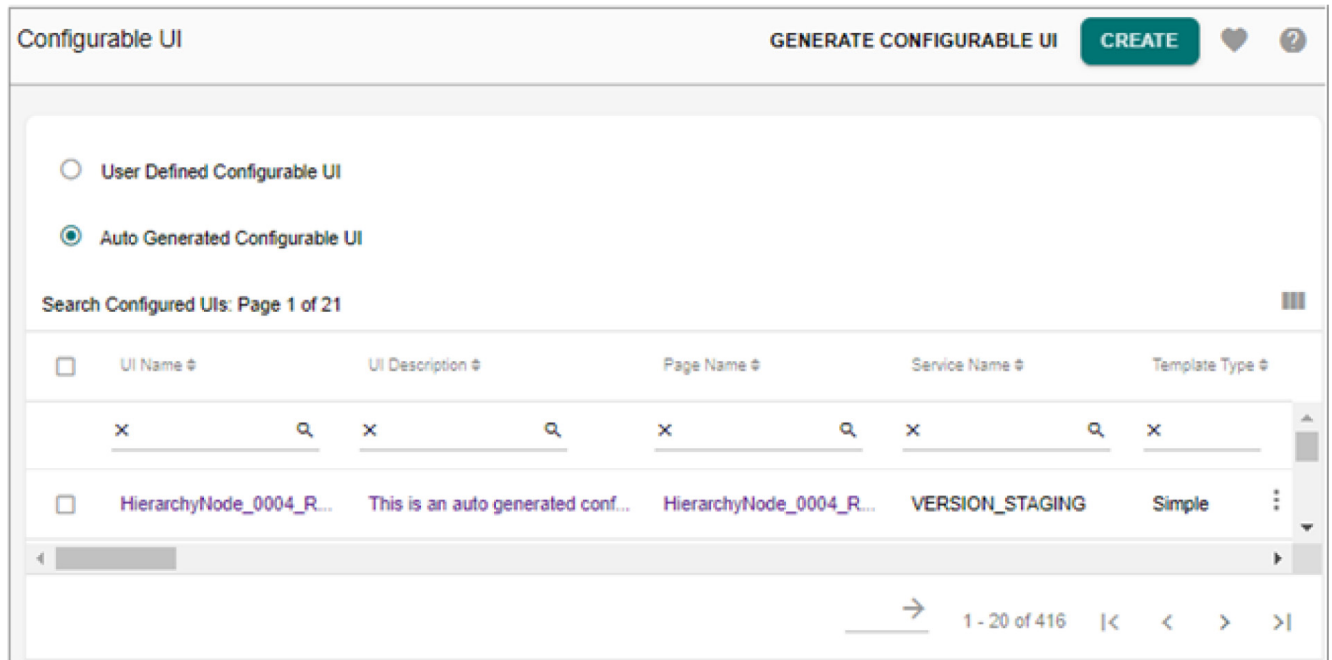
Configurable UIs display the records stored in the table. Configurable UIs are of two types.

- **User Defined Configurable** UIs are created from the Manage Configurable UI link in the left navigation pane where you can customize the following:
  - Update the description.
  - Select columns to display.
  - Change column names for display.
  - Define join conditions.
  - Specify maximum rows per page (configure pagination)
  - Add color to the column.
  - Make the column editable or searchable.
  - Add a default value to the column.
  - Select a key column.
  - Configure help file.
  - Assign possible actions available for the configurable UI such as: Create, Copy, Delete, Edit and Mass update.
- **Auto Generated Configurable** UI are created or generated automatically from the Generate Configurable UI in the left navigation pane. You can select the service name and select the tables to be generated.

### ***Manage Base and Version Config UI***

The Manage Base Configurable and Manage Version Configurable options on the Manage Hierarchy Objects UI allows you to navigate to the Manage Base and Manage Version Configurable UI ([Figure 163](#)) of the corresponding hierarchy object. The Manage Base Config and Manage Version Config options calls an out of the box feature of RDM, which filters the UI Name based on the selected table from the Manage Hierarchy Objects UI. On the Configurable UI, using the Preview option, you can download template and upload the data for the object and manage the hierarchy object details (manage activities such as create, copy, delete, edit and regenerate).

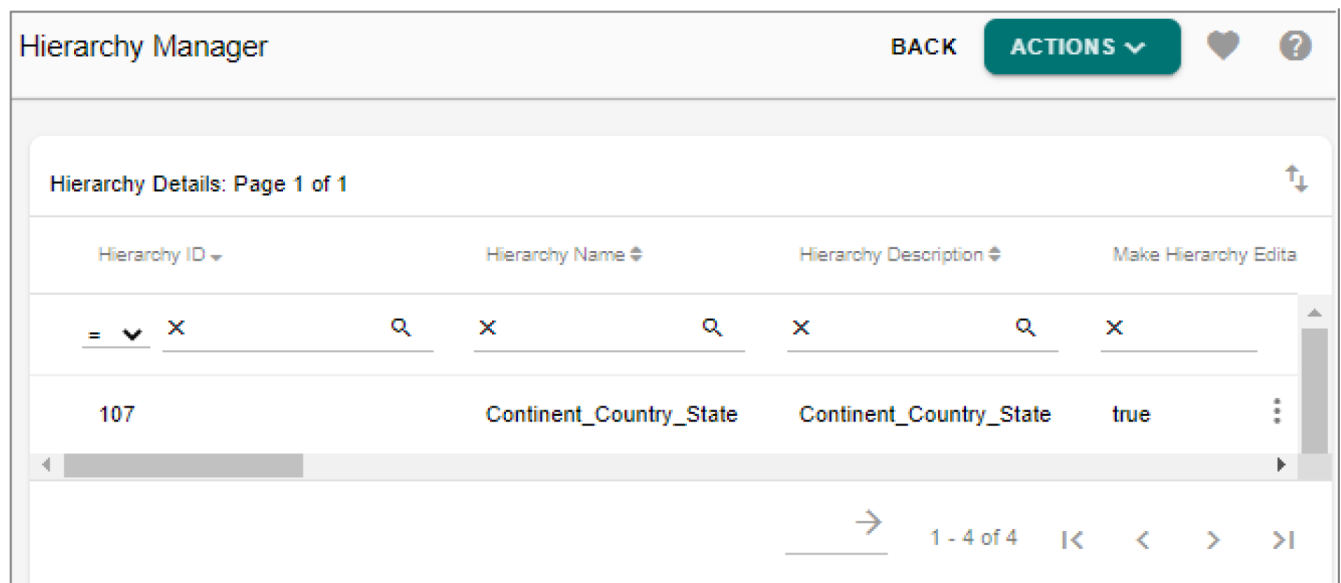
Figure 163: Configurable UI



### Show Hierarchy and Show Relationship

The Show Hierarchy option on the Manage Hierarchy Objects UI allows you to navigate to Hierarchy Manager UI, which displays the list of hierarchies where the selected table is used in. On the Hierarchy Manager UI, you can view hierarchy and perform all the functionalities available on the Hierarchy Manager UI as in [Figure 164](#).

Figure 164: Hierarchy Manager



The Show Relationship option on the Manage Hierarchy Objects UI allows you to navigate to corresponding Manage Relationship UI, which lists the tables where the relationship object is used either as parent object or a child object. The relationships are stored in the

RelationalObjectMap table where it stores the relationship of the RO\_Ids of the objects. You can see on this UI, the Parent object name, Child object name and the target object name that connects the two by storing the node Id of the parent object in the parent node Id column and it is mapped to the node Id of the child object.

Figure 165: Manage Relationships

Manage Relationships			
			<a href="#">BACK</a> <a href="#">CREATE</a> <span>?</span>
Data Relationship: Page 1 of 1			
RomId	Relation Name	Relation Description	Parent Object Name
= <input type="text"/> <input type="button" value="X"/>	<input type="text"/> <input type="button" value="X"/>	<input type="text"/> <input type="button" value="X"/>	<input type="text"/> <input type="button" value="X"/>
220	Continent_Country	Continent_Count...	Continent
<div> <input type="button" value="←"/> <input type="button" value="→"/> 1 - 1 of 1 <input type="button" value="↶"/> <input type="button" value="↷"/> </div>			

On the Manage Relationships UI, click on the Relation Name link to display the details of the selected Relation Name. For various activities performed on the Manage Relationships UI, see [Section : “Manage Relationship.”](#)

Figure 166: Relationship Details

Relationship Details	
Relationship Name	Parent Object
Continent_Country	Continent
Child Object	Relationship Object
Country	Continent_Country
CLOSE	

## Manage Relationship

The second step in creating a hierarchy is to define the Relationship between the Hierarchy Objects or Levels. The Manage Relationships UI displays the list of relationships defined where you may choose to delete, view data, link data and create new relationships.

Navigate to **Reference Data -> Hierarchy Management ->Manage Relationships**.

Figure 167: Manage Relationships

Table 8 displays the UI fields on Manage Relationships UI.

Table 8: Manage Relationship UI Fields

UI Field	Purpose
RomId	Uniquely identify a relationship object
Relation Name	Relationship name
Relation Description	Description of the relationship
Parent Object Name	Parent object name serves as the parent table where the NODE_ID will serve as the PARENT_NODE_ID to its child table. Ex: HierarchyNode1,Continent
Child Object Name	Child object name serves as the child table linked to the parent table. Ex: HierarchyNode2, Country
Target Object Name	Target object name contains the relationship between the parent and the child objects. Ex: HierarchyNodeRltn1, Continent_Country

You can perform the following tasks on the Manage Relationship:

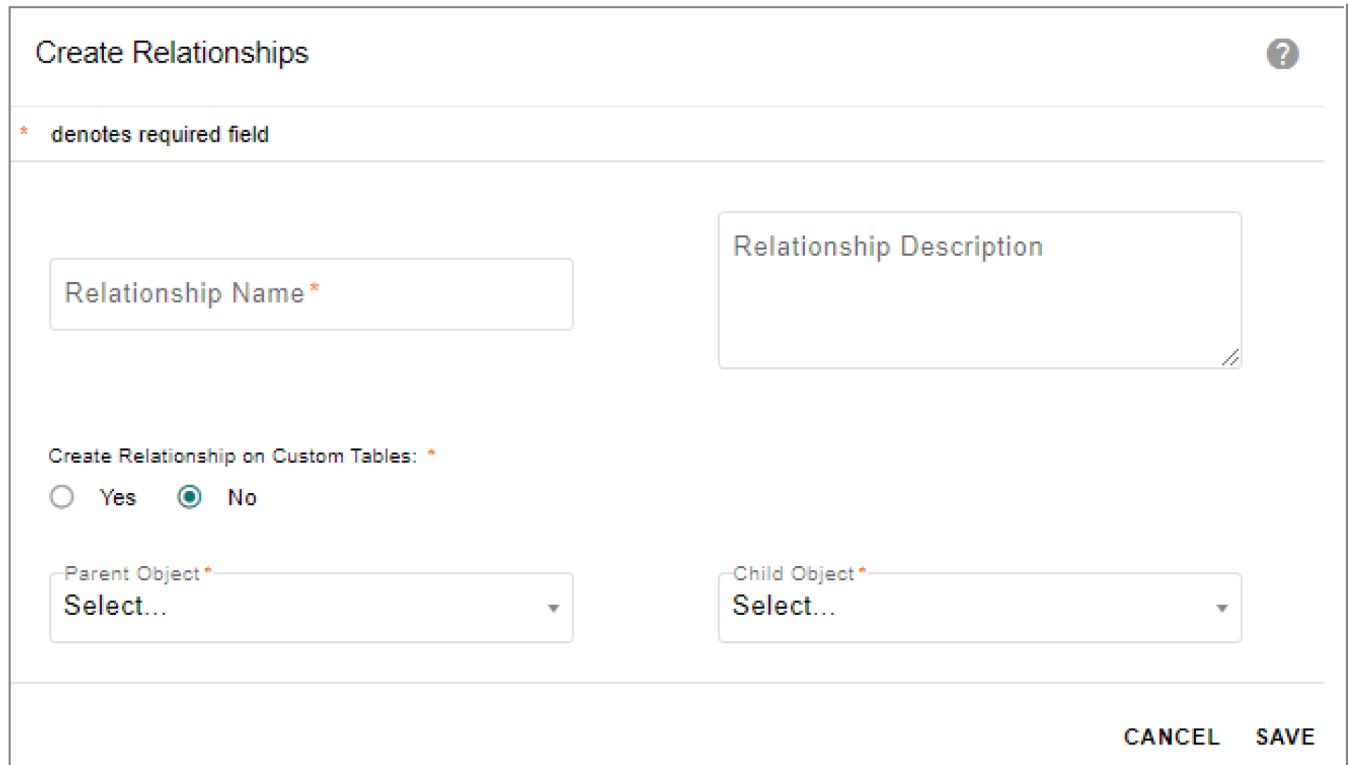
- [Create Relationship](#)

- [Load Relationship Data](#)
- [Link and View Data](#)
- [Delete Relationship](#)

## Create Relationship

To create new relationship, navigate to **Define Relationship** UI from **Manage Relationships** UI using **Create** button.

Figure 168: Create Relationship



The screenshot shows the 'Create Relationships' form. At the top, there's a title bar with a question mark icon. Below it, a legend indicates that an asterisk (\*) denotes a required field. The form contains several input fields: 'Relationship Name \*' (a text box), 'Relationship Description' (a larger text area), 'Create Relationship on Custom Tables: \*' (a section with radio buttons for 'Yes' and 'No', where 'No' is selected), 'Parent Object \*' (a dropdown menu showing 'Select...'), and 'Child Object \*' (another dropdown menu showing 'Select...'). At the bottom right, there are 'CANCEL' and 'SAVE' buttons.

On the **Create Relationship** page ([Figure 168](#)), enter the following and click **Save**.

- **Relationship Name:** unique name of the relationship.



The relationship name should not be more than 50 characters. In case of names having more than 50 characters, Teradata Database will truncate value of name with maximum length and in this case it would be 50 characters.

- **Relationship Description:** brief description of the relationship being created.
- **Parent Object:** displays the list of the objects. You can select the required hierarchy object to be used as parent object.
- **Child Object:** displays the list of the objects. You can select required hierarchy object to be used as child object.

- **Create Relationship on Custom Tables:** select the option as “No” to create relationship on OOTB Master tables.

Once the relationship being created is saved, the relationship will be displayed on the Manage Relationship UI as in [Figure 169](#).

Figure 169: Manage Relationships

RomId	Relation Name	Relation Description	Parent Object Name	Child Object
1101	Continent_Country	Continent_Count...	Continent	Count

## Load Relationship Data

The load for Continent\_Country relationship is demonstrated in the following section. First, prepare the Excel sheet to be loaded to the HierarchyNodeRelationship table (for example - Continent\_Country table) as in [Figure 170](#).

Figure 170: Load Hierarchy Relationship Data

A	B	C	D	E	F
Child Node Name	Parent Node Name	Target Id	Authorization	Source	Created By
Algeria	Africa				
Liberiya	Africa				
Libiya	Africa				
Malawi	Africa				
Mali	Africa				
Benin	Africa				
Botswana	Africa				

Now navigate to the Continent\_Country configurable UI (on the left navigation, click **Tables** -> **Master Table** -> **Hierarchy Relations** and click Continent\_Country .

On the Continent\_Country Configurable UI, click **Upload Excel** as in [Figure 171](#) to upload the relationship data.

Figure 171: Hierarchy Relationship—Excel Upload

The screenshot shows a web interface for managing hierarchy relationships. At the top, there's a header bar with the title "Continent\_Country" and several icons including a search icon, "UPLOAD EXCEL", and a plus sign. Below the header is a table with columns: "Child Node Name", "Parent Node Name", "Target Id", "Authorization", and "Source". The table is currently empty. At the bottom right, there are pagination controls showing "1 - 0 of 0" and navigation arrows.

Now the uploaded data is available on the Continent\_Country Configurable UI.

Similarly create other required hierarchy relationship table (Country\_State) as in [Figure 172](#) and load the corresponding data.

Figure 172: Manage Relationships

The screenshot shows the "Manage Relationships" UI. At the top right is a "CREATE" button. Below the header is a table with columns: "RomId", "Relation Name", "Relation Description", "Parent Object Name", and "Child Object". The table contains two rows of data. Below the table is a pagination bar showing "1 - 2 of 2" and navigation arrows.

RomId	Relation Name	Relation Description	Parent Object Name	Child Object
1102	Country_State		Country	State
1101	Continent_Country	Continent_Count...	Continent	Count

## Link and View Data

Data of parent and child objects of a relationship can be linked using Link data facility. The same linked data can be viewed using view data facility. Using the Link Data option, you can select the parent data and link it with the child data manually. On the Manage Relationships UI, select the relationship and click **Link Data**. The **Parent Data** page displays all the available parent data as in [Figure 173](#). You can select a parent data and click **Child Data**. The **Child Data** page displays all the available child data. You can select multiple child data to link as in [Figure 174](#) and click **Link Data**.



Figure 173: Parent Data

Parent Data

CANCELSELECT CHILD DATA?

Relation Name

Continent\_Country

^

Continent: Page 1 of 1

Node Id	Node Name	Description	Node External Id
<div>x</div> <div>Q</div>	<div>x</div> <div>Q</div>	<div>x</div> <div>Q</div>	<div>x</div>
<div><input checked="" type="radio"/></div> <div>6</div>	Africa		
<div><input type="radio"/></div> <div>5</div>	Europe		

→

1 - 6 of 6

|<

<

>

>|

Figure 174: Select Child Data

Child Data

CANCEL LINK DATA

Selected Parent Object Data

Select Child Data

Relation Name

Continent\_Country

Country: Page 1 of 1

<input type="checkbox"/>	Node Id ▾	Node Name ▴	Description ▴	Node External Id ▴
	x	Q	x	Q
	x	Q	x	Q
<input checked="" type="checkbox"/>	195	Egypt		
<input checked="" type="checkbox"/>	186	Sudan		

1 - 2 of 2

|< < > >|

You can also view the linked data using **View Data** option. On the Manage Relationships UI, click on the Menu icon corresponding to the required relationship and click **View Data**. The **Parent Data** page displays all the available parent data as in [Figure 175](#). You can select a parent data and click **Linked Data**. The **Selected Parent Object Data** page displays all the linked child data as in [Figure 176](#). On the **Selected Parent Object Data** page, you can delete the linked child using Delete option.

Figure 175: Select Parent Data

Parent Data

BACKLINKED CHILD?

Relation Name

Continent\_Country

^

Continent: Page 1 of 1

Node Id	Node Name	Description	Node External Id
<div>x</div> <div>Q</div>	<div>x</div> <div>Q</div>	<div>x</div> <div>Q</div>	<div>x</div>
<div><input checked="" type="radio"/></div> <div>6</div>	Africa		
<div><input type="radio"/></div> <div>5</div>	Europe		
<div><input type="radio"/></div> <div>4</div>	Antarctica		

→

1 - 6 of 6

I<<<>>>I

Figure 176: Selected Parent Object Data

BACKCANCELDELETE

Selected Parent Object Data

Linked Child Object Data

Country: Page 1 of 1

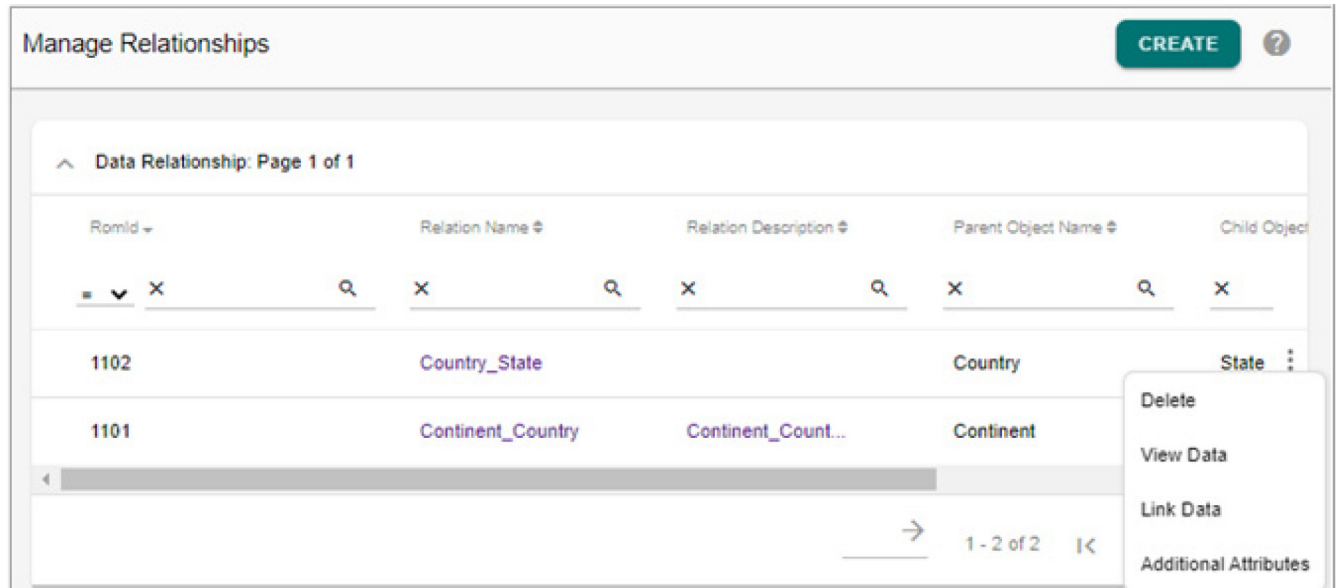
Node Id	Node Name	Description	Node External Id
<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>
<input checked="" type="radio"/> 181	Gambia		
<input type="radio"/> 177	Malawi		
<input type="radio"/> 176	Malawi		

→ 1 - 18 of 18<<<>>>

Delete Relationship

The Delete option on the Manage Relationship UI as in [Figure 177](#) allows you to delete the selected relationship.

Figure 177: Manage Relationships-Delete Option



If the relationship is being used by any Hierarchy, you need delete the hierarchy and then delete the relationship.

## Hierarchy Manager

Hierarchy is defined as a logical grouping of objects or components in the system. Instances of any arbitrary object or component in the system can be grouped together to form a tree structure, thus constituting a hierarchy.

Navigate to **Reference Data -> Hierarchy Management -> Hierarchy Manager**

[Table 9](#) displays the Hierarchy table details.

Table 9: Hierarchy Table Details

UI Field	Purpose	Comments
Hierarchy ID	Uniquely identifies a hierarchy	Auto generated when creating a hierarchy.
Hierarchy Name	Name of the Hierarchy being created	Required
Hierarchy Description	Description of the hierarchy	Not Required but strongly recommended.

Table 9: Hierarchy Table Details

UI Field	Purpose	Comments
Hierarchy Type	Determines the type of Hierarchy created.	There are 3 types: Balanced or structured hierarchy Unbalanced hierarchy Ragged hierarchy. For more details on refer to Master Data Management Server Guide.pdf.
Hierarchy View Role	Only the selected roles will be able to view the hierarchy.	
Hierarchy Edit Role	Only the selected roles will be able to edit the hierarchy.	

Navigate to **Reference Data -> Hierarchy Management ->Hierarchy Manager**

You can perform the following tasks on the Hierarchy Manage UI.

- [Create Hierarchy](#)
- [Edit Hierarchy](#)
- [Create Version](#)
- [View Version](#)
- [Promote Hierarchy](#)
- [Delete Hierarchy and Hierarchy Version](#)
- [Manage Custom Actions](#)
- [View Hierarchy](#)
- [Compare Hierarchy](#)
- [Hierarchy Viewer](#)
- [Export Hierarchy](#)
- [Import Hierarchy](#)
- [Merge Hierarchy](#)
- [Hierarchy Logs](#)
- [Manage Rollup Attributes](#)
- [Hierarchy Statistics](#)
- [Manage Cross Reference](#)

## Create Hierarchy

To create, select **Create Hierarchy** option from **Tasks** dropdown on **Hierarchy Manager** UI as in [Figure 178](#).

Figure 178: Hierarchy Manager

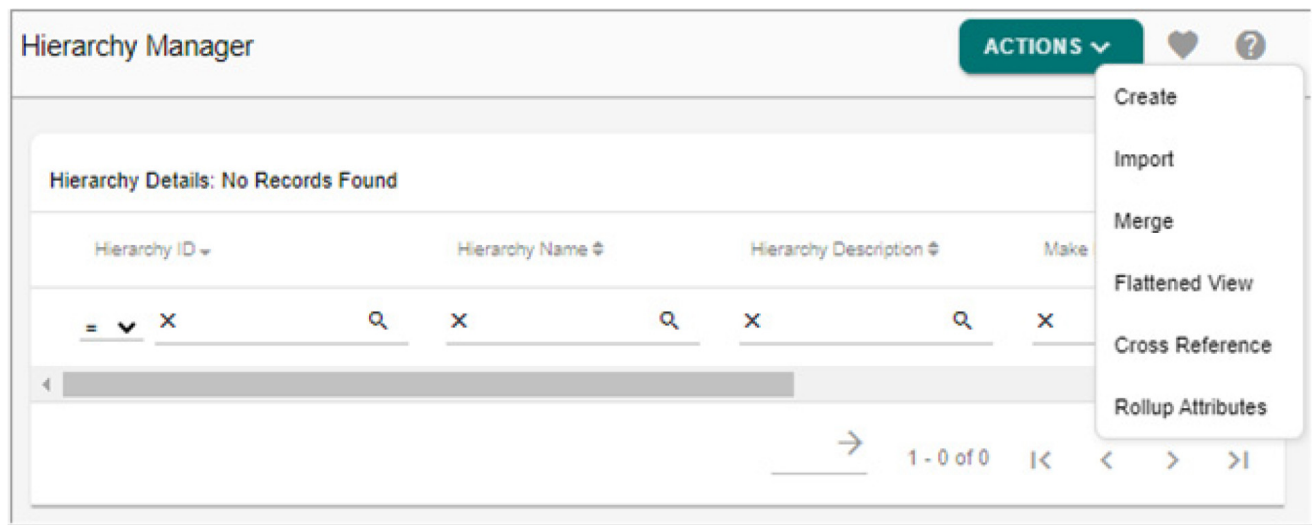


Figure 179: Create Hierarchy

On the **Create Hierarchy** page (Figure 179),

- **Hierarchy Name:** unique name of the hierarchy.
- **Description:** brief description about the hierarchy being created.  
You can configure a set of advanced settings for hierarchy.
- Click on **Advanced Settings** checkbox to configure that.

- **Base hierarchy editable:** you can make the base hierarchy editable or non editable.

If base hierarchy is selected as non-editable, the hierarchy will be non-editable to all the user groups irrespective of what role has been selected for 'Select role for edit'. You cannot perform any changes including drag and drop to the underlying hierarchy data from Hierarchy Viewer. By default, the hierarchy is editable.



- **Hierarchy on Custom Tables:** select No option to create hierarchy on OOTB master table. Hierarchy created on OOTB master tables are template based hierarchies.

Hierarchy can be also be created on custom Master tables in addition to the OOB data model provided for Hierarchy. In create object, create relationship and create Hierarchy UIs, an option to create objects, relationships and hierarchy on custom tables are provided. On selection of that option, custom master tables/views will be available for use. For more details, see [Section : “Hierarchy on Custom Tables.”](#)

- **Select role for view:** the selected role can only view the composition of the hierarchy and cannot make changes to hierarchy details.
- **Select role for edit:** the selected role can make modifications to the structure or composition of the hierarchy.
- **Publish Hierarchy Data:** you can enable to publish flattened representation view of a Hierarchy data by selecting **Yes** option. By default, No option is selected.  
If publishing is enabled for a hierarchy, the required publishing metadata for the hierarchy will be created automatically as in [Figure 180](#). The actual publish of hierarchy data can be triggered from Manage Publication Objects UI when required.
- **Display Unassigned Nodes:** You can use this option to display orphan nodes (Nodes that do not have a parent) in Hierarchy Viewer. The orphan nodes will be displayed under a dummy parent node called Unassigned\_<HierarchyObjectName> in viewer. The orphan nodes then can be dragged to valid parent.

Figure 180: Manage Publication Objects

Manage Publication Objects

ADD

Add New

\* denotes required field

Service   
 Input

Name

Description

Publication Objects: Page 1 of 1

Name	Description
V_HIER_DENORM_1	Publishing View

1 - 1 of 1

- **Available Hierarchy Objects:** displays a list of all the available hierarchy objects that can be selected to create a hierarchy relationship.
- **Selected Hierarchy Objects:** displays a list of all the selected hierarchy objects to create hierarchy relationship.
- Navigate to the next page using the **Next** button to select the relationships as in [Figure 181](#).

Figure 181: Create Hierarchy—Manage Relationships

Create Hierarchy?

Selected Details

Manage Relationships

Data Relationship

<input type="checkbox"/>	RomId	Relation Name	Relation Description	Parent Object Name	Child Object Name	Target Object Name
<input checked="" type="checkbox"/>	1101	Continent_Country	Continent_Count...	Continent	Country	Continent_Count...
<input checked="" type="checkbox"/>	1102	Country_State		Country	State	Country_State

BACKSAVE

- On the **Create Hierarchy—Manage Relationships** page, select the data relationships and click **Save** to save the hierarchy being created.

Figure 182: Hierarchy Manager

Hierarchy Manager

ACTIONS

Hierarchy created successfully.

Hierarchy Details: Page 1 of 1

Hierarchy ID	Hierarchy Name	Hierarchy Description	Make Hierarchy Editable	Hierarchy View Role
1	Continent_Cpuntry_State	Continent_Cpuntry_State	true	ALL

Message: “Hierarchy created successfully” is displayed as in [Figure 182](#).

Hierarchy on Custom Tables

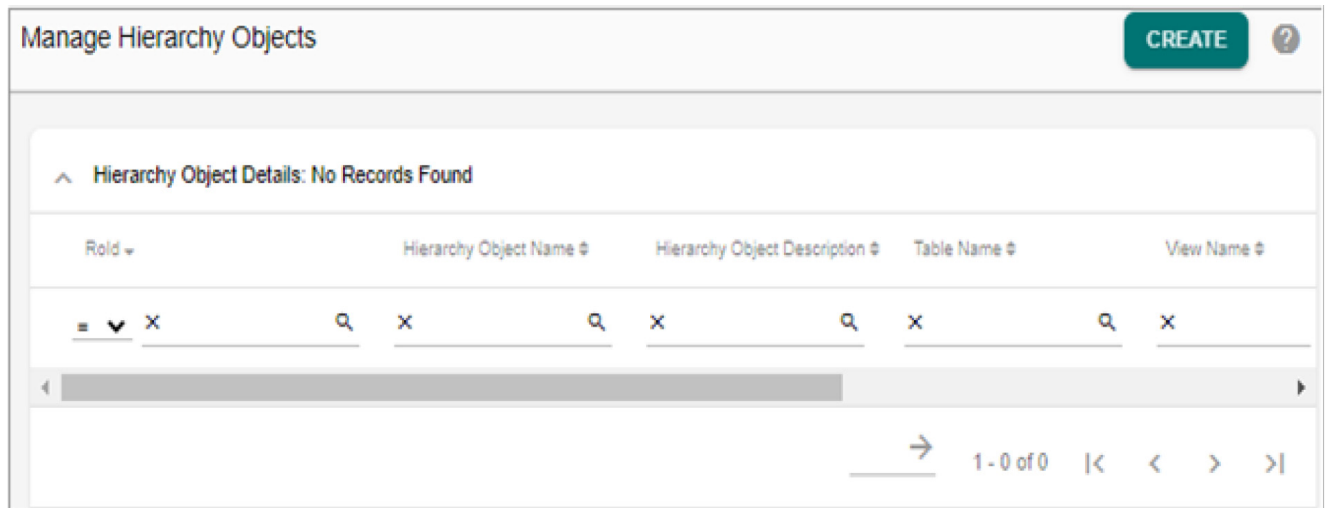
RDM provide support to create hierarchy objects, relationships and hierarchy on custom master tables/views. In create object, create relationship and create Hierarchy UIs, an option to select custom master tables/views is displayed for selection. The following section

describes the creation of object, relationship and hierarchy on custom Master tables (Account and Store).

## Create Hierarchy Object on Custom Tables

To create new hierarchy objects on custom tables, navigate to **Create Hierarchy Object** UI from **Manage Hierarchy Objects** UI using **Create** button.

Figure 183: Manage Hierarchy Objects



On **Create Hierarchy Object** page, enter the following as in [Figure 184](#) and click **Save**.

- **Object Name:** unique name of the object.
- **Object Description:** brief description about the object being create.
- **Create Object on Custom Tables:** select option **Yes** to create objects on custom master tables. If **Yes** option is selected, **Create Hierarchy Objects** UI refreshes to display the following fields:
  - **Object Data Source:** data source for the object being created. The Object Data Source list display both the master tables and user created views.
  - **Master URL:** URL configured for the object.
  - **URL Parameter:** the URL parameter should contain the key column names (logical column name in case of Master tables). Columns of URL parameter should be able to uniquely identify records of table on which object is created. Multiple columns can be added separated by ||. For example: AccountId || PartyTypeCd.
  - **No. of Linked Columns:** it specifies the number of columns used to link the objects.
  - **Link Column:** link columns selected should be able to uniquely identify rows in the table on which object is created.
  - **No. of Display Columns:** it specifies the number of columns to be displayed on the hierarchy viewer.
  - **Display Column:** it specifies the column that will be displayed on the hierarchy viewer.
  - **No. of Sort Columns:** it specifies the number of columns used for sorting.

- **Sort Column:** it specifies the column that will be used for sorting nodes in hierarchy viewer.

Figure 184: Create Hierarchy Objects

The screenshot shows the 'Create Hierarchy Objects' form. At the top, there is a title bar with a question mark icon. Below the title bar, a legend indicates that an asterisk (\*) denotes a required field. The form contains several input fields and controls:

- Object Name \***: A text input field containing the value 'Account'.
- Object Description**: A text area containing the value 'Account'.
- Create Object on Custom Tables: \***: A section with two radio buttons, 'Yes' (selected) and 'No'.
- Link Column 1:**: A dropdown menu showing 'AccountId'.
- No. of Display Columns:**: A dropdown menu showing '1'.
- Display Column 1:**: A dropdown menu showing 'AccountName'.
- No. of Sort Columns:**: A dropdown menu showing '1'.
- Sort Column 1:**: A dropdown menu showing 'AccountName'.

At the bottom right of the form, there are two buttons: 'CANCEL' and 'SAVE'.

Similarly create custom object on Store table.

Once saved, the Hierarchy Object will be created and will be available on the Manage Hierarchy Objects UI as in [Figure 185](#).

Figure 185: Manage Hierarchy Objects

Role	Hierarchy Object Name	Hierarchy Object Description	Table Name
1107	Store	Store	Store
1106	Account	Account	Account

### Create Relationship on Custom Tables

To create new relationship on custom tables, navigate to **Define Relationship** UI from **Manage Relationships** UI using **Create** button.

On the **Create Relationship** page (Figure 168), enter the following and click **Save**.

- **Relationship Name:** unique name of the relationship.
- **Relationship Description:** brief description of the relationship being created.
- **Parent Object:** displays the list of the objects. You can select the required hierarchy object to be used as parent object.
- **Child Object:** displays the list of the objects. You can select required hierarchy object to be used as child object.
- **Create Relationship on Custom Tables:** select this option to create relationship on custom tables. If **Yes** option is selected, the **Relationship Mapping** section is displayed. The relational object selected in Relationship Mapping section holds the relationship of parent and child objects. It can be either Relationship Data or Predefined.
  - Select Relationship Data if you want to store the relationship details in out of the box table.
  - Select the option Pre-defined Relationship Object, if you want to use custom defined relational object to store the relationship details.

If the Pre-defined Relationship Object is selected, the Create Relationship page displays the Relationship Object dropdown.

  - The Relationship Object dropdown displays all the existing objects as well as user defined objects. On selection of a relationship object, parent and child object link columns are displayed. For each of the parent or child link column, a drop down populated with the columns of the selected relationship object is displayed. You can select appropriate column mappings to store the relationship details.

Figure 186: Create Relationship

Create Relationships

\* denotes required field

Relationship Name \*

Store\_Account

Relationship Description

Store\_Account

Create Relationship on Custom Tables: \*

☒ Yes ☐ No

Parent Object \*

Store

Child Object \*

Account

Relationship Mapping:

☒ Relationship Data

☐ Pre-Defined Relationship Object

CANCEL

SAVE

Once the relationship being created is saved, the relationship will be displayed on the Manage Relationship UI as in [Figure 187](#).

Figure 187: Manage Relationships

Manage Relationships CREATE ?

✔ Relationship created successfully.

^ Data Relationship: Page 1 of 1

RomId ▾	Relation Name ⇅	Relation Description ⇅	Parent Object Name ⇅	Child Object Name ⇅	Target Object Name ⇅	
<div><div>=</div><div>▾</div><div>×</div></div> <div>Q</div> <div>×</div>	<div>×</div> <div>Q</div>	<div>×</div> <div>Q</div>	<div>×</div> <div>Q</div>	<div>×</div> <div>Q</div>	<div>×</div> <div>Q</div>	
1218	Code set1	Code set	CodeSet_Object	Domain_Object	RODV	⋮
1217	Code_Domain_Relation	Code_Domain_Rel...	CodeSet_Object	Domain_Object	RODV	⋮
1122	RagCountry_City	RagCountry_City	RagCountry	RagCity	RagCountry_City	⋮
1121	RagState_City	RagState_City	RagState	RagCity	RagState_City	⋮
1120	RagCountry_State	RagCountry_Stat...	RagCountry	RagState	RagCountry_Stat...	⋮
1119	UnbalReIn	UnbalReIn	UnbalObj	UnbalObj	UnbalReIn	⋮
1118	Country_State	Country_State	Country	State	Country_State	⋮

→

1 - 8 of 8

|<

<

>

>|

Create Hierarchy

To create, select **Create** option from **Actions** dropdown on **Hierarchy Manager** UI.



Figure 188: Create Hierarchy

**Create Hierarchy** ?

\* denotes required field

Hierarchy Name \*  
Code Set

Description  
Code set

Available Hierarchy Objects

- Customer4
- CustomerMaster
- PEPSI\_SRC\_CCENTER\_US\_CUSTOMER
- PEPSI\_SRC\_LEGACY\_US\_CUSTOMER
- Customer3
- Customer4
- RODV\_0001
- RelationalObjectColumnData
- SMKTST\_SRC\_EXT\_CUSTOMER
- SYS\_MANUAL\_OVERRIDE
- TestDictionaryModel

Selected Hierarchy Objects

- \* CodeSet\_Object
- Domain\_Object

☒ Advanced Settings

**Advanced Settings**

Base hierarchy editable: \*  
☒ Yes ☐ No

Hierarchy on Custom Tables: \*  
☒ Yes ☐ No

Select role for view \*  
ALL

Select role for edit \*  
ALL

Publish Hierarchy Data: \*  
☐ Yes ☒ No

Display Unassigned Nodes: \*  
☐ Yes ☒ No

**NEXT**

On the **Create Hierarchy** page (Figure 188), all the field details remain same as creating hierarchy on OOTB tables and only for **Hierarchy on Custom Tables** option, select **Yes** option to create hierarchy on custom master table. The Available Hierarchy Objects list displays the objects created on custom tables. For details on all other fields, see [Section : “Create Hierarchy.”](#)

- Navigate to the next page using the **Next** button to select the custom relationships as in [Figure 189](#) and click **Save**.

Figure 189: Create Hierarchy—Manage Relationships

Create Hierarchy?

^ Selected Details

Hierarchy Name

Code Domain

Description

Code Domain HIE

Select role for view

ALL

Select role for edit

ALL

Editable Hierarchy

Yes

Publish Hierarchy Data

No

Display Unassigned Values

No

Manage Relationships

^ Data Relationship

☐

RomId

Relation Name

Relation Description

Parent Object Name

Child Object Name

Target Object Name

☒

1217

Code\_Domain\_Relation

Code\_Domain\_Rel...

CodeSet\_Object

Domain\_Object

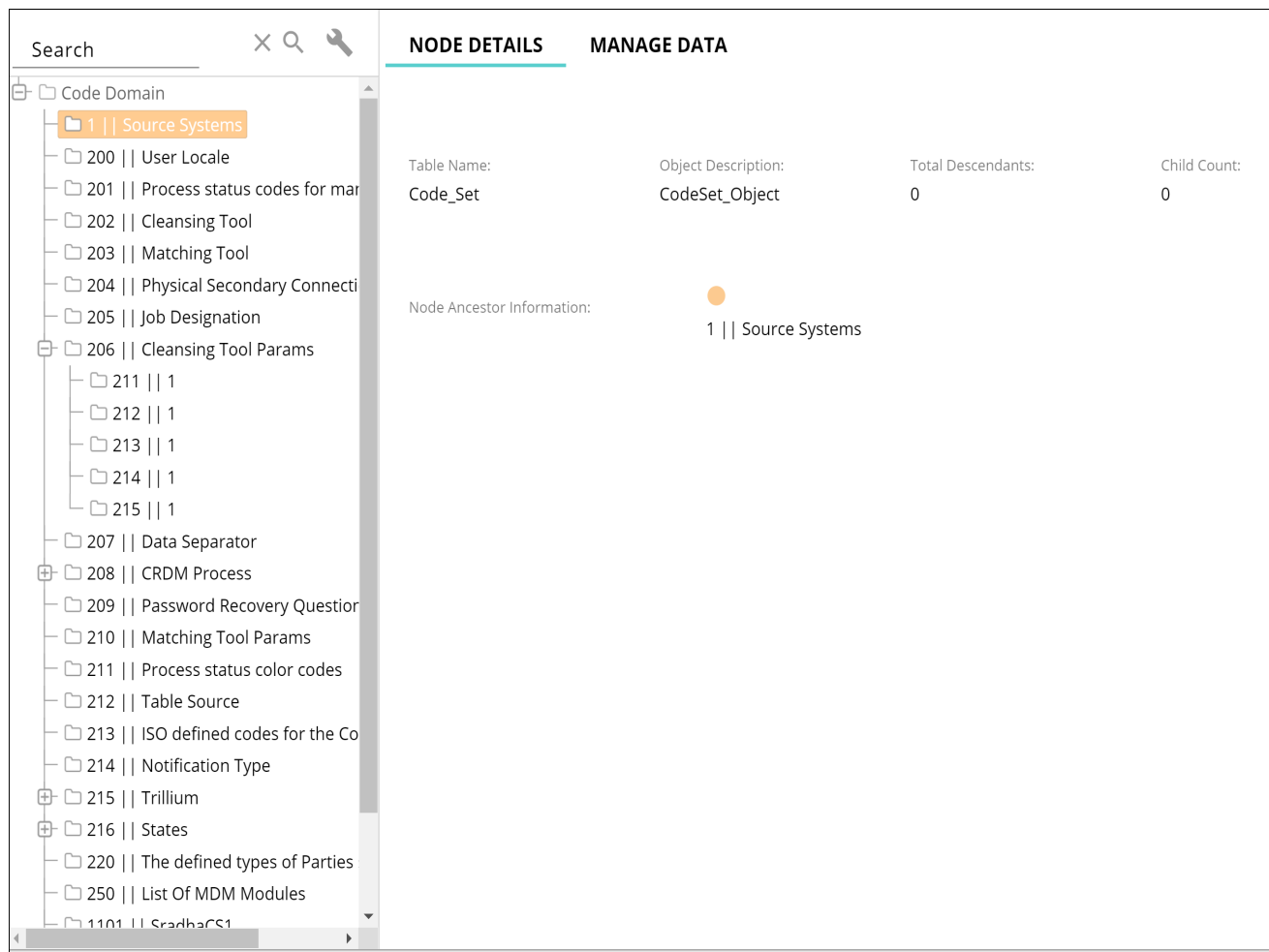
RODV

BACK

SAVE

For Custom Hierarchies created using custom objects and relations, on the Hierarchy Viewer, Some of the advanced options (Show children, Show Peers, Show Siblings) available in Hierarchy Viewer of OOB template based hierarchy are not available in custom table based hierarchy as in [Figure 190](#). Also, the options (Manage Roll-up attributes, Import hierarchy, Hierarchy Merging) available in Hierarchy Designer of OOB template based hierarchy are not available in custom table based hierarchy.

Figure 190: Hierarchy Viewer



Edit Hierarchy

Once the hierarchy is created, you can modify the hierarchy details using the **Edit** option. On the **Hierarchy Manager** UI, click on the Menu icon corresponding to the required hierarchy and click **Edit** to **Edit Hierarchy**.

Figure 191: Edit Hierarchy

Edit Hierarchy

?

Hierarchy Name

Balanced Hierarchy

Description

Balanced Hierarchy

Base hierarchy editable: \*

☒ Yes ☐ No

Display Unassigned Nodes: \*

☒ Yes ☐ No

Select role for view \*

ALL

Select role for edit \*

ALL

CANCEL

SAVE

Selected ROs

Object Details

Ro Id	Table/View Name	Table/View Description	Logical Table Name	Physical View Name	Uri	Uri Par
220	State	State	State		/mdm/start.x2ps?STAR...	N
219	Country	Country	Country		/mdm/start.x2ps?STAR...	N

Selected Relations

Relation Details

Rom Id	Relation Name
221	Country_State
220	Continent_Country

On the **Edit Hierarchy** page (Figure 191), you can:

- Modify hierarchy description, select different roles for view and edit hierarchy. The Is Promotable checkbox is non editable.
- Select any hierarchy object and click on **Edit** to modify hierarchy object details or to define, update or delete filter. For details, see [Section : “Edit Hierarchy Objects.”](#)
- Select any relationship and link parent and child data using **Link Data** option or view the linked data using the **View Data** option. For details, see [Section : “Link and View Data.”](#)

## Create Version

Versions of hierarchy can be created using create version feature. On creation of a version, the entire hierarchy with data is versioned against a version ID. Multiple versions of a hierarchy can be created at different stages of the hierarchy. Validation rules are supported on data of version staging. Rules created on Master staging gets applied for corresponding version staging tables.

Creating a version allows you to perform drag and drop to make versions of hierarchy and submit the version for approval.

Note: Before you submit a version for approval, an approval object must first be created on the selected hierarchy. For creating approvals, refer to Manage Approvals section in RDM Table Maintenance chapter for more details.

Figure 192: Create Version

**Create Version** ?

\* denotes required field

Reason \*  
Hierarchy Version for analysis

Version \*  
Version1

Make version editable:  
☒ Yes ☐ No

Display Unassigned Nodes:  
☐ Yes ☒ No

Effective Start Date(MM/DD/YYYY) \*  
2/26/2021

Effective End Date(MM/DD/YYYY) \*  
12/31/9999

**CREATE**

On the **Create Version** pop-up (Figure 192),

- **Reason:** brief description about the purpose of creating the hierarchy version.
- **Version:** specifies the version details like version number etc.
- **Copy Version:** select this checkbox to create a copy of any versioned hierarchy into another version. For more details, see [Section : “Copy Hierarchy Version.”](#)
- **Make Version Editable:** you can either make the version editable or not. If **Yes** option selected, on the **Hierarchy Version-Viewer: <version>** page, you would be able to perform

cut, paste, drag, drop, and clear operations on any node. If **No** option selected, then the cut, paste, drag, drop, and clear options would be disabled.

- **Display Unassigned Nodes:** select this option if you want to manage orphan nodes in Hierarchy Viewer. You can use this option to display orphan nodes (Nodes that do not have a parent) in Hierarchy Viewer. The orphan nodes will be displayed under a dummy parent node called Unassigned\_<HierarchyObjectName> in viewer. The orphan nodes then can be dragged to valid parent.
- **Start Date and End Date:** the start date and end date specifies the start and end date for the version being created.

## Copy Hierarchy Version

The Copy Version feature provides an option to create a copy of any versioned hierarchy. By creating a copy of version hierarchy, the version hierarchy metadata is copied into a new version hierarchy being copied.

To copy hierarchy version, perform the following steps:

- 1 On the **Create Version** pop-up, enter all the required fields as described in the [Section : “Create Version”](#).

- 2 Select the **Copy Version** checkbox.

The **Select version to be copied** dropdown displays all the non expired hierarchy versions of the selected hierarchy.

- 3 Select the required version hierarchy to be copied as in [Figure 193](#) and click **Create**.

Message: “Version created successfully” is displayed.

Figure 193: Create Version

**Create Version** ?

\* denotes required field

Reason \*  
Hierarchy Version for analysis

Version \*  
Version1

Make version editable:  
☒ Yes ☐ No

Display Unassigned Nodes:  
☐ Yes ☒ No

Effective Start Date(MM/DD/YYYY) \*  
2/26/2021

Effective End Date(MM/DD/YYYY) \*  
12/31/9999

**CREATE**



During the copy hierarchy version process, the hierarchy metadata is copied from the master hierarchy and not from the corresponding version hierarchy. If the master hierarchy configurations (like display columns, sort columns of the hierarchy objects, edit/view role of hierarchy etc.) are changed after the creation of the version, which is being copied, then the copied version will reflect the configuration of the master hierarchy and not the version hierarchy.

## View Version

You can view the hierarchy version using **Manage Version** option on **Hierarchy Manager** UI. On the **Manage Version** page (Figure 194), you can specify the version to view on the hierarchy viewer UI.

Figure 194: Manage Versions

**Manage Versions**

Hierarchy Name  
Continent\_Country\_State

Select Version  
Version1

Version Status  
DRAFT

☐ Show Archived Versions

Effective Start Date(MM/DD/YYYY) \*  
02/26/2021

Effective End Date(MM/DD/YYYY) \*  
12/31/9999

**CANCEL VIEW VERSION PROMOTE VERSION**

Note: If filters were applied on the selected hierarchy version, enter the filter value and the hierarchy viewer displays the filtered versioned data. If the hierarchy is not selected for promotion during the creation of hierarchy, the Manage Versions popup displays only the Hierarchy Name and Select Version fields.

On the Manage Version popup, enter the following:

- Select the hierarchy version to view from **Select Version** dropdown.
- Click **View Version** to view the hierarchy on the hierarchy viewer UI.

## Promote Hierarchy

The Hierarchy Promotion is the process of promoting a version of the hierarchy as production hierarchy. Multiple versions of a hierarchy can be created and any one version can be promoted at a time. The promoted hierarchy can be sent for approval. When a hierarchy is promoted, the data of production hierarchy is archived in version staging tables and deleted from master staging tables. The data of promoted hierarchy version is then copied to master tables. The promotion of hierarchy is triggered via scheduler and the scheduler is scheduled to trigger promotion on the start date of version being promoted.

The following states are involved in the hierarchy promotion:

- **Draft:** when a version is created, the state of version hierarchy is in draft stage. Hierarchy versions can be changed in this state.
- **Pending Approval:** when a version is sent for promotion and approval is enabled for the hierarchy, the status will be in pending approval stage. Hierarchy version cannot be changed in this state.
- **Pending:** when a version is sent for promotion and approval is not enabled for the hierarchy, the status will be in pending stage. Hierarchy versions cannot be changed in this state.
- **Approved:** when a hierarchy version is approved for promotion, the status will be in approved state. Hierarchy version cannot be changed in this state.
- **Production:** when a hierarchy version is promoted to production, the status will be in production state.
- **Expired:** When a version of the Hierarchy is in Production state and the user promoted another version of the same Hierarchy to Production then the previous Production Version will be changed to Expired state.
- **Archived:** the current production hierarchy version is archived when a new version of hierarchy is promoted. Hierarchies cannot be changed in this state.

It is recommended to define the approval for a particular hierarchy before promoting any hierarchy version. This allows the admin to accept and reject the submitted version of hierarchy. Navigate to Manage Approvals UI and create an approval for hierarchy by selecting the Approval Type as “Hierarchy” and specify the hierarchy name on which the approval needs to be created as in [Figure 195](#).



Figure 195: Approval Details

Approval Details

Approval Name \*

PromoteHierarchy

Approval Description \*

PromoteHierarchy

Approval Object Type \*

Hierarchy

Hierarchy \*

Continent\_Country\_State

Approval Type \*

Serial

Action Type \*

ALL

Approval User Type \*

User Based

Available Approvers

Assigned Approvers

rdm\_admin

tasuser

tduser1

tduser2

admin

rdm\_user

UP

DOWN

☐ Advanced Settings

BACK

SAVE

On the Manage Version popup (Figure 196), enter the following:

- Select the hierarchy to promote from **Select Version** dropdown.
- Select the **Show Archived Versions** checkbox to see the archived versions.
- **Version Status:** it will be in draft state until the hierarchy is promoted.
- Select the Start and End dates for hierarchy promotion.
- Click **Promote Version**.

Figure 196: Manage Versions

**Manage Versions** ?

Hierarchy Name  
Continent\_Country\_State

Select Version  
Version1

Version Status  
DRAFT

☐ Show Archived Versions

Effective Start Date(MM/DD/YYYY) \*  
02/26/2021

Effective End Date(MM/DD/YYYY) \*  
12/31/9999

**CANCEL VIEW VERSION PROMOTE VERSION**

The hierarchy will be promoted and the message: “Hierarchy sent for Promotion” is displayed. If approval is enabled on hierarchy, hierarchy will go for approval and the message “Hierarchy sent for approval” is displayed and on the Manage Versions pop-up, the version status will be displayed as “Pending Approval”.

You can navigate to Approval Monitor to view the records or compare with base hierarchy and click Accept to approve the hierarchy. For details on Approval Monitor activities, *refer to Approval Inbox section in RDM Table Maintenance chapter.*

Figure 197: Approval Inbox

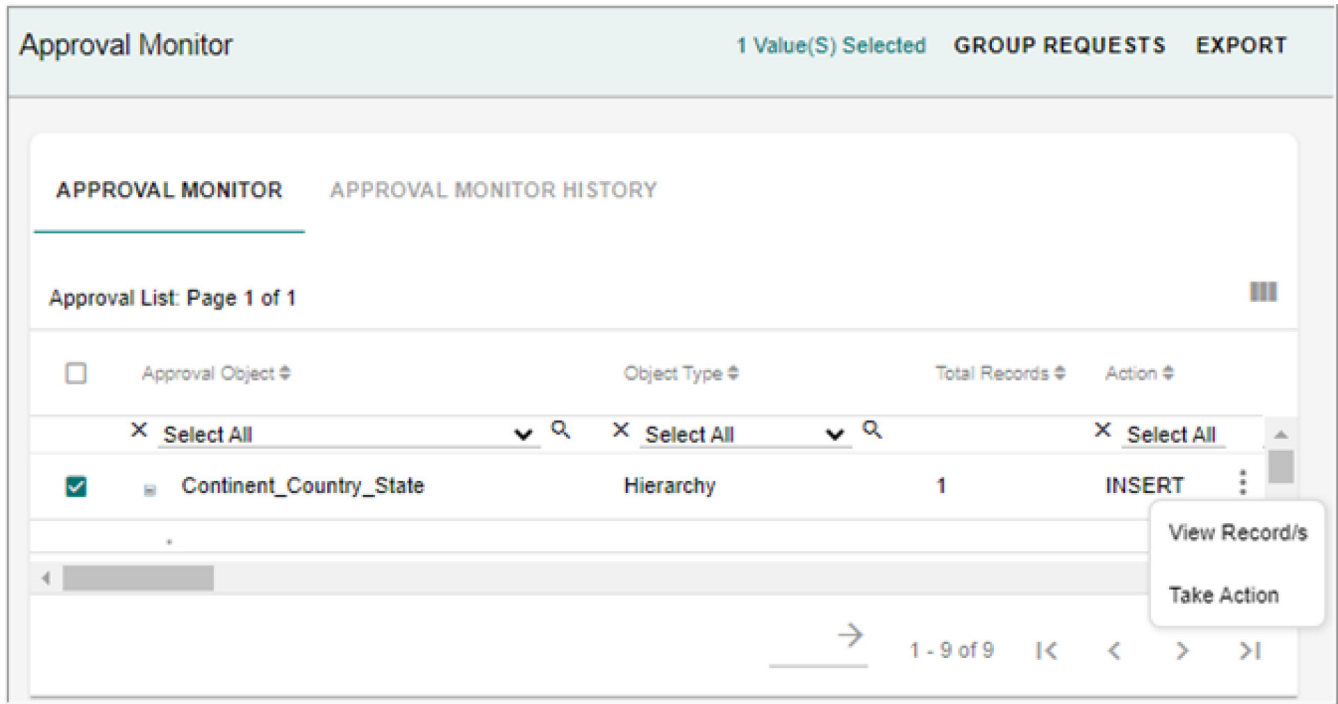
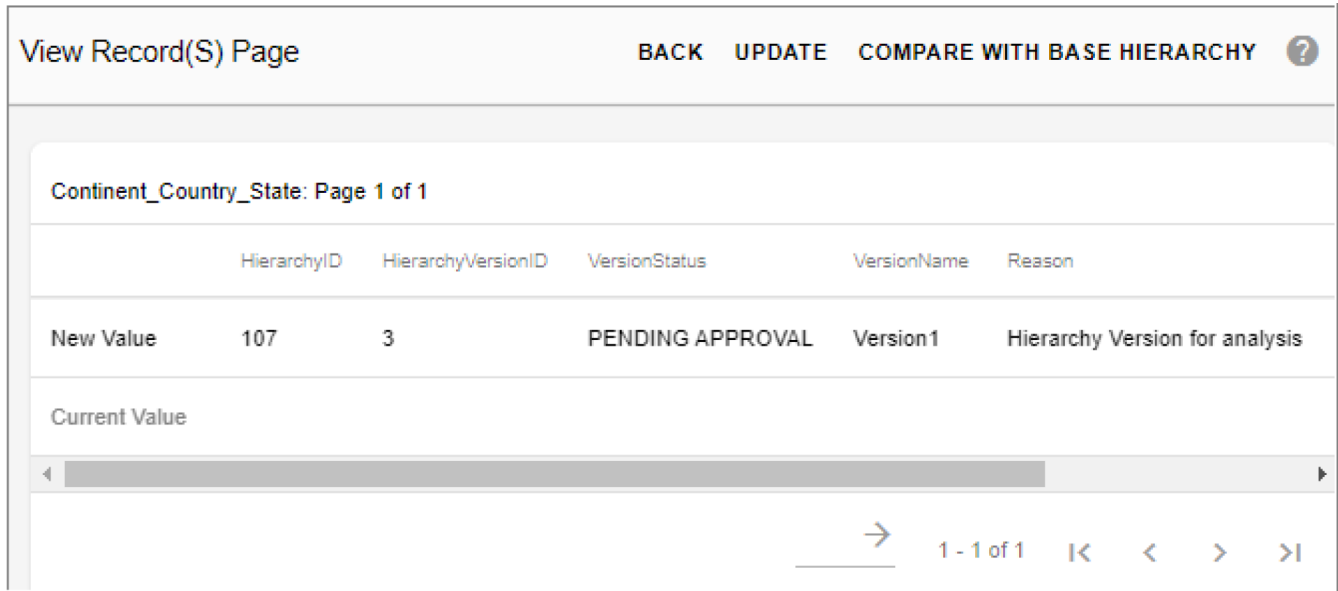


Figure 198: Approval Inbox—View Record



Once the hierarchy is approved, the Version status will be changed to Approved on the Manage Versions popup. A scheduler is automatically created to promote hierarchy as in [Figure 199](#). Once the hierarchy is promoted, it moves to production state and all the changes on version hierarchy will be reflected on the actual hierarchy.

Figure 199: Manage Scheduler

Scheduler Details

Scheduler Parameters

Name

HierarchyPromotion-111614344974342

HierarchyPromotion

Server Date and Time

27/02/2021 01:52:52

Start At (Server Time): \*

2/26/2021

00

00

00

☐ Recurring

☐ Dependent

Workflow Parameters

Hierarchy Id

101

Promoted By

USR\_1

Version Id

1

Hierarchy Name

Continent\_Country\_State

BACK

DELETE

UPDATE

**Withdraw Promoted Hierarchy**

Perform the following steps to withdraw promoted hierarchy:

- 1 On the **Manage Versions** UI, click **Withdraw Promotion** as in [Figure 196](#).



The Withdraw Promotion option is displayed only for a user assigned to a role for which the Withdraw Hierarchy Promotion Activity is enabled.

You can withdraw the promoted hierarchy version only if the version status is either Approved or Pending state.

Message: “Promotion withdrawn successfully” is displayed on the Hierarchy Manager UI.

On withdrawal of the promoted hierarchy version, the scheduler job created for promotion is deleted automatically and the version will be changed to Draft state as in [Figure 200](#) and the actual hierarchy will not be modified with any changes from version hierarchy.

Figure 200: Manage Versions

The screenshot shows the 'Manage Versions' window. It has a title bar with a question mark icon. Inside, there's a form with several fields: 'Hierarchy Name' with the value 'Continent\_Country\_State', a 'Select Version' dropdown menu showing 'Version1', a 'Version Status' field showing 'DRAFT', a checkbox for 'Show Archived Versions', an 'Effective Start Date' field with the value '02/26/2021', and an 'Effective End Date' field with the value '12/31/9999'. At the bottom of the form are three buttons: 'CANCEL', 'VIEW VERSION', and 'PROMOTE VERSION'.

## Delete Hierarchy and Hierarchy Version

To delete hierarchy:

- 1 On the **Hierarchy Manager** UI, select any hierarchy and click **Delete**.
- 2 On the **Confirmation** pop-up, click **Yes**.

Message: “Hierarchy deleted successfully” is displayed.

To delete hierarchy version:

- 1 Navigate to **Scheduler Details** UI (**Utilities** ->**Schedules** -> and click **Create Schedule**) and create a schedule using the DeleteHierarchyVersions workflow.

The hierarchy version is deleted once the scheduler is triggered.

---



You can delete the hierarchy version only if the version status is either in Draft or Expired state.

---

Figure 201: Scheduler to Delete Hierarchy Version

Scheduler Details

Scheduler Parameters

Name \*  
DeleteHierarchyversion

Select Workflow \*  
DeleteHierarchyVersions

Server Date and Time  
26/02/2021 20:40:52

Start At (Server Time): \*  
2/26/2021
00
00
00

☐ Recurring

☐ Dependent

Workflow Parameters

Hierarchy Name \*  
Continent\_Country\_State

Hierarchy Version Name \*  
Version1

Reason  
Delete Version1

BACK SAVE

## Manage Custom Actions

Custom Actions in Hierarchy Manager enables you to navigate to a custom page of your choice and perform custom actions based on the key values of the hierarchy node selected in hierarchy viewer. To create a custom action, you need to select the desired MDM page and type of browser action. Once the custom actions are defined for a hierarchy, the defined actions will be common to every node in the hierarchy.

On the **Hierarchy Manager** UI, click on the Menu icon corresponding to the required hierarchy and click **Custom**. On the **Manage Custom Actions** page, you can create, edit and delete custom actions.

Figure 202: Manage Custom Actions

Figure 203: Create Custom Actions

On the **Create Custom Action** pop-up ([Figure 203](#)),

- **Name:** unique name for the custom action.
- **Page Name:** specify the MDM page name to navigate.

The **Page Name** dropdown displays all the pages available on the **Manage Pages** UI of **Menu Builder**.

- **Browser Action:** specifies the mode of opening the custom action URL either as Popup or Inline.



If the custom action is defined with browser type as popup, the custom action link will open in a different window and if the custom action is defined with browser type as inline, the custom action link will open in the same window in configurable UI panel.

Once the custom actions are defined, you can navigate to the hierarchy viewer and on each hierarchy node of the selected hierarchy, the custom actions will be displayed as custom action links. The Manage Custom Actions option will be disabled when on the Menu if no custom actions are defined on that hierarchy.



You can create maximum three numbers of custom actions.

---

## View Hierarchy

You can view the hierarchy structure on the Hierarchy Viewer UI. On the Hierarchy Viewer page, you can also view hierarchy based on the Version as well based on Search String. You can launch the hierarchy viewer either from left navigation pane (**Reference Data -> Hierarchy Management -> Hierarchy Viewer**) or from **Hierarchy Manager** UI using **View Hierarchy** option.

For details on the tasks that can be performed on the hierarchy viewer, see [Section : “Hierarchy Viewer.”](#)

## Compare Hierarchy

Compare hierarchy feature compares the data (nodes) of two hierarchies displaying them in side by side panels. The nodes where comparison difference is found are highlighted in both the hierarchies. You can further check the details of difference between the highlighted nodes. Hierarchy comparison identifies differences based on the following criteria:

- Checks if two nodes to compare are at same level.
- Checks if the parent of node of the two nodes to compare are same.
- When the nodes at the top level matches, further comparison happens on the descendants. If the top level nodes do not match, the comparison stops at the first level itself.

A hierarchy can be compared against its version; a version of hierarchy can be compared against a different version of same hierarchy or different hierarchy. Also metadata of two hierarchies can be compared using compare hierarchy.

On **Hierarchy Manager** UI, click on the Menu icon corresponding to the required hierarchy and click **Compare**.

Figure 204: Compare Hierarchy

Compare Hierarchy

BACK COMPARE METADATA COMPARE DATA ?

\* denotes required field

Selected Hierarchy Compared Hierarchy

Selected Hierarchy SrcHierarchy

Compared Hierarchy TrgHierarchy

☐ Compare Version ☐ Compare Version

On the **Compare Hierarchy** UI, the hierarchy selected on the Hierarchy Manager UI is selected by default in the **Selected Hierarchy** dropdown. You can select a different hierarchy if required. In the **Compared Hierarchy** pane, select the hierarchy to be compared. Optionally, you can select the **Compare Version** checkbox for comparing hierarchies based on version.

On the **Compare Hierarchy**, you can:

- q Click **Compare Metadata** to compare the hierarchies metadata.

The **Compare Hierarchy** page displays the comparison results as in [Figure 205](#).

Figure 205: Hierarchy Comparison—Compare Metadata

Compare Hierarchy

[BACK](#)
[COMPARE METADATA](#)
[COMPARE DATA](#)

denotes required field

Selected Hierarchy

Selected Hierarchy

SrcHierarchy

Compared Hierarchy

Compared Hierarchy

TrgHierarchy

Compare Version

Compare Version

Comparison Result

Selected Hierarchy		Compared Hierarchy	
Level	Object	Level	Object
1	SrcCountry	1	TrgCountry
2	SrcState	2	TrgState
3	Src_City	3	Trgcity

Properties

Name	SrcHierarchy	TrgHierarchy
Created By	USR_1	USR_1
Created Date	02/26/2021 23:04:54	02/26/2021 23:05:27
Hierarchy Type	Balanced	Balanced
Hierarchy View Role	ALL	ALL
Hierarchy Edit Role	ALL	ALL

- q Click **Compare Data** to compare the hierarchies on hierarchy viewer.
- The **Hierarchy Viewer** displays both the hierarchy structures side by side as in [Figure 206](#). On the **Compare Hierarchy Viewer** UI,
- By default on loading it compares only upto two levels. You can select the number of levels to compare from dropdown on the top of the hierarchy viewer as in [Figure 206](#) and click ShowDiff icon.
  - If you scroll left hierarchy vertical panel, the right hierarchy also gets auto scrolled.

- If any node is expanded in left hierarchy panel, the corresponding node in right hierarchy panel gets expanded if the two nodes compared are same.
- Right-click on any hierarchy node, **Compare Details** option is displayed. Click on **Compare Details** option, **Compare Details** popup is displayed as in [Figure 207](#).

Figure 206: Hierarchy Comparison on Viewer



Figure 207: Compare Details

Compare Details	
Hierarchy Name: SrcHierarchy	Hierarchy Name: TrgHierarchy
Description: SrcHierarchy	Description: TrgHierarchy
Created By: USR_1	Created By: USR_1
Created Date: 2021-02-26 23:04:54.0	Created Date: 2021-02-26 23:05:27.0
View Role: ALL	View Role: ALL
Edit Role: ALL	Edit Role: ALL
Hierarchy Type: R	Hierarchy Type: R
Table Name: SrcCountry	Table Name: TrgCountry
Object Description: SrcCountry	Object Description: TrgCountry
Total Descendants: 6	Total Descendants: 5
Child Table Name: SrcState	Child Table Name: TrgState
Child Count: 2	Child Count: 2
Key Column Details: NODE_ID803	Key Column Details: NODE_ID902
Display Column Details: NODE_NAMEUSA	Display Column Details: NODE_NAMEUSA
RO URL: /mdm/start.x...	RO URL: /mdm/start.x...

Ok

## Hierarchy Viewer

The hierarchy viewer provides the ability to visually view and interact with hierarchical data. Hierarchy Viewer allows you to view the hierarchy nodes and its related data in a tree structure. The hierarchy tree structure is rendered using jstree technology. You can also modify or create node data from hierarchy viewer.

You can launch the hierarchy viewer either from left navigation pane (**Reference Data -> Hierarchy Management -> Hierarchy Viewer**) or from **Hierarchy Manager**,

If dynamic filter was applied on a particular hierarchy, then the **Dynamic Filter** popup would be displayed before the display of hierarchy viewer.

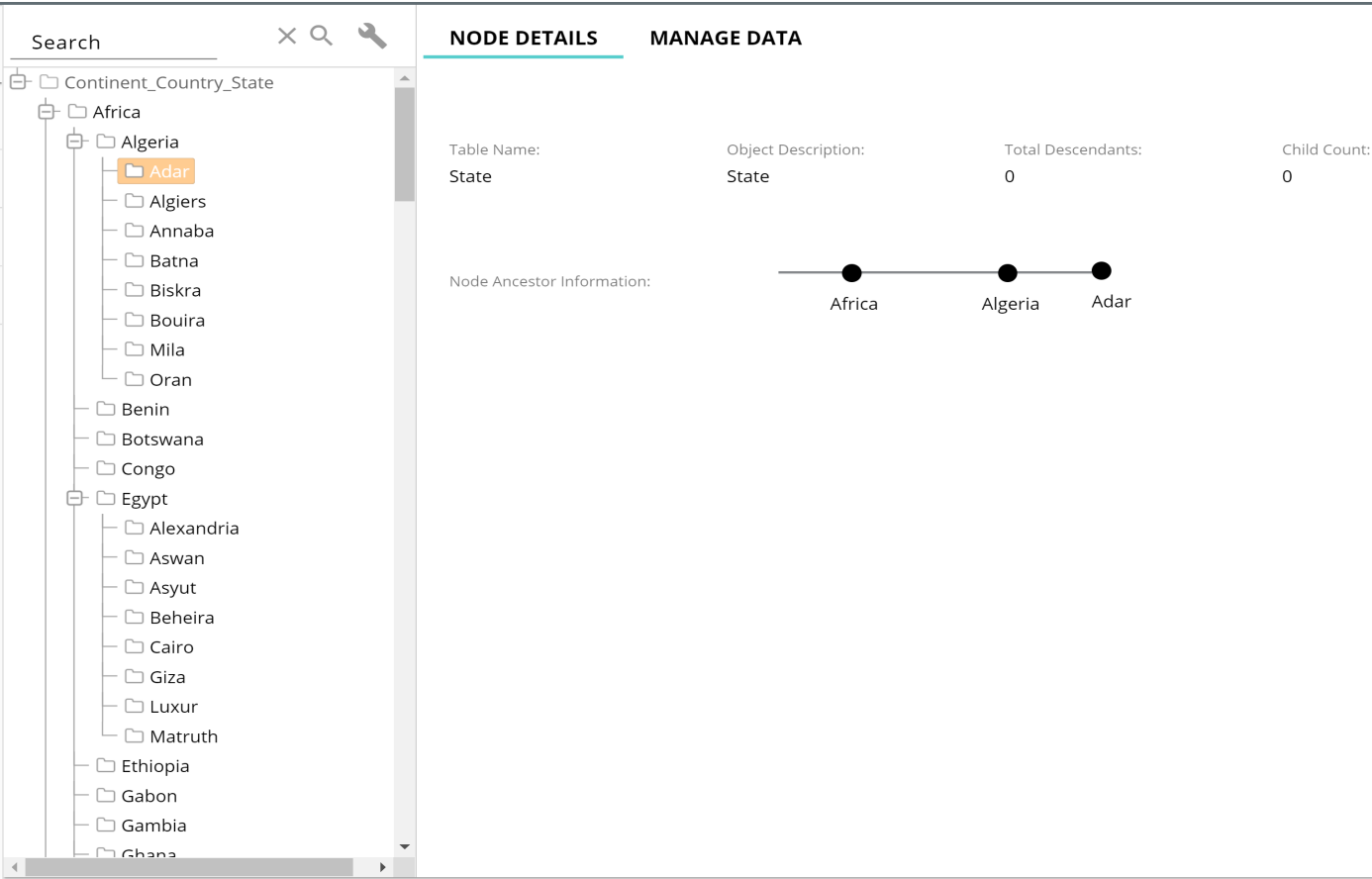
On the **Dynamic Filter** pop-up, enter the filter values and click **View**. The **Hierarchy Viewer** displays the filtered data.



When you navigate to hierarchy viewer from **Hierarchy Manager** (click on the Menu icon corresponding to the required hierarchy and click **View**), the last viewed hierarchy will be loaded by default if the user returns to hierarchy viewer page within that particular session.

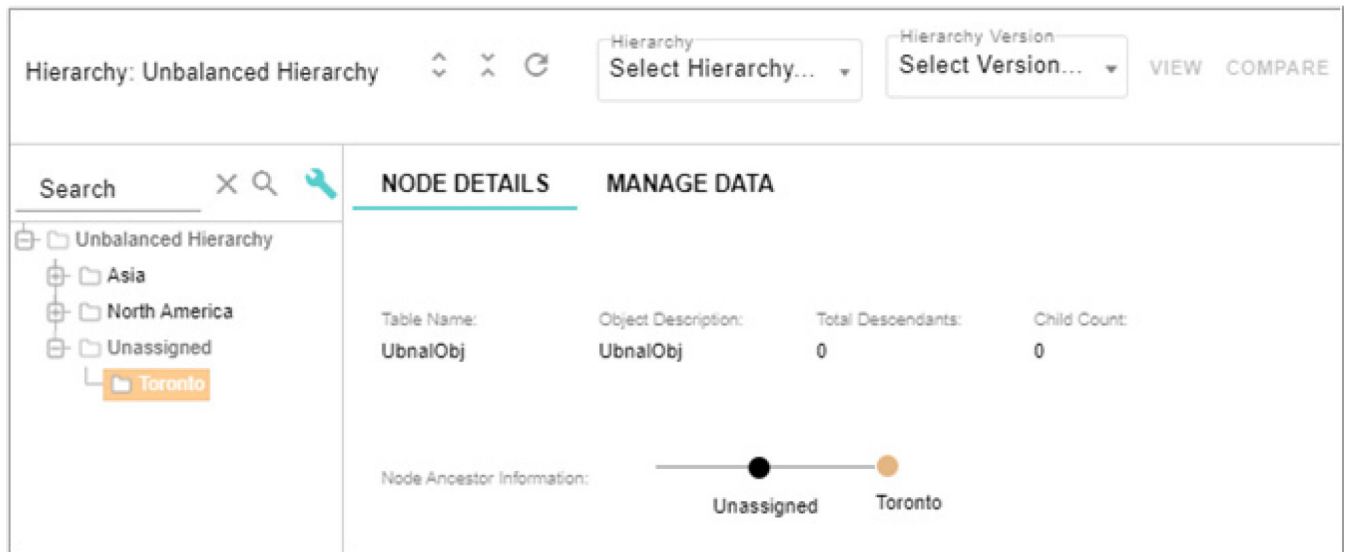
---

Figure 208: Hierarchy Viewer—Node Details



When the Hierarchy is configured to display & manage Unassigned Nodes, it will display the orphan nodes at the bottom of the hierarchy viewer tree. A dummy parent in each level of hierarchy will be assigned to all orphan nodes. The name of the dummy parent will be Unassigned\_<HierarchyObjectName>. The orphan nodes can be dragged and dropped into the right parent based on requirement.

Figure 209: Hierarchy Viewer: Unassigned Nodes



On the Hierarchy Viewer UI ([Figure 208](#)),

- **Left Panel:** the left panel is the tree view of the selected hierarchy and is rendered using jstree. The left panel displays all the node levels of the selected hierarchy. Node level 1 without child (orphan node level 1) is also displayed on the hierarchy viewer.
- **Node Details:** the right panel displays two tabs (Node Details and Manage Data). Under the Node Details tab, the node details like Table Name, Object Description, Total Descendants, Child Count are displayed. It also displays the Node Ancestor Information.
- **Manage Data:** under the Manage Data, it displays the node details in tabular form and under the Actions tab, you have options can view all peers, siblings and children of a selected node. It also provides option to add child and siblings for the selected node. You can also edit the node details using the Edit option as in [Figure 210](#).

Figure 210: Hierarchy Viewer: Manage Data

NODE DETAILS		MANAGE DATA			
	Node Id	Node Name	Description	Node External Id	Target
<input checked="" type="checkbox"/>	101	Algeria			
<input type="checkbox"/>	102	Liberiya			
<input type="checkbox"/>	103	Libiya			
<input type="checkbox"/>	104	Malawi			
<input type="checkbox"/>	105	Mali			
<input type="checkbox"/>	106	Benin			

ACTIONS

Edit Node

Show Children

Show Siblings

Show All Peers

Add Child

Add Sibling

1 - 20 of 100

Under the Manage Data tab, selecting on any option from the Actions dropdown displays the corresponding configurable UI. No UI is displayed in bottom right panel, if the corresponding configurable UI is not generated and a valid message is given to the user.



If Extended Attributes and Rollup Attributes are added to the Hierarchy Objects, ConfigUI is changed from AutoGenConfigUI to UserDefinedConfigUI.

- Show All Peers:** initial view when a node is selected in the tree view. Displays all of the records that are of the same levels without any filter.
- Show Children:** displays all child records belonging to the selected node. Show Children option is not available to leaf nodes.
- Show Siblings:** displays all records that are of the same level with the selected node and belongs to the same parent node. This is not available to the root node.
- Add Child:** allows you to add a child node to the selected parent node.
- Add Siblings:** allows you to add sibling node of the same level of the selected node.



- Add Child and Add Siblings will work only for hierarchies created on HierarchyNode and HierarchyRelation Tables. If the hierarchy is created on other master tables, Add



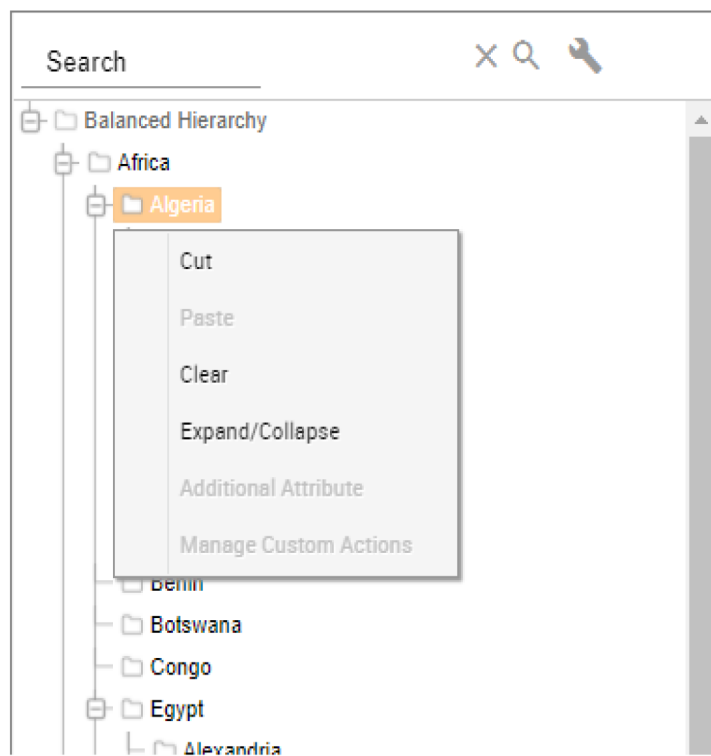
Child and Add Sibling will only add a record in the corresponding table and no relation will be added.

- Add child/sibling may not work properly if approval is enabled only on the child table. It is recommended to define approval on both node and relationship tables and approve/reject them as a group while adding approval enabled child/siblings.
- In case of Unbalanced Hierarchy with Unassigned Node display, add Sibling and Show sibling option will not be available for the Unassigned/Orphan Nodes.

**Edit:** allows you to edit the node details of the selected node.

- **Menu Options:** right click on any hierarchy node to display the menu option.

Figure 211: Hierarchy Viewer: Menu Option



The menu displays the following options:

- **Cut:** select this option to cut a particular node.
- **Paste:** select this option to paste a particular node.



You can paste the selected node at least one level up not at the same level or below level. To perform cut and paste action when hierarchy version is launched on Hierarchy Viewer, you need to set the version editable to true while creating the version.

- **Clear:** select this option to clear the contents on the clipboard.






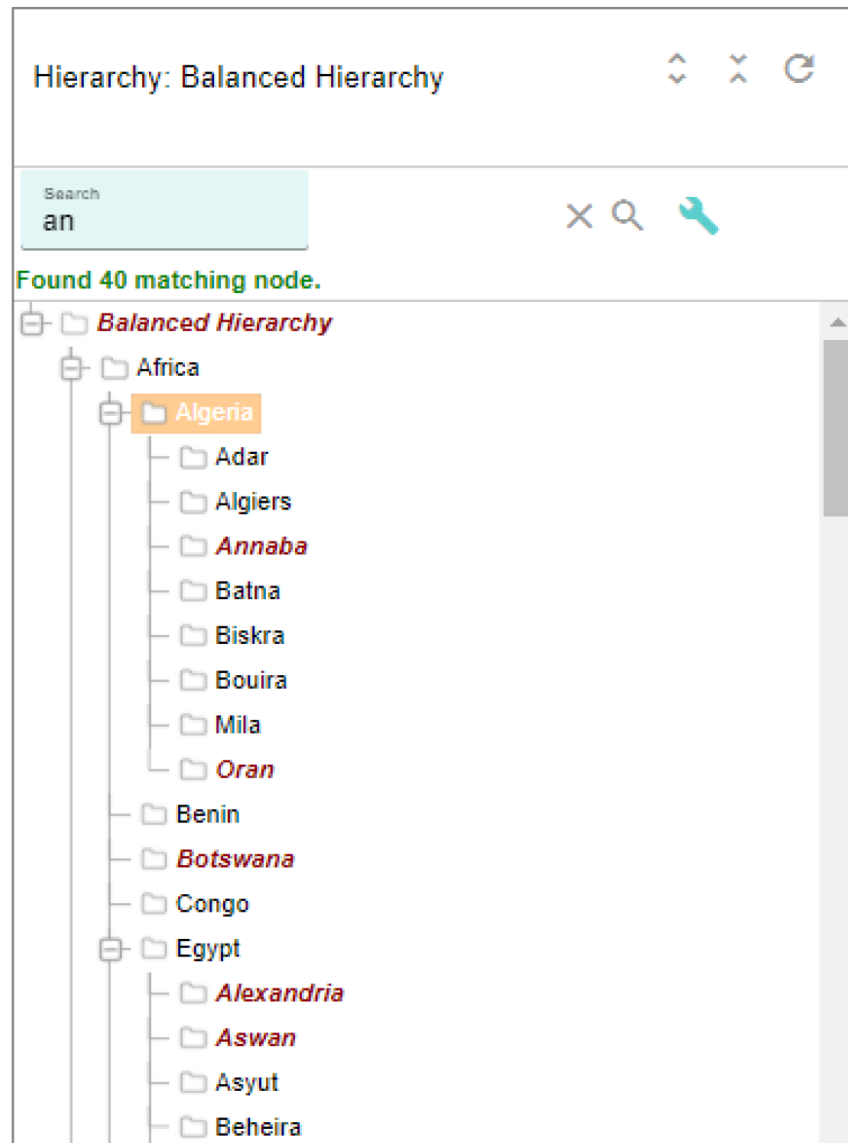
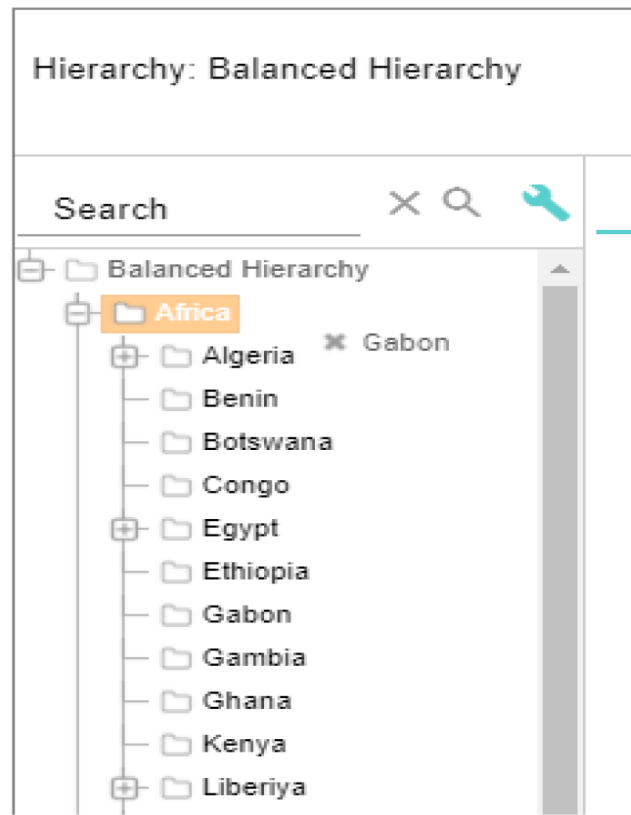
- **Expand/Collapse:** select this option to expand or collapse the tree of a particular node.
  - **Additional Attributes:** While creating a Relationship, you may choose to add additional attributes to the Relationship. You can provide values for this additional attributes while saving the Relationship on the UI. You can view these additional attributes and values from Hierarchy Viewer by selecting this option.
  - **Manage Custom Actions:** select this option to navigate to the custom defined page and perform custom actions based on the key values of the hierarchy node selected. This option will be disabled if no custom actions are defined on the selected hierarchy.
  - **Expand and Collapse Nodes:** to access the Expand and collapse hierarchy nodes icon, click icon  and then click on  icon to expand all the nodes and click  icon button to collapse all the nodes.
- q **Search Node:** you can search for a particular node on **Hierarchy Viewer** UI.
- On the **Hierarchy Viewer**, enter the key word in the text field and click  icon.  
You can do a global search either by specifying a word or a set of characters. That is you can search for a complete word or do a partial search by specifying a set of characters. For example, enter “ORE” as a search parameter in the **Search** field on the Hierarchy Viewer.  
The **Hierarchy Viewer** displays all the nodes having the character “an”. All the matched nodes will be highlighted and the total number of nodes will be displayed on the **Tree View** header line.  
You can specify the search parameter and select the level to search from the **Select Level** drop down. The search operation will search for the specified parameter only in the selected level. For structured hierarchy, the level can be RO(1), RO(2) and leaf node. For unbalanced hierarchy, the level can be leaf node or all the nodes.  
You can click  icon to remove the search string.

Figure 212: Hierarchy Viewer: Search



- **Drag and Drop:** you can drag and drop the nodes on the Hierarchy Viewer. Once you select a node and drag it, the selected node will be added to the node where you want to drop. If the node where you want to drop is not a valid position, it displays the wrong icon.

Figure 213: Hierarchy Viewer: Drag and Drop Icon



- Drag and drop honors approval or business rules defined on the underlying table. Appropriate message is displayed on the viewer for any validation triggered. The changes that go through approval are not reflected in viewer unless the approval request is approved.
- If approval is enabled on any of the hierarchy node or relation tables for the particular hierarchy, deactivate approval on these tables for the cut and paste and drag and drop feature to work properly.
- The current hierarchy viewer can display hierarchy of up to 12 levels (level 0 to level 11) for structured hierarchies.
- If a hierarchy has multiple objects defined at the same level, the Cut and Paste or Drag and Drop functionality will work only if there is a relation defined between the objects.

For example:

Consider a hierarchy with objects defined as Country, State1, State2, City1, City2 (Country, State1, State2, City1, City2 are different tables). If the relations are defined as Country-State1, Country-State2, State1-City1, State2 - City2, the Cut and Paste from City1 to City2 under State2 will not be allowed even though City1 and City2 are at the same level in hierarchy as there is no relation defined between State2 and City1 (Paste

option will be disabled in the UI). In this case, you can perform Cut and Paste or Drag and Drop from City1 records to under State1 records as there exists State1-City1 relation.

## Hierarchy Flattened View

MDM hierarchy feature is enhanced to provide flattened view of hierarchy structure for both master hierarchy and versioned hierarchy. The flattened view of hierarchy structure provides the following advantages:

- Displays the hierarchy structure data in tabular form for easy view of complete hierarchy data.
- Displays all the linked nodes and all orphan or unlinked nodes in the hierarchy. The unlinked or orphan nodes can exist at any level in the hierarchy structure.

The flattened hierarchy view is supported only for template based hierarchies and of type balanced and ragged hierarchies.

To view hierarchy in flattened structure, perform the following steps:

- 1 Navigate to **Hierarchy Flattened View** UI from left navigation pane (**Reference Data** -> **Hierarchy Management** -> **Hierarchy Manager**) and on the **Hierarchy Manager** UI, select **Hierarchy Flattened View** from **Actions** dropdown.
- 2 On the **Hierarchy Flattened View** UI, select the hierarchy from the **Hierarchy Name** dropdown and if required you can also select any version of the selected hierarchy and click **View** as in [Figure 214](#).

Figure 214: Hierarchy Flattened View

Hierarchy Flattened View

BACK CLEAR VIEW ?

\* denotes required field

Hierarchy Name \*  
Balanced Hierarchy

☐ Select Version

The hierarchy flattened view structure is displayed as in [Figure 215](#). In flattened view, you can view all the nodes (linked and unlinked nodes) available at each level. In the tree structure view, you can view only the linked nodes and not the orphan nodes.

Figure 215: Hierarchy Flattened View

Hierarchy Flattened View

BACKCLEARVIEW?

\* denotes required field

Hierarchy Name \*

Balanced Hierarchy


☐ Select Version

^ Balanced Hierarchy: Page 1 of 2

LEV1\_NODE\_NAME\_ContinentLEV2\_NODE\_NAME\_CountryLEV3\_NODE\_NAME\_State

Europe	Cyprus	
Asia	Afghanistan	Farah
Asia	India	Manipur
Asia	India	Goa

1 - 100 of 159<<<>>>

On the hierarchy flattened view structure (Figure 215), click on the Export to Excel icon  to export the hierarchy details to an excel file.

The data is exported with file name as “FlattenedView\_data\_export.xlsx”.

Export Hierarchy

On the **Hierarchy Manager** UI, click on the Menu icon corresponding to the required hierarchy and click **Export**. The Export option provides the ability to export the hierarchy data and hierarchy metadata with all its element and all child elements into Excel format. On the **Export Hierarchy** UI, you can select either **Data** or **MetaData** option as required. When **MetaData** option is selected, you can export the metadata of the selected hierarchy.

Figure 216: Export Hierarchy

Export Hierarchy

\* denotes required field

Hierarchy Name

Continent\_Country\_State

Select Option

Select..

Select..

Data

MetaData

When **Data** option is selected, the **Export Hierarchy** UI displays the option to select hierarchy version as in [Figure 217](#). Select the **Hierarchy Version** checkbox if you want to export a particular version of hierarchy. On selection of this option, all available versions of the selected hierarchy will be populated in a drop down. When hierarchy data is exported, the data of all nodes and the relationships between the nodes are exported in separate sheets of the same excel.

Figure 217: Export Hierarchy

Export Hierarchy

\* denotes required field

Hierarchy Name

Continent\_Country\_State

Select Option

Data

☒ Hierarchy Version

Select Version\*

Select...

EXPORT



- All records in parent and child table are exported during export hierarchy.
- 

## Import Hierarchy

MDM supports import of hierarchy data. When a hierarchy is selected for importing its data, it provides options to import meta data or data (nodes and relations) separately. Hierarchy data import utilizes excel upload mechanism and supports error corrections.

Navigate to **Import Hierarchy** UI from **Hierarchy Manager** UI by selecting **Import** option from **Actions** dropdown.



Figure 218: Import Hierarchy

Import Hierarchies

☐ Meta Data

☒ Data

Hierarchy\*  
Continent\_Country\_State

☐ Master ☒ Version

☒ Nodes ☐ Relations

Hierarchy Nodes\*  
Select...

Hierarchy Versions\*  
Select...

File\*  
Choose File No file chosen

UPLOAD

On the **Import Hierarchy** UI (Figure 218):

- **Meta Data:** select **Meta Data** option to import the hierarchy metadata. Select **Meta Data** option, choose the file to import and click **Upload**.
- **Data:** select the **Data** option to import the hierarchy data. When **Data** option is selected, the **Hierarchy** drop down is displayed.
- **Hierarchy:** select the hierarchy whose data needs to be imported from **Hierarchy** dropdown. When a hierarchy is selected, it displays options to import:
  - **Master** hierarchy: select the **Master** option and the corresponding nodes/relations to import.

- **Version hierarchy:** when **Version** option is selected, the **Hierarchy Versions** dropdown is displayed. You can select the required version and the corresponding nodes/relations to import.
- **Nodes:** select **Nodes** option to import node details. When **Nodes** option is selected, **Hierarchy Nodes** drop down displays all the existing nodes (Hierarchy Objects) of the selected hierarchy. Select the required node to import from the **Hierarchy Nodes** dropdown.
- **Relations:** select **Relations** option to import relation details. When **Relations** option is selected, **Hierarchy Relations** dropdown displays all the existing relationships of the selected hierarchy. Select the required relationship to import from the **Hierarchy Relations** dropdown.



Nodes should be imported before relations as validation is done on the relation table to check if the node names exist in the corresponding node tables.

The Import Hierarchy is not supported in custom table based hierarchy.

---

- Click **Choose File** to locate the corresponding excel file and click **Upload**.

The **View Reports** page is displayed. If any error occurs, you can perform error correction and reload data.

#### Download Document Template for Import

The Document Template feature available on **Import Hierarchies** UI allows you to download the template in excel for each import type like Meta Data, Data (Nodes and Relations). On the **Import Hierarchies** UI, select the import type (meta-data or nodes or relations) and select the corresponding data and click on the Document Template icon as in [Figure 219](#).

Figure 219: Import Hierarchies—Document Template

Import Hierarchies

☐ Meta Data

☒ Data

Hierarchy\*  
Country\_State

☒ Master ☐ Version

☐ Nodes ☒ Relations

Hierarchy Relations\*  
Country\_State

File\*  
Choose File No file chosen

UPLOAD

Open the downloaded template file as in [Figure 220](#) and [Figure 221](#). You can enter the required details in the excel file and later upload using the Upload Excel option.

Figure 220: Template File—Node Template

	A	B	C	D	E	F	G	H
2	Node Id	Node Name	Node External I	Description	EXT_ATTR	EXT_ATTR	EXT_ATTR	EXT_ATTR
3								
4								
5								

In the Node Template file (Figure 220), Node Id is optional. The upload of node expects node names as mandatory inputs. If nodes with same name exists it will be considered as update otherwise it will be insert. In case user provides explicit node Ids in the upload excel template, user defined Ids will be considered. In that case, it is user’s responsibility to maintain uniqueness of Ids.

In the Relation Template file (Figure 221), node name and parent node name will be taken as input from user.

Figure 221: Template File—Relation Template

	A	B	C	D	E	F	G
2	Child Node Name	Parent Node Name	EXT_ATTR	EXT_ATTR	EXT_ATTR	EXT_ATTR	EXT_ATTR
3							
4							
5							

Points to remember on import hierarchy process:

- Validation Rules, Cleansing and Standardization Rules defined on master tables are honored on version tables also.

- Approval, ALCR and Insert Update Rules are applicable only on master tables and not supported on version tables.
- Error correction is supported on both master and version tables.

## Merge Hierarchy

Using the Merge Hierarchy feature, you can merge the changes from two different hierarchies or two different versions of hierarchies. The main features of hierarchy merging includes the following:

- The changes of source hierarchy can be merged into a copy of the target hierarchy or directly into the target hierarchy.
- The process of hierarchy merging is done through Do DB Persist framework.
- In merge hierarchy process, each node is uniquely identified using the NodeName. Node Id and NodeName will not be updated.
- Node name must be unique at each level.
- Optionally, validation rules defined on objects and relationships of hierarchy can be honored.
- Hierarchy merging is currently supported for Balanced Hierarchies.
- When a parent is selected for merge, all its descendants are considered for merging.
- The Merge Hierarchy is not supported on custom table based hierarchy.
- Removing node will remove node from Relation as well as from Node table.
- Deleting or adding new level is not supported through Merge hierarchy process.

Navigate to **Merge Hierarchy** UI from **Hierarchy Manager** UI using **Merge** option from the **Actions** dropdown.

On the **Merge Hierarchy** UI, perform the following steps:

- 1 Select the source and target hierarchy to be merged from **Source Hierarchy** and **Target Hierarchy** dropdowns as in [Figure 222](#).
- 2 Select the **Version** checkbox of source or target hierarchy, if you want to merge any version hierarchy.

Figure 222: Merge Hierarchy

denotes required field

Source Hierarchy\*  
SrcHierarchy

Version

Target Hierarchy\*  
TrgHierarchy

Version

**SELECT LEVEL**

Source Node Name

Target Node Name

☐ Create a Copy

☐ Skip Validation Rules

☐ Merge Hierarchy Nodes, Relations

**MERGE**

- 3 Click **Select Level** to select the nodes to be merged.

The **Select Level** pop-up is displayed with the selected source and target hierarchies side by side.

- 4 On the **Select Level** pop-up,
  - From the source hierarchy, select the node which has to be used for merging.
  - From the target hierarchy, select the node which has to be merged with that of the source.
  - Click **Select**.

The nodes selected to be merged need not be in same level in the source and target hierarchy.

The **Merge Hierarchy** UI is displayed again.

Figure 223: Merge Hierarchy

Merge Hierarchy

\* denotes required field

Source Hierarchy\*  
SrcHierarchy

☐ Version

Target Hierarchy\*  
TrgHierarchy

☐ Version

SELECT LEVEL

Source Node Name  
USA

Target Node Name  
USA

☐ Create a Copy

☐ Skip Validation Rules

The **Source Node Name** and **Target Node Name** fields would be populated with the selected node names as in [Figure 223](#).

The **Merge Hierarchy** UI also displays the following options as in [Figure 224](#):

- **Create Copy:** select the Create Copy option to merge the source hierarchy with a new version of target hierarchy. The original target hierarchy will not be updated, only the version target hierarchy will have the changes of merging.
- **Skip Validation Rules:** select the Skip Validation Rules option to skip the validation the underlying node and relationship tables used in the merging process.

Figure 224: Merge Hierarchy

Merge Hierarchy
?

\* denotes required field

Source Hierarchy \*  
SrcHierarchy

☐ Version

Target Hierarchy \*  
TrgHierarchy

☐ Version

**SELECT LEVEL**

Source Node Name  
USA

Target Node Name  
USA

☐ Create a Copy
☐ Skip Validation Rules

☒ Merge Hierarchy Nodes, Relations

Delete

☒ Nodes
☒ Relations

☒ Update Properties

**Node Properties**

<input type="checkbox"/>	Target Level	Property
<input checked="" type="checkbox"/>	TrgCountry	EXT_ATT1

**Relational Properties: No Records Found**

<input type="checkbox"/>	Target Level	Property
--------------------------	--------------	----------

MERGE

- Merge Hierarchy Nodes, Relations:** select the Merge Hierarchy Nodes, Relations option to update the target hierarchy structure to be in sync with source hierarchy by performing Inserts or Deletes or Updates on the data in the target hierarchy in comparison to source hierarchy. The updates or inserts or deletes will be performed based on the following criteria:
  - If the source hierarchy has records that are not available in the target hierarchy, insert operation is performed on the target hierarchy.
  - If the target hierarchy have records that are not available in the source hierarchy, by default, the delete of nodes and relations option is selected and delete operation is performed in the target hierarchy to delete the records. If you do not want to



perform delete operation, deselect the delete option to either delete both the nodes and relations or only the relations.

- If the target hierarchy have records that have different parent in the source hierarchy, the parent of the node in the target hierarchy is updated.

The following use cases are supported in merge hierarchy process:

- Case 1: if the records are available in node table and relation table of both source and target hierarchy, update properties action is performed to update node and relation table properties in target hierarchy.
- Case 2: if the records are available in target node table, source node table and source relation table and have no records in target relation table, then update action is performed on target node table and insert action is performed on target relation table to populate the target relation table with the data that is available in source relation table.
- Case 3: if the records are available in target node table, target relation table, source node table and have no records in source relation table, then update action is performed on target node table and delete action is performed on target relation table to delete in the target relation table that is not available in the source relation table. By default, the delete action is performed. Deselect the delete option to either delete both the nodes and relations or only the relations.
- Case 4: if no records are available in target node and target relation table and have records in source node and relation table, then insert action is performed to insert the data in target node and relation table with the data that is available in source node and relation table.
- Case 5: if records are available in target node and target relation table and have no records in source node and relation table, then delete action is performed to delete the data in target node and relation table that are not available in source node and relation table. By default, the delete action is performed. Deselect the delete option to either delete both the nodes and relations or only the relations.
- **Update Properties:** select the Update Properties option to update the nodes and relation properties. The **Node Properties** and **Relational Properties** panel is displayed. Select the node properties and relation properties that have to be merged.
  - **Node Properties:** the Node Properties pane displays all the properties (extended attributes) of the node tables that are common to both the hierarchies, starting from the nodes that were initially selected.
  - **Relation Properties:** the Relation Properties pane displays all the properties (extended attributes) of the relation tables that are common to both the hierarchies.

## 5 Click **Merge**.

Message: “Hierarchy merged successfully. Check email for details” is displayed.

An example of hierarchy merging: The [Figure 225](#) displays the nodes before Merge Hierarchy process.

Case1: The source hierarchy has records that are not available in the target hierarchy, insert operation is performed on the target hierarchy. In [Figure 226](#), New York and its child nodes will be inserted in target hierarchy.

Case 2: The target hierarchy have records that are not available in the source hierarchy, delete operation is performed in the target hierarchy to delete the records. In [Figure 226](#), Texas and its child nodes will be removed from target hierarchy if Delete option is enabled. If delete option is disabled, then the target hierarchy records that are not available in the source hierarchy are still retained in target hierarchy as in [Figure 227](#).

Case 3: The target hierarchy have records that have same parent and different child in the source hierarchy, the child node in the target hierarchy is updated. In [Figure 226](#) (under USA -> California, Los Angeles is removed if delete is enabled and San Francisco and Sacramento are added). In [Figure 227](#) (under USA -> California, Los Angeles is retained if delete is disabled and San Francisco and Sacramento are added).

The [Figure 226](#) displays the hierarchy structure after merge hierarchy process with Delete option enabled. [Figure 227](#) displays the hierarchy structure after merge hierarchy process with Delete option disabled.

Figure 225: Example—Hierarchy Structure Before Merge Process

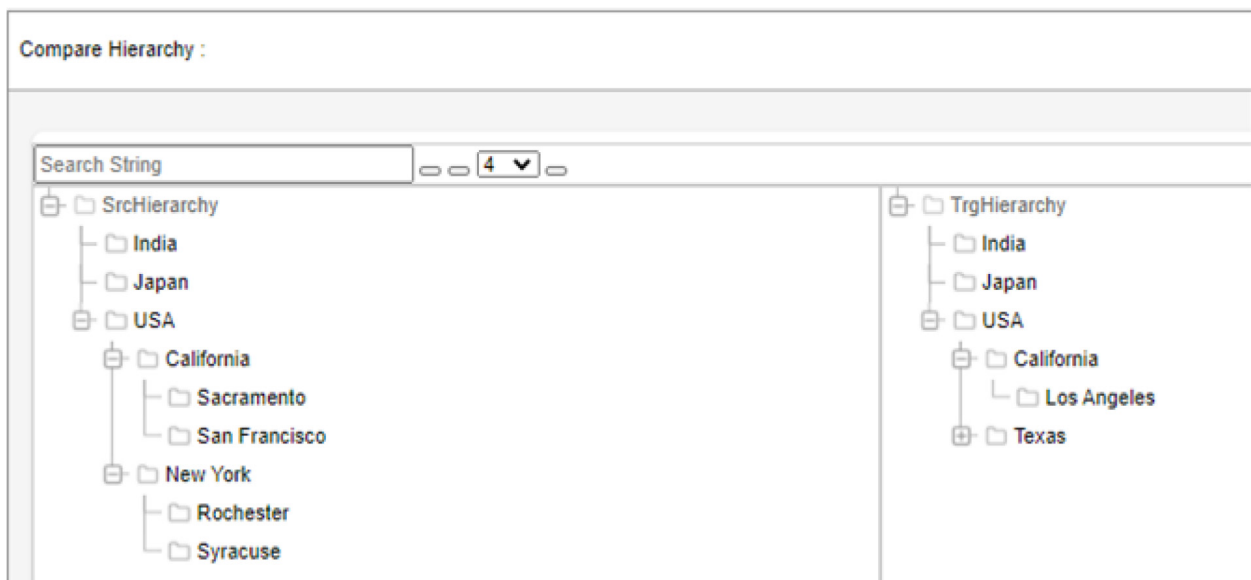


Figure 226: Example—Hierarchy Structure After Merge Process with Delete Enabled

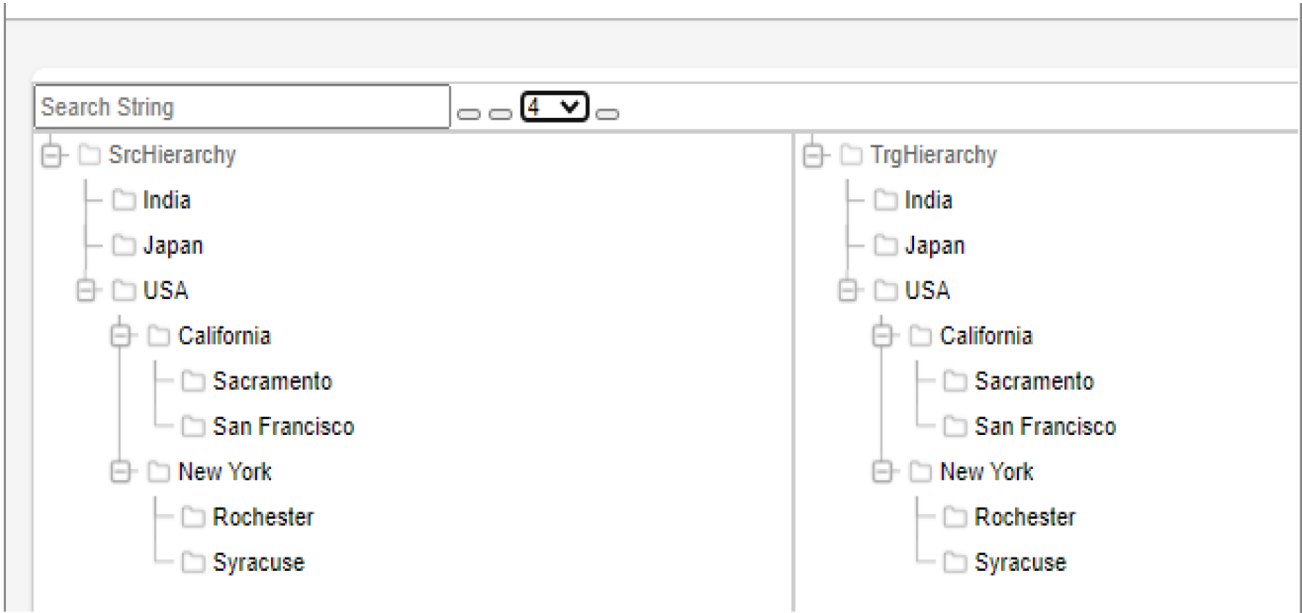
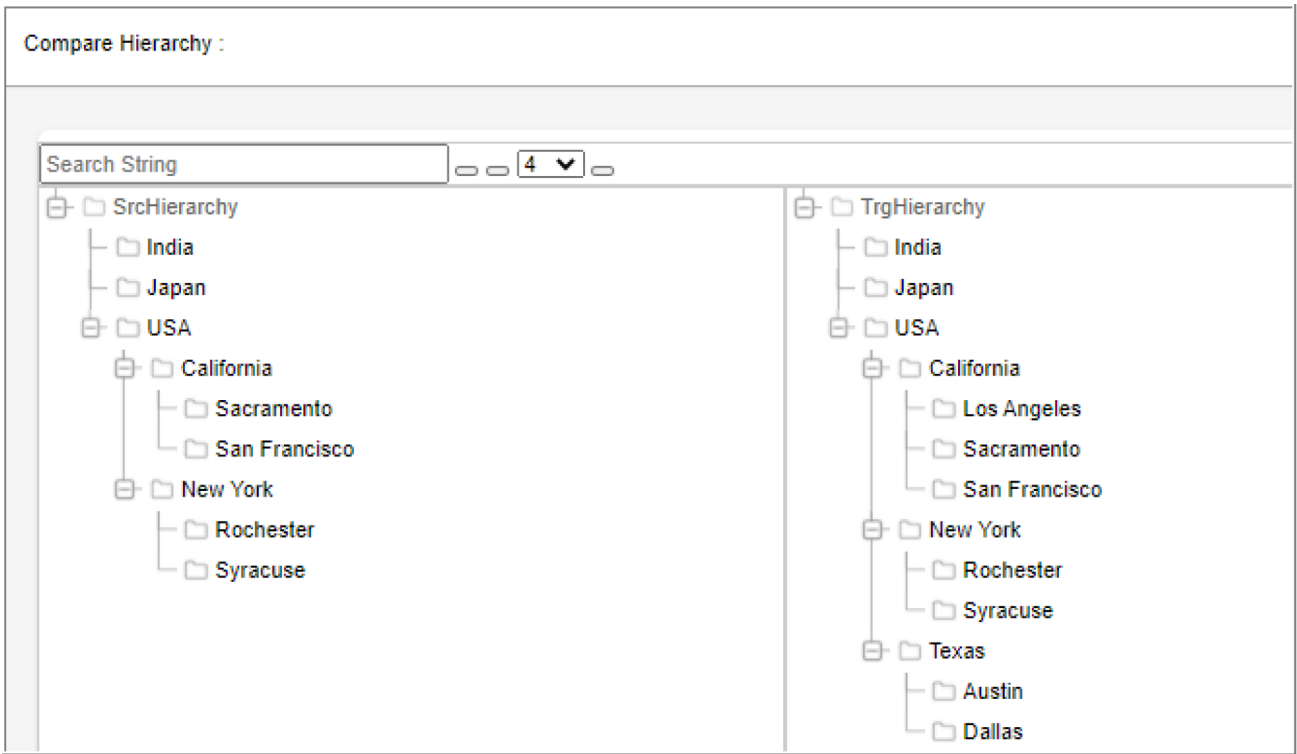
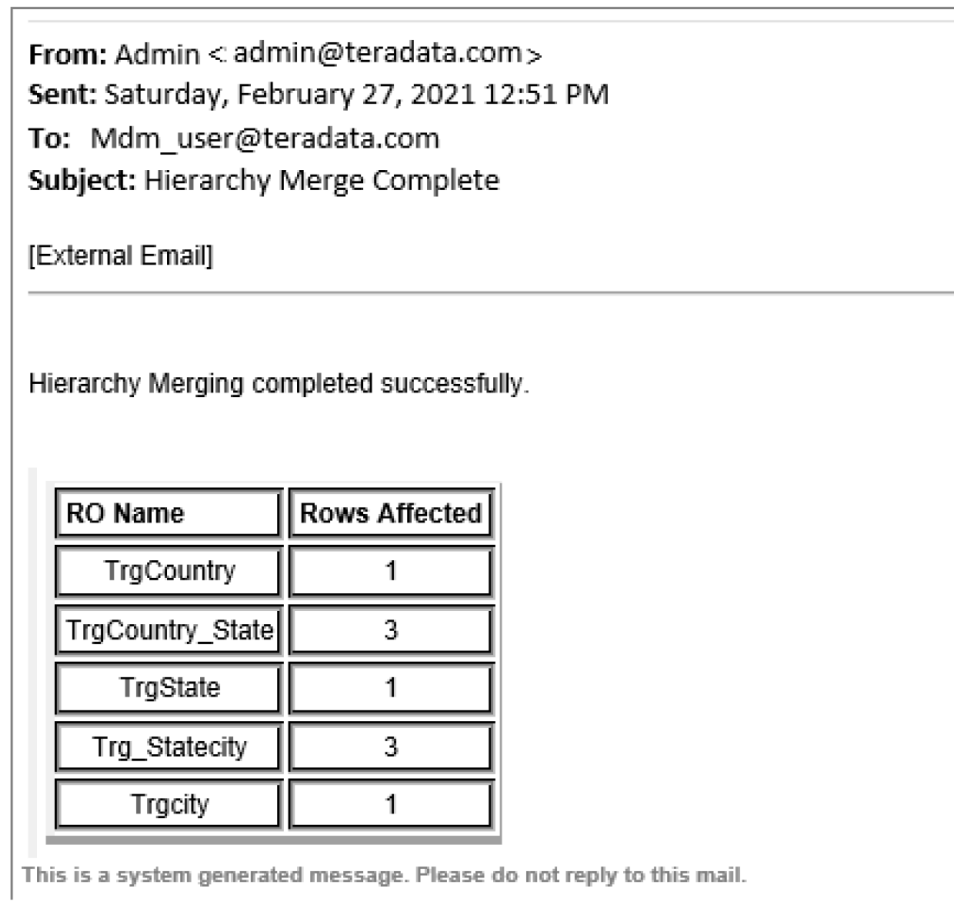


Figure 227: Example—Hierarchy Structure After Merge Process without Delete Enabled



After the merging process is complete, an email is sent to the application user who performs the operation. The email contains summary of merging process listing the number of rows impacted per table as in [Figure 228](#).

Figure 228: Email Notification



## Hierarchy Logs

The hierarchy manager module logs will be captured in log files of MDM\_HM, BCM\_MASTER and UI\_WORKFLOW services based on the type of request fired. The log levels will be as set in the respective services.

## Manage Rollup Attributes

The Rollup attributes are the attributes assigned to a particular hierarchy. Roll up attributes are defined independent of hierarchy creation. On the Manage Rollup Attributes, you can define global attributes and associate them to a particular hierarchy. When rollup attributes are assigned to hierarchies, the system auto generates a rollup view that contains aggregated value created by the user. It displays values calculated by applying functions against transaction data available in the warehouse.

Once an attribute is assigned to a hierarchy, it will be applicable for each node within the hierarchy. When a roll-up attribute is associated with a Hierarchy, a view is created to display the Node ID and the corresponding rollup attribute value of every node in the Hierarchy. The configurable UIs for base and version of the hierarchy are changed to add join condition to display the value of the rollup attribute.

Example: Sales data for a city. The transactions would presumably be for nodes at leaf levels. When clicking on parent levels, these values would be calculated for those levels as an aggregate of all their children.

Navigate to **Manage Rollup Attributes** UI from **Reference Data -> Hierarchy Management -> Hierarchy Manager** and select **Rollup Attributes**.



The Manage Roll-up Attributes feature is not supported for custom table based hierarchy. The following grant accesses have to be given for the rollup attribute assignment to work.

In case of topology,

```
grant all on CBUSR1_HM to EXT_SRC_09 with grant option;
grant all on EXT_SRC_09 to CBUSR1_HM with grant option;
grant all on CBUSR1 to EXT_SRC_09 with grant option;
grant all on EXT_SRC_09 to CBUSR1 with grant option;
grant all on CBUSR1 to CBUSR1_HM with grant option;
grant all on CBUSR1_HM to CBUSR1 with grant option;
```

Here, CBUSR1\_HM is the database on which the rollup view is created, that is, the name of the database assigned for the MDM\_HM service.

EXT\_SRC\_09 is the name of the external database, whose table (NonMDM) is joined with the Hierarchy table to create the rollup.

CBUSR1 is the name of the main database.

In case of non-topology,

```
grant all on CBUSR1 to EXT_SRC_09 with grant option;
grant all on EXT_SRC_09 to CBUSR1 with grant option;
```

---

You can perform the following tasks on the Manage Rollup Attributes UI:

- [Create Hierarchy Attribute](#)
- [Edit Hierarchy Attribute](#)
- [Delete Hierarchy Attribute](#)
- [Attach Rollup Attribute to Hierarchy](#)

## Create Hierarchy Attribute

To create attribute:

- 1 On the **Manage Rollup Attributes** UI, click **Create** as in [Figure 229](#).

Figure 229: Manage Rollup Attributes

The screenshot shows the 'Manage Rollup Attributes' interface. At the top right, there is a green 'CREATE' button, a heart icon, and a question mark icon. Below the header, a message states 'Search Results: No Records Found'. A table with four columns is visible: 'Attribute Name', 'Attribute Display Name', 'Attribute Type', and 'Attribute Data Type'. Each column has a search icon and a clear 'x' button. The table is currently empty, and a pagination bar at the bottom indicates '1 - 0 of 0' records.

- 2 On the **Add Hierarchy Attributes** UI ([Figure 230](#)), enter the values as in below table:

Figure 230: Add Hierarchy Attributes

The screenshot shows the 'Add Hierarchy Attributes' form. A legend indicates that an asterisk (\*) denotes a required field. The form contains the following fields:

Field Name	Value
Attribute Name *	CreditScoreSum
Attribute Display Name *	CreditScoreSum
RollUp Function *	SUM
Transaction Table *	SMKTST_SRC.EXT_CUSTOMER
Transaction Join Property *	CUST_ID
Transaction Measure Property *	CUST_CREDIT_SCORE
Description	

At the bottom right of the form, there are 'BACK' and 'SAVE' buttons.

Table 10: Hierarchy Rollup Attributes

Column	Purpose
Attribute Name	A required field; limited to 15 characters; recommended not to use spaces as the spaces will be deleted in the name when saved. The reason why spaces are deleted is because the Attribute Name is used to dynamically create the view for this particular rollup.
Attribute Display Name	A required field; limited to 30 characters; may contain spaces.
RollUp Function	List of available types of aggregates such as MIN, MAX, AVG, and SUM.
Transaction Table	List down tables/views that is stored in the sandbox database. Tables that contain transaction records.
Transaction Join Property	List of columns of the selected Transaction Table; column that will be used in the join condition when we associate this attribute in a particular hierarchy. Should contain NodeId of the Leaf Node Level.
Description	Provide short description for the attribute. Limited to 200 characters.

### 3 Click **Save**.

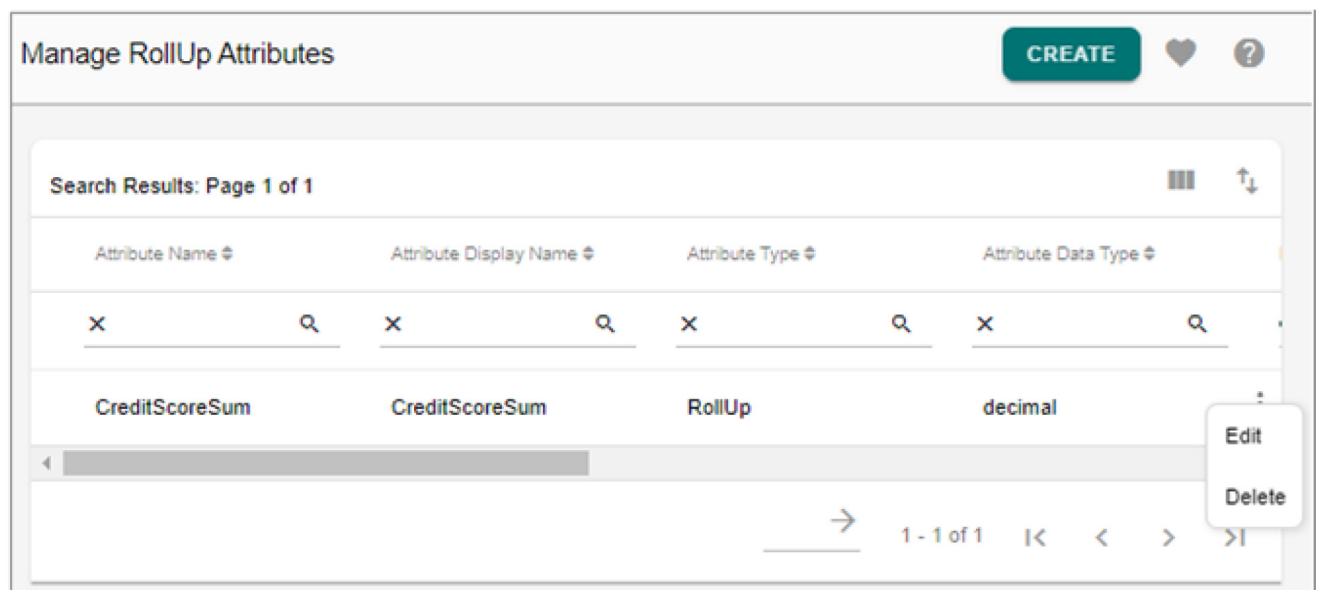
On the **Add Hierarchy Attributes** UI, the above entered data will be stored in the HierarchyExtAttributes table which serves as a pool for all hierarchy attributes.

## Edit Hierarchy Attribute

To edit hierarchy attribute details:

- On the **Manage Rollup Attributes** UI (Figure 231), click on the Menu icon corresponding to the attribute and click **Edit**.

Figure 231: Manage Hierarchy Attributes



- On the **Edit Hierarchy Attributes** UI (Figure 232), modify the required values and click **Save**.

Figure 232: Edit Hierarchy Attribute

?

\*

denotes required field

Id

111614857999667

Attribute Name \*

CreditScoreSum

Attribute Display Name \*

CreditScoreSum

RollUp Function \*

SUM

Transaction Table \*

SMKTST\_SRC.EXT\_CUSTOMER

Transaction Join Property \*

CUST\_ID

Transaction Measure Property \*

CUST\_CREDIT\_SCORE

Description

BACK

SAVE

## Delete Hierarchy Attribute

To delete hierarchy attribute details:

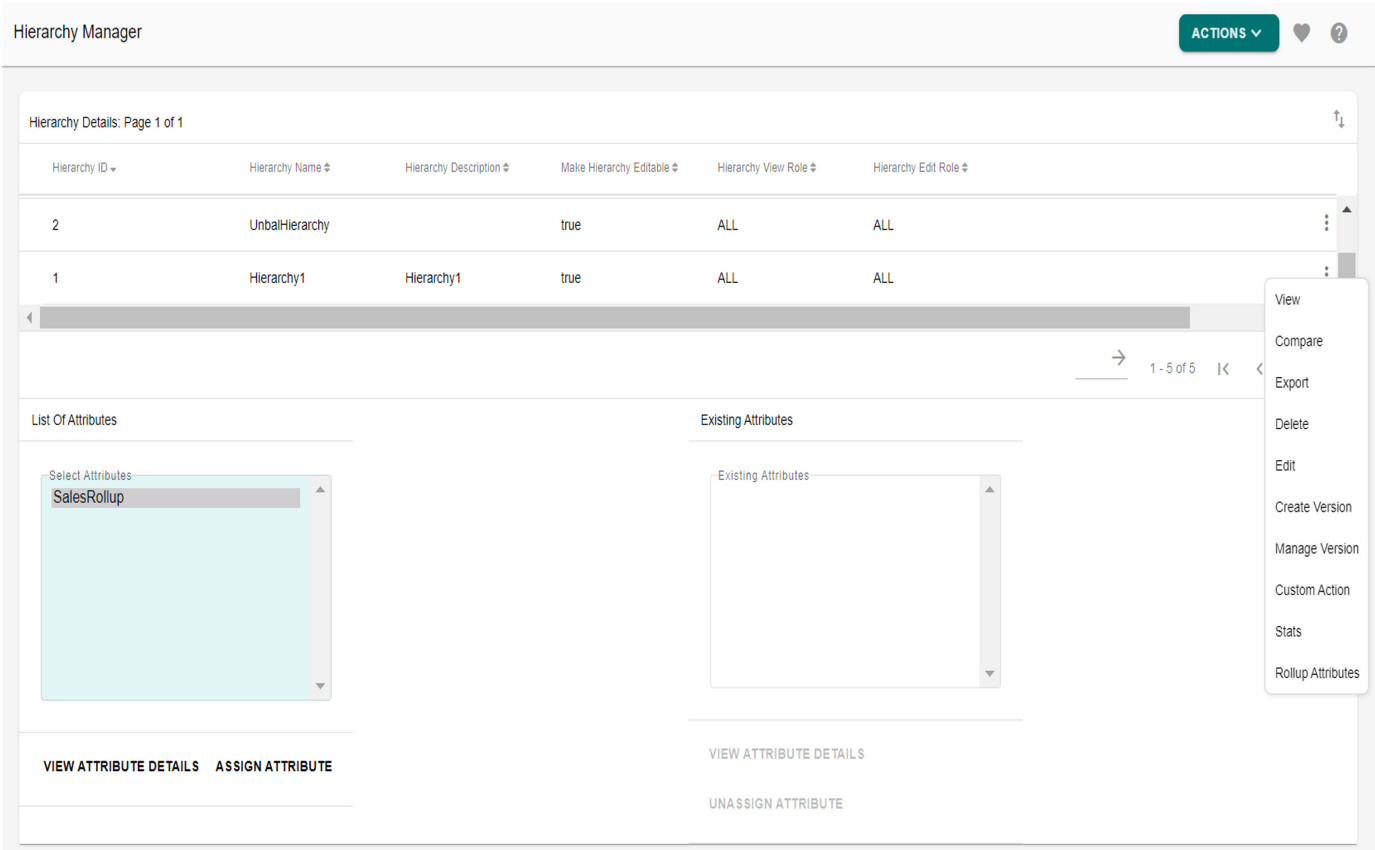
- On the **Manage Hierarchy Attributes** UI (Figure 231), click on the Menu icon corresponding to the required attribute and click **Delete**.
- On the Confirmation popup, click **Yes**.

## Attach Rollup Attribute to Hierarchy

Once the attribute is created, you can associate the attribute to any particular hierarchy from the **Hierarchy Manager** UI. On the **Hierarchy Manager** UI, select the hierarchy and click **Manage Rollup Attributes**.



Figure 233: Hierarchy Manager: Rollup Attributes



On the **Hierarchy Manager** UI,

- The **List of Attributes** pane displays all the available attributes that can be assigned to the selected hierarchy.
- The **Existing Attributes** pane displays the assigned attributes.
- Using the **Assign Attributes** button, you can assign the attributes to the selected hierarchy.
- Before assigning the attribute to the selected hierarchy, using the **View Attribute Details** button, you can view the attribute details as in [Figure 233](#).

Figure 234: Hierarchy Attributes Details

Hierarchy Attributes Details	
Attribute Name SalesRollup	Attribute Display Name SalesRollup
Attribute Type RollUp	Transaction Table EXT_DB.CITY_SALES
Transaction Join Property CITY_ID	Transaction Measure Property CITY_SALE
Transaction Rollup Function SUM	Description
<b>BACK</b>	

When the Assign Attribute button is clicked, an API is called internally that formulates a dynamic query and automatically creates the Rollup View stored in the HM database. The name of the view is in format “V\_HIER\_(hierarchyID)\_(AttributeName)\_RLP”.

Example: V\_HIER\_1\_SalesRollup\_RLP- this view contains two columns (NODE\_ID and SalesRollup). This sums up the sales of each nodes defined in the hierarchy.

After creating the rollup view, an entry gets inserted in the HIER\_EXT\_ATTRB\_MAP table that stores the Hierarchy Name, Attribute Name, Rollup View Name and the database name of the rollup view. Once this entry gets inserted successfully, you will be notified that the selected attribute is successfully assigned to the selected hierarchy.

- Using the **Unassign Attributes** button, you can unassign the attributes for the selected hierarchy.

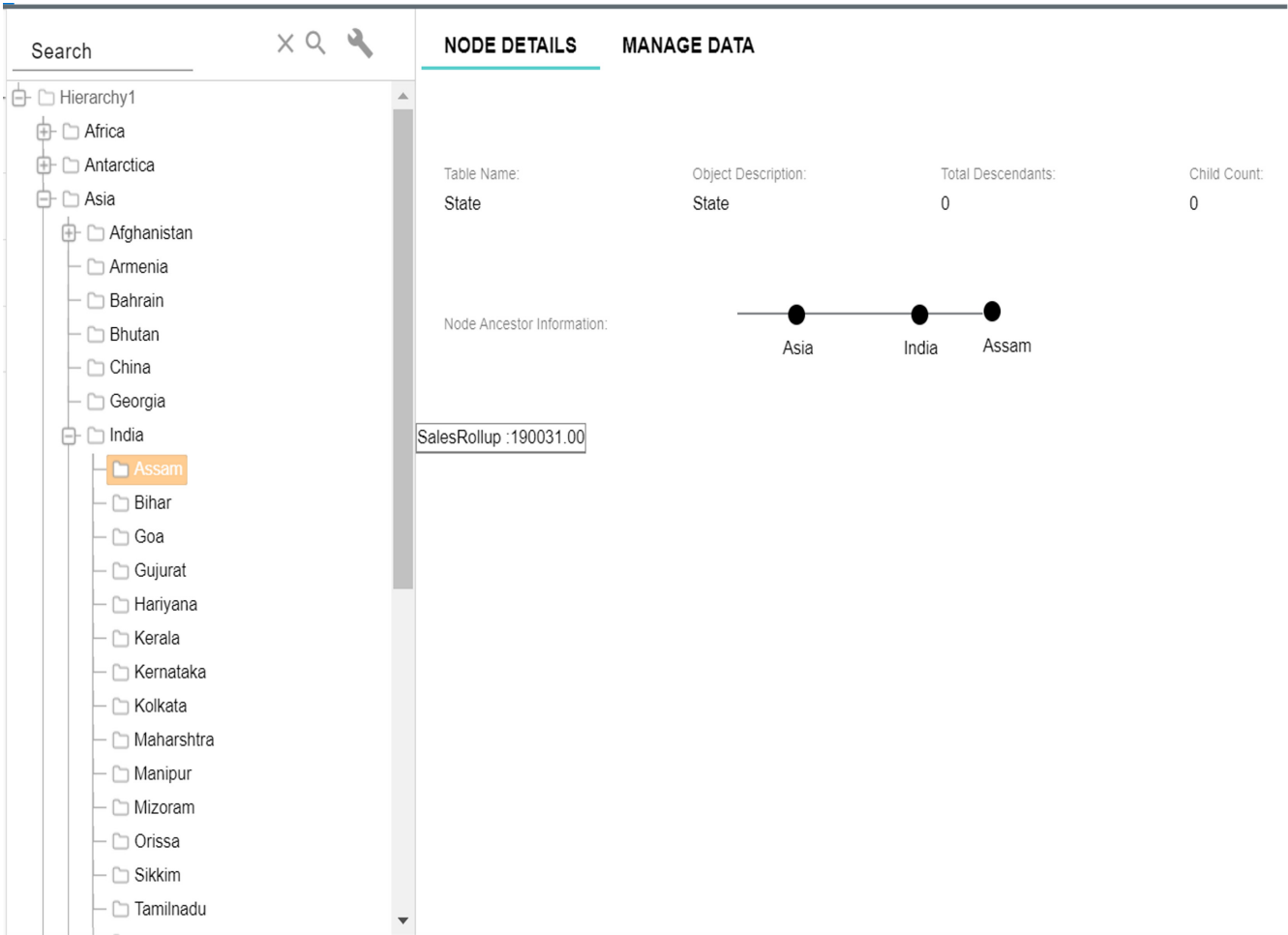
If you chooses to unassign an attribute, the rollup view created will be deleted from the database and also deletes the record in the MST\_HIER\_EXT\_ATTRB\_MAP given the Hierarchy Name and Attribute Name.

### ***View Hierarchies with Rollup Attributes***

Once the rollup attribute is assigned to a hierarchy, the assigned attribute will be added to all the nodes of the selected hierarchy and can be viewed on the **Hierarchy Viewer** as in [Figure 235](#).

If a rollup attribute is added to a hierarchy, the underlying configurable UI will have the rollup attribute added. If the objects of that hierarchy are reused in other hierarchies, the rollup attribute will be displayed in other hierarchies as well.

Figure 235: Hierarchy Viewer—Rollup Attribute



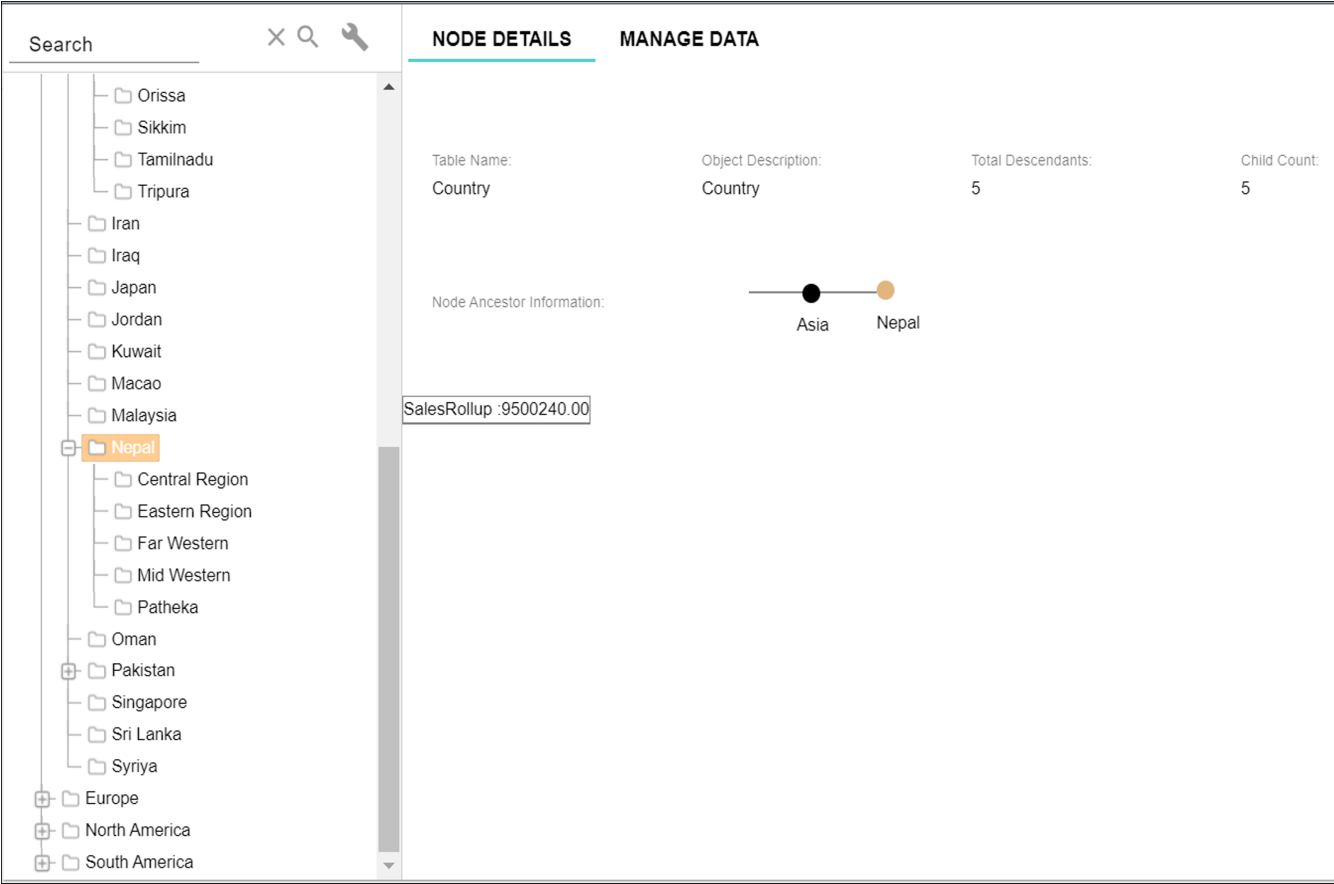
**Drag and Drop and Instant Rollup Impact**

When you move nodes in the hierarchy, the rollup attributes are automatically updated and reflected in the new hierarchy structure without the need of any BI process. This allows business users to do instant impact analysis.

An example is shown below.

Parent node Nepal has 5 Child and has an overall SalesRollup of 9500240 as in [Figure 236](#).

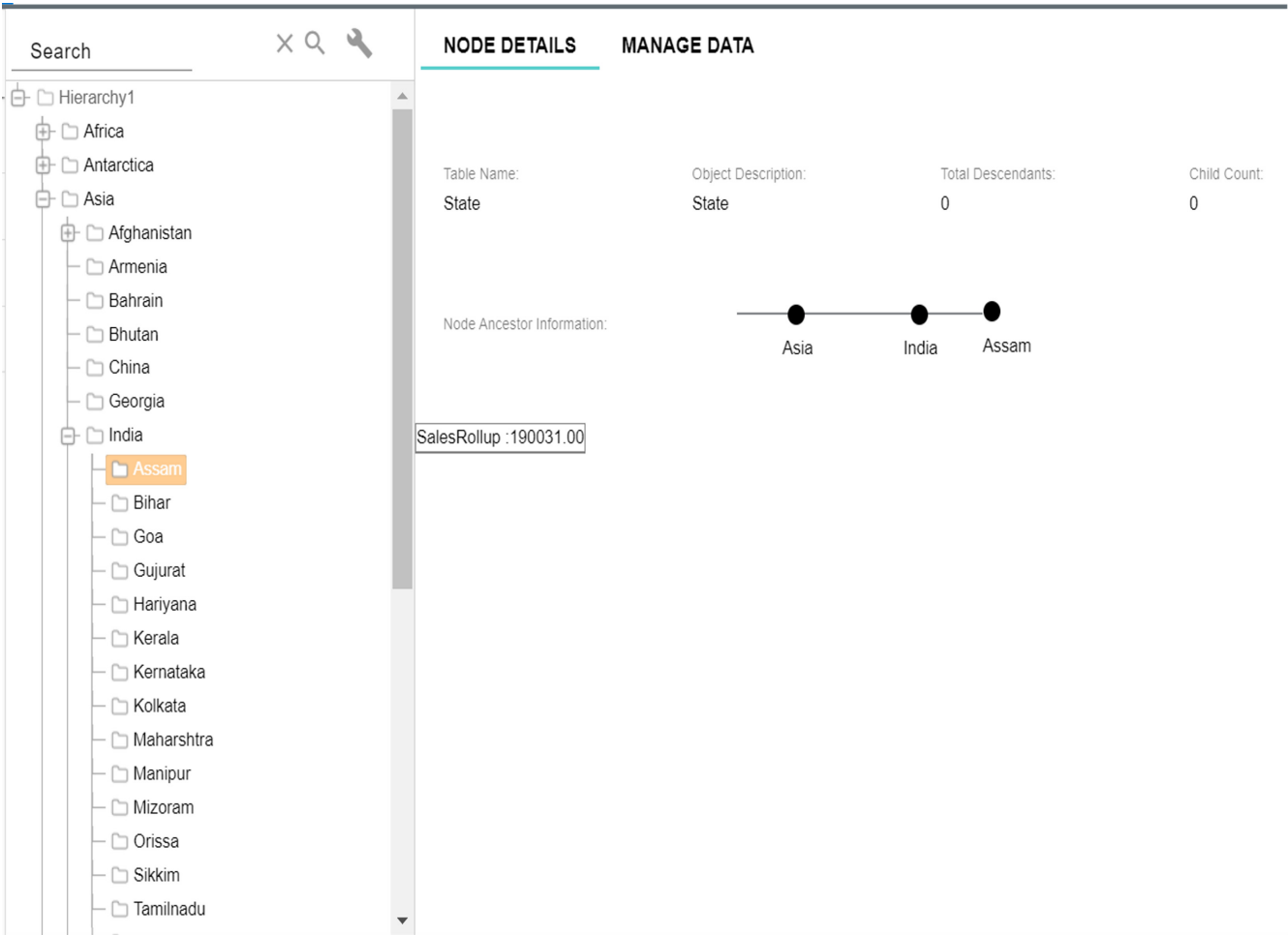
Figure 236: Hierarchy Viewer—Rollup Attribute—Drag and Drop



Child node Assam under parent node India has rollup of 1900031 as in [Figure 237](#). Now let us drag and drop Assam to Nepal parent node.

Once the drag and drop is successful, Nepal now has the reflected SalesRollup of Assam combined with the existing 5 child nodes as in [Figure 237](#).

Figure 237: Hierarchy Viewer—Rollup Attribute: Drag and Drop



Hierarchy Statistics

The **Stats** option button on the Hierarchy Manager allows you to collect statistics related to the selected hierarchy. The collected statistics provides information about the number of levels in the hierarchy, number of records in each level and leaf nodes at each level. You can view the statistics and save the data collected for a particular hierarchy.

Based on the data collected, a decision can be made to get all the records or fetch the records at required levels in hierarchy viewer tree structure there by reducing the loading time of the hierarchy tree structure.



If the statistics are not captured for a particular hierarchy, the decision on which mode to be used for loading will be based on the value set in hierarchy.xml file.

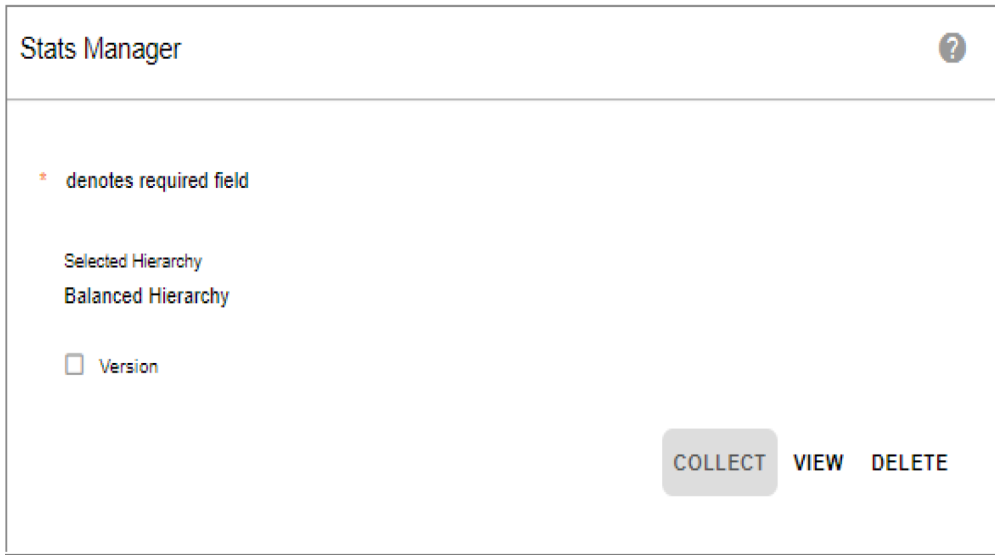
The Manage Stats button is visible only for the members of the Super User group.

Navigate to **Stats Manager** UI from **Hierarchy Manager** UI using **Manage Stats** option.

On the **Stats Manager** UI ([Figure 207](#)), you can collect, view and delete stats of the selected hierarchy or any version of the selected hierarchy. Select the Version checkbox to collect the stats of the any hierarchy version.

To collect statistics, click **Collect**.

Figure 238: Stats Manager



Stats Manager

\* denotes required field

Selected Hierarchy  
Balanced Hierarchy

☐ Version

COLLECT VIEW DELETE

Message: “Stats collection in process” is displayed.

On the **Stats Manager** UI ([Figure 207](#)), click on **View** to see the hierarchy statistics details as in [Figure 238](#).

Figure 239: Hierarchy Statistics

Hierarchy Statistics

Hierarchy Name

Balanced Hierarchy

Category

Small

Total Records

176

Recommended Mode

Pre-fetch

Selected Mode

Pre-fetch

Statistics Last Collected

02/26/2021 10:04:40 AM

Mode of Loading:

☐ on Demand
 ☒ Pre-fetch

Stats

^ Hierarchy Details

Level	Records
1	6
2	101
3	69

BACK

SAVE

On the **Hierarchy Statistics** UI [Figure 238](#),

- **Hierarchy Name:** displays the name of the hierarchy for which the statistics details are displayed.
- **Category:** specifies the size of the hierarchy.
- **Total Records:** specifies the total number of records in the selected hierarchy.
- **Recommended Mode:** specifies the recommended mode of loading hierarchy on the viewer as Pre-fetch.
- **Selected Mode:** displays the selected mode of loading the hierarchy on the viewer.
- **Statistics Last Collected:** displays the date and time of the last collected statistics.

- **Mode of Loading:** the Mode of Loading option will be selected either as On Demand or Pre-fetch. Based on the statistics collected and based on the threshold value of on Demand load set in System Properties table, mode of loading is decided. If the number of nodes exceeds the set limit, system automatically recommends the mode of hierarchy load as on Demand else its recommended as pre-fetched.
- **On Demand:** only first two levels of data are fetched and loaded initially. The remaining levels are loaded on Demand. When user expands a node of a hierarchy, query is made to database and its corresponding children are fetched and loaded.
- **Pre-fetched:** entire hierarchy data are fetched and loaded during the initial load of it. Recommended for hierarchies containing less (approximately below 1000 nodes) amount of data.
- **Hierarchy Details** section displays the different levels and the total number of records in each level.
- Click **Save** to save the collected stats.

## Manage Cross Reference

Navigate to **Manage Cross Reference** UI (Figure 240) from **Reference Data -> Hierarchy Management -> Hierarchy Manage** and on **Hierarchy Manage** UI, select **Cross Reference** from **Actions** dropdown.

Figure 240: Manage Cross Reference

## View Cross Reference Data

To view cross reference data:

- 1 On the **Manage Cross References** UI, click on the Menu icon corresponding to the required cross reference object and click **View Data**.  
The **Select Parent Data** UI is displayed.
- 2 On the **Select Parent Data** UI, select the parent object data and click **Linked Child**.  
The **Selected Parent Object Data—Linked Child Object Data** UI is displayed.  
The **Linked Child Object Data** pane displays the child objects linked to the cross referenced object of the selected parent object data.



- 3 Click **View Additional Attributes** to view the defined additional attributes of the selected cross reference data.

You can optionally delete the child object from the **Selected Parent Object Data—Linked Child Object Data** UI.

- 4 On the **Selected Parent Object Data—Linked Child Object Data** UI, select the required child object and click **Delete**.

The **Confirmation** dialog box is displayed.

- 5 On the **Confirmation** dialog box, click **Yes**.

## APPENDIX A Landing Page and Favorites

---

### What's In This Appendix

This appendix provides information about RDM landing page and Manage favorites.

Topics include:

- [Managing RDM Landing Page](#)

## Managing RDM Landing Page

When a user logs in to RDM, the RDM landing page or home page can be configured for different users based on their roles. By default, RDM provides the following landing page (nav\_home).

You can create any landing page from Manage Pages UI and these pages can then be assigned to user based on their roles from **Manage Users->Create User page**.

### Create Page

To create a new page:

- 1 Navigate to Manage Pages UI from **Administration -> Menu Builder** and on the **Menu Builder** UI, in the **Customize Menu** pane, click **Manage Pages**.

Figure 241: Manage Pages

Manage Pages

BACK

ADD PAGE

ADD WORKFLOW

?

Search Results: Page 1 of 18

Page Name #	Page Type #	Page URL #	
x	x	x	
about	PAGES	/bcm/framework/about.jsp	
AccountConfigAdv	PAGES	/start.x2ps?START_WORKFLOW=ConfigUIWFRuntime&SERVICE_NAME=MDMServices&SOLUTIO	
Account_302	PAGES	/start.x2ps?START_WORKFLOW=ConfigUIWFRuntime&SERVICE_NAME=MDMServices&SOLUTIO	
Account_303	PAGES	/start.x2ps?START_WORKFLOW=ConfigUIWFRuntime&SERVICE_NAME=MDMServices&SOLUTIO	
activitySearchHelp	PAGES	/bcm/help/usermanagement/activity_search_help.htm	

1 - 20 of 343

<<

<

>

>>

- On the **Manage Pages** (Figure 241), click **Add Page**.

Figure 242: Add Page

Add Page

?

\* denotes required field

Page Details

Page Name \*

Home\_Alerts

Page URL \*

/bcm/framework/alerts/xsl/alerts.xsl

CANCEL

SAVE

- On the **Add Page** (Figure 242), enter the name of the page and the URL of the page being created.

The page to be accessed must be copied at any folder under *<RDM\_Install\_Directory/web/mdmclient/>*. If the file is copied in bcm folder located at *<RDM\_Install\_Directory/web/mdmclient/bcm/>*, the path would be */bcm/<filename>*

## Assign Landing Page

To assign landing page:

- 1 Navigate to **Create User-User Details** UI (**Administration -> User Management -> Users**) and on the **Manage Users** UI, click **Create User**.
- 2 On the **Create User-User Details** UI, enter the user details and select **Assign Landing Page** checkbox.
  - **Use User-Level landing page:** if Use User-Level landing page option is selected, the landing page dropdown is displayed as in [Figure 243](#). Select the landing page and click **Save**.

Figure 243: User Details—Landing Page

User Details

Organization

System

Postal Code

92127

User Group

SU\_UGP

Designation

Select...

Phone

Manager

Select...

Fax

Absentee

Select...

E-mail Address

mdm\_admin@teradata.com

Locale

English (United States)

☒ Assign Landing Page

☒ Use User-Level landing page

☐ Use Role-Level landing page

Landing Page

Default Home Page

Address Line 1

Del Campo

CANCEL

RESET

SAVE

- **Use Role-Level landing page:** if Use Role-Level landing page option is selected, the landing page dropdown is displayed as in [Figure 244](#). Select the role and click **Save**.

Figure 244: User Details—Landing Page

User Details

Organization \*

System

Postal Code \*

92127

User Group \*

SU\_UGP

Designation

Select...

Phone

Manager

Select...

Fax

Absentee

Select...

E-mail Address \*

mdm\_admin@teradata.com

Locale \*

English (United States)

☒ Assign Landing Page
   
☐ Use User-Level landing page
   
☒ Use Role-Level landing page

Landing Page \*

Super User

Address Line 1 \*

Del Campo

CANCEL

RESET

SAVE

- The landing page for role can be configured from **Manage Roles->Create Role** UI as in [Figure 245](#). Select the landing page and click **Save**.

Figure 245: Role Details—Landing Page

Role Details

\* denotes required field

Role Name\*

MDM Admin Role

Role Id\*

MDM Admin Role

☒ Enable Edit Access

Landing Page\*

Default Home Page

CANCEL

SAVE

Changing the landing page for a particular role will change the landing page for all users belonging to that role.

## User Favorites

Customizations defined once on search screens of generated configurable view table UI or PGL UI workflow can be saved for future re-use as favorites. You can create and save the favorite at user level as well as at the role level.



- To enable Favorites on PGL UI node, set the below attribute:  
`showFavorite="true"` attribute on the `<ui:container>`
- In order to support Save Search functionality on PGL favorites, you must customize the code as below at the end of pre processing in the UI node.  

```
<ADD_CHILDREN DocVar="pglFormInput" FromSelectList="$thisParam/
*"/>
```

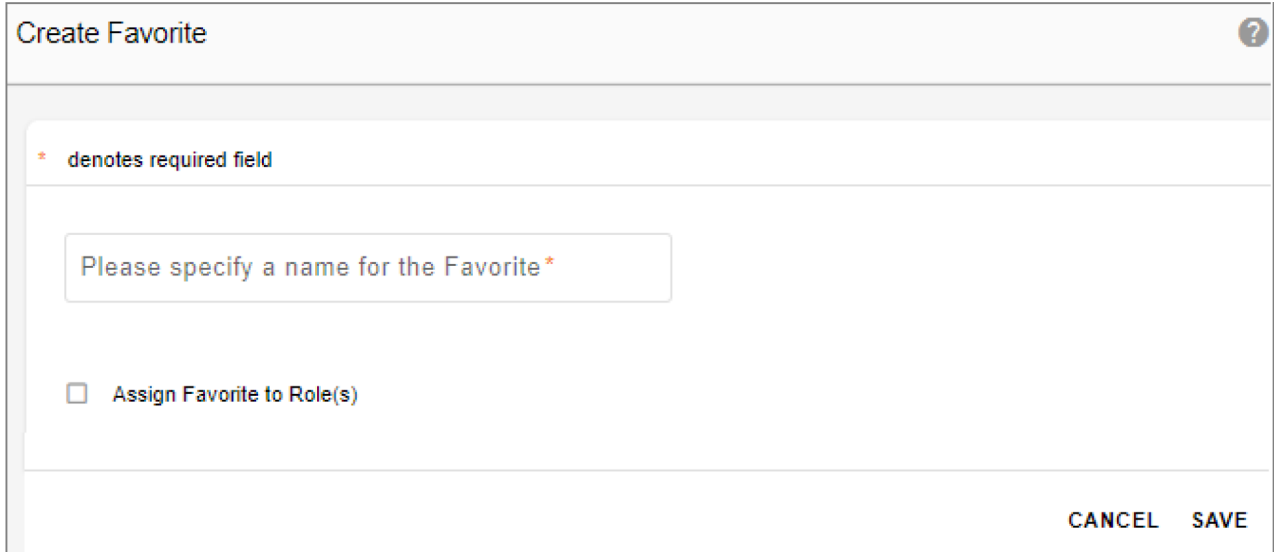
For more details on achieving Save Search on PGL Favorites, refer to *Section Search and Results in chapter 3 Basic Controls in Master Data Management Reference Guide*.

### Create Favorite at User Level

Favorites created at the user level will be available only to that user who created the favorites.

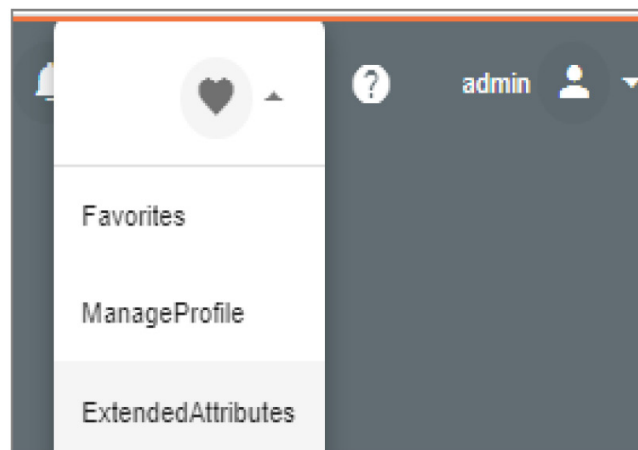
- On any generated configurable view table page or PGL UI, click the **Create Favorite** icon. The **Create Favorite** page (Figure 246) is displayed.

Figure 246: Create Favorite



- 2 On the **Create Favorite** page (Figure 246), enter a name for the favorite and click **Save**.  
Message: Favorite saved successfully is displayed. The favorite is created and saved for the user under the **Favorites** icon as in Figure 247 on MDM UI. Click on the Manage Favorites link to navigate to the Favorites list.

Figure 247: Favorites List



## Create Favorite at Role Level

MDM provides an OOTB feature to create and save the favorite at role level; that is, user can create a favorite and assign it to any role. Any user with that role will be able to view all the favorites created for that particular role as well as the favorites created by the user.





- Only a user assigned with “Create Role Based Favorites” activity would be able to create favorites for any role irrespective of whether the user belongs to that role or not, but cannot view, delete or organize role based favorites unless assigned with “Delete Role Based Favorites” activity.
- A user can manage the role based favorites only when a local favorite exists for the user. In the absence of a local favorite, the Manage Favorites icon will not be visible on the left navigation pane even if the “Delete Role Based Favorites” activity is assigned to the user.

To create favorite at role level:

- 1 On the **Create Favorite** page, enter a name for the favorite and select the **Assign Favorite to Role(s)** checkbox as in [Figure 246](#).

The Create Favorite page refreshes with the Available Roles and Selected Roles pane.

Figure 248: Save Favorite Page

Create Favorite

\* denotes required field

Please specify a name for the Favorite \*

ConfigurableUI

☒ Assign Favorite to Role(s)

Available Role(S)

Selected Role(S)

New\_Name\_mod  
Admin  
Toolkit User  
New\_Name  
Default Role  
Super User  
TAS User

RDM\_USER  
RDM\_ADMIN

CANCEL SAVE

- 2 From the **Available Roles**, select the required role and move it to the **Selected Roles** pane using the arrow button and click **Save**.

Login with the required user role details. On the MDM UI, on the top right corner, click on the favorite icon. The favorite icon displays all the favorites created for the user as well as all the favorites that exist for the roles to which the user belongs to.

## Manage Favorites

The manage favorites page displays all the favorites created for the role as well as the favorites created by the user.



Only a user assigned with “Delete Role Based Favorites” activity would be able to manage all the role level favorites.

---

To manage favorites:

- 1 Select **Manage Favorites** from **Favorite** icon on MDM UI.

The **Manager Favorites** page ([Figure 249](#)) is displayed.

On the manage favorite page, the “Created for Role/User” column displays the role name if the favorite was created for a role and displays the “user name” if the favorite was created by the user. The “Created By” column displays the “user name” of the user who created the favorite.

Figure 249: Manage Favorites Page

Manage Favorites

Search

Name

SEARCH

Favorites: Page 1 of 1

<input type="checkbox"/>	Select Favorite	Name	Created For Role/User	Created By
<input type="checkbox"/>	Select Favorite	ManageProfile	USR_1	USR_1
<input type="checkbox"/>	Select Favorite	ExtendedAttributes	USR_1	USR_1
<input type="checkbox"/>	Select Favorite	TestConfigUI	USR_1	USR_1

ORGANIZE FAVORITES

DELETE

CLEAR

1 - 3 of 3

<

>

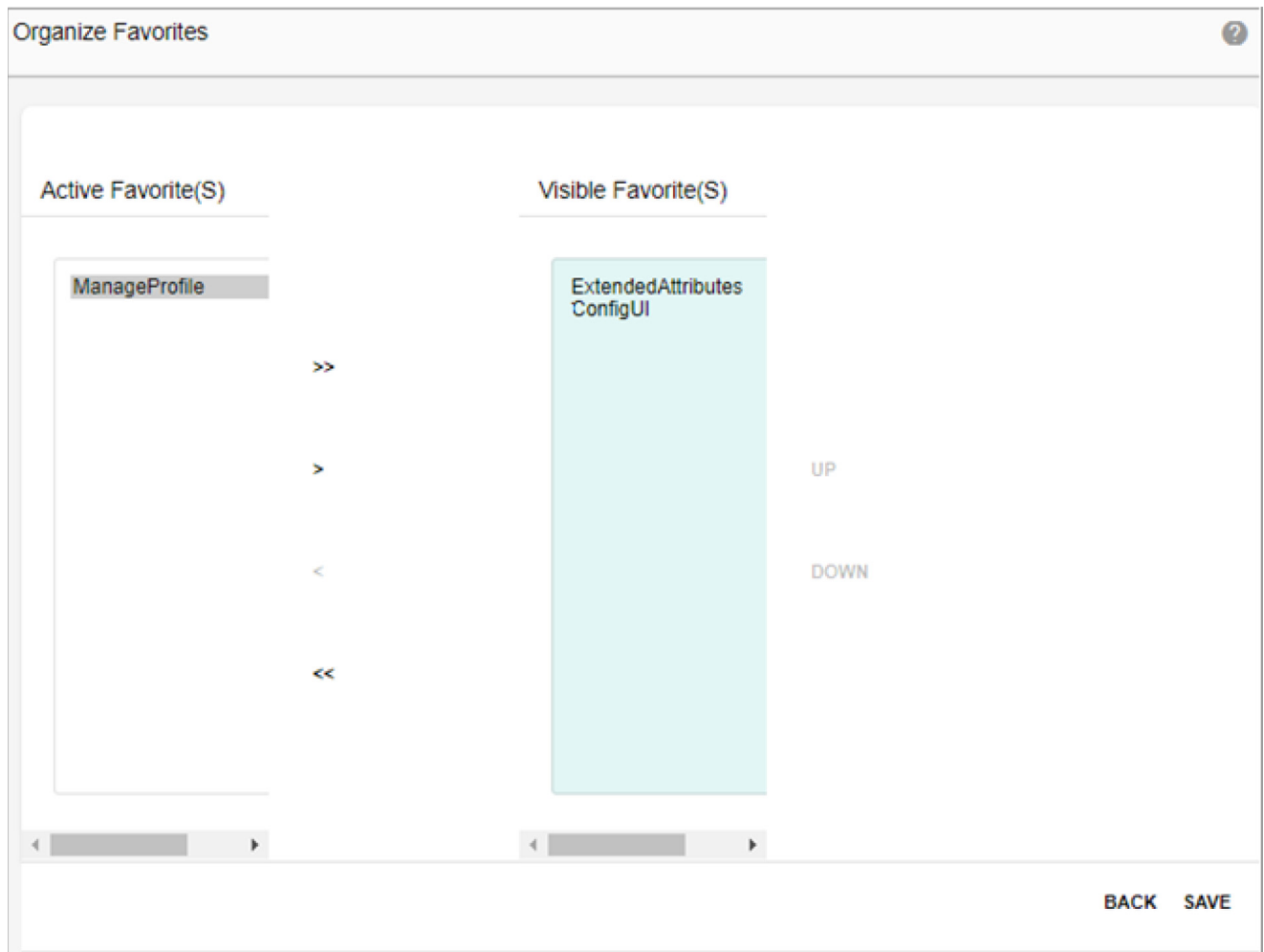
- 2 On the **Manager Favorites** page (Figure 249), you can perform the following:
- To search for a specific favorite; in the **Search** pane, in the **Name** field, enter the favorite name and click **Search**.  
The **Favorites** pane displays the search result.
  - To navigate to a favorite page; in the **Favorites** pane, in the **Select Favorite** column, click the required icon to navigate to the favorite page.
  - To delete a favorite; in the **Favorites** pane, select the favorite by clicking the corresponding checkbox and click **Delete**.



Only a user assigned with “Delete Role Based Favorites” activity would be able to delete favorites for any role irrespective of whether the user belongs to that role or not.

- To organize favorites: in the **Favorites** pane, click **Organize Favorites** to organize the favorites.  
The **Organize Favorites** page (Figure 250) is displayed.

Figure 250: Organize Favorites



On the **Organize Favorites** page (Figure 250), the **Visible Favorites** column displays all the favorites displayed on the MDM left navigation pane and the **Active Favorites** column displays the favorites that are not visible on the left navigation pane.

- To hide any favorites on the MDM left navigation pane, select the required favorites from the **Visible Favorites** column and move to **Active Favorites** column using the appropriate arrow buttons. To hide all favorites, click Unselect All Fields arrow button.
- To display any favorites on the MDM left navigation pane, select the required favorites from the **Active Favorites** column and move to **Visible Favorites** column using the appropriate arrow buttons. To display all favorites, click Select All Fields arrow button.
- Click **Up** or **Down** button to reorder the display of favorites in the MDM left navigation pane and click **Save**.

Accordingly, the Favorite bar on left navigation pane displays the favorites.

#### Limitations on Favorites:

The following list provides the limitations on favorites:

- While creating favorite for a role, the favorite name must be unique across the system.
- The Favorite bar and the Manage Favorites icon on the MDM left navigation pane will be displayed only when a favorite is created for the user or favorite is created for role to which the user belongs to.
- Favorites are not supported on MDM Breadcrumbs.
- Due to Framework limitations, the Role based favorites assigned to particular user is not displayed in the tiles form on the home page of the logged in user.

## APPENDIX B Model Builder

---

### What's In This Appendix

This appendix provides detailed information on defining RDM based tables from RDM web UI using Model Builder UI.

Topics include:

- [Introduction](#)
- [Model Builder UI](#)

## Introduction

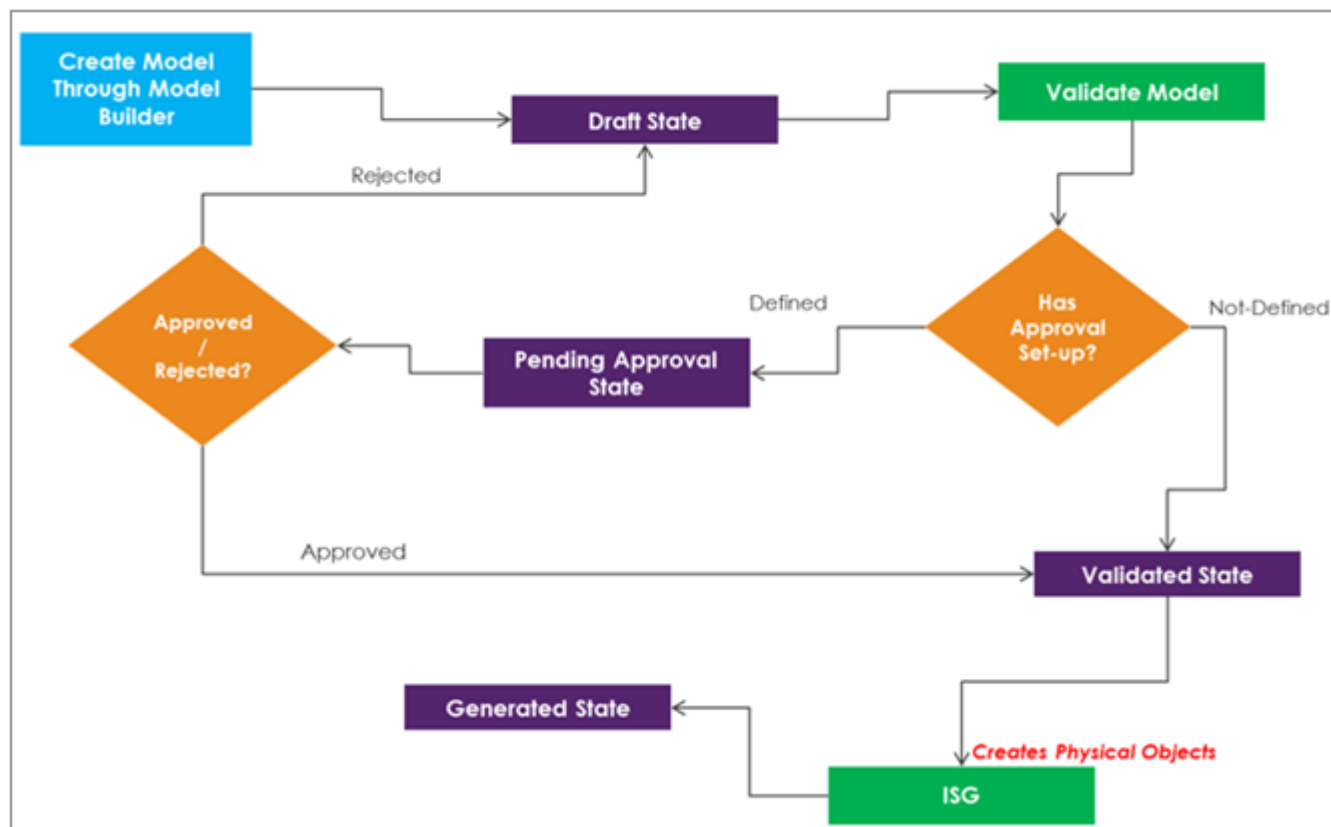
Model builder is the capability to model (Create, Edit) and deploy RDM based tables at runtime. Standard features (such as, Approval, Hierarchy, Configurable, E2E, Excel Upload, Business Rules etc.) supported on RDM based tables are supported on tables created through model builder.

Key features of Model Builder:

- Provides an easy to use UI to model RDM based tables including staging tables from RDM web UI with facilities to define column definition, key definition, facet definition, etc.
- Provide support of excel upload to model tables as an alternate option to using web UI.
- Provides facility to create dictionary and migrate studio built dictionaries.
- Provides capability to define approval on model to review the SQL before being executed in ISG.
- Provides facility to view the table and its column description on the database.
- Provides user security access to metadata of the generated tables.
- Provides the advantage of applying several validations to prevent potential failure during ISG process.
- Provides facility to create template based tables for Hierarchy and RDM modules.
- Provides detailed ISG logs with status of each step and pre and post ISG metadata count.
- The ISG process run through model builder is truly incremental in nature. In case of any failure, entire changes will be reverted back so that existing tables work fine.

[Figure 251](#) displays the basic flow of deploying tables in model builder.

Figure 251: Model Builder Stages



- **Draft State:** When a new model of table is created through model builder, it first enters into DRAFT state. All the details of model in draft state can be edited. The models uploaded through excel upload also enters into DRAFT state. If an already generated model is edited, it will move to DRAFT state.
- **Pending Approval State:** When a model in DRAFT state is validated and if approval is enabled on models, it moves to PENDING APPROVAL state. If the approver approves the model, the model enters the VALIDATED state. If the approver rejects the model, the model enters the DRAFT state again. When a model is in PENDING APPROVAL state, the model cannot be edited or deleted. You can only view DDLs and View ISG logs.
- **Validated State:** When a model in DRAFT state is validated and if approval is not enabled, it moves to VALIDATED state. The process of validation involves complete checking of schema in order to make sure the model is ready for ISG. The process of validation is equivalent to service set-up performed through studio for studio generated models. A log is generated and displayed on completion of validation process. In case of any errors, the model needs to be validated once again after correcting the errors. Models in validated state are eligible to take part in ISG process.
- **Generated state:** When ISG is executed for a model in VALIDATED state, it moves to GENERATED state. The detail logs of model in generated state can be viewed for checking successful completion of ISG. When a model in GENERATED state is edited, it moves to DRAFT state.

- **Deleted Model:** When the Deleted Model option is selected, the Model Builder UI displays all the deleted models. You can activate the deleted models using the Reactivate Model button on the Model Builder UI. All the reactivated models will be in draft state.

## User Security Access to Metadata of Generated Tables

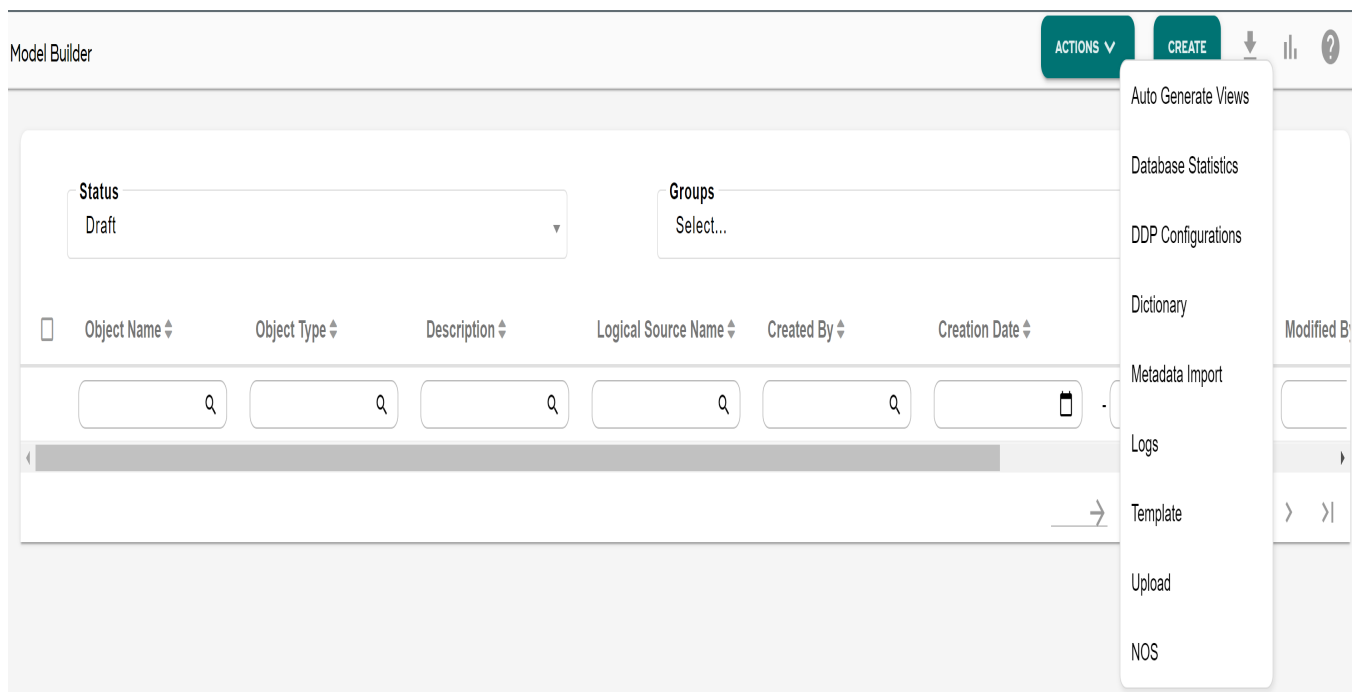
The Metadata of models created through model builder can be accessed only by the creator of the model and the super user when the model is in Draft, Pending Approval and Validated state. When the model moves to Generated state, the creator will have access by default. To provide access to other user roles, Metadata access permission must be enabled for the RDM tables on the Role Details—Activity Search UI. If a role is provided access to activity for any staging table, user having that role will be able to access the metadata of the table.

Once the metadata access is enabled, users having the assigned roles will be able to access the metadata of the tables in the generated model state.

## Model Builder UI

You can navigate to **Model Builder** UI from **Model Setup ->Model Builder**. The [Figure 252](#) displays the Model Builder UI. On the **Model Builder** UI, by default, the Draft Models radio button is selected and all the models in the draft state will be displayed. You can select Validated or Generated radio buttons to see models that are in Validated or Generated state.

Figure 252: Model Builder



On the Model Builder UI, you can perform the following tasks:

- [Create or Edit Model](#)



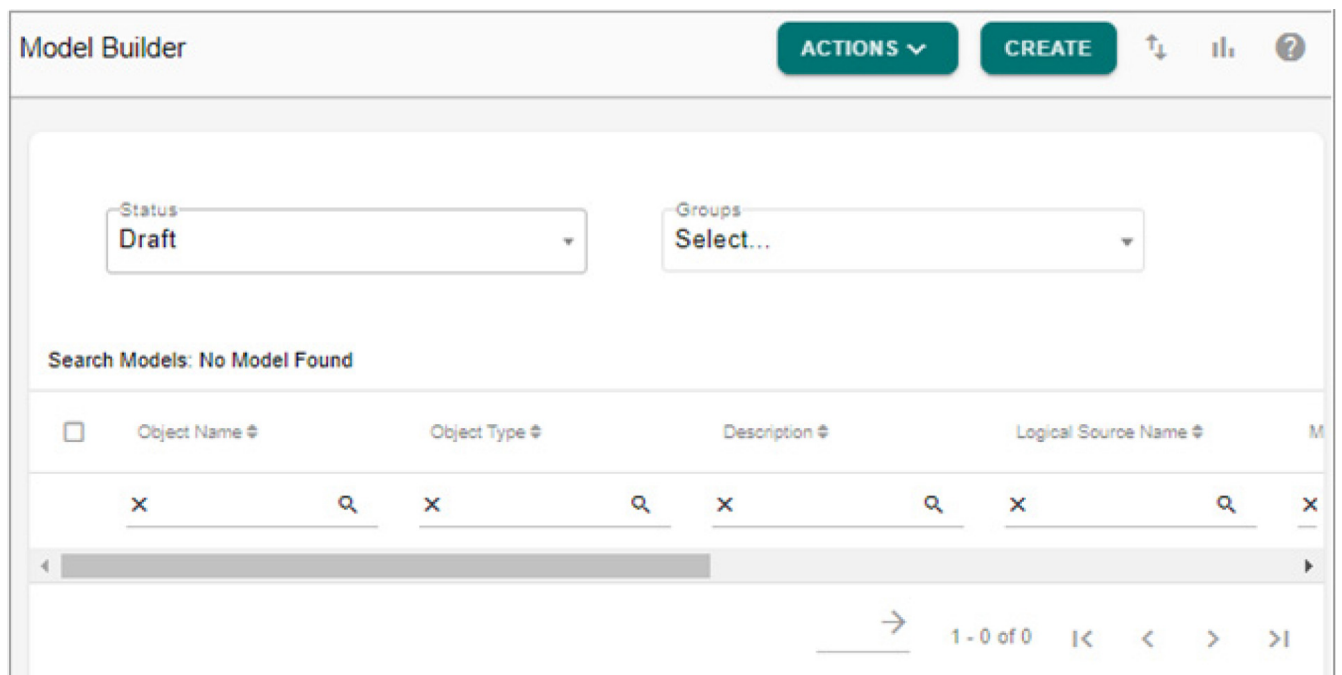
- [Create RDM Based View](#)
- [Auto Generate View](#)
- [Validate Models](#)
- [Deploy Model](#)
- [Deployment Logs](#)
- [API to Validate and Generate ISG](#)
- [Modify Table Primary Key](#)
- [Delete Model and Delete Model Metadata](#)
- [Export Model](#)
- [Import from X-Document](#)
- [Manage Dictionaries](#)
- [Manage Templates](#)
- [Manage Metadata Import](#)

## Create or Edit Model

Detailed process on how to create a table using Model Builder is described below. Account table is taken as an example.

- 1 Navigate to Model Builder UI and click **Create** to initiate model creation process as in [Figure 253](#).

Figure 253: Model Builder



- 2 On the **Model Builder** UI, with **Object Details** tab selected, enter the following:
  - **Object Name:** unique logical name of the object and the name should not contain spaces.

- **Description:** brief description about the object being created. The descriptions can also be viewed in the Database level after ISG.
- **Deploy in Secondary Connection:** select this checkbox to deploy the model in the secondary connection. This option can be used to deploy non-MDM tables in databases other than RDM user.
- **Create New Group:** select this checkbox to create a new group. If this checkbox is selected, the Group Name text field is displayed. Enter a name for the group under which the table needs to be created in the Group Name field.
- **Group Name:** displays the list of existing groups. Select a group, if you want to create a table under any of the existing group.
- **Object Type:** Object type can be either RDM based table or RDM based view.
  - **View:** select View option to create RDM based view.  
**View and Advanced Settings option:** if you select View option and the Advanced Settings option, the View Details pane is displays the View Display Name and Physical View Name fields with default values based on the specified object name. You can modify the default values as required.
  - **Table:** select Table option to create RDM based table. If Table option is selected, the Table Type is displayed. The Table type can be either Set or MultiSet.  
**Set:** if the Table Type is selected as Set, then the table does not allow duplicate rows.  
**MultiSet:** if the Table Type is selected as MultiSet, then the table allows duplicate rows.  
**Table and Advanced Settings option:** you can select the Table option and the Advanced Settings option to set the staging, configuration details and Web service actions as in [Figure 254](#). The Advanced Settings pane displays the default values based on the specified object name. You can modify the default values as required.

Figure 254: Object Details—Advanced Setting-Staging Details

OBJECT DETAILS

COLUMNS

KEYS

LINKS

BACK

SAVE

?

Object Details

Object Name\*

Account

Description

Account

☐

Deploy in Secondary Connection

☒

Create New Group

Groups\*

Account

Object Type\*

Table

Table Type\*

Set

☒

Advanced Settings

☒

Statistics

STAGING DETAILS

CONFIGURATION DETAILS

WEB SERVICE ACTIONS

Table Display Name

Account

Physical Table Name

ACCOUNT

Master Table Name

MST\_ACCOUNT

Error Table Name

ERR\_ACCOUNT

Output Table Name

OUT\_ACCOUNT

Version Table Name

VER\_ACCOUNT

Source

MASTER

- Staging Details tab contains details of staging tables. All names of staging tables are derived from Table Name field under Model Details.



If the table names are modified, it is user's responsibility to update the staging table names.

---

With **Staging Details** tab selected as in [Figure 254](#), enter the following:

- **Table Display Name**—the display name of the table.
- **Physical Table Name**—the physical name of the table.
- **Input Table Name**—the name of the input staging table.
- **Error Table Name**—the name of the error staging table.
- **Master Table Name**—the name of the master staging table.
- **Output Table Name**—the name of the output staging table.
- **NetChange Table Name**—the name of the netchange staging table.
- **Version TableName**—the name of the version staging table.
- **Source**—select the source for the table. Can be Master or Backend or both.

If Source is BackEnd, tables are created for Input, Netchange, Master, Error, Version and Output staging.

If Source is Master, Netchange and Input tables are not created. Only Master, Error, Version and Output table are created.

If Source set to BOTH, tables are created for Error, Output, Input, Netchange, Version and Master staging area. Table for Netchange service will carry fields that are sourced from Backend. In Netchange, by default only PK of the table will be created, that is, if user does not define any columns as backend.

---



Default facet values will be populated if no value provided for facets. Staging table names will follow standard RDM based staging table naming convention; all other properties will have default values false.

---

- Configuration Details tab consists of various configurations related to the properties of the table. With **Configuration Details** tab selected as in [Figure 255](#), select the following:

Figure 255: Model Builder—Advanced Setting-Configuration Details

The screenshot shows the 'CONFIGURATION DETAILS' tab in the Model Builder UI. It contains two columns of configuration options, each with a checkbox. The left column includes 'Approval History' (checked), 'Audit Trail' (unchecked), 'Is Native' (unchecked), and 'Is Temporal' (unchecked). The right column includes 'Is Extendable' (unchecked), 'Upload Row Number Enabled' (unchecked), 'Cached' (unchecked), 'Surrogate' (unchecked), 'Hard Delete' (unchecked), 'Auto Sort Columns' (checked), and 'In-Line Edit' (unchecked). Below the left column is a 'Notify' button.

- **Approval History:** select the checkbox to enable approval history on table
- **Audit Trail**—select this checkbox to enable audit trail on table. If Audit Trail option is enabled, the Audit End Date checkbox is displayed. When Audit End Date checkbox is selected, the existing entry in audit table will be updated with an end date and a new entry will be made to the audit table with a start date.
- **Is Native**—select this checkbox exclude the table in RDM process and table will not be included in SYS\_TABLE\_MAP table.
- **Is Temporal**—select this checkbox to make the table temporal. When the Is Temporal checkbox is selected, the following options are displayed.
- **Schema Gen**—by default, this checkbox is selected to include the table in schema generation process and it is a non-editable field.
- **Hard Delete**—select this checkbox to enable hard delete on the table. That is any delete on table will permanently delete records from table.
- **Upload Row Number Enabled**—while loading the data through excel upload process for the model being created, if any error(s) are encountered, the corresponding row number will be captured and displayed on the error details UI if the “Upload Row Number Enabled” option is selected. If the “Upload Row Number Enabled” option is not selected, the “UPLOAD\_ROW\_NUMBER” column will not be displayed on the Error Details UI.



The display of error record row number is not supported on Input staging tables.

- **Notify**—enter the email addresses of all the users to be notified with the ISG execution results. Multiple email Ids to be separated by commas.
- **Auto Sort Columns**—if this checkbox is selected, the table columns on the configurable UI will be sorted and displayed in the following order: primary key columns, required columns and other columns. If this checkbox is not enabled, the columns will be displayed in the order in which the columns were created during model definition.
- **Inline Edit**—select the Inline-Edit checkbox to enable inline editing on the generated table UI.
- **Is Extendable**—if this checkbox is selected, you can dynamically add attributes or columns to an existing table without performing any changes to the underlying data model. When Is Extendable option is selected, it displays three entry fields where you can enter the number of extended attributes to be created as in [Figure 256](#). For String attributes, the default and minimum value is 5 and even if the entered value is less than five, five extended String attributes will be created.

Figure 256: Model Builder—Advanced Setting-Configuration Details

The screenshot shows the 'CONFIGURATION DETAILS' tab in the Model Builder interface. It contains several configuration options:

- ☒ Approval History
- ☐ Audit Trail
- ☐ Is Native
- ☐ Is Temporal
- Notify: [Text input field]
- ☒ Is Extendable
  - Number of String Attributes: [Text input field with value 5]
  - Number of Decimal Attributes: [Text input field]
  - Number of Date Attributes: [Text input field]
- ☐ Upload Row Number Enabled
- ☐ Cached
- ☐ Surrogate
- ☐ Hard Delete
- ☒ Auto Sort Columns
- ☐ In-Line Edit

- **Cached**—select this checkbox if data needs to be cached.
- **Surrogate**—select this checkbox to make the column surrogate key. When this checkbox is selected, the Id Prefix textbox is displayed. Specify the prefix parameter to system generated surrogate key.

- **Is Temporal**—select this checkbox to make the table temporal.  
When the **Is Temporal** checkbox is selected, the following options are displayed.
- **Select Temporal Type:** select the temporal type.
- **Valid Time Column and Transaction Time Column:** valid time column and transaction time column provide physical or logical name of the column. Enter the required values in the valid time column and transaction time column as in [Table 11](#).  
The above fields are optional and if not provided, the default names: SYS\_VALID\_TIME and SYS\_TRANSACTION\_TIME will be used.



Temporal feature is supported only on Teradata database versions 13.10 and above. All the limitations with temporal table on Teradata database will be applied to the RDM temporal table. For more details, refer to the Teradata Documentation.

Figure 257: Model Builder—Advanced Setting-Configuration Details

OBJECT DETAILS COLUMNS KEYS LINKS

BACK SAVE ?

OBJECT DETAILS COLUMNS KEYS LINKS

BACK SAVE ?

☒ Approval History

☐ Audit Trail

☐ Is Native

☒ Is Temporal

Select Temporal Type  
TEMPORAL\_VALID\_TIME\_DATE

ValidTime Column Name

Transaction Time Column Name

Notify

☐ Is Extendable

☐ Upload Row Number Enabled

☐ Cached

☐ Surrogate

☐ Hard Delete

☒ Auto Sort Columns

☐ In-Line Edit

Table 11: Temporal Data Type Description

Temporal Data Types	Valid Time		Transaction Time	
	Data Type	Required	Data Type	Required
Temporal_Valid_Time_Date	Date	User defined	--	--
Temporal_Valid_Time_Timestamp	Timestamp	User defined	--	--
Temporal_TransactionTime	--	--	Timestamp (6) with timezone	Not Null
Bi_Temporal_Date	Date	User defined	Timestamp (6) with timezone	Not Null
Bi_Temporal_Timestamp	Timestamp	User defined	Timestamp (6) with timezone	Not Null

- Web Service Actions tab: with Web Service Actions tab selected, you can enable the web service components for the model being created. You can enable the CRUD (Create, Read, Update, and Delete) interfaces.
- 3 On the **Model Details** page (Figure 258), select the **Columns** tab and Click **+** to add column names by entering the below details
- **Column Name:** unique logical name of column.

Figure 258: Model Builder—Columns Tab

OBJECT DETAILS
COLUMNS
KEYS
LINKS

BACK
ACTIONS
DELETE
SAVE
+
?

Search Model Properties

<input type="checkbox"/>	Column Name *	Data Type *	Length	Extend	Required	Case Specific	Statistics	Character Set
<input type="checkbox"/>	col1_int	Integer		EXTEND	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col2_str	String	100	* EXTEND	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col3_str	String	100	* EXTEND	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col4_str	String	50	* EXTEND	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col5_bool	Boolean		EXTEND	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col6_bigint	BigInt		EXTEND	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col7_byteint	ByteInt		EXTEND	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col8_char	Char	50	* EXTEND	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...
<input type="checkbox"/>	col9_clob	Clob		EXTEND	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Select...



- **Data Type**—data type of the column. The data type can be (Boolean, Char, Clob, Date, Decimal, Double, EncryptedString, File, Float, Integer, BigInt, PeriodDate, PeriodTimestamp, String, ST\_geometry, Timestamp, VarByte, Xml and JSON)



If the data type default value is defined as SYSTEM\_DATE, SYSTEM\_TIME and SYSTEM\_TIMESTAMP, the default current date, time and timestamp would be taken as data value.

- **Length**—length of the data type. For decimal data type, number of digits after decimal can be specified by appending number with underscore and count of digits after decimal. For example: 30\_5.
- **Extend**—using extend option, details of columns can be extended from a dictionary where it is already defined.
- **Required**—select the checkbox, if you want the column to be a required field.
- **Case Specific**—select this checkbox if you want the column value to be case specific. By default, it is not case specific.



- Case specific is not supported on primary key columns.

- **Character Set**—the Character Set dropdown decides the character set for the column of the table. If LATIN is selected, the column character set will be set as Latin, this means it can accept only Latin characters. If UNICODE is selected, the column character set will be set as Unicode, this means it can accept any Unicode character. If no value is provided, by default LATIN is taken.
- **Compression Statement**—any compression syntax should be provided here. Compression plays an important role in saving some space and increasing the performance of SQL query. COMPRESS practically eliminates the data storage of nulls (or 256 constant values) for fixed-length, non-primary index columns. To implement Compression, add COMPRESS with the column definition as in the below example.  

```
COMPRESS ( 'RAJ' , 'KEVIN' , 'OBAMA' )
COMPRESS
COMPRESS ( 'HR' , 'IT' , 'FS' )
```
- **Default Value**—default value of the column if any.
- **Constraint**—you can define an expression based constraint that is evaluated dynamically at the database level. All the constraints must follow a rooted Boolean tree. If there is no root element present, by default AND is taken as the root element. For example to apply the constraint  
 $((\text{Role\_ID} > 50 \text{ AND } \text{Role\_ID} < 55) \text{ OR } (\text{Role\_ID} = 60))$  you must add the

constraint as:

OR

|

|\_\_\_\_AND

||

||\_\_\_\_(Role\_ID >50)

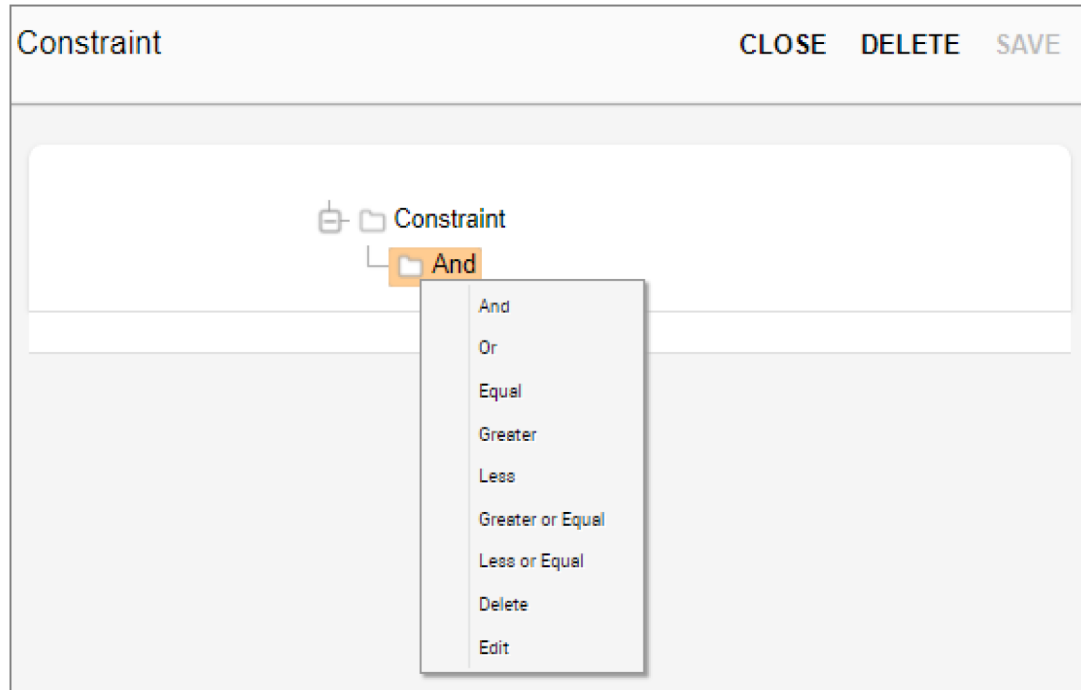
||\_\_\_\_(Role\_ID <55)

|

|\_\_\_\_Role\_ID=60

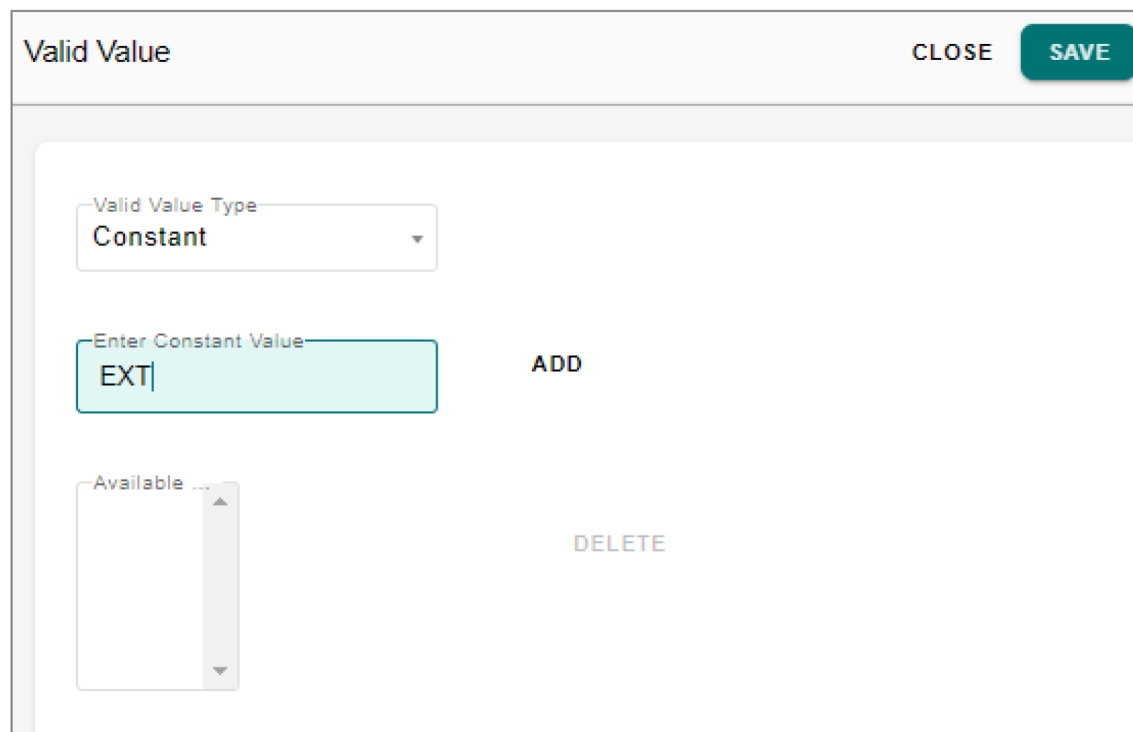
- On the **Constraint** column, click Constraint link.  
The **Constraint** pop-up ([Figure 259](#)) is displayed.
- On the **Constraint** pop-up, right-click the Constraint, the (And/Or) list is displayed.
- Select And or Or to add under the constraints and right-click on any added And or Or.  
The Menu option with list of operations is displayed.
- Select the required option from menu. For example: Select Greater and enter the input value.  
The value gets added to the Constraint.
- Click **Save**.
- On the Model Builder—Columns UI, under Constraint column, the link is displayed as Defined.  
You can click the Defined link to navigate to Constraint pop-up to modify the defined Constraints if required.

Figure 259: Constraint



- **Enum**—click Edit to provide the type of ENUM. The Valid Value popup is displayed as in [Figure 260](#).

Figure 260: Valid Value—Constant



In the **Valid Value** popup, the valid value type can be Constant, Lookup or Reference type.

- If **Constant** type is selected, enter the constant values in the **Enter Constant Value** text field and click **Add** and then click **Save** as in [Figure 260](#). The values will get added to the **Available Constant Values** list. You can rearrange the values using the **Up** and **Down** button. You can also delete the constant values using **Delete** button.
- If **Lookup** type is selected, select the code set and click **Save** as in [Figure 261](#).  
For any code set used as a lookup, by default, on the UI, code value would be displayed in the dropdown. To display code value description instead of code value, select the **Display Description** checkbox.

Figure 261: Valid Value—Lookup

The screenshot shows a 'Valid Value' dialog box with a title bar containing 'Valid Value', 'CLOSE', and a green 'SAVE' button. The main content area has a light gray background and contains three elements: a 'Valid Value Type' dropdown menu with 'Lookup' selected, a 'Code Set' dropdown menu with 'DESIGNATION' selected, and a checked checkbox labeled 'Display Description'.

- If **Reference** type is selected, enter the following and click **Save** as in [Figure 262](#).  
**Source Table:** select the source table from which the data need to be fetched.  
**Source Property:** select the property of the source table from which the value need to be fetched.  
**Display Description:** by default, the source property is displayed on the UI. To display the description of the property, select Display Description checkbox.  
**Source Display Property:** select the property of the source table to be used as the description to be displayed.  
**Validation Rule:** select this checkbox to create referential integrity rules for links like reverse link, but without creating soft link in database. By default, this is not enabled. You can select either reverse link or validation rule option, but not both.

Figure 262: Valid Value—Reference

The screenshot shows a 'Valid Value' dialog box. At the top, there's a title bar with the text 'Valid Value' and two buttons: 'CLOSE' and 'SAVE'. The main area of the dialog contains several configuration options:


- Valid Value Type:** A dropdown menu currently showing 'Reference'.
- Source Table:** A dropdown menu currently showing 'AttributeSet'.
- Source Property:** A dropdown menu currently showing 'Attribute\_Set\_Id'.
- Validation Rule:** A checkbox that is checked.
- Display Description:** A checkbox that is checked.
- Source Display Property:** A dropdown menu currently showing 'Attribute\_Set\_Id'.

- Constant: define a list of valid values on fly.
- Lookup: use an existing RDM code set.
- Reference: use a lookup table.

- **Physical Column Name**—provide physical name of column
- **Column display Name**—provide display name of column.
- **Description**—provide the description for the column.
- **Source**—select the source for the table column. Can be Master or Backend or both.
- **Editable**—select the checkbox if you want the column value to be editable.
- **No Wrap**—select the checkbox if you want the column value not to be wrapped to fit into a cell in table.
- **Sortable**—select the checkbox if you want the column value to be sortable.
- **Hidden**—select the checkbox if you want the column value to be hidden.
- **Searchable**—select the checkbox if you want the column value to be searchable.
- **l18nize**—select the checkbox if you want the column value to be l18nized.

If Secondary Connection option is selected in the Table Details then the below columns will be displayed:

- **Available Column Source**—the Available Column Source field displays all the available physical sources for a selected logical connection.

- **Selected Column Source**—the Selected Column Source field displays the selected source from all the available sources. By default all the sources are selected. If all the sources are selected then the column will not be source specific otherwise it will be source specific.
  - **Constraint Value**—Column level constraint can be placed here and the only permitted value is CHECK: allows to specify the range of value or value constraints permissible in the column.
  - **Manage Sequence**: the position of columns can be reordered within a table by using the **Manage Sequence** button. During ISG, the position of columns defined in Model Builder will be considered and auto generated configurable UI will be created. The auto generated configurable UI will have the columns in same order as defined through model builder.
- 4 On the **Model Details** page, select the **Keys** tab (Figure 263) and click  to enter key values with the below details: to enter the key values for the table (PK, NUPI, UNIQUE, UNIQUE\_INDEX).
- **Name**—unique name of the key. When you select the key type from **Type** drop-down, the Name and Physical Name field will be populated by default with a value as TABLENAME\_KEYTYPEPREFIXINTEGER(number of occurrence of particular key type). For example, if the selected key type is UNIQUE and the table name is Employee, the Key Name and Physical Name will be EMPLOYEE\_U1.  
You can change the default value if required. The below table displays the Key Prefix for different key types.

Key Type	Key Name
PRIMARY	TABLENAME_PK
UNIQUE	TABLENAME_U
UNIQUE INDEX	TABLENAME_I
INDEX	TABLENAME_I
NUPI	TABLENAME_PI

- **Physical Name**—physical name of the key. If the physical name is not provided, it will be derived from KEY NAME.
- **Type**—type of Key. The Key type can be (PRIMARY/UNIQUE/INDEX/UNIQUE\_INDEX/NUPI)  
**Note:** Ensure to have primary column or at least one column to be Not NULL.
- **Available Columns**—displays all the existing columns. Select the column name that will be part of this key.
- **Selected Columns**—displays the column name selected from Available Columns list.
- **Active**—By default the Active check box is selected. You can deactivate by deselecting the Active check box.

Figure 263: Model Builder—Keys Tab

- On the **Model Details** page, select the **Links** tab (Figure 264) and click **+** to add key values with the below details: to create links or relationship to other documents in the model.

Figure 264: Model Builder—Links Tab

- **Name**—enter name of the Foreign Key (FK) link. Name should be unique.
- **From Column Name**—select the column name defined under Columns tab from which FK link needs to be created.
- **To Table Name**—select the table name to which FK link needs to be created.
- **To Column Name**—enter the logical name of the column to which FK link needs to be created.
- **Physical Name**—enter the physical name of the link.

- **Is Reverse Link**—Foreign keys can be defined with or without reverse links. Select this checkbox to create reverse link from the referenced table. If reverse link option is checked, a database foreign key constraint will get created with no check (soft referential integrity) option only and default business rules will generate. This will restrict the delete on parent table if references are found in child table. If reverse link is not checked, the database constraint is not created but the forward link will be available in the X-document.
  - **Reserve Link Name**—enter a reverse link name. If no name is provided, it will be derived from link name.
  - **Validation Rule**— select this checkbox to create referential integrity rules for links like reverse link, but without creating soft link in database. By default, this is not enabled. You can select either reverse link or validation rule option, but not both.
  - **Active**—By default the Active check box is selected. You can deactivate by deselecting the Active check box.
- 6 Once all the model details are entered, click on **Save Model**. The created model will be in the draft state as in [Figure 265](#).

Figure 265: Model Builder

The screenshot shows the Model Builder interface. At the top, there's a header with 'Model Builder' and buttons for 'ACTIONS', 'CREATE', and a help icon. Below the header, a green checkmark and the message 'Model created successfully.' are displayed. Underneath, there are two dropdown menus: 'Status' set to 'Draft' and 'Groups' set to 'Select...'. Below these is a search bar with the text 'Search Models: Page 1 of 1'. A table follows with columns: Object Name, Object Type, Description, Logical Source Name, Modified By, and Modified Date. The table has one row with the following data: 'Account' (Object Name), 'Table' (Object Type), 'Account' (Description), 'Default\_DB' (Logical Source Name), and empty cells for 'Modified By' and 'Modified Date'. At the bottom right, there are pagination controls showing '1 - 1 of 1' and navigation arrows.

Object Name	Object Type	Description	Logical Source Name	Modified By	Modified Date
Account	Table	Account	Default_DB		



You can also create models through Excel Upload. You can define all the required properties of table (with details on logical/physical names, data types, PK/FK, PI, Character set, Table Subject Area etc.) in excel and upload it to create the table through ISG process. The



uploaded table will be in DRAFT state. You can download excel template from the Model Builder UI.

## Create Model using Excel Upload

On the Model Builder UI, excel upload facility is provided to model tables. You can define all the required properties of table (with details on logical/physical names, data types, PK/FK, PI, Character set, Table Subject Area etc.) in excel and upload it to create the table through ISG process. The uploaded table will be in DRAFT state. You can download excel template from the Model Builder UI.

In this scenario, Department table is taken as example.

To upload an excel file:


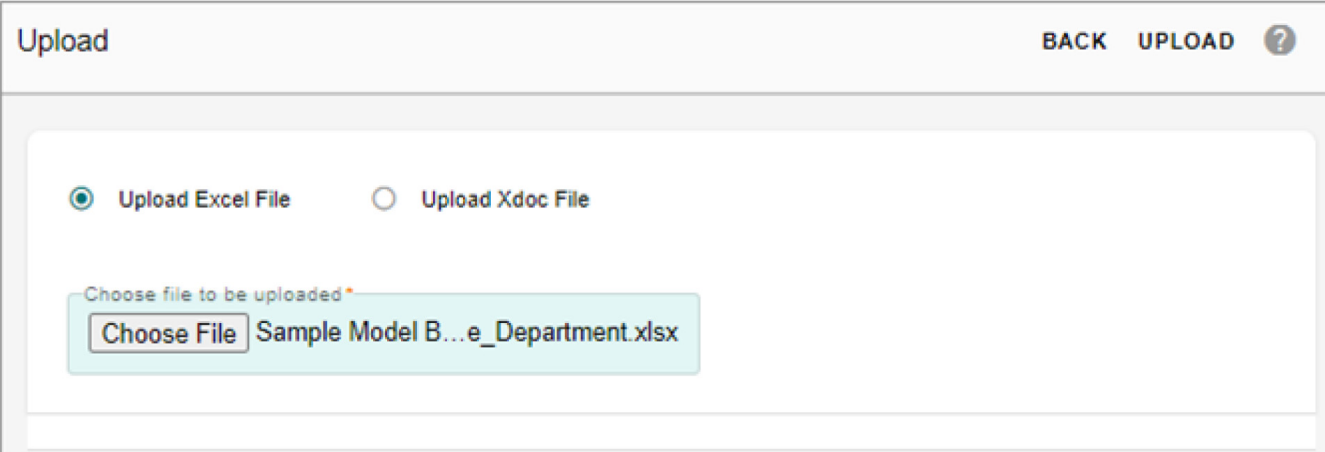
- 1 Download the document template by clicking on  icon, fill in the required table data under different tabs for object details, columns, keys, links and facet documents.  
For details on the various fields, see [Section : “Template Guidelines”](#)
- 2 On the **Model Builder** UI, select **Upload** from **Actions** dropdown.

Figure 266: Upload File



- 3 On the **Upload File** UI, select **Upload Excel File** option and click **Browse** to select the file and then click **Upload** as in [Figure 266](#).

Message: “Data uploaded successfully” is displayed.

The uploaded model will be in the draft state if there were no errors in the excel file. If there were any errors in the excel file, the upload will fail and the error message would be displayed. The error log can be viewed from **Administration -> Logs UI** under Platform service.



The following notes are applicable for table upload:

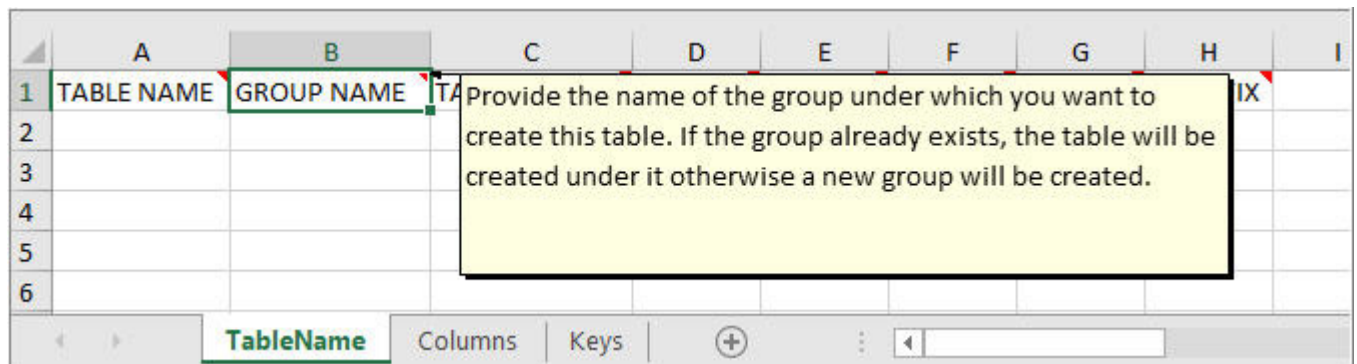
- Only one model at a time can be uploaded using excel upload template. You can upload multiple table metadata one by one and then run ISG for all of them at a time.
- Logical name and physical name are derived from display name if not provided.
- For specified property information, if column name for that property is not provided, the entire row will be skipped and other properties will be inserted.
- By default all entries in excel will be ACTIVE.

## Template Guidelines

The Model Builder UI provides Simple Excel Template, Advanced Excel Template and Non MDM Template option to download the document template. The Simple Excel Template contains only the fields that are required for creating a table. Advanced Excel Template contains all fields that are supported for a table. The Non MDM Template contains all the fields that are supported for non MDM table.

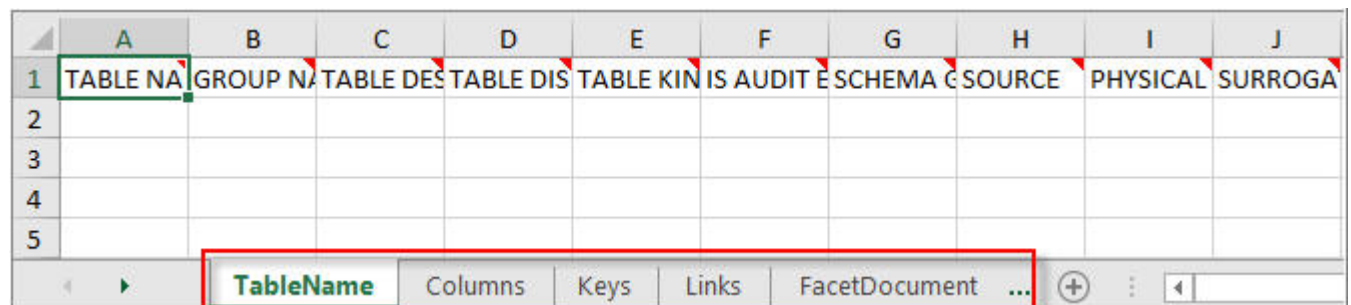
In Simple Excel Template option, the template for upload of Tables definition contains the three required tabs (TableName, Columns and Keys) as in [Figure 267](#). You can hover over on each column name in the excel sheet to see a brief description about the column name as in [Figure 267](#).

Figure 267: Model Builder Template—Simple Template



In Advance Excel Template option, the template for upload of Tables definition contains all the five tabs (TableName, Columns, Keys, Links and FacetDocuments) as in [Figure 268](#).

Figure 268: Model Builder Template—Advanced Template



In Non MDM Template option, the template for upload of Tables definition contains four tabs (TableName, Columns, Keys and Links) as in [Figure 269](#).

Figure 269: Model Builder Template—Non MDM Template

	A	B	C	D	E	F	G	H
1	TABLE NAME	GROUP NAME	TABLE DESCRIP	LOGICAL SOUR	TABLE KIND			
2								
3								
4								
5								

TableName	Columns	Keys	Links	+
-----------	---------	------	-------	---

The following guidelines will help to fill in the template correctly. In the template guide table:

- Tab Name: specifies the individual tabs in the excel template like table name, columns, keys, links and facet document.
- Columns: specifies the columns of individual tabs in the template. Columns that are marked in italics in the Excel template are not mandatory fields and even if these field values are not entered, default values will be populated.
- Description: specifies the description of the attributes or columns.
- Valid Value: specifies the valid values for the columns. Only the specified valid values will be accepted.
- Default Value: the specified default values will be taken if no valid values are entered.

Sheet Name	Columns	Description	Valid values	Default Value
TableName	TABLE NAME	Provide the Logical name of the table.		
	GROUP NAME	Provide the name of the group under which you want to create this table. If the group already exists, the table will be created under it otherwise a new group will be created.		
	TABLE DESCRIPTION	Provide description of the table.		
	TABLE DISPLAY NAME	Provide display name of the table. Default table display name will be the table name.		
	TABLE KIND	Provide the Kind of table either Set or MultiSet.	Set/MultiSet	Set
	IS AUDIT ENABLED	Provide TRUE if you want to create AUDIT tables else FALSE. If no value is provided, default value is taken as FALSE.	Valid values: TRUE/FALSE.	FALSE
	SCHEMA GEN	Provide TRUE if you want to create schema for this tables else FALSE. If no value is provided, default value is taken as TRUE.	Valid values: TRUE/FALSE.	TRUE
	SOURCE	Provide the source of table.	Valid values: BACKEND/MASTER/BOTH	MASTER
	PHYSICAL NAME	Provide the physical name of the table. If no physical name is provided, it will be derived from DISPLAY NAME		
	SURROGATE	Provide TRUE if you want to make the column surrogate key else provide FALSE.	Valid values: TRUE/FALSE.	FALSE
	IDPREFIX	Provide ID prefix if you want to generate your surrogate keys with the prefix.		
	SEQUENCE NAME	Provide sequence name for IDGEN creation		
	CACHED	Provide TRUE if you want to maintain this table in cache else FALSE	Valid values: TRUE/FALSE.	FALSE
	IS NATIVE	Provide TRUE if you want to make this table native else FALSE	Valid values: TRUE/FALSE.	FALSE
	NOTIFY	Provide email addresses separated by comma.		

Columns	COLUMN NAME	Provide display name of column		
	PHYSICAL NAME	Provide physical name of the column.		
	DATA TYPE	Provide data type of column	Valid values: Boolean, Char, Clob, Date, Decimal, Double, EncryptedString, File, Float, Integer, BigInt, JSON, PeriodDate, PeriodTimestamp, String, st_geometry, Timestamp, VarByte, Xml	
	LENGTH	Provide the length of the column	This property is valid for data types: String, Char, Decimal, EncryptedString, VarByte. For decimal data type, number of digits after decimal can be specified by appending number with underscore and count of digits after decimal. Eg. Length = 30_5	
	REQUIRED	Provide TRUE if you want the column to be a required field.	Valid values: TRUE/FALSE.	FALSE
	CASE SPECIFIC	Provide case specific value for STRING/CHAR columns	Valid values: TRUE/FALSE.	FALSE
	CHARACTER SET	Provide character set of the column.	Valid values: UNICODE/LATIN.	LATIN
	COMPRESSION STATEMENT	Provide the compression at column level. For compression statement user has to give data in this format e.g. string data type >> ('BR','HR','TEST').		
	DEFAULT VALUE	Provide default value of the column if any.		
	ENUM	Provide the type of ENUM	Valid values: Constant/Lookup/Reference	
	CONSTANT VALUE	If ENUM type is selected as Constant, Provide the constant values separated by comma		
	CODE SET	If ENUM type is selected as Code Set, provide the code set name on which lookup will get created.		
	REF SOURCE TABLE	If ENUM type is selected as Reference, Provide the table for reference		

Columns	REF SOURCE PROPERTY	If ENUM type is selected as <b>Reference</b> , Provide the physical column for reference		
	REF SOURCE DISPLAY PROPERTY	If ENUM type is selected as Reference, Provide the column name to be displayed for reference		
	CONSTRAINT VALUE	Provide the constraint value in XML form similar to what we have in mtt files.		
	DESCRIPTION	Provide description of the column		
	DICT NAME	Provide the dictionary name		
	DICT DATA TYPE NAME	Provide the dictionary data type name		
	SOURCE	Provide source of the table	Valied Values: BACKEND/MASTER	MASTER
	COLUMN DISPLAY NAME	Provide display name of column. Default display name will be column name.		
	EDITABLE	Provide FALSE if you want the column value to be non-editable else provide TRUE	Valied Values: TRUE/FALSE	TRUE
	HIDDEN	Provide TRUE if you want the column value to be hidden else provide FALSE	Valied Values: TRUE/FALSE	FALSE
	SORTABLE	Provide TRUE if you want the column value to be sortable else provide FALSE	Valied Values: TRUE/FALSE	TRUE
	SEARCHABLE	Provide FALSE if you don't want the column to be searchable else provide TRUE	Valied Values: TRUE/FALSE	TRUE
	NO WRAP	Provide TRUE if you want the column value to be wrapped else provide FALSE	Valied Values: TRUE/FALSE	FALSE
	I18NIZE	Provide TRUE if you want the column value to be i18nize else default value is taken as FALSE	Valied Values: TRUE/FALSE	FALSE
	ACTIVE	Provide false to deactivate column, default will be true.	Valied Values: TRUE/FALSE	TRUE

Keys	KEY NAME	Provide Name of the key. For composite key specified the column Name separated by comma.		
	KEY PHYSICAL NAME	Provide physical name of the key. If the physical name is not provided, it will be derived from KEY NAME.		
	KIND	Provide type of KEY you want to create.	Valid values: PRIMARY/UNIQUE/INDEX/UNIQUE_INDEX/NUPI	
	SELECTED COLUMN NAME	Provide COLUMN NAME on which the key will be defined. In case of composite key, the column names need to be separated by comma.		
	ACTIVE	Provide false to deactivate keys, default will be true.	Valied Values: TRUE/FALSE	TRUE

Links	LINK NAME	Provide Name of the Foreign Key link. Name should be unique.		
	TO DOCUMENT NAME	Provide the Logical name of the table to which FK link needs to be created		
	FROM COLUMN NAME	Provide the column display name of the column which is specified in property tab from which FK link needs to be created.		
	TO COLUMN NAME	Provide the logical name of the column to which FK link needs to be created.		
	REVERSE LINK	Provide TRUE if reverse link needs to be created else provide FALSE	Valid values: TRUE/FALSE.	FALSE
	REVERSE LINK NAME	Provide a reverse link name. If no name is provided, it will be derived from link name.		
	VALIDATION RULE	Provide TRUE if referential integrity default validation rules needs to be created else default value is taken as FALSE. Valid values: TRUE/FALSE.		FALSE
	ACTIVE	Provide false to deactivate link, default will be true.	Valid Values: TRUE/FALSE	TRUE

Facet Document	INPUT_TABLE_NAME	Provide the name of the input staging table		IN_<TableName>
	ERROR_TABLE_NAME	Provide the name of the error staging table		ERR_<TableName>
	MASTER_TABLE_NAME	Provide the name of the master staging table		MST_<TableName>
	OUTPUT_TABLE_NAME	Provide the name of the output staging table		OUT_<TableName>
	NETCHANGE_TABLE_NAME	Provide the name of the netchange staging table		NC_<TableName>
	VERSION_TABLE_NAME	Provide the name of the version staging table		VER_<TableName>
	IS_HARD_DELETE_ENABLED	Provide TRUE to enable hard delete on the table else provide FALSE	Valid values: TRUE/FALSE.	FALSE
	IS_EXTENDABLE	Provide TRUE if you want to extend column, default value is FALSE	Valid values: TRUE/FALSE.	FALSE
	EXT_STR_ATTR_COUNT	Provide the number of extended string columns, Default value is 5. IS_EXTENDABLE should be set to TRUE for this.		
	EXT_DEC_ATTR_COUNT	Provide the number of extended decimal columns, Default value is 0. IS_EXTENDABLE should be set to TRUE for this.		
	EXT_DATE_ATTR_COUNT	Provide the number of extended date columns, Default value is 0. IS_EXTENDABLE should be set to TRUE for this.		
	IS_TEMPORAL	Provide TRUE to make the table temporal else provide FALSE	Valid values: TRUE/FALSE.	FALSE



Facet Document	SELECT_TEMPORAL_TYPE	If the table is temporal select the type of temporal.	Valid values: TEMPORAL_VALID_TIME_DATE, TEMPORAL_VALID_TIME_TIMESTAMP, TEMPORAL_TRANSACTION_TIME, BI_TEMPORAL_DATE, BI_TEMPORAL_TIMESTAMP	
	VALIDTIME_COLUMN	Provide name of the validtime column.		
	TRANSACTION_TIME_COLUMN	Provide name of the transaction time column.		
	CREATE	Provide TRUE if you want to generate SOAP request for CREATE action else provide FALSE		FALSE
	READ	Provide TRUE if you want to generate SOAP request for READ action else provide FALSE		FALSE
	UPDATE	Provide TRUE if you want to generate SOAP request for UPDATE action else provide FALSE		FALSE
	DELETE	Provide TRUE if you want to generate SOAP request for DELETE action else provide FALSE		FALSE

## Create RDM Based View

RDM's model builder feature supports creation of RDM based views that are used to reference multiple tables and other databases (external database in same database server as RDM database user). Views can be defined as simple view pointing to RDM table or complex view supporting joins to multiple tables.

Creation of RDM based view is similar to creation of tables as in the above section. Under the Object Details tab, select the Object type as View, an additional tab - View SQL is displayed as in [Figure 270](#) wherein you can define the view details. The [Figure 270](#) displays the sample view SQL.

The view details entered under **View SQL** tab is created during the schema generation process.



Figure 270: Model Builder—View SQL

OBJECT DETAILS COLUMNS KEYS LINKS **OBJECT DEFINITION**

BACK SAVE ?

Object Definition \*

```
REPLACE VIEW DOCVIEW_PHYSICALNAME ( CUSTOM_COLUMNS1 , CUSTOM_COLUMNS2 , CUSTOM_C
SYS_TARGET_ID, SYS_AUTH_ID, SYS_SOURCE, SYS_CREATED_BY, SYS_CREATION_DATE, SYS_ENT_S
SYS_LAST_MODIFIED_BY, SYS_LAST_MODIFIED_DATE, SYS_NC_TYPE, SYS_ERR_CODE, SYS_ERR_SVI
SYS_APPROVAL_ID ) AS ( SELECT CUSTOM_COLUMNS1 , CUSTOM_COLUMNS2 , . . . . . CUSTOM_COLUM
NULL (INTEGER) AS SYS_TARGET_ID, NULL (VARCHAR(1000)) AS SYS_AUTH_ID, NULL (VARCHAR(256
SYS_SOURCE, NULL (VARCHAR(70)) AS SYS_CREATED_BY, NULL (TIMESTAMP(0)) AS SYS_CREATION_
'ACTIVE' (VARCHAR(25)) AS SYS_ENT_STATE, NULL (VARCHAR(70)) AS SYS_LAST_MODIFIED_BY, NUL
(TIMESTAMP(0)) AS SYS_LAST_MODIFIED_DATE, NULL (VARCHAR(25)) AS SYS_NC_TYPE, NULL (VARCH
AS SYS_ERR_CODE, NULL (VARCHAR(30)) AS SYS_ERR_SVDTY, NULL (VARCHAR(256)) AS SYS_APPRO
```

The view definition must adhere to the below template for proper functioning of RDM based views.

```
REPLACE VIEW < VIEW NAME>
(
View Column list
SYS_Column_List
)
As
(
Select
Custom column list...
SYS_Column_List
From Tablename
);
```

Once the view is saved, it will be in draft state and you can validate and generate the model. For details, see section: [Validate Models](#) and [Deploy Model](#).



The view definition will not be validated by Model Builder. It is the responsibility of the user to verify and validate the view definition before entering the view details under View SQL tab.

### Supported Operations on RDM Based Views

- Configurable UI can be created on document view. Search operation can be performed on the underlying view through configurable UI.
- The document view supports all standard X-commands.

## Auto Generate View

The Model Builder UI provide an option to auto generate view for the selected tables in a database on which the user has access to.

Perform the following steps to auto generate view:

- 1 On the **Model Builder** UI, select **Auto Generate Views** from **Actions** dropdown.

Figure 271: Auto Generate Views UI

Auto Generate Views UI

BACK GENERATE ?

GENERATE VIEWS VIEW GENERATION LOGS

Database Name \*  
EXT\_DB

Service Name \*  
Master

☐ Dirty Read

^ Tables In Selected Service

<input type="checkbox"/> Table Display Name	<input type="checkbox"/> Table Display Name
<input checked="" type="checkbox"/> AttributeSet	<input checked="" type="checkbox"/> AttributeSetDetail
<input type="checkbox"/> CUSTOMER_T	<input type="checkbox"/> CUSTOMER_WITHOUT_
<input type="checkbox"/> CUSTOMER_WITH_ALTER_ACCESS	<input checked="" type="checkbox"/> Code_Set
<input checked="" type="checkbox"/> Code_Set_Stats	<input checked="" type="checkbox"/> Code_Set_Table_Associal

- 2 On the **Auto Generate Views UI** ([Figure 273](#)), under the **Generate Views** tab:
  - **Database Name:** the Database Name dropdown lists all the databases to which the user has create access (that is, the database user where RDM is installed should have create view access on the target database where the view needs to be deployed). Select the database name where the views need to be generated.
  - **Service Name:** select the service name (Master, Error, Audit, Input, Netchange, Output or Version) from Service Name dropdown.

The Tables pane displays the list of tables available in the selected service.
  - In the Tables pane, select all the tables for which the view need to be generated.
  - **Dirty Read:** two types of views can be generated.
    - With Locking Row Access in the DDL: select the Dirty Read checkbox to generate views with locking row access in DDL.
    - Without Locking Row Access: if the Dirty Read option is not selected, Teradata (database) will apply a read lock when reading data from the view.

- 3 Click **Generate**.

Message: Views generated successfully is displayed.

On the **Auto Generate Views UI**, the checkboxes corresponding to the tables for which the views were generated will be selected by default and will not be editable as in [Figure 273](#).

Figure 272: Auto Generate Views UI—View Generation Logs

Auto Generate Views UI
BACK GENERATE ?

GENERATE VIEWS
VIEW GENERATION LOGS

✓ View generated successfully.

Database Name
EXT\_DB

Service Name
Master

☐ Dirty Read

^ Tables In Selected Service

☐ Table Display Name
☐ Table Display Name

x

☒ AttributeSet
☒ AttributeSetDetail

☐ CUSTOMER\_T
☐ CUSTOMER\_WITHOUT\_

☐ CUSTOMER\_WITH\_ALTER\_ACCESS
☒ Code\_Set



If the structure of a table for which view were already generated is modified or if ISG is executed for the table, then for the changes to be reflected in View, the view must be regenerated.

Known limitation on auto generate views:

- Auto generate views does not provide option to create and deploy view in a different database server.
- Option to specify Join's in view definition is not supported.

- All columns of the table will be part of the view created for that table and option to choose attributes is not available.

### Auto Generate Views Log

Once the generation of views is completed successfully, click **View Generation Logs** tab to view the logs of the generated view as in [Figure 273](#).

Figure 273: Auto Generate Views UI—View Generation Logs

Auto Generate Views UI

BACK ?

GENERATE VIEWS

VIEW GENERATION LOGS

Model Builder View Generation Logs: Page 1 of 1

View Name	Database Name	TableName	Service Name
<div>X</div> <div>Q</div>	<div>X</div> <div>Q</div>	<div>X</div> <div>Q</div>	<div>X</div>
MST_CODE_SET_View	EXT_DB	Code_Set	BCM_MASTER
MST_CODE_SET_STATS_View	EXT_DB	Code_Set_Stats	BCM_MASTER
MST_CODE_SET_TABLE_ASSOCIATION_View	EXT_DB	Code_Set_Table_Association	BCM_MASTER
GP_MST_GRP_ATTR_SET_DETAIL_View	EXT_DB	AttributeSetDetail	BCM_MASTER
GP_MST_GRP_ATTR_SET_View	EXT_DB	AttributeSet	BCM_MASTER

→

1 - 5 of 5

<

>

>>

The View Generation Logs display all the view DDL execution details as below:

- View Name: the view name will be physical table name\_view and the maximum size is 100.
- Database Name, Table Name and Service Name on which the views were generated.
- Created By and Creation Date: displays the view creator name and creation date and time.
- Refresh Date and Refreshed By: displays the refresh details when the views are regenerated.
- Status: if the view generation is completed successfully for that table, the status corresponding to that table name, service name and database name will be displayed as

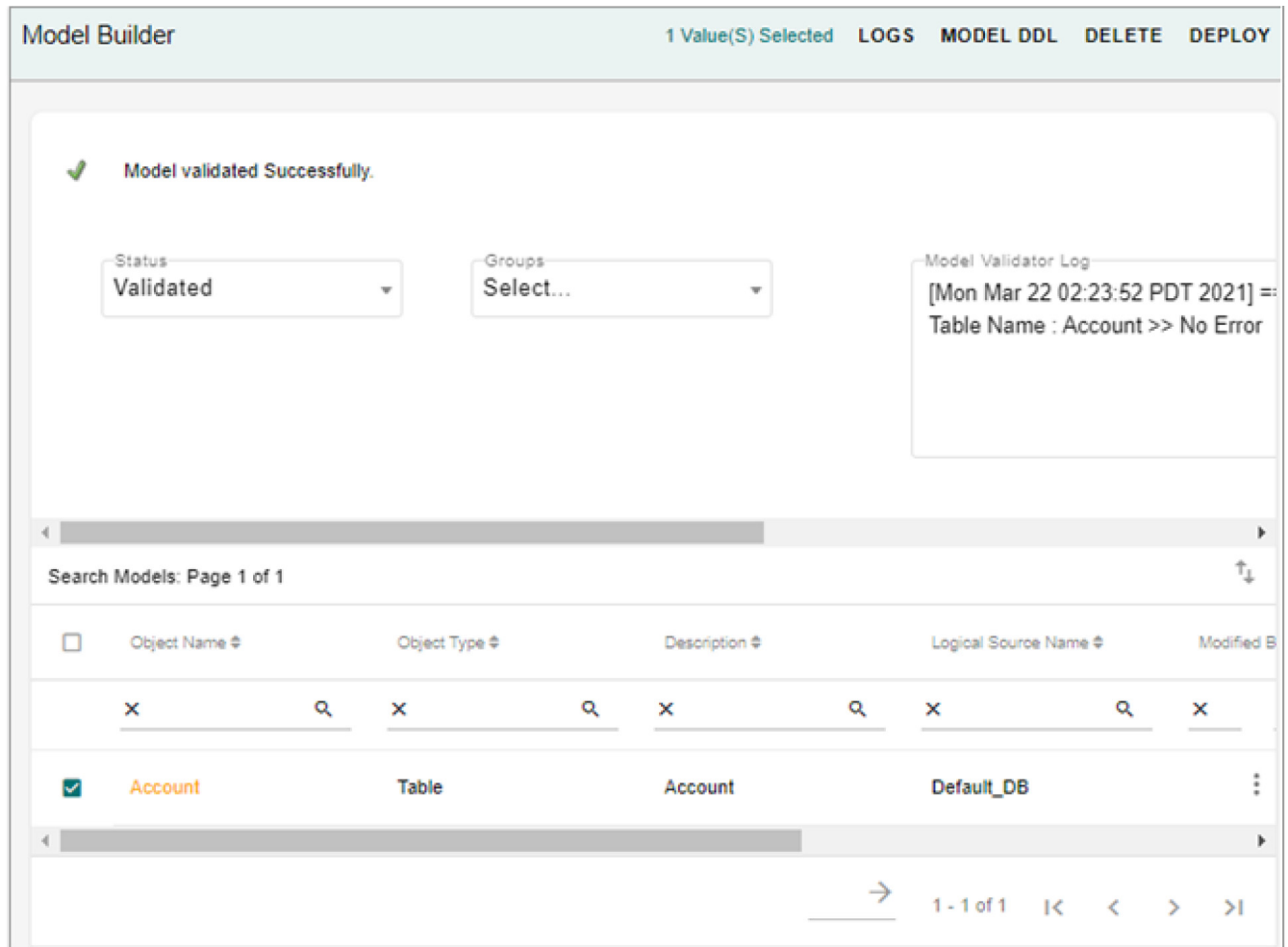
success. If the view generation failed for any table(s) then the status is displayed as failed for that table(s).

## Validate Models

Once the model is in draft state, the Validate Model process can be triggered.

- 1 On the **Model Builder** UI, select Account Model and click **Validate**.

Figure 274: Model Builder



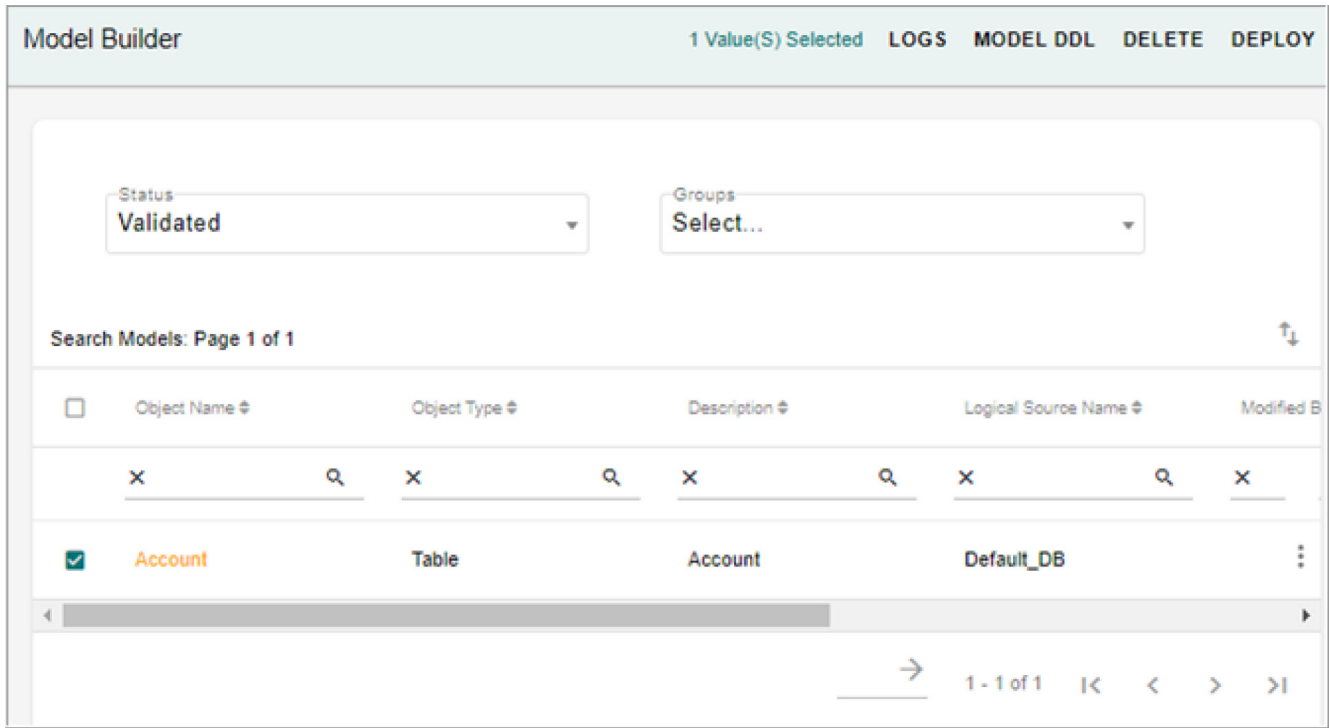
Once validation is completed, logs will be displayed to check for any errors. If there is an approval process defined, model will be moved to Pending Approval models. In this scenario, there is an approval defined. At this point, the approver has to login and access the Approval Inbox from where the DDL can be reviewed and approved or rejected.

Once approved, it will be moved to Validated Models. If there is no approval process defined, it will go to Validated Models immediately.

## Deploy Model

- Once model is validated, the Schema generation process can be triggered which will create the physical database objects and update the corresponding metadata tables.
- 1 On the **Model Builder** UI with Validated Models option selected, select the Account Model and click **Deploy** as in [Figure 275](#).

Figure 275: Model Builder



## Deployment Logs

- 1 On the **Model Builder** UI, select the Account Model and click **Logs** to monitor the status of ISG process.

Figure 276: Model Builder ISG Logs

Execution Id	Deployment Status	Source Name	Generated Tables	Start Time
2	ISG IN PROGRESS	RDM	Account	03/22

- 2 On the **Model Deployment Logs** UI (Figure 276), click on the Menu icon corresponding to the required execution Id of the log and click **Log Details** to view the complete deployment log information.

The **Model Deployment Log Details** UI displays the log information incrementally for each request or SQL executed by the ISG process. The details are displayed under two different tabs as **Log Status** and **Pre/Post ISG Count** as in Figure 277.

- **Log Status:** Log Status tab displays the log details for each request or SQL execution along with the status incrementally in two tabular sections as Executed Requests and Executed SQLs as in Figure 277.



Figure 277: Model Deployment Log Details

Model Deployment Log Details			BACK   DOWNLOAD LOGS   ?	
LOG STATUS			PRE/POST ISG COUNT	
Execution Id 2				
Executed Requests			Executed SQLs	
ISG Log Details			SQL Statements	
Parameter Name	Status	Operation Start Time	Execution Time	SQLs
Final Status	SUCCESSFUL	03/22/2021 02:3		
customPrelsgReq	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:12	CREATE MULTISET TABLE RDMUSR
Schema Generation	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:12	REPLACE VIEW MST_ACCOUNT (AC
Create Activity	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:12	COMMENT ON TABLE RDMUSRTST_
Populate SysTableMap	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:12	ALTER TABLE RDMUSRTST_MST.MS
Add Model Template	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	ALTER TABLE RDMUSRTST_MST.MS
Populate FacetInfo	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	CREATE TABLE RDMUSRTST_MST.E
Populate Column Map	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	REPLACE VIEW ERR_ACCOUNT AS :
RefreshSysDBC	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	COMMENT ON TABLE RDMUSRTST_
IncrementalLoadValues	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	ALTER TABLE RDMUSRTST_MST.ER
Business Validation	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	CREATE TABLE RDMUSRTST_VER.V
RefreshLoadTables	SUCCESSFUL	03/22/2021 02:3	03/22/2021 02:36:13	REPLACE VIEW VER_ACCOUNT AS :

### Executed Requests

The Executed Requests section displays the list of all the requests executed and provides the following details as in [Figure 278](#).

- **Parameter Name:** displays the name of each request or step executed.
- **Status:** displays the status as either Successful or Failed based on the execution result. The “Successful” status indicates that the step was executed correctly without any errors. The “Failed” status indicates that an error had occurred during step execution. The status of each request execution is displayed as soon as its execution is complete.

Figure 278: Model Builder ISG Log Details

Model Deployment Log Details

LOG STATUS

PRE/POST ISG COUNT

Execution Id

2

Executed Requests

ISG Log Details

Parameter Name	Status	Operation Start Time	Operation End Time	Error De
Final Status	SUCCESSFUL	03/22/2021 02:36:08	03/22/2021 02:37:14	NA
customPrelsgReq	SUCCESSFUL	03/22/2021 02:36:09	03/22/2021 02:36:09	NA
Schema Generation	SUCCESSFUL	03/22/2021 02:36:10	03/22/2021 02:36:14	NA
Create Activity	SUCCESSFUL	03/22/2021 02:36:14	03/22/2021 02:36:14	NA
Populate SysTableMap	SUCCESSFUL	03/22/2021 02:36:14	03/22/2021 02:36:15	NA
Add Model Template	SUCCESSFUL	03/22/2021 02:36:15	03/22/2021 02:36:15	NA
Populate FacetInfo	SUCCESSFUL	03/22/2021 02:36:15	03/22/2021 02:36:15	NA
Populate Column Map	SUCCESSFUL	03/22/2021 02:36:15	03/22/2021 02:36:15	NA
RefreshSysDBC	SUCCESSFUL	03/22/2021 02:36:15	03/22/2021 02:36:20	NA
IncrementalLoadValues	SUCCESSFUL	03/22/2021 02:36:20	03/22/2021 02:36:24	NA
Business Validation	SUCCESSFUL	03/22/2021 02:36:20	03/22/2021 02:36:20	NA
RefereshLoadTables	SUCCESSFUL	03/22/2021 02:36:24	03/22/2021 02:36:30	NA

- **Operation Start Time and End Time:** displays each request execution start and end time.
- **Error Detail:** in case of request execution failure, the Error Detail column displays the reason for execution failure.

You can mouse hover the error message to see the complete error or click on the error message link, a pop-up displays the complete error.

#### Executed SQLs

The Executed SQLs section displays the list of all the SQLs executed during ISG process and provides the following details as in [Figure 279](#).

- **Execution Time:** displays each SQL execution duration time and date.
- **SQLs:** displays the SQL executed.
- **Status:** displays the status as either Success or Failed based on the SQL execution result. The status of each SQL execution is displayed as soon as its execution is complete.

Figure 279: Model Deployment Log Details

BACK   DOWNLOAD LOGS   ?		
Executed SQLs		
SQL Statements		
Execution Time	SQLs	Status
X Select All ▼ 🔍		
03/22/2021 02:36:12	CREATE MULTISET TABLE RDMUSRTST_MST.MST_...	Success
03/22/2021 02:36:12	REPLACE VIEW MST_ACCOUNT (ACCOUNTID , AC...	Success
03/22/2021 02:36:12	COMMENT ON TABLE RDMUSRTST_MST.MST_ACCOU...	Success
03/22/2021 02:36:12	ALTER TABLE RDMUSRTST_MST.MST_ACCOUNT AD...	Success
03/22/2021 02:36:13	ALTER TABLE RDMUSRTST_MST.MST_ACCOUNT AD...	Success
03/22/2021 02:36:13	CREATE TABLE RDMUSRTST_MST.ERR_ACCOUNT (...	Success
03/22/2021 02:36:13	REPLACE VIEW ERR_ACCOUNT AS SELECT * FRO...	Success
03/22/2021 02:36:13	COMMENT ON TABLE RDMUSRTST_MST.ERR_ACCOU...	Success
03/22/2021 02:36:13	ALTER TABLE RDMUSRTST_MST.ERR_ACCOUNT AD...	Success
03/22/2021 02:36:13	CREATE TABLE RDMUSRTST_VER.VER_ACCOUNT (...	Success
03/22/2021 02:36:13	REPLACE VIEW VER_ACCOUNT AS SELECT * FRO...	Success
03/22/2021 02:36:14	ALTER TABLE RDMUSRTST_VER.VER_ACCOUNT MO...	Success

- **Pre/Post ISG Count:** displays the list of the Pre-ISG count and Post-ISG count as in [Figure 280](#). In case of any error in metadata count calculation, the status is displayed as error.
- **Parameter Name:** displays the list of items involved in ISG process like tables,

views, left navigation entries, publication objects, indexes, stored procedures, user activities, business rules etc.,

- **Pre ISG Count:** displays the count of each item involved in ISG prior to ISG execution.
- **Post ISG Count:** displays the count of each item involved in ISG after ISG process.

Figure 280: Model Builder ISG Log Details

Model Deployment Log Details			BACK	DOWNLOAD LOGS	?
LOG STATUS			PRE/POST ISG COUNT		
Execution Id			2		
Count Details					
Parameter Name	Pre Deployment Count	Post Deployment Count			
Business Rules	42	42			
Cumulative Columns	18562	18718			
Cumulative Tables	668	673			
Cumulative Views	207	210			
Default Business Rules	2313	2326			
Error Tables	46	47			
Facets	4616	4687			
Foreign Keys	130	130			
GT Tables	46	47			
Indexes	1272	1280			

- **Export to Excel:** using the export to excel option on **Executed Request UI** section, **Executed SQLs UI** section and on **Pre/Post ISG Count UI**, you can export the log details individually to an excel file.
- **Download ISG Logs:** using the **Download ISG Logs** button, you can export all the

log details in a single excel file under different tabs. The Download ISG Logs option is enabled only when the entire ISG process is completed.

## API to Validate and Generate ISG

The following API requests are used to validate and generate ISG for models created using model builder. The APIs are available in modelBuilderCustomRulesInIsrg.xml file at:

`<RDM_Install_Directory>\cfg\xservice\mdmservices\modules\modelBuilder\rules`

### API to validate model:

```
<REQUEST Name="validateDraftModels_Req" ServiceName="RDMServices">
  <DOCUMENT Value="Employee"/>
</REQUEST>
```

### API to generate ISG for model:

```
<REQUEST Name="triggerModelBuilderISG_Req"
ServiceName="RDMServices">
  <DOCUMENT Value="Employee"/>
  <DOCUMENT Value="Customer"/>
</REQUEST>
```

### API to generate Synchronous ISG for model:

```
<REQUEST Name="triggerRuntimeSynchronousISGReq"
ServiceName="RDMServices">
  <DOCUMENT Value="Employee"/>
  <DOCUMENT Value="Customer"/>
</REQUEST>
```

In all the above APIs, the input is the Model name provided in Document Value tag. Example  
`<DOCUMENT Value="Employee"/>`

### For validate draft Models Request, the response is as below:

```
<RESPONSES Status="Success">
  <RESPONSE Status="Success">
    <Status Value="Success"/>
    <Description Value="Model Validated Successfully"/>
    <SELECTED_IDS>
      <SELECTED_ID Value="31"/>
    </SELECTED_IDS>
    <modelValidatorLog Value="Fri Oct 06 15:29:59 IST 2017--&gt;
Model Validator Logs : TABLE NAME : Employee --&gt; Warning, Renaming
unique key name since original is already taken
Details ...
Unique Key : ::models/Toolkit Models/CCVC [Unique Key] pk
```

```
Note : Original name was, pk. New name is, pk0

Warning, Renaming unique key name since original is already taken
Details ...
Unique Key : ::models/Toolkit Models/CCVC [Unique Key] pk
Note : Original name was, pk. New name is, pk1

"/>
</RESPONSE>
</RESPONSES>
```

**For trigger model builder ISG request, the response is as below:**

```
<RESPONSES Status="Success">
  <RESPONSE Status="Success">
    <Status Value="Success"/>
    <Description Value="ISG_REQUEST_IS_FIRED_CHECK_LOG"/>
    <SELECTED_IDS>
      <SELECTED_ID Value="31"/>
    </SELECTED_IDS>
  </RESPONSE>
</RESPONSES>
```

## Modify Table Primary Key

MDM's Model Builder UI provides an option to modify the primary key of the selected table. You can modify the primary key of a table in any model state (draft, validated or generated model state).



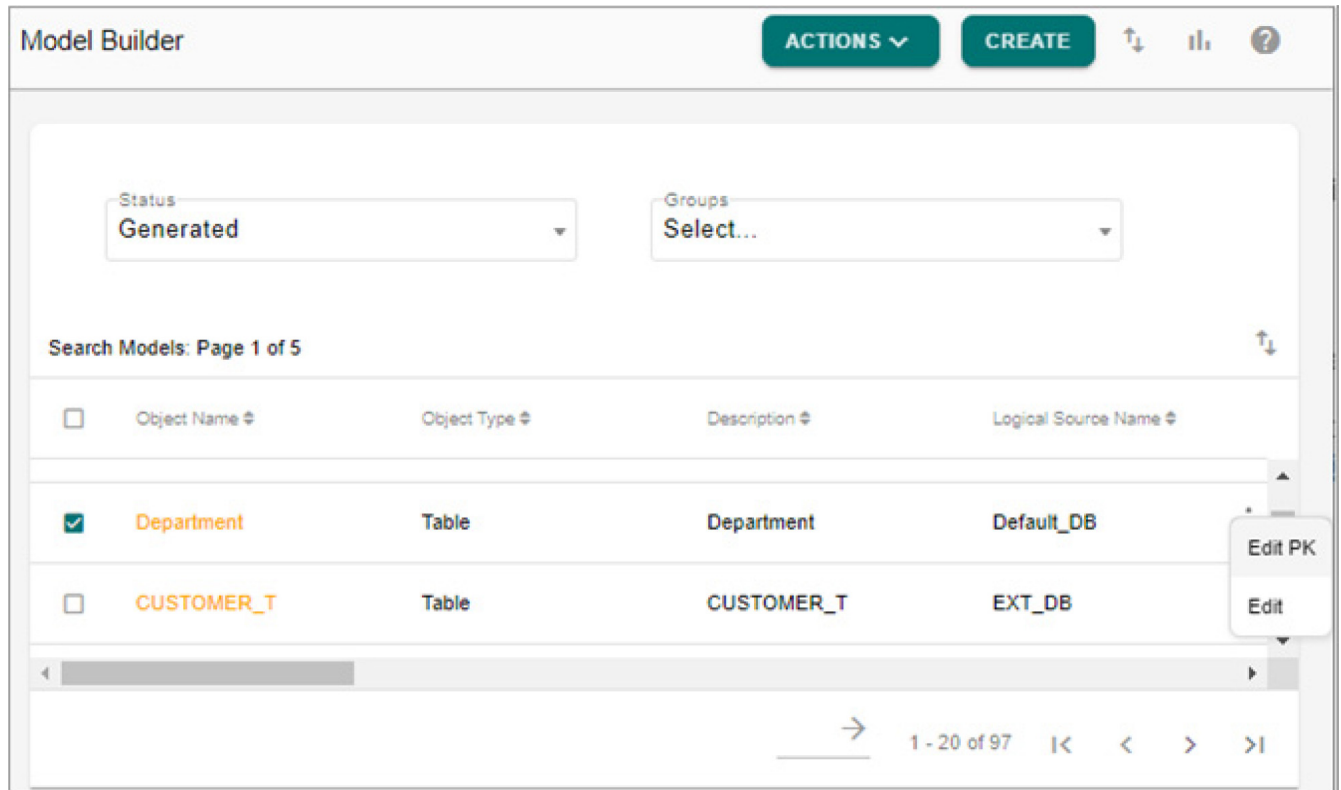
For any user to modify the Primary key of the table, the user must be assigned to a role for which the Model Builder Edit Primary Key Activity is added. The Edit PK option will not be displayed on the Model Builder UI if the Model Builder Edit Primary Key Activity is not assigned to the particular role.

---

Perform the following steps to modify the table PK value:

- 1 On the **Model Builder** UI, click on the Menu icon corresponding to any data model and click **Edit PK**.

Figure 281: Model Builder



The Object Details UI is displayed.

- 2 On the **Object Details** UI, click **Keys** tab.

The **Object Details—Keys** UI displays the existing PK of the selected table as in [Figure 282](#).

Figure 282: Model Builder

The screenshot shows the 'KEYS' tab in the Model Builder UI. At the top, there are four tabs: 'OBJECT DETAILS', 'COLUMNS', 'KEYS' (which is active), and 'LINKS'. To the right of the tabs are buttons for 'BACK', 'DELETE', and 'SAVE' (in a green box), along with a '+' icon and a help icon '?'. Below the tabs is a search bar labeled 'Search Model Keys'. Underneath the search bar is a table with the following columns: 'Name', 'Physical Name', 'Type', and 'Available Columns'. The table has one row with the following values: 'DepKey' in the 'Name' column, 'DepKey' in the 'Physical Name' column, and 'PK' in the 'Type' column. To the right of the table, the 'Available Columns' dropdown menu is open, showing a list of columns: 'Description', 'Id', and 'Name'. The 'Id' column is currently selected. At the bottom of the table, there is a horizontal scrollbar.

3 The Object Details—Keys UI, you can modify the PK value and click **Save**.

The following tasks are performed during save of PK changes:

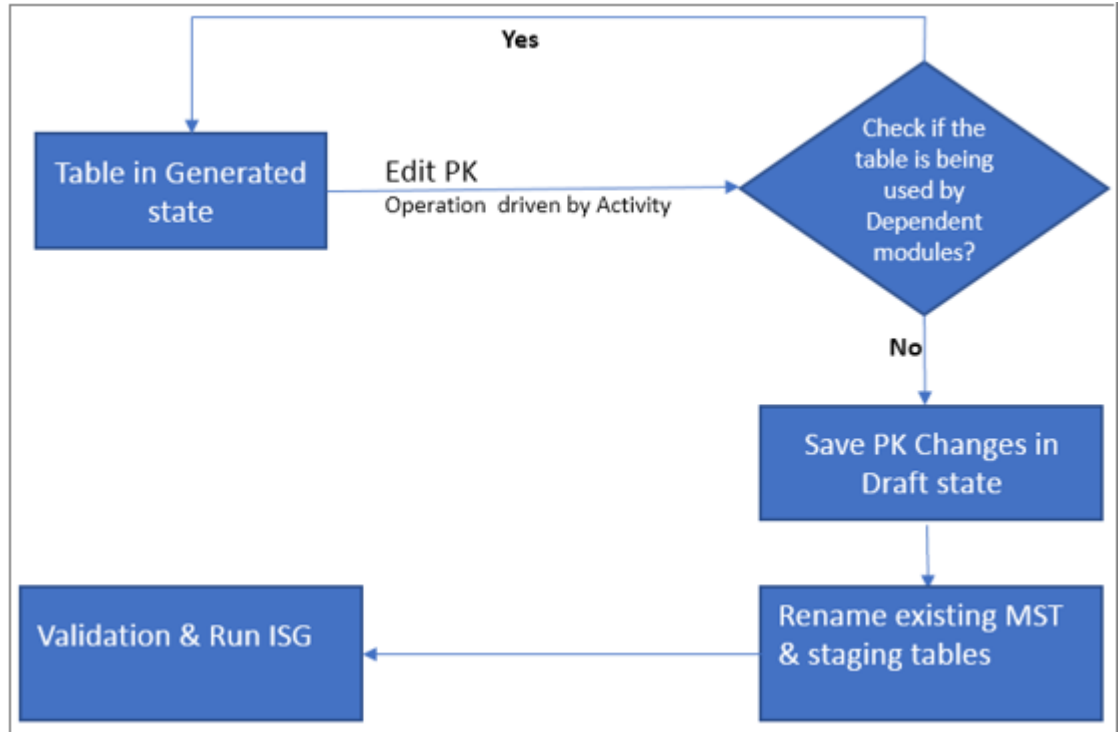
- The existing master table and all its staging tables (Input, Netchange, Error, Master, Output, Version, Audit Trail and load tables) will be renamed.
- Clean-up of some existing meta-data of the master table. Backup will be created for all the staging tables (Input, Netchange, Error, Master, Output, Version, Audit Trail and load tables).

The model will now move to Draft state and you can perform regular validation process and run normal ISG to create fresh tables. For details, see section: [Validate Models](#) and [Deploy Model](#).

The [Figure 283](#) displays the flow diagram of the Edit PK process.



Figure 283: Model Builder—Edit PK Flow



### Scenarios where PK Change not Allowed

The below points list the scenarios where PK change is not allowed:

- **Approval:** if there are in-flight approval records for the table, PK change is not allowed. In-transit records have to be cleared before initiating PK change.
- **Matching:** if matching profiles are created on the table, PK change is not allowed.
- **Survivorship:** if survivorship profiles and ALCR are defined on the table, PK change is not allowed.
- **Hierarchy:** if RO is created on the table, PK change is not allowed.
- **Grouping:** if grouping is defined on the table, PK change is not allowed.
- **Publication:** if publication object is defined on the table, PK change is not allowed.

### Scenarios not Handled in PK Change

The below points list the scenarios not handled in PK change:

- **Configurable UI:** customized configurable UI re-creation is not handled in edit PK process.
- **Business Rules:** custom business rules not handled in edit PK process.  
The out of the box Configurable UIs and Business rules are re-generated with ISG after the PK change is made.

### Scenarios where PK Change is not Supported

The below points list the scenarios where PK change is not supported:

- The PK change is not supported on Hierarchy and LRDM template creation.

- The PK change is not supported on a table if the table has FK reference to any other tables. Though PK change UI may allow to edit and save PK, undesired issues may occur in modules that use the table or the parent/child table referred by it.

## Scenarios where PK Change is Supported

The below points list the scenarios where PK change is supported:

- **Master Tables:** PK change is supported on standalone master tables.
- **Model Builder UI:** PK change is supported only through model builder UI. Excel upload or Import does not support PK change.

## Points on Backup due to PK Change

The below points list some of the important information on backup data due to PK change:

- The data from the renamed tables must be restored manually.
- The backup tables need to be cleared before initiating another PK change else the rename of tables will not be allowed. Valid error message will be displayed to the user.

## Delete Model and Delete Model Metadata

Model builder allows deletion of models in all states (Draft, Validated and Generated).

If models are in Draft and Validated states and were never generated, delete will simply soft delete the metadata of the models.

If models are in Draft, Validated or Generated state and underlying physical tables are generated, delete of these models will require delete metadata activity (Delete Table Metadata) to be added. On successful deletion, the underlying metadata and physical tables will be renamed and a new table with same name can be recreated later.

To delete a model:

- 1 On the **Model Builder** UI, select any data model and click **Delete** as in [Figure 284](#).



- The Delete button is visible always on the Draft and Validated UI.
  - Models with underlying physical tables cannot be deleted unless activity (Delete Table Metadata) to delete metadata is added to user role to perform metadata delete.
-

Figure 284: Model Builder

The screenshot shows the Model Builder interface. At the top, there's a header bar with 'Model Builder' on the left and '1 Value(S) Selected', 'LOGS', 'DELETE', and 'VALIDATE' on the right. Below the header, there are two dropdown menus: 'Status' set to 'Draft' and 'Groups' set to 'Select...'. Below these is a search bar with the text 'Search Models: Page 1 of 1'. The main area contains a table with the following columns: 'Object Name', 'Object Type', 'Description', and 'Logical Source Name'. The table has one row with the following data: 'Department' (checked), 'Table', 'Department', and 'Default\_DB'. At the bottom right of the table, there are pagination controls showing '1 - 3 of 3' and navigation arrows.

Object Name	Object Type	Description	Logical Source Name
<input checked="" type="checkbox"/> Department	Table	Department	Default_DB

2 On the **Confirmation Webpage** dialog, click **Yes**.

Message: “Model deleted successfully” is displayed.

Message: “Models deleted successfully. Check Email or Metadata Deletion logs for more details” is displayed if it is metadata delete.

If the table is not used by any MDM modules (Configurable UI, Business Rules, Business Entities, ALCR, Authorization, Approval, E2E, Matching, Survivorship, Publication and Hierarchy) deletion process is performed else an error message is displayed as “Model Deletion Failed for Some Table(s). Check Email or Metadata Deletion logs for more details.”

During the metadata delete, the table names of staging tables (Mater, Error, Audit trail, Version, Input staging, Netchange, Output staging) and load table will be renamed to tablename\_timestamp. The logical name of the table will be renamed to TableName\_MD\_objectId. You can then create a new table with same name as that of a metadata deleted table.

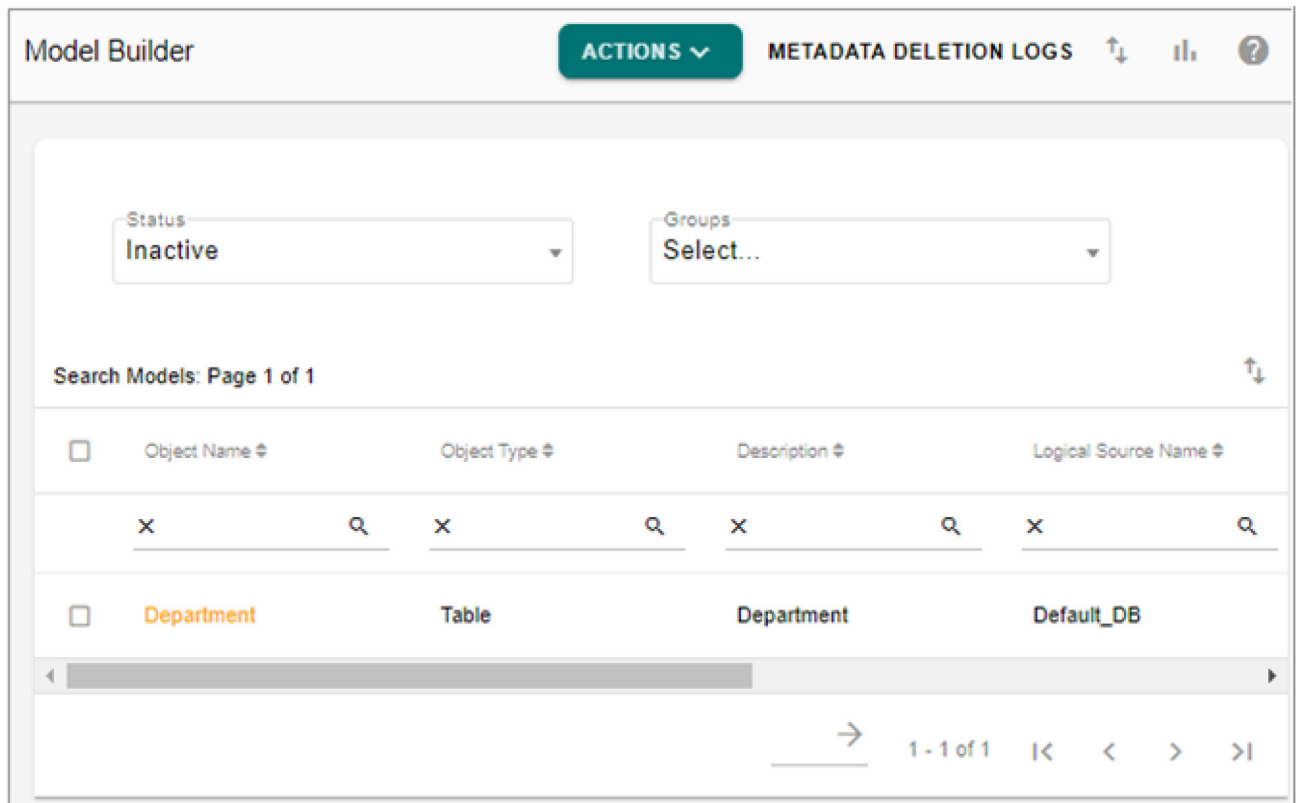


- Metadata deletion is not supported if the table (as Parent Table) has a foreign key link with a child table.
- Metadata deletion is not supported for template based master tables and OOTB tables.

- For metadata deletion on a temporal table, you must drop the join indexes if any created on temporal table through backend.
- Delete model will not check dependencies of external views or tables.
- For successful metadata deletion of model builder tables used in survivorship, you must delete the survivorship profile along with the relational object (RO) and cross references.

3 On the **Model Builder** UI, select the **Inactive** option from **Status** dropdown as in [Figure 285](#).

Figure 285: Model Builder



4 Click **Metadata Deletion Logs** to view the metadata deletion log details as in [Figure 286](#).

Figure 286: Metadata Deletion Logs

Metadata Deletion Logs						BACK
Search Metadata Deletion Logs: Page 1 of 1						
TableName	Physical Table Name	Renamed Table Name	Metadata Deletion Time	Status	Details	
<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	
Department	-NA-	-NA-	2/18/2021 8:26 AM	Success	Model deactivated success	
Department	-NA-	-NA-	2/18/2021 7:50 AM	Success	Model deactivated success	

On the Model Builder UI with Inactive option selected, the deleted models can also be activated and can also export the model details to an excel sheet.

- To activate the deleted model, select the deleted model and click **Activate Model** to reactivate the model.

Message: “Model activated successfully” is displayed and the model will be displayed under Draft state.

Figure 287: Model Builder

Model Builder

1 Value(S) Selected

ACTIVATE MODEL

Status  
Inactive

Groups  
Select...

Search Models: Page 1 of 1

☐

Object Name

Object Type

Description

Logical Source Name

x

Q

x

Q

x

Q

x

Q

☒

Department

Table

Department

Default\_DB

1 - 1 of 1

|<

<

>

>|



The model names of reactivated models can be edited as per the requirement.

6 Click Export to Excel icon to export the model details to an excel file.

Table 12 displays the delete and metadata delete use case scenarios.

Table 12: Delete and Delete Metadata Use Cases

Object Type	Generated		Validated		Draft	
	Metadata Delete	Normal Delete	Metadata Delete	Normal Delete	Metadata Delete	Normal Delete
MDM Table	✓ If Metadata Deletion Activity is added	✗ If Metadata Deletion Activity is <b>Not</b> added	✓ If Metadata Deletion Activity is added	✓ If Metadata Deletion Activity is <b>Not</b> added	✓ If Metadata Deletion Activity is added	✓ If Metadata Deletion Activity is <b>Not</b> added
Non MDM Table/View	✗	✗	✗	✗	✗	✗
MDM View	✗	✗	✗	✓	✗	✓
Tables/Views deployed using secondary connection	✓ Only logical name of the table will be renamed	✗	✓ Only logical name of the table will be renamed	✓	✓ Only logical name of the table will be renamed	✓

## Export Model

RDM's Model builder UI provide an option to export any data model into excel format. You can export the model data from any of the model stage (draft, pending approval, validated, generated or Deleted model state). To export model data, perform the following steps:

1 On the **Model Builder** UI, select any data model and click Export to Excel icon.

The confirmation pop-up displaying option to Open or Save the Tablename\_model\_data\_export.xlsx is displayed.

- 2 On the confirmation pop-up, click **Open** to view the records in Excel or click **Save** to save the data in excel file.

For details on the various fields in excel file, see [Section : “Template Guidelines”](#)

## Import from X-Document

Tables can be imported from X-document into model builder. On import, the tables will be available in Draft state. Default facet will be added to the model, which is required for ISG through model builder. You can modify the required changes, validate and deploy model.



- You can import a table from X-document only once and incremental changes are not supported.
- The format of X-Document should be similar to MST X-Document file.
- During import from X-document, model column field attributes are treaded as non case specific.

To import X-Document:

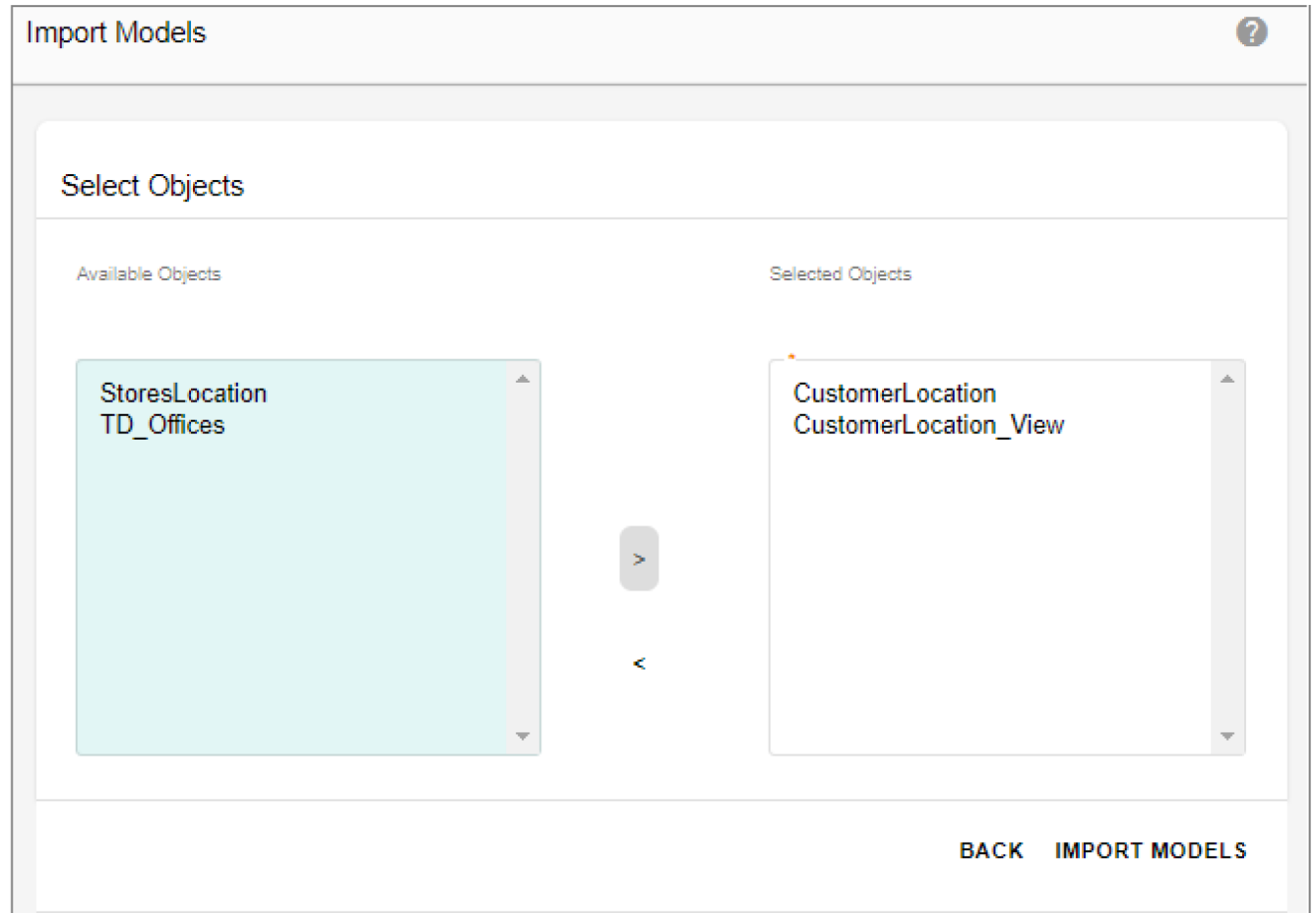
- 1 On the **Model Builder** UI, select **Upload** from **Actions** drop-down.  
The **Upload** UI is displayed.
- 2 On the **Upload** UI, select the “Upload Xdoc File” option, click **Choose File** to select the X-doc file and then click **Upload** as in [Figure 288](#).

Figure 288: Upload X-Doc File

The **Import Models** UI is displayed. On the **Import Models** UI, the **Available Objects** list displays all the available tables in the selected X-doc file.

- 3 On the **Import Models** UI, select the required tables that you want to import from the **Available Objects** list and move it to **Selected Objects** list as in [Figure 289](#)

Figure 289: Import Models



4 Click **Import Models**.

Message: “Model’s imported successfully” is displayed. The imported tables will be in Draft state as in [Figure 290](#) and the subsequent steps to validate the tables and generate ISG for the tables remains the same. For details, see [Section : “Validate Models”](#) and [Section : “Deploy Model”](#).



Figure 290: Model Builder

Model Builder

**ACTIONS** **CREATE**

✓ **Model's Imported Successfully.**

Status: **Draft** Groups: **Select...**

Search Models: Page 1 of 1

<input type="checkbox"/>	Object Name	Object Type	Description	Logical Source Name	
<input type="checkbox"/>	CustomerLocation	Table	CustomerLocation	Default_DB	⋮
<input type="checkbox"/>	CustomerLocation_View	Table	CustomerLocation_View	Default_DB	⋮

1 - 4 of 4

## Manage Dictionaries

The Dictionaries are standard repositories of data types and its related details such as name, length, physical name, required filed, character set, default value, Enum, Constant, Description etc. Normally dictionaries are used to maintain data type standards and naming conventions for implementations. While defining a table, you can refer to the already stored data types in dictionary. When a data type is selected for a column in table, all the details defined for the data type will be inherited by the column.

Navigate to **Manage Dictionary** UI from Model Builder page by selecting **Dictionary** from **Actions** dropdown.

On the **Manage Dictionary** UI (Figure 291), you can create, edit or delete dictionaries.

Figure 291: Manage Dictionary

Manage Dictionary

BACK

CREATE

?

Search Dictionary: Page 1 of 1

Dictionary Name	Description	Created By	Creation Date
<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>	<input type="text" value="x"/>
System			
WfProcess			
Query			
abc			
Batch_Validation			
MDRM			
Generic			
Entity			
LRDM			

1 - 9 of 9

<<

<

>

>>

- 1 On the **Manage Dictionary** UI, to create a dictionary, click **Create** and under the **Dictionary Details** tab, enter the basic details like Name and Description as in [Figure 292](#).

Figure 292: Manage Dictionary—Dictionary Details

DICTIONARY DETAILS    DICTIONARY PROPERTIES

Dictionary Info

Dictionary Name \*  
SampleDictionary

Description  
SampleDictionary

BACK    SAVE    ?

- 2 Under the **Dictionary Properties** tab(Figure 293), specify the data type details similar to Column creation details and click Save icon. For details on each field, see [Section : “Create or Edit Model.”](#)

Figure 293: Manage Dictionary—Dictionary Properties

DICTIONARY DETAILS    **DICTIONARY PROPERTIES**

BACK    WHERE USED    DELETE    SAVE    +    ?

Search Dictionary Properties

<input type="checkbox"/>	Name *	Type	Length	Extend	Required	Case Specific	Character
<input type="checkbox"/>	Department	String	250	EXTEND	<input type="checkbox"/> Required	<input type="checkbox"/> Case Specific	Select

- Once saved, data types defined in the dictionary can be used during model creation—Column specification.
- 3 On the **Manage Dictionary** UI, to edit existing dictionary details, click on the Menu icon corresponding to the required dictionary and click **Edit** and you can modify the values under different tabs and click Save icon.
- 4 On the **Manage Dictionary** UI, to delete existing dictionary details, click on the Menu icon corresponding to the required dictionary and click **Delete**.

## Manage Templates

The Manage Template feature is used to generate additional RDM Code Set Tables and Hierarchy Nodes and Relationship Tables to be used in Code Set management and Hierarchy Creation.

- 1 On the **Model Builder** UI, select **Template** option from **Actions** dropdown.

Figure 294: Manage Template

Manage Template 3 Value(S) Selected **VALIDATE TEMPLATE** **GENERATE MODEL**

Template Type: **RDM Template** ☐ SourceMap Template

Enter number of tables to be generated.\*  
2

Search Templates: Page 1 of 1

<input type="checkbox"/>	Template Name	Folder Name	Generated Table Count	Modified By
<input checked="" type="checkbox"/>	Reference_Code	LRDM	0	
<input checked="" type="checkbox"/>	Reference_Desc	LRDM	0	
<input checked="" type="checkbox"/>	Standard_Map	LRDM	0	

1 - 3 of 3

On the Manage Template UI (Figure 294), by default, the **RDM Template** option is selected and all the tables defined under the selected template is displayed in the Search Templates pane.

- 2 On the Manage Template, select the desired template and enter the tables to be generated and click Generate Model.

When RDM Template option is selected, option to create Source Map table is provided in addition to the system tables of RDM. To create Source map tables, select the **SourceMap Template** checkbox and enter number of tables to be generated.

The process of model generation is an asynchronous process. An email will be sent to the user with the status of table generation. Maximum 50 templates can be generated at a time. In case of any error during table generation, entire changes will be rolled back.

Once the ISG is completed successfully, the **Generated Table Count** column on the **Manage Template** UI displays the number of tables generated as in [Figure 295](#) and the models will appear in the Generated Models List on the **Model Builder** UI as in [Figure 296](#) and will be available for use.

Figure 295: Manage Template—Table Count

Manage Template

BACK ?

Template Type

RDM Template ▼

☐ SourceMap Template

Enter number of tables to be genera... \*

Search Templates: Page 1 of 1

<input type="checkbox"/>	Template Name ↕	Folder Name ↕	Generated Table Count	Modified By ↕
	<input type="text" value="x"/> 🔍	<input type="text" value="x"/> 🔍		<input type="text" value="x"/> 🔍
<input type="checkbox"/>	Reference_Code	LRDM	2	⋮
<input type="checkbox"/>	Reference_Desc	LRDM	2	⋮
<input type="checkbox"/>	Standard_Map	LRDM	2	⋮

1 - 3 of 3

→

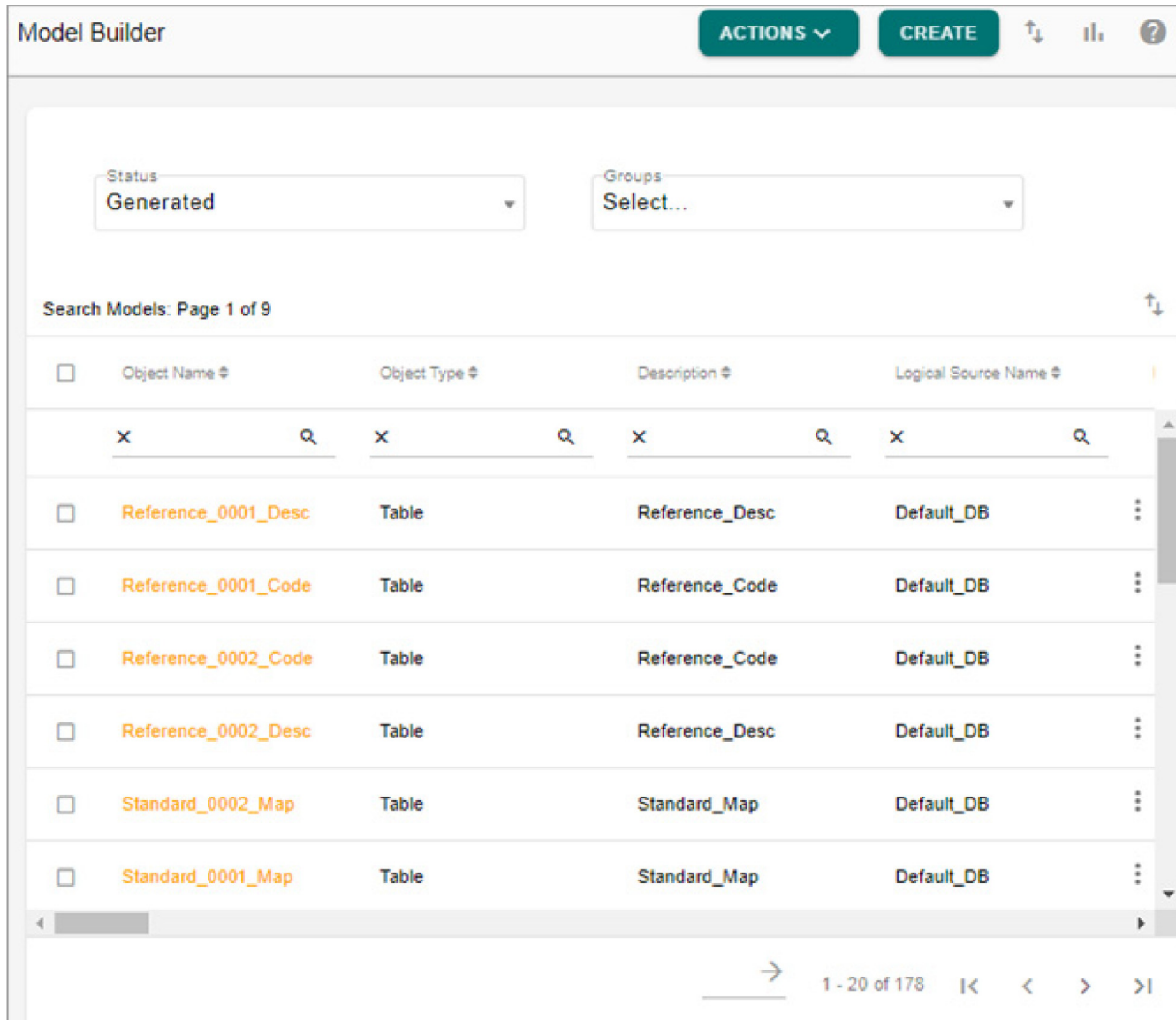
|<

<

>

>|

Figure 296: Model Builder—Generated Models



### Edit Template Tables (RDM, Hierarchy and Cross Reference)

Once the RDM Code Set Tables or Hierarchy Tables or Corss reference template are created, you can optionally add additional columns or change the attributes in the generated table templates.

- 1 On the **Model Builder** UI with **Generated Models** option selected, select any template table to modify and click **Edit**.
- 2 On the **Model Builder** UI, you can modify the values under different tabs and click **Save Model** as in [Figure 297](#).

Figure 297: Model Builder—Object Details

OBJECT DETAILS
COLUMNS
KEYS
LINKS

BACK
SAVE
?

Object Details

Object Name  
Reference\_Code

Description  
Reference\_Code

☐ Create New Group

Groups  
LRDM

Object Type  
Table

Table Type  
Set

☒ Advanced Settings
☐ Statistics

STAGING DETAILS
CONFIGURATION DETAILS
WEB SERVICE ACTIONS

Table Display Name  
Reference\_Code

Physical Table Name  
REFERENCE\_CODE

Master Table Name  
MST\_REFERENCE\_CODE

Error Table Name  
ERR\_REFERENCE\_CODE

Output Table Name  
OUT\_REFERENCE\_CODE

Version Table Name  
VER\_REFERENCE\_CODE

Input Table Name  
IN\_REFERENCE\_CODE

NetChange Table Name  
NC\_REFERENCE\_CODE

Source  
BACKEND

Once the model is modified and saved, the model will move to draft state and the subsequent steps to validate and generate ISG for the table remains the same.

## Manage Metadata Import

MDM provides the ability to import metadata of database objects (table, view, stored procedure, macro and function) from external database to MDM database.

Navigate to **Manage Metadata Import** UI from Model Builder UI by selecting **Metadata Import** from **Actions** dropdown.

On the **Manage Metadata Import** UI, you can perform the following:

- [Import Metadata](#)
- [View Import Log Details](#)
- [Configure](#)

## Import Metadata

To import metadata:

- 1 On the **Manage Metadata Import** UI, click **Import Metadata** from **Actions** dropdown.

The **Import Metadata Objects** UI ([Figure 298](#)) is displayed.

Figure 298: Import Metadata Objects

- 2 On the **Import Metadata Objects** UI ([Figure 298](#)), enter the following details:
  - **Object Type:** select the object type (table, view, stored procedure, macro or function) to import.
  - **Create New Group:** select this checkbox to create a new group. If this checkbox is selected, the **Group** text field is displayed. Enter a name for the Group under which the table needs to be imported in the Group field.
  - **Group:** displays the list of existing groups. Select a group, if you want to import a table under any of the existing group.



- **Import Statistics:** select this checkbox to import the statistics details.
- **Connection:** displays the list of existing connections. Select the required connection from where you want to import the metadata objects. The Connection dropdown will be disabled if **Register External Tables** option is selected.
- **Register External Tables:** this option is selected by default. If this option is selected, the imported tables or views can be used as non-MDM table and this option is available only for tables and views. If this option is not enabled, the imported metadata of objects will be stored in the MDM metadata tables and can be viewed, but the tables cannot participate in the MDM process.

If the Register External Tables option is selected, the following fields will be displayed:

- **Register New DB:** select this option if the database that you want to register is not already registered. The DB Name field is displayed.
- **DB Name:** enter the database name that you want to register and click Validate. The database registered must be within the same database server where the MDM is installed.
- **DB Name (drop-down):** If the database is already registered, select the existing database from the dropdown.
- Select the option **Register External Tables**, select **Register New DB** option and enter the db name in **DB Name** field and click **Validate** as in [Figure 299](#).

Figure 299: Import Metadata Objects

**Import Metadata Objects** BACK VALIDATE ?

☒ **Register External Tables**

☒ **Create new group**

Object Type \*  
Table

Group \*  
NewGroup

Connection \*  
Select...

☒ **Register new db**

DB Name \*  
EXT\_DB

☒ **Import Statistics**

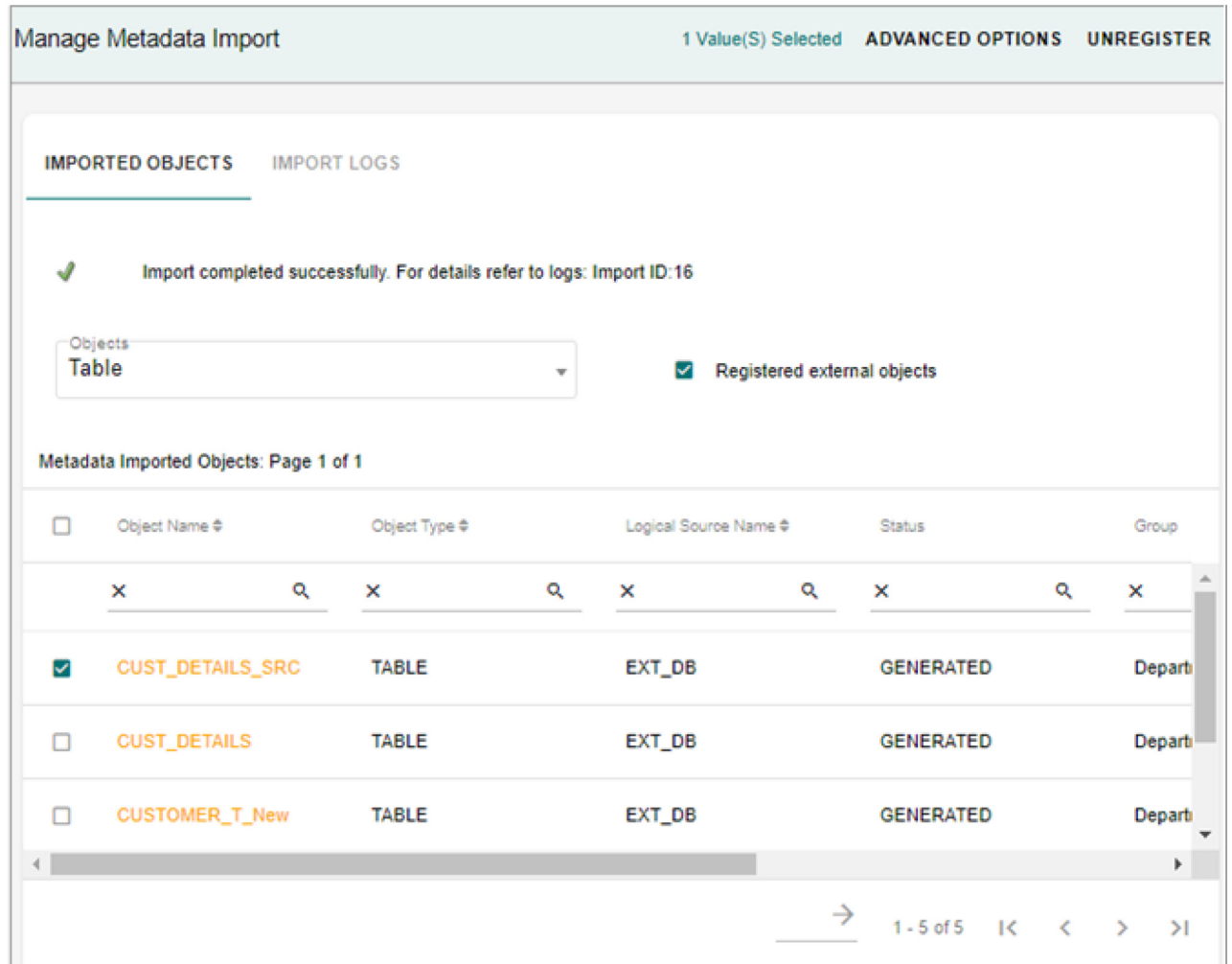
The **Select Objects** pane is displayed. On the **Select Objects** pane, the **Available Objects** list displays all the available objects of the selected database as in [Figure 300](#).

Figure 300: Import Metadata Objects

- On the **Select Objects** pane, select the required objects from **Available Objects** list and move it to **Selected Objects** list.
- 3 Click **Import**.
- Message: "Import completed successfully. For details, refer to logs: Import ID: <Id no>." is displayed as in [Figure 301](#).
- Points on Import Metadata:**

- If same object from same source is imported multiple times (Re-import), the additional attributes will be inserted and the deleted attributes will be made as 'INACTIVE'.
- During model import, the column details of the database table like case specific, title and comment information will be populated in the imported model under columns tab on model builder UI in case specific, column display name and description columns respectively.

Figure 301: Import Metadata Objects



### **Other Functions on Manage Metadata Import**

On Manage Metadata Import UI, you can perform the following tasks on imported objects:

- [Set Advanced Options](#)
- [Unregister](#)
- [View](#)

Points to remember while performing the above operations on Manage Metadata Import UI:

- MDM as well as non MDM database should reside in a single Teradata Box.

- The source database name must not contain any special characters except underscore(\_), hash (#) and Dollar (\$).
- You cannot register two sources with the same name.
- At one time, you can register/unregister maximum of 200 tables for the selected source.
- In Advanced Options setting, select single table from the registered list. If multiple tables selected, the Advanced Options for Primary key columns mapping and macro mapping would be performed on table selected in alphabetical order of precedence (ascending order).
- In the Advanced Options, when the Primary Key Column(s) tab is selected, the Non MDM table's Primary Key will be displayed by default in the Selected Column(s) pane.
- For Non MDM source (table or view), Key definition is a mandatory step if the table or view has no keys defined on it for successful execution of Survivorship/Matching process.
- If Primary Key (PK) of a table or view is changed via source registration advanced option, and matching and survivorship process needs to be executed followed by the change of PK, following order of steps needs to be followed for successful execution.
  - Refresh matching profile.
  - Execute matching profile.
  - Refresh Survivorship profile.
  - Execute Survivorship profile.

The survivorship profile will be executed successfully to insert or update the master records, but the cross reference or relationship capturing will not be correct as the ROM will not be updated when the PK is changed.

- If any changes are made in the non MDM physical database objects that are registered to MDM, run REFRESH\_SYS\_DBC stored procedure manually for the changes to be reflected on the UI.
- When a table is unregistered and registered again, the mapped macro details on the Advanced Options UI will not be deleted and the UI displays the mapping details.
- If you delete the code set value under the RDM Source code set, the deleted code set value will not be available under **Source Name** drop-down on the Source Registration UI. The table/view(s) registered under the deleted source name will not be accessible, but the table/view(s) will continue to work with other RDM functionality.
- In case of AJAX error on source registration page, the page needs to be reloaded. The AJAX error is specific to Internet Explorer and appears when connection to server is closed. Connection might close when there is no net connection or local applications might be blocking server to access the required page.

### ***Set Advanced Options***

On Manage Metadata Import UI, perform the following steps to define key columns:

- Select an imported object and click **Advanced Objects** to define key columns as in [Figure 301](#).



For configurable UI runtime RDM operations, map the database name with the corresponding macro for create, edit, and delete operations on configurable UI for the underlying table.

The **Advanced Options** page ([Figure 302](#)) is displayed. The **Advanced Options** page displays two tabs **Primary Key Column(s)** and **Macro Mapping**. By default, the Primary Key Column(s) tab is selected.

Figure 302: Advanced Options—Primary Key Column(s)

- On the **Advanced Options** page ([Figure 302](#)), with the **Primary Key Column(s)** tab selected, the Available Column(s) list displays all the columns from the table or view, and Selected

Column(s) list displays the key columns if the underlying table or view has primary keys defined.

To add Primary Key column:

- From the Available Column(s) list, select the required column and click **Add**.  
The selected column will be added to the Selected Column(s) list.

To remove Primary Key column:

- From the Selected Column(s) list, select the required column and click **Remove**.  
The selected column will be removed from the Selected Column(s) list.

**Note:** If a primary key column is updated (added/removed), a warning message is displayed if the corresponding table is used in any of the existing matching or survivorship profile.

For the primary key changes to be reflected, the matching and survivorship profile must be refreshed (profile should be edited and saved).

- On the **Advanced Options** page, select **Macro Mapping** tab to map macro for configurable UI Create, Edit and Delete operations and click Save as in [Figure 303](#).

These macros will be used on the configurable runtime UI for the underlying table/view.

Figure 303: Advanced Options—Macro Mapping

Advanced Options

BACK DELETE SAVE ?

\* denotes required field

Table/View Name  
CUST\_DETAILS\_SRC

Database Name  
EXT\_DB

Advanced Options

PRIMARY KEY COLUMN(S) MACRO MAPPING

	Database	Macro
Create Macro	EXT_DB	MacroCreate
Edit Macro	EXT_DB	Macrouupdate
Delete Macro	EXT_DB	MacroDelete

Message: Macro mapping saved successfully is displayed. The user can update macro mapping by following the same above steps.

### **Unregister**

On the **Manage Metadata Import** UI, select the imported object and click Unregister and on the confirmation pop-up, click **Yes**.

### **Impact of Unregistration of Tables**

Unregistration of tables affects the following:

- Cleansing and Standardization Dashboard  
On Cleansing and Standardization Dashboard, all the details related to unregistered tables remains unchanged like the graph etc., but you cannot execute any cleansing and standardization rules.
- Business rules

The unregistered tables will not be displayed in the list, but View Business Rules will still display the rules.

- Matching Profile

On the Matching Profile, you can view the match profiles of the unregistered tables and view the execution log, but cannot execute the match profiles.

- Survivorship Profile

On the Survivorship Profile, you can view the profiles of the unregistered tables and history of execution, but cannot execute the profiles.

- Configurable UI

If a UI is created for the source and if the source is unregistered, you cannot perform any operation on the UI.

- Unregistration will delete all the Key mappings assigned to the table and the macro mappings will not be deleted and will be available if the same table is registered again.

- In Import Metadata, the existing group will be automatically deleted from the System once all the underlying tables are unregistered.

### View

On the **Manage Metadata Import** UI, click on the hyperlink of the imported object to view the metadata of the selected object as in [Figure 304](#).

Figure 304: View Object Details

The screenshot displays the 'View Object Details' window in the Model Builder UI. At the top, there are four tabs: 'OBJECT DETAILS', 'COLUMNS', 'KEYS', and 'LINKS'. The 'OBJECT DETAILS' tab is selected and underlined. In the top right corner of the window, there is a 'BACK' button and a help icon (a circle with a question mark). Below the tabs, the title 'Object Details' is displayed. The main content area is divided into two columns. The left column contains the 'Object Name' field with the value 'CUST\_DETAILS\_SRC' and a text input box below it also containing 'CUST\_DETAILS\_SRC'. Below this is a checkbox labeled 'Create New Group' which is unchecked. At the bottom of the left column is a 'Groups' dropdown menu with 'Department' selected. The right column contains the 'Database Name:' label, followed by an 'Object Type' dropdown menu set to 'Table', and a 'Table Type' dropdown menu set to 'Set'. At the bottom of the right column are two unchecked checkboxes: 'Statistics' and 'Source Specific Definition'.



## Configure

MDM Model Builder UI provides an option to configure an external table (Non-MDM table) available in the external database other than the MDM database so that the configured table can be used in the same manner as MDM Master table.

Configuring an external table can be done in the following two ways:

- **Configure for non MDM Operations:** tables configured using this option can be used in all MDM processes (Eg: Validation, Approval etc.). In this option, a reference to the actual tables in external database will be made in MDM server and database so that the tables can seamlessly be used in all MDM processes. The advantage of this process is that the underling data of the table can be managed using MDM processes similar to master tables of MDM. The database of the external table needs to be within same database server as that of MDM user.

While Configuring for non MDM operations, you can select the option to either alter the external table and add System (SYS) columns required for MDM processes as part of the external table itself or select the option to not alter the external tables and add the SYS columns in a new table. For more details, see [Configure for MDM Operations](#).

- **Convert as MDM:** Using this option the external database table can be converted as Master table of MDM. In this option, the external table will be re-created as Master table using the model builder UI. Once the table is created as Master table, the data from original external table needs to be copied to the Master table. There will not be any automatic sync-up of data available between the original table and Master table. For more details, see [Convert As MDM](#).

### **Configure for MDM Operations**

#### **Prerequisite**

The following access rights have to be granted to configure any external table to be used for MDM operations:

- To configure any external table with an option to alter the external table, the MDM user must have the following ALTER/INSERT/UPDATE access rights.

An example of access SQL can be:

- All access to the external table to user database.

```
GRANT ALL ON <EXT_DB>.<EXT_TABLE> TO <USER_DB>
```

- All access to the external table to user Master DB with grant option

```
GRANT ALL ON <EXT_DB>.<EXT_TABLE> TO <USER_DB>_MST WITH GRANT  
OPTION
```

- To configure any external table without an option to alter the external table, the MDM user must have the SELECT/INSERT/UPDATE access and ALTER access is not required.



If the above access rights are not granted, an error will be displayed and cannot be proceeded further. Also, the tables will not be allowed to configure if it is used as non-MDM table in any features of MDM.

To configure non MDM tables for MDM operations:

- 1 On the **Manage Metadata Import** UI, select **Configure External Tables** from **Actions** dropdown.

The **Configure External Tables** UI is displayed.

Figure 305: Configure External Tables

Configure External Tables

BACK CONFIGURE ?

^

\* denotes required field

Select Options \*

Configure For MDM operations

Select...

Configure For MDM operations

Convert as MDM

EXT\_DB.CUSTOMER\_T

EXT\_DB.CUSTOMER\_T\_New

EXT\_DB.CUST\_DETAILS

EXT\_DB.CUST\_DETAILS\_SRC

EXT\_DB.CUS\_TMP\_01

>>

>

<

<<

☒ Alter external table

Selected non MDM Objects

Tables Configured for MDM operations: No Tables Configured for MDM operations

Non MDM Table	Table Status	Alter non MDM
x	Q	x

→ 1 - 0 of 0 < > >|

- 2 On the **Configure External Tables** UI, select the option **Configure for MDM Operations** from the **Select Options** drop down.
- 3 Select the Non MDM table from the **Available Non MDM Objects** list and move it to **Selected Non MDM Objects** list and click **Configure**. Tables imported as non-MDM objects will appear in **Available Non MDM Objects** list box.

- 4 **Alter Non MDM Table:** select this option if you want to alter the external table and add System (SYS) columns as part of the external table itself. By default, this option is selected. If this option is selected, after configuring, run the SQL

```
update externalDb.externalTablename set sys_ent_state='ACTIVE'
```

If this option is not selected, the System (SYS) columns will not be added as part of external table and MDM adds the SYS columns in a new table. If this option is not selected, after configuring, run the sql in the below format to insert data into the SYS columns table.

```
INSERT INTO masterDb.sysColumnsTableName(pkcolumns comma
separated,SYS_ENT_STATE) select pkcolumns comma separated,
'ACTIVE' from externalDb.externalTablename
```

For example:

```
INSERT INTO
qauser01_mst.MST_CUSTOMER_TEST_NOALTERACCESS_SYS_COLUMNS(CUS
T_ID,SYS_ENT_STATE) select cust_id, 'ACTIVE' from
ext_db.CUSTOMER_TEST_NOALTERACCESS
```

- 5 Click **Configure**.

Message: “A copy of selected object has been created as MDM object, you will need to run ISG from Model Builder to use it.” is displayed and the configured table will be displayed in the Tables Configured for MDM operations section as in [Figure 306](#).

Figure 306: Configure Non MDM Tables

Configure External Tables

BACK CONFIGURE ?

^

\* denotes required field

✓

A copy of selected object has been created as MDM object, you will need to delpoy model to use it.

Select Options \*

Configure For MDM operations

☒ Alter external table

Available non MDM Objects

EXT\_DB.CUST\_DETAILS  
EXT\_DB.CUST\_DETAILS\_SRC  
EXT\_DB.CUS\_TMP\_01

>>  
>  
<

Selected non MDM Objects

^

Tables Configured for MDM operations: Page 1 of 1

Non MDM Table	Table Status	Alter non MDM
<div><div>×</div><div>Q</div></div>	<div><div>×</div><div>Q</div></div>	
EXT_DB.CUSTOMER_T_New	DRAFT	false
EXT_DB.CUSTOMER_T	DRAFT	false

→

1 - 2 of 2

◀

<

>

▶

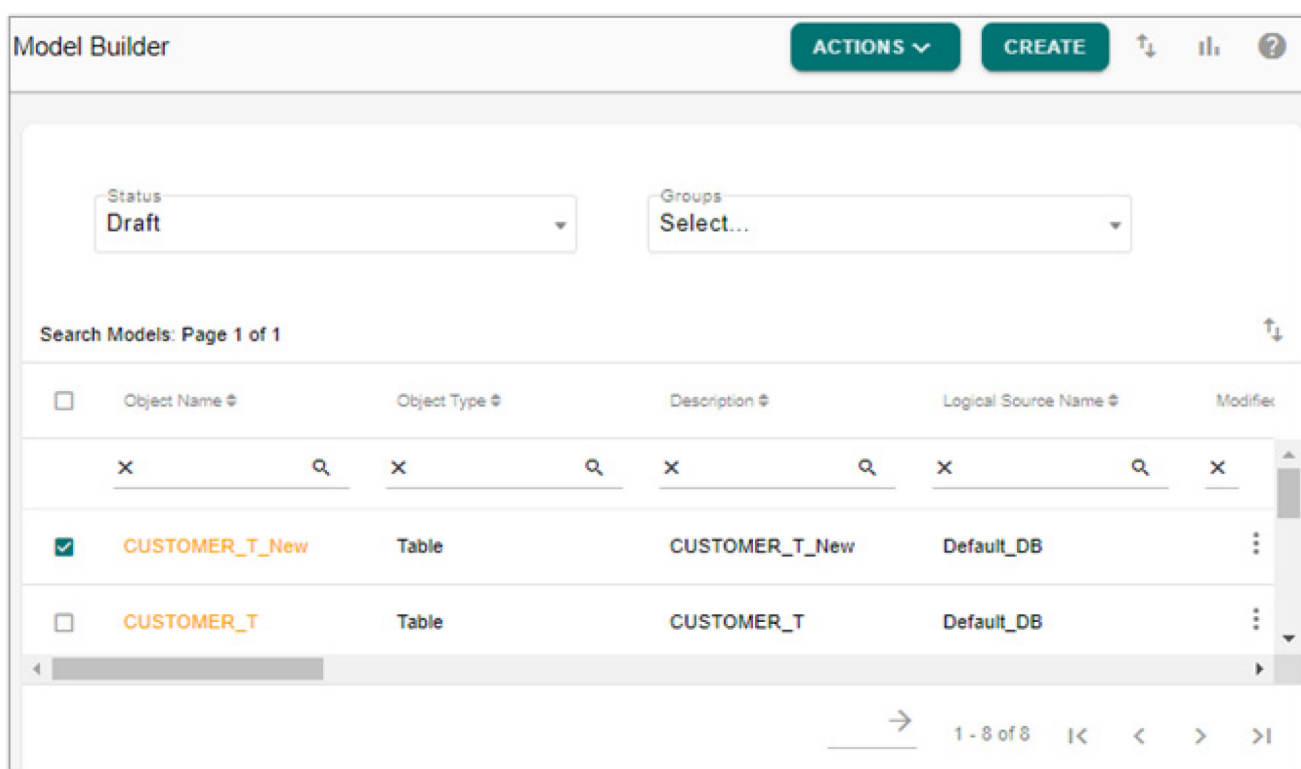
The metadata of the selected tables get copied in Model Builder and the tables will be available in draft state in the Model Builder UI as in [Figure 307](#). You can modify the table details and deploy the model for the changes to be reflected. All data management activities happens directly on the table through MDM persistence framework.



On the draft model, you can update the column display name and facets details and cannot edit the object for any structural changes if the option to alter the external table is not selected while configuring the external table.

In case of configuring an external table with an option to alter the external table, you can update the table with all the required changes.

Figure 307: Model Builder



Some of the points to be noted while configuring the external table in Model Builder UI.

- The configured non MDM table may have columns with any data type as supported in the external database. Some of those data types may not be supported in MDM and those unsupported datatypes will be mapped as String data type during the configuration process. You can then modify the data types, facets and other details from the model builder UI as per the requirement. Once all the changes are done, you can run ISG to generate the table as any other MDM based tables from the Model Builder UI. Once the deployment is completed successfully, the generated table will appear under Master list and will not be displayed under non MDM list.
- The configured external table on the model builder UI supports the following features that are supported by MDM tables.
  - LookUp addition to any columns
  - Editable/non-Editable
  - Source selection as backend or Master

- Surrogate Key
- Audit
- Hard Delete only if user has delete access to the external database
- WebServices
- While defining the business rule on external table (without alter option), the system columns cannot be updated by enrichment and transformation rules. In case of custom business rule, it is the user's responsibility to write the business rule accordingly to include the sys columns otherwise execution will fail.
- In case of configuring an external table without an option to alter the external table, the Surrogate Key and Hard Delete will not be supported.
- Configured external table will not support the features: Extended Attributes, Conversion of non temporal to temporal and editing of PK values.
- Creation of foreign key (FK) link between Non MDM tables with or without system columns is not supported.

### ***Convert As MDM***

To convert a non MDM table to MDM table, perform the following steps:

- 1 On the **Configure External Tables** UI, select the option **Convert as MDM** from the **Select Options** dropdown.
- 2 Select the Non MDM table from the **Available Non MDM Objects** list and move it to **Selected Non MDM Objects** list and click **Configure** as in [Figure 308](#).

Figure 308: Configure Non MDM Tables-Convert as MDM

Message: “A copy of selected object has been created as MDM object, you will need to run ISG from Model Builder to use it”

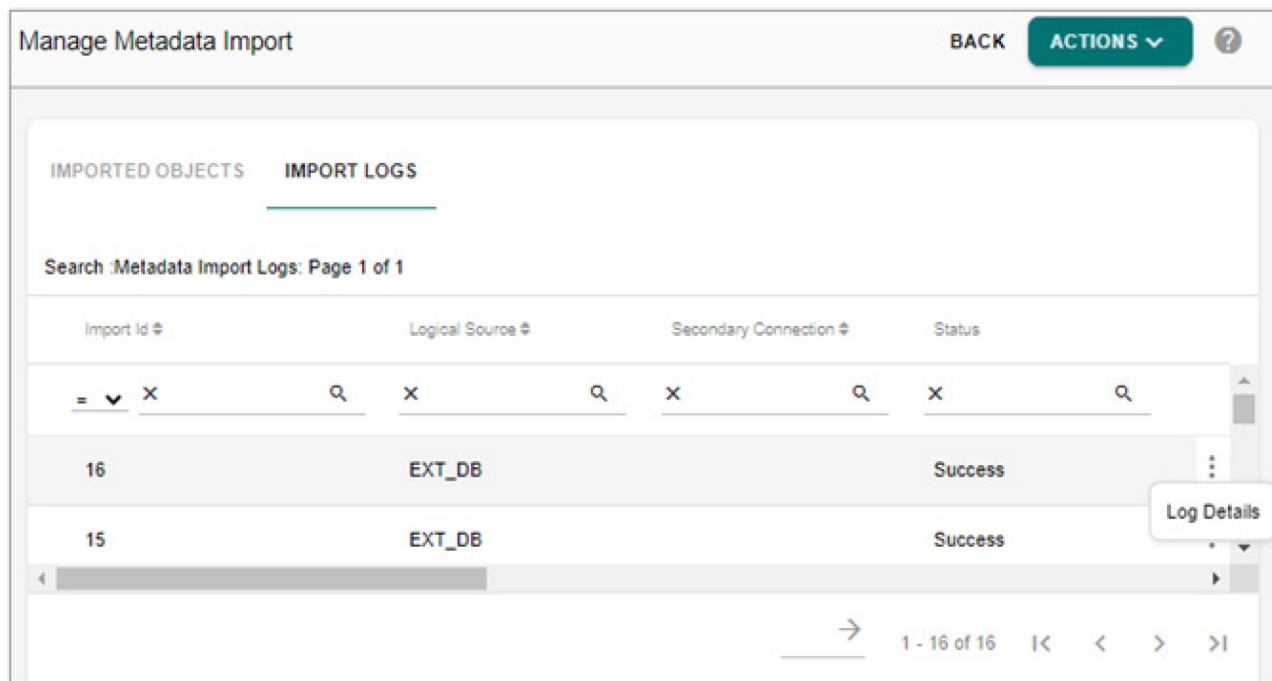
The metadata of the selected object gets copied as MDM based table in draft state. You can then modify the facets and other details from the model builder UI as per the requirement. Once the table is in Draft state, the subsequent steps to generate the table remains same as any other MDM based tables.

### View Import Log Details

To view import log details:

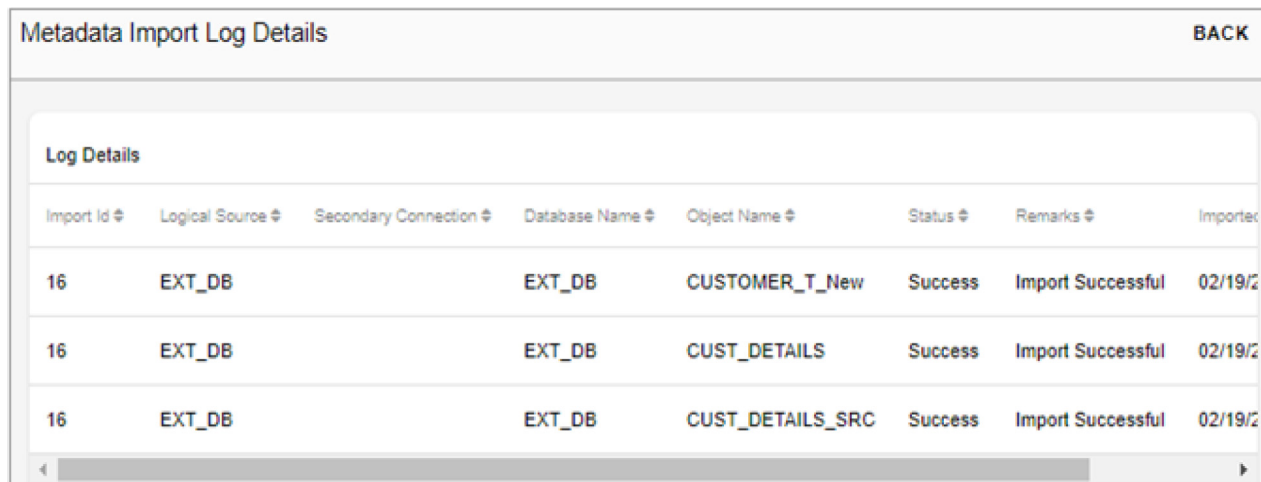
- 1 On the **Manage Metadata Import** UI, click **Import Logs** tab.  
The Import Logs UI displays the list of logs of imported metadata as in [Figure 309](#).
- 2 Click on the Menu icon corresponding to the required metadata import and click **Log Details**.

Figure 309: Manage Metadata Import



The **Metadata Import Log Details** UI (Figure 310) is displayed.

Figure 310: Metadata Import Log Details



The **Metadata Import Log Details** UI displays the log details like Import ID, logical source name, source name, database name, object name, status, remarks and imported date.

### Comprehensive Support for External Tables

An MDM table object will be created in model Builder with the extra SYS\_COLUMNS needed for MDM

User can verify the model structure and can run ISG



While running ISG all the staging tables like IN,NC,VER,GT,ERR will be created on the corresponding child DB's except the master table.

Document Object will be created similar to a master table

The view under the parent DB will point to the external Table and will be used by the data retrieving and persisting APIs of MDM

MDM will be interacting with the external table through the document object and all the Master table operations should be supported

MDM will perform Insert and Update operations on the External object. If the External object is a complex view INSERT and UPDATES will fail, hence only External tables and simple views will be supported.

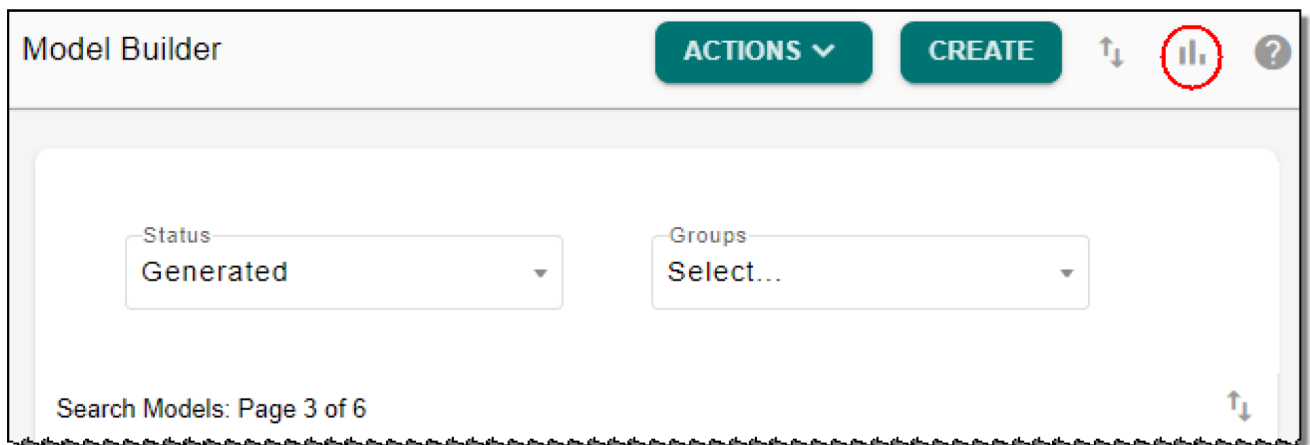
## Model Builder Dashboard

Model Builder Dashboard provides a quick glance of end-to-end progress for each model created. For each model, the dashboard displays the Model Status, Groups of Models, recent deployment Logs and Model Configuration Details. Data administrator can track overall status of model builder tables using the Dashboard. Also, with the help of drill downs provided on Dashboard, users can track interested component as required.

The Model Builder Configuration on the dashboard helps to monitor the list of activities for all the modules configured for a specific table. Using the drill down option, you can navigate directly to the set-up pages of the selected module to resolve the missing components.

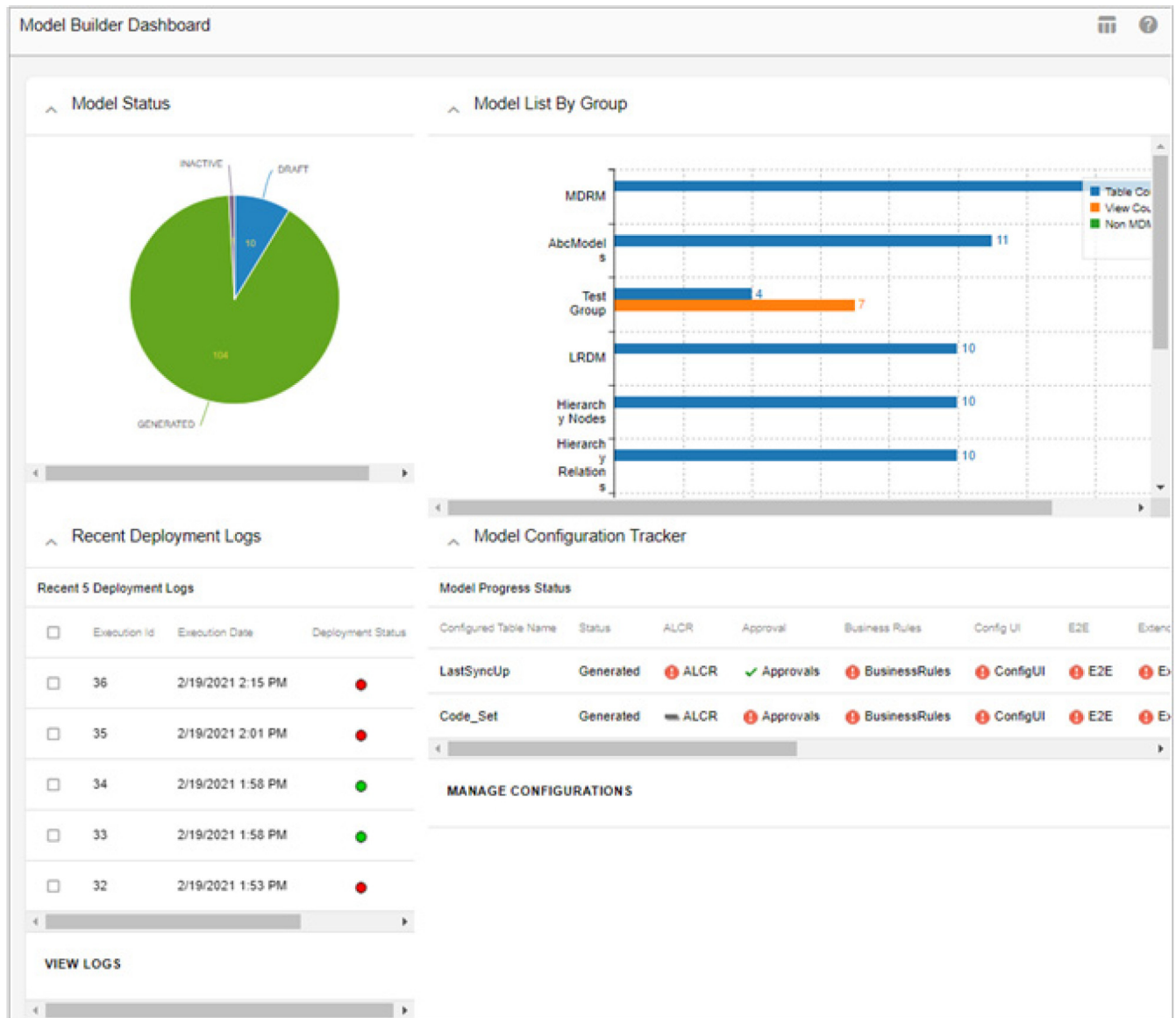
You can navigate to **Model Builder Dashboard** UI by clicking the graphical view icon on the **Model Builder** UI as in [Figure 311](#).

Figure 311: Model Builder Dashboard



The **Model Builder Dashboard** ([Figure 312](#)) is displayed.

Figure 312: Model Builder Dashboard



The **Model Builder Dashboard** provides the following details:

- **Model Status**—a pie chart displays the total number of models under each model stages (Draft, Validated, Generated, Pending Approval and Inactive). Click on each model stage to drill down to model builder UI to edit, export, delete models and to validate and deploy models.
- **Model List By Group**—a bar chart displays for each group of model, its table and view count group wise like the count of Non MDM Models, MDM Views and MDM Tables.
- **Recent Deployment Logs**—displays the recent five model builder deployment status. To view detail log of each execution, select the execution Id and click **View Logs** to navigate to **Model Deployment Logs** UI.

If you select the model and click on **View Logs**, it displays only the logs of selected model. To view logs of all the deployments, click on **View Logs** button without selecting any model. For detailed description of deployment logs, see [Section : “Deployment Logs.”](#)

- **Model Configuration Tracker**—the Model Configuration Tracker highlights the status of all the end to end activities of the different modules configured for each of the model created on Model Builder UI. A workflow or code set (MDM Modules) is used to configure what modules to be displayed on the configuration dashboard for each model.

The Model Configuration Tracker pane displays the latest five configurations created along with the execution status in tabular form. Different execution status includes the following:



shows that the module configuration is completed.



shows that the module configuration is incomplete.



shows that the module is not selected for configuration.

An hyper link is provided on each of the module execution status that helps to navigate to the set-up pages of the specific module. For example, clicking on “Config UI” status link would redirect to Configurable UI to configure the missing left navigation components.

For detailed description on Model Configurations, see [Section : “Manage Model Configuration.”](#)

## Manage Model Configuration

Every model created on the Model builder UI can be associated with different MDM modules like (ISG, Config UI creation, User Access, E2E Set-up etc.) using the **Manage Dashboard Configurations** UI. The Manage Dashboard Configurations UI provides an easy way to verify what all components were created or pending or missing for a table.

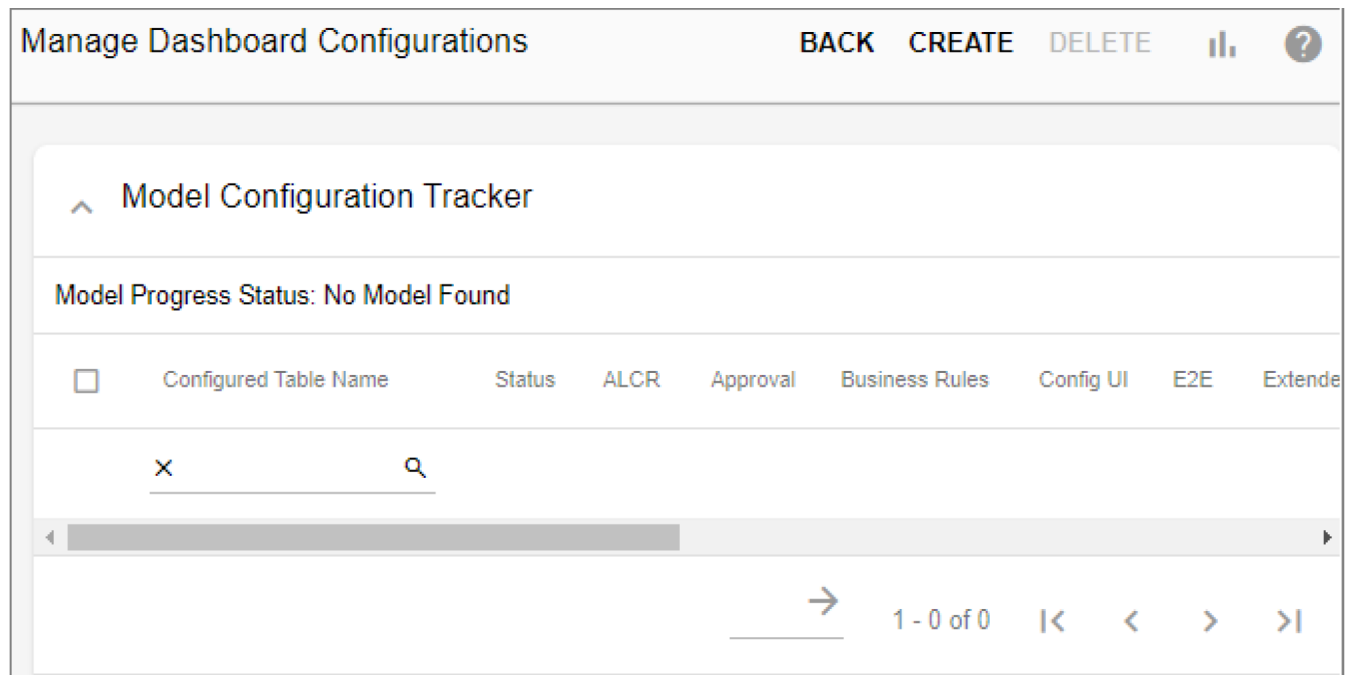
Navigate to **Manage Dashboard Configurations** UI from **Model Builder Dashboard** by clicking on **Manage Configurations** button under **Model Configuration Tracker** pane.

### Create Model Configuration

Perform the following steps to create model configuration:

- 1 On the **Manage Dashboard Configurations** page ([Figure 313](#)), click **Create**.

Figure 313: Manage Dashboard Configurations



The **Create Configuration** pane is displayed as in [Figure 314](#).

Figure 314: Manage Dashboard Configurations—Create Configuration

The screenshot displays the 'Manage Dashboard Configurations' interface. At the top, there is a header bar with the title 'Manage Dashboard Configurations' and several action buttons: 'BACK', 'CREATE' (highlighted), 'DELETE', and 'SAVE' (in a green box). To the right of these buttons are icons for a bar chart and a help/question mark.

Below the header, the main content area is divided into two sections. The first section, titled 'Model Configuration Tracker', shows a status message: 'Model Progress Status: No Model Found'. Below this is a table with columns: 'Configured Table Name', 'Status', 'ALCR', 'Approval', 'Business Rules', 'Config UI', 'E2E', and 'Extend'. The table is currently empty, with a search bar and a scrollbar visible.

The second section, titled 'Create Configuration', contains a list of modules with their status. Each module has a checkbox and a label 'Enabled'.

Module	Status
Hierarchy	Enabled
LRDM	Enabled
Manage Authorization	Enabled
Matching	Enabled
Survivorship	Enabled

On the **Create Configuration** pane, the **Table Name** dropdown lists all the tables on which you can enable or disable different module configurations. By default, all the modules will be enabled.

- 2 On the **Create Configuration** pane, optionally deselect any module operation to be skipped in configuration and click **Save**.

Message: “Configuration created successfully” is displayed and the status of each module is displayed as in [Figure 315](#).

Figure 315: Manage Dashboard Configurations

Manage Dashboard Configurations																																		
						1 Value(S) Selected <a href="#">DELETE</a>																												
<div> <div>Model Configuration Tracker</div> <div>Model Progress Status: Page 1 of 1</div> <table> <tr> <th><input type="checkbox"/></th><th>Configured Table Name</th><th>Status</th><th>ALCR</th><th>Approval</th><th>Business Rules</th><th>Config UI</th></tr> <tr> <td colspan="7"> <div> <div>X</div> <div>Q</div> </div> </td></tr> <tr> <td><input checked="" type="checkbox"/></td><td>LastSyncUp</td><td>Generated</td><td><span>! ALCR</span></td><td><span>✓ Approvals</span></td><td><span>! BusinessRules</span></td><td><span>! ConfigUI</span></td></tr> <tr> <td><input type="checkbox"/></td><td>Code_Set</td><td>Generated</td><td><span>■ ALCR</span></td><td><span>! Approvals</span></td><td><span>! BusinessRules</span></td><td><span>! ConfigUI</span></td></tr> </table> </div>							<input type="checkbox"/>	Configured Table Name	Status	ALCR	Approval	Business Rules	Config UI	<div> <div>X</div> <div>Q</div> </div>							<input checked="" type="checkbox"/>	LastSyncUp	Generated	<span>! ALCR</span>	<span>✓ Approvals</span>	<span>! BusinessRules</span>	<span>! ConfigUI</span>	<input type="checkbox"/>	Code_Set	Generated	<span>■ ALCR</span>	<span>! Approvals</span>	<span>! BusinessRules</span>	<span>! ConfigUI</span>
<input type="checkbox"/>	Configured Table Name	Status	ALCR	Approval	Business Rules	Config UI																												
<div> <div>X</div> <div>Q</div> </div>																																		
<input checked="" type="checkbox"/>	LastSyncUp	Generated	<span>! ALCR</span>	<span>✓ Approvals</span>	<span>! BusinessRules</span>	<span>! ConfigUI</span>																												
<input type="checkbox"/>	Code_Set	Generated	<span>■ ALCR</span>	<span>! Approvals</span>	<span>! BusinessRules</span>	<span>! ConfigUI</span>																												
						<div>Edit</div>																												

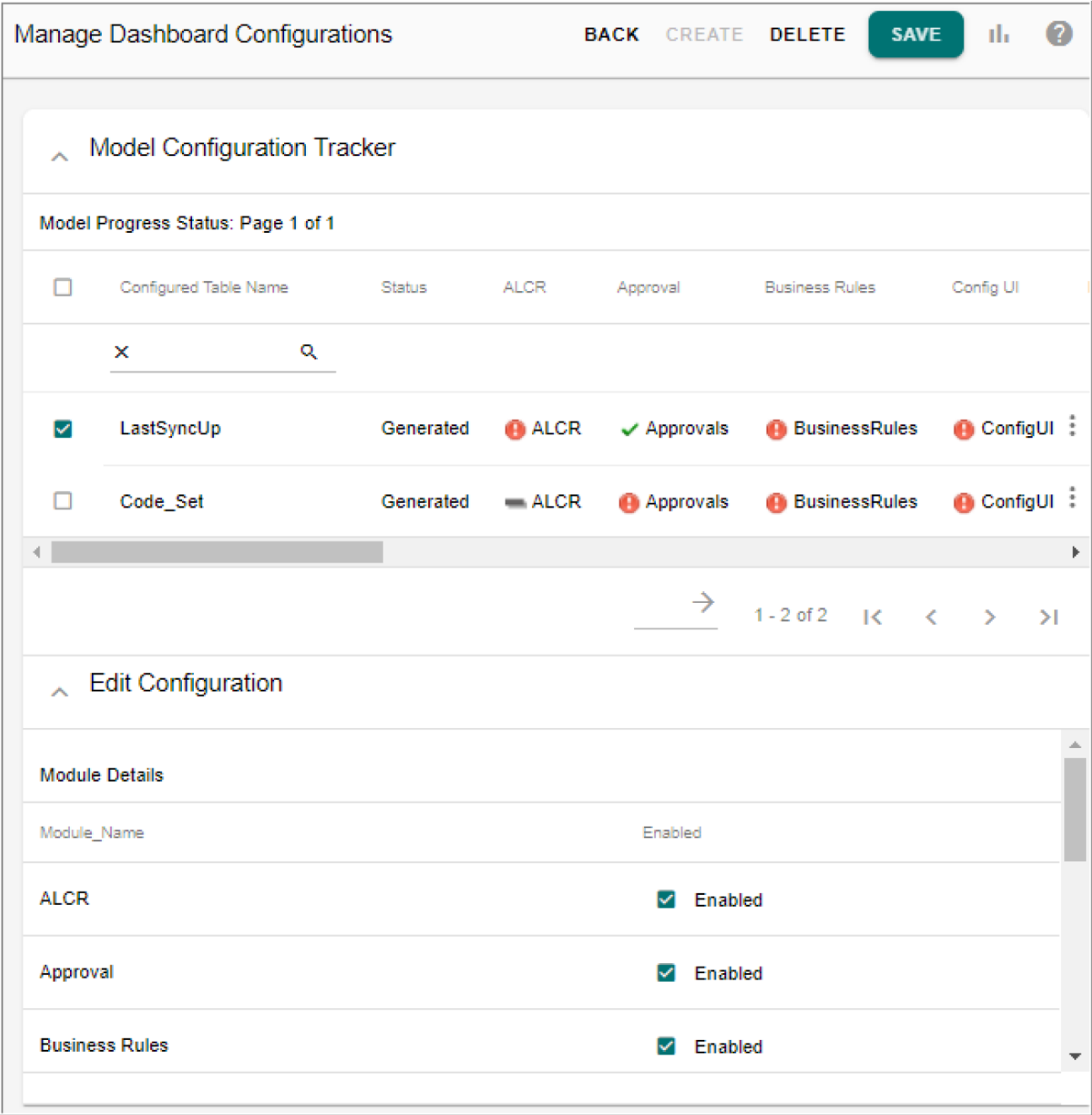
- 3 Click on each of the status icon to navigate to the corresponding set-up UI where you can configure all the missing components.

### ***Edit Model Configuration***

Perform the following steps to edit model configuration:

- 1 On the **Manage Dashboard Configurations** page ([Figure 315](#)), click on the Menu icon corresponding to the required configuration and click **Edit**.

Figure 316: Manage Dashboard Configurations—Edit Configuration



- 2 On the **Edit Configuration** pane, optionally select / deselect any module operation and click **Save**.

Message: “Configuration updated successfully” is displayed.

**Delete Model Configuration**

Perform the following steps to delete model configuration:

- 1 On the **Manage Dashboard Configurations** page, click on the Menu icon corresponding to the required configuration and click **Delete**.
  - 2 On the confirmation pop-up, click **Yes**.
- Message: “Configuration deleted successfully” is displayed.



## APPENDIX C Configurable UI

---

### What's In This Appendix

This appendix provides detailed information on Configurable UI.

Topics include:

- [Generating Configurable UI](#)

## Generating Configurable UI

RDM's Configurable UI feature allows the business user to create a new user interface screen using a simple wizard method or generate a new UI in a single click on Generate Configurable UI. Configurable Web UIs can either be Simple or Advanced Template.

### Generate Simple Configurable UI

The below section describes the creation of simple configurable UI using a single click. The simple configurable UI creation generates a basic template of Table Maintenance UI with basic functions of view, delete, add, copy, edit, mass update, excel upload, filters, sort and show history.

Perform the following steps to generate a simple or automated Configurable UI:

- 1 Navigate to Generate Configurable UI page from **Model Setup -> Configurable UI**. On the **Configurable UI** page, click **Generate Configurable**.

Figure 317: Generate Configurable UI

Generate Configurable UI

BACK GENERATE ?

Service Name  
Select...

☐ In-line Edit

Tables in selected service: No Tables Found

<input type="checkbox"/> Table Display Name	<input type="checkbox"/> Table Display Name
---	---

- 2 On the **Generate Configurable UI** (Figure 317), select the service name of the Table from the **Service Name** dropdown. In this example, select service name as “Master”.  
The Tables panel displays all the tables of the selected Master service. Tables for which configurable UIs are already generated will be grayed out.
- 3 In the **Tables** pane, select “Account” Table and click **Generate** as in Figure 318.

Figure 318: Generate Configurable UI

Generate Configurable UI

BACK GENERATE ?

Service Name  
Master

☒ In-line Edit

Tables in selected service

☐ Table Display Name

☒ Account

☒ AttributeSetDetail

☒ Code\_Set\_Stats

☒ CrossRefRelation

☒ Custom\_Code\_Set\_Clm

☒ Group

Message “A notification will be sent to the configured email Id after the completion of configurable UI generation” is displayed.

Refresh your RDM Landing Page and the Configurable UI will be displayed under **Tables** -> **Account**. [Figure 319](#) displays a sample simple configurable UI.

Figure 319: Sample of Simple Configurable UI—Account Table

Account				
<div> <div> <div> <div> <div></div> <div>Account</div> </div> <div> <div>UPLOAD EXCEL</div> <div> <div></div> <div></div> </div> <div> <div></div> <div></div> </div> <div> <div></div> <div></div> </div> <div> <div>CREATE</div> <div></div> <div></div> <div></div> </div> </div> </div> </div> </div>				
<input type="checkbox"/>	AccountId	AccountName	Amount	Department
<input type="checkbox"/>	6	Back Office Account	90000	Back Office
<input type="checkbox"/>	7	Branch Account	90000	Branch
<input type="checkbox"/>	9	Departmental Account	90000	Departmental
<div> <div></div> <div>1 - 3 of 3</div> <div> <div></div> <div></div> <div></div> <div></div> </div> </div>				

## Advanced Configurable UI

The Advanced Configurable UI generates an advanced template of Table Maintenance UI with basic functions of simple configurable UIs in addition to advance filters and search features, defining custom buttons and edit functionalities.

Perform the below steps to generate an advanced configurable UI:

1. Navigate to **Configurable UI** page from **Model Setup -> Configurable UI**.

On the **Configurable UI** page (Figure 320), by default, the **User Defined Configurable UI** option is selected and the search pane displays user defined configurable UIs. The Auto generated Configurable UI option will show all UIs defined through the Generate Configurable UI.

Figure 320: Configurable UI

- 2 On the **Configurable UI** page, with the **User Defined Configurable UI** option selected, click **Create** as in [Figure 320](#).
- 3 On the **Define New UI** page ([Figure 321](#)), enter the required details (UI Name—will appear as UI Header, Page Name—is used in Menu Builder, Table Names and Template Type—Advanced) and click **Next**.

Figure 321: Configurable UIs—Define New UI

The screenshot shows the 'Define New UI' form. At the top, there is a title 'Define New UI' and navigation buttons 'BACK' and 'NEXT'. A help icon (?) is also present. The form contains several input fields:

- UI Name**: A text field containing 'Account UI'.
- Page Name**: A text field containing 'AccountConfigAdv'.
- Business Entity**: A dropdown menu with 'Select...' selected.
- Repository**: A dropdown menu with 'Master' selected.
- Primary Table**: A dropdown menu with 'Account' selected.
- Template Type**: A dropdown menu with 'Advanced' selected.
- Description**: A text area containing 'Account Configurable UI Advanced'.
- Join Mechanism**: A section with two radio buttons: 'Document Links' (selected) and 'Explicit Join'.

- 4 On the **Define Search Columns** UI, select the columns to appear in the Search feature of the Advanced UI as in [Figure 322](#).  
For this example, select AccountName, Department, Amount and SOURCE.

Figure 322: Configurable UIs—Define Search Columns

### Define Search Columns

CANCEL
BACK
CONFIGURE ADVANCED FILTER
CONFIGURE SEARCH COLUMNS
?

Page Name  
AccountConfigAdv

UI Name  
Account UI

#### Select Columns

Available Columns

- Account.Target Id
- Account.Authorization
- Account.Source
- Account.Created By
- Account.Creation Date
- Account.Entity State
- Account.System Type
- Account.Error Code
- Account.Error Severity
- Account.Approval Id
- Account.TemporalValidTime

>>

>

<

<<

Selected Columns

- Account.AccountId
- Account.AccountName
- Account.Amount
- Account.Department
- Account.StoreId
- Account.Last Modified By
- Account.Last Modified Date

UP

DOWN

☒ Perform Default Search

- Click **Configure Advanced Filter** to specify advanced filters for the search results if needed. Here is an example.

Figure 323: Configurable UI—Configure Advanced Filter

The screenshot shows the 'Configure Advanced Filter' dialog. At the top, there's a title bar with 'Configure Advanced Filter', 'CANCEL', and 'APPLY AND RETURN' buttons. Below the title bar, there's a 'Target Table' dropdown set to 'Account' and a checked 'Global Filter' checkbox. The main area is divided into three panels: 'All Properties' (a list of properties with 'Amount' selected), 'Filter Description' (fields for 'Property Name' (Amount), 'Data Type' (int), 'Operation' (=), and 'Filter Value' (1000)), and 'Expression Editor' (displaying 'Account.Amount = 1000'). At the bottom, there are 'AND', 'OR', and parentheses buttons, and an 'ADD' button.

- 6 On the **Configure Advanced Filter** UI (Figure 323), perform the following steps:
  - From the **All Properties** list, select the required property (example: Amount).
  - In the **Filter Description** pane, select the operation (say =) and enter the filter value in the Filter Value field (say 10000) and click **Add**.  
The added filter expression is displayed in the **Expression Editor** field.
  - Click **Apply and Return** to save the filter or **Cancel** to go back to the **Define Search Columns** page.
- 7 Click **Configure Search Columns**.



Figure 324: Configurable UIs—Configure Search Columns

Search Column	Display Name	Field Type	Lookup Code
Account.AccountId	AccountId	Entry Field	Select...
Account.AccountName	AccountName	Entry Field	Select...
Account.Amount	Amount	Entry Field	Select...
Account.Department	Department	Drop Down	Department
Account.StoreId	StoreId	Entry Field	Select...
Account.LAST_MODIFIED_BY	LAST_MODIFIED_BY	Entry Field	Select...

- 8 On the **Configure Search Columns**, you can configure the field type for the search columns you have defined and click **Next** as in [Figure 324](#).

The field type can be either Entry Field or dropdown. Necessary look up code values must be defined in for drop down values.

- 9 On the **Define Result Columns** UI ([Figure 325](#)), you can specify the result columns you want the Configurable UI to display on searches.

The Create, Copy, Edit, Mass Update, Delete and Upload Excel checkboxes will be selected by default to be displayed as action buttons on the UI being configured.

Figure 325: Configurable UI—Define Result Columns

**Define Result Columns** CANCEL BACK CONFIGURE RESULT COLUMNS ?

Page Name  
AccountConfigAdv

UI Name  
Account UI

**Select Columns**

Available Columns Selected Columns

Account.System Type  
Account.Error Code  
Account.Error Severity  
Account.Approval Id  
Account.TemporalValidTime

>>  
>  
<  
<<

Account.AccountId  
Account.AccountName  
Account.Amount  
Account.Department  
Account.StoreId  
Account.Locale  
Account.Target Id

UP  
DOWN

☒ Create ☒ Copy ☒ Edit ☒ Mass Update ☒ Delete ☒ Upload Excel

**Configure Help File:**

Path for help file(ex: bcm/help/confi...

**Configure Pagination:**

Number of rows to be displayed per ...

On the **Define Result Columns** UI, you can also specify the following:

- **Custom Buttons:** you can specify the action buttons to be available on the configurable UI. Enable this by clicking the Custom Button checkbox and enter the custom button details as in [Figure 326](#).

Figure 326: Configurable UIs—Define Result Columns

10 Click **Configure Result Columns** once done.

On the **Configure Result Columns** UI, you can configure the following:

- Change the display name, hover text (by default, the logical names would be displayed) and specify the default values of the result columns.
- Specify the background color for each of the result column.
- **Editable:** make any column non editable by clicking the **Configure Create/Edit Columns** button to enable deselection of the corresponding columns checkboxes. By default, all the selected result columns will be editable on the configured Create or Edit page except SYS and Primary Key columns.

Additionally for String Column, click on Configure Create/Edit Columns button then click on respective Field Type column to choose among checkbox, radio button, text area, and entry field.

- **Sortable:** by default, all the selected result columns will be sortable on the Search Result page of the configured UI. Make any column non sortable by deselecting the corresponding checkbox.
- **Display NoWrap:** by default, the hover text of all the selected result columns will be wrapped on the Search Result page of the configured UI. De-selecting this option on any column results in No wrapping of the hover text.

- **i18nized**: by default, the i18nized checkbox is not selected. If the i18nized option is selected for a column, the corresponding column data will be displayed in i18nized format on the Preview page.

Figure 327: Configurable Uis—Configure Result Columns

**Configure Result Columns** CANCEL BACK CONFIGURE CREATE/EDIT COLUMNS SAVE ?

Selected Result Columns

Editable	Sortable	Display NoWrap	i18nized	Add Code Value	Result Column	Field Type
<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Account.AccountId	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Account.AccountName	Entry Field
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Account.Amount	
<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	Account.Department	

☐ Default Sort

Select Key Columns

Available Columns

Selected Key Columns

Account.AccountName  
Account.Amount  
Account.Department  
Account.StoreId  
Account.Locale  
Account.Target Id  
Account.Authorization

Account.AccountId

- Select the Key columns from the **Available Columns** list and move it to **Selected Key Columns** list. By default, the Selected Key Columns list displays the Primary Key columns of all the participating tables in the UI. The selection of **Key Columns** is a mandatory step as runtime operations use Selected Key Columns to operate on the selected rows.



The key columns are used to fetch data for configurable UI from the underlying tables/ views.

- **Default Sort:** Select the Default Sort option to sort the records in each column either in ascending or descending order on the configured UI at runtime. When the Default Sort option is selected, the **Sort Customization** panel is displayed. On the Sort Customization panel, select the columns on which you want to apply the sort and the corresponding sorting order.

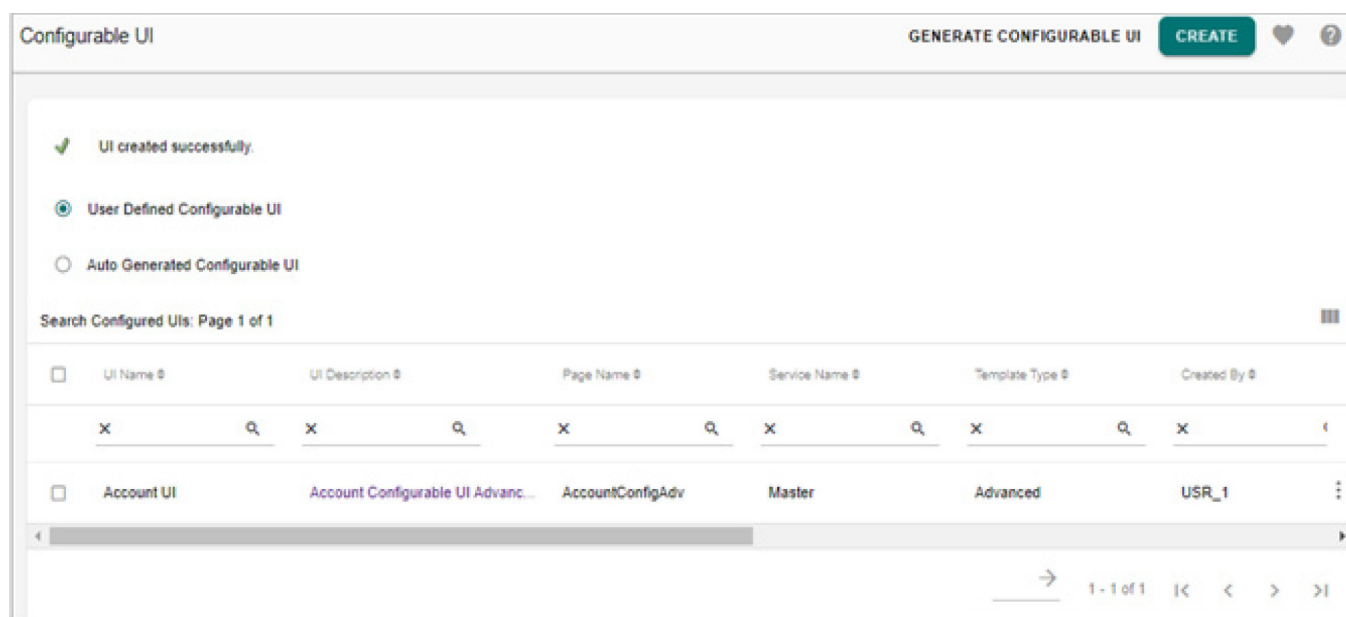
The configured default sort cannot be modified at runtime on the configured UI. To modify the sorting configuration, navigate back to Configure Result Columns UI and modify the default settings.

- **Hide Non Editable Columns:** Select the Hide Non Editable Columns checkbox to hide the selected editable columns on the Create and Edit page of the configured UI.

11 Click **Save**.

Message: “UI created successfully” is displayed on the Configurable UI as in [Figure 328](#).

Figure 328: Configurable UI



12 Select the UI and click **Preview** to review the generated UI.

[Figure 329](#) displays the preview of the advanced configurable UI created using the above steps.

Figure 329: Preview of Advanced Configurable UI

The screenshot shows a web application titled "Account UI". At the top right, there are buttons for "UPLOAD EXCEL", "CREATE", and a heart icon. Below the title bar is a search bar. The main form area contains several input fields: "Accountid", "AccountName", "Amount", "Department" (a dropdown menu showing "Select..."), "Storeid", "LAST\_MODIFIED\_BY", and two "LAST\_MODIFIED\_DATE" fields with calendar icons. Below the form is a "CLEAR" button and a "SEARCH" button. The bottom section of the interface is a table with the following columns: Accountid, AccountName, Amount, Department, Storeid, Locale, Target Id, Authorization, Source, Created By, Creation Date, and Entity State. The table contains five rows of data, each with a checkbox in the first column. The data rows are:

	Accountid	AccountName	Amount	Department	Storeid	Locale	Target Id	Authorization	Source	Created By	Creation Date	Entity State
<input type="checkbox"/>	6	Back Office Account	90000	Back Office	2				UI	USR_1	3/24/2021 6:43 AM	ACTIVE
<input type="checkbox"/>	7	Branch Account	90000	Branch	1				UI	USR_1	3/24/2021 6:41 AM	ACTIVE
<input type="checkbox"/>	8	Customer Services Account	90000	Branch	2				BackEnd	USR_1	3/24/2021 7:39 AM	ACTIVE
<input type="checkbox"/>	9	Departmental Account	90000	Departmental	1				UI	USR_1	3/24/2021 6:42 AM	ACTIVE
<input type="checkbox"/>	12	Help Desk Account	90000	Branch	2				BackEnd	USR_1	3/24/2021 7:39 AM	ACTIVE

At the bottom right of the table, there is a pagination bar showing "1 - 6 of 6" and navigation arrows.

## Configurable UI Management

On the Manage Configurable UI, apart from configurable UI creation, you can perform the following activities:

- Preview—preview configurable UI layout.
- Copy—copies an exact replica of existing selected configurable UI.
- Delete—removes the selected configurable UI from web.
- Edit— allows you to modify the selected configurable UI specifications (auto-generated UIs that are edited becomes user-defined UIs).

## APPENDIX D Data Publication and Integration

---

### What's In This Appendix

This appendix provides information on RDM data publication.

Topics include:

- [Data Publishing and Integration](#)

## Data Publishing and Integration

Publication object is defined as any collection of information that can be published to a downstream application, process, or end user. Teradata RDM allows publishing of data from existing RDM tables to publication tables through Manage Publication Objects UI.

Navigate to Manage Publication UI from **Data Integration** -> **Publications**. On the Manage Publication UI, you can perform the following:

- [Add Publication Objects](#)
- [Publish Table](#)

### Add Publication Objects

To add publication object, follow the steps below:

- 1 On the **Manage Publication Objects** UI, in the **Add New** pane, enter the new publication object for the table (select service as Master and publication name as Accounts Publication) and click **Add** as in [Figure 330](#).

The service can be input, master, net change, version or views.

Figure 330: Manage Publication Objects

Manage Publication Objects

ADD

Add New

\* denotes required field

Service\*  
Master

Name\*  
Account Publication

Description  
Account Publication

Publication Objects: No Records Found

Name	Description
x	x

1 - 0 of 0

- 2 On the **Add Publication Object** UI, select the table (say Account) and click **Add** as in [Figure 331](#).



Figure 331: Add Publication Object

Add Publication Object

Service

Master

Name

Account Publication

Description

Account Publication

^ Service Documents

☐

Service Documents

☒

Account

☐

AttributeSet

☐

AttributeSetDetail

☐

Code\_Set\_Stats

☐

Code\_Set\_Table\_Association

BACK

ADD

The publication object is now added as in [Figure 332](#).

Figure 332: Manage Publication Objects

## Publish Table

To publish a table, perform the following steps:

- 1 On the **Manage Publication Objects** UI (Figure 332), click on the Menu icon corresponding to the publication object and click **Publish**.
- 2 On the **Publish Table** page (Figure 333), enter the following:
  - **Publication Method:** select the publication method from Publication Method the dropdown.
  - **Maintain Audit Copy:** to maintain an audit copy, select the Maintain Audit Copy checkbox.
  - **JMS Provider Queue:** to publish to JMS Provider Queue, enter the name of the JMS queue in the JMS Provider Queue field.

- **Email Address:** enter the email address in the Email Address field, if publishing to excel or comma separated values format.
- **Delimiter:** select the delimiter, if publishing to comma separated values format.
- **Database Mappings:** in the **Database Mappings** pane, enter the following:
  - **Source Database:** this is the database in which the data to be published resides.
  - **Source Table Name:** this is the name of the database table that is being published.
  - **Target Table Name:** this is the name of the table that will be created in the Publication database that will contain the published data.

Figure 333: Publish Table

**Publish Table** BACK PUBLISH ?

Publication Object  
Account Publication

Publication Method  
Publication Database Tables

☐ Maintain Audit Copy

JMS Provider Queue

Email Address

**Delimiter**

☐ Comma ☐ Space ☐ Tab ☐ Semi Colon ☐ Other

**Database Mappings**

^ Publication Objects

Document Name	Source Database	Source Table Name	Target Table Name
Account	RDM	MST_Account	PUB_Account

3 Click **Publish**.

Message: “Publish table submitted successfully” is displayed.

## View Published Data

Perform the following steps to view the published data through the Configurable UI:

- 1 Register the publication database (for example: RDM\_PUB).

To register new database, perform the following:

- Navigate to **Model Builder** UI from **Model Setup** -> **Model Builder**.
- On the **Model Builder** UI, select **Metadata Import** from **Actions** dropdown.
- On the **Manage Metadata Import** UI, select **Import Metadata** from **Actions** dropdown.
- On the **Import Metadata Objects** UI (Figure 334), enter the following:
  - Select the object type as Table from **Object** dropdown.
  - Select the Register External Tables checkbox and Register new db checkbox.
  - Select the Group and enter the DB Name.
- Click **Validate**.
- Select the objects from **Available Objects** and move it to **Selected Objects** and click **Import**.

Figure 334: Import Metadata Objects

Once the publication database is registered, the published table is available on the Web UI.

- 2 Generate the Configurable UI for the published table.

To generate configurable UI for published table, perform the below steps:

- Navigate to **Generate Configurable UI** from **Model Setup** -> **Configurable UI**. On the **Configurable UI** page, click **Generate Configurable**.

- On the **Generate Configurable UI**, select the service name (example: Non MDM) from **Service Name** dropdown.
  - Select the table (RDM\_PUB.PUB\_Account) and click **Generate**.  
“A notification will be sent to the configured email Id after the completion of configurable UI generation”.
- By default, configurable UI links will be generated in the left navigation pane under the Tables folder.
- 3 View the Configurable UI and check the published data (navigate to **Tables** and click PUB\_Account table)

The [Figure 335](#) displays the published data.

Figure 335: Configurable UI—PUB\_Account

PUB_ACCOUNT					
<input type="checkbox"/>	Accountid	AccountName	Amount	Department	StoreId
<input type="checkbox"/>	6	Back Office Account	90000	Back Office	2
<input type="checkbox"/>	7	Branch Account	90000	Branch	1
<input type="checkbox"/>	8	Customer Services Account	90000	Branch	2
<input type="checkbox"/>	9	Departmental Account	90000	Departmental	1
<input type="checkbox"/>	12	Help Desk Account	90000	Branch	2
<input type="checkbox"/>	30	Sample Account	100000	Back Office	2

1 - 6 of 6

## APPENDIX E Extended Attributes

---

### What's In This Appendix

This appendix provides information on extended attributes feature.

Topics include:

- [Manage Extended Attributes](#)

## Manage Extended Attributes

The Extended Attributes feature allow the user to dynamically add new attributes or columns to an existing table without actually involving the normal process of data model change. The newly added attributes would be handled in the same way as any other pre-existing attributes in the table and all existing functionalities will be supported on the new attributes.

Using extended attributes involves two major steps:

- On MDM Studio or Model Builder, select the table and mark it as extendable and enter the number of additional attributes and their data types, perform service setup and schema generation to add the extended attributes for the table marked as extendable.
- On the RDM UI, define the extended attributes and navigate to configurable where the extended attributes are actually used.

### Step 1: Studio or Model Builder Configuration

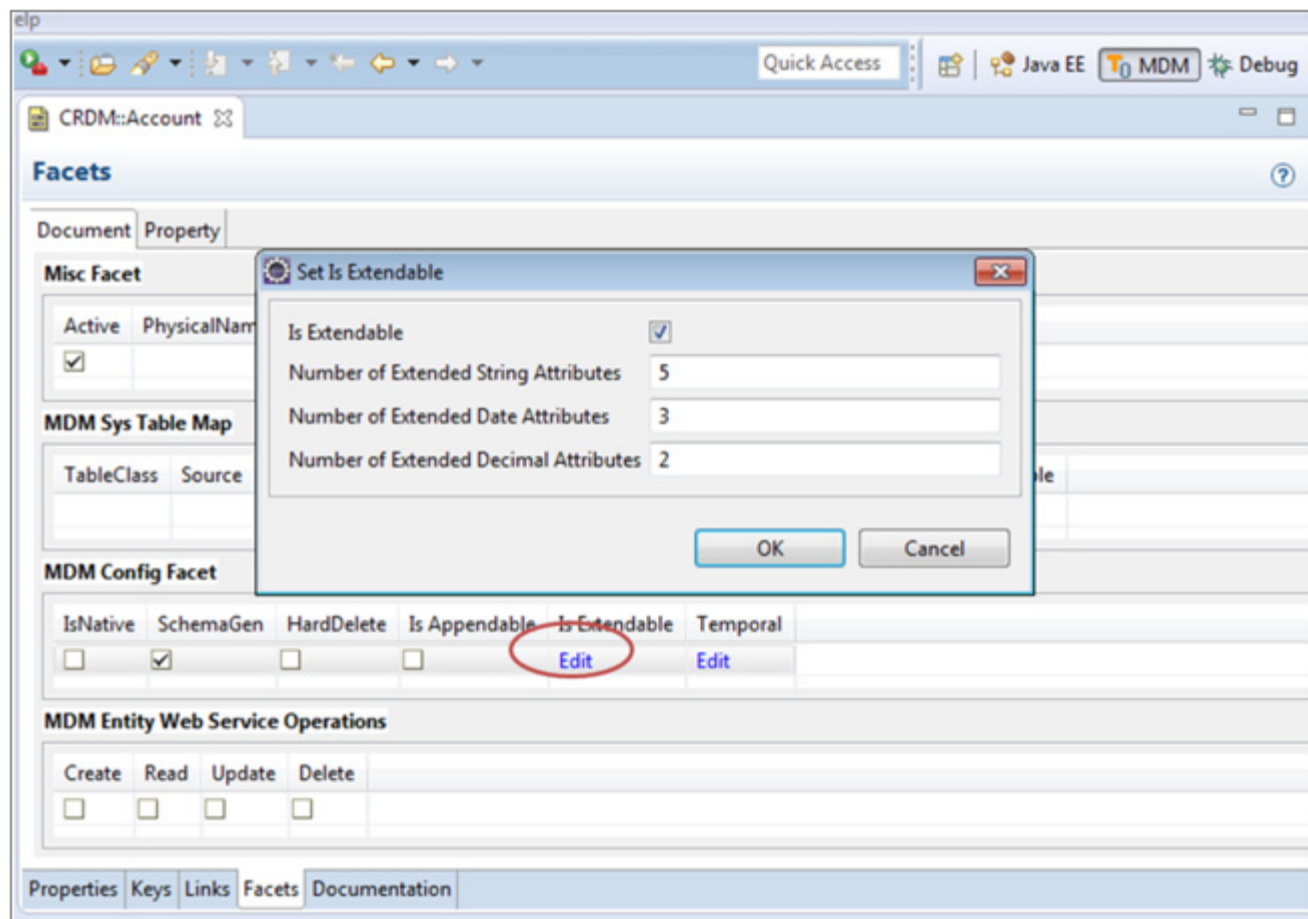
On the MDM Studio, perform the following steps to enable extended attributes:

- 1 On the **MDM Studio**, open the required table and on the editor pane, click **Facets** and in the **MDM Config Facet** pane, under **Is Extendable** column, click **Edit** as in [Figure 336](#).
- 2 On the **Set Is Extendable** pop-up, select the option **Is Extendable** and enter the number of additional attributes for each of the data type in their respective text fields and click **Ok**.
- 3 Perform service setup to generate additional properties and add the extended attributes for the selected table in the database.

### Support on Document View

Extended Attributes is supported only for those properties in Document Views which are based on MDM Documents marked as Extendable. If you rename an extended attribute in the base Document and if the same property is referred in the Document View, then the mapped property in the Document will be also be renamed as per the extended attribute definition.

Figure 336: Extended Attributes—MDM Studio



Similar configuration can be done for Model Builder models. The [Figure 337](#) displays the extended attributes configurations for a sample model on Model Builder UI.

Figure 337: Model Builder—Extended Attribute

The screenshot displays the 'Model Builder—Extended Attribute' interface. At the top, there are tabs for 'OBJECT DETAILS', 'COLUMNS', 'KEYS', and 'LINKS'. The 'OBJECT DETAILS' tab is active, showing a 'BACK' button, a 'SAVE' button, and a help icon. Below the tabs, the 'Object Details' section contains the following fields and options:

- Object Name:** Channel
- Description:** Channel
- Groups:** Account
- Object Type:** Table
- Table Type:** Set
- ☐ Create New Group
- ☐ Deploy in Secondary Connection
- ☒ Advanced Settings
- ☒ Statistics

The bottom section, 'CONFIGURATION DETAILS', is also active and contains the following fields and options:

- ☐ Audit Trail
- ☐ Is Native
- ☐ Is Temporal
- ☒ Is Extendable
- Number of String Attributes:** 5
- Number of Decimal Attributes:** 3
- Number of Date Attributes:** 2
- ☐ Upload Row Number Enabled
- Notify:** (Text area)

## Step 2: RDM UI Configurations: Manage Extended Attributes

Once the table is defined as extendable, you can use the Manage Extended Attributes UI to create new extended attributes for the tables marked as “Extendable”.

Perform the following steps to create new extended attributes:



- 1 Navigate to **Manage Extended Attributes** UI from **Model Setup -> Extended Attributes**.

Figure 338: Manage Extended Attributes

Manage Extended Attributes

CREATE

Extended Attributes Search

Objects  
Select...

Table Name  
All

CLEAR SEARCH

Search Results: No Records Found

Table Name	Attribute Logical Name	Attribute Physical Name	Attribute Data Type	Description	Valid Value Type	Code Set Name	Reference Table
------------	------------------------	-------------------------	---------------------	-------------	------------------	---------------	-----------------

1 - 0 of 0

- 2 On the **Manage Extended Attribute** UI ([Figure 338](#)), click **Create**.

Figure 339: Add Extended Attribute

Add Extended Attribute

\* denotes required field

Objects

Tables

Table Name \*

Account

Data Type \*

string

☒ Is Valid Value Type

Valid Value Type \*

LookUp

Code Set Name \*

LOCALE

Extended Attribute Name \*

Locale

☐ Is Required

Description

Locale

CANCEL

SAVE

- 3 On the **Add Extended Attributes** UI (Figure 339), enter the following details and click **Save**.

Table 13: Add Extended Attribute Fields

Attribute Name	Details	Comments
Objects	Object Type	Tables, Code Sets, Hierarchy Objects
Table Name	Select Table Name	Table on which extended attribute to be created

Table 13: Add Extended Attribute Fields

Attribute Name	Details	Comments
Data Type	Select the data type	string, decimal or date
Is Valid Value Type	Select the Is Valid Value Type option	<p>If string option is selected, the Is Valid Value Type checkbox is displayed.</p> <p>RDM supports two types of valid values.</p> <p>Lookup: Code Set lookup values.</p> <p>Reference: Values of a property of the selected table.</p>
Code Set Name	If Lookup option is selected as valid type, the Code Set Name field is displayed.	Select the code set name from the Code Set Name dropdown.
Reference Table Name	If Reference option is selected as valid type, the Reference Table Name field is displayed.	Select the reference table name from the Reference Table Name dropdown.
Reference Property	If Reference option is selected as valid type, the Reference Property field is displayed.	Select Reference Attribute Name of the selected Reference table.
Extended Attribute Name	Extended attribute name	Enter the extended attribute name being created
Is Required	Select Is Required checkbox	Select this checkbox if it is a required field.
Default Value	Enter the default value in the Default Value field.	These details are stored in the Extended attributes table and also displayed on the Manage Extended Attributes Results page. These details are not supported in any OOTB UIs, but you can use in custom UIs.
Description	Enter the description of the extended attribute	Description of the extended attribute being created.

Message: “Extended Attribute added successfully” is displayed on the Manage Extended Attributes page as in [Figure 340](#).

Figure 340: Manage Extended Attributes

Manage Extended Attributes

CREATE

Extended Attributes Search

Extended Attribute added successfully.

Objects  
Select...

Table Name  
All

CLEAR SEARCH

Search Results: Page 1 of 1

Table Name	Attribute Logical Name	Attribute Physical Name	Attribute Data Type	Description	Valid Value Type	Code Set Name	Reference Table	Reference Property	Default Value
Account	Locale	EXT_ATTRB_1	string	Locale	LookUp	LOCALE			

1 - 1 of 1

After creating the extended attribute, Configurable UIs should be edited to add the extended attributes on the UIs:

Auto-Generated Configurable UI— Delete Configurable UI and Regenerate from “Generate Configurable UI” Page.

User-Defined Configurable UI - Edit Configurable UI in “Manage Configurable UI” Page and add the new attribute.

Figure 341 displays the added extended attribute available on the configurable UI.

Figure 341: Extended Attribute—Configurable UI

The screenshot shows a 'Create Account' form. The title bar at the top left says 'Create Account' and has a heart icon on the right. Below the title bar are five input fields arranged in three rows. The first row has 'AccountId\*' (highlighted in light blue) and 'AccountName'. The second row has 'Amount' and 'Department'. The third row has a dropdown menu labeled 'Locale' with 'Select...' text (highlighted with a red border). At the bottom right of the form are two buttons: 'BACK' and 'CREATE'.

**Notes and Limitations:**

- Extended attributes with the same name for a single table cannot be defined. An error message “You cannot create the extended attribute, the entered attribute name already exists for the selected table” is displayed on the Add Extended Attributes page.
- When you rename an extended attribute and make it as lookup data type at runtime, the default business validation rules will not be created as the business rules for validating the code set values are created during ISG or SG. You have to create your own custom business rules for the same.
- Business rules must not be defined on extended attributes until extended attributes are created (defined a business name) using Manage Extended Attributes UI.
- Attribute logical names should not have white spaces.
- The attribute logical name should not be more than 30 characters. Otherwise during ISG, it will fail to proceed further.
- Metadata needs to be refreshed in order to use extended attributes on search metadata feature.

## APPENDIX F Glossary

---

### What's In This Appendix

This appendix provides a list of technical and business terminologies used in RDM.

Topics include:

- [Technical & Business Glossary](#)

## Technical & Business Glossary

This section contains the list of terminologies used in Teradata RDM.

- ROs—Relational Object
- RDM—Reference Data Management  
Please navigate to Manage Reference Data link in left navigation for creating values in RDM, also known as Code Sets, Reference Data, Valid Values, etc.
- RODV—It means to select Relationship Data in Relationship Mapping while creating a Relationship. These are RODV (Relational Object Data Value) tables which store the relation between ROs in this case.
- Custom RODV—It means to select Pre-Defined Relationship Data in Relationship Mapping while creating a Relationship. The relationship is stored in the selected object from the drop down

# Index

- A
- Add
  - New Code Set Group 91
- Add Code Values 68
- Alerts 7
  - Fatal Alerts 8
  - Information Alerts 8
  - Severe Alerts 8
  - Warning Alerts 8
- Approval
  - Advanced Options 39
  - Assignment Approval 39
  - Auto Approval 39
  - Column Level Approval 39
  - Create 34
- Approval Inbox 174
- Approval Management 34
- Assign Members to Group 93
- Assignment Approval 46
- Audit Trail 240
- Auto Approval 47
- Auto Matching Logic 87
- B
- Business Rules 22
  - Add 23
  - Modify 33
- C
- Code Set Groups 90
- Code Sets
  - Enable Authorization 120
- code value
  - edit 72
- Column Level Approval 44
- Configurable UI 10, 144
  - Advanced 297
  - CRUD Operations 11
  - User defined 144
  - User Defined Configurable 144
- Copy Record 13
- Create
  - Approval Object 132
  - Code Set 64
  - Custom Action 178
  - Extended Attribute 143
  - Hierarchy 131, 157, 164
  - Hierarchy Attribute 208
  - Hierarchy Object 133, 161
  - Hierarchy Version 132
  - Model 236
  - New Source System 73
  - Record 11
  - Relationships 148, 163
  - Rollup Attributes 132
  - Rule Based Approvals 39
  - Source Mapping 75
  - Version 168
- CRUD 8, 9, 11
- Custom Actions 177
- Custom RODV 321
- Custom Table/View 66
- D
- Data Publishing 315
- Delete
  - Record 18
- E
- Edit
  - Hierarchy Objects 137
  - Model 236
  - Template Tables 282
- Enum 245
- Excel Upload 19
  - Simple Upload 19, 48
  - Upload with Rules 48
- Extended Attribute
  - Process Flow 144
- F
- Favorite
  - Role Level 228
  - User Level 227

## H

## Hierarchy

- Attach Rollup Attribute 211
- Compare 179
- Compare Metadata 180
- Compare on Viewer 181
- Data Model 125
- Edit 166
- Export 195
- Generate and Validate Tables 127
- Import 197
- Logs 207
- Management Functionalities 131
- Manager 156
- Merge 201
- Promote 170
- Statistics 216
- View 179
- View with Rollup Attributes 212
- Viewer 183

## Hierarchy Viewer

- Add Child 186, 190
- Add Sibling 186, 191
- Menu Options 192
- Show All Peers 185, 186
- Show Children 185, 188
- Show Siblings 185, 187

## L

## Landing Page

- Assign 224
- Create Page 223

## Landing page

- Role Level 225

## Limitation 242, 312

## Limitations

- Favorites 231

## Link and View Data 151

## Load from Excel 81

## Load Hierarchy Object Data 136

## Load Relationship Data 149

## LRDM 321

## M

## Manage

- Authorization 120
- Code Sets 1
- Cross Reference 218

## Custom Actions 193

## Dictionaries 278

## Extended Attributes 141, 307

## Favorites 229

## Hierarchy Objects 132

## Relationship 147

## Rollup Attributes 207

## Sequence 248

## Templates 280

## Manage Authorization

## Code Set Group 122

## Individual Code Set 121

## Mass Update 15

## MDM Landing Page 223

## Merge Hierarchy 201

## Model Builder

## Create using Excel 250

## Features 233

## Hierarchy 126

## ISG Logs 264

## Template Guidelines 252

## Validate Models 262

## Modify Approval 37

## O

## Override Code Value Sequence 66

## P

## Publication Object

## Add 315

## Run 317

## R

## RDM User Case

## Hierarchy Management 3

## Record

## Delete 18

## Edit 14

## Remove Members Assigned to Group 95

## Reserve Link Name 250

## RODV 321

## ROs 321

## Rule Editor Wizard 25

## Action Type 30

## Error Data 25

## From Context 26

## Join Context 27

## Search Context 28



## S

Sample Hierarchy Structure 126

Surrogate 241

SYS\_EXT\_ATTRBS 143

SYS\_EXT\_ATTRBS\_MAP 142

## T

Temporal 241

Triggering Advanced Approvals 44

## U

User Favorites 227

## W

Web Services 8

Workflow 7