



THE
POWER
TO KNOW.

SAS[®] MDM 4.1

User's Guide

Second Edition

The correct bibliographic citation for this manual is as follows: SAS Institute Inc. 2014. *SAS® MDM 4.1: User's Guide, Second Edition*. Cary, NC: SAS Institute Inc.

SAS® MDM 4.1: User's Guide, Second Edition

Copyright © 2014, SAS Institute Inc., Cary, NC, USA

All rights reserved. Produced in the United States of America.

For a hard-copy book: No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

For a web download or e-book: Your use of this publication shall be governed by the terms established by the vendor at the time you acquire this publication.

The scanning, uploading, and distribution of this book via the Internet or any other means without the permission of the publisher is illegal and punishable by law. Please purchase only authorized electronic editions and do not participate in or encourage electronic piracy of copyrighted materials. Your support of others' rights is appreciated.

U.S. Government License Rights; Restricted Rights: The Software and its documentation is commercial computer software developed at private expense and is provided with RESTRICTED RIGHTS to the United States Government. Use, duplication or disclosure of the Software by the United States Government is subject to the license terms of this Agreement pursuant to, as applicable, FAR 12.212, DFAR 227.7202-1(a), DFAR 227.7202-3(a) and DFAR 227.7202-4 and, to the extent required under U.S. federal law, the minimum restricted rights as set out in FAR 52.227-19 (DEC 2007). If FAR 52.227-19 is applicable, this provision serves as notice under clause (c) thereof and no other notice is required to be affixed to the Software or documentation. The Government's rights in Software and documentation shall be only those set forth in this Agreement.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513-2414.

April 2014

SAS provides a complete selection of books and electronic products to help customers use SAS® software to its fullest potential. For more information about our offerings, visit support.sas.com/bookstore or call 1-800-727-3228.

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

Other brand and product names are trademarks of their respective companies.



Gain Greater Insight into Your SAS® Software with SAS Books.

Discover all that you need on your journey to knowledge and empowerment.



SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies. © 2013 SAS Institute Inc. All rights reserved. S107969US.0613

Contents

<i>What's New in SAS MDM 4.1</i>	<i>vii</i>
<i>Recommended Reading</i>	<i>xiii</i>

PART 1 Introduction to SAS MDM 1

Chapter 1 • Overview of SAS MDM	3
About SAS MDM	3
How SAS MDM Works	4
Architecture	4
Deployment Models	8
Integrating Other Applications with SAS MDM	8
Chapter 2 • Understanding SAS MDM	9
The Enterprise Environment	9
What Are Entities?	11
How Are Entities Related?	12
How Are Entities Constructed?	12
Inside the Cleansing Process	13
Inside the Clustering Process	14
Inside the Survivorship Process	18
Chapter 3 • Deploying SAS MDM	21
Adding Metadata	21
Generating Jobs	28
Loading the MDM Hub	30
Loading Data	34
Migrating from a Master Data Management Foundation Hub	35
Reporting on the Hub	36
Chapter 4 • Controlling Access	39
About Access Control	39
Defining Groups	41
Interaction with DataFlux Data Management Server Permissions	43
Managing Access	43
Designing Jobs	43

PART 2 SAS MDM Interface 45

Chapter 5 • Accessing SAS MDM	47
Log On to SAS MDM	47
SAS Data Management Console Home Page	48
Specifying Your Preferences	49
Overview of the SAS MDM Interface	51
Chapter 6 • Using Master Data Management	55

About Master Data Management	55
Searching Entities	56
Working with Search Results	59
Editing Entities	62
The Records Tab	62
The Hierarchies Tab	67
The Relationship Diagram Tab	67
The History Tab	70
Creating New Entities	70
Creating a Remediation Issue	72
Searching Hierarchies	74
Chapter 7 • Using Master Data Management Data Model	77
About Master Data Management Data Model	77
Understanding Entity Type Definitions	80
Working with the Entity Type Editor	80
Understanding Hierarchy Type Definitions	93
Working with the Hierarchy Type Editor	94
Understanding Relationship Type Definitions	98
Working with the Relationship Type Editor	99
Chapter 8 • Using Master Data Management Administration	105
About Master Data Management Administration	105
Working with the Source System Editor	107
Working with the Language Editor	109
Chapter 9 • Using SAS Data Remediation	113
About SAS Data Remediation	113
Understanding SAS Data Remediation Administration	115
Understanding SAS Data Remediation	127
Chapter 10 • Using SAS Task Manager	141
About SAS Task Manager	141
Understanding SAS Task Manager	144
SAS Task Manager Roles	148
Deep Linking to Workflows in Task Manager	149
 PART 3 SAS MDM Examples	 151
Chapter 11 • Working with SAS MDM	153
Example 1: Creating an Abstract Entity Type	153
Example 2: Creating a Concrete Entity Type	166
Example 3: Creating Entity Relationships	173
Glossary	181
Index	183

What's New in SAS MDM 4.1

Overview

The main enhancements and changes for SAS MDM 4.1 include the following:

- integration between DataFlux and SAS
- a redesigned and improved product
- a redesigned user interface
- SAS MDM integrated with SAS Data Management Console
- new SAS Data Remediation application
- new SAS Task Manager application
- expanded role-based interaction
- SAS MDM enabled for SAS Workflow Technology
- redesigned batch jobs and real-time services
- custom functionality support through SAS MDM tools
- hierarchy support for entity relationships
- deep linking through URLs
- remediation performance improvements

SAS and DataFlux Integration

SAS has fully integrated its DataFlux suite of data quality, data integration, data governance, and master data management solutions. The newly branded SAS DataFlux products help you build a more integrated information management approach that goes beyond data management and governance to support analytics and decision management.

Rather than being licensed separately, SAS DataFlux products are now being combined into software offerings that include other SAS products as well. Changes in offerings might affect your license renewals. Some DataFlux products are being rebranded and enhanced as SAS products, such as SAS Federation Server.

The [SAS Customer Support](#) site now provides support and resources for both SAS and SAS DataFlux products. The MyDataFlux Portal is no longer supported. Some of these revised resources are as follows:

- [SAS Downloads](#) site to download SAS DataFlux products, data updates, and Quality Knowledge Bases
- [SAS Documentation](#) site and [SAS Install Center](#) to find current documentation
- [SAS Administrative Services](#) to manage your account, including how you can contact Contracts Support to obtain or renew a license
- [SAS Support](#) to submit questions or a request for technical support

Redesigned Product

SAS MDM 4.1 is a new product that integrates master data management technologies with those in SAS 9.4. SAS MDM is a web-based application that operates using the SAS logon page and SAS Data Management Console. SAS MDM provides a single, accurate and unified view of corporate data, integrating information from various data sources into one master record. SAS gives businesses the tools to develop master data management processes, and it provides the technology required to analyze existing data resources, build a unified view of that information, and manage that master view of data over time.

SAS MDM 4.1 runs with SAS 9.4 and DataFlux Data Management Studio and Server 2.41 and later. In this release, SAS MDM replaces DataFlux qMDM Solution.

The product, SAS MDM, is available in the SAS MDM software offerings (Standard and Advanced), which include a number of SAS and SAS DataFlux products. Both offerings include the new SAS MDM middle-tier applications, SAS Data Remediation, and SAS Task Manager. SAS Data Remediation and SAS Task Manager are two applications that integrate with SAS Workflow technology behind the scenes to orchestrate user or system activity.

Redesigned User Interface

The SAS MDM web application (known as Master Data Manager in previous releases) has been completely redesigned. It uses design elements consistent with other SAS applications to merge seamlessly with them. For example, some elements, Cluster Move and Cluster Compare, have been redesigned to be more intuitive while also requiring fewer steps to complete.

SAS Data Management Console

The SAS Data Management Console, a central, web-based location for all master data management activities, is new for this release. From the console, users can quickly see information that is relevant to their assigned role in the SAS MDM environment. For

example, a data steward can see the most recent data exceptions from a batch load of the previous night, navigate directly to the issue, and correct it, all from the SAS Data Management Console. This new environment also provides a launch location for all associated SAS MDM components, such as master data management and data remediation. Users can customize the console by specifying the type and quantity of information that they want to see.

SAS Data Remediation

A new set of features to manage exceptions in the SAS MDM hub is now available. The data remediation module allows users to manage and correct issues triggered by business rules in SAS MDM batch jobs and real-time processes. These issues might also be triggered by external systems that communicate with data remediation through its REST-based web service API. Data remediation allows user- or role-based access to data exceptions, which are categorized by application and subject area. Once data remediation issues have been reviewed, they can be corrected through the same application, eliminating the need for another user to complete the correction process. All data remediation issues can also be associated with workflow definitions that route the issues to the correct decision maker for approval or instructions for additional action.

SAS Task Manager

SAS Task Manager is a complementary application to others that integrate with SAS Workflow technologies. It gives users direct access to a workflow that might have been initiated from one of the other SAS applications. With SAS Task Manager, users can start, stop, and transition workflows that have been uploaded to the SAS Workflow server environment. SAS Task Manager also lets users interact with certain elements of active workflows called data objects. These data objects can be used in such a way to trigger additional activity that goes beyond what is supplied from standard workflow interaction.

Enhanced Roles

The concept of role-based interaction with SAS MDM has been greatly expanded in this release. Related sets of features have been collected into modules that can be shared with or hidden from groups of users, depending on the type of work they typically do in the web-based SAS MDM applications.

Within these modules, individual features can also be enabled or disabled as needed. For example, it is possible to configure data remediation so that users can view data quality issues that have been generated from SAS MDM batch or real-time processes, but cannot edit the data involved in the issues.

Enabled for SAS Workflow Technology

As in previous releases, workflow technology underlies several key processes of master data management, such as entity lifecycle management and ad hoc alert creation. In this release, SAS workflow technology has replaced previous workflow capabilities. With SAS workflow technology, users have access to a robust workflow engine. In addition, a drag-and-drop application, SAS Workflow Studio, can be used for editing existing or designing new workflows. Workflow definitions can initiate other workflow processes, call web services, integrate with SAS DataFlux processes, and invoke SAS code.

Redesigned Job Templates

The SAS MDM batch jobs and real-time services that are generated through Master Data Manager now more easily accommodate customizations to meet project requirements. As with data quality templates previously, data input and best record creation templates are now separately generated and linked to the main load and update processes. In addition, SAS MDM load templates have been changed so that data flows through user-designed business rules. This has the potential to send data rows to the data remediation application for review and correction.

SAS MDM Tools

Because every SAS MDM project has unique requirements, data stewards need the capability to design and deploy custom functionality to groups of SAS MDM users through the SAS MDM web application. Using DataFlux Data Management Studio, data stewards can build virtually any process and integrate it with existing SAS MDM web application functionality. This enables them to extend the capabilities of SAS MDM to meet project requirements or end-user expectations.

The following types of actions can be added to SAS MDM:

- displaying an entity address on an internet-based mapping application
- displaying search engine results for a selected entity
- making the same change to multiple entities at the same time
- copying search results data and e-mail it to a business user as an attachment
- creating saved queries that can be exposed to specific groups of SAS MDM users
- allowing SAS MDM users to see entity undermatch and overmatch reports

Hierarchy Support

It is now possible to overlay hierarchy structures on top of already defined entity relationships. For example, for suppliers and products that are defined in a relationship, you can also define a hierarchy on that relationship. Once the hierarchy is in place, you can search on and display all suppliers and the products that they manufacture in a hierarchical view. These hierarchies can be inferred from the data and the rules set up for the relationship, constructed programmatically through a Data Management job, or created by users in the SAS MDM web application.

Deep Linking

It is now possible to perform deep linking into SAS MDM, SAS Data Remediation, and SAS Task Manager. Deep linking refers to constructing a URL hyperlink that, once initiated, can take a user directly to an item in the referenced application rather than requiring the user to search for it. For example, you can construct a URL link in an e-mail and send it to a data steward. When the data steward clicks the URL link, it opens the SAS MDM application and loads the entity data encoded in the URL. SAS Task Manager now contains an extension to this functionality so that specially designated data objects can deep link into web applications.

Remediation Performance Improvements

SAS Data Remediation users now have more control over the number of data remediation issues that are displayed in the application. Additional enhancements improve the response time for filter, search, and delete functionality.

Recommended Reading

- *DataFlux Data Management Studio Installation and Configuration Guide*
- *DataFlux Data Management Studio User's Guide*
- *DataFlux Data Management Server Administrator's Guide*
- *DataFlux Data Management Server Users's Guide*
- *SAS Workflow Studio User's Guide*
- *SAS Intelligence Platform Middle-Tier Administration Guide*
- *SAS Management Console User's Guide*

For a complete list of SAS books, go to support.sas.com/bookstore. If you have questions about which titles you need, please contact a SAS Book Sales Representative:

SAS Books
SAS Campus Drive
Cary, NC 27513-2414
Phone: 1-800-727-3228
Fax: 1-919-677-8166
E-mail: sasbook@sas.com
Web address: support.sas.com/bookstore

Part 1

Introduction to SAS MDM

<i>Chapter 1</i>	
Overview of SAS MDM	<i>3</i>
<i>Chapter 2</i>	
Understanding SAS MDM	<i>9</i>
<i>Chapter 3</i>	
Deploying SAS MDM	<i>21</i>
<i>Chapter 4</i>	
Controlling Access	<i>39</i>

Chapter 1

Overview of SAS MDM

About SAS MDM	3
How SAS MDM Works	4
Architecture	4
Overview	4
Development	5
Run Time	5
SAS MDM Components	6
Deployment Models	8
Integrating Other Applications with SAS MDM	8

About SAS MDM

SAS MDM is a combination of software, templates, documentation, data models, and services that provide the functionality and processes necessary to build and maintain a master entity database. An entity can be a customer, product, patient, site, or any business data object that you define. You can also define attributes for each entity, as well as transformations that support data quality and identity management functionality.

The SAS MDM hub is a set of tables and table relationships that contain entity information and required entity keys and linking information. This hub provides end-users or other applications with a complete view of an entity and its relationships with other entities. In addition, the hub can be used as the single source for creating and maintaining survivor records that can be used in other applications or processes. The original source system IDs, which are important to other enterprise applications or data sources, are also maintained in the hub to facilitate linking activities with other data providers or consumers.

Note: SAS MDM is integrated with DataFlux Data Management Platform. Therefore, to use SAS MDM successfully, you must be familiar with DataFlux Data Management Platform.

SAS Data Remediation enables users to manage and correct issues triggered by business rules in SAS MDM batch jobs and real-time processes. These issues can also be triggered by external systems that communicate with data remediation through its REST-based web service application programming interface (API). Data remediation allows user- or role-based access to data exceptions, which are categorized by application and subject area. Once data remediation issues have been reviewed, they can be corrected through the same application, eliminating the need for another user to complete the

correction process. All data remediation issues can also be associated with workflow definitions that route the issues to the correct decision maker for approval or instructions for additional action.

SAS MDM includes the following features:

- batch and real-time modes
- entity matching, de-duplication, and survivor record creation
- entity linking through relationships
- data quality functionality such as address verification, standardization, and parsing
- fielded and advanced search capabilities
- capability of adding, modifying, and retiring SAS MDM entities
- workflow-based data remediation functionality to view and correct SAS MDM data errors
- metadata management for entities, attributes, and transformations
- historical tracking, including changes saved for contributor records and survivor records
- data access, data profiling, data monitoring, data quality, and data enrichment

How SAS MDM Works

SAS MDM performs the following functions:

- extracts business information from your data sources
- validates and standardizes the data
- captures data errors through user-defined business rules and sequesters the data for review and correction
- consolidates the information into a single view of the information available from all the data sources

The organizational data that you provide can be customer data, product data, services data, or data for other entity types. SAS MDM applies a rigorous methodology to the problem of integrating disparate enterprise data.

Architecture

Overview

SAS MDM is built on a sophisticated data management platform that provides both development and run-time support for applications.

Development

During development, DataFlux Data Management Studio provides the ability to design and test processes (batch jobs) and real-time services (data jobs), which can then be deployed and run using the following:

- DataFlux Data Management Server
- DataFlux Data Management Studio
- scripting with a command line interface

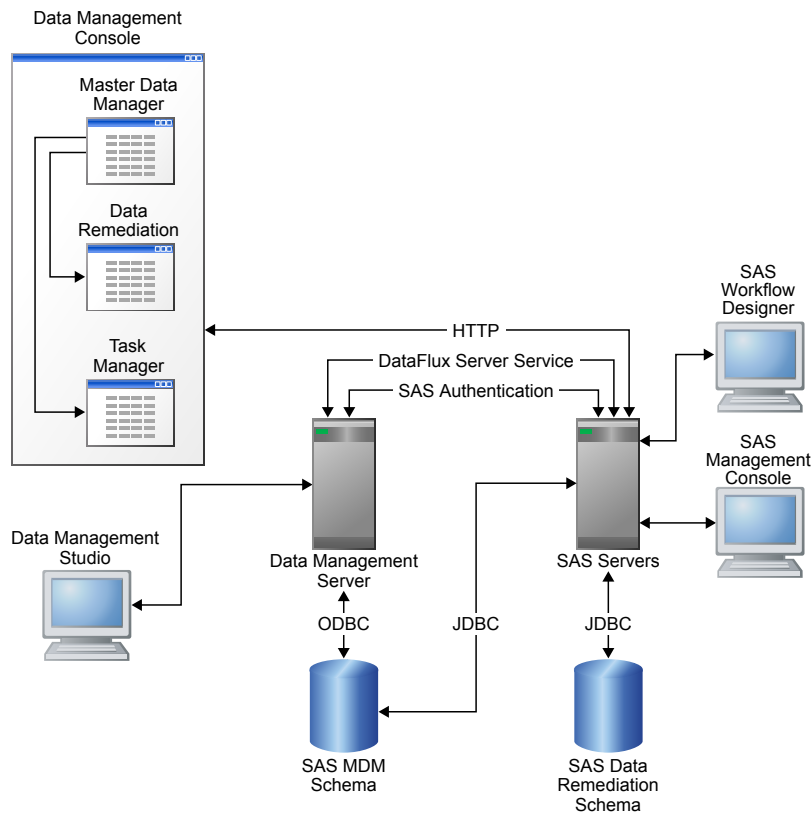
The DataFlux Data Management Platform relies on the Quality Knowledge Base (QKB) for data quality processing. The QKB provides capabilities such as standardization of data and computation of match codes. The QKB is used both during development and at run time.

Run Time

At run time, a multi-tier architecture that consists of presentation, business, and data tiers is used.

- The presentation tier hosts the web interface in a Servlet 2.5-compliant container and includes a Java application server and a browser. Batch interfaces might be available for some operations such as bulk loads and updates.
- The business tier runs on a DataFlux Data Management Server and is implemented using DataFlux Data Management Studio processes (batch jobs) and real-time services (data jobs). It implements a Service Oriented Architecture (SOA) that enables any system to use its services through SOAP requests. Common web service standards are used to maintain simplicity when integrating with other systems. Authentication within the business tier runs on a SAS Metadata Server. This server can authorize user permissions for applications, assign roles and security restrictions, and define authentication parameters for accessing other systems, such as the database, for each user or group. The real-time or transactional component of SAS MDM interfaces with other applications through an SOA environment. Specifically, the ability to call business services hosted on DataFlux Data Management Server is accomplished through web services. Process jobs can also be used as business services that accept data on a record-by-record basis. This enables users to use SAS MDM data and services within an enterprise architecture.
- The data tier hosts the supporting data and metadata for the system, which consists of a relational database such as Oracle or Microsoft SQL Server. The data model is documented and accessible to third-party tools and can also be accessed through DataFlux Data Management Studio jobs.

The following display shows how the components fit together:

Figure 1.1 Architecture Diagram

SAS MDM Components

DataFlux Data Management Studio

DataFlux Data Management Studio is a graphical user interface (GUI)-based application that provides access to other tools and services. You can use DataFlux Data Management Studio for job flow management and customization, to manage data source connections, to manage user accounts through SAS Metadata Server, to deploy and execute jobs on the DataFlux Data Management Server, and to run batch jobs and data services within a Windows environment.

DataFlux Process Jobs and Real-time Business Services

A DataFlux process job is a multi-step data management process that can be run from DataFlux Data Management Studio, from DataFlux Data Management Server, or from a command line interface or script. Real-time business services, also called data jobs, are deployed through DataFlux Data Management Server and are used by the SAS MDM web application to provide data to the application through SOAP calls.

Quality Knowledge Base

The Quality Knowledge Base (QKB) contains the files, file relationships, and metadata needed to correctly parse, match, standardize, and otherwise process data as required for SAS MDM. The QKB can be customized to meet an organization's needs. The QKB can support different locales to support processing for many different languages.

DataFlux Data Management Server

The DataFlux Data Management Server enables data management and integration processes to be deployed across an entire IT environment. SAS MDM uses batch jobs and real-time services deployed through DataFlux Data Management Server to manage the master data hub and to provide data through the web interface and reports. DataFlux Data Management Server handles all aspects of database connectivity and service availability through SOAP calls.

Note: In many instances, you should have at least two DataFlux Data Management Servers working together through network load balancing or failover hardware or software, to achieve the desired results for performance and meet server uptime goals, particularly for real-time service processing.

Databases

SAS MDM relies on access to a Database Management System (DBMS), such as Oracle or SQL Server, to act as a repository for metadata and data. The Master Data Management component of SAS MDM requires two database schemas for its operation.

SAS Data Management Console

The SAS Data Management Console is a central, web-based location for all master data management activities. This environment provides a launch location for all associated SAS MDM applications, as well as other SAS applications.

The SAS MDM applications Master Data Manager, Data Remediation, and Task Manager enable you to perform the following functions:

- Master Data Manager
 - create and manage a hub
 - define new entity types and their attributes
 - search for data within the hub
 - create and update data within the hub
- Data Remediation
 - monitor and correct errors in data loaded into SAS MDM
- Task Manager
 - start, stop, view, and interact with active workflow instances.

SAS Workflow Designer

This application is a drag-and-drop workflow designer. Users can modify existing MDM workflow templates or create new ones for use with other SAS MDM processes. Workflows can also be uploaded and activated in the SAS server environment and can be downloaded for additional editing.

SAS Application Server

An application server hosts the SAS Data Management Console and other associated applications. This is installed and configured during the deployment process. No other third-party application servers are supported.

SAS Management Console

This application enables users to manage groups, users, and roles on the SAS Metadata Server for all SAS MDM web applications. It also is the location where various web application configuration parameters are set.

Deployment Models

The most common type of deployment for SAS MDM is the physical deployment model. This model assumes that all the information necessary to construct a complete survivor record for an entity is stored in the hub. Source system keys are also stored in the hub so that information used to identify and describe entities can be recorded. This supports the ability to link or reference source system data, provides traceability for the contributor records, and describes key relationships with the data as it exists in the hub.

With this approach, uniquely identifying information for entities is stored in the hub, along with other dependent attributes. You can access the survivor record by querying the hub by itself, without the need to refer to the systems of record. This approach simplifies the task of constructing the survivor record, but results in higher storage requirements.

The physical deployment model is the primary implementation, but SAS MDM also works in conjunction with other technologies to provide functionality that supports deployment styles such as hybrid MDM, registry-based MDM, reference-style MDM, and other integration scenarios.

Integrating Other Applications with SAS MDM

You can integrate other end-user or enterprise applications, as well as other Extraction, Transformation, and Loading (ETL) software with SAS MDM.

Two methods for integration are as follows:

- direct-database or flat-file access, running batch load or batch update processes. In this case, the DataFlux Data Management Server can connect directly to data sources, transform the data, and load it into the hub.
- communicating with the DataFlux Data Management Server using web services. Any application or process that can make a web service call can send data to the hub through DataFlux Data Management Server and receive data from the hub in the same manner.

Both methods are typically used to interact with SAS MDM. In some cases it might make sense to communicate directly with DataFlux Data Management Server from an application integration layer of an enterprise application such as Siebel or SAP. In other cases, using an enterprise messaging mechanism such as TIBCO or webMethods might be more practical if that infrastructure has already been deployed within your organization.

You also have the option of making web service calls directly from a .NET or Java component or application that can use web services. This might be common in circumstances where internally designed and built operational or reporting processes need to interact with the DataFlux Data Management Server to access customer or service data inside the hub.

Chapter 2

Understanding SAS MDM

The Enterprise Environment	9
Which Data Is Most Important to Your Business?	9
Data Profiling	10
What Are Entities?	11
How Are Entities Related?	12
Relationships	12
Hierarchies	12
How Are Entities Constructed?	12
Inside the Cleansing Process	13
Validation Rules	13
Handling Invalid Data	13
Enabling Data Quality	13
Inside the Clustering Process	14
Unique and Related Entities	14
How Are Entities Identified?	16
Using Match Codes	17
Inside the Survivorship Process	18

The Enterprise Environment

Which Data Is Most Important to Your Business?

It is important to understand the environment in which SAS MDM is to be deployed. The source systems that provide the data for SAS MDM have a large impact on the implementation choices that you make.

The following types of data should be considered for incorporation into SAS MDM:

- data that represents the nouns of your business. Customers, Partners, Products, and Services are examples of entities that many organizations can have in common.
- data that exists in multiple silos. It does not matter if the silos are a result of acquisition or in-house development. Any redundant data can pose a risk to your organization.

When starting a master data management initiative, your business needs must be considered first and foremost. All of your data does not need to be managed as master data.

The following types of data should not be considered for incorporation into SAS MDM:

- data that rarely changes. Reference data is easier to synchronize across the enterprise because it rarely changes.
- data that is not shared. If a data set is used by only one part of your business, the merge and reconciliation aspects of a master data management initiative are less valuable.

Data Profiling

It might not be obvious where the entities in your organization are defined or are being used. Data profiling can help you discover which enterprise systems contain entity information and the attributes, including identifiers, associated with it. With this information, you can understand more about your master data management implementation.

Consider the following questions related to data profiling:

Table 2.1 Data Profiling

Question	Discussion
Which source systems are involved and which entities do they manage?	In SAS MDM, entities are constructed from records in source systems where the data originally resides. Some examples of source systems are Enterprise Resource Planning (ERP) systems, accounts receivable systems, and payroll systems. Data profiling can help discover similarities among the source systems in an enterprise, thereby suggesting entities for incorporation into the hub.
How are the entities identified?	By identifying the keys used within each source system, you can identify strategies for matching the same entity across different source systems.
What attributes do the entities manage?	As you reconcile the views of an entity that have been adopted by each source system, you can begin to formulate the data needs of the entity.
What steps are needed to address quality issues in each of the systems?	Standardization of data elements within the hub provides better entity results and a consistent view of the data.
Do the different systems provide different levels of accuracy and reliability?	A customer-facing system might be more accurate and reliable than third-party enrichment data that is only occasionally refreshed. In this case, survivorship rules can be implemented to capture the best data from the most trustworthy system.
How big will the final data set be?	Addressing operational issues, such as database size and processing time, are critical for success.

Question	Discussion
How frequently does the data change in each system?	Especially in the case of Extraction, Transformation, and Loading (ETL) scenarios, where the hub is updated at regular intervals, it is important to know the latency associated with these updates. Is it acceptable for the hub data to lag behind the source system by a week? Or must the hub data always be up-to-date, thereby requiring a real-time interaction between the systems? Knowing the answers to these types of questions ensures that you have realistic expectations from the SAS MDM system.

What Are Entities?

A SAS MDM entity is a survivor record that is constructed from contributing data from the source systems in the enterprise. This section explains more about entities within the hub.

An entity has a type, where the types represent the nouns of your business. For example, an entity can be an individual, organization, product, or service.

SAS MDM does not pre-define which data your entities require. Instead, SAS MDM accepts metadata about your entities and modifies the hub accordingly.

This metadata includes the following:

attributes

data elements used by entities. Attributes are analogous to columns in a database table. Examples include addresses, e-mail addresses, and dimensions. Attributes have metadata that describes them, such as data type, length, and whether the attribute is a required value for a particular entity.

cluster conditions

rules used to determine which records contribute to the survivor record.

The metadata is collectively known as an entity type, which describes the information that is used by all the instance data for that type. Parent entity type references enable child entities to inherit definitions from their ancestors. For example, the top-level SAS MDM entity type (named ENTITY) defines a set of attributes that are used to maintain instance data in the hub. Because every other entity type derives (either directly, or through other entity types) from the ENTITY entity type, modeling these attributes in the derived types is unnecessary. Attributes, cluster conditions, and relationship types are all inherited by derived entity types.

The metadata is used in several ways in the operation of the hub:

- Entity type information (including the attributes) is used to construct tables in the hub, where each attribute is expressed as a column. These tables are regular database tables and can be used by reporting and other line-of-business applications.
- Cluster conditions are used to generate batch jobs and data services that drive the operation of the hub. You can use these services to add, query, and retire information in the hub.

- Metadata is used at run time to determine how to present the contents of the hub, including the layout of forms and tables.

How Are Entities Related?

Relationships

Entities often maintain relationships among one another. For example, Companies employ People, or Cities are part of a Geographic region. The hub supports relationships between entities by modeling binary relationships between two entity types. A relationship type defines one entity type as the "source" of the relationship, and the other entity type as the "target" of the relationship. Each relationship type can describe conditions that enable SAS MDM to determine whether two entities are related.

Hierarchies

Relationships can also be used to construct hierarchies of entities. Hierarchies provide a tree-structured view of the relationships between entities, based on the types of relationship types that you specify when designing the hierarchy.

How Are Entities Constructed?

Each entity in the hub is represented by a set of records, loaded from source systems, that have undergone the following processes:

cleansing

Data quality, standardization, and other transformations are applied to the incoming records to ensure that the data meets business requirements. The cleansing process removes irregularities, standardizes values so that records can be compared, and computes match codes, which are used in the clustering process. Records that do not meet (or cannot be made to meet) the standards are redirected for further processing.

clustering

Records that represent the same entity are clustered based on criteria specified by the entity type. The criteria include sets of attributes to identify records that contribute to an entity. For example, if two customers have the same name, e-mail, and address, you can create a cluster condition that specifies that the two customers should be considered as part of the same entity.

survival

Records in the cluster are used to construct a single survivor record (also known as the best record or golden record) based on rules that are specified by the entity type.

The following sections provide additional details about each of these processes.

Inside the Cleansing Process

Validation Rules

The cleansing step is a key component of SAS MDM. The hub keeps a clean, standardized version of each core attribute that matches the semantics of the attribute type. The possibilities for validation rules are unlimited. A rule can accomplish simple tasks such as determining whether a field is null, or more complex tasks such as cross-field calculations.

The following are examples of logical validation rules for SAS MDM:

full name

modifies the name field to use proper case.

address

performs address verification, including augmenting the record with missing fields that can be provided by third-party services or databases.

telephone

removes records with no area code.

Handling Invalid Data

Most organizations have business rules that define the suitability of data for various business purposes. For example, an e-mail address might be a required data element in a customer record on which other business processes like marketing or customer billing rely. If the e-mail address is not present in a customer record, this omission might have ripple effects across the business. Using data remediation, invalid data can be captured before it goes into SAS MDM or after it has been loaded into SAS MDM and combined with other data. Business rules can be built into SAS MDM to identify incorrect or invalid data and isolate it from other data until it has been reviewed and corrected. Once it has been isolated, data administrators can review and correct the data issues and return the data to the load process. If additional processes are needed for correction and approval, workflows can be used to route the data through different groups before it is finally committed to the SAS MDM database.

Enabling Data Quality

SAS MDM can perform data cleansing and standardization by using an extensive suite of data profiling and data quality nodes, including the Quality Knowledge Base (QKB). Standardized values and other transformed attributes can be stored with the original data in the hub. For more information about the available standardization schemes, see the DataFlux Quality Knowledge Base online Help. For the QKB to function, the record must be associated with a locale that guides interpretation of the data. This is provided by the `mdm_language_id` attribute, which is part of every entity type. The ID is used to look up the QKB locale based on its value.

In addition to standardizing values, you can use the QKB to compute match codes that assist in clustering similar records into a survivor record. Match codes must be modeled as attributes and become part of the entity record as part of data ingestion processing. For more information about match codes, see [“Inside the Clustering Process” on page 14](#).

Note: Remember that if you are creating standardized values or match codes, they must be modeled as attributes in the metadata as well.

Standardized attributes and match codes are collectively referred to as flux attributes. A flux attribute is always associated with another attribute.

To make it easy to reuse data cleansing and data quality processing, the standardized and match code computations must occur in the `mdm_std_<entity_type>.ddf` job. This job is referenced by the parts of SAS MDM that need to use a consistent view of standardized attributes. For example, both data loading and search jobs can use standardized values and match codes.

To process data in different languages or from different countries at the same time, you must load all the required locales into the QKB both in your DataFlux Data Management Studio design environment and on the DataFlux Data Management Server. Then, you must modify the standardization jobs that clean data and generate match codes so that the appropriate algorithms are used.

To support additional locales:

- You must have a valid and supported QKB license for each locale that you want to use.
- You must update the embedded `mdm_std_<entity_type>.ddf` job with the processing for the additional locales.
- You must update your language definitions to include the new locale, which is used to verify the data when adding information to the hub.

Note: You need individual license codes in your DataFlux Data Management Studio and DataFlux Data Management Server license files to enable you to use more than one locale in your SAS MDM system. By default, you are licensed to use one locale. In most cases, the licensing is for US English, but this can vary according to your needs.

Inside the Clustering Process

Unique and Related Entities

Clustering helps identify unique entities. First, you must determine which fields can be used to identify related records. These are usually combinations of attributes, which can include standardized values or match codes that were computed in the cleansing step. For example, to determine unique entities in your database, you might set up the following rules.

Table 2.2 Clustering Rules

Fields	Result
If all of the fields within a rule match: FULL NAME (Match Code) ADDRESS LINE 1 (Match Code) CITY NAME (Match Code) POSTAL CODE (Match Code) or FULL NAME (Match Code) CUSTOMER_ID (Match Code)	A cluster that contains several records representing the same individual has been identified. You can use similar logic to find match clusters for organizations, households, and others.

Successful identity management uses as many fields as necessary to limit the potential for false matches. For example, if you use only full name fields for match criteria, the result can be hundreds of instances of John Smith, most of which are not duplicates because they live at different addresses. The inclusion of address information in the matching criteria significantly improves the matching and linking results in this case. The entity type should specify as many cluster conditions as needed to define the identity criteria.

Note: Be aware of the following:

- We recommend building indexes for attributes that are used in the clustering process. For large hubs, the use of an index can speed up the processing significantly.
- Different database types can handle empty strings differently. For example, Oracle converts empty strings into NULLs, but SQL Server preserves the empty strings. This can cause anomalous results, especially when these values are used in cluster conditions. We recommend that you test your data carefully, and ensure that null fields and empty strings are being handled as you expect in your jobs.
- The more match conditions that you add, the larger the impact on overall processing time. Use functionality available to you in DataFlux Data Management Studio to evaluate the suitability of your clustering strategy relative to the cost-benefit ratio of performance and final matching results.

When new records are brought into the hub, any of the following conditions can occur:

Table 2.3 Condition Results

Condition	Result
The record is unique and does not match any records in the hub.	A new cluster is created; a survivor record is created for the cluster.
The record matches records in exactly one other cluster.	The record is added to the existing cluster, the cluster's previous survivor record is retired, and a new survivor record is constructed from all the active contributor records.

Condition	Result
The record matches records in more than one cluster.	The hub looks at all the clusters that are matched to determine whether any of the matches present are records that have been manually moved. The presence of moved records indicates that the natural clustering process either overmatched or undermatched the data, and should not be used.

This processing can be changed by modifying the jobs that assign records to clusters (mrm_add_<entity_type>.ddf and add_update<entity_type>.djf). By default, SAS MDM proceeds as follows:

- If "natural" clusters do not contain moved records, the new record is clustered with these records. If more than one natural cluster matches, the clusters are collapsed into the cluster that has the lowest value of mdm_entity_cluster_id, usually the oldest one.
- If all the clusters contain moved records, the new record is placed in the cluster that has the lowest value of mdm_entity_cluster_id, usually the oldest one.
- In either case, existing survivor records are retired, and a new survivor record is generated.

In all cases, when an incoming record is already represented in the hub (based on its mdm_src_sys_id and mdm_src_sys_rec_id fields), the previous contributor is retired. This prevents data that has been replaced in the source system from being used in clustering and survivorship decisions.

If a bulk update includes two or more records with the same values for mdm_src_sys_id and mdm_src_sys_rec_id, the hub is unable to determine which record is the most recent. In this case, all the records are applied to the hub. If your source data enables you to prefer one record over another one, consider filtering out the other records before providing the data to the bulk jobs.

How Are Entities Identified?

Entities are constructed from contributing records that are selected based on cluster conditions for the entity type. The cluster conditions encode matching rules, which are used to bring records with the same sets of values together. You can use any attribute within the cluster condition. However, if you choose the raw values from the incoming records, you might miss some matches. For example, if two records use different cases for a customer name but are otherwise identical, the matching rules places these records in different clusters.

By leveraging the [cleansing process](#), you can improve the match results by the following methods:

- standardizing values with the QKB, enabling case and common derivations, such as 'Robert,' 'Bob,' and 'Rob,' to be normalized
- using match codes to allow fuzziness to be built into the process

Using Match Codes

Overview of Match Codes

Match code generation is probably the most critical aspect of using SAS MDM. Match codes ensure that data that shares fundamental similarities come together into match clusters when match codes are compared. When used in groups with conditional logic, match codes provide a powerful way to match and link records that do not contain obvious similarities. Match codes are used where a degree of fuzziness is required to find all potential matches or relationships.

For example, to identify all the records in a data set where Robert Brauer exists, you might simply scan the data set for exact matches. However, that approach can miss Bob Brauer and Mr. Robert Brauer, who might be the same person. Using match codes, SAS MDM can identify these entities as potential duplicates, and process business rules to determine the nature of their relationship.

The following three name strings are considered a match:

Table 2.4 Match Codes

Input Data	Match Code
Bob Brauer	MYLY\$\$M@M\$\$\$\$\$
Mr. Robert Brauer	MYLY\$\$M@M\$\$\$\$\$
Robert Brauer	MYLY\$\$M@M\$\$\$\$\$

Typically, match processing is applied to several fields simultaneously. Records are considered to match if data strings from each field produce the same match codes. For example, a cluster condition that parses the **Name** and **Organization** fields from a contact table might return the following results:

Table 2.5 Multiple Field Match Codes

Input Data Name	Organization	Match Code (Name)	Match Code (Organization)
Bob Brauer	DataFlux	MYLY\$\$M@M\$\$\$\$\$ \$	8~GWX\$\$\$\$\$\$\$\$\$
Mr. Robert Brauer	DataFlux Corp	MYLY\$\$M@M\$\$\$\$\$ \$	8~GWX\$\$\$\$\$\$\$\$\$
Robert Brauer	General Motors	MYLY\$\$M@M\$\$\$\$\$ \$	B3Y4G~\$\$\$\$\$\$\$\$\$

The first and second records match, because both have the same match codes for Name and Organization. The third record does not match either of the other records, because the match code generated for the **Organization** field differs from those of the other records.

Note: The length of the match code data fields for most QKB locales is set at 40 characters. Some QKB locales (for example, French) require a longer match code data field. The match code data field lengths might need to be modified based on the locale used to process the data.

In DataFlux Data Management Studio, cleanse the data before matching for standardization or display purposes, because, in addition to generating the match codes from the data, you also want to store the data in a clean and valid format.

Match Definitions

The DataFlux Data Management Studio **Match Codes** node provides many match definitions for various data types and performs specific matching techniques and algorithms on each of those types. For example, the logic used to match name fields is very different from that used to match address fields. If a field is to be included in a match job for a given table, you must associate one of the available field types with that field. The accuracy of a match can be adversely affected by not choosing the field type that most closely resembles the data in the field. For example, the phrases William Street and Bill Street are considered duplicates using the Name match definition, but they are not considered duplicates using the Address match definition.

Match Sensitivity

Match sensitivity defines how closely the DataFlux Data Management Studio **Match Codes** node compares data to determine whether duplicate records exist. Adjusting the sensitivity provides the flexibility needed to accurately identify a wide array of duplicate and near-duplicate records. You must apply one of the available match sensitivities to each field that is used to identify duplicate records. The sensitivity and match definition combine to make up the match job for the selected field.

Match sensitivity uses DataFlux proprietary data quality algorithms to create match codes that are clustered together to identify the duplicate and near-duplicate records contained in a database. You can apply a sensitivity level from 50 to 95. Adjusting the sensitivity level increases or decreases the accuracy and precision of the generated match codes. The default sensitivity level is 85.

If the Exact option is specified in the **Match Definition** field for a given record, a character-by-character match is performed on the specified field. You can select the number of characters on which to match. These numbers are available in the **Sensitivity** field. Exact sensitivity does not use any of the DataFlux data quality algorithms. It clusters data in the same manner as in an SQL ORDER BY statement.

Inside the Survivorship Process

After the clustering process is complete, SAS MDM will have identified a set of records that are logically part of the same entity. You can use SAS MDM to add survivorship rules that use these records to create a survivor record.

For example, when establishing the identity of unique entities, you might find two similar addresses listed for a particular individual. The business logic has identified these two records as matches, but SAS MDM must determine which address should be used as the standard.

Your rules must process the related clusters of information and pull together the correct pieces to create an accurate whole. As in the earlier example of how to determine unique entities in the database, a rule might state that when you have near-duplicate address information, you should use the address that comes from the Customer Relationship

Management (CRM) system because the addresses are updated more regularly than in the other source systems. SAS MDM can translate this rule into usable business logic to correctly identify that individual. All of the logic used for identity management is completely customizable within SAS MDM.

Chapter 3

Deploying SAS MDM

Adding Metadata	21
Overview of Metadata	21
Entity Type Definitions	22
Attribute Definitions	23
Cluster Conditions	26
Relationship Types	27
Sample Metadata	27
Generating Jobs	28
Loading the MDM Hub	30
Overview of Loading the Hub	30
Managing Data Issues through Data Remediation	31
Managing Updates	33
Recovering from Load Problems	33
Trimming Historical Data	34
Loading Data	34
Migrating from a Master Data Management Foundation Hub	35
Overview of Migration	35
Migrating Metadata	35
Importing Source Systems and Languages	35
Importing Master Data Management Foundation Metadata	36
Migrating Instance Data	36
Reporting on the Hub	36
Enabling the Master Data Dashboard Report	36
Auditing and Reporting	36

Adding Metadata

Overview of Metadata

To support different types of entities, SAS MDM uses metadata extensively to describe the information that it manages. Every entity and attribute in the system has associated metadata that tells SAS MDM what it is and how it should be handled. One of the first tasks in a SAS MDM deployment is determining the entities and their attributes that are to be managed by SAS MDM. Metadata can be entered into the system either from the interface or by using a batch job to load metadata definitions that are located in text files. In either case, metadata must be loaded before the introduction of instance data. After

metadata has been added to the hub, data administrators can generate jobs through the Master Data Management user interface. We recommend that you start with the generated jobs and adjust them according to your needs.

Entity Type Definitions

Overview

Entity types are managed in the SAS MDM hub. Each entity type has attributes that must be added to the system to manage instances of that type. Entity type metadata is stored in the MDM_ENTITY_TYPES table. The required attributes are described in “Fields on the Entity Properties Tab” on page 84.

Inherited Metadata

An entity type can inherit metadata from another entity type by specifying the parent_id of the other entity type. When metadata is inherited, it is available for use within the entity type.

The ENTITY entity type defines the following required attributes. These attributes are inherited by every other entity type in the hub.

Table 3.1 Required Attributes

Attribute	Description
mdm_entity_type_id	Foreign key reference to the MDM_ENTITY_TYPES table, which identifies the type of the entity.
mdm_language_id	Foreign key reference to the MDM_LANGUAGES table, which identifies the locale to be used when processing the record with the QKB.
mdm_entity_id	The primary key for the MDM_ENTITY table. Each record in the hub has a unique value.
mdm_entity_cluster_id	The cluster that includes this record. Each record in the hub is part of exactly one cluster at a time.
mdm_src_sys_id	Foreign key reference to the MDM_SRC_SYS table, which identifies the source system for the record. SAS MDM defines two source systems: one for the Master Data Management user interface (for changes from the user interface), and another for the hub itself, which always consists solely of survivor records.
mdm_src_sys_rec_id	An opaque identifier for the originating source system. By using the source system ID along with the source system record ID, the hub can trace records back to the original source data.
mdm_valid_from_dttm	The earliest date and time at which the data in the record should be considered valid.

Attribute	Description
mdm_valid_to_dttm	The latest date and time at which the data in the record should be considered valid. Either leave mdm_valid_to_dttm blank (in database terms, set this value to null) or set it to a date where the data is no longer used. Clustering and nomination of survivor records must use information that has not expired. For example, clustering uses only records where the following condition exists: mdm_valid_from_dttm <= current time and (mdm_valid_to_dttm is null or current time < mdm_valid_to_dttm). Entities are retired by changing the mdm_valid_to_dttm attribute to the current time.

Abstract Entity Types

An entity type can be designated as abstract, which prevents instances of that entity type from being created. Abstract entity types can be useful for modeling purposes. For example, an attribute can be modeled once and inherited by other concrete entity types.

Attribute Definitions

Overview

Attributes define the data elements that collectively constitute the information represented by an entity type. As part of the data profiling phase, the necessary attributes for each entity type must be identified, along with standardization rules and any other necessary data quality rules. Attribute metadata is stored in the MDM_ATTRS table.

These attributes are then merged with a small set of attributes required for hub operation. These attributes are defined by the ENTITY entity type and are inherited by any derived entity types. By building on this base, you can extend SAS MDM to support any type of entity.

Attributes have a number of properties, such as a name (used for uniqueness constraints within the hub), presentation parameters, such as labels and display controls, and an associated data type, which identifies how the attribute is stored within the hub. These properties are described in [“Fields on the Attribute Properties Tab” on page 85](#)

The information in the attribute definition is used to construct a column of the same name, with the data type, length (if any), and constraints as indicated by the metadata. In order for the entity type to be published successfully, the attribute name must be compatible with any restrictions imposed by the underlying database. Reserved words in the underlying database cannot be used as attribute names although they can be used as attribute labels.

Note: The total number of attributes that can be modeled on an entity type is subject to database limits. SAS MDM uses dynamic SQL to support flexible models, so the number of attributes that can be supported is a function of the length of the attribute names expressed as a comma-delimited list. For example, with Oracle the maximum size of this list is 4000 bytes, which allows for approximately 129 attributes if each attribute uses the full 30-byte name.

Shorter names allow a greater number of attributes, whereas encrypted attributes consume approximately five times the space of an unencrypted attribute.

Attribute Data Types

Here are the supported data types for SAS MDM attributes:

Table 3.2 *Supported Data Types*

Data Type	Description
String	Presented in the Master Data Management user interface as a single-line text entry field, with a maximum length of 4000 bytes.
Text	Presented as a multi-line text entry field, with a maximum length of 4000 bytes.
List	Presented as a drop-down selection list. Values are provided by a semicolon-delimited string in the attribute's constraint field. The constraint field is limited to 200 bytes, including the semicolon delimiters. If the attribute is not marked as a required field, a blank choice indicates that there is no value for this attribute.
Date	Presented as a text field with an attached calendar widget. Clicking in the text field enables you to choose the date from the widget. Dates do not include a timestamp component.
Timestamp	Presented as a date, but with a time component. Note that changes to the default time must be entered into the text field directly.
Boolean	Presented as a drop-down selection list with the values Yes and No. If the attribute is not marked as a required field, a blank choice indicates that there is no value for this attribute.
Integer	Presented as a text field, which accepts up to 10 digits.
Real	Presented as a field limited to 31 digits in total length, with up to 5 digits behind the decimal point. In database terms, these are numbers with a precision of 31 and a scale of 5.
URL	Presented as a text field with a link widget to the right. Clicking the link widget opens a new browser window or tab for the URL specified in the text field. URL fields can be up to 4000 bytes in length. Note that no validation of the URL is performed.

Building Indices for Staging Tables

Custom indices can be created on user-defined attribute columns in staging and transpose tables either at the time of loading metadata or afterward. The list of indices is maintained in `mdm_entity_type_indices` table in the hub. At the time of loading metadata, the indices are created by `add_sample_meta.djf` job. The job `add_indices.djf` can be used to add custom indices anytime after the initial metadata creation. This job drops any existing custom indices for entity types in the indices input file and creates new ones as specified in the input file.

Presentation Metadata

You can control the order in which attributes appear in various elements of the Master Data Management user interface by specifying ordinal values in the attributes shown in the following table:

Table 3.3 *Attribute Order*

Attribute	UI Element
mdm_search_display_order	Entity search form.
mdm_table_display_order	Entity search results tables.
mdm_cluster_display_order	Cluster table on entity editor or Clustering tab.
mdm_form_display_order	Entity editor Properties tab.
mdm_preview_display_order	Preview panels throughout the interface.

The ordinal values are used to designate where in the Master Data Management user interface the element is placed, with lower-numbered items that appear vertically above those with higher numbers. The table views (for example, entity search results tables and entity editor cluster tables) display items from lowest to highest and from left to right.

All display order attributes follow the same rules. Enter either zero (0) or a negative value to hide the attribute in the Master Data Management user interface, or enter a value greater than 0 to show the attribute in the corresponding user interface element. Fields appear in the order specified by the display order values of their attributes.

An attribute does not appear in the search form or search results if access control has been applied to the attribute in either the current entity type or any of its derived types.

Including sufficient padding between the attribute sort order values is recommended. This allows for future attributes to be added, or for attributes to be reorganized.

Attribute Modifiers

The following additional modifiers can be used to alter the processing of the data within your hub.

Table 3.4 *Additional Modifiers*

Modifier	Description
MDM_READONLY	This modifier causes Master Data Management to display values, but does not allow changes to the values. This can be useful to prevent users from attempting to modify the computed standardized values.

Modifier	Description
MDM_REQUIRED	<p>When you model an attribute as a required attribute, SAS MDM uses that information to add a NOT NULL constraint in the database. Attempts to load a record that does not have a value for a required attribute is disallowed by the database.</p> <p><i>Note:</i> If you retire a required attribute, the NOT NULL constraint is still present and can lead to difficulties when adding more data. Because the attribute is retired, it is not presented in the Master Data Management user interface; and because it is required, if it does not have a value, it cannot be added to the hub. SAS MDM cannot drop the NOT NULL constraint automatically, because the attribute might be involved in indices that need to be rebuilt. Contact your database administrator if you need to retire a required attribute, and ensure that the NOT NULL constraint is dropped, and any indices that reference the attribute have been rebuilt.</p>
MDM_ENCRYPT_FLAG	<p>SAS MDM provides the ability to encrypt sensitive data in the hub. The data is encrypted while it is stored in the database, but is transparent to the SAS MDM application. To enable encryption for an attribute, select the Encrypt attribute field in its attribute definition. After an attribute is published, this flag is set to read-only and cannot be modified. For this reason, it is important to decide what attributes to encrypt before an entity type is published and its jobs are generated. The following limitations apply:</p> <ul style="list-style-type: none"> • Database limitations for encryption apply. • Encryption can increase the size of the data to be stored. When modeling encrypted attributes, set the field length approximately five times higher than the normally anticipated field length. • Only string fields can be encrypted. To encrypt integer values (for example, Social Security numbers or credit card numbers), they should be defined as string fields in the input data. • Use of encrypted fields in cluster conditions and relationship type definitions is not supported. • Encrypting flux attributes (attributes holding standardization and match code values) is not supported.

Cluster Conditions

Cluster conditions are sets of attributes that are used to provide an identity to an entity. If records have the same identity, they become part of the same cluster, and are used when constructing the survivor record for the cluster. This enables data from disparate systems to be matched as a way of consolidating information in the hub.

Relationship Types

Relationship types describe a linkage between two entity types. For example, you might want to maintain a linkage between a manager and his or her employees. If the data provided to the hub includes information that enables the manager-employee relationship to be determined automatically (for example, by including the Manager's employee ID in the Employee entity), then these relationships can be determined automatically by the hub and used to explore the resulting graphs of objects within the Master Data Management user interface.

Sample Metadata

SAS MDM provides the following sample entity types, which demonstrate the concepts that are used to construct a hub. You can use these samples for installation verification, or as study guides to understand how the hub works.

Table 3.5 Sample Entity Types

Entity Type	Description	Notes
Part	Represents a product with dimensions.	Part includes a supplier name attribute that supports a one-to-many relationship with the Company entity type.
Party	An abstract entity type used to define common attributes, cluster conditions, and relationship types for Individual and Company.	<ul style="list-style-type: none"> Party defines the name attribute shared by both Individual and Company, which enables search capabilities for entity types derived from it. You can search for Parties as a way of retrieving both Individuals and Companies. Party provides a standardization job (mdm_stnd_party.ddf) that must be either referenced or reimplemented by derived entity types. Tool_party_us_addver requires a license for the USPS address verification module of DataFlux Data Management Platform.
Individual	Represents a person as a type of Party. Individual builds on Party to introduce additional attributes for a shipping address, do-not-call processing, and demographic information.	<ul style="list-style-type: none"> Individual provides a standardization job (mdm_stnd_individual.ddf) that references the Party standardization job. Individual extends the cluster conditions defined by Party. The tool_individual_multi_update.ddf job requires credentials to function. Add the user ID and password to the real-time service node named Realtime Service - Call Add/Update Service.

Entity Type	Description	Notes
Company	Represents an organization as a type of Party. Company builds on Party to introduce fax and website attributes, and adds a relationship type that can be used to derive relationships to Parts.	<ul style="list-style-type: none"> Company provides a standardization job (mdm_stnd_company.ddf) that references the Party standardization job. Company extends the cluster conditions defined by Party.

Generating Jobs

Many of the jobs that you use in SAS MDM have a similar structure. The Master Data Management user interface makes it easy to generate jobs that embody these default behaviors. After creating the metadata for an entity type, you can use the **Generate Jobs** option from the entity type editor toolbar to create and publish jobs to the DataFlux Data Management Server. Selecting this option generates the jobs listed in the following table.

Note: Not all of these jobs are required in all cases. For example, abstract entity types generally support search operations, but do not support any of the other jobs. The unnecessary jobs are shown as dimmed in the **Generate Jobs** dialog box.

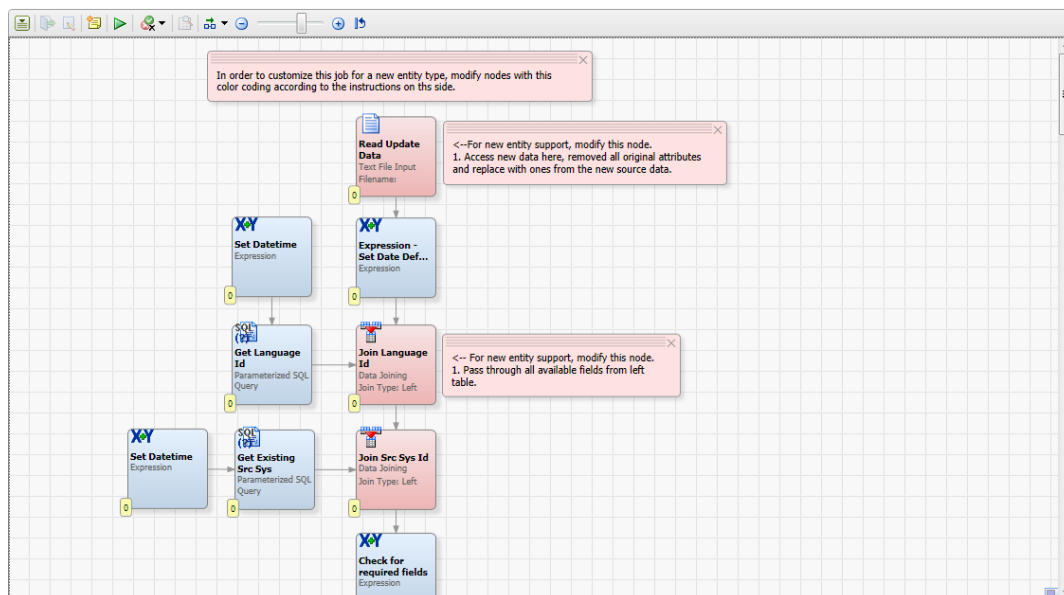
Table 3.6 Jobs

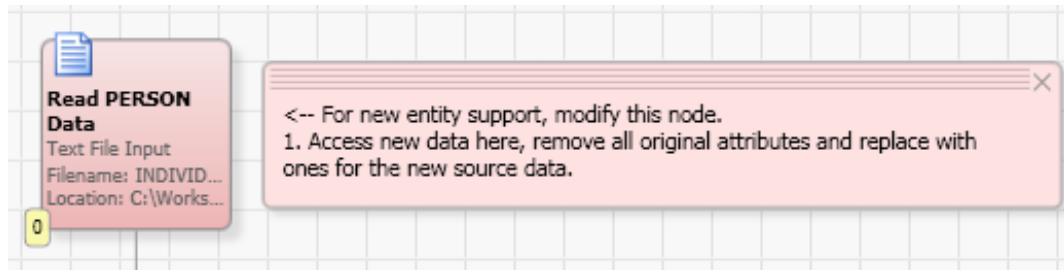
Name	Description
mdm_extract_<entity_type>.ddf	Extracts data from a source system.
mdm_survive_<entity_type>.ddf	Applies survivorship rules.
mdm_cluster_<entity_type>.ddf	Applies clustering rules
mdm_read_remediation_<entity_type>.ddf	Reads records used by SAS Data Remediation and SAS Task Manager <i>Note:</i> Must have access privileges to the record files.
mdm_stnd_<entity_type>.ddf	Computes standardized and match codes for use in searches and creating new records.
mdm_query_survivor_<entity_type>.ddf	Searches the hub for matching records based on specified criteria.
mdm_query_relationship_<entity_type>.ddf	Searches the hub for related records based on specified criteria.
mdm_query_cluster_<entity_type>.ddf	Searches for contributor records for a specified entity.

Name	Description
mrm_add_<entity_type>.ddf	Adds new records to the hub from the Master Data Management user interface.
mrm_edit_<entity_type>.ddf	Supports lifecycle workflows from the Master Data Management user interface.
mrm_move_<entity_type>_records.ddf	Supports moving records between clusters.
mrm_revert_move_<entity_type>.ddf	Reclusters moved records so that they can cluster according to the conditions specified by the entity type.
mrm_write_<entity_type>.ddf	Writes records used by SAS Data Remediation and SAS Task Manager. <i>Note:</i> Must have access privileges to the record files.
add_update_<entity_type>.djf	Bulk-loads data into the hub.
mdm_archive_<entity_type>.djf	Removes retired records from the hub.

You can modify these templates to meet your needs. When modifying them, look for comments that provide information, as shown in the following examples.

Figure 3.1 Sample Job Template





Loading the MDM Hub

Overview of Loading the Hub

SAS MDM provides sample jobs that demonstrate how to load records into the SAS MDM hub using batch interfaces. These jobs are named `add_update_<entity_type>.djf`, and can be generated using the Master Data Management user interface. The sample jobs generally read their data from a file, perform the previously described [clustering](#) and [survivorship](#) steps, and then add the data to the hub.

Each `add_update_<entity_type>.djf` job starts with an Echo node that provides defaults for parameters used by the jobs, such as the following:

- the filename for the data
- database credentials
- parameters controlling the use of bulk-load mechanisms such as the Oracle `sqlldr` or SQL Server `bcp` utilities

These parameters can be updated in the Echo node, or provided on the command line using `dmexec`. For more information about running jobs from the command line, see the *DataFlux Data Management Studio User's Guide*.

The sample jobs use the metadata defined for the entity type to ensure that all the required fields (including the label attribute and the QKB locale) are present. Without these values, the data cannot be written to the database. Records that fail this check are sent to SAS Data Remediation, where they can be reviewed and corrected.

For optimized bulk loading, configure the jobs to use the appropriate database utility by specifying `IS_MAX_LOAD = Y` on the **Inputs** tab for the Echo node. Update the other inputs to specify your credentials, database server, database, and input file.

Note: Running more than one occurrence of an `add_update_<entity_type>.djf` job when `IS_MAX_LOAD = Y` is not supported. The DBMS-provided bulk-load utilities make changes directly to the underlying database storage that might conflict when running multiple occurrences of the job. If `IS_MAX_LOAD = N`, multiple occurrences of the job can be run simultaneously.

If you are using a SQL Server database, you need to generate the `.bcp` files used to load the staging tables and place them in `[SAS MDM Install Dir]\Solutions\SASMDM\samples\scripts`:

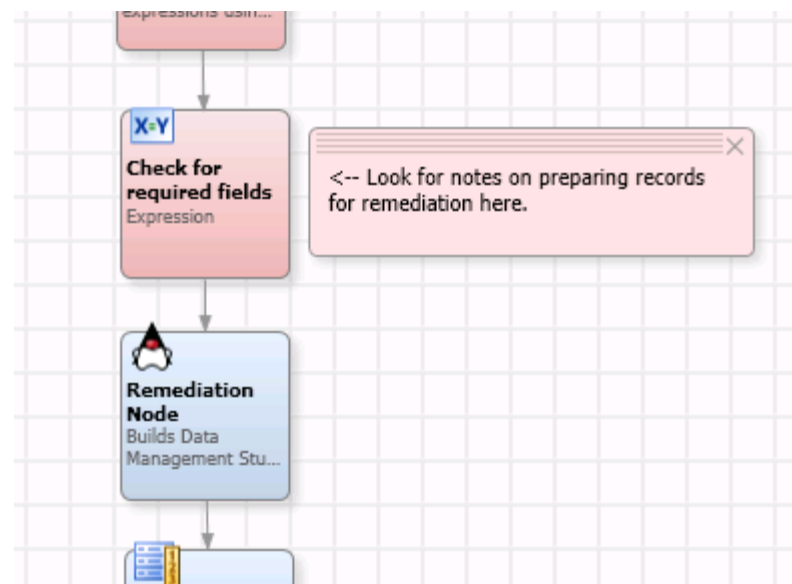
- `mdm_company_st.bcp`
- `mdm_individual_st.bcp`
- `mdm_part_st.bcp`

For more information about generating .bcp files and adjusting the generated files to match the input data files as prepared by the job, see <http://msdn.microsoft.com/en-us/library/ms191479.aspx>. The sample jobs generate their files by placing the field names in alphabetical order, which does not necessarily correspond to the table structure.

Managing Data Issues through Data Remediation

Jobs that are generated for each entity type have two default locations where interaction with the data remediation environment is expected: the `mdm_extract_[entity type]` job and the `mdm_survive_[entity type]` job. Both have a set of nodes that work together to flag data issues and send them to data remediation. These nodes are labeled Check for required fields and Remediation Node.

Figure 3.2 Nodes



The expression code in the Check for required fields node flags records for data remediation based on user-defined business rules. Some rules are in place by default to capture data rows that are missing required fields, but you can add others as needed.

The flagged rows are passed to Remediation Node, which invokes a small Java application that moves them to a text file and sends metadata about the rows to data remediation using REST-based service calls. The information that you supply in the Remediation Node determines how the data issues appear in data remediation. The following two tables describe the parameters. [Table 3.7](#) lists parameters that you should not modify. [Table 3.8](#) parameters that you might need to modify.

Note: Some value are represented by macro variables, and you must not change them at this location.

Table 3.7 Non-Modifiable Parameters

Name	Value or Example	Notes
Application	SAS MDM	Corresponds to application name defined in data remediation.

Name	Value or Example	Notes
Description	INDIVIDUAL Bulk Extract Staging Id % %MDM_STAGING_ID%%	Corresponds to package note.
File Name	%%MDM_DATA% %ExtractINDIVIDUALRemediation_% %MDM_STAGING_ID%%_%%FILEDATE% %.txt [C:/dataflux/mdm/Solutions/SASMDM/data/ ExtractINDIVIDUALRemediation__.txt]	Name and location of text file that contains remediation issues.
Issue Description	ISSUE	Corresponds to issue type in data remediation.
Issue Field	ISSUE_FIELD	Field where an issue has been identified.
Label Attribute	NAME	Label of field for issue name, also used to supply the value for issue name.
Remediation URL	%%MDM_REMEDIATION_URL%% [http:// localhost:80/SASDataRemediation/rest/groups]	Location of Remediation REST API.
Subject Area	INDIVIDUAL	Corresponds to subject area defined in data remediation.
Remediation User	%%REMEDATION_USER%%	Identity used for authentication for REST API.
Remediation Password	%%REMEDATION_PASSWORD%%	Password of identity used to communicate with REST API. <i>Note:</i> This must be a base64-encoded string
Workflow Name Field	WORKFLOW_NAME	Passes a workflow name to data remediation to initiate a workflow. The name must be a valid and active workflow available in the SAS Workflow environment.

Table 3.8 Modifiable Parameters

Name	Value or Example	Notes
Name	BULK EXTRACT	Name of remediation package.
Owner	%%USERNAME%%	Pre-defined assignee for each issue.
Importance	MEDIUM	Pre-defined priority for each issue.
User-Defined Field 1	ADDRESS1	Data value from this field shows in the first user-defined field in data remediation. This label is also tied to the package.
User-Defined Field 2	CITY	Data value from this field shows in the second user-defined field in data remediation.

Name	Value or Example	Notes
User-Defined Field 3	STATE_PROVINCE	Data value from this field shows in the third user-defined field in data remediation.

Managing Updates

It is possible to reprocess every row that might have changed in a weekly update using the same method, but this is not the most efficient way to handle changes. If each source system records changes to its own records and identifies new or deleted records, then only those changes are selected and processed as updates against the hub. The change files could be created using any Extraction, Transformation, and Loading (ETL) process, assuming it can access the files themselves. However, even if changed, added, or deleted records cannot be found prior to direct hub interaction, then it is still possible to pre-process the data file and get better performance from the main hub update process.

Together, data clustering and survivor record creation, or survivorship, are the slowest processes in SAS MDM. At initial batch load time, these processes can be designed to run quickly because much of it can be done in memory. However, when the customer hub is already loaded, the process must first query the database to find similar rows, after which clustering and survivor record creation can take place. Rather than recluster the entire hub or even recluster for every possible new or updated customer record, you can identify the types of changes in data updates bound for the hub. You can route only a subset of the batch update through the more intensive clustering processing, in this case the new rows and cluster-changing rows, after determining the following:

- which rows in the update set are new
- which rows could create changes to clusters
- which rows are not new or will not create changes to clusters

Other rows can be dropped or sent through a much more efficient process that can update attributes that do not affect any clustering or survivor records. The main requirement for this type of procedure is to use a tool with data access and data comparison capabilities and to have constant unique identifiers in source systems. The source system record ID and source system code are saved with every entity in the hub. Using these keys, you can relate what is currently in the hub to potential updates to the hub and categorize the types of updates before initiating the load process.

Use the following strategy for the update process:

1. Run batch processes on source systems that pull out potentially changed data.
2. Parse the potential updates and categorize the data into groups bound for the hub.
3. Run the new and modified data through a process to update the hub.

Recovering from Load Problems

The sample jobs write incoming data into a staging table (one per entity type) before updating the hub. Only after the data has been successfully staged is it applied to the hub. If a problem occurs during the load operation, the transaction is rolled back. The records remain in the staging table, and the hub is unaffected.

The MDM_PROCESS_CONTROL table contains information about all the load jobs and their status, and can be used to determine the staging ID. Failing loads are identified

by the column in the MDM_PROCESS_CONTROL table, which has a unique value for each load process. The values can be used to recover from the failure by either deleting those records from the staging table or restarting the go-live process.

Trimming Historical Data

The hub continues to expand with each insert and update. Retired and not-yet-active records are maintained in the hub. This information can be used for historical tracking purposes. No data is deleted during normal SAS MDM operation, although if the information is no longer needed, it can be moved to offline storage to free up resources.

The exact requirements for removing data from the hub can vary greatly. SAS MDM provides a sample job that is suitable for customization. Within the job are notes about how to modify the job for your particular entity type. The `mdm_archive_<entity_type>.djf` job, in its original state, is used to generate archive files for the entity type. It cannot be accessed through the Master Data Management; you must run it manually. The `mdm_archive_<entity_type>.djf` job is generated using the Master Data Management interface. This job does the following:

- generates the archive files for expired audit records, expired error records, affected hierarchy records, expired cluster members, and expired transpose records.
- deletes any expired audit records, expired error records, affected hierarchy records, expired cluster members, and expired transpose records.
- cleans empty clusters.
- cleans up entities from the database.

Loading Data

SAS MDM provides sample metadata and instance data to load into your hub. By default, the jobs to load the instance metadata are in `[SAS MDM Install Dir]\batch_jobs\sasmdm`. Installing the sample metadata and instance data is optional.

Table 3.9 Metadata Jobs

Job	Description
<code>add_sample_meta.djf</code>	Loads the required metadata for the samples.
<code>add_ui_language_sample_<locale>.djf</code>	Loads translatable metadata for the samples.

The jobs to load the instance data are generated on the DataFlux Data Management Server.

Import the jobs from the DataFlux Data Management Server into the following folders on your DataFlux Data Management Studio repository, mirroring their locations on the DataFlux Data Management Server:

- `batch_jobs\SASMDM`
- `data_services\SASMDM`

Execute the following jobs to load the sample data. You should examine and, if necessary, modify the Echo node at the top of the job to identify the data set that you want to load.

Table 3.10 *Instance Data*

Job	Description
add_update_individual.djf	Loads sample INDIVIDUAL data.
add_update_company.djf	Loads sample COMPANY data.
add_update_part.djf	Loads sample PART data.

You can modify the add_update jobs to specify a file that you want to load or provide the value as a command line option to dmpexec: `-o EXTRACT_PARAMETER=`.

Migrating from a Master Data Management Foundation Hub

Overview of Migration

If you have used DataFlux Data Management Studio to create a hub, you can migrate the metadata and instance data from that hub into SAS MDM. For more information about the Master Data Management Foundation feature, see the *Using Master Data Management Foundation* section of the *DataFlux Data Management Studio User's Guide*.

Migrating Metadata

The metadata maintained by Master Data Management Foundation is a subset of the metadata maintained by SAS MDM. After migrating the metadata, you should review the entity type definition and augment the metadata with presentation metadata—for example, configuring which attributes appear in the Master Data Management search and editor interfaces. To migrate data from Master Data Management Foundation, you must configure a data source for the Master Data Management Foundation hub, and you must know the schema name for the hub.

Importing Source Systems and Languages

The source systems that provide data to the Master Data Management Foundation hub, and the language for the hub, might not be known to SAS MDM. You can import the source systems and languages defined in the Master Data Management Foundation hub by using the `migrate_mdf_meta.ddf` job, available in the `sample\templates\meta` directory. You must configure the Echo node at the top of the job, and ensure that the `MDF_DSN` and `MDF_SCHEMA` inputs correspond to your hub. When you run the job, it imports any source systems and languages that are not already known to the hub.

Importing Master Data Management Foundation Metadata

Master Data Management Foundation entity types are configured from XML files with a .met extension. These files include information about an entity type, its attributes, and its cluster conditions, but lack the presentation information that SAS MDM requires to display the instance data within the Master Data Management user interface. After importing a .met file, you should review the entity type definition, and augment the metadata appropriately. Use load_met.djf batch job, available in the templates\meta directory, to load the .met files into the hub.

Note:

- In the Echo node, set the metFile variable to the fully qualified path of your .met file.
- Set the locale variable to the name of the locale to use during the import. Only one locale can be imported from a .met file. It must match the locale originally used to generate the .met file.
- The load_met job is intended for the initial load of metadata into the hub. Objects that already appear in the hub are not reprocessed during job execution. To merge the .met file into an existing hub (potentially including updates to existing objects), use the import facilities provided by SAS MDM. See [“Import Model” on page 78](#) for more information.

Migrating Instance Data

Data can be extracted from a Master Data Management Foundation hub and loaded into a SAS MDM hub. See mdf_import_person.djf for an example.

Reporting on the Hub

Enabling the Master Data Dashboard Report

A batch job named dashboard_stats.djf is provided in the templates\dashboard directory as part of the SAS MDM installation. Running this job updates the metrics that can be used to populate a dashboard with information about the hub. Running the job again removes the previous metrics and regenerates the data. You might want to use a task scheduler to run this job at regular intervals.

See the job for more details about the information that it computes and customize it as necessary for your requirements.

Auditing and Reporting

About Auditing and Reporting

Developers often use DataFlux Data Management Studio to analyze data in the hub by means of data profiling metrics. These metrics make it possible to preview the structure of data in an effort to meet the specific needs of an organization. With DataFlux Data Management Studio, you can easily find and correct defects within the data, maintaining data quality and integrity.

DataFlux Data Management Studio enables you to:

- select and connect to multiple databases through ODBC without concern about the location of the database source.
- create virtual tables using business rules from your data sources to scrutinize and filter your data.
- simultaneously run multiple data metrics operations on different data sources.
- run primary and foreign key analysis, as well as redundant data analysis, to maintain the referential integrity of your data.
- monitor the structure of your data as you change and update your content.

Using SAS MDM Transpose Tables for Reporting

When the metadata for entities is defined and loaded, a SAS MDM database transpose table is created for each entity. For example, if you create an entity named CUSTOMER, a SAS MDM database table named MDM_CUSTOMER_TT is created. This table manages the attribute information for records of the corresponding entity type. The MDM_ENTITY and MDM_ENTITY_CLUSTER_MEMBERS tables manage additional metadata about the record, such as the expiration date and cluster membership. When new CUSTOMER data is added to the hub, a new row is added to each of these tables, and a new row of data is added to the MDM_CUSTOMER_TT table.

After the data is stored in this format, it is available in a view that brings additional fields together to facilitate reporting and querying the hub. For COMPANY, an MDM_COMPANY_TV view is created, joining data from the MDM_COMPANY_TT, MDM_LANGUAGES, and MDM_ENTITY_CLUSTER_MEMBERS tables. The result is a view with all contributor, survivor, active, and inactive records for the COMPANY entity type.

You might typically want only the active survivor records in a result set when reading from this table. For standard reports against this table, your select query should include the following in the WHERE clause:

```
mdm_src_sys_id=1 and mdm_valid_from_dttm <= (current date) and
(mdm_valid_to_dttm is null or mdm_valid_to_dttm > (current date))
```

In the query, **(current date)** should be replaced with the appropriate function for your database, either **sysdate** for Oracle or **getdate()** for SQL Server.

Chapter 4

Controlling Access

About Access Control	39
Overview	39
SAS MDM Capabilities	40
SAS MDM Roles	41
Defining Groups	41
Overview of Defining Groups	41
Interaction with DataFlux Data Management Server Permissions	43
Managing Access	43
Designing Jobs	43

About Access Control

Overview

A centralized repository for use across an enterprise requires security. Organizations need the ability to secure access to data, to restrict access to people and systems with a legitimate business use for the data, and to audit changes to the data. Access control restricts access based on business requirements. By configuring access control, you can control which groups have access to different types of information. You can specify controls for the following:

- capabilities that are available in the user interface. For example, some users might be prevented from modifying records.
- complete restriction of access to information about an entity type. For example, a subset of users might have access to COMPANY data, but be unable to access PART data.
- restriction to attributes within an entity type. For example, a subset of users might have access to COMPANY data, but be unable to access contact information modeled within the COMPANY entity type. Access is either completely allowed or denied. The sole read-only support for attributes is the MDM_READONLY modifier or the specification of capabilities within the SAS Metadata Server that allows access to, but not modification of, data within the hub.

Every hub has a minimal level of access control. Users must be authenticated before using SAS MDM and have a minimal capability that grants access to the SAS MDM

application. Users are registered through the SAS Management Console, where they are assigned roles and group membership.

If you do not want to use the SAS MDM access control capability, no action is required. By default, all access is granted. However, when an explicit grant is made to one of the controlled types of information, grants for that information become mandatory.

SAS MDM Capabilities

SAS MDM defines one capability, View Application, that provides access to the application. Every SAS MDM user must have the View Application capability.

SAS MDM defines three capabilities that govern functions related to data in the hub:

Table 4.1 Capabilities Related to Hub Data

Capability	Function
Manage Entities	Allows users with this capability to create, edit, and retire entities.
Manage Clusters	Allows users with this capability to address overmatch and undermatch issues in the clusters constructed by the hub.
Manage Relationships	Allows users with this capability to create and retire relationships between clusters.

SAS MDM defines four capabilities that govern functions relating to the metadata model and administrative aspects of the hub:

Table 4.2 Capabilities Related to Metadata and Administration

Capability	Function
Manage Entity Types	Allows users to create, modify, and retire entity type definitions within the hub.
Manage Relationship Types	Allows users to create, modify, and retire relationship types within the hub.
Manage Source Systems	Allows users to create, modify, and retire source system definitions that are contributors to the hub.
Manage Languages	Allows users to create, modify, and retire language definitions used for data cleansing within the hub.

In addition, you can require that a user be a member of a specific group or groups to access records for a particular entity type, or for individual attributes of those entity types. By default, all access is granted. However, when an explicit grant is made to one of the controlled types of information, grants for that information become mandatory, as shown in the following list:

- If no access rule is defined, access is granted.
- If an access rule is defined and requires at least one role for the user, access is granted.
- If an access rule is defined and requires no roles for the user, access is denied.

Note: For entity types, the label attribute must be accessible. For example, if the entity type has been explicitly granted, but the label attribute has been restricted, the entity type itself is restricted.

SAS MDM Roles

SAS MDM creates five new roles in SAS Management Console. These roles are tied to distinct capabilities in the data remediation application. Depending on assigned role of the user, the experience in SAS MDM changes. Certain features are available to those assigned one role, but the same features are not available to users in another role. Users can also create new groups and roles with distinct sets of capabilities.

Here are the roles for SAS MDM:

Table 4.3 SAS MDM Roles

Role	Capabilities
MDM: Administration (Default Group: Data Management Administrators)	View: Application Manage: Entities; Clusters; Relationships; Entity Types; Relationship Types; Source Systems; Languages
MDM: Business Authorization (Default Group: Data Management Business Approvers)	View: Application Manage: Entities; Clusters; Relationships
MDM: Business Usage (Default Group: Data Management Power Users)	View: Application Manage: Clusters; Relationships
MDM: Business View (Default Group: Data Management Business Users)	View: Application
MDM: Stewardship (Default Group: Data Management Stewards)	View: Application Manage: Entities; Clusters; Relationships; Entity Types; Relationship Types

Defining Groups

Overview of Defining Groups

Groups are used to represent sets of users or a set of privileges within the SAS Metadata Server. Users can belong to multiple groups, and the set of roles and capabilities for a user is a combination of roles and capabilities from all the groups to which the user belongs.

When creating additional groups for access control, you should consider the needs of the users of the information stored within SAS MDM. The groups should be defined at the level appropriate to provide required access to the information, but no access beyond that. For example, consider these cases:

Table 4.4 *Employee Situations*

Employee	Department	Requires
Alice	Payroll	both employee address information and employee financial information.
Bob	Human Resources	employee address information, but does not require employee financial information.
Charlie	Manufacturing	no employee information.

This situation requires two groups:

Table 4.5 *Required Groups*

Employee	Group	Permissions
Alice	PAYMASTER	Grants access to the employee entity type and the financial attributes.
Bob	HR	Grants access to the employee entity type, but does not have any particular restrictions on attributes. Bob cannot access any financial attributes because he is not a member of the PAYMASTER group.
Charlie	Not assigned to a group	Charlie is not assigned to a group because he does not need to access employee data.

If another employee, David, requires access to financial attributes, a new group that has access to those attributes can be created. Alice and David would both be assigned that role, or a new role could be created just for David.

Note that granting groups access to entity types and attributes is complementary to assigning capabilities to a user. A user must still be assigned the View Application capability to be able to access the user interface. Assigning a user to a group with authorization for access to an entity type does not replace this capability.

You must decide whether it is more appropriate to assign roles to a user or to assign roles to a group, after which you assign users to the group.

Interaction with DataFlux Data Management Server Permissions

DataFlux Data Management Server enables you to configure permissions that either allow or deny access to jobs for users and groups. In some cases, entity-type-specific jobs can be controlled effectively by using permissions. In other cases, only one job provides access to all the entity types. Because of this difference, not all security configurations can be handled through permissions alone. In addition, permissions are insufficient to provide control over individual attributes because they operate at the job level. For these reasons, permissions should be considered a necessary but insufficient component of data security, which is augmented by the SAS MDM access control capability.

For more information about DataFlux Data Management Server permissions, see the *DataFlux Data Management Server Administrator's Guide*.

Managing Access

Access can be granted through batch interfaces. The `add_sample_meta.djf` job is an example of loading authorization information from text files. Access can also be granted interactively using the association tables available from the **Data Model** tab. Batch interfaces are more appropriate if you have to set up a large number of authorizations.

Note: Authorization changes made after a user logs on are not recognized until the user logs off and logs on again.

Designing Jobs

Access control operates in the Business tier of SAS MDM. This requires that extensions or modifications to the jobs provided by SAS MDM take access control into account, as follows:

- Jobs that can authorize access must use a variable named `MDM_USER_ROLES`, which is a comma-delimited string of roles assigned to the current user.
- Jobs that query the database directly must join the appropriate table to the roles table and ensure that the query factors in retirement status for authorization records. They must conform to the grant decisions described previously. For example, to determine the entity types that are accessible to a user with the `MDM_BIZUSER` role, the following query might be used:

```
select et.mdm_entity_type_id,et.mdm_entity_type from mdm_entity_types et where
mdm_entity_type_id in
(
  select etr.mdm_entity_type_id from mdm_entity_type_access etr
    where sysdate between etr.mdm_valid_from_dttm and etr.mdm_valid_to_dttm
        and lower (etr.mdm_group) in
union
select et2.mdm_entity_type_id from mdm_entity_types et2
```

```
where et2.mdm_entity_type_id not in
      (select mdm_entity_type_id from mdm_entity_type_access
       where sysdate between mdm_valid_from_dttm and mdm_valid_to_dttm)
```

The two clauses in the sub-select check the following conditions:

- If an active grant exists for the user based on the roles, the entity type is included.
- If no active grants exist, the entity type is included.

Note: Group names are tested using the **in** predicate because any user can belong to more than one group. Compare group names in a case-insensitive manner.

Part 2

SAS MDM Interface

<i>Chapter 5</i>	
Accessing SAS MDM	<i>47</i>
<i>Chapter 6</i>	
Using Master Data Management	<i>55</i>
<i>Chapter 7</i>	
Using Master Data Management Data Model	<i>77</i>
<i>Chapter 8</i>	
Using Master Data Management Administration	<i>105</i>
<i>Chapter 9</i>	
Using SAS Data Remediation	<i>113</i>
<i>Chapter 10</i>	
Using SAS Task Manager	<i>141</i>

Chapter 5

Accessing SAS MDM

Log On to SAS MDM	47
SAS Data Management Console Home Page	48
Specifying Your Preferences	49
Specifying Global Preferences	49
Specifying SAS MDM SAS Data Management Console Preferences	50
Overview of the SAS MDM Interface	51
Menu Options	51
Tables	52
Fields	53

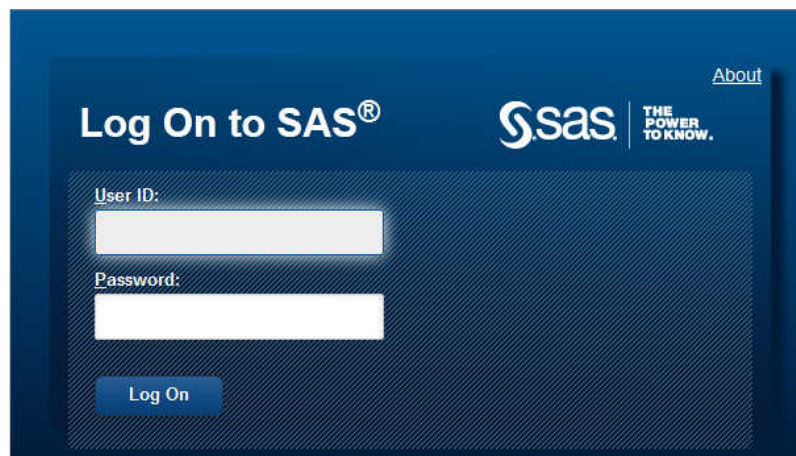
Log On to SAS MDM

SAS MDM is accessed through the standard logon window for SAS applications. Logging on from this window opens SAS Data Management Console from which you can launch SAS MDM.

To log on to SAS Data Management Console:

1. Click the URL that is supplied by your system administrator, or paste it into the address field of your browser to display the SAS logon window:

Figure 5.1 Logon Window for SAS Data Management Console



2. In the **User ID** field, enter your user ID.

3. In the **Password** field, enter the password for your user ID.

Note: Your password is case-sensitive. Your user ID might be case-sensitive, depending on the operating system that is used to host the web application server. If you need assistance, contact your system administrator.

4. Click **Log On** to display SAS Data Management Console.

Note: If you log on to SAS MDM in one browser tab, then log on to SAS MDM or a different SAS web application in another browser tab, the same credentials are used automatically for subsequent authentication attempts.

To log off from SAS Data Management Console:

1. Click **Log Off** in the upper right corner of the user interface:

Figure 5.2 SAS Data Management Console



Note: When you select **Log Off**, you are logged off from all tabs opened by the console.

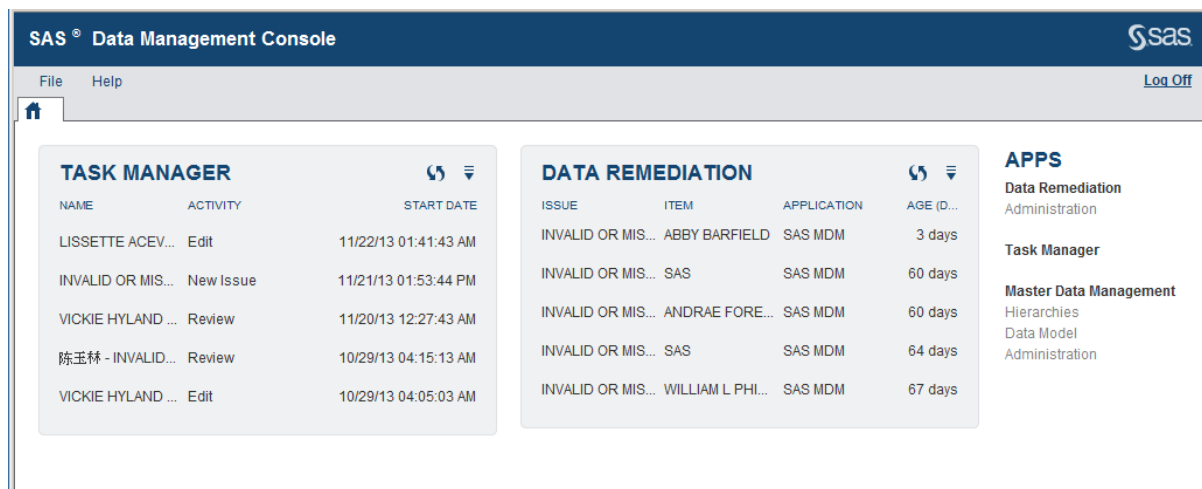
2. If you are prompted about unsaved changes, click **Log Off** to exit without saving or click **Continue** to return to SAS Data Management Console and keep working.

SAS Data Management Console Home Page

The SAS Data Management Console home page enables you to launch the various components of SAS MDM.

Note: The following image is an example. Your specific content might be different.

Figure 5.3 SAS Data Management Console



- 1 The **File** menu on the menu bar enables you to set preferences.
- 2 The **Help** menu enables you to access SAS on the Web.

- 3 The **DATA REMEDIATION** portlet lists, and enables you to open, current remediation tasks.
- 4 The **TASK MANAGER** portlet lists, and enables you to open, current task manager items.
- 5 The **APPS** listings link you to the component applications of SAS MDM.
Note: The list of applications available to you varies according to your assigned role and to which SAS products you have installed.

Here are the SAS MDM applications:

- **Data Remediation**
 - **Administration**
- **Task Manager**
- **Master Data Management**
 - **Hierarchies**
 - **Data Model**
 - **Administration**

Each of these applications is explained in detail in this document.

Specifying Your Preferences

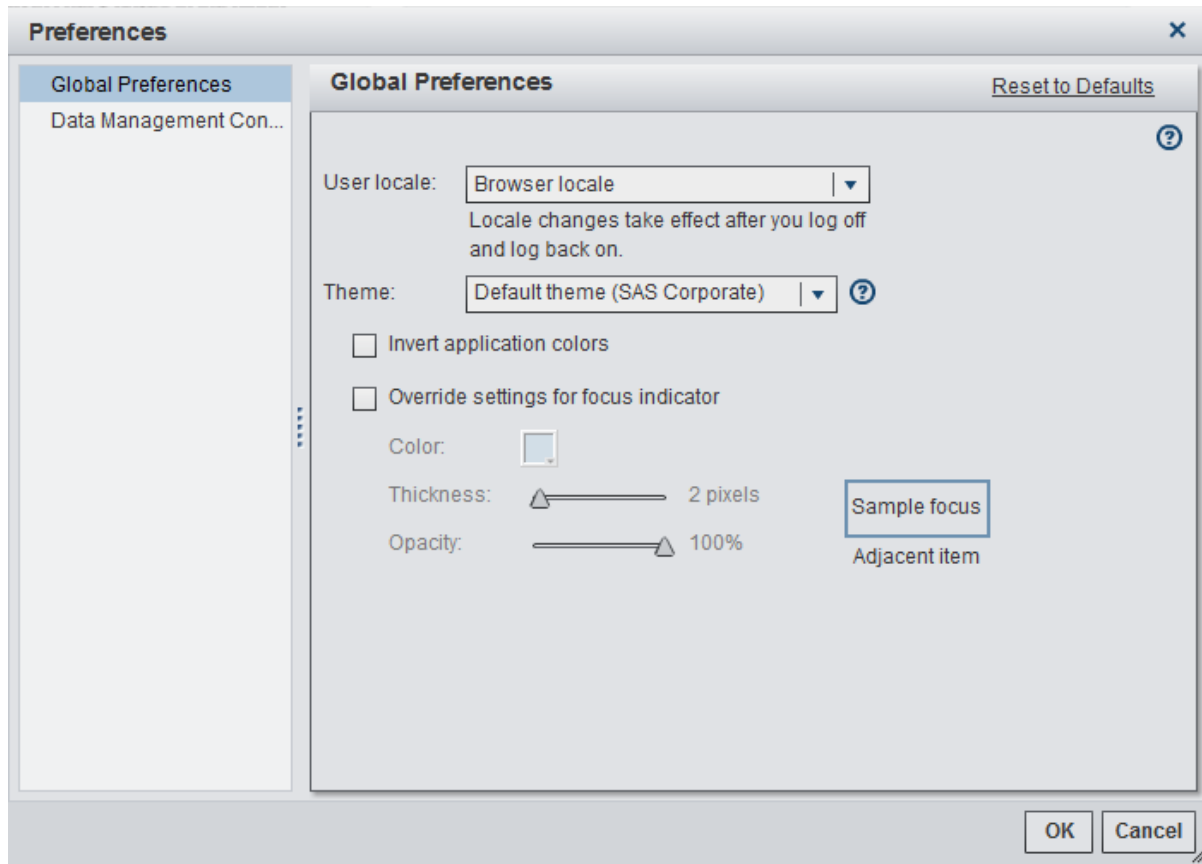
Specifying Global Preferences

You can specify global preferences to apply to all SAS web applications that are displayed with the Adobe Flash player. These preferences are set by each user.

To specify global preferences:

1. Select **File** ⇒ **Preferences** to open the Preferences window.

Figure 5.4 Global Preferences



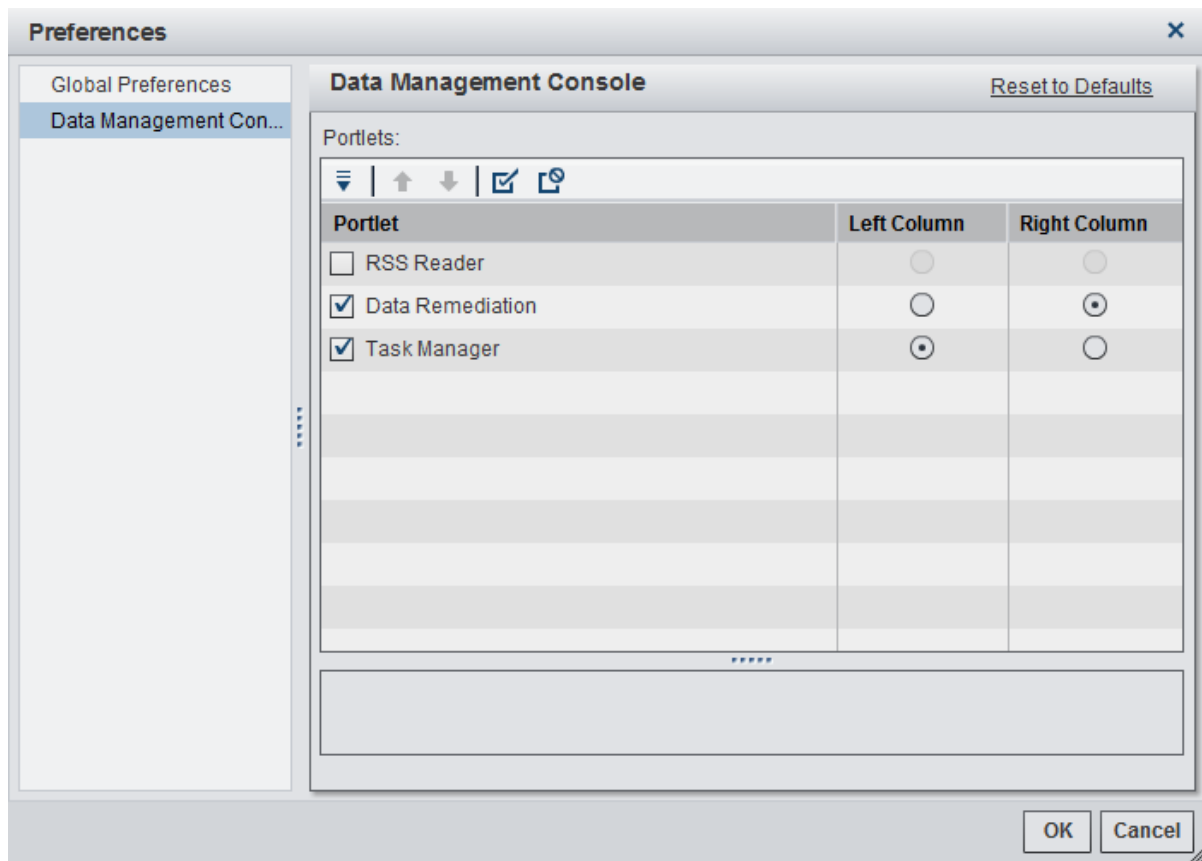
2. Click **Global Preferences** in the left pane.
3. Select a **User locale** to specify your language and geographic region.
 Select a **Theme** to change the color scheme and other visual settings for all of your SAS web applications.
 Select **Invert application colors** to invert all of the colors in your SAS web applications.
 Select **Override settings for focus indicator** to change the color, thickness, and opacity of the focus in your SAS web applications.
4. Click **OK** to apply your changes.
5. Click **Reset to Defaults** to restore all of the Global Preferences options.

Note: If you changed the **User locale**, then you must log off and log back on to SAS MDM for the change to take effect.

Specifying SAS MDM SAS Data Management Console Preferences

To specify SAS Data Management Console preferences:


1. Select **File** ⇒ **Preferences** to open the **Preferences** window.
2. Click **Data Management Console** in the left pane.

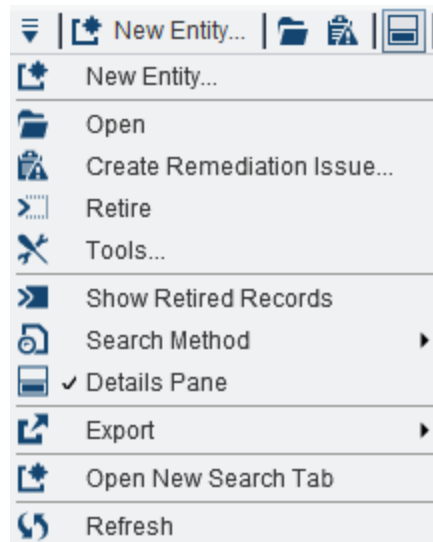






3. Specify whether Portlet panes appear on SAS Data Management Console and their location.
4. Click **OK** to apply your changes.
5. Click **Reset to Defaults** to restore all of the SAS Data Management Console options.

Overview of the SAS MDM Interface

Menu Options

The SAS MDM interface has several common features. The toolbars of the interface contain an **Actions** menu . Clicking  displays the **Actions** menu and enables you to select specific options. Most options available from the **Actions** menu are also available through icons on the same toolbar. Holding the cursor over the icon displays the function of the icon. Clicking the icon performs the function.











You can click icons to expand and collapse elements of the interface. Click  or  to expand an item. Click  or  to collapse an item.



Tables

SAS MDM presents lists of objects in a tabular form. These tables show a subset of the information associated with the objects. All entries in a table are the same type of object. To open an object in an editor, double-click or right-click the row and select **Open** or **Edit** from the pop-up menu.

Entity Types

Entity Types			
  New Entity Type 			
	Type ▲	Display Name	Description
	COMPANY	Company	Company
	ENTITY	ENTITY	Abstract Entity Type
	INDIVIDUAL	Individual	Individual
	PART	Part	Part
	PARTY	Party	Party

To open an object, you can do any of the following:

- Click  from the toolbar.
- Click  and select **Open** or **Edit**.
- Double-click a table row.
- Right-click a table row and select **Open** or **Edit** from the pop-up menu.

To sort a table column in either ascending or descending order, click the column heading. Sorted columns display up or down arrows to indicate the sort direction.

Fields

Required fields on the interface are marked with an asterisk (*). Read-only fields appear dimmed.

Figure 5.5 Fields

▼ Entity Type Properties

Type: * ENTITY

Description: Abstract Entity Type

Parent Type: ▼

Valid From: 05/05/13 11:07:25 AM


Last Modified: 05/05/13 11:07:58 AM

Last Published: 05/05/13 11:07:25 AM

Valid To:

☒ Entity is an abstract entity (cannot be instantiated)

If you enter invalid information in a field, a red border appears around the field. You can hold the mouse over the red border to view information about the invalid field entry.

For date fields, you can either enter the date directly or click  to display a calendar. From the calendar, click a valid date to load it into the field. Click the navigation buttons to change the month and year. Click anywhere outside the calendar to hide it.

Chapter 6

Using Master Data Management

About Master Data Management	55
Overview	55
Managing Entities from the Master Data Management Tab	56
Searching Entities	56
Overview	56
Quick Search	56
Field Search	57
Deep Linking to Master Data	59
Working with Search Results	59
Overview	59
Using Tools	61
Editing Entities	62
The Records Tab	62
Overview	62
The Entity Editor	63
The Recluster Editor	65
The Hierarchies Tab	67
The Relationship Diagram Tab	67
The History Tab	70
Creating New Entities	70
Creating a Remediation Issue	72
Searching Hierarchies	74
Overview	74
Quick Search	74
Field Search	74

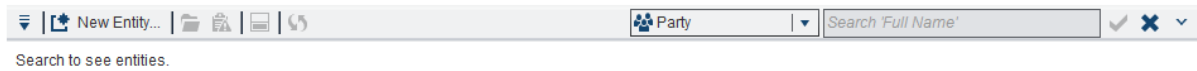
About Master Data Management


Overview

From the **Master Data Management** tab, business users and data administrators can search for, edit, create, and retire master data entities. The role of a user determines the level of access to entities and their attributes.

To access the **Master Data Management** tab, click **Master Data Management** on the SAS Data Management Console page. By default, the tab opens to the search toolbar:

Figure 6.1 Master Data Management Toolbar



Note: In the following descriptions, **Actions** menu items are accessed by clicking  from the toolbar. Most functions available from the **Actions** menu are also available from icons on the same toolbar. Clicking the icon performs the function.

Managing Entities from the Master Data Management Tab

You can perform the following actions from the **Master Data Management** tab:

- Search master data entities using either a quick search or a field search.
- Edit and retire master data entities returned from a search.
- Create a new master data entity for the hub.
- View, create, and retire relationships
- Create new remediation issues to be resolved in SAS Data Remediation.

The following sections describe these actions in detail.

Searching Entities

Overview

In order for a master data entity to be available for a search, the following must be true:

- The entity must be active (non-retired) and have been published for use.
- The entity's sort order must be greater than zero.
- If the entity has been modeled as an abstract entity type, it must contain at least one concrete, derived entity type that meets these criteria.

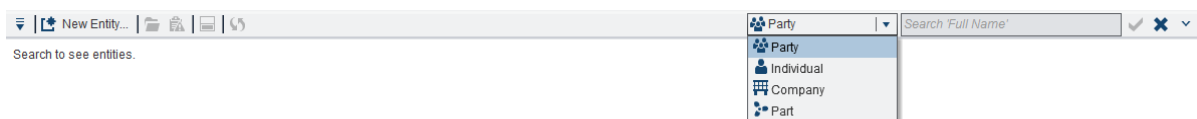
Note: You can perform either a quick search or a field search.

Quick Search

To perform an entity quick search:



1. Select an entity type from the drop-down list on the right side of the search toolbar:

Figure 6.2 Quick Search Toolbar



The default attribute in the search field corresponds to the label attribute for that entity type. For example, when Party is the selected entity type, Full Name is the default attribute for the data search.


Note: You can also change the search method from the default of DataFlux Match to another type of match. See the description of [search methods on page 57](#).

2. Enter the appropriate search information into the search field.
3. Do one of the following:
 - Click  to begin the search.
 - Click  to clear the search data.

Best records are returned in a table. See “[Working with Search Results](#)” on [page 59](#) for information about exploring data returned from the search.

Field Search

To perform an entity field search:

1. Click  on the right side of the toolbar to expand the search pane.

The search pane is constructed from the entity type's metadata. Only attributes that have been configured to appear by an administrator are shown in the pane.

Figure 6.3 Search Pane

Entity Type:

Party

▼ Contact

Full Name:

Primary Phone Number:

Primary Email Address:

► MDM Identifiers

▼ Address


Address Line 1:

City:

State/Province:

Postal Code:

Country:

2. Enter the appropriate search criteria in the fields of the search pane.
3. Click  to display the menu.
4. Click **Search Method** to display the search method options.

By default, the search is based on a DataFlux match. The search methods are:

DataFlux Match

uses match codes during searches and enables fuzzy matching capabilities. If match codes are not available for the fields, SAS MDM attempts to use standardized fields. If match codes or standardized fields are not available, SAS MDM searches for an exact match.

Similar Match



uses database-specific, approximate match functions. For Oracle and SQL Server, the `soundex()` function is used in combination with an exact match. This is because `soundex()` does not support some values, such as numbers and dates.

Case Insensitive

disregards the case of the search term.

Exact Match

searches for the input string exactly as it was entered.

5. Do one of the following:
 - Click  to begin the search.
 - Click  to clear the search data.

See [“Working with Search Results” on page 59](#) for information about exploring data returned from the search.

Deep Linking to Master Data

Deep linking refers to constructing a URL hyperlink that, once initiated, can take a user directly to master data in SAS MDM rather than requiring the user to search for it. For example, you can construct a URL link in an e-mail and send to a data steward. When the data steward clicks the URL link, it opens the SAS MDM application and loads the entity data encoded in the URL.

From within the appropriate environment, create a URL using the following syntax:

```
http://<your_host>/SASDataManagement/  
#type=<entity_type>&entity=<entity_ID>&module=MDM
```

For example, you might create the following:

```
http://localhost:8080/SASDataManagement/  
#type=COMPANY&entity=21682&module=MDM
```

Note: Creating a remediation issue submits a deep link along with the issue, which is accessible through the SAS Data Remediation web service API. For more information, see [“Creating a Remediation Issue” on page 72](#) in SAS MDM.

Working with Search Results

Overview

Successful master data searches return one or more entities (best records).

Figure 6.4 Search Results

Entity Type: 'PARTY'; State/Province: 'nc'

New Entity... [Icons] Party Search 'Full Name' [Check] [X] [Dropdown]

MDM Entity Type	MDM Cluster ID	Full Name	Address Line 1 (Standardized)	City	State/Province	Postal Code (Standardized)	Country
Individual	49	ABBIE BARFI...	788 Saralyn Rd	PITTS...	NC	27312	
Individual	59	ABE BINSTE...	4900 Jay Matthews Ln	SANF...	NC	27332	
Individual	107	ABELINA HA...	1494 Piedmont Dr NW	LENO...	NC	28645	
Individual	135	GABRIEL GA...	8748 Argy Pl	KENLY	NC	27542	
Individual	167	SADHANA W...	24 Howell St	GREE...	NC	27834	
Individual	195	SALLIE KEITH	118 E Counsil St	SALIS...	NC	28144	
Individual	221	SALLY NAIL	166 Race Path Rd	UNIO...	NC	28689	
Individual	315	ADRIA HIRTH	12 Starlifter St	FORT ...	NC	28307	
Individual	333	AERI MARTY	177 Jermane St	RICH...	NC	28574	

Full Name:	ABE BINSTEON	Address Line 1:	4900 JAY MATTHEWS LN	City:	
Name Title:		Address Line 2:		County:	
Name Suffix:		City:	SANFORD	State/Province:	
Name Appendage:		County:	LEE	Postal Code:	
Gender Code:	M	State/Province:	NC	Country:	
Birth Date:	04/26/90	Postal Code:	27332	Primary Phone Number:	(919) 212-2523
Do not Call:	Yes	Country:		Secondary Phone Number:	
Last Contact:	03/16/10 08:20:29 AM	Address Line 1:		Primary Email Address:	
Notes:		Address Line 2:		Secondary Email Address:	

The toolbar at the top of the results table contains an **Actions** menu with the following options:

New Entity

see “[Creating New Entities](#)” on page 70 for a description of this option.

Open

opens the entity selected in the results table and displays cluster details about a new tab.

Create Remediation Issue

see “[Creating a Remediation Issue](#)” on page 72 for a description of this option.

Retire

retires the selected records.

Tools

opens a dialog box that enables you to access available tools.

Search Method

see [Step 4 on page 58](#) for a description of this option.

Details Pane

displays or hides details of the selected record in a separate pane.

Show Retired Records

toggles between exposing or hiding retired records. When exposed, retired records appear dimmed.

Export

exports the results table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files.

When a character cannot be represented in the PDF file, it is replaced with #.

Open New Search Tab


opens a separate tab for a new search.

The contents of the list vary depending the type of entity and what your administrator has made available. For example, if you select **Company Search** while editing a Company entity, the tool opens a Google search for the company name in a separate browser tab.

Note: you must enable pop-ups in your browser to see tool results in a different tab.

Editing Entities

After entities are returned from a search, you can edit them. To edit an entity:

1. Select a record in the result table.
2. Click  and select **Open**.

The cluster of related records opens on a new tab:

Figure 6.6 Search Results

[illegible]

This tab contains four sub-tabs: **Records**, **Hierarchies**, **Relationship Diagrams**, and **History**. The **Records** tab is selected by default.

The Records Tab

Overview

The toolbar at the top of the **Records** tab contains an **Actions** menu with the following options:

Edit

opens the selected record for editing. See [“The Entity Editor” on page 63](#) for a description of this option.

Create Remediation Issue

see [“Creating a Remediation Issue” on page 72](#) for a description of this option.

Retire

retires the selected record.

Tools

opens a dialog box that enables you to access available tools.

Highlight Differences

compares the selected record with the other members of the cluster and highlights the differences among them.

Recluster

moves one or more selected contributor records into a another cluster. See [“The Recluster Editor” on page 65](#) for more information about reclustering.

Revert Move

reclusters selected contributor records that have been previously moved to another cluster. Reverting a move operation causes the record to recluster against the hub data using the defined cluster conditions. The **Revert Move** option is enabled only when a previously moved record is selected.

Note: If the record being reverted is the last contributor in the cluster, the entire cluster might be retired.

See [“The Recluster Editor” on page 65](#) for more information about reclustering.

Show Retired Records

toggles between exposing or hiding retired records. When exposed, retired records appear dimmed.

Details Pane

displays or hides details of the selected record(s) in a separate pane.

Export

exports the records table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files. When a character cannot be represented in the PDF file, it is replaced with #.

Refresh


refreshes the table of records.

The Entity Editor

Overview

The entity editor enables you to make changes to the information for a master data entity.

To open the entity editor:

1. Select a record for editing on the **Records** tab.
2. Click  and select **Edit**. The entity editor opens.

The toolbar at the top of the entity editor contains an **Actions** menu with the following options:

Save

saves changes to the record.

Close Entity Editor

closes the entity editor. The editor can also be closed by clicking the left navigation

breadcrumb, such as

ABE BINSTEON > Orion (59)

Create Remediation Issue

see “Creating a Remediation Issue” on page 72 for a description of this option.

Retire

retires the selected record.

Matrix Edit

toggles between standard Edit mode and matrix Edit mode.

Tools

opens a dialog box that enables you to access available tools.

Reset

resets the record to its initial state without saving any changes.


Standard Edit Mode

The entity editor opens in standard mode unless you have previously selected matrix mode. The following is a sample standard Edit mode screen:

Figure 6.7 Entity Editor

The screenshot shows the Entity Editor interface. At the top, a breadcrumb trail reads 'ABE BINSTEON > Orion (59)'. Below this is a toolbar with icons for Save, Matrix Edit, and a Reset button (a circular arrow). The main content area is divided into sections: 'MDM Identifiers', 'Contact', 'Personal Information', 'Address', 'Shipping Address', and 'Drives'. The 'Contact' section is expanded, showing fields for Full Name (ABE BINSTEON), Name Title (dropdown), Name Suffix, Name Appendage, Do not Call (Yes), Last Contact (March 16, 2010 08:20:29 AM), Notes, Primary Phone Number ((919) 212-2523), Standardized Primary Phone Number ((919) 212 2523), Secondary Phone Number, Standardized Secondary Phone Number, Primary Email Address, and Secondary Email Address. Below the 'Contact' section are sections for 'Personal Information', 'Address', 'Shipping Address', and 'Drives'. At the bottom, there is a table with columns: MDM Entity Type, MDM Cluster ID, Full Name, Do not Call, Last Contact, Address Line 1 (Standardized), and City. The table is currently empty.

MDM Entity Type	MDM Cluster ID	Full Name	Do not Call	Last Contact	Address Line 1 (Standardized)	City

In standard Edit mode, you can change and save the values of individual properties for the selected record. If you change your mind after modifying values, click  to reset the all values in the form.

You can also view, add, or retire relationships using the tables at the bottom of the form. See [“Working with the Relationship Type Editor” on page 99](#) for more information.

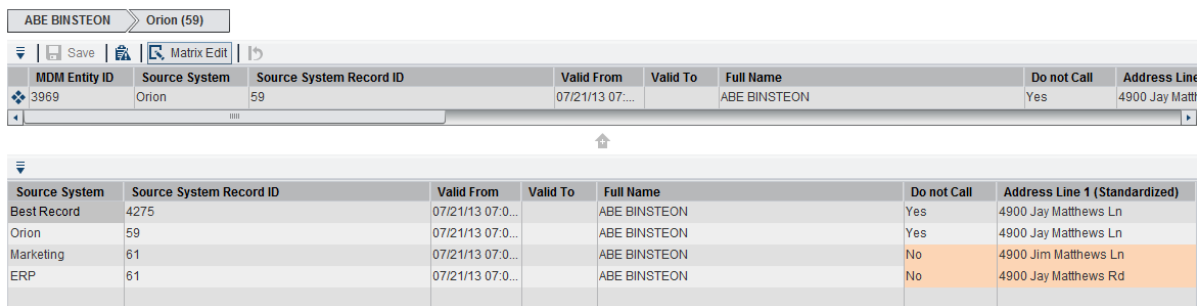
Matrix Edit Mode

Matrix Edit mode enables you to move properties to create a new best record from the contributor records in a cluster.

To enter matrix Edit mode:

1. Click  and select **Matrix Edit** or click  from the toolbar. The tab changes to matrix Edit mode:

Figure 6.8 Matrix Edit






MDM Entity ID	Source System	Source System Record ID	Valid From	Valid To	Full Name	Do not Call	Address Line
3969	Orion	59	07/21/13 07:...		ABE BINSTEON	Yes	4900 Jay Matt

Source System	Source System Record ID	Valid From	Valid To	Full Name	Do not Call	Address Line 1 (Standardized)
Best Record	4275	07/21/13 07:0...		ABE BINSTEON	Yes	4900 Jay Matthews Ln
Orion	59	07/21/13 07:0...		ABE BINSTEON	Yes	4900 Jay Matthews Ln
Marketing	61	07/21/13 07:0...		ABE BINSTEON	No	4900 Jim Matthews Ln
ERP	61	07/21/13 07:0...		ABE BINSTEON	No	4900 Jay Matthews Rd

The tab remains in matrix Edit mode until you toggle it back to standard mode.

Note: Differences between properties in the record in the top table and records in the bottom table are automatically highlighted.

2. Select a property from a record in the bottom table that you want to move to the corresponding location of the record in the top table.
3. Click  to move the selected property. Continue to move properties until the record in the top table contains the correct properties for new best record.
4. If you change your mind after moving a selected property, click  to reset the properties.
5. Click  **Save** to save the new best record when you have finished moving properties.

The Recluster Editor

The recluster editor enables you to move records to new or different clusters. In some cases, the cluster conditions that have been configured for an entity might not work as intended. Records can be overmatched, resulting in clusters that are too large. Records can also be undermatched, resulting in too many clusters. In the case of overmatched records, you might need to move one or more contributor records to a new cluster. In the case of undermatched records, you might need to move one or more contributor records to an existing cluster.

Moving a record indicates that the cluster conditions are not processing the record as intended, so updates to the record are processed differently. Specifically, an update to a moved record (based on its source system ID and source system record ID) never causes clusters to merge together, even if the update matches multiple clusters. If SAS MDM allowed the merge, the move operation would be undone.

Similarly, if an incoming record matches both the original and target clusters, but does not match based on source system ID and source system record ID, the incoming record is placed in the original cluster. SAS MDM has a default preference for clusters without moved records compared to clusters with moved records.

Note:

- Only entities of the same type can be moved to an existing cluster. For example, you cannot move a COMPANY contributor record to an existing PART cluster.
- Survivor records are always generated as part of the operation of the hub and cannot be moved.
- Retired entities (contributor and survivor records) cannot be moved to either a new cluster or an existing cluster.
- Records cannot be moved into retired clusters.
- If you move the last contributor record out of a cluster, the entire cluster is retired.

To open the recluster editor:


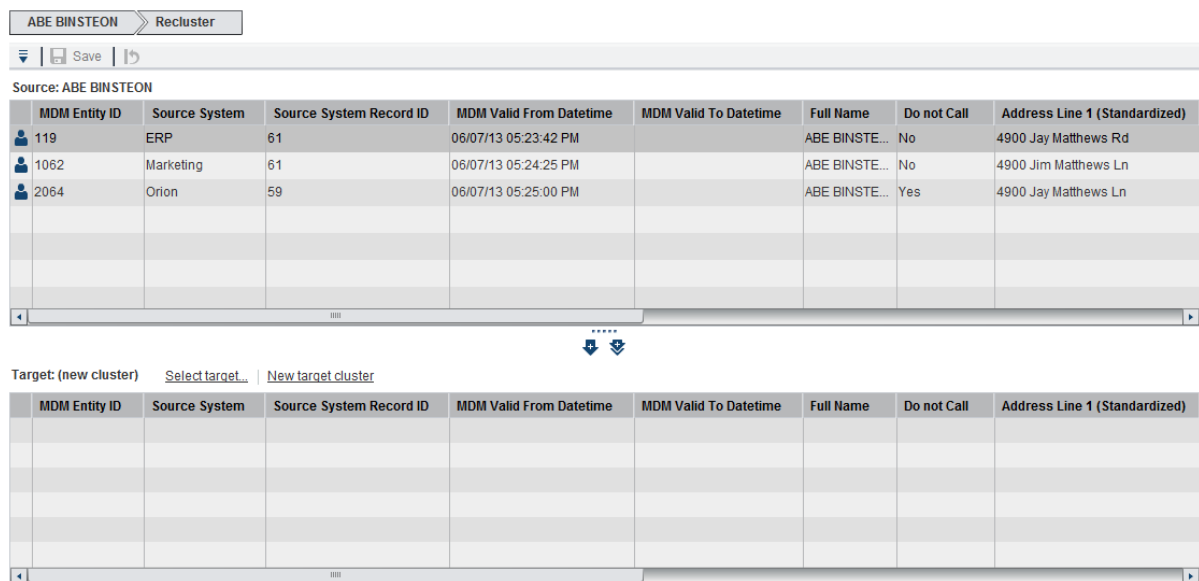
1. Select a record for reclustering on the **Records** tab.
2. Click  and select **Recluster**. The recluster editor opens:

Figure 6.9 Recluster Editor



ABE BINSTEON >> Recluster

Save

Source: ABE BINSTEON

MDM Entity ID	Source System	Source System Record ID	MDM Valid From Datetime	MDM Valid To Datetime	Full Name	Do not Call	Address Line 1 (Standardized)
119	ERP	61	06/07/13 05:23:42 PM		ABE BINSTE...	No	4900 Jay Matthews Rd
1062	Marketing	61	06/07/13 05:24:25 PM		ABE BINSTE...	No	4900 Jim Matthews Ln
2064	Orion	59	06/07/13 05:25:00 PM		ABE BINSTE...	Yes	4900 Jay Matthews Ln

Target: (new cluster) [Select target...](#) [New target cluster](#)

MDM Entity ID	Source System	Source System Record ID	MDM Valid From Datetime	MDM Valid To Datetime	Full Name	Do not Call	Address Line 1 (Standardized)

The toolbar at the top of the recluster editor contains an **Actions** menu with the following options:

Save

saves changes to the cluster.

Close Recluster Editor

closes the recluster editor. The editor can also be closed by clicking the left

navigation breadcrumb, such as

ABE BINSTEON >> Recluster

Select target

opens a dialog box that enables you to search for and select a target cluster to which to move the selected records.



New Target Cluster

Clears the Target table and enables you to move records to create a new cluster.

Reset

resets the cluster to its initial state without saving any changes.





To move one or more records to another cluster:

1. Select either **Select target** or **New target cluster** from the **Actions** menu.
2. If you specified **Select target**, choose a target cluster.
3. Select one or more records in the Source table to be reclustered.
4. Click  to move selected records or  to move all records to the Target table.

The Hierarchies Tab

The **Hierarchies** tab enables you to open and view hierarchies for the entity. This list shows only hierarchies where the entity is the root or top-level element.

Figure 6.10 Hierarchies Tab

Records	Hierarchies	Relationship Diagram	History
<div>    </div>			
Name	Description	Type	
 Bueno Book Customers	Purchasers of parts from Bueno Books	Manufacturer to Purchaser	

The toolbar at the top of the hierarchies table contains an **Actions** menu with the following options:

Open Hierarchy

opens the hierarchy selected in the hierarchies table and displays details about a new tab. See [“Searching Hierarchies” on page 74](#) for more information about hierarchy details.

Refresh

refreshes the table of hierarchies.

The Relationship Diagram Tab

The **Relationship Diagram** Tab enables you to view and create relationships among entities. Entities are referred to as nodes on the relationship diagram.

The toolbar at the top of the **Relationship Diagram** tab contains an **Actions** menu with the following options:

Show Related Entities

searches for and displays entities related to the selected entity.

Open Entity

opens the selected entity (node) in an editor.

Create Remediation Issue

see “[Creating a Remediation Issue](#)” on page 72 for a description of this option.

Hide Node on Diagram

removes the selected node from view on the diagram.

Tools

opens a dialog box that enables you to access available tools.

New Relationship

opens a dialog box for creating a new relationship.

Details Pane

displays or hides details of the selected node in a separate pane.

Node Control Pane

toggles on and off the pane that enables you to specify which related entities to display.

Fit to Area

centers the relationship diagram in its display pane.

Zoom Out

reduces the size of the relationship diagram.

Zoom In

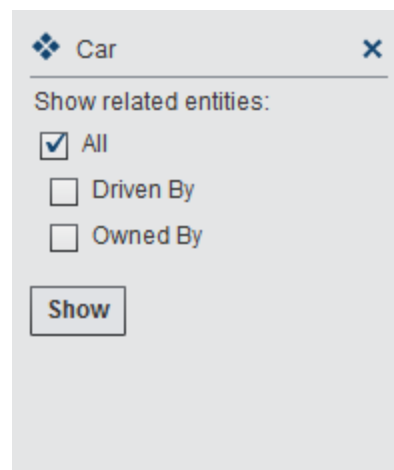
increases the size of the relationship diagram.

Reset Zoom

resets the relationship diagram to its original size.

The Node Control Pane enables you to select what type of relationships to display.

Figure 6.11 Node Control Pane



After selecting relationship types, click **Show**. The related entities are displayed in nodes on the diagram:

3. Fill in the appropriate fields and search to find entities.
4. Select the entity or entities with which to establish relationships. Then click **OK**.

The History Tab

The **History** tab displays the history of insertions, movements, and retirements of records.

Figure 6.14 History Tab

Records

Hierarchies

Relationship Diagram

History

Number of days: 30

MDM Entity ID	Source System	Source System Rec..	Label	Modified By	Modified Date	Action
119	ERP	61	ABE BINSTEON		06/07/13 05:23:42 PM	INSERT
120	Best Record	120	ABE BINSTEON		06/07/13 05:23:42 PM	INSERT
120	Best Record	120	ABE BINSTEON	robarn	06/07/13 05:24:24 PM	RETIRE
1062	Marketing	61	ABE BINSTEON		06/07/13 05:24:25 PM	INSERT
1063	Best Record	1063	ABE BINSTEON		06/07/13 05:24:25 PM	INSERT
1063	Best Record	1063	ABE BINSTEON	robarn	06/07/13 05:25:00 PM	RETIRE
2064	Orion	59	ABE BINSTEON		06/07/13 05:25:00 PM	INSERT
2065	Best Record	2065	ABE BINSTEON		06/07/13 05:25:00 PM	INSERT

The toolbar at the top of the **History** tab contains an **Actions** menu with the following options:

Refresh

refreshes the table of records.

Export

exports the records table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files.

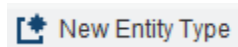
When a character cannot be represented in the PDF file, it is replaced with #.

The toolbar also contains a **Number of Days** field to filter the table of records based on the number of days since the record was modified. Set this field to 0 to show all records.

Creating New Entities

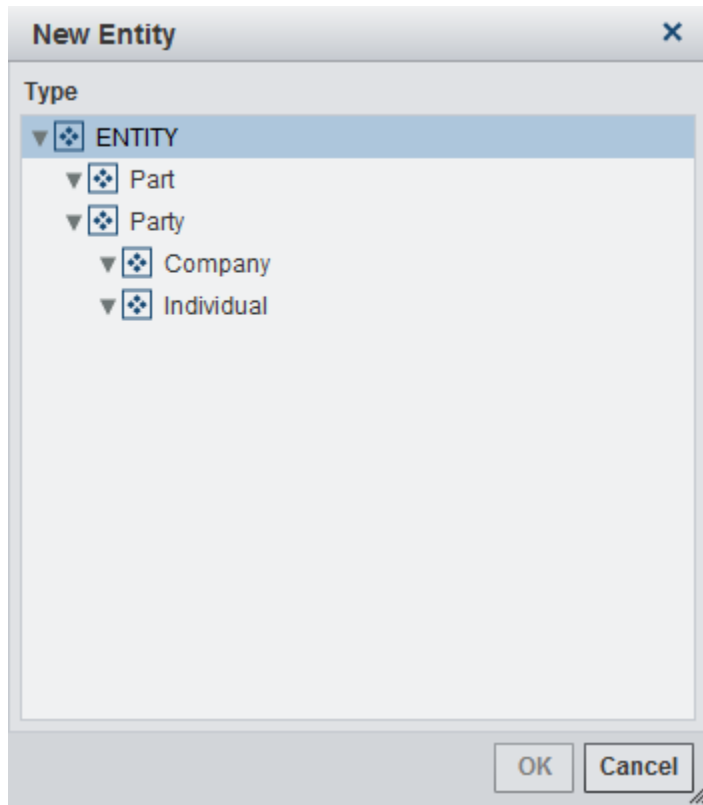
To create a new entity:

1. On the toolbar at the top of the **Master Data Management** tab, click



to display the **New Entity** dialog box:

Figure 6.15 New Entity Dialog Box





2. Select the type of entity. Then click **OK** to display the entity editor:

Figure 6.16 Entity Editor

The screenshot shows the Entity Editor interface. At the top is a toolbar with a 'Save' button (floppy disk icon) and a 'Clear' button (circular arrow icon). Below the toolbar are two main sections:

- MDM Identifiers:** This section contains several input fields:
 - MDM Entity ID:
 - MDM Cluster ID:
 - Source System: (pre-filled with 'MD Manager')
 - Source System Record ID:
 - MDM Valid From Datetime:
 - MDM Valid To Datetime:
- Contact:** This section contains a list of input fields for contact information:
 - Full Name: (marked with a red asterisk, indicating it is required)
 - Website: (with a small 'Go' button to its right)
 - Primary Phone Number:
 - Standardized Primary Phone Number:
 - Secondary Phone Number:
 - Standardized Secondary Phone Number:
 - Fax Number:
 - Standardized Fax Number:
 - Primary Email Address:
 - Secondary Email Address:

3. After filling in the appropriate fields, do one of the following:

- Click  **Save** to save the new entity.
- Click  to clear the fields.

Creating a Remediation Issue



You can create a remediation issue to address invalid or missing data. This option is available by selecting **Create Remediation Issue** from the appropriate  drop-down menu or by clicking  on the toolbar. Either action displays the **Create Remediation Issue** dialog box:

Figure 6.17 Create Remediation Issue

Create Remediation Issue

Package name: *

Subject area:

Issue type: * ▼

Importance: ▼

Assignee:

Due date:

Notes

☐ Select a task template to address the issue: ▼

OK **Cancel**

Complete the following items in the dialog box. Then click **OK** to create the remediation issue.

Package name [required]

is a container where the bad data or non-compliant processes come from. This could be any system your enterprise uses to store and process data..

Subject Area

is the name or label of a category or group within an application. This allows problems to be reported from one or more areas within the reporting application.

Issue Type [required]

select an issue type from the drop-down list.

Importance

select an importance status from the drop-down list.

Assignee

click to select an assignee for the issue.

Due Date

click to select a due date for resolution of the issue.

Notes

add any relevant notes concerning the issue.

Select a task template to address the issue

select this check box to use a template to address the issue. Then choose a template from the drop-down list. See [“Understanding SAS Data Remediation” on page 127](#) for more information.

Creating a data remediation issue sends the issue to the data remediation environment where it can be reviewed and corrected. If you select a task to associate with the data issue, a user-defined workflow can be associated with the data issue, and the issue is routed as defined by the workflow template. You can use SAS Data Remediation to review all outstanding data issues, assign them to others, correct them or act on them, depending on the custom tasks that were associated with the issue.

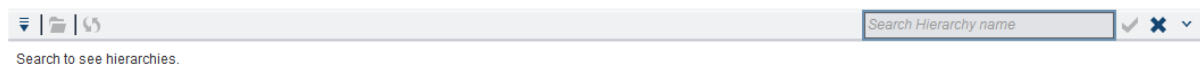
Searching Hierarchies

Overview

Hierarchies are used to display relationships between entities. For example, you can build hierarchies between a household and the people that constitute that household, or between a company and its suppliers.

To access hierarchy searches, click **Hierarchies** on the SAS Data Management Console page. By default, the tab opens to the search toolbar:

Figure 6.18 Hierarchy Search Toolbar





You can perform either a quick search or a field search.

Quick Search

To perform a hierarchy quick search:

1. Enter the hierarchy name into the search field.

Note: The search field is case-insensitive and, by default, searches on even a single letter as a wildcard. For example, searching on the letter *B* returns search results for any type of hierarchy with a name that contains either *B* or *b*.

2. Do one of the following:
 - Click  to begin the search.
 - Click  to clear the search data.

Hierarchies are returned in a table.

Field Search

To perform a hierarchy field search:




1. Click  on the right side of the toolbar to expand the search pane.
2. Enter appropriate search criteria in the fields of the search pane.

Figure 6.19 Search Pane

Hierarchy Type:
 Manufacturer to Purchaser | ▼
 Name:
 Bueno Book Customers
 Description:


Note: The search fields are case-insensitive and, by default, search on even a single letter as a wildcard. For example, searching on the letter *B* in the **Name** field returns search results for any type of hierarchy with a name that contains either *B* or *b*. If you want to limit the types of hierarchies returned, you must specify the hierarchy type in the **Hierarchy Type** field.

3. Do one of the following:
 - Click  to begin the search.
 - Click  to clear the search data.

Successful hierarchy searches return one or more results.

Figure 6.20 Search Results

Search Hierarchy Type : 'Manufacturer to Purchaser'

Name	Description	Root Entity	Type
 Bueno Book Customers	Purchasers of parts from Bueno Books	BUENO BOOKS	Manufacturer to Purchaser

The toolbar at the top of the results table contains an **Actions** menu with the following options:

Open

opens the hierarchy selected in the results table on a new tab.

Export

exports the results table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files. When a character cannot be represented in the PDF file, it is replaced with #.


Open New Search Tab

opens a separate tab for a new search.

Refresh

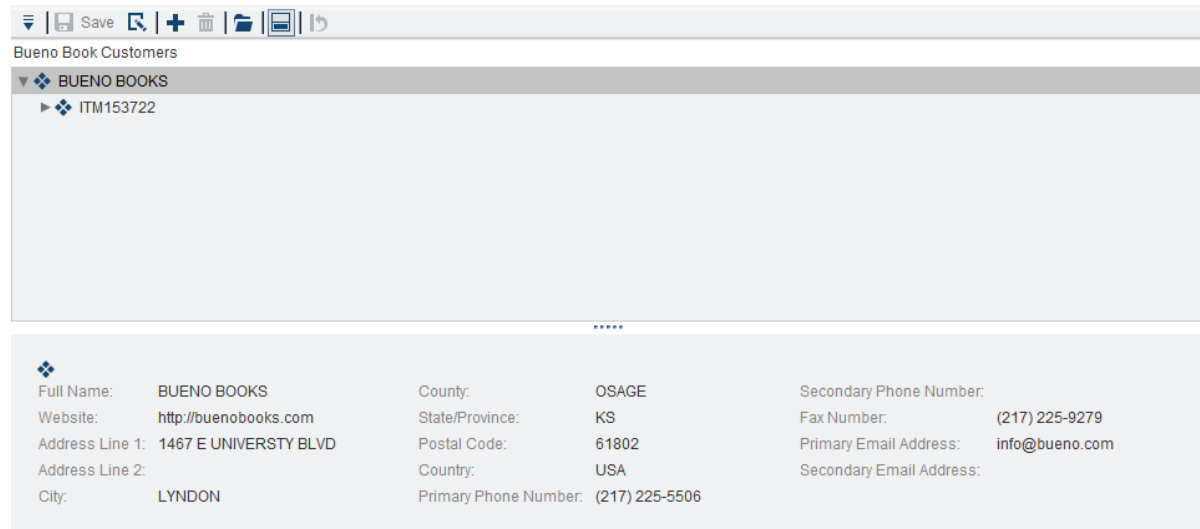
refreshes the table of results.

After hierarchies are returned from a search, you can edit them. To edit a hierarchy:

1. Select a hierarchy in the result table.
2. Click  and select **Open**.

The hierarchy opens on a new tab:

Figure 6.21 Search Results



The toolbar at the top of the hierarchy editor contains an **Actions** menu with the following options:

Save

saves all changes that have been made to the hierarchy.

Name

opens a dialog box that enables you to change the name and description of the hierarchy. Naming hierarchies enables you to search on them later and locate them more easily. Names are not required.

Tools

opens a dialog box that enables you to access available tools for the entities in the hierarchy.

Add Entity (New Relationship)

opens a dialog box that enables you to add a new relationship, which also adds the new relationship to the relationship diagram. See [“The Relationship Diagram Tab” on page 67](#) for more information.

Remove Entity (Retire Relationship)

retires the selected relationship, which also removes the relationship from the relationship diagram. See [“The Relationship Diagram Tab” on page 67](#) for more information.

Open Entity

opens the selected item for editing on a separate tab

Details Pane

displays or hides details of the selected item in a separate pane.

Reset

resets the record to its initial state without saving any changes.

Chapter 7

Using Master Data Management Data Model

About Master Data Management Data Model	77
Overview	77
Importing and Exporting	78
Managing Definition Types	79
Understanding Entity Type Definitions	80
Working with the Entity Type Editor	80
Overview	80
Entity Type Editor Tabs	83
Using the Definition Editor	83
The Permissions Tab	87
Using the Clustering Editor	88
Using the Tools Editor	89
Using the Hierarchy Type Editor	93
Using the Relationship Type Editor	93
Understanding Hierarchy Type Definitions	93
Working with the Hierarchy Type Editor	94
Overview	94
Hierarchy Type Editor Tabs	95
Using the Properties Editor	95
Using the Levels Editor	96
Understanding Relationship Type Definitions	98
Working with the Relationship Type Editor	99
Overview	99
Relationship Type Editor Tabs	100
Using the Properties Editor	100
Using the Rules Editor	102

About Master Data Management Data Model

Overview

From the **Data Model** tab, data administrators can maintain model information for the hub. This information includes the metadata definitions of entity types, hierarchy types, and relationship types and is used to manage and present information in the hub and SAS MDM.

To access the **Data Model** tab, click the **Data Model** link under Master Data Management on the **Data Management Console** page. The following display appears:

Figure 7.1 Data Model Tab

Highlight related types across lists

Entity Types

New Entity Type

Type	Display Name	Description	Last Published	Valid From	Valid To
PARTY	Party	Party	12/04/13 02:37:06 PM	12/04/13 02:37:06 PM	
INDIVIDUAL	Individual	Individual	12/04/13 02:37:07 PM	12/04/13 02:37:07 PM	
COMPANY	Company	Company	12/04/13 02:37:08 PM	12/04/13 02:37:08 PM	
PART	Part	Part	12/04/13 02:37:06 PM	12/04/13 02:37:06 PM	
ENTITY	ENTITY	Abstract Entity Type	12/04/13 02:36:37 PM	12/04/13 02:36:37 PM	

.....

Hierarchy Types

New Hierarchy Type


Type	Display Name	Description	Valid From	Valid To
MANUFACTURER TO PURCHASER	Manufacturer to Purchaser	Displays Manufacturer to Purchaser tr...	12/04/13 02:37:05 PM	

.....

Relationship Types

New Relationship Type

Type	Display Name	Description	Source Entity Type	Target Entity Type	Valid From	Valid To
MANUFACTURER	Manufactures	Company manufactures Part	Company	Part	12/04/13 02:37:0...	
PURCHASER	Purchased	Party purchases Part	Party	Part	12/04/13 02:37:0...	
EMPLOYS	Employs	Company employs Individual	Company	Individual	12/04/13 02:37:0...	

Note: In the following descriptions, **Actions** menu items are accessed by clicking  from the toolbar. Most functions available from the **Actions** menu are also available from icons on the same toolbar. Clicking the icon performs the function.

The toolbar at the top of the **Data Model** tab contains an **Actions** menu with the following options:

Import Model

opens a dialog box from which you choose an entity definition file (.met file) for import.

Export Model

opens a dialog box from which you choose the parts of the current data model that are to be exported to an entity definition file (.met file). Only active objects are shown. You cannot export retired objects.

The toolbar also includes a check box. Selecting this check box highlights related types across the lists in the tables. For example, when the check box is selected, selecting the COMPANY entity type highlights the MANUFACTURER and EMPLOYS relationship types, both of which refer to the COMPANY entity type

Importing and Exporting

Users of the Master Data Management Foundation feature of DataFlux Data Management Studio can also export entity type definitions into .met files, which SAS MDM can import to create new entity types. For more information about the Master Data Management Foundation feature, see the *DataFlux Data Management Studio User's Guide*.

After choosing a .met file from **Import Model** dialog box, the contents of the file are parsed and presented in another dialog box from which you can choose the objects, such as entity types, to import into the hub. When objects are selected in this dialog box, their dependencies are automatically included as well. For example, selecting the sample MANUFACTURES hierarchy type references the PART and COMPANY entity types, which are derived from ENTITY and PARTY respectively.

When you import metadata using this interface, the objects in the hub are merged with the objects represented in the .met file. Importing does not cause any previously existing objects to be retired; importing only adds to the hub. Any objects that were previously retired in the hub but are present in the .met file are left unchanged.

Similarly, when you select objects in the **Export Model** dialog box, their dependencies are automatically included.

Managing Definition Types

Here are the metadata definition types managed from the **Data Model** tab:

entity types

user-defined entities and their attributes. Attributes are data elements for their entity types, in much the same way that columns are part of a table in a relational database.

hierarchy types

user-defined hierarchies of entities.

relationship types

user-defined relationships that exist between the associated entity types.

Each definition type table contains several entries. Selecting an entry under either of the definition types and clicking **Open** from its **Actions** menu opens an editor on a separate tab. For example, selecting COMPANY under **Entity Types** opens an entity editor:

Figure 7.2 Entity Editor Properties Tab

The screenshot shows the 'Entity Editor Properties Tab' for the 'Company' entity type. The interface is divided into two main sections: a left sidebar and a right main area.

Left Sidebar: Contains a tree view of the 'Company' entity type. The root node is 'Company', which has several sub-nodes: 'MDM_ENTITY_TYPE_ID', 'MDM_VALID_TO_DTTM', 'MDM_VALID_FROM_DTTM', 'MDM_LANGUAGE_ID', 'MDM_SRC_SYS_REC_ID', 'MDM_SRC_SYS_ID', 'MDM_ENTITY_CLUSTER_ID', 'MDM_ENTITY_ID', 'NAME', 'ADDRESS1', 'ADDRESS2', 'CITY', 'COUNTY', 'STATE_PROVINCE', 'POSTALCODE', 'COUNTRY', 'PHONE1', 'PHONE2', 'EMAIL1', 'EMAIL2', 'WEBSITE', and 'FAX'.

Right Main Area: Contains two tabs: 'Properties' and 'Permissions'. The 'Properties' tab is active and shows the following fields:

- Type:** * COMPANY
- Description:** Company
- Parent Type:** PARTY
- Valid From:** 08/23/13 02:17:45 PM
- Last Modified:** 08/23/13 02:18:11 PM
- Last Published:** 08/23/13 02:17:45 PM
- Valid To:**
- ☐ Entity is an abstract entity (cannot be instantiated)

Below these fields is the **Display Settings** section:

- Display name:** * Company
- Sort Order:** * 300
- Attribute to use as entity label:** NAME

The individual components of the editors are described in the following sections.

Understanding Entity Type Definitions

Entity type definitions enable you to configure metadata for the hub and Master Data Management to define entity-specific attributes for clustering, match codes, and standardization. For step-by-step examples that demonstrate how to create, generate jobs for, and publish an entity type definition, see [Chapter 11, “Working with SAS MDM,” on page 153](#).

Working with the Entity Type Editor

Overview

The entity type editor is used to create a new entity type definition, to edit an existing definition, or to retire a definition. From the entity type editor, you can collaborate on an entity type definition and publish the definition when you are ready to move it into production. This cycle can be repeated as necessary to achieve the correct definition for an entity type.

Note:

- A retired item becomes inactive, but is not deleted from the hub. It can be viewed as read-only, but cannot be edited. When an entity type is retired, all the entity types that are derived from it are also retired, along with the attributes, clustering conditions, and relationship types for the entity type or types.
- You cannot retire an entity type that is part of a hierarchy type. You must first disassociate the entity type from the hierarchy type. See [“Understanding Hierarchy Type Definitions” on page 93](#) for more information.

The entity type editor is accessed from the **Entity Types** table:

Figure 7.3 Entity Types Table

Entity Types

☰ | + New Entity Type | 📁

Type	Display Name	Description	Last Published	Valid From	Valid To
ENTITY	ENTITY	Abstract Entity Type	06/07/13 04:18:37 ...	06/07/13 04:18:37 ...	
PARTY	Party	Party	06/07/13 04:19:59 ...	06/07/13 04:19:59 ...	
INDIVIDUAL	Individual	Individual	06/07/13 04:19:59 ...	06/07/13 04:19:59 ...	
COMPANY	Company	Company	06/07/13 04:19:59 ...	06/07/13 04:19:59 ...	
PART	Part	Part	06/07/13 04:19:59 ...	06/07/13 04:19:59 ...	

Note: Entity type definition and entity type are equivalent terms.

The toolbar at the top of the **Entity Types** table contains an **Actions** menu with the following options:

New Entity Type

creates a new entity type.

Open

opens the existing entity type selected in the table.

Retire

retires the entity type selected in the table.

Show Retired

displays the retired entity types in the table.

Generate Jobs

generates default batch jobs and data services in support of the entity type. The generation process uses the metadata that has been saved in the hub and publishes .djf and .ddf files to your DataFlux Data Management Server. Changes to the metadata definition (including the metadata for any parent types) are not fully functional until the jobs for the type and any derived types have been regenerated.

Publish

makes the entity type active. This includes generating the tables used to store the data in your DBMS, as well as updating the metadata that the Master Data Management references when determining which entity types to display.

Export

exports the **Entity Types** table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files.

When a character cannot be represented in the PDF file, it is replaced with #.

Selecting **New Entity Type** or **Open** from the **Actions** menu opens the entity type editor on a new tab. For example, opening the COMPANY definition displays the following:

Figure 7.4 Entity Type Editor

The screenshot displays the Entity Type Editor interface. At the top, there is a toolbar with icons for Save, Settings, and other actions. Below the toolbar are tabs for Definition, Clustering, Tools, Hierarchy Types, and Relationship Types. The 'Definition' tab is active, showing a tree view on the left with the 'Company' entity type selected. The right pane shows the 'Properties' and 'Permissions' for the 'Company' entity type. The 'Entity Type Properties' section includes fields for Type (COMPANY), Description (Company), Parent Type (PARTY), Valid From (08/23/13 02:17:45 PM), Last Modified (08/23/13 02:18:11 PM), Last Published (08/23/13 02:17:45 PM), and Valid To. There is also a checkbox for 'Entity is an abstract entity (cannot be instantiated)'. The 'Display Settings' section includes fields for Display name (Company), Sort Order (300), and Attribute to use as entity label (NAME).

The toolbar at the top of the tab contains an **Actions** menu with the following options:

Save

saves the entity type. You can make changes to the entity type by adding attributes as needed.

Retire

retires the entity type.

Generate Jobs

generates default batch jobs and data services in support of the entity type. The generation process uses the metadata that has been saved in the hub and publishes .djf and .ddf files to your DataFlux Data Management Server. Changes to the metadata definition (including the metadata for any parent types) are not fully functional until the jobs for the type and any derived types have been regenerated.

Publish

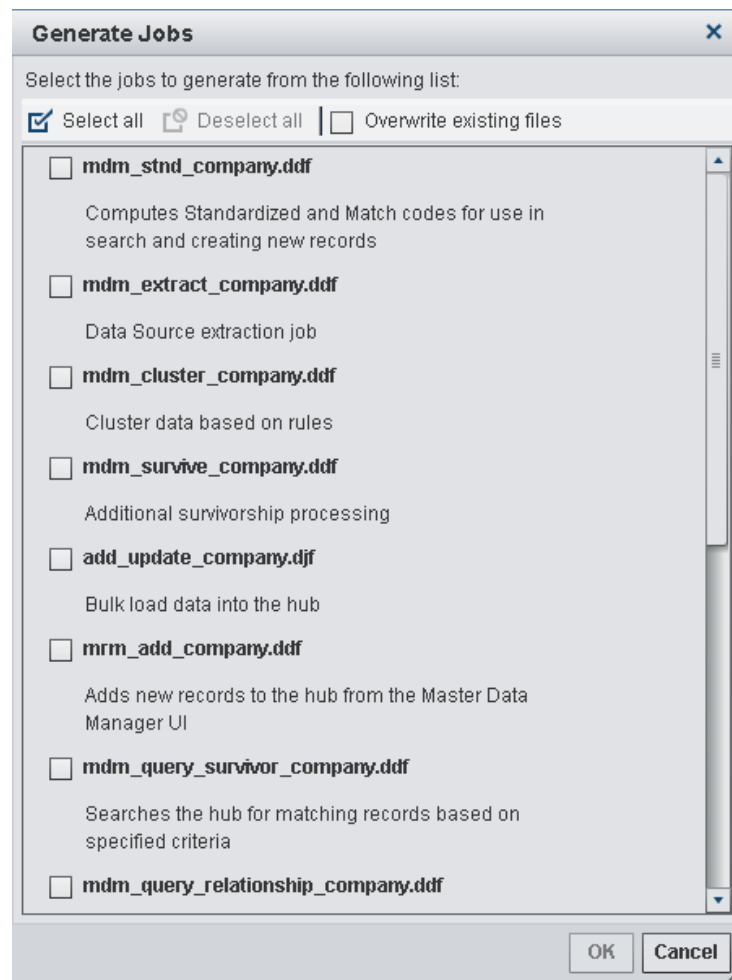
makes the entity type active. This includes generating the tables used to store the data in your DBMS, as well as updating the metadata that the Master Data Management references when determining which entity types to display.

Reset

discards changes and resets the entity type editor to its original state.

The **Generate Jobs** option presents a dialog box from which you can select the jobs that you want to generate:

Figure 7.5 Generate Jobs Dialog Box



For abstract entity types, not all jobs are required. For example, abstract entity types generally support search operations but do not support any of the other jobs. The unnecessary jobs are shown as dimmed in the Generate Jobs dialog box. Also, you might not want to regenerate the standardization job (mdm_stnd_<entity_type>.ddf) if you have customized it for your needs.

Note: Use the **Overwrite existing files** check box with caution. If you select the **Overwrite existing files** check box, files, including customized files, are replaced on your DataFlux Data Management Server, if they exist.

If you are generating jobs from the **Entity Types** table and have selected more than one entity type in the table, the selections that you make apply to every entity type in the selection. The job names appear as '<MULTIPLE>' instead of the entity type to emphasize that you are generating jobs for multiple entity types.

These jobs generally need to be customized to incorporate your standardization and match code generation requirements, cluster conditions, and survivorship rules. Download the batch jobs and services to your DataFlux Data Management Studio environment, and modify them. Then, republish them on the server.

Note: Changes that have an impact on the model of the entity type in the SAS MDM target database must be republished.

The following guidelines apply to publishing, republishing, and saving:

- Publish is required when you create a new entity type.
- Republish is required when you add a new attribute.
- Saving, not republishing, is required when you:
 - change the label, description, or sort order of the entity type
 - change the label, description, group, display orders, read-only bit, or constraint of the attribute
 - add or retire cluster conditions

To see the result of changes after publishing, republishing, or saving items, you must refresh or restart SAS MDM.

Entity Type Editor Tabs

The entity type editor contains the following tabs:

- **Definition**, from which you can view and modify the properties and attributes for an entity type.
- **Clustering**, from which you can create and modify cluster conditions for an entity type.
- **Tools**, from which you can create and modify tools for an entity type definition.
- **Hierarchy Types**, from which you can create and modify hierarchy types for an entity type definition.
- **Relationship Types**, from which you can create and modify relationship types for an entity type.

Using the Definition Editor

Overview

The definition editor is used to view and modify properties and attributes of an entity type. It is also used to specify which groups have permission for access to specific entities and attributes. To display the definition editor, click the **Definition** tab in the entity type editor.

Figure 7.6 Entity Types Definition Editor

The screenshot shows the 'Entity Types Definition Editor' interface. On the left, a tree view lists various entity types under the 'Company' parent. The 'Properties' tab is selected, displaying fields for 'Type' (COMPANY), 'Description' (Company), 'Parent Type' (PARTY), 'Valid From' (08/23/13 02:17:45 PM), 'Last Modified' (08/23/13 02:18:11 PM), 'Last Published' (08/23/13 02:17:45 PM), and 'Valid To'. There is also a checkbox for 'Entity is an abstract entity (cannot be instantiated)'. The 'Display Settings' section shows 'Display name' (Company), 'Sort Order' (300), and 'Attribute to use as entity label' (NAME).

The definition editor includes a pane on the left, under the toolbar, displaying the entity (for example, COMPANY) and listing its attributes. On the right are two additional tabs, **Properties** and **Permissions**. The **Properties** tab displays the properties of the item selected in the left pane, either the entity or one of its attributes. The **Permissions** tab displays the permissions on the item selected in the left pane, either the entity or one of its attributes. By default, the definition editor opens to the **Properties** tab of the entity.

Fields on the Entity Properties Tab

When the entity is selected in the left pane, the **Properties** tab displays the following fields:

Type [required]

the entity type that is used by the hub, bulk jobs, data services, stored procedures, job generation, database naming conventions, and processes that run outside of SAS MDM.

Description

a description of the entity type.

Parent Type

the parent entity type that contributes metadata to the entity type. All entity types ultimately derive from the ENTITY entity type, which is the default.

Valid From [read-only]

the date from which the entity type is active. This date is automatically set when a new entity type is created, and is automatically updated when the entity type is saved.

Last Modified [read-only]

the date and time at which the entity type was last updated. This date is automatically updated when the entity type is saved.

Last Published [read-only]

the date and time at which the entity type was last published.

Valid To [read-only]

the date on which the entity type is inactive or retired. This date is automatically updated when the entity type is retired.

Entity is an abstract entity (cannot be instantiated)

indicates whether the entity type is an abstract entity type. Selecting this option means that concrete instances of the entity type cannot be created. Deselecting this option means that concrete instances of the entity type can be created.

Display Name [required]

the entity type display name, listed under the Label column in the **Entity Types** table.

Sort Order [required]

the entity type's sort order in selection lists. To hide the entity type and its associated attributes in selection lists, enter either zero (0) or a negative value.

Attribute to use as entity label [required]

selecting an attribute from the drop-down list indicates that this attribute is to be used as the default for a SAS MDM search. The display name of this label attribute appears in the search field. A label attribute must be designated for each entity type. For example, you can choose to use either the company name or a stock ticker as the label attribute for a COMPANY entity type.

Note:

- A label attribute must be a string data type; non-string attributes do not appear in the drop-down list.
- Retired attributes cannot be used as a label attributes.
- An attribute must have been saved to be available as a label attribute.

Fields on the Attribute Properties Tab

When an attribute is selected in the left pane, the following fields are displayed on the **Properties** tab

Figure 7.7 Attributes Fields

The screenshot displays the SAS MDM Entity Type Editor interface. On the left, a tree view shows the hierarchy of attributes under the 'Company' entity type. The 'ATTRIBUTE' attribute is selected. The main panel shows the 'Properties' tab with the following fields:

- Attribute Type Properties:**
 - Type: * ATTRIBUTE
 - Group:
 - Description:
 - Data Type: STRING
 - Length: * 50
 - Constraint:
 - Valid From:
 - Valid To:
 - ☐ Attribute is required
 - ☐ Attribute is read only
 - ☐ Encrypt attribute
- Display Settings:**
 - Display name: * ATTRIBUTE
 - Search form order: * 0
 - Search results order: * 0
 - Details panel order: * 0
 - Cluster table order: * 0
 - Record editor order: * 0

Attributes that share the same Group field value are visually grouped as parent and child nodes. The attributes and the higher-level groups are listed according to their respective display order values.

The toolbar over the left pane contains an **Actions** menu with the following options:

New Attribute

creates a new attribute.

New Flux Attribute

creates a new flux attribute, which contains standardized or match code values, based on the attribute selected in the attributes pane. A flux attribute is always associated with another attribute. SAS MDM assigns a default name for a flux attribute by adding a 'DF_' prefix and a '_MC' (match code) suffix to the parent attribute name. For flux attributes containing standardized values, change the default '_MC' suffix to a more meaningful suffix such as '_STND'. See [“Enabling Data Quality” on page 13](#) for more information.

Note: Flux attributes cannot be added to inherited attribute types.

Retire Attribute Type

retires the selected attribute. An attribute cannot be retired if it is used in either a clustering or a relationship type match condition.

Note: For information about considerations when retiring required attributes, see the discussion of the MDM_REQUIRED modifier in [“Attribute Modifiers” on page 25](#).

The attribute fields on the **Properties** tab are as follows:

Type [required]

the attribute type.

Description

the attribute description.

Group

groups similar attributes together in the entity editor. The groups are the collapsible and expandable field sets in the entity editor.

Note: If you enter a new Group value (one that is not already known to SAS MDM), and you subsequently want to use a different locale for a Master Data Management session, you must log on specifying the different locale and create the Group value in that locale's language. If you enter an existing Group value for which the corresponding value for a different locale already exists, the other locale's value is used without requiring further modifications.

Data Type [required]

maps the attribute to a given data type: STRING, TEXT, LIST, DATE, TIMESTAMP, BOOLEAN, INTEGER, REAL, or URL. The data type determines how the attribute data is stored and accessed in the database. By default, all newly defined attributes are set to the STRING data type.

Length

specifies the character length of certain types of data, depending on the type selected in the **Data Type** field. Fields with LIST, STRING, or TEXT data types require a default length with a minimum length of 1. Fields with other data types ignore this length.

Constraint

indicates whether the attribute has a regular expression constraint that requires the incoming attribute data to match the defined regular expression. If the attribute data does not match this constraint, the data is marked as invalid. Typical attributes that

have constraints include phone numbers, addresses, and e-mail addresses. Fields with a LIST data type have built-in constraints that limit specific characters (for example, M, F, and U for Gender). Semicolons (;) act as delimiters. Other constraints specify the allowable format for input characters as regular expressions. For more information about regular expressions, see the *DataFlux Expression Language Reference Guide for Data Management Studio*.

Valid From [read-only]

indicates the date from which the attribute is active. This date is automatically set when a new attribute is created, and is automatically updated when the attribute is saved.

Valid To [read-only]

indicates the date on which the attribute is inactive or retired. This date is automatically updated when the attribute is retired.

Attribute is required

selecting this option indicates that the attribute is required in the entity editor.

Attribute is read only

selecting this option indicates that the attribute is not editable in the entity editor. Read-only fields and labels appear dimmed. In most cases, except the select menus, the text inputs are shown as static text. The select menus are shown as disabled menus.

Encrypt Attribute

selecting this option indicates that the attribute is encrypted in the database. This field can be modified only if the attribute or the entity type has not been published. After the attribute or entity type is published, this field becomes a read-only field.

Display Name [required]

indicates the attribute name. The display name is returned in searches.

Search form order [required]

indicates the display order of the attribute in the entity search form.

Search results order [required]

indicates the display order of the attribute in the search results.

Details panel order [required]

indicates whether the attribute appears in a specific order in the details panel.

Cluster table order [required]

indicates whether the attribute appears in a specific column order in the cluster table on the entity editor **Clustering** tab.

Record editor order [required]

indicates whether the attribute appears in a specific column order in the record editor.

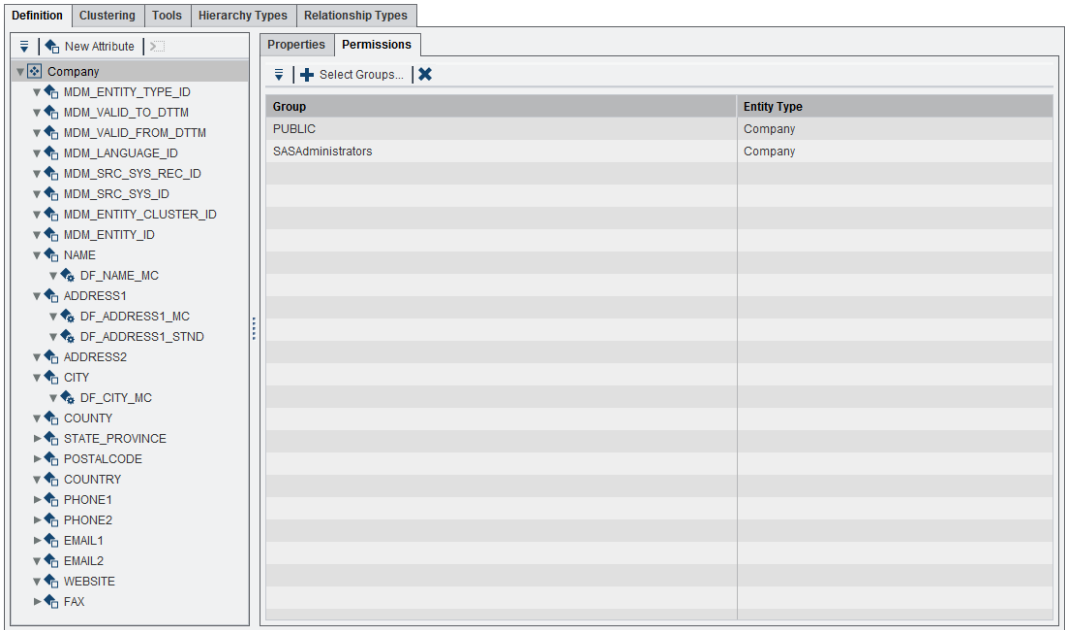
Inherited Attributes

If the attribute is inherited from another entity type, you can change only the display properties. Other values must be changed in the context where the attribute is defined.

The Permissions Tab

The **Permissions** tab displays the permissions on the entity or attribute selected in the left pane.

Figure 7.8 Permissions Tab



For more information about groups, see [“Defining Groups” on page 41](#).

The toolbar on the **Permissions** tab contains an **Actions** menu with the following options:

- Select Groups**
opens a dialog box from which you can select groups to have permissions on the selected attribute or entity.
- Remove Groups**
removes groups that have been selected in the groups list.

Using the Clustering Editor

Overview

The clustering editor is used to create and modify cluster conditions for an entity type. A rule specifies certain attributes for the cluster condition. Records that match the attributes specified in the rule are clustered together. To display the clustering editor, click the **Clustering** tab in the entity type editor.

Figure 7.9 Clustering Editor

Rules

Rules	Entity Type
Cluster records if the following attributes match: MDM_SRC_SYS_REC_ID,MDM_SRC_SYS_ID	ENTITY
Cluster records if the following attributes match: DF_NAME_MC,DF_ADDRESS1_MC,DF_POST...	Party
Cluster records if the following attributes match: DF_NAME_MC,DF_ADDRESS1_MC,DF_CITY...	Party
Cluster records if the following attributes match: DF_NAME_MC,DF_FAX_MC	Company

Rule Editor

Available attributes:

Type	Description
MDM_ENTITY_TYPE_ID	MDM Entity Type from Hub
MDM_VALID_TO_DTTM	MDM Valid To Datetime from Hub
MDM_VALID_FROM_DTTM	MDM Valid From Datetime from Hub
MDM_LANGUAGE_ID	Language from Hub
MDM_SRC_SYS_REC_ID	Source System Record ID from Hub
MDM_SRC_SYS_ID	Source System from Hub
MDM_ENTITY_CLUSTER_ID	MDM Cluster ID from Hub
MDM_ENTITY_ID	MDM Entity ID from Hub
NAME	Full Name from Src Sys

Selected attributes:

Type	Description
------	-------------

The toolbar on the **Clustering** tab contains an **Actions** menu with the following options:

New Rule

creates a new rule with no attributes associated with it.

Retire

retires the selected rule from the list of rules associated with the entity type.

Show Rule Editor

displays or hides the **Available** and **Selected** panes.

When you use the **Generate Jobs** option from the entity type editor, the cluster conditions that you define in the clustering editor are automatically added to the jobs that require this information.

Rules Panes

The Rules pane of the editor lists the cluster conditions that have been defined for an entity type. Select a condition in the list to load it into the Rules Editor panes where you can add attributes to and remove attributes from the condition. Move attributes between the Available and Selected lists either by double-clicking the attribute or clicking the arrow icons.

The condition list includes conditions that are inherited by the entity type (for example, the source system ID and source system record ID condition from ENTITY). However, inherited conditions cannot be modified in the clustering editor. To update an inherited condition, you must open an entity type editor on the entity type where the condition is defined.

Using the Tools Editor

Overview

The tools editor is used to create and modify tools that run from the SAS MDM interface. Some uses of the tools editor are as follows:

- initiating a query for transactional data in external systems
- starting a batch process to load data or generate a report
- creating saved queries
- validating address data when editing records
- making changes to a group of records at the same time
- showing a selected address on a Google map
- performing a web search on selected data
- sending several rows of data via e-mail to a colleague for review

To define a new tool, select an entity, and then choose a data job, process job, or real-time service on the DataFlux Data Management Server. Indicate the output type of the tool and the number of rows to be returned. This defines the locations from which the tool is available. SAS MDM attempts to align entity attributes with tool input parameters. If they do not align, the user is prompted at run time to supply data values for the unsupplied attributes. Once saved on the **Tools** tab, the tool becomes available in SAS MDM wherever entity data tables or entity form editors are used. Output can be returned to existing fields in a form, new data tables, a message, or a new browser tab (if a URL is the tool output). SAS MDM does not validate the logic of the job.

To display the tools editor, click the **Tools** tab in the entity type editor.

Figure 7.10 Tools Editor Properties

The screenshot shows the SAS MDM Tools Editor interface. On the left, a list of tools is displayed, with 'Company Email Process' selected. The main area on the right is divided into two tabs: 'Properties' and 'Permissions'. The 'Properties' tab is active, showing the following configuration for the selected tool:

- Display name:** * Company Email Process
- Description:** Creates and sends email messages
- Inheritance:** Company
- Job/service:** * sasmdm/tool_company_email_process.djf
- Job/service type:** Process Service
- # Input rows:** 0-1
- Result:** Status Only
- Sort Order:** * 10
- Contact person:**
- Valid From:** 12/10/13 10:10:47 PM
- Valid To:**

The tools editor includes a pane on the left, under the toolbar, displaying the tool list. On the right are two additional tabs, **Properties** and **Permissions**. The **Properties** tab displays the properties of the tool selected in the left pane, and the **Permissions** tab displays the permissions on the tool.

The toolbar on the **Tools** tab contains an **Actions** menu with the following options:

New Tool

creates a new tool type

Retire

removes tools that have been selected in the tools list.

By default, the tools editor opens to the **Properties** tab.

The Properties Tab

The toolbar on the **Properties** tab contains an **Actions** menu with the following options:

Inputs and Outputs

Opens a window that displays input and output fields. This window is available after you have selected a job or service. The window can be used to see the mapping of entity attributes to the service parameters.

Show MDM User Prompt - With Input

Opens a window that displays the prompt that the user sees when running the tool with inputs. This window is available after you have selected a job or service.

Show MDM User Prompt - With No Inputs

Opens a window that displays the prompt that the user sees when running the tool without inputs. For example, if a user does not select a row of data but still invokes the tool, this prompt is displayed. This window is available after you have selected a job or service.

When a tool is selected in the left pane, the **Properties** tab displays the following fields:

Display Name [required]

the tool display name.

Description

a description of the tool.

Inheritance [read-only]

the parent entity type.

Jobs/service [required]

the job or service to run for the tool. Click  to open a dialog box and select a job or service.

Jobs/service type [read-only]

displays the type of job or service selected, such as data service, process service, or batch job.

Input rows

the number of rows of input data for the tool to use. Batch jobs and Process Services support only 0-1 input rows.

Result

the type of result to be returned by the tool. Select one of the following options from the drop-down list:

- Attribute Values (Data Services). Attribute Value tools can modify fields in an Entity editor form. Modified fields are highlighted.
- Table (Data Services). Table tools return tabular information that is presented in a new tab.
- URL (Process Services, Data Services). URL tools return a URL that is opened in a new browser tab or window. You might need to enable pop-ups to see the results of URL tools.
- Status Only (Process Services, Data Services). Status Only tools return a status message that is presented as a message pop-up within the user interface.
- Job start notification only (Batch Jobs). Job start notifications are status messages returned by batch jobs. These notifications indicate that the batch tool has started successfully. Completion messages might be generated by the tool, but no further updates are available within the user interface.

The available options depend on whether the tool is a data service, process service, or batch job.

Note: Tools that return URLs require that the browser allow pop-ups.

Sort Order [required]

the sort order of the tool in selection lists. To hide the tool in selection lists, enter either zero (0) or a negative value.

Contact person

person to contact regarding the tool.

Valid From [read-only]

the date from which the tool is active. This date is automatically set when a new tool is created, and is automatically updated when the tool is saved.

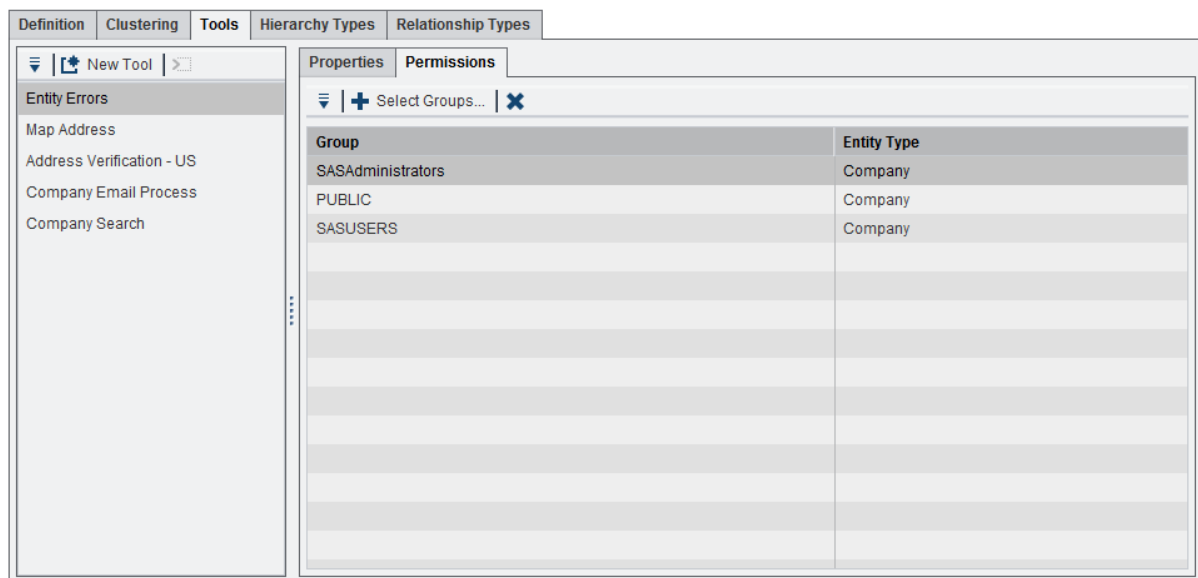
Valid To [read-only]

the date on which the tool is inactive or retired. This date is automatically updated when the tool is retired.

The Permissions Tab

The **Permissions** tab displays the permissions on the tool selected in the left pane. Permissions determine which groups of users can access the tool. General permissions for entity types apply. Therefore, if a group has access to a tool but does not have access to the entity on which the tool is defined, the group cannot access the tool.

Figure 7.11 Tools Editor Permissions



For more information about groups, see [“Defining Groups” on page 41](#).

The toolbar on the **Permissions** tab contains an **Actions** menu with the following options:

Select Groups

opens a dialog box from which you can select groups and assign permissions on the selected tool.

Remove Groups

removes groups that have been selected in the groups list.

Using the Hierarchy Type Editor

The hierarchy type editor is used to create and modify hierarchy types for an entity type. The hierarchy type editor can be accessed either from the **Hierarchy Types** tab of the entity type editor or from the main **Data Model** tab. The **Hierarchy Types** table on the **Hierarchy Types** tab lists hierarchies of which the entity type being edited is a member. The **Hierarchy Types** table on the **Data Model** tab lists hierarchies that have been defined for all entities. For more information about the hierarchy type editor, see [“Working with the Hierarchy Type Editor” on page 94](#).

Using the Relationship Type Editor

The relationship type editor is used to create and modify relationship types for an entity type. The relationship type editor can be accessed either from the **Relationship Types** tab of the entity type editor or from the main **Data Model** tab. The **Relationship Types** table on the **Relationship Types** tab lists relationships of which the entity being edited is a member. The **Relationship Types** table on the **Data Model** tab lists relationships that have been defined for all entities. For more information about the relationship type editor, see [“Working with the Relationship Type Editor” on page 99](#).

Understanding Hierarchy Type Definitions

Hierarchy type definitions enable you to define hierarchy types that are used to build hierarchies of entities. Hierarchy types are built on relationship types that you have already defined. Only defined relationships appear in the hierarchy type editor. The relationship types used to build the hierarchy type determine the types of entities that can exist within a hierarchy and the structure in which they appear.

Consider hierarchy types as defined views of relationship types. For example, if you have defined relationship types that show the connection between a supply company, the parts that it sells, and the people that it employs, you can create a hierarchy type that shows only the supplier-to-part relationship in a hierarchical view. This view does not show the people employed by the supplier even though the supplier-employee relationship exists.

Some constraints and defined behaviors exist for hierarchy types:

- Only one relationship type can be specified for each hierarchy level.
- Entities that are related to each other either through the data elements defined for the relationship type or through explicit linking of the entities in the Master Data Management application are reflected in hierarchies. If you relate new entities, those changes are reflected in hierarchies in which those relationships are defined as levels.
- You cannot retire an entity type or relationship type that is part of a hierarchy type. You must first disassociate the entity type or relationship type from the hierarchy type.

Working with the Hierarchy Type Editor

Overview

The hierarchy type editor is used to create a new hierarchy type definition, to edit an existing definition, or to retire a definition.

Note: A retired item becomes inactive, but is not deleted from the hub. It can be viewed as read-only, but cannot be edited.

The hierarchy type editor is accessed from a **Hierarchy Types** table:

Figure 7.12 Hierarchy Types Table

Hierarchy Types

▼ | + New Hierarchy Type | 📁

Type	Display Name	Description	Valid From	Valid To
MANUFACTURER TO PURCHASER	Manufacturer to Purchaser	Displays Manufacturer to Purchaser tr...	12/04/13 02:37:05 PM	

Note: Hierarchy type definition and hierarchy type are equivalent terms.

The toolbar at the top of the **Hierarchy Types** table contains an **Actions** menu with the following options:

New Hierarchy Type

creates a new hierarchy type.

Open

opens the existing hierarchy type selected in the table.

Retire

retires the hierarchy type selected in the table.

Show Retired

displays retired hierarchy types in the table.

Selecting **New Hierarchy Type** or **Open** from the **Actions** menu opens the hierarchy type editor on a new tab. For example, opening Manufacturer to Purchaser displays the following:

Figure 7.13 Hierarchy Types Editor

The screenshot shows the 'Properties' tab of the Hierarchy Types Editor. At the top, there is a toolbar with a dropdown arrow, a 'Save' button, and a circular arrow icon. Below the toolbar, the 'Properties' tab is active, showing the following fields:

- Type:** * MANUFACTURER TO PURCHASER
- Display name:** * Manufacturer to Purchaser
- Description:** Displays Manufacturer to Purchaser
- Valid From:** 12/03/13 09:35:24 PM
- Valid To:**

The 'Levels' tab is also visible but not selected.

The toolbar above the **Properties** tab contains an **Actions** menu with the following options:

Save

saves the hierarchy type.

Retire

retires the hierarchy type.

Reset

discards changes and resets the hierarchy type editor to its original state.

Hierarchy Type Editor Tabs

The hierarchy type editor contains the following tabs:

Properties

enables you to view and modify properties for a hierarchy type.

Levels

enables you to add or delete hierarchy levels.

Using the Properties Editor

The properties editor is used to view and modify properties of the hierarchy type. To display the properties editor, click the **Properties** tab in the hierarchy type editor.

Figure 7.14 Relationship Type Editor Properties Tab

The screenshot shows a web-based form for editing relationship type properties. At the top, there are two tabs: 'Properties' (selected) and 'Levels'. Below the tabs, the form contains the following fields:

- Type:** A required field (indicated by a red asterisk) containing the text 'MANUFACTURER TO PURCHASER'.
- Display name:** A required field (indicated by a red asterisk) containing the text 'Manufacturer to Purchaser'.
- Description:** A text area containing the text 'Displays Manufacturer to Purchaser'.
- Valid From:** A date and time field containing '12/03/13 09:35:24 PM'.
- Valid To:** An empty date and time field.

The properties editor contains the following fields:

Type [required]

the hierarchy type name. Hierarchy type names must be unique.

Display Name [required]

the display name of the hierarchy type.

Description

a description of the hierarchy type.

Valid From [read-only]

the date from which the hierarchy type is active. This date is automatically set when a new hierarchy type is created and is automatically updated when the hierarchy type is saved.

Valid To [read-only]

the date on which the hierarchy type is inactive or retired. This date is automatically updated when the hierarchy type is retired.

Using the Levels Editor

The levels editor enables you to add levels to a hierarchy. To display the levels editor, click the **Levels** tab in the hierarchy type editor.

Figure 7.16 New Level Dialog Box

The dialog box is titled "New Level" and has a close button (X) in the top right corner. Below the title bar is a "Location:" label followed by a dropdown menu currently showing "Below current selection". Below this is a table with three columns: "Source", "Relationship", and "Target". The first row of the table contains the text "Party", "Purchased", and "Part". There are 14 empty rows below the first one. At the bottom right of the dialog box are two buttons: "OK" and "Cancel".

Source	Relationship	Target
Party	Purchased	Part

3. Select the relationship.
4. Select **Below current selection** or **Above current selection** from the drop-down list.
5. Click **OK**.

Understanding Relationship Type Definitions

Relationship type definitions describe the types of relationships that can exist between the associated entity types. Relationship types definitions specify source and target entity types and have optional match conditions that can be used to determine whether two entities are related. For step-by-step examples that demonstrate how relationship types are established, see [Chapter 11, “Working with SAS MDM,” on page 153](#).

Working with the Relationship Type Editor

Overview

The relationship type editor is used to create a new relationship type definition, to edit an existing definition, or to retire a definition.

Note:

- A retired item becomes inactive, but is not deleted from the hub. It can be viewed as read-only, but cannot be edited.
- You cannot retire a relationship type that is part of a hierarchy type. You must first disassociate the relationship type from the hierarchy type. See [“Understanding Hierarchy Type Definitions” on page 93](#) for more information.

The relationship type editor is accessed from a **Relationship Types** table:

Figure 7.17 Relationship Types Table

New Relationship Type							
Type	Display Name	Description	Source Entity Type	Target Entity Type	Valid From	Valid To	
MANUFACTURER	Manufactures	Company manufactures Part	Company	Part	06/30/13 07:50:3...		
PURCHASER	Purchased	Party purchases Part	Party	Part	06/30/13 07:50:3...		
EMPLOYS	Employs	Company employs Individual	Company	Individual	06/30/13 07:50:3...		

Note: Relationship type definition and relationship type are equivalent terms.

The relationship type table includes relationship types that are inherited by the entity type. However, inherited relationship types cannot be modified in the relationship types editor. To update an inherited relationship type, you must open the entity type where the relationship type is defined.

The toolbar at the top of the **Relationship Types** table contains an **Actions** menu with the following options:

New Relationship Type

creates a new relationship type.

Open

opens the existing relationship type selected in the table.

Retire

retires the relationship type selected in the table.

Show Retired

displays retired relationship types in the table.

Export

exports the **Relationship Types** table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files. When a character cannot be represented in the PDF file, it is replaced with #.

Selecting **New Relationship Type** or **Open** from the **Actions** menu opens the relationship type editor on a new tab. For example, opening Purchaser displays the following:

Figure 7.18 Relationship Types Editor

Save

Properties

Rules

▼ Relationship Properties

Type: * PURCHASER

Description: Party purchases Part

Source Entity Type: * PARTY

Target Entity Type: * PART

Valid From: 05/13/13 09:48:47 AM

Valid To:

▼ Display Settings

Display Name: * Purchased

Inverse Display Name: * Purchased By

Sort Order: * 20

Inverse Sort Order: * 25

The toolbar above the **Properties** tab contains an **Actions** menu with the following options:

- Save**
saves the relationship type.
- Retire**
retires the relationship type.
- Reset**
discards changes and resets the relationship type editor to its original state.

Relationship Type Editor Tabs

The relationship type editor contains the following tabs:

- Properties**
enables you to view and modify properties for a relationship type.
- Rules**
enables you to create rules that define match conditions for a relationship type.

Using the Properties Editor

Overview

The properties editor is used to view and modify properties of the relationship type. To display the properties editor, click the **Properties** tab in the relationship type editor.

Figure 7.19 Relationship Type Editor Properties Tab

The screenshot shows the 'Properties' tab of the Relationship Type Editor. It is divided into two sections: 'Relationship Properties' and 'Display Settings'. In the 'Relationship Properties' section, the 'Type' is 'PURCHASER', 'Description' is 'Party purchases Part', 'Source Entity Type' is 'PARTY', 'Target Entity Type' is 'PART', 'Valid From' is '05/05/13 11:08:22 AM', and 'Valid To' is empty. In the 'Display Settings' section, 'Display Name' is 'Purchased', 'Inverse Display Name' is 'Purchased By', 'Sort Order' is '20', and 'Inverse Sort Order' is '25'.

Properties Editor Fields

The properties editor contains the following fields:

Type [required]

the relationship type name. Relationship type names must be unique.

Description

a description of the relationship type.

Source Entity Type[required]

the source entity type of the relationship. Changing the source entity type removes any match conditions that have been defined for the relationship type.

Valid From [read-only]

the date from which the relationship type is active. This date is automatically set when a new relationship type is created and is automatically updated when the relationship type is saved.

Valid To [read-only]

the date on which the relationship type is inactive or retired. This date is automatically updated when the relationship type is retired.

Display Name [required]

the display name of the relationship type, listed under the Label column in the **Relationship Types** table.

Inverse Display Name [required]

the display name used by the target entity type when presenting the relationship.

Sort Order [required]

the form display order of the relationship type in the entity editor for the source entity type.

Inverse Sort Order [required]

the form display order of the relationship type in the entity editor for the source target type.

Note: The **Sort Order** and **Inverse Sort Order** fields control whether the relationship type appears in the entity editor. To prevent the relationship type from appearing in

the entity editor, enter either zero (0) or a negative value in these fields. To show the relationship type in the entity editor, enter a value greater than zero. Include sufficient padding to allow for future relationship types to be added or for relationship types to be reorganized.

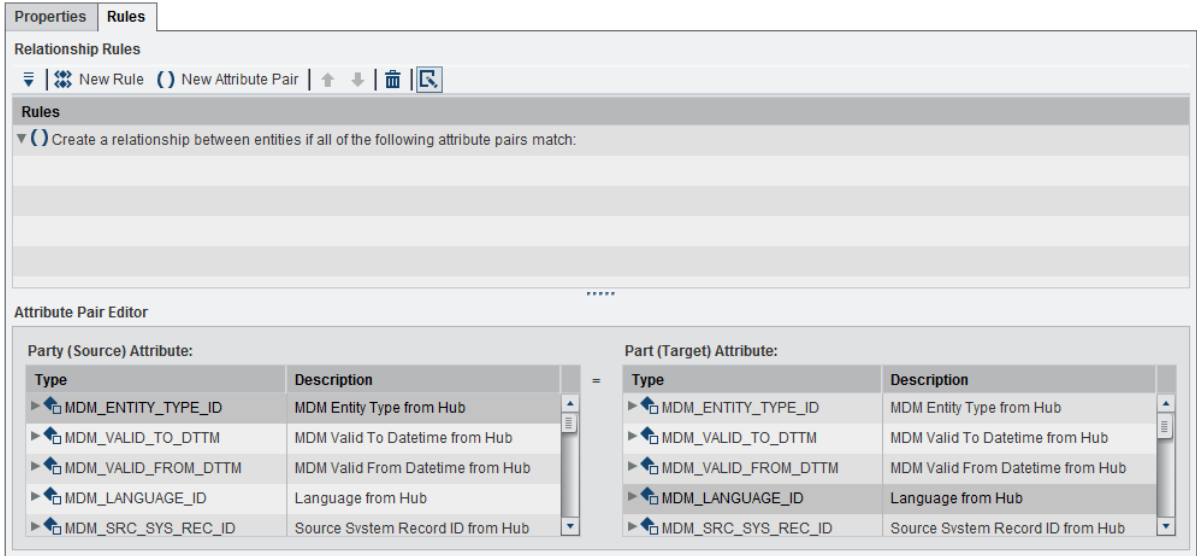
Using the Rules Editor

Overview

The rules editor is used to create rules that define match conditions for a relationship type. You can identify the attributes in the source and target entity types that are used to relate instances of their entity types automatically. All the attributes matches in a rule are used when determining if two entities are related.

The properties editor is used to view and modify properties of the relationship type. To display the rules editor, click the **Rules** tab in the relationship type editor.

Figure 7.20 Relationship Type Editor Rules Tab



The toolbar on the **Rules** tab contains an **Actions** menu with the following options:

New Rule

creates a new rule folder in the **Rules** pane.

New Attribute Pair

creates an empty attribute pair to which you add source and target attributes.

Move Up

moves the selected attribute pair up to a different rule folder.

Move down

moves the selected attribute pair down to a different rule folder.

Delete

deletes the selected item in the **Rules** pane, either a rule folder or an attribute pair.

Show Attribute Pair Editor

displays or hides the Attribute Pair Editor.

Creating and Deleting Rules

To create a rule with attribute pairs:

1. Select **New Rule** from the **Actions** menu. A new rule folder appears in the **Rules** pane.
2. Select **New Attribute Pair** from the **Actions** menu. A new, empty attribute pair appears in the selected rule folder.
3. Select a source attribute and a target attribute in the **Attribute Pair Editor** to add to the selected empty attribute pair.
4. Continue to add attribute pairs. To change the folder location of attribute pairs, select the attribute pair. Then, select **Move Up** or **Move Down** from the **Actions** menu.

To delete a rule or individual attribute pairs:

1. Select the rule or attribute pair that you want to delete.
2. Select **Delete** from the **Actions** menu.

Note: Be aware of the following requirements:

- When creating match conditions, the source and target attributes must have comparable data types. For example, an Integer attribute cannot be compared against a String attribute.
- The source and target attributes must be unique within each group. If redundant rules exist within a group, they are automatically merged when you save the entity type.

Chapter 8

Using Master Data Management Administration

About Master Data Management Administration	105
Overview	105
Managing Languages and Sources Systems	106
Working with the Source System Editor	107
Overview	107
Source Systems Editor Fields	108
Working with the Language Editor	109
Overview	109
Language Editor Fields	110

About Master Data Management Administration

Overview

From the **Administration** tab, data administrators configure source systems and languages that are used in the hub and by Master Data Management.

Note: The **Administration** tab is available only to users who have an administrative role.

To access the **Administration** tab, click the **Administration** link under **Master Data Management** on the SAS Data Management Console page. The following display appears:

Figure 8.1 Administration Tab


Source Systems

New Source System			
Display Name	Description	Valid From	Valid To
Best Record	Best records constructed by the Hub	06/07/13 04:18:37 PM	
MD Manager	Master Data Manager	06/07/13 04:18:37 PM	
ERP	ERP (US)	06/07/13 04:19:56 PM	
Marketing	Marketing Database (US)	06/07/13 04:19:56 PM	
Orion	Orion	06/07/13 04:19:56 PM	
Call Center UK	Call Center (UK)	06/07/13 04:19:56 PM	
Call Center US	Call Center (US)	06/07/13 04:19:56 PM	
CRM	CRM (US)	06/07/13 04:19:56 PM	

.....

Languages

New Language					
Display Name	ISO Locale	QKB Locale	Default	Valid From	Valid To
English (US)	USA	ENUSA	Y	06/07/13 04:19:00 PM	

Note: In the following descriptions, **Actions** menu items are accessed by clicking  from the toolbar. Most functions available from the **Actions** menu are also available from icons on the same toolbar. Clicking the icon performs the function.

Managing Languages and Sources Systems

Here are the items managed from the **Administration** tab:

- languages
 - languages that SAS MDM supports and their associated QKBs and ISO locales.
- source systems
 - source systems from which the contributor records in the hub originate.

Each table contains several entries. Selecting an entry under either of the tables and clicking **Open** from its **Actions** menu opens an editor on a separate tab. For example, selecting **English (US)** under **Languages** opens a language editor:

Figure 8.2 Language Editor

Save

Display Name: * English (US)

MDM QKB Locale: * ENUSA

ISO Locale: * USA

☒ Default

Valid From: 06/21/13 10:11:14 PM

Valid To:

The individual components of the editors are described in the following sections.

Working with the Source System Editor

Overview



The source system editor is used to create a new source system definition, to edit an existing definition, or to retire a definition. The source system is used to specify from which system entity data and cluster information originally came.









Note: A retired item becomes inactive, but is not deleted from the hub. It can be viewed as read-only, but cannot be edited.

The source system editor is accessed from the **Source Systems** table.

Figure 8.3 Source Systems Table

Source Systems

▼ |  New Source System | 

Display Name	Description	Valid From	Valid To
 Best Record	Best records constructed by the Hub	06/07/13 04:18:37 PM	
 MD Manager	Master Data Manager	06/07/13 04:18:37 PM	
 ERP	ERP (US)	06/07/13 04:19:56 PM	
 Marketing	Marketing Database (US)	06/07/13 04:19:56 PM	
 Orion	Orion	06/07/13 04:19:56 PM	
 Call Center UK	Call Center (UK)	06/07/13 04:19:56 PM	
 Call Center US	Call Center (US)	06/07/13 04:19:56 PM	
 CRM	CRM (US)	06/07/13 04:19:56 PM	

Note: The source systems shown in the image are samples provided with SAS MDM and might not match the definitions in your hub.

The following sample source systems are included with SAS MDM and are required for correct operation:

Best Record

a source system representing the hub. All survivor records appear in this source system.

MD Manager

a source system representing Master Data Management. All records created by Master Data Management appear in this source system.

The toolbar at the top of the **Source Systems** table contains an **Actions** menu the following options:

New Source System

creates a source system definition.

Open

opens the existing source system definition selected in the table.

Retire

retires the source system definition selected in the table.

Show Retired

displays retired source system definitions in the table.

Export

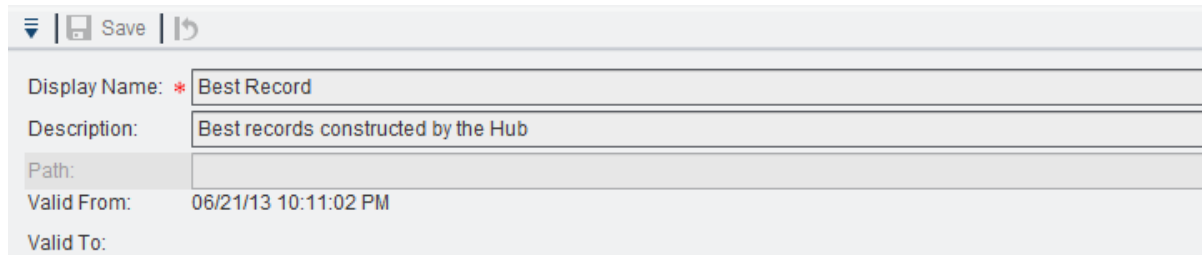
exports the **Source Systems** table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files.

When a character cannot be represented in the PDF file, it is replaced with #.

Selecting **New Source System** or **Open** from the **Actions** menu opens the source system editor on a new tab. For example, opening the Best Record definition displays the following editor:

Figure 8.4 Source System Editor



Display Name: *	Best Record
Description:	Best records constructed by the Hub
Path:	
Valid From:	06/21/13 10:11:02 PM
Valid To:	

The toolbar at the top of the tab contains an **Actions** menu the following options:

Save

saves the source system definition.

Retire

retires the source system definition.

Reset

discards changes and resets the source system editor to its original state.

Source Systems Editor Fields

The source system editor contains the following fields:

Display Name [required]

the source system definition display name, listed under the Name column in the **Source Systems** table.

Description

the source system definition description.

Path [read-only]

path is applicable only for source system data that is migrated from DataFlux Master Data Management Foundation. It is used to correlate the source system with its representation in the Master Data Management Foundation hub, where it is referred to as a source table

Valid From [read-only]

the date from which the source system definition is active. This date is automatically set when a new source system definition is created and is automatically updated when the language definition is saved.

Valid To [read-only]

the date on which the source system definition is inactive or retired. This date is automatically updated when the source system definition is retired.

Working with the Language Editor

Overview

The language editor is used to create a new language definition, to edit an existing definition, or to retire a definition. You can register languages that are used to associate entity data with a given locale. Standardized values and match codes are computed based on the language associated with the entity data.

Note:

- To use the language locale features with standardizations and match codes, you must have a valid QKB license for the language that you want to use. The English (US) language is included with SAS MDM by default.
- A retired item becomes inactive, but is not deleted from the hub. It can be viewed as read-only, but cannot be edited.

The language editor is accessed from the **Languages** table.

Figure 8.5 Languages Table

Languages

☰ | [New Language](#) | 📁

Name	ISO Locale	QKB Locale	Default	Valid From	Valid To
English (US)	USA	ENUSA	Y	05/05/13 11:07:41 AM	

The toolbar at the top of the **Languages** table contains an **Actions** menu with the following options:

New Language

creates a language definition.

Open

opens the existing language definition selected in the table.

Retire

retires the language definition selected in the table.

Show Retired

displays retired language definitions in the table.

Export

exports the **Languages** table in either CSV or PDF format.

Note: Not all characters can be represented in the fonts used in PDF export files. When a character cannot be represented in the PDF file, it is replaced with #.

Selecting **New Language** or **Open** from the **Actions** menu opens the language type editor on a new tab. For example, opening the English (US) language definition displays the following editor:

Figure 8.6 Languages Editor

Display Name: * English (US)

MDM QKB Locale: * ENUSA

ISO Locale: * USA

☒ Default

Valid From: 06/21/13 10:11:14 PM

Valid To:

The toolbar at the top of the tab contains an **Actions** menu with the following options:

Save

saves the language definition.

Retire

retires the language definition.

Reset

discards changes and resets the language editor to its original state.

Language Editor Fields

The language editor contains the following fields:

Display Name [required]

the language definition display name, listed under the Name column in the **Languages** table.

MDM QKB Locale [required]

QKB locale abbreviation that matches the QKB locale. The format is [ISO 3166-1-alpha-2] [ISO 3166-1-alpha-3] (two-letter country code followed by three-letter country code). For example:

- United States = [EN][USA] = ENUSA
- Germany = [DE][DEU] = DEDEU
- France = [FR][FRA] = FRFRA

MDM ISO Locale [required]

ISO 3166-1-alpha-3 code (three-letter country code). For example:

- United States = USA
- Germany = DEU
- France = FRA

Default [required]

selecting this check box indicates that the language definition is the default used by SAS MDM. Only one default language definition can exist for the system.

Valid From [read-only]

the date from which the language definition is active. This date is automatically set when a new language definition is created and is automatically updated when the language definition is saved.

Valid To [read-only]

the date on which the language definition is inactive or retired. This date is automatically updated when the language definition is retired.

Chapter 9

Using SAS Data Remediation

About SAS Data Remediation	113
Overview	113
How Issues Are Organized	114
SAS Management Console	114
Understanding SAS Data Remediation Administration	115
Overview	115
SAS Data Remediation Administration Tab	115
The Properties Tab	117
The Subject Areas Tab	119
The Issue Types Tab	122
SAS Data Remediation Roles	126
Understanding SAS Data Remediation	127
Overview	127
Working with Issues and Items	127
The SAS Data Management Console Portlet	128
SAS Data Remediation Tab	129
Deep Linking to Issues in Data Remediation	140

About SAS Data Remediation

Overview

For many data-intensive IT projects, anomalies or inconsistencies in the data prevent the systems involved from operating optimally and providing clean and timely data to each other and end users. Data remediation provides a means to identify, review, and correct the problem data before it reaches the downstream systems.

SAS Data Remediation makes it easy to capture and review problems found in enterprise data. SAS Data Remediation has a web-based interface for data administrators and a representational state transfer (REST) web service API for system integration. Both of these interfaces interact with a remediation database that contains information about where the problem data is located, which system generated the data, who should see the data, and how the data might be corrected.

How Issues Are Organized

IT systems or data-centric applications (such as SAS MDM) can interact with SAS Data Remediation through a REST web service API. This API enables the originating system, which contains data flagged by data quality business rules, to send information about the state and location of the data errors to SAS Data Remediation. SAS Data Remediation organizes the data and gives specific sets of users access to the remediation issues, as follows:

packages

are collections of items and issues. A single package is generated by a single batch process so that all issues that come from the same process can be viewed collectively.

items

are rows of data in a system. For example, a customer name is an item. In SAS Data Remediation, items are the elements against which business rules have been run to look for erroneous data. Items do not capture the complete external record, but generally point to the record through a key value. One or more issues can be associated with an item.

issues

are descriptions of the problems in the data and include information such as due dates, importance indicators, owners, and statuses.

Issues are linked to Issue Types that can have Tasks (workflow templates) associated with them. Tasks define the correction processes.

Issues can link to data elements that can be transmitted to remediation plug-ins. The plug-ins provide a way to correct the data flagged as having an issue, because source system data is not brought into SAS Data Remediation. Only a reference to the data is brought into SAS Data Remediation.

SAS Management Console

SAS Data Remediation uses the SAS Management Console to define and configure various aspects of the processing environment. Users and roles are set within SAS Management Console. See *SAS Management Console: Guide to Users and Permissions* for more information.

Erroneous data or non-compliant processes can be sent to SAS Data Remediation in several ways. These methods include, but are not limited to, the following programs:

SAS MDM

Data jobs invoke the REST API. SAS MDM sends problem records to SAS Data Remediation for review and correction. When issues that need attention enter SAS Data Remediation, the issues appear in a list in the Data Remediation work space.

SAS Data Integration Studio

Users can write data integration processes that can use the REST API to send problem records to SAS Data Remediation for review and correction.

DataFlux Data Management Studio

Data jobs can invoke the REST-based SAS Data Remediation web service.

Understanding SAS Data Remediation Administration

Overview

The **SAS Data Remediation Administration** tab provides a work area where the administrator configures settings, sets access controls, and plans permissions within SAS Data Remediation views. The administrator sets up each application that interacts with SAS Data Remediation and includes the name of the application, subject areas, and issue types.

Note:

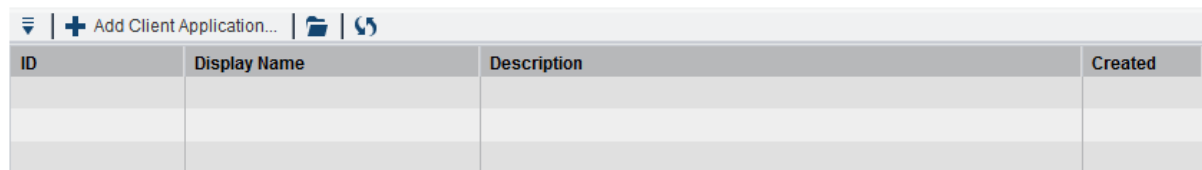
- **SAS Data Remediation Administration** tab is available only to users with administrator privileges.
- All the processes that read from and write to the remediation data files must have access privileges on the data files, even if the processes run on different hosts.

SAS Data Remediation Administration Tab

Overview

To open the **SAS Data Remediation Administration** tab, click the **Administration** option under **Data Remediation** from the SAS Data Management Console. From here, you can add new client applications for data remediation or view and edit existing applications.

Figure 9.1 SAS Data Remediation Administration



ID	Display Name	Description	Created

The toolbar at the top of the tab contains an **Actions** menu with the following options:

Add Client Application

adds a new client application used to interact with SAS Data Remediation. These applications can include SAS MDM, DataFlux Data Management Studio, SAS Data Integration Studio, or other external processes.

Open

opens the client application selected in the client application list on a new tab.

Remove

removes the client application selected in the client application list.

Data Remediation Settings

opens a dialog box that enables you to set the values for various remediation options. See the [“Data Remediation Settings”](#) topic for details.

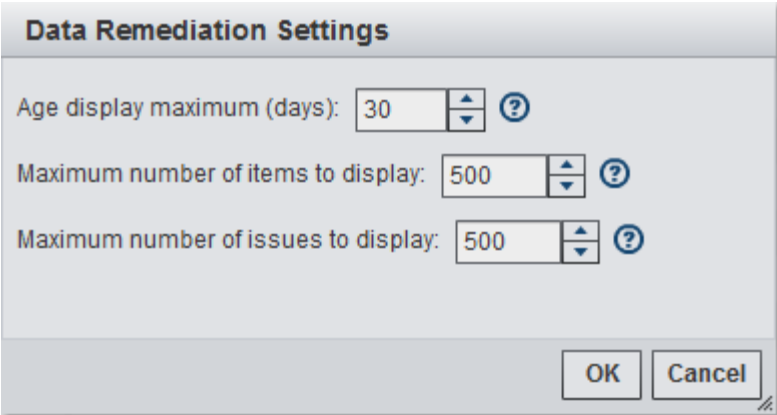
Refresh

refreshes information about the tab.

Data Remediation Settings

The Data Remediation Settings dialog box enables you to set the values for various remediation options.

Figure 9.2 Data Remediation Settings



Age display maximum

sets the maximum number of days displayed in the Age column of the issues list on the **Data Remediation** tab. The default is 30 days. See “[SAS Data Remediation Tab](#)” on page 129 for details.

Maximum number of issues to display

sets the number of issues returned to the application from the data remediation database when queries are made to create Package, Item (issues per item) and Issue views. The default is 500.

Maximum number of items to display

sets the number of items returned to the application from the data remediation database when queries are made to create Package (items per package) and Item views. The default is 500.

Adding and Editing Client Applications

When the **SAS Data Remediation Administration** tab first opens, no applications are associated with the software. You can configure any client applications that you want to use within SAS Data Remediation. Alternatively, the first time an application uses the data remediation service, an association is made for the application if it does not already exist.

1. To add an application to SAS Data Remediation, click **+ Add Client Application...**
2. (Optional) To edit an existing application, select the application in the application list

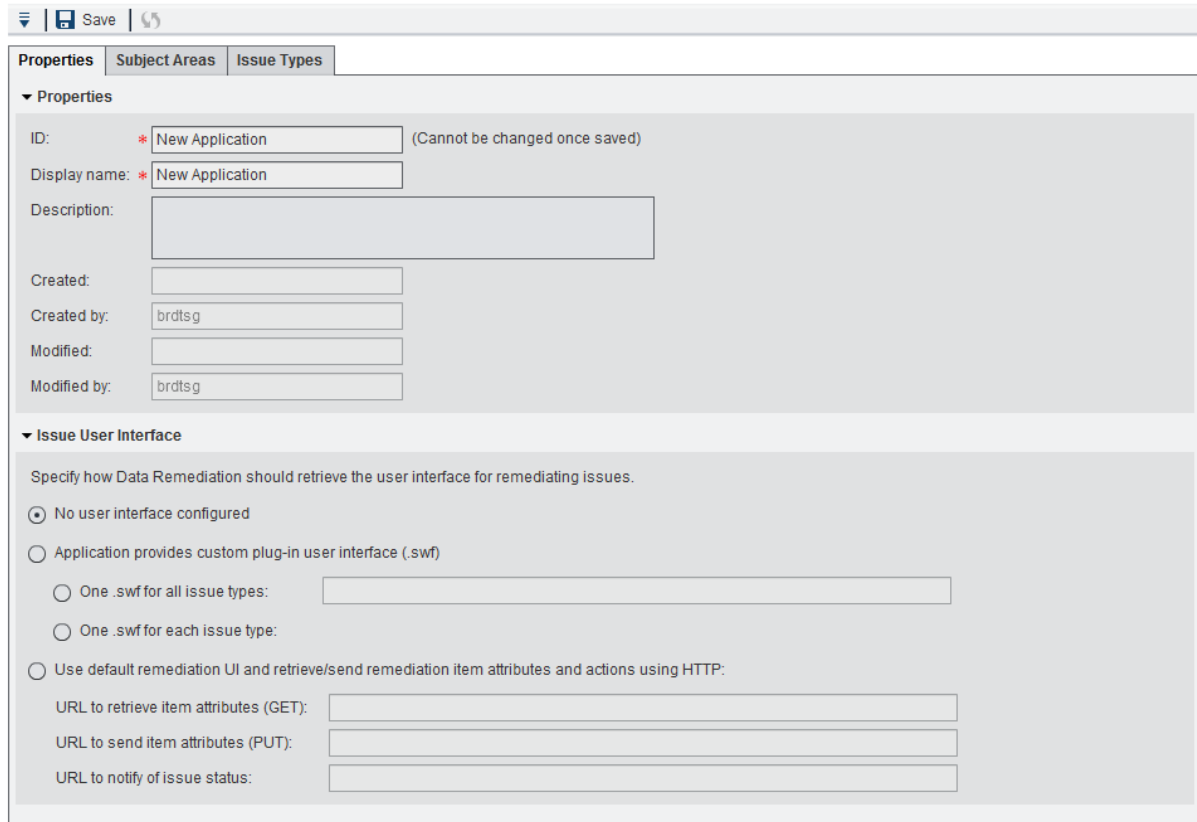
Figure 9.3 SAS Data Remediation Administration

▼ + Add Client Application... 📁 ↺			
ID	Display Name	Description	Created
SAS MDM	SAS MDM		09/18/13 02:...

3. Click .

After you add or open an application, a new tab opens with three sub-tabs: **Properties**, **Subject Areas**, and **Issue Types**.

Figure 9.4 Application Sub-tabs



The screenshot shows a web application interface with a toolbar at the top containing a dropdown menu, a 'Save' button, and a refresh icon. Below the toolbar are three tabs: 'Properties' (selected), 'Subject Areas', and 'Issue Types'. The 'Properties' tab is expanded, showing the following fields:

- ID:** * New Application (Cannot be changed once saved)
- Display name:** * New Application
- Description:** (empty text area)
- Created:** (empty text field)
- Created by:** brdtsg
- Modified:** (empty text field)
- Modified by:** brdtsg

Below the 'Properties' section is the 'Issue User Interface' section, which contains the following options:

- ☒ No user interface configured
- ☐ Application provides custom plug-in user interface (.swf)
 - ☐ One .swf for all issue types: (empty text field)
 - ☐ One .swf for each issue type: (empty text field)
- ☐ Use default remediation UI and retrieve/send remediation item attributes and actions using HTTP:
 - URL to retrieve item attributes (GET): (empty text field)
 - URL to send item attributes (PUT): (empty text field)
 - URL to notify of issue status: (empty text field)

The toolbar at the top of the tab contains an **Actions** menu with the following options:

Save

saves changes to the application information.

Refresh

refreshes information about the tab.

The **Properties** tab is selected by default. The following sections describe the use of these tabs for adding or editing applications.

The Properties Tab

The **Properties** tab displays the following fields:

ID [required]

a unique identifier. Once this field is saved, you cannot make changes to the ID.

Display Name [required]

the name that is displayed for the client application.

Description

the description of the application.

Created

the date on which the application was added to remediation.

Created by

the user name of the person who created the link to the application for remediation.

Modified

the date on which the application was last changed.

Modified by

the user name of the person who last changed the application information.

No user interface configured

enables users to manage issues without a specific application selected. With this setting, no corrective action can take place on the data that originally contained the issue. Only issue tracking properties such as Importance or Assignee are enabled.

Application provides custom plug-in user interface (.swf)

specifies that an external application provides a web-based user interface when a value is set for the .swf file. You can fix issues in this interface.

One .swf for all issue types

selecting this option enables you to specify the location of one .swf file for all Issue Types that are defined for this application. The **plug-in .swf issue code** field, on the **Issue Types** tab, enables you to specify a form to present to the user if the same .swf file contains several form types.

Note: For SAS MDM, set the **One .swf for all issue types** field to

```
/SASMDM/modules/sas.dm.mdm.remediation.flex/sas.dm.mdm.remediation.flex.swf.
```

Leave the **plug-in .swf issue code** field, on the **Issue Types** tab, blank.

One .swf for each issue type

selecting this option enables you to specify the location of one .swf file for each Issue Type that is defined for this application. The **plug-in .swf** field, on the **Issue Types** tab, enables you to specify the full path for the .swf file associated with the issue.

Use default remediation UI and retrieve or send remediation item attributes and actions using HTTP

specifies a server that supports external REST web services. This option provides a simple edit form from which users can correct issues if an external set of REST web services is available to read and write data.

URL to retrieve item attributes (GET)


the URL invoked when SAS Data Remediation opens an issue to retrieve data from an external process. Design the custom REST web service so that the HTTP call that uses the GET function retrieves one data row.

URL to send item attributes (PUT)

the URL invoked when SAS Data Remediation accepts the correction and sends the data back to an external process. Design the custom REST web service so that the HTTP call that uses the PUT function takes data elements from one row in data remediation and sends it to the external process.

URL to notify of issue status

the URL invoked when issue status changes have been made.

Click  on the toolbar to refresh the **Properties** tab view.

The Subject Areas Tab

Overview

The **Subject Areas** tab displays a list of subject areas available in the application that you are currently viewing. A subject area is a way to categorize remediation issues. You can specify that only certain users are given access to a subject area. From this tab, you can also add subject areas to and delete them from the application. The **Subject Areas** tab is divided into three sections: a list of subject areas in the left pane, a **Properties** tab, and a **Task Templates** tab. The **Properties** tab is displayed by default.

Figure 9.5 Subject Areas Tab

The screenshot shows the 'Subject Areas' tab selected. The left pane contains a toolbar with a dropdown menu, a 'New Subject Area' button, and a delete icon. The right pane shows the 'Properties' sub-tab with the following fields:

- Name:** * [text input] (Cannot be changed once saved)
- Description:** [text input]
- Created:** [text input]
- Created by:** [text input]
- Modified:** [text input]
- Modified by:** [text input]

The toolbar at the top of the tab contains an **Actions** menu with the following options:

New Subject Area

creates a new subject area in the subject areas list.

Delete

deletes a selected subject area from the subject areas list.

The Properties Tab

The **Properties** tab displays the following fields for the subject area selected in the subject areas list:

Name [required]

the subject area name. Once you save this name, you cannot change the field.

Description

the description of the subject area.

Created

the date on which the subject area was created.

Created by

the user name of the person who created the subject area.

Modified

the date on which the subject area was last changed.

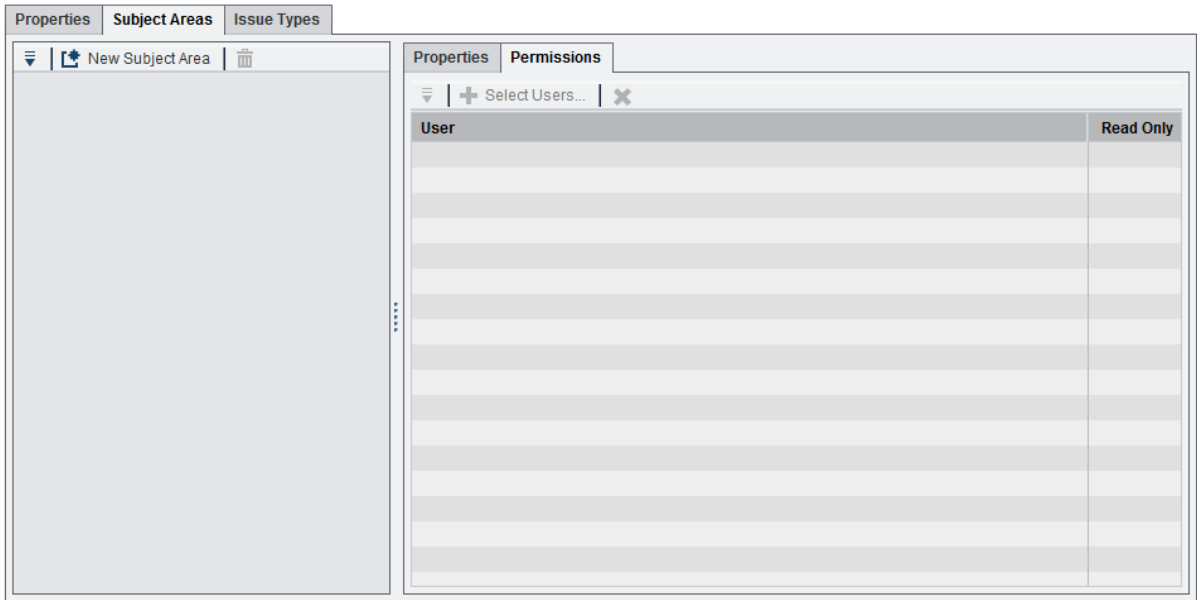
Modified by

the user name of the person who last changed the subject area.

The Permissions Tab

On the **Permissions** tab, you can set permissions for SAS Data Remediation users. You can also add and remove users. The list of available users has been added through SAS Management Console. They have been assigned roles that use capabilities defined for SAS Data Remediation. Users defined in SAS Management Console who are not assigned a role using SAS Data Remediation capabilities do not appear in the list of users. Setting permissions grants or revokes access for a set of users for a given Subject Area. For information about SAS Management Console, see *SAS Management Console: Guide to Users and Permissions*.

Figure 9.6 Subject Areas Permissions Tab




The toolbar at the top of the tab contains an **Actions** menu with the following options:

Select Users

opens a dialog box that enables you to select users who have permissions on the selected subject area.

Delete

deletes the selected user from the list of users who have permissions on the selected subject area. You can select multiple users for deletion.

To select users, click  **Select Users...**. The following dialog box appears:

The Issue Types Tab

Overview

The **Issue Types** tab displays a list of issue types assigned to the application that you are currently viewing. Issue types are a way to categorize issues, making it easy to sort and filter a set of issues before working with them. Issue types are also associated with Task Templates. These templates are user-defined workflows that can be used to route issues to the right user or through the right process. From this tab, you can add new issue types to the application, edit existing issues, or delete issues. The **Issue Types** tab is divided into three sections: a list of issue types in the left pane, a **Properties** tab, and a **Task Templates** tab. The **Properties** tab is displayed by default.

Figure 9.9 Issue Types Tab

The screenshot shows the 'Issue Types' tab in a software application. At the top, there are three tabs: 'Properties', 'Subject Areas', and 'Issue Types'. Below these is a toolbar with a dropdown arrow, a 'New Issue Type' button (with a plus icon), and a delete icon. The main content area is divided into two panes. The left pane is empty. The right pane has two sub-tabs: 'Properties' (selected) and 'Task Templates'. The 'Properties' sub-tab contains the following fields:

- Name:** A text input field with a red asterisk and a note '(Cannot be changed once saved)'.
- Description:** A large text input field.
- Created:** A text input field.
- Created by:** A text input field.
- Modified:** A text input field.
- Modified by:** A text input field.

The toolbar at the top of the tab contains an **Actions** menu with the following options:

New Issue Type

creates a new issue type in the issue types list.

Delete

deletes a selected issue type from the issue types list.

The Properties Tab

The **Properties** tab displays the following fields for the issue type selected in the issue types list:

Name [required]

the issue type name. Once you save this name, you cannot change the field.

Description

the description of issue type

plug-in .swf issue code

this option is displayed only if, on the main **Properties** tab, you have selected both **Application provides custom plug-in user interface (.swf)** and **One .swf for all issue types**. The **plug-in .swf issue code** field enables you to specify a form to present to the user if the same .swf file contains several form types.

Note: For SAS MDM, leave the **plug-in .swf issue code** field blank.

plug-in .swf

this option is displayed only if, on the main **Properties** tab, you have selected both **Application provides custom plug-in user interface (.swf)** and **One .swf for each issue type**. The **plug-in .swf** field enables you to specify the full path for the .swf file associated with the issue.

Created

the date on which the issue type was created.

Created by

the user name of the person who created the issue type.

Modified

the date on which the issue type was last changed.

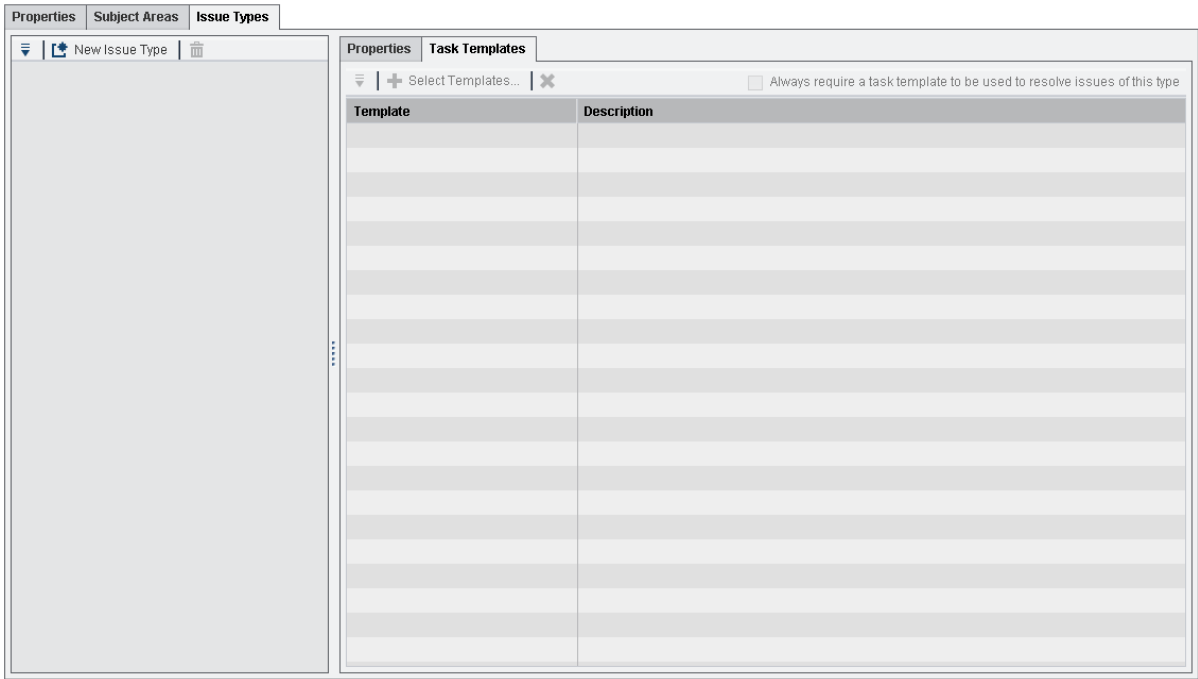
Modified by

the user name of the person who last changed the issue type.

The Task Templates Tab

On the **Task Templates** tab, you can view the workflow templates that have been uploaded through SAS Workflow Studio to the SAS Workflow server. You can associate one or more task templates (workflow definitions) with the selected issue type. For more information about SAS Workflow Studio, see *SAS Management Console: Guide to Users and Permissions*.

Figure 9.10 Issue Types Task Templates Tab



The toolbar at the top of the tab contains an **Actions** menu with the following options:

Select Templates
opens a dialog box that enables you to select task templates to associate with the issue type selected in the issue types list.

Delete
deletes the selected task template from the list of task templates. You can select multiple task templates for deletion.


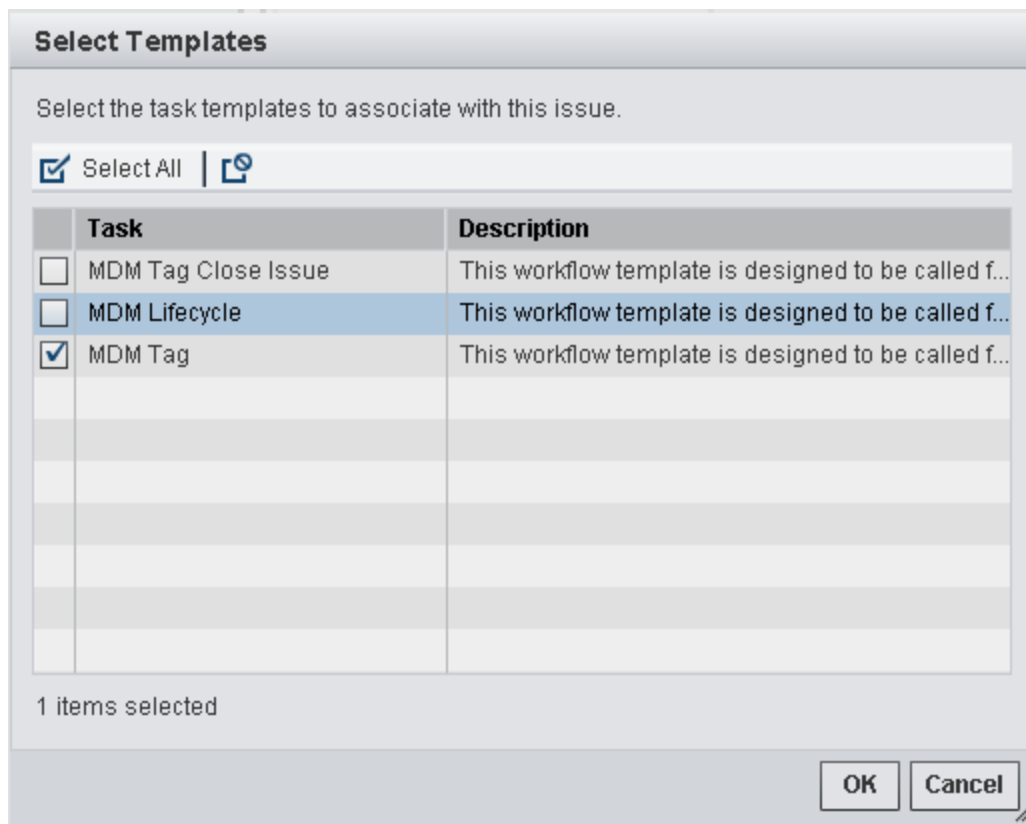


To select templates, click  **Select Templates...**. The following dialog box appears:

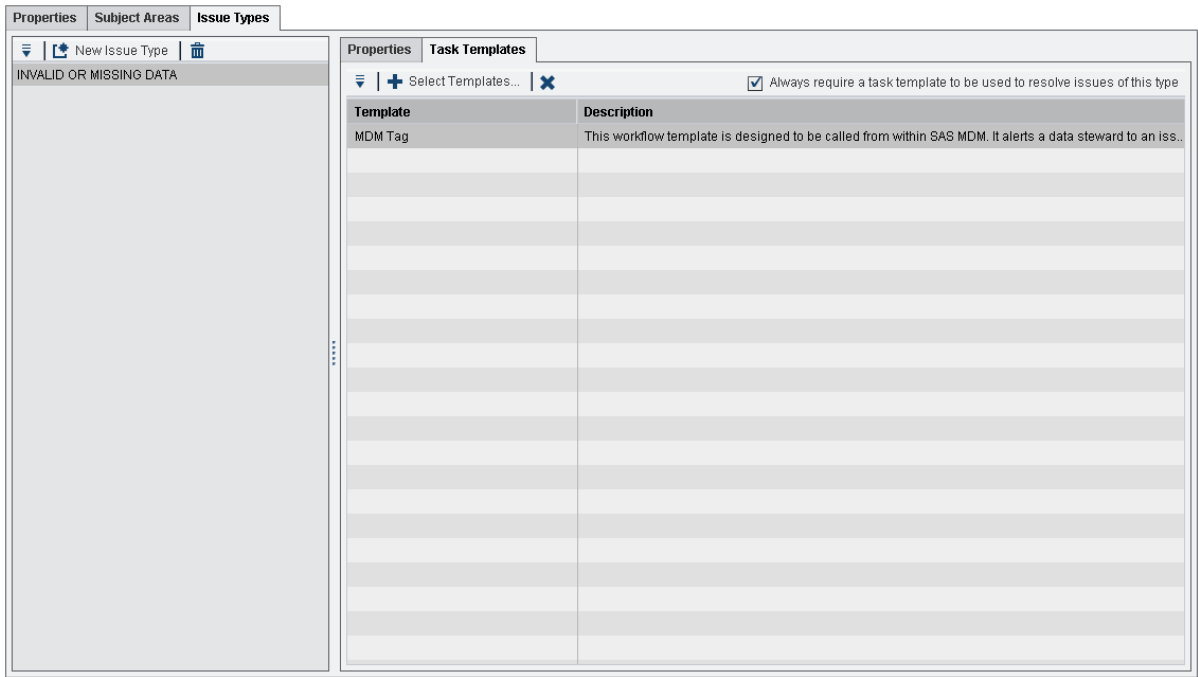
Figure 9.11 Select Templates



You can click the box next to tasks to select them, click  **Select All** to select all tasks, or click  to deselect all tasks.

You can require a user to address a remediation issue through a defined workflow. To enable this, select the **Always require a task template to be used to resolve issues of this type** check box. If this option is selected, the user only can create a task for an issue of this issue type unless a workflow has already been started. Once the workflow has started, the workflow design provides additional menu actions.

Figure 9.12 Issue Types Task Templates Tab



SAS Data Remediation Roles

SAS Data Remediation creates three new roles in SAS Management Console. These roles are tied to distinct capabilities in the data remediation application. Depending on assigned role of the user, the experience in SAS Data Remediation changes. Certain features are available to those assigned one role, but the same features are not available to users in another role.

Here are the roles for SAS Data Remediation:

Table 9.1 SAS Data Remediation Roles

Role	Default Groups	Capability
Data Remediation: Issue Administration	Data Management Administrators	<ul style="list-style-type: none">View Application: The user can view packages, items, and issues.Manage Tasks: The user can edit packages, items, and issues.Issue Administration: The user can configure SAS Data Remediation applications, subject areas, and issue types.
Data Remediation: Issue Management	Data Management Stewards	<ul style="list-style-type: none">View Application: The user can view packages, items, and issues.Manage Issues: The user can edit packages, items, and issues.

Role	Default Groups	Capability
Data Remediation: Issue View	Data Management Business Users	<ul style="list-style-type: none"> View Application: The user can view packages, items, and issues.

Understanding SAS Data Remediation

Overview

The **Data Remediation** tab enables you to choose from the Issue, Item, and Package views and to take action necessary to resolve data issues. An issue is a problem caused by a conflict with user-defined business rules that has been found in data or a process. An item is a record that is associated with an issue. A package is a collection of items and issues. The package can originate from any system your enterprise uses to store and process data.

Your system administrator defines applications in SAS Data Remediation that send data remediation issues to the data remediation environment. Applications can be used to correct data issues. Subject areas categorize the data in such a way that groups of users are given access only to the necessary domain. For example, in SAS MDM, the subject areas are the different entity types that are defined in SAS MDM.

Working with Issues and Items

SAS Data Remediation is designed to manage a practical number of issues assigned to it by an application. A practical number of issues means that they can be reasonably managed by the data stewards assigned to the task. For example, it would be unreasonable for SAS Data Remediation to track 50,000 issues a week because, without a large number of data stewards, it would not be possible to review and correct all of these issues. Although the SAS Data Remediation database might be able to handle very large numbers of issues, those issues that can be displayed at a given time in the web application and allow it to remain responsive to user control is much smaller.

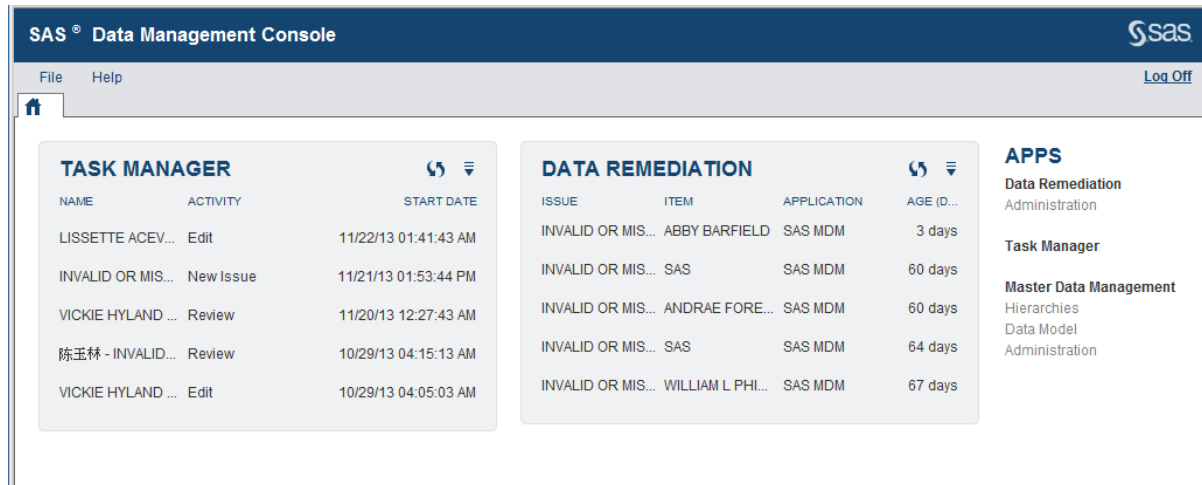
SAS Data Remediation has two settings that enable you to control the maximum number of issues and the maximum number of items that are retrieved at any given time in the SAS Data Remediation application views. See [“Data Remediation Settings” on page 116](#) for more information about these two settings.


If you set the maximum number of issues to be viewed to a value that is less than the total number of issues in an item or package, you need to use a combination of sorting and filtering in the appropriate view to see all data in the remediation database that might be relevant. For example, a default view for items might show only the first 500 items. You can use a filter to limit that list to only the items with issues that you own and that have been open for longer than one week. After refreshing the view, you see updated data with your filter applied. The maximum number limit is applied after the filter is applied. Users can work through the entire set of remediation issues, while ensuring that the web application continues to work optimally, by applying selective filters to data and refreshing the view periodically to retrieve the latest issues.

The SAS Data Management Console Portlet

On the SAS Data Management Console, the **Data Remediation** portlet displays an overview of issues and a sampling of data:

Figure 9.13 SAS Data Management Console



Note: In the following descriptions, **Actions** menu items are accessed by clicking . Some functions available from the **Actions** menu are also available from icons next to the **Actions** menu. Clicking the icon performs the function.

The top right area of the portlet contains an **Actions** menu with the following options:

View

enables you to select whether to view issues, items, or packages in the portlet.

Display

enables you to check the number of items to display in the portlet. You can choose from 5, 10, 20, or 40 items.

Refresh

refreshes the list of items in the portlet list.

Close the Portlet

closes the portlet on the SAS Data Management Console.

Note: See “[Specifying SAS MDM SAS Data Management Console Preferences](#)” on [page 50](#) for additional information about specifying portlet display preferences.

When you click an item in the portlet list, the item opens in SAS Data Remediation on a new tab and is displayed in the view that you have chosen for the portlet (issue, item, or package). You can also click **Data Remediation** in the portlet or **Data Remediation** under Apps to open SAS Data Remediation on a new tab, which displays the item view by default:

Figure 9.14 SAS Data Remediation Tab

Name	Package	Item	Subject Area	Application	Importance	Status	Assignee	Age (Days)
INVALID OR MIS...	BULK EXTRACT	KIMBERLY ELLINGS...	INDIVIDUAL	SAS MDM	★★★★☆	Open	(unknown)	68
INVALID OR MIS...	BULK EXTRACT	VICKIE HYLAND	INDIVIDUAL	SAS MDM	★★★★☆	Edit	Data Manageme...	68
INVALID OR MIS...	BULK EXTRACT	MARK A MCCOY	INDIVIDUAL	SAS MDM	★★★★☆	Open	(unknown)	68
INVALID OR MIS...	BULK EXTRACT	陈玉林	INDIVIDUAL	SAS MDM	★★★★☆	Review	Data Manageme...	68
INVALID OR MIS...	BULK EXTRACT	AUDREA BECKETT	INDIVIDUAL	SAS MDM	★★★★☆	Open	(unknown)	68
INVALID OR MIS...	BULK EXTRACT	WILLIAM L PHILLIPS	INDIVIDUAL	SAS MDM	★★★★☆	In Progress	brdtsg	68
INVALID OR MIS...	BULK EXTRACT	WILLIAM L PHILLIPS	INDIVIDUAL	SAS MDM	★★★★☆	Open	(unknown)	68
INVALID OR MIS...	error	SAS	COMPANY	SAS MDM	★★★★☆	Closed	brdtsg	64

INVALID OR MISSING DATA

Importance: ★★★★★

Status: In Progress

Assignee: brdtsg

Due date:

Issue: INVALID OR MISSING DATA

Item: WILLIAM L PHILLIPS

Package: BULK EXTRACT

Subject Area: INDIVIDUAL

Application: SAS MDM

ADDRESS1: Le Fief Du Parc

STATE_PROVINCE:

POSTALCODE: 44190

Note: INVALID OR MISSING DATA for MDM_LANGUAGE_ID

SAS Data Remediation Tab

Quick Search

The SAS Data Remediation tab opens in the **Issue View** and displays a complete list of issues. You can perform either a quick search or an advanced search to narrow this list to the issues that you want to view.

To perform a quick search:

1. Enter the search term into the search field in the toolbar:

Figure 9.15 Search Field



Note: Click to clear the search characters.

2. After specifying the search criteria, click to initiate the search.

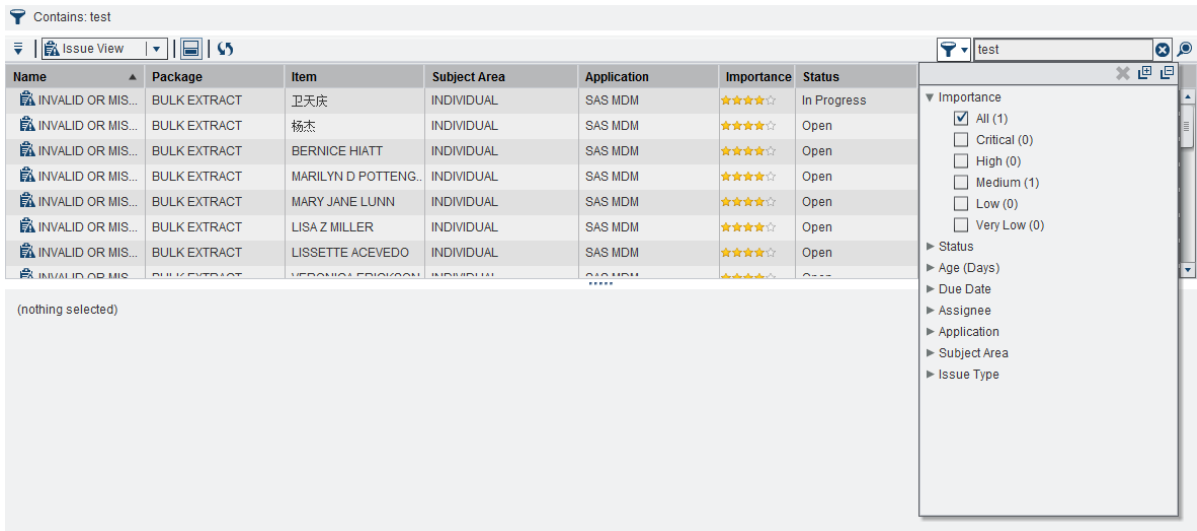
Note: To view the complete issues list again, click to clear the search characters, and then click to restore the list.






Advanced Search



To perform an advanced search:

1. Click to display search criteria:

Figure 9.16 Advanced Search Options



2. You can do any of the following:
- Click  to expand all the search categories.
 - Click  to collapse all the search categories.
 - Click  to restore the search criteria to their default settings.
 - Click  again to close search options.
3. After specifying the search criteria, click  to initiate the search.

Note: To view the complete issues list again, click  to restore the search criteria to their default settings, and then click  to restore the list.

Managing Issues

The **Issue View** toolbar contains an **Actions** menu with the following options:

- View**
expands to display three view options: **Package View**, **Item View**, and **Issue View**.
- **Package View** displays packages, which are collections of items and issues.
 - **Item View** displays items, which are collections of issues.
 - **Issue View** displays issues, which are descriptions of the problems in the data.
- Open**
opens the issue selected in the issues list on a new tab. Selecting multiple issues opens multiple new tabs.
- Edit Group**
opens all the issues selected in the issues list on a single new tab. Available only when more than one issue is selected.
- Details Pane**
displays or hides details of the selected issue in a separate pane. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

Select Columns

enables you to select which columns appear in the table view.

Refresh

refreshes information about the tab.

The Issue View

The table in the **Issue View** lists individual issues. The Details pane, if open, displays details about the selected issue. The fields on the Details pane are the same as those displayed when the issue is opened on a separate tab:

Figure 9.17 Issue View Details

The screenshot shows the 'Details' pane for an issue titled 'INVALID OR MISSING DATA'. The pane includes a toolbar at the top with icons for 'Close', 'Reject', 'Create Task...', and a refresh icon. Below the toolbar, the details are organized into two columns. The left column contains fields for 'Importance' (3 stars), 'Status' (Open), 'Assignee' (brdtsg), and 'Due date' (11/28/13). The right column contains fields for 'Issue' (INVALID OR MISSING DATA), 'Item' (ABBY BARFIELD), 'Package' (Test), 'Subject Area' (INDIVIDUAL), and 'Application' (SAS MDM). To the right of these fields is a large 'Note' field with a text area and a link icon.

In the **Issue View**, the Details pane includes a toolbar at the bottom of the pane. When an individual issue is opened on a new tab, the toolbar is displayed at the top of the pane.

The toolbar contains an **Actions** menu with the following options:

Close

closes the issue when it is resolved. An issue with a closed status cannot be changed. Information about the closed status can be entered in the **Note** field.

Reject

rejects the issue when it is invalid. Information about the invalid issue can be entered in the **Note** field.

Create Tasks

creates a new task for the selected issue. Available only if you have associated one or more task templates with the issue type of the selected issue.

Note: **Create Tasks** opens a dialog box where you can associate a SAS Workflow Studio template with the issue.

Refresh

refreshes information about the tab.

Note: If a task has already been created for the issue, the options available in the toolbar **Actions** menu depend on the SAS Workflow Studio template associated with the issue. See [Figure 9.19 on page 133](#).

In the Details pane, you can view or edit specific information about the issue. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

The Details pane displays the following fields:

Importance

the priority of the issue represented by star icons. The scale ranges from very low (1 star) through low, medium, and high, up to critical (5 stars).

Status


the current state of the issue, such as Open, In Progress, Closed, or Rejected. If a task was created for the issue, then the workflow supplies the status, depending on the current step in the workflow.

- **Open** is the initial status of a data remediation issue. This status can also be selected if the assigned owner wants to reassign the issue to a new owner.
- **In Progress** indicates that ownership has been accepted and the issue is being actively addressed. Users other than the assigned owner cannot work on the issue.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.


- **Closed** indicates that a data remediation issue is resolved. An issue cannot be changed from a **Closed** status. Enter resolution information in the **Note** field before closing the issue.
- **Rejected** indicates an invalid data remediation issue that requires no additional action. Enter rejection information in the **Note** field before closing the issue.

Assignee

is the user name of the person assigned to the issue. Click  to select from an available list of assignees. The list of users is derived from those who have been assigned at least one data remediation role in SAS Management Console.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.

Due Date

is the date by which an issue should be resolved. Click  to select a date from the calendar. This option is not connected to alerts. To set up alerts, you must create a task for the issue that includes defined notifications.

Issue

is the label for the type of issue found in the data.

Item

is the record that triggered the remediation process.

Package

is the package in which the issue is located.

Subject Area

is the subject area assigned to the issue.

Application

is the application where the issue is found.

Field 1

is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.


Field 2

is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.

Field 3

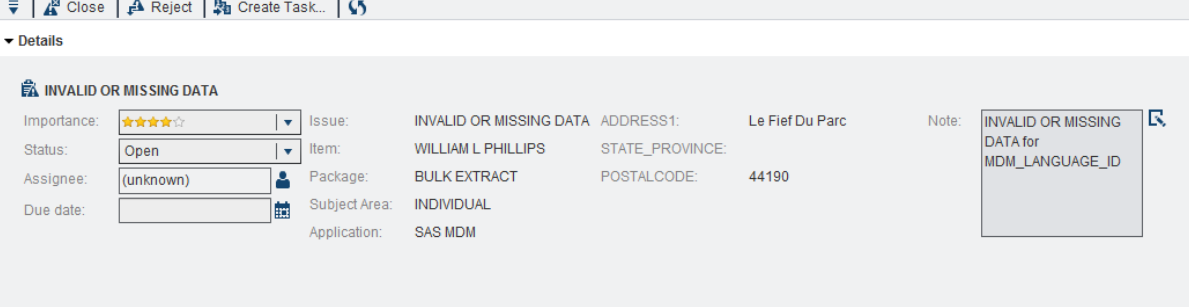
is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.

Note

is a field used to record notes about the issue. Click  to enter your comments in the **Note** field.

The following example shows how the user-defined fields might appear:

Figure 9.18 Issue View Details

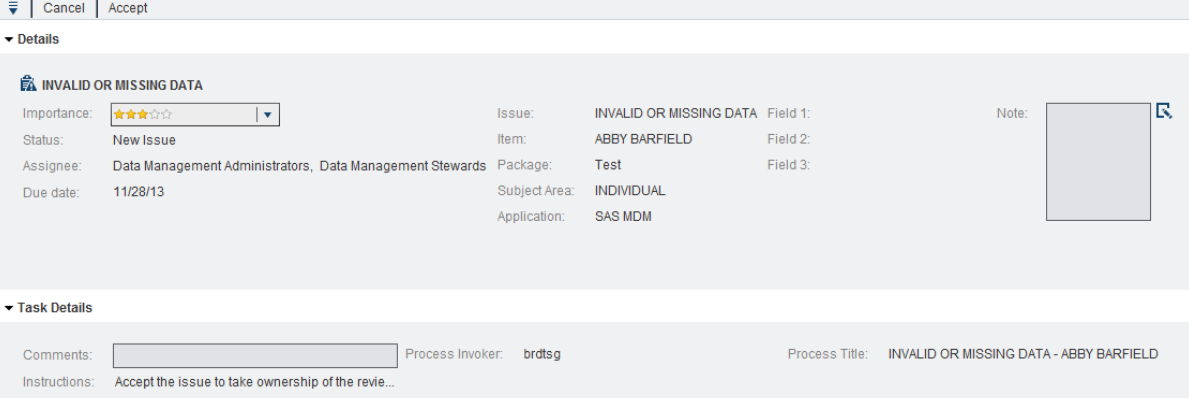


The screenshot shows the 'Details' pane for an issue titled 'INVALID OR MISSING DATA'. The fields are as follows:

Importance:	★★★★☆	Issue:	INVALID OR MISSING DATA	ADDRESS1:	Le Fief Du Parc	Note:	INVALID OR MISSING DATA for MDM_LANGUAGE_ID
Status:	Open	Item:	WILLIAM L PHILLIPS	STATE_PROVINCE:			
Assignee:	(unknown)	Package:	BULK EXTRACT	POSTALCODE:	44190		
Due date:		Subject Area:	INDIVIDUAL				
		Application:	SAS MDM				

If a task has already been created for the issue and you are viewing the issue on its own tab, a Task Details pane appears below the Details pane:

Figure 9.19 Issue View Details



The screenshot shows the 'Details' pane for an issue titled 'INVALID OR MISSING DATA'. The fields are as follows:

Importance:	★★★★☆	Issue:	INVALID OR MISSING DATA	Field 1:		Note:	
Status:	New Issue	Item:	ABBY BARFIELD	Field 2:			
Assignee:	Data Management Administrators, Data Management Stewards	Package:	Test	Field 3:			
Due date:	11/28/13	Subject Area:	INDIVIDUAL				
		Application:	SAS MDM				

Below the Details pane is the 'Task Details' pane:

Comments:		Process Invoker:	brdtsq	Process Title:	INVALID OR MISSING DATA - ABBY BARFIELD
Instructions:	Accept the issue to take ownership of the review...				

If you are viewing the main **Issue View** tab, the Details pane includes two sub-tabs, **Issue Details** and **Task Details**, the fields on which correspond to the fields on the individual issue tab.

The options available in the toolbar **Actions** menu depend on the SAS Workflow Studio template associated with the issue.

The following data object values are always provided by the workflow engine in the Task Details:

Process Invoker

the user that sent the task to SAS Task Manager.

Process Title

the name used in SAS Data Remediation for the selected task.

The display of other values in Task Details are controlled by the design of your workflow template. These values are data objects defined in the workflow template itself. See “[About SAS Task Manager](#)” on page 141 for details.

The Item View

An item is a collection of one or more data remediation issues, all associated with the same record in the reporting application. The Details pane, if open, displays details about the selected item.

Figure 9.20 Item View

Name	Package	Subject Area	Application	Issues	Completed ...	Importance	Status	Assignee	Age (Days)
ABBY BARFIELD	Test	INDIVIDUAL	SAS MDM	1	0				
ANDRAE FOREST	test	COMPANY	SAS MDM	1	0				
AUDREA BECK...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
BERNICE HIATT	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
KIMBERLY ELL...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
LISA Z MILLER	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
LISSETTE ACE...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
MARILYN D PO...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				

ANDRAE FOREST			
Issues:	1	Importance:	☆☆☆☆
Completed(%):	0% (0/1)	Status:	In Progress
		Assignee:	brdtsg
		Due date:	
Item:	ANDRAE FOREST	Field 1:	
Package:	test	Field 2:	
Subject Area:	COMPANY	Field 3:	
Application:	SAS MDM		

The **Item View** toolbar contains an **Actions** menu with the following options:

View

expands to display three view options: **Package View**, **Item View**, and **Issue View**.

- **Package View** displays packages, which are collections of items and issues.
- **Item View** displays items, which are collections of issues.
- **Issue View** displays issues, which are descriptions of the problems in the data.

Details Pane

displays or hides details of the selected issue in a separate pane. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

Select Columns

enables you to select which columns appear in the table view.

Expand Selected

expands the item selected in the table to the issue level.

Expand All

expands all items in the table to the issue level.

Collapse Selected

collapses the item selected in the table to the item level.

Collapse All

collapses all items in the table to the item level.

Refresh

refreshes information about the tab.

In the **Item View**, the Details pane includes a toolbar at the bottom of the pane.

The toolbar contains an **Actions** menu with the following options:

Close All

closes all issues within the selected item. An issue with a closed status cannot be changed.

Reject All

rejects all issues within the selected item.

Change Importance for All Issues

changes the importance of all issues within the selected item.

Change Status for All Issues

changes the status of all of the issues within the selected item.

Change Assignee for All Issues

changes the assignee of all issues within the selected item.

Change Due Date for All Issues

changes the due date for all issues within the selected item.

In the Details pane, you can view or edit specific information. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

The Details pane displays the following fields:

Issues

the number of issues associated with this item.

Completed (%)

the percentage of resolved issues associated with this item.

Importance

the priority of the issue represented by star icons. The scale ranges from very low (1 star) through low, medium, and high, up to critical (5 stars).

Status


the current state of the issues within the item, such as Open, In Progress, Closed, or Rejected. If a task was created for an issue, then the workflow supplies the status, depending on the current step in the workflow.

- **Open** is the initial status of a data remediation issue. This status can also be selected if the assigned owner wants to reassign the issue to a new owner.
- **In Progress** indicates that ownership has been accepted and the issue is being actively addressed. Users other than the assigned owner cannot work on the issue.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.


- **Closed** indicates that a data remediation issue is resolved. An issue cannot be changed from a **Closed** status. Enter resolution information in the **Note** field before closing the issue.
- **Rejected** indicates an invalid data remediation issue that requires no additional action. Enter rejection information in the **Note** field before closing the issue.

Assignee

is the user name of the person assigned to the issue. Click  to select from an available list of assignees. The list of users is derived from those who have been assigned at least one data remediation role in SAS Management Console.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.

Due Date

is the date by which an issue should be resolved. Click  to select a date from the calendar. This option is not connected to alerts. To set up alerts, you must create a task for the issue that includes defined notifications.

Item

is the record that triggered the remediation process.

Package

is the package in which the item is located.

Subject Area

is the subject area assigned to the issues in the item.

Application

is the application where the issues in the item are found.

Field 1

is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.

Field 2

is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.

Field 3

is a user-defined field. It contains a data value from the external application that originally flagged the item as an issue. The label for this field can be changed programmatically.

Note: A value for the fields **Importance**, **Status**, **Assignee**, and **Due Date** is displayed only if all issues in the item have the same value. If the values do not match, the field is blank.



From the **Item View**, you can expand a selected item to display all the associated issues. Click  to expand the item and  to collapse it:

Figure 9.21 Item View with Issue

The screenshot shows the 'Item View' interface. At the top, there's a toolbar with icons for Item View, Package View, and Issue View. Below the toolbar is a table with columns: Name, Package, Subject Area, Application, Issues, Completed, Importance, Status, Assignee, and Age (Days). The table lists several items, with 'INVALID OR MISSING DATA' selected. Below the table is a details pane for the selected item. The details pane includes fields for Importance (3 stars), Status (In Progress), Assignee (brdtsg), Due date, Issue (INVALID OR MISSING DATA), Item (ANDRAE FOREST), Package (test), Subject Area (COMPANY), and Application (SAS MDM). There is also a Note field with a text area and a link icon.

Name	Package	Subject Area	Application	Issues	Completed	Importance	Status	Assignee	Age (Days)
ABBY BARFIELD	Test	INDIVIDUAL	SAS MDM	1	0				
ANDRAE FOREST	test	COMPANY	SAS MDM	1	0				
INVALID OR ...						☆☆☆	In Progress	brdtsg	68
AUDREA BECK...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
BERNICE HIATT	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
KIMBERLY ELL...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
LISA Z MILLER	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				
LISSETTE ACE...	BULK EXTRACT	INDIVIDUAL	SAS MDM	1	0				

INVALID OR MISSING DATA

Importance: ☆☆☆

Status: In Progress

Assignee: brdtsg

Due date:

Issue: INVALID OR MISSING DATA

Item: ANDRAE FOREST

Package: test

Subject Area: COMPANY

Application: SAS MDM

Note:

Close Reject Create Task...

When you select an issue, the Details pane displays the same **Actions** menu and fields as those displayed in the “The Issue View” on page 131. The main **Actions** menu also changes to the **Issue View** options.

The Package View

A package is a collection of one or more remediation items. It enables the reporting system or application to report multiple items and issues inside a single container. The Details pane, if open, displays details about the selected package.

Figure 9.22 Package View

The screenshot shows the 'Package View' interface. At the top, there's a toolbar with icons for Package View, Item View, and Issue View. Below the toolbar is a table with columns: Name, Subject Area, Application, Items, Issues, Completed, Importance, Status, Assignee, and Age (Days). The table lists several packages, with 'test' selected. Below the table is a details pane for the selected package. The details pane includes fields for Items (1), Issues (1), Completed (0%), Importance (3 stars), Status (In Progress), Assignee (brdtsg), Due date, Package (test), Subject Area (COMPANY), and Application (SAS MDM). There is also a Note field with a text area and a link icon.

Name	Subject Area	Application	Items	Issues	Completed	Importance	Status	Assignee	Age (Days)
BULK EXTRACT	INDIVIDUAL	SAS MDM	22	22	0				
error	COMPANY	SAS MDM	1	1	100				
ss	COMPANY	SAS MDM	1	1	0				
test	COMPANY	SAS MDM	1	1	0				
Test	INDIVIDUAL	SAS MDM	1	1	0				

test

Items: 1

Issues: 1

Completed(%): 0% (0/1)

Importance: ☆☆☆

Status: In Progress

Assignee: brdtsg

Due date:

Package: test

Subject Area: COMPANY

Application: SAS MDM

Note:

Close All Reject All

The **Package View** toolbar contains an **Actions** menu with the following options:

View

expands to display three view options: **Package View**, **Item View**, and **Issue View**.

- **Package View** displays packages, which are collections of items and issues.

- **Item View** displays items, which are collections of issues.
- **Issue View** displays issues, which are descriptions of the problems in the data.

Delete

deletes the selected package. Deleting a package is the only way to delete items and their associated issues.

Details Pane

displays or hides details of the selected package in a separate pane. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

Select Columns

enables you to select which columns appear in the table view.

Expand Selected

expands the package selected in the table to the item and issue levels.

Expand All

expands all packages in the table to the item and issue levels.

Collapse Selected

collapses the package or item selected in the table to the package level.

Collapse All

collapses all items and issues in the table to the package level.

Refresh

refreshes information about the tab.

In the **Package View**, the Details pane includes a toolbar at the bottom of the pane. The **Actions** menu of the toolbar is context-sensitive to the types of items and issues contained in the package. The menu options that might appear are described in [“The Issue View” on page 131](#) and [“The Item View” on page 134](#).

In the Details pane, you can view or edit specific information. Changes made in the Details pane are saved right away. The Details pane does not have a save button.

The Details pane displays the following fields:

Items

the number of items associated with this package.

Issues

the number of issues associated with this package.

Completed (%)

the percentage of resolved issues associated with this package.

Importance

the priority of the issue represented by star icons. The scale ranges from very low (1 star) through low, medium, and high, up to critical (5 stars).

Status


the current state of the issues in the package, such as Open, In Progress, Closed, or Rejected. If a task was created for an issue, then the workflow supplies the status, depending on the current step in the workflow.

- **Open** is the initial status of a data remediation issue. This status can also be selected if the assigned owner wants to reassign the issue to a new owner.
- **In Progress** indicates that ownership has been accepted and the issue is being actively addressed. Users other than the assigned owner cannot work on the issue.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.


- **Closed** indicates that a data remediation issue is resolved. An issue cannot be changed from a **Closed** status. Enter resolution information in the **Note** field before closing the issue.
- **Rejected** indicates an invalid data remediation issue that requires no additional action. Enter rejection information in the **Note** field before closing the issue.

Assignee

is the user name of the person assigned to the issue. Click  to select from an available list of assignees. The list of users is derived from those who have been assigned at least one data remediation role in SAS Management Console.

Note: When you select a status of **In Progress**, you automatically become the **Assignee** for the issue.

Due Date

is the date by which an issue should be resolved. Click  to select a date from the calendar. This option is not connected to alerts. To set up alerts, you must create a task for the issue that includes defined notifications.

Package

is the package name.


Subject Area

is the subject area assigned to the issues in the package.

Application

is the application where the issues are found.

Note

is a field used to record notes about the package. Click  to enter your comments in the Note field.

Note: A value for the fields **Importance**, **Status**, **Assignee**, and **Due Date** is displayed only if all issues in the package have the same value. If the values do not match, the field is blank.



From the **Package View**, you can expand a selected package to display all the associated items and issues. Click  to expand the package and  to collapse it:

Figure 9.23 Package View

Name	Subject Area	Application	Items	Issues	Completed ...	Importance	Status	Assignee	Age (Days)
BULK EXTRACT	INDIVIDUAL	SAS MDM	22	22	0				
error	COMPANY	SAS MDM	1	1	100				
ss	COMPANY	SAS MDM	1	1	0				
test	COMPANY	SAS MDM	1	1	0				
INVALID OR MISSING DATA						★★★★☆	In Progress	brdtsg	68
Test	INDIVIDUAL	SAS MDM	1	1	0				

INVALID OR MISSING DATA			
Importance:	★★★★☆	Issue:	INVALID OR MISSING DATA
Status:	In Progress	Item:	ANDRAE FOREST
Assignee:	brdtsg	Package:	test
Due date:		Subject Area:	COMPANY
		Application:	SAS MDM
		Field 1:	
		Field 2:	
		Field 3:	
		Note:	

When you select an item or issue the, the Details pane displays the same **Actions** menu and fields as those displayed in the “[The Item View](#)” on page 134 or “[The Issue View](#)” on page 131. The main **Actions** menu also changes to the **Item View** or **Issue View** options.

Deep Linking to Issues in Data Remediation

Deep linking refers to constructing a URL hyperlink that, once initiated, can take a user directly to an issue in SAS Data Remediation rather than requiring the user to search for it. For example, you can construct a URL link in an e-mail and send to a data steward. When the data steward clicks the URL link, it opens the SAS Data Remediation application and loads the issue encoded in the URL.

From within the appropriate environment, create a URL using the following syntax:

```
http://<your_host>/SASDataManagement/
#issue=<issue_ID>&module=REMEDIATION
```

For example, you might create the following:

```
http://localhost:8080/SASDataManagement/
#issue=4312&module=REMEDIATION
```

Note: Creating a remediation issue in SAS MDM submits a deep link along with the issue, which is accessible through the SAS Data Remediation web service API. See “[Creating a Remediation Issue](#)” on page 72 for more information.

Chapter 10

Using SAS Task Manager

About SAS Task Manager	141
Overview	141
Working with Data Objects	142
Understanding SAS Task Manager	144
Overview	144
The SAS Data Management Console Portlet	144
The SAS Task Manager Tab	145
The Task Tab	147
SAS Task Manager Roles	148
Deep Linking to Workflows in Task Manager	149

About SAS Task Manager

Overview

Many organizations need to coordinate people, processes, and technology through defined business workflows. SAS Workflow Studio is a suite of applications and services that work together to meet these needs. It helps organizations model, automate, integrate, and streamline business processes. Several SAS solutions use SAS Workflow Studio to orchestrate user or system activity. SAS Task Manager is one of these applications. SAS Task Manager gives users direct access to a workflow that is initiated from one of the other SAS applications. With SAS Task Manager, users can start, stop, and transition workflows that have been uploaded to the SAS Workflow Studio server environment. SAS Task Manager also lets users interact with certain elements of active workflows called data objects. These data objects can be used to trigger additional activity that goes beyond standard workflow interaction.

SAS Task Manager was designed for the following users:

Table 10.1 SAS Task Manager Users

User	Privileges
Administrators	Administrators can terminate active workflows or transition them to other users in case of errors or deadlocks.

User	Privileges
Workflow template designers	Designers can test workflow behavior outside of the parent application that consumes the workflow definition.
Designated Users	<p>Designated users can perform the following tasks:</p> <ul style="list-style-type: none"> • Start a workflow that acts as a scheduling process to initiate programs or web services in a defined way. • Provide data input to workflow-backed processes through editable data objects in the workflow itself • Initiate or terminate project-based actions. Because users can start and stop workflows, they can control activities that tie into policies supported by the workflow design, including notifications, web service calls, scheduling a task, or invoking a SAS program. • Interact with the workflow process through a custom form.

Working with Data Objects

Overview

Two unique features make SAS Task Manager more than just a window to the SAS Workflow Studio environment:

- ability to add special attributes to workflow data objects
- ability to respond to particular data object sets in a workflow

Adding Special Attributes to Workflow Data Objects

By default, data objects and their values are always shown to users in the **Data** tab. By adding specific attributes to the data objects, users can hide data objects from view and make the values of certain data objects editable. Only the data objects short text, long text, number, and date are supported for editing. You can use the following attributes:

Table 10.2 Data Object Attributes

Attribute	Purpose
HIDE	Adding a HIDE attribute to a data object controls its visibility in SAS Task Manager. Setting it to 0 allows users to see the data object and value. Setting it to 1 hides the data object and value. By default, the data object and its value are displayed.
WRITE	Adding a WRITE attribute to a data object lets users edit the data object value in SAS Task Manager. Setting it to 0 sets the data object value to read-only. Setting it to 1 enables users to edit the value. By default, the data object value is read-only.

Other than the short text, long text, number, and date data objects, all other data object data type values are set to read-only by default, even if you set the WRITE attribute to 1. If you need to edit values other than the supported types, use one of the supported types to capture the information. Then use data object substitution in the workflow design to transform the value to another type.

Note:

- Not all data types can be transformed into other data types without loss of information.
- Use all uppercase text for the HIDE and WRITE attributes.

Responding to Particular Data Object Sets in a Workflow

SAS Task Manager can respond to a particular set of data objects the workflow. This data object set, if found, instructs SAS Task Manager to present custom edit form capabilities to the user. The workflow might contain steps in which you want to present information to the user for modification, review, or approval. For each task in the workflow, you must add several data objects that indicate which plug-ins to use.

The following table describes the special set of data objects attached to each task in the workflow that present a customer edit form:

Table 10.3 *Data Objects*

Task Data Object	Description
UI_APP	Corresponds to the name of the application that is invoked.
UI_APPMODULE	Shows the location of the .swf file to load.
UI_ENABLED	Notifies SAS Task Manager to present the plug-in (0=No, 1=Yes).
UI_FORMID	Is an optional value to let the plug-in know what form to show to the user.
UI_KEY	Displays a value known by the plug-in and used to fetch the correct data row for display in the plug-in.
UI_SUBJECTAREA	Is an optional value known by the plug-in and typically used to help the plug-in find the correct external data to display.
UI_HOLD	Determines whether the action taken in the plug-in is a final action. The object commits a record to a target data source or is an action that saves the change in a temporary location (0=final save, 1=temporary save).

Other SAS applications or external applications must provide the specialized plug-in that can be invoked through these data objects. For example, SAS MDM provides a plug-in that allows editing of master data in the context of a workflow process. Contact your SAS representative for more information about the design and support of custom plug-ins.

In addition, special treatment is given to certain workflow data objects if they are found in your workflow templates:

BDN Term URL

If a valid URL that deep-links to a Business Data Network term ID is found in a data object with this name, a **Link** field appears in the workflow data panel. Its value is a hyperlink labeled **Open Term**. When you click the link, the referenced term is opened in Business Data Network in a new browser tab.

URL

If a valid URL is found in a data object with this name, a **Link** field appears in the workflow data panel. Its value is a hyperlink labeled **Open in Application**. When you click the link, the referenced term is opened in the application that you specify. For example, you can deep-link into other applications like SAS MDM (“[Deep Linking to Master Data](#)” on page 59) or SAS Data Remediation (“[Deep Linking to Issues in Data Remediation](#)” on page 140).

Understanding SAS Task Manager

Overview

SAS Task Manager provides a location to manage workflow instances that have been initiated in SAS Data Remediation, SAS MDM, or other external applications. See [Chapter 9, “Using SAS Data Remediation,” on page 113](#) for additional information about SAS Data Remediation.

SAS Task Manager displays a summary of information from external processes. Users can monitor and manage assignments for workflow tasks that are displayed in SAS Task Manager, correct the problems in an external system, and then close the issue within SAS Task Manager.

SAS Task Manager does not use a separate administration environment. SAS Task Manager relies on SAS Workflow Studio for workflow template design and management on the SAS Workflow server. For information about SAS Workflow Studio, see the *SAS Workflow Studio: User's Guide*.

The SAS Data Management Console Portlet

On the SAS Data Management Console, the **Task Manager** portlet displays an overview of tasks and a sampling of data:

Display 10.1 SAS Data Management Console


The screenshot shows the SAS Data Management Console interface. At the top is a dark blue header with the SAS logo and 'Log Off' link. Below the header is a navigation bar with 'File' and 'Help' menus. The main content area is divided into three portlets: 'TASK MANAGER', 'DATA REMEDIATION', and 'APPS'. The 'TASK MANAGER' portlet contains a table with columns: NAME, ACTIVITY, and START DATE. The 'DATA REMEDIATION' portlet contains a table with columns: ISSUE, ITEM, APPLICATION, and AGE (D...). The 'APPS' portlet lists various applications like Data Remediation Administration, Task Manager, and Master Data Management.

NAME	ACTIVITY	START DATE
LISSETTE ACEV...	Edit	11/22/13 01:41:43 AM
INVALID OR MIS...	New Issue	11/21/13 01:53:44 PM
VICKIE HYLAND ...	Review	11/20/13 12:27:43 AM
陈玉林 - INVALID...	Review	10/29/13 04:15:13 AM
VICKIE HYLAND ...	Edit	10/29/13 04:05:03 AM

ISSUE	ITEM	APPLICATION	AGE (D...)
INVALID OR MIS...	ABBY BARFIELD	SAS MDM	3 days
INVALID OR MIS...	SAS	SAS MDM	60 days
INVALID OR MIS...	ANDRAE FORE...	SAS MDM	60 days
INVALID OR MIS...	SAS	SAS MDM	64 days
INVALID OR MIS...	WILLIAM L PHIL...	SAS MDM	67 days

APPS

- Data Remediation Administration
- Task Manager
- Master Data Management Hierarchies
- Data Model Administration

Note: In the following descriptions, **Actions** menu items are accessed by clicking . Some functions available from the **Actions** menu are also available from icons next to the **Actions** menu. Clicking the icon performs the function.

The top right area of the portlet contains an **Actions** menu with the following options:

Display

enables you to check the number of items to display in the portlet. You can choose from 5, 10, 20, or 40 items.

Refresh

refreshes the list of items in the portlet list.

Close the Portlet

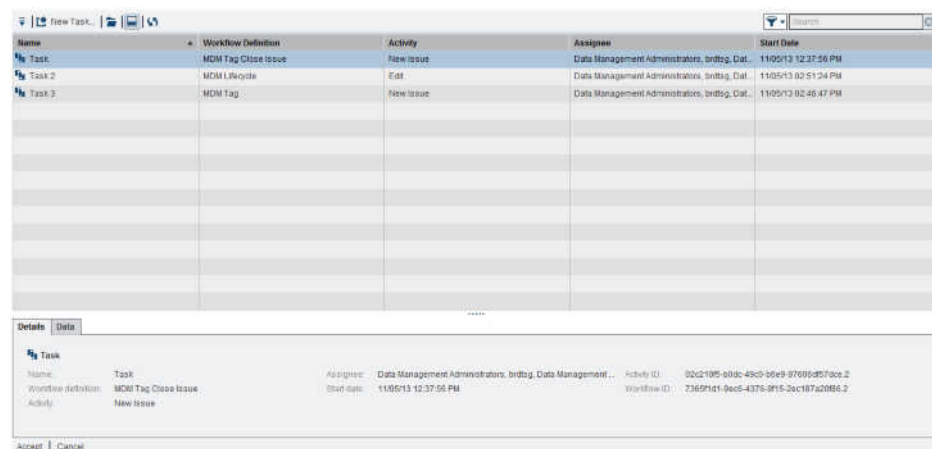
closes the portlet on the SAS Data Management Console.

Note: See “[Specifying SAS MDM SAS Data Management Console Preferences](#)” on [page 50](#) for additional information about specifying portlet display preferences.

When you click an item in the portlet list, the item opens in SAS Task Manager. You can also click **Task Manager** in the portlet or **Task Manager** under Apps to open SAS Task Manager.

SAS Task Manager is displayed in a new tab:

Display 10.2 SAS Task Manager



Name	Workflow Definition	Activity	Assignee	Start Date
Task	MDM Tag Close Issue	New Issue	Data Management Administrators, brtbg, Dat...	11/05/13 12:37:56 PM
Task 2	MDM Lifecycle	Edit	Data Management Administrators, brtbg, Dat...	11/05/13 02:51:24 PM
Task 3	MDM Tag	New Issue	Data Management Administrators, brtbg, Dat...	11/05/13 02:46:47 PM

Task		Assignee		Activity ID	
Name:	Task	Data Management Administrators, brtbg, Data Management ...	Activity ID:	00c2195-6bdc-49c0-b6e9-9769d5f70e.2	
Workflow definition:	MDM Tag Close Issue	Start date:	11/05/13 12:37:56 PM	Workflow ID:	73659d61-9eef-4376-9f16-29c187a20886.2
Activity:	New Issue				

You can perform the following actions from the **SAS Task Manager** tab:

- Search for tasks based on different criteria.
- Edit and close tasks.
- Create new tasks.

The following sections describe these actions in detail.

The SAS Task Manager Tab

Quick Search

When the SAS Task Manager tab opens, it displays a complete list of tasks. You can perform either a quick search or an advanced search to narrow this list to the tasks that you need to view.

To perform a quick search:

1. Enter the search term into the search field in the toolbar:

Figure 10.1 Search Field



As you enter characters in the search field, the search automatically populates the task list with matching tasks.

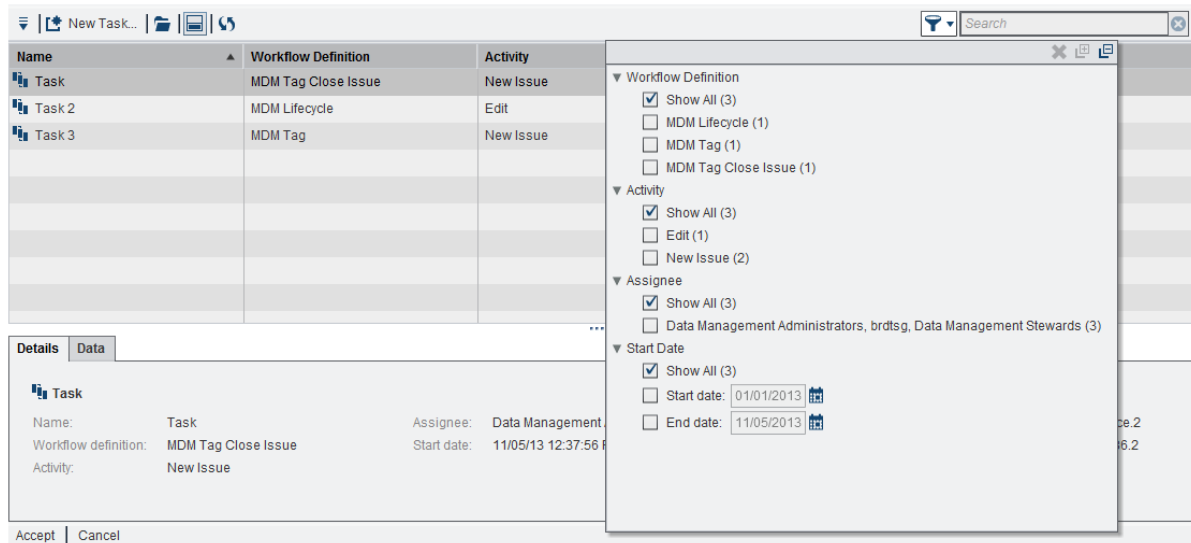
2. Click to clear the search characters and restore the default task list.

Advanced Search

To perform an advanced search:

1. Click to display search criteria:

Figure 10.2 Advanced Search Options



2. As you specify any or all of the search criteria, the task list is modified accordingly.

3. You can do any of the following:

- Click to expand all the search categories.
- Click to collapse all the search categories.
- Click to restore the search criteria to their default settings.
- Click again to close search criteria.

Note: To view the complete task list again, click to restore the search criteria to their default settings and restore the list.

Managing Tasks

The toolbar at the top of the task list contains an **Actions** menu with the following options:

New Task

creates a new task. You must have a role assignment that enables you to perform this action.

Open

opens the task selected in the task list and displays details about it in a new tab.

Terminate

removes the selected task from SAS Task Manager. You must have a role assignment that enables you to perform this action.

Details Pane

displays or hides details of the selected task in a separate pane.

Refresh

refreshes the task list.

The details pane at the bottom of the tab contains two sub-tabs: **Details** and **Data**. The **Details** tab is selected by default. See [the Task tab](#) description for detailed information about the fields on these tabs.

Action buttons might appear below the **Details** and **Data** sub-tabs. These are determined by the workflow definition and might not be a part of your template.

The Task Tab

When you select a task from the task list and click **Open**, the task opens in a new tab.

Figure 10.3 Task Tab

▼ | Approve | Reject | Release | ↺

▼ Details

MARK A MCCOY - INVALID OR MISSING DATA

Name:	MARK A MCCOY - INVALID OR MISSING DATA	Assignee:	Data Management Administrators, Data Management Stewar...	Activity ID:	dd739fce-538b-47d4-9c46-dcc26c023ca8.4
Workflow definition:	MDM Tag	Start date:	11/05/13 03:42:21 PM	Workflow ID:	1205ce8a-be6d-4429-96ad-88428871d03d.4
Activity:	Review Issue				

▼ Data

Comments: Process Invoker: brdtsq Process Title: MARK A MCCOY - INVALID OR MISSING DATA

Instructions: Save the record and approve the issue to confirm changes. ...

▼ SAS MDM: Actions

▼ | Save | ↺

▼ MDM Identifiers

MDM Entity ID:	0
MDM Cluster ID:	0
Source System:	Call Center UK
Source System Record ID:	100012
Valid From:	
Valid To:	

► Contact

► Personal Information

► Address

► Shipping Address

The toolbar contains an **Actions** menu with the following options:

Terminate

removes the selected task from SAS Task Manager. You must have a role assignment that enables you to perform this action.

Refresh

refreshes the task list.

Action buttons might appear in the toolbar. These are determined by the workflow definition and might not be a part of your template.

The **Details** and **Data** panes are comparable to the **Details** and **Data** sub-tabs on the SAS Task Manager tab.

The **Details** pane provides an overview of information related to the selected task:

Name

the name of the selected task.

Workflow definition

the name of the workflow template used for the selected task.

Activity

a value that comes from the workflow definition. It is the current step in the workflow. When you change a status, the **Activity** state changes depending on the design of the workflow.

Assignee

the user or users assigned to the selected task.

Start date

the date on which the task entered SAS Task Manager

Activity ID

identification defined in SAS Workflow Studio.

Workflow ID

identification defined in SAS Workflow Studio.

The values that are displayed on the Data pane depend on the design of your workflow template. The information under **Data** comes from the data objects in the workflow definition. You can control the visibility and editability of the data object values by setting certain attributes on the data objects through SAS Workflow Studio. See [“About SAS Task Manager” on page 141](#) for more information.

If you add or edit values in data object fields that are editable, they are passed into the next step of the workflow when you use one of the actions available on the task.

The following data object values are always provided by the workflow engine:

Process Invoker

the user that sent the task to SAS Task Manager.

Process Title

the name used in SAS Data Remediation for the selected task.

Some steps in the workflow provide a specific interface for user interaction. For example, if the workflow template is so designed, then SAS Task Manager can interact with SAS MDM. In [Figure 10.3](#), this appears as the SAS MDM: Actions pane. The fields displayed in this pane depend on your data source and are shown here only as examples.

After you have reached a terminal step in the workflow, the task no longer appears in SAS Task Manager.

SAS Task Manager Roles

SAS Task Manager creates three new roles in SAS Management Console. These roles are tied to distinct capabilities in the task management application. Depending on the

assigned role of the user, the experience in SAS Task Manager changes. Certain features are available to those assigned one role, but the same features are not available to users in another role.

Here are the roles for SAS Task Manager:

Table 10.4 SAS Task Manager Roles

Role	Default Groups	Capability
Task Manager: Task Administration	Data Management Administrators	<ul style="list-style-type: none"> View Application: The user can view a workflow in SAS Task Manager. Manage Tasks: The user can edit set workflow statuses and edit data object values (if they are set to be editable). Initiate Tasks: The user can start a workflow. Terminate Tasks: The user can cancel active a workflow.
Task Manager: Task Management	Data Management Stewards Data Management Business Approvers	<ul style="list-style-type: none"> View Application: The user can view a workflow in SAS Task Manager. Manage Tasks: The user can edit set workflow statuses and edit data object values (if they are set to be editable). Initiate Task: The user can start a workflow.
Task Manager: Task View	Data Management Business Users	<ul style="list-style-type: none"> View Application: The user can view a workflow in SAS Task Manager.

Deep Linking to Workflows in Task Manager

Deep linking refers to constructing a URL hyperlink that, once initiated, can take a user directly to a named workflow instance in SAS Task Manager rather than requiring the user to search for it. For example, a workflow can send a URL link in an e-mail to a user. When the user clicks the URL link, it opens the SAS Task Manager application and loads the specific workflow instance encoded in the URL.

From within the appropriate environment, create a URL using the following syntax:

```
http://<your_host>/SASDataManagement/  
#workflowID=<workflow_instance_ID>&module=TASKMANAGER
```

For example, you might create the following:

```
http://localhost:8080/SASDataManagement/  
#workflowID=6a12f1cf-813a-42db-90dd-041dde4034f1.103&module=TA  
SKMANAGER
```


Part 3

SAS MDM Examples

Chapter 11

Working with SAS MDM 153

Chapter 11

Working with SAS MDM

Example 1: Creating an Abstract Entity Type	153
Overview of Creating an Abstract Entity Type	153
Open the Entity Type Editor	154
Define Entity Type Properties	154
Define Attributes	155
Define Flux Attribute	158
Define Cluster Conditions	159
Define Relationship Types	160
Generate Jobs	162
Add Match Code Generation to the Standardization Job	163
Publish the Entity Type Definition	165
Example 2: Creating a Concrete Entity Type	166
Overview of Creating a Concrete Entity Type	166
Open the Entity Type Editor	166
Define Entity Type Properties	166
Define Attributes	167
Generate Jobs	170
Add Match Code Generation to the Standardization Job	171
Publish the Entity Type Definition	173
Example 3: Creating Entity Relationships	173
Overview of Creating Entity Relationships	173
Create Entity	173
Create Entity Relationships	176
Explore Entity Relationships	178

Example 1: Creating an Abstract Entity Type

Overview of Creating an Abstract Entity Type

The example presented in this section creates an abstract entity type named Vehicle with VIN (vehicle identification number), MANUFACTURER, MODEL, YEAR, and PLATE attributes, as well as a DF_VIN_MC flux attribute. The example defines a single cluster condition based on the DF_VIN_MC flux attribute, and defines two relationship types: Owner and Driver.

Because Vehicle is modeled as an abstract entity type, instances of this entity type cannot be created. Instead, concrete entity types must be created to use this metadata in the hub. The example presented in [“Example 2: Creating a Concrete Entity Type”](#) on

page 166 creates a concrete entity type named Car, which inherits metadata from the Vehicle entity type.

Open the Entity Type Editor

To open the entity type editor:

1. Click **Data Model** on the SAS Data Management Console page. The **Data Model** tab opens.
2. Click **New Entity Type** on the **Entity Types** toolbar to open the entity type editor. The entity type editor opens on a new tab.

The following sections explain how to complete the fields on the entity type editor **Definition**, **Clustering**, and **Relationship Types** tabs.

Define Entity Type Properties

The definition editor is used to view and modify properties and attributes for an entity type definition.

To use the definition editor:

1. Click the **Definition** tab in the entity type editor to display the **Properties** tab.
2. Enter **VEHICLE** as the entity type in the **Type** field.
3. Enter **Abstract Vehicle type; extend with concrete type** in the **Description** field.
4. Leave the **Parent Type** field set to its default value, **ENTITY**. This value specifies the parent entity type that contributes metadata to the Vehicle entity type.

Note: The **Valid From**, **Last Modified**, **Last Published**, and **Valid To** fields in the **Entity Type Properties** pane are blank. The **Valid From**, **Last Modified**, and **Last Published** fields are automatically completed when you save and publish the entity type definition in later steps. The **Valid To** field remains blank until the entity type definition is retired.

5. Select the **Entity is an abstract entity** check box. Selecting this option indicates that Vehicle is an abstract entity type, meaning that instances of this entity type cannot be created.
6. Enter **vehicle** in the **Display Name** field as the label for the entity type.
7. Enter **1** in the **Sort Order** field. This causes the VEHICLE entity type to appear near the top of the selection lists in the Master Data Management user interface.

Figure 11.1 Definition Tab

The screenshot shows the 'Definition' tab of the entity type editor. On the left, a tree view shows the 'Vehicle' entity type and its attributes: MDM_ENTITY_TYPE_ID, MDM_VALID_TO_DTTM, MDM_VALID_FROM_DTTM, MDM_LANGUAGE_ID, MDM_SRC_SYS_REC_ID, MDM_SRC_SYS_ID, MDM_ENTITY_CLUSTER_ID, and MDM_ENTITY_ID. The main panel is divided into 'Properties' and 'Permissions' tabs. The 'Properties' tab is active, showing 'Entity Type Properties' and 'Display Settings'.

Entity Type Properties:

- Type: * VEHICLE
- Description: Abstract Vehicle type; extend with concrete type
- Parent Type: ENTITY
- Valid From:
- Last Modified:
- Last Published:
- Valid To:
- ☒ Entity is an abstract entity (cannot be instantiated)

Display Settings:

- Display name: * Vehicle
- Sort Order: * 1
- Attribute to use as entity label:

- Click **Save** on the entity type editor toolbar to save the changes.

Define Attributes

Use the following task to add five new attributes to the Vehicle entity type: VIN, MANUFACTURER, MODEL, YEAR, and PLATE.

To define attributes:

- Click **New Attribute** on the toolbar to add a new attribute. The new attribute appears in the attribute list on the left side of the **Definition** tab.

Figure 11.2 New Attribute

The screenshot shows the 'Definition' tab of the entity type editor. On the left, the 'Vehicle' entity type is selected, and a new attribute named 'ATTRIBUTE' has been added to the list. The main panel is divided into 'Properties' and 'Permissions' tabs. The 'Properties' tab is active, showing 'Attribute Type Properties' and 'Display Settings'.

Attribute Type Properties:

- Type: * ATTRIBUTE
- Group:
- Description:
- Data Type: STRING
- Length: * 50
- Constraint:
- Valid From:
- Valid To:
- ☐ Attribute is required
- ☐ Attribute is read only
- ☐ Encrypt attribute

Display Settings:

- Display name: * ATTRIBUTE
- Search form order: * 0
- Search results order: * 0
- Details panel order: * 0
- Cluster table order: * 0
- Record editor order: * 0

Note:

- The **Length** field is not applicable for INTEGER attributes.
 - The **Valid From** and **Valid To** fields are blank. The **Valid From** field is automatically completed when you save and publish the entity type definition in later steps. The **Valid To** field remains blank until the entity type definition is retired.
2. Enter data from the Attribute 1 Value column of the following table in the appropriate locations on the **Properties** tab.

Note:

- For check box options, **Yes** means select the check box. **No** means clear the check box.
- The **Search form order** values for the five new attributes are lower than 100, which is the lowest **Search form order** value for attributes that are inherited from the ENTITY (top-level) entity type. This causes the five new attributes to appear above the MDM Identifiers group containing the inherited attributes in the search fields on the **Master Data Management** tab.

Table 11.1 New Attributes

Attribute Option	Attribute 1 Value	Attribute 2 Value	Attribute 3 Value	Attribute 4 Value	Attribute 5 Value
Type	VIN	MANUFACTURER	MODEL	YEAR	PLATE
Group	Vehicle Information	Vehicle Information	Vehicle Information	Vehicle Information	Vehicle Information
Description	Vehicle Identification Number	Vehicle manufacturer	Vehicle model designation	Year of manufacture	Current plate number
Data Type	String	String	String	Integer	String
Length	50	50	50	none	10
Constraint	(leave blank)	(leave blank)	(leave blank)	(leave blank)	(leave blank)
Required	No	No	No	No	No
Read-only	No	No	No	No	No
Encrypt	No	No	No	No	No
Display name	Vehicle Identification Number	Manufacturer	Model	Year	Plate
Search form order	90	92	94	96	98
Search results order	90	92	94	96	98

Attribute Option	Attribute 1 Value	Attribute 2 Value	Attribute 3 Value	Attribute 4 Value	Attribute 5 Value
Details panel order	90	92	94	96	98
Cluster table order	90	92	94	96	98
Record editor order	90	92	94	96	98

Figure 11.3 VIN Attribute

The screenshot shows the 'Definition' tab of the entity type editor. On the left, a tree view shows the 'Vehicle' entity with several attributes listed, including 'VIN' which is selected. The main area displays the 'Properties' tab for the 'VIN' attribute. The 'Attribute Type Properties' section includes fields for 'Type' (set to 'VIN'), 'Group' (set to 'Vehicle Information'), 'Description' (set to 'Vehicle Identification Number'), 'Data Type' (set to 'STRING'), 'Length' (set to '50'), 'Constraint', 'Valid From', and 'Valid To'. Below these are checkboxes for 'Attribute is required', 'Attribute is read only', and 'Encrypt attribute'. The 'Display Settings' section includes fields for 'Display name' (set to 'Vehicle Identification Number'), 'Search form order' (set to '90'), 'Search results order' (set to '90'), 'Details panel order' (set to '90'), 'Cluster table order' (set to '90'), and 'Record editor order' (set to '90').

3. Click **Save** on the entity type editor toolbar to save the new VIN attribute. The VIN attribute now appears in the attributes list.
4. Repeat this task to add each of the four remaining new attributes specified in the table (MANUFACTURER, MODEL, YEAR, and PLATE).
5. Select Vehicle at the top of the **Definition** tab, which displays the properties of Vehicle on the **Properties** tab.
6. Select VIN from the **Attribute to use as entity label** drop-down list.

Figure 11.4 Label Attribute

The screenshot shows the 'Entity Type Editor' interface. On the left, a tree view lists attributes for the 'Vehicle' entity type, including MDM_ENTITY_TYPE_ID, MDM_VALID_TO_DTTM, MDM_VALID_FROM_DTTM, MDM_LANGUAGE_ID, MDM_SRC_SYS_REC_ID, MDM_SRC_SYS_ID, MDM_ENTITY_CLUSTER_ID, MDM_ENTITY_ID, VIN, MANUFACTURER, MODEL, YEAR, and PLATE. The 'Properties' tab is active, showing 'Entity Type Properties' and 'Display Settings'. Under 'Entity Type Properties', the 'Type' is 'VEHICLE', the 'Description' is 'Abstract Vehicle type; extend with concrete type', the 'Parent Type' is 'ENTITY', and the 'Valid From' date is '08/29/13 11:51:56 AM'. A checkbox 'Entity is an abstract entity (cannot be instantiated)' is checked. Under 'Display Settings', the 'Display name' is 'Vehicle', the 'Sort Order' is '1', and the 'Attribute to use as entity label' is 'VIN'. A dropdown menu is open for the 'Attribute to use as entity label', showing a list of attributes with 'VIN' selected.

Note: Only one attribute can be designated as a label attribute in an entity type definition. The label attribute is inherited from the parent entity type by derived entity types. A label attribute is always required.

- 7. Click **Save** on the entity type editor toolbar.

Define Flux Attribute

A flux attribute, based on the VIN attribute, is used for field mapping in the standardization job for the Vehicle entity type (mdm_std_vehicle.ddf).

To define a flux attribute:

- 1. Select the VIN attribute in the attributes list of the attribute editor. On the attribute editor toolbar, click and select **New Flux Attribute** from the menu. A flux attribute named DF_VIN_MC appears in the attributes list. SAS MDM automatically assigns the attribute name by adding a 'DF_' prefix and a '_MC' (match code) suffix to the original attribute name.
- 2. Enter data from the following table in the appropriate locations on the **Properties** tab.

Table 11.2 Flux Attribute

Attribute Option	Attribute Value
Type	DF_VIN_MC
Group	Vehicle Information
Description	VIN Match Code

Attribute Option	Attribute Value
Data Type	String
Length	40
Constraint	(leave blank)
Required	No
Read-only	No
Encrypt	No
Display name	VIN Match Code
Search form order	0
Search results order	0
Details panel order	0
Cluster table order	0
Records editor order	0

3. Click **Save** on the entity type editor toolbar to save the changes.

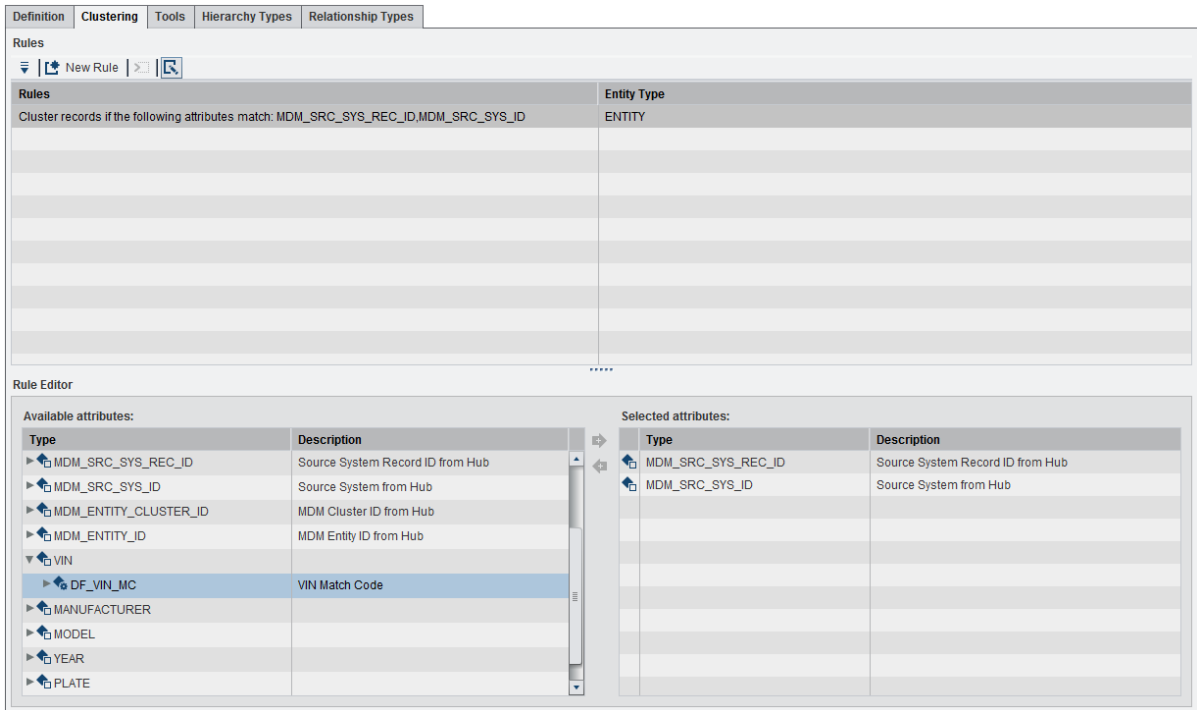
Define Cluster Conditions


The clustering editor is used to create and modify cluster conditions for an entity type definition. Records with the same attribute values specified in the cluster condition are clustered together. A default cluster condition is inherited from the ENTITY (top-level) entity type. The five previously added attributes appear in the **Available** list, as well as the flux attribute (DF_VIN_MC).

To use the clustering editor:

1. Click the **Clustering** tab in the entity type editor to display the clustering editor.

Figure 11.5 Clustering Tab



- 2. Click **New Rule** on the clustering editor toolbar. A new entry, **<No attributes in rule>**, appears in the **Rules** list.
- 3. In the **Available** list, select the **DF_VIN_MC** flux attribute. Then click  to transfer **DF_VIN_MC** to the **Selected** list.
- 4. Click **Save** on the entity type editor toolbar to save the changes. The flux attribute now appears in the **Rules** list.

Define Relationship Types

The relationship type editor is used to create and modify relationship types for an entity type definition. Relationship types describe the types of relationships that can exist between the associated entity types. Relationship types define source and target entity types and have optional match conditions that can be used to determine whether two entities are related.

To use the relationship type editor:

- 1. Click the **Relationship Types** tab in the entity types editor.
- 2. Click **New Relationship Type** on the relationship type toolbar to add a new relationship type. The relationship type editor opens on a new tab.

Figure 11.6 Relationship Properties Tab

PropertiesRules

Relationship Properties

Type: * New Relationship Type

Description:

Source Entity Type: *

Target Entity Type: *

Valid From:

Valid To:

Display Settings

Display Name: * New Relationship Type

Inverse Display Name: * New Relationship Type

Sort Order: * 30

Inverse Sort Order: * 30

Note: The **Valid From** and **Valid To** fields are blank. The **Valid From** field is automatically completed when you save and publish the entity type definition in later steps. The **Valid To** field remains blank until the entity type definition is retired.

- Enter data from the Relationship Type 1 Value column of the following table in the appropriate locations on the tab.

Table 11.3 Relationship Values

Relationship Type Editor Field	Relationship Type 1 Value	Relationship Type 2 Value
Type	DRIVER	OWNER
Description	Used to associate people with vehicles that they drive	Used to associate people with vehicles that they own
Source Entity Type	VEHICLE	VEHICLE
Target Entity Type	INDIVIDUAL	PARTY
Display Name	Driven By	Owned By
Inverse Display Name	Drives	Owns
Sort Order	10	20

Relationship Type Editor Field	Relationship Type 1 Value	Relationship Type 2 Value
Inverse Sort Order	10	20

Figure 11.7 Relationship Properties Tab

Properties Rules

▼ Relationship Properties

Type: * DRIVER

Description: Used to associate people with vehicles that they drive

Source Entity Type: * VEHICLE

Target Entity Type: * INDIVIDUAL

Valid From: 07/30/13 04:05:25 PM

Valid To:

▼ Display Settings

Display name: * Driven By

Inverse Display Name: * Drives

Sort Order: * 10

Inverse Sort Order: * 10

Note: The **Target Entity Type** for the DRIVER relationship type is INDIVIDUAL, because only a person can drive a vehicle.

- Click **Save** on the entity type editor toolbar to save the new DRIVER relationship.
- Repeat this task to add the new OWNER relationship.

Note: The **Target Entity Type** for the OWNER relationship type is PARTY. Both INDIVIDUAL and COMPANY are concrete subtypes of PARTY, because both individuals and companies can own a vehicle.

Generate Jobs

After you have created and saved the Vehicle entity type definition, you generate default batch jobs and data services in support of the entity type.


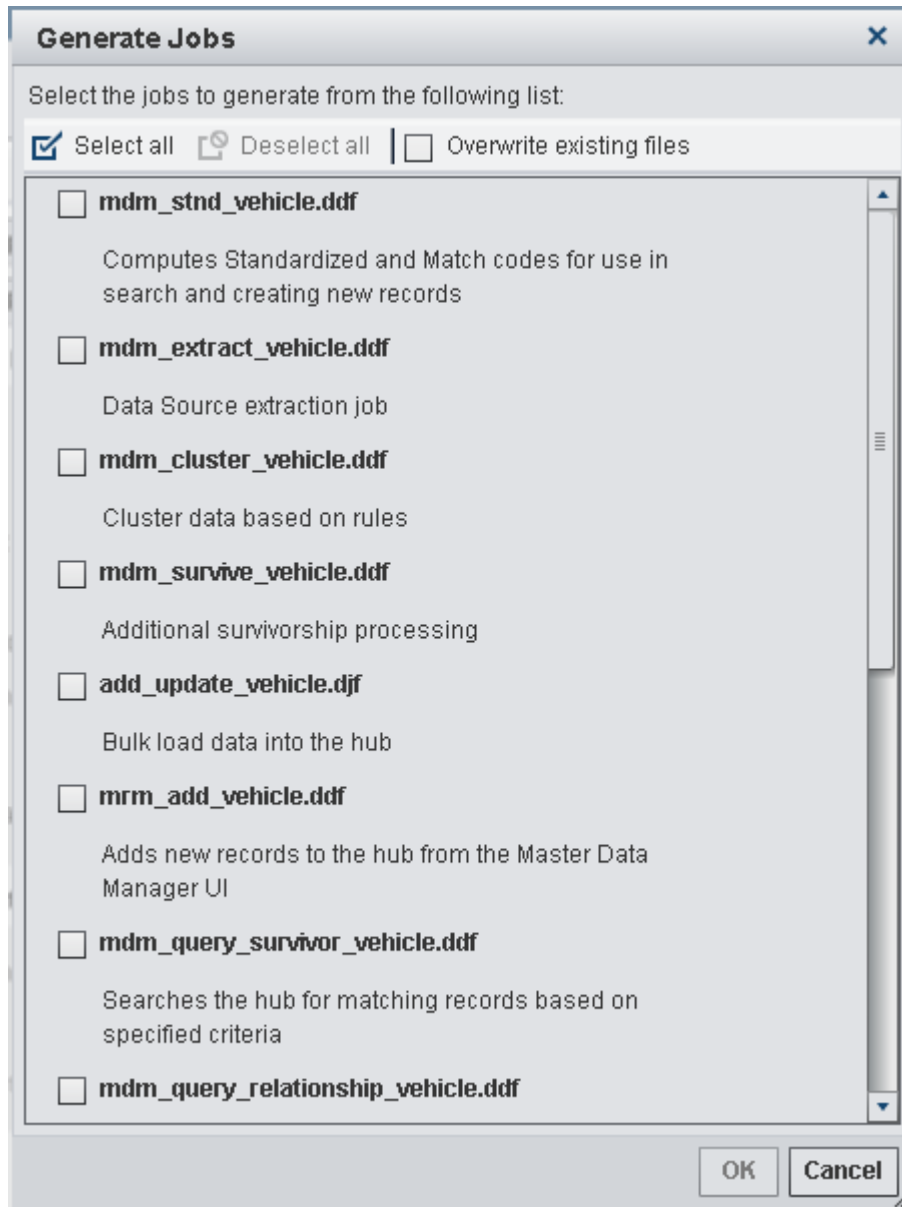
- Click  on the toolbar of the entity type editor. The Generate Jobs dialog box appears:

Figure 11.8 Generate Jobs Dialog Box



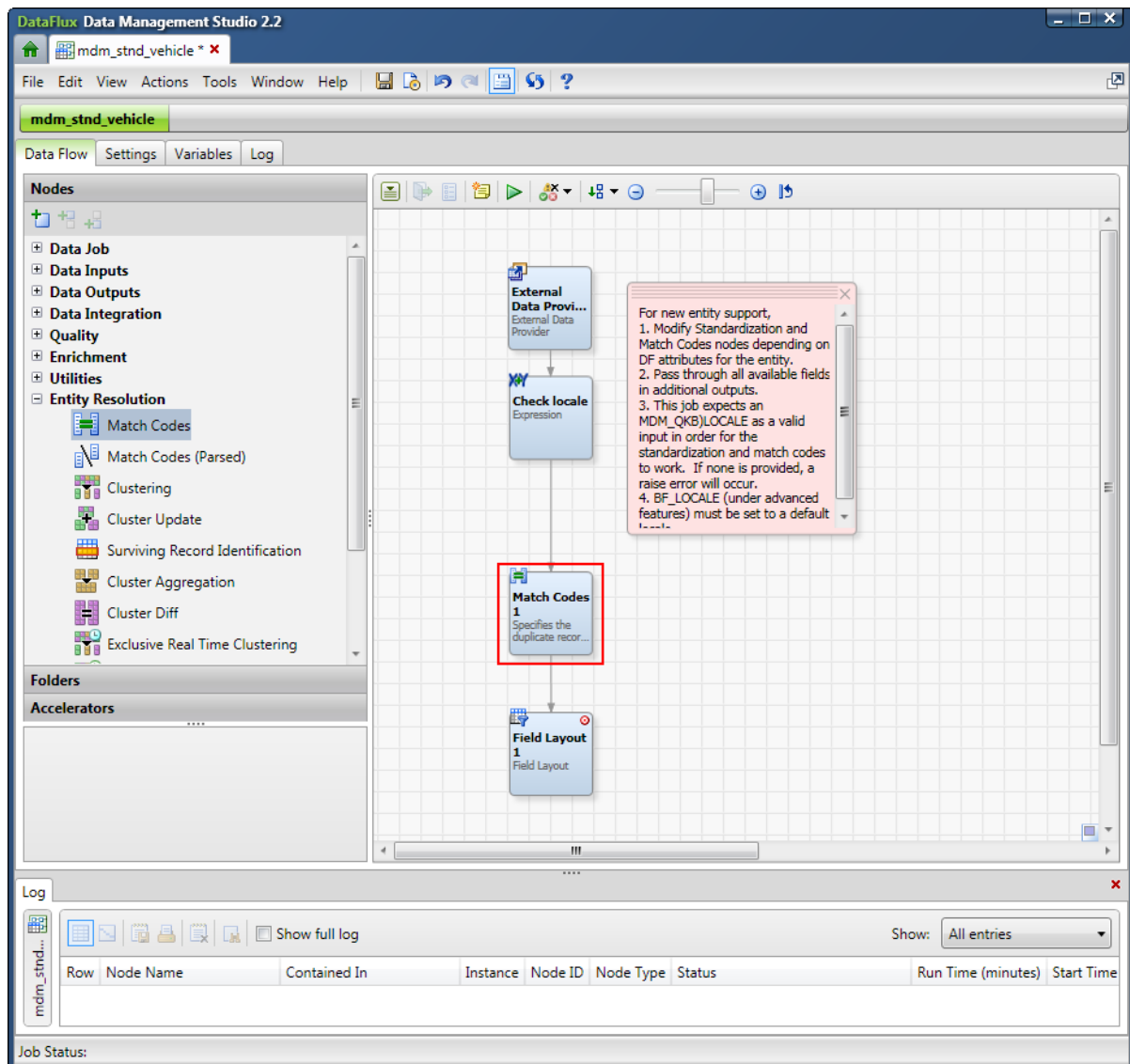
2. Select all three available jobs, `mdm_std_vehicle.ddf`, `mdm_query_survivor_vehicle.ddf`, and `mdm_query_relationship_vehicle.ddf`, in the Generate Jobs dialog box. Notice that several of the listed jobs are dimmed because these jobs are not required for abstract entity types.
3. Click **OK** to generate the jobs. The job generation process uses the metadata that has been saved in the hub and publishes the jobs to your DataFlux Data Management Server.

Add Match Code Generation to the Standardization Job

The steps in this section add match code generation to the standardization job for the Vehicle entity type (`mdm_std_vehicle.ddf`).

1. Select and log on from the DataFlux Data Management Server **Data Management Servers** riser in DataFlux Data Management Studio. Expand the **Real-Time Data Services** folder and the **SAS MDM** folder, and locate the `mdm_stnd_vehicle.ddf` job.
2. Right-click the `mdm_stnd_vehicle.ddf` job and select **Export**. From the Export To Folder dialog box, export the job to your SAS MDM repository folder.
3. Locate and open the `mdm_stnd_vehicle.ddf` job in the SAS MDM repository folder on the **Folders** riser in DataFlux Data Management Studio.
4. Add a **Match Codes** node between the **Check locale** node and the **Field Layout 1** node. The **Match Codes** node generates match codes for the selected fields when the job is run. Match codes enable fuzzy matching of fields.

Figure 11.9 Match Codes Node



5. Double-click the **Match Codes** node to display its properties dialog box. In the properties dialog box, do the following:
 - a. Double-click the VIN match code field in the **Available** list to transfer it to the **Selected** list.
 - b. Select **Text** in the **Definition** column.

- c. Leave the **Sensitivity** value set to its default value of 85.
- d. Enter **DF_VIN_MC** in the **Output Name** column.
- e. Select the **Generate null match codes for blank field values** check box. Selecting this option causes null match codes to be created for null or blank input field values. This prevents null or blank field values from being clustered together.

Figure 11.10 Match Codes Properties

Match Codes Properties

Name: Match Codes 1 Notes...

Locale: English (United States)

☐ Allow generation of multiple matchcodes per definition for each sensitivity

Match code fields

Available:

- vehide_offset
- MDM_QKB_LOCALE
- MDM_ENTITY_CLUSTER_ID
- MDM_ENTITY_ID
- MDM_ENTITY_TYPE_ID
- MDM_SRC_SYS_ID
- MDM_SRC_SYS_REC_ID
- MDM_VALID_FROM_DTTM
- MDM_VALID_TO_DTTM
- MANUFACTURER
- MODEL
- PLATE
- VIN**
- YEAR

Selected:

Field Name	Definition	Sensitivity	Output Name
VIN	Text	85	DF_VIN_MC

☒ Generate null match codes for blank field values

☐ Preserve null values

Additional Outputs...

OK Cancel Help


6. Save and close the mdm_std_vehicle.ddf job.
7. On the **Data Management Servers** riser, right-click the **Real-Time Data Services** folder and select **Import**. Import the modified mdm_std_vehicle.ddf job from your SAS MDM repository folder to the DataFlux Data Management Server.

Publish the Entity Type Definition

The final step is to publish the Vehicle entity type definition. Publishing the entity type definition makes the definition active. This includes generating the tables used to store

the data in your DBMS, as well as updating the metadata that Master Data Management references when determining which entity types are presented in the user interface.

To publish the definition:

1. Click  on the toolbar of the entity type editor. The Publish dialog box appears.
2. Click **OK** in the confirmation dialog box to publish the definition.

Example 2: Creating a Concrete Entity Type

Overview of Creating a Concrete Entity Type

The example presented in this section builds on the Vehicle example described in “[Example 1: Creating an Abstract Entity Type](#)”. Because the Vehicle entity type is modeled as abstract, instances of this entity type cannot be created. Instead, concrete entity types must be created from this metadata in the hub. This example creates a concrete entity type named Car, which inherits metadata from the Vehicle entity type.

Open the Entity Type Editor

To open the entity type editor:

1. Click **Data Model** on the SAS Data Management Console page. The **Data Model** tab opens.
2. Click **New Entity Type** on the **Entity Types** toolbar to open the entity type editor. The entity type editor opens on a new tab.

Define Entity Type Properties

To use the definition editor:

1. Click the **Definition** tab in the entity type editor to display the **Properties** tab.
2. Enter **CAR** as the entity type in the **Type** field.
3. Enter **Automobile** in the **Description** field.
4. Select **VEHICLE** in the **Parent Type** field. This value specifies the parent entity type that contributes metadata to the Car entity type.

Note: The **Valid From**, **Last Modified**, **Last Published**, and **Valid To** fields in the **Entity Type Properties** pane are blank. The **Valid From**, **Last Modified**, and **Last Published** fields are automatically completed when you save and publish the entity type definition in later steps. The **Valid To** field remains blank until the entity type definition is retired.

5. Clear the **Entity is an abstract entity** check box.
6. Enter **Car** in the **Display Name** field as the label for the entity type.
7. Enter **2** in the **Sort Order** field. This causes the CAR entity type to appear after the VEHICLE entity type in the selection lists in the Master Data Management user interface.

Figure 11.11 Definition Tab

The screenshot shows the 'Definition Tab' of a software interface. On the left, a tree view lists attributes under the 'Car' entity type, including MDM_ENTITY_TYPE_ID, MDM_VALID_TO_DTTM, MDM_VALID_FROM_DTTM, MDM_LANGUAGE_ID, MDM_SRC_SYS_REC_ID, MDM_SRC_SYS_ID, MDM_ENTITY_CLUSTER_ID, MDM_ENTITY_ID, VIN, MANUFACTURER, MODEL, YEAR, and PLATE. On the right, the 'Properties' tab is active, showing 'Entity Type Properties' and 'Display Settings'. The 'Entity Type Properties' section includes fields for Type (CAR), Description (Automobile), Parent Type (VEHICLE), Valid From, Last Modified, Last Published, and Valid To. There is also a checkbox for 'Entity is an abstract entity (cannot be instantiated)'. The 'Display Settings' section includes fields for Display name (Car), Sort Order (2), and Attribute to use as entity label (VIN).

8. Click **Save** on the entity type editor toolbar to save the changes.

Define Attributes

Use the following task to add a Capacity attribute to the Car entity type.

To define the attribute:

1. Click **New Attribute** on the toolbar to add a new attribute. The new attribute appears in the attribute list on the left side of the **Definition** tab.

Figure 11.12 New Attribute

The screenshot shows the 'New Attribute' dialog in SAS MDM. The left pane shows a tree structure under 'Car' with various attributes, and 'ATTRIBUTE' is selected. The right pane has two tabs: 'Properties' and 'Permissions'. The 'Properties' tab is active and contains two sections: 'Attribute Type Properties' and 'Display Settings'. In 'Attribute Type Properties', 'Type' is set to 'ATTRIBUTE', 'Group' is empty, 'Description' is empty, 'Data Type' is 'STRING', 'Length' is '50', 'Constraint' is empty, 'Valid From' is empty, and 'Valid To' is empty. There are three unchecked checkboxes: 'Attribute is required', 'Attribute is read only', and 'Encrypt attribute'. In 'Display Settings', 'Display name' is 'ATTRIBUTE', and 'Search form order', 'Search results order', 'Details panel order', 'Cluster table order', and 'Record editor order' are all set to '0'.

Note:

- The **Length** field is not applicable for INTEGER attributes.
- The **Valid From** and **Valid To** fields are blank. The **Valid From** field is automatically completed when you save and publish the entity type definition in later steps. The **Valid To** field remains blank until the entity type definition is retired.

2. Enter data from the following table in the appropriate locations on the **Properties** tab.

Note:

- For check box options, **Yes** means select the check box. **No** means clear the check box.
- The **Search Form Order** value for the new attribute is lower than 100, which is the lowest **Search Form Order** value for attributes that are inherited from the ENTITY (top-level) entity type. The value 98 causes the Capacity attribute to appear immediately after the Plate attribute in the Search fields on the **Master Data Management** tab.

Table 11.4 Capacity Attribute

Attribute Option	Attribute Value
Type	CAPACITY
Group	Vehicle Information
Description	Number of Passengers
Data Type	Integer

Attribute Option	Attribute Value
Length	(leave blank)
Constraint	(leave blank)
Required	No
Read-Only	No
Encrypt	No
Display name	Number of Passengers
Search form order	99
Search results order	99
Details panel order	99
Cluster table order	99
Record editor order	99

- Click **Save** on the entity type editor toolbar to save the new CAPACITY attribute. The CAPACITY attribute now appears in the attributes list.

Figure 11.13 Capacity Attribute

The screenshot shows the 'Definition' tab of the entity type editor. On the left, a tree view lists attributes under the 'Car' entity, with 'CAPACITY' selected. The main panel shows the 'Properties' tab for the 'CAPACITY' attribute. The 'Attribute Type Properties' section includes fields for Type (CAPACITY), Group (Vehicle Information), Description (Number of Passengers), Data Type (INTEGER), Length (0), Constraint, Valid From, and Valid To. Below these are checkboxes for 'Attribute is required', 'Attribute is read only', and 'Encrypt attribute'. The 'Display Settings' section includes fields for Display name (Number of Passengers), Search form order (99), Search results order (99), Details panel order (99), Cluster table order (99), and Record editor order (99).

Generate Jobs

After you have created and saved the CAR entity type definition, generate default batch jobs and data services in support of the entity type:


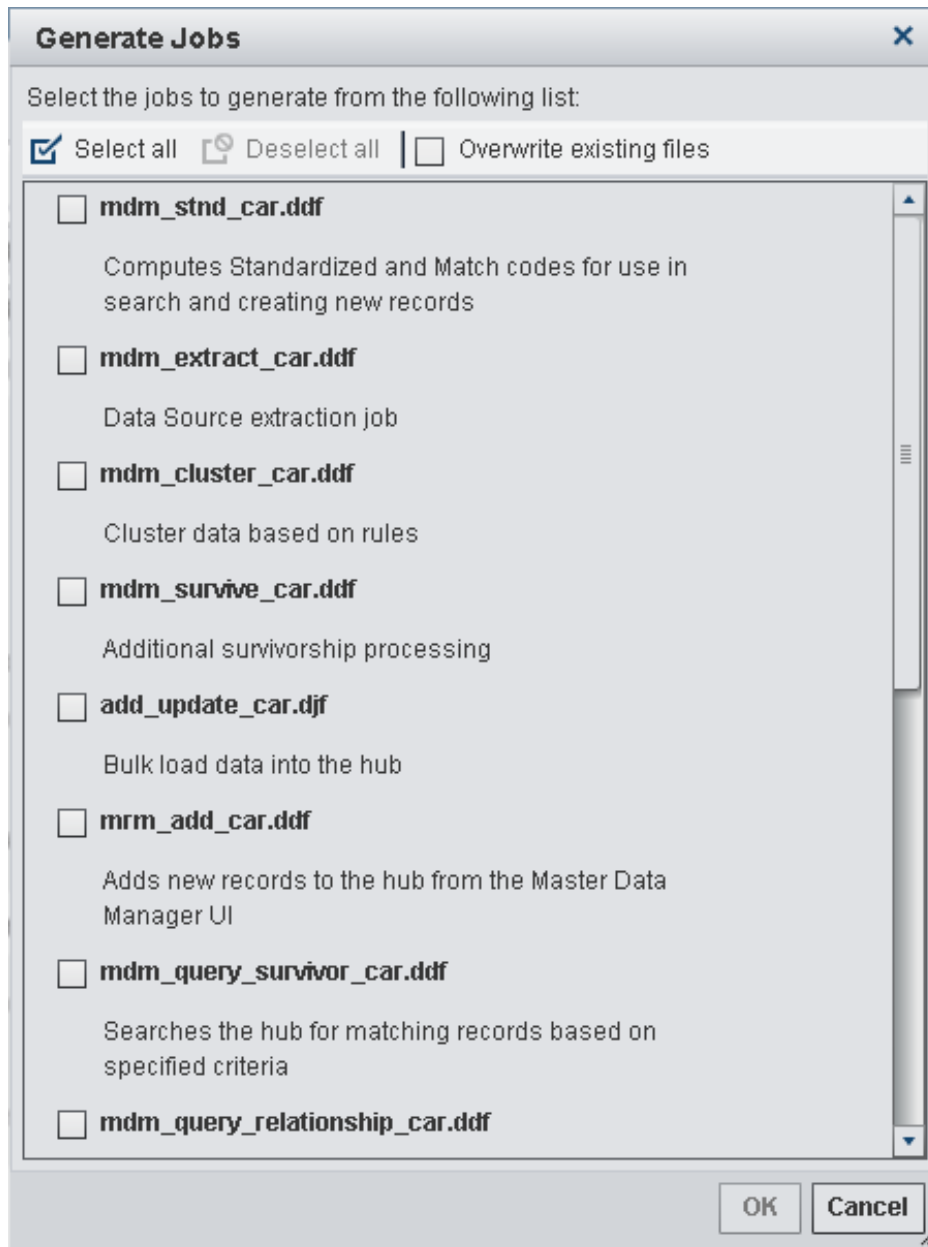
1. Click  on the toolbar of the entity type editor. The Generate Jobs dialog box appears:

Figure 11.14 Generate Jobs Dialog Box



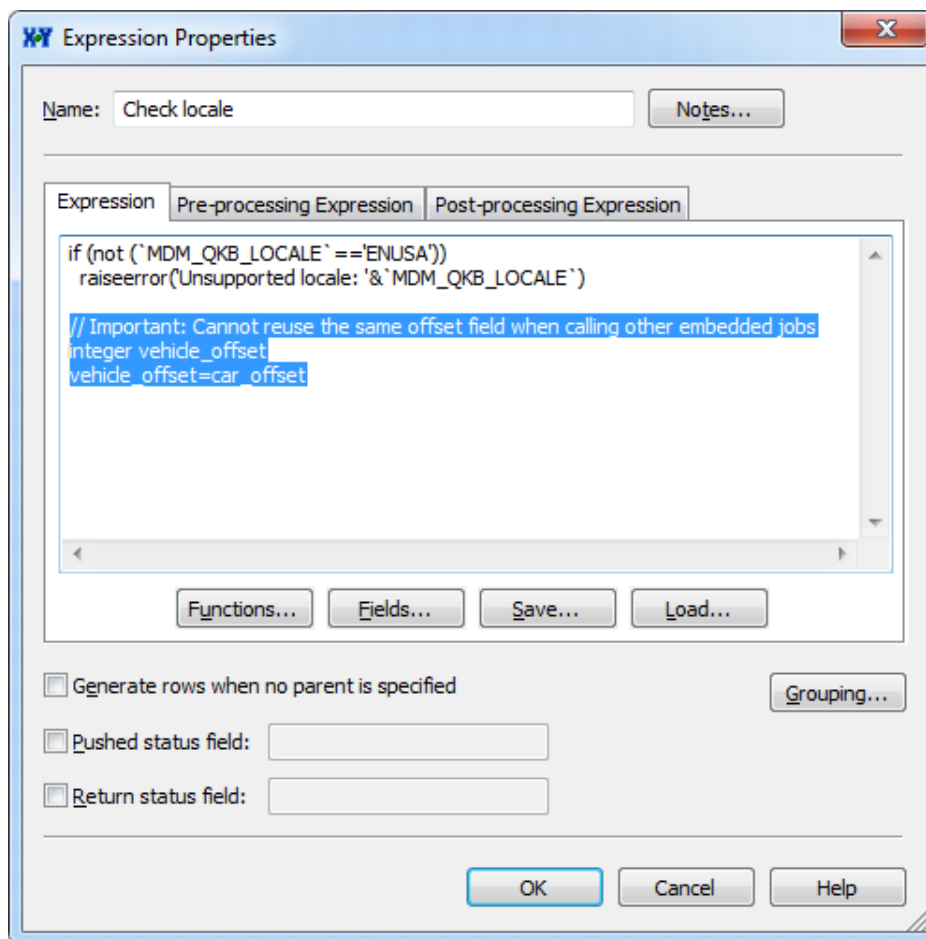
2. Select **Check All** in the Generate Jobs dialog box.
3. Click **OK** to generate the jobs. The job generation process uses the metadata that has been saved in the hub and publishes the jobs to your DataFlux Data Management Server.

Add Match Code Generation to the Standardization Job

The steps in this section modify the `mdm_std_car.ddf` standardization job to reference the `mdm_std_vehicle.ddf` standardization job so that the CAR entity type uses the standardization defined by the Vehicle entity type.

1. Select and log on to the DataFlux Data Management Server from the **Data Management Servers** riser in DataFlux Data Management Studio. Expand the **Real-Time Data Services** folder and the **SAS MDM** folder, and locate the `mdm_std_car.ddf` job.
2. Right-click the `mdm_std_car.ddf` job and select **Export**. From the Export To Folder dialog box, export the job to your SAS MDM repository folder.
3. Locate and open the `mdm_std_car.ddf` job in the SAS MDM repository folder on the **Folders** riser in DataFlux Data Management Studio.
4. Double-click the **Check Locale** node to display its properties dialog box. On the **Expression** tab, declare an integer variable named 'vehicle_offset', and set the variable to 'car_offset'.

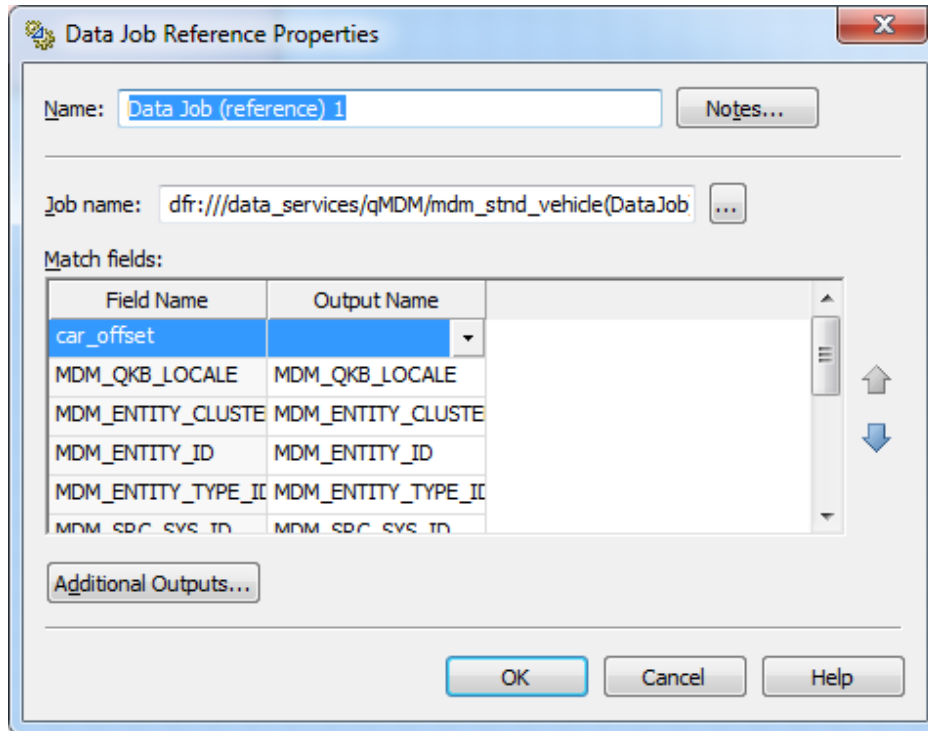
Figure 11.15 Check Locale Node



5. Click **OK** to save the changes.
6. Add a **Data Job Reference** node between the **Check Locale** node and the **Field Layout 1** node. The **Data Job Reference** node uses the `mdm_std_vehicle.ddf` job to compute the VIN match code.

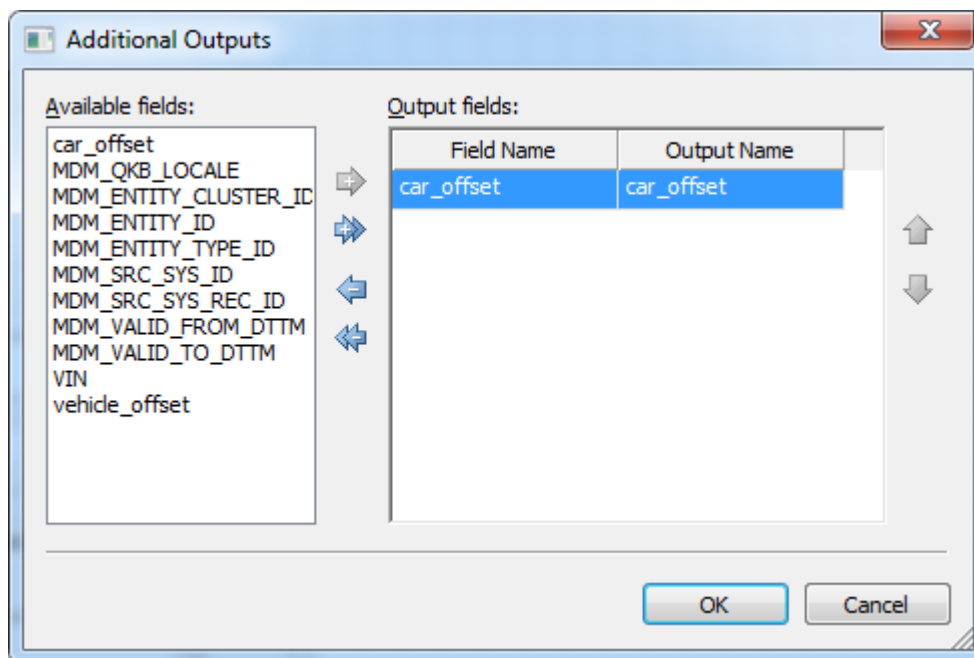
- Double-click the **Data Job Reference** node to display its Properties dialog box. In the Properties dialog box, match each field with its corresponding parameter in the `mdm_std_vehicle.ddf` standardization job. Note that the `car_offset` field does not have a corresponding parameter.

Figure 11.16 Data Job Reference Properties



- Click **Additional Outputs** to display the Additional Outputs dialog box.

Figure 11.17 Additional Outputs




- In the Additional Outputs dialog box, select the `car_offset` field as an additional field to be returned from the `mdm_std_car.ddf` standardization job.

10. Click **OK** to save the changes.
11. Click **OK** to close the Properties dialog box for the **Data Job Reference** node.
12. Save and close the mdm_stnd_car.ddf job.
13. On the **Data Management Servers** riser, right-click the **Real-Time Data Services** folder and select **Import**. Import the modified mdm_stnd_car.ddf job from your SAS MDM repository folder to the DataFlux Data Management Server.

Publish the Entity Type Definition

The final step is to publish the CAR entity type definition. Publishing the entity type definition makes the definition active. This includes generating the tables used to store the data in your DBMS, as well as updating the metadata that Master Data Management references when determining which entity types are presented in the user interface.

To publish the definition:

1. Click  on the toolbar of the entity type editor. The Publish dialog box appears.
2. Click **OK** in the confirmation dialog box to publish the definition.

Because CAR has been defined as a concrete entity type, new instances of this entity type can be created.

Example 3: Creating Entity Relationships

Overview of Creating Entity Relationships

The example presented in this section demonstrates the following:

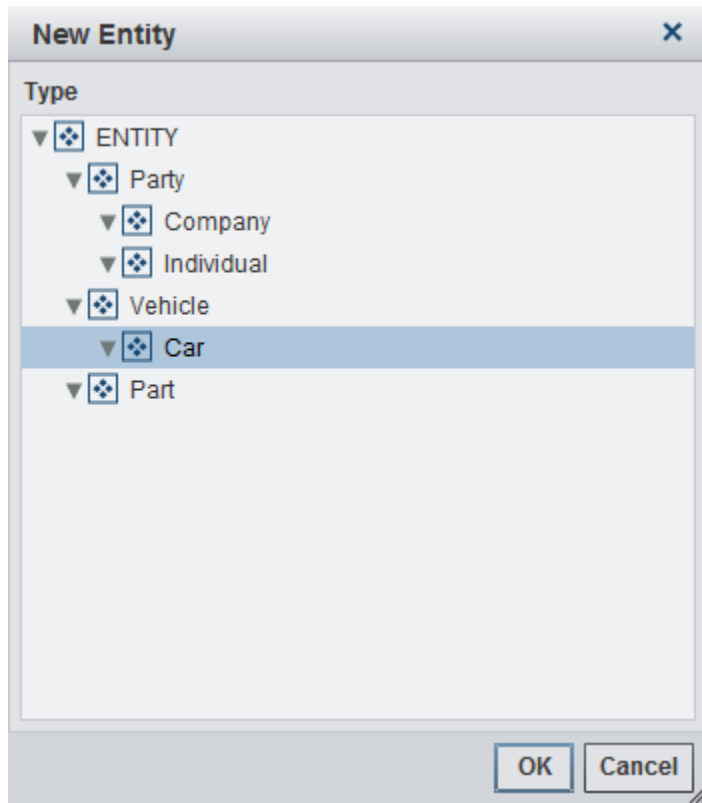
- creating an instance of the Car entity type created in [“Example 2: Creating a Concrete Entity Type” on page 166](#)
- creating Owned By and Driven By entity relationships for the entity
- exploring the Owned By and Driven By entity relationships graphically

The example also illustrates the inheritance of presentation metadata specified for the Car entity type and for the Vehicle entity type created in [“Example 1: Creating an Abstract Entity Type” on page 153](#).

Create Entity

To create an entity:

1. Click **Master Data Management** on the SAS Data Management Console page. The **Master Data Management** tab opens.
2. Click **New Entity** on the toolbar to open the New Entity dialog box:

Figure 11.18 New Entity Dialog Box

3. Select the Car entity type in the New Entity dialog box.
4. Click **OK** to open the entity editor.

Figure 11.19 New Entity Editor

The entity editor contains a **Vehicle Information** pane. This name of this pane is determined by the **Group** values of the five attributes created for the VEHICLE entity type. The **Vehicle Information** pane contains Vehicle Identification Number, Manufacturer, Model, Year, and Plate fields. These field labels are based on the **Label** values of the five attributes created for the VEHICLE entity type. These

attributes are inherited by the CAR entity type. The **Vehicle Information** pane also contains a **Number of Passengers** field. This field label is based on the **Label** value of the Capacity attribute created for the CAR entity type.

5. In the entity editor, enter values from the following table for the fields in the **Vehicle Information** pane.

Table 11.5 Vehicle Information Values

Field	Value
Vehicle Identification Number	A94875
Manufacturer	Acme Car Company
Model	FastRide GT
Year	2012
Plate	ABC-1234
Number of Passengers	4

Figure 11.20 New Entity Editor

The screenshot shows the 'New Entity Editor' interface. At the top is a toolbar with a 'Save' button and a 'Cancel' button. Below the toolbar are two expandable panes. The first pane, 'MDM Identifiers', contains fields for 'MDM Entity ID:', 'MDM Cluster ID:', 'Source System:' (with a dropdown menu showing 'MD Manager'), 'Source System Record ID:', 'Valid From:', and 'Valid To:'. The second pane, 'Vehicle Information', contains input fields for 'Vehicle Identification Number:' (with a red asterisk indicating it's required), 'Manufacturer:', 'Model:', 'Year:', 'Plate:', and 'Number of Passengers:'. Each field has a corresponding text input box with the values from Table 11.5 entered: A94875, Acme Car Company, FastRide GT, 2012, ABC-1234, and 4.

6. Click **Save** on the toolbar to save the changes. Two new panes, **Driven By** and **Owned By**, appear below the **Vehicle Information** pane:

Figure 11.21 New Entity Editor with Relationship Panes

The screenshot shows the 'New Entity Editor' interface. At the top is a toolbar with icons for Save, Matrix Edit, and other functions. Below the toolbar are two main panes:

- MDM Identifiers**: This pane contains the following fields:
 - MDM Entity ID: 21787
 - MDM Cluster ID: 6383
 - Source System: Best Record
 - Source System Record ID: 21787
 - Valid From: 07/31/13 11:32:49 AM
 - Valid To:
- Vehicle Information**: This pane contains the following fields:
 - Vehicle Identification Number: * A94875
 - Manufacturer: Acme Car Company
 - Model: FastRide GT
 - Year: 2012
 - Plate: ABC-1234
 - Number of Passengers: 4

Below the Vehicle Information pane are two expandable panes:

- ▶ Driven By
- ▶ Owned By

Create Entity Relationships

The **Driven By** and **Owned By** panes correspond to the Driver and Owner relationship types created for the VEHICLE entity type. You can search for and select entities among which you want to create relationships.

To create entity relationships:


1. Click  to expand the **Driven By** pane.
2. Click **New Relationship Type** on the toolbar to display the New Relationship dialog box:

Figure 11.23 New Relationship Dialog Box

Save | Matrix Edit

MDM Identifiers

Vehicle Information

Driven By

New Relationship

MDM Entity Type	MDM Cluster ID	Full Name	Do not Call	Last Contact	Address Line 1 (Standardized)
Individual	282	ABBEY CATA	No	01/14/11 08:58:...	206 Odonnell Dr

Owned By

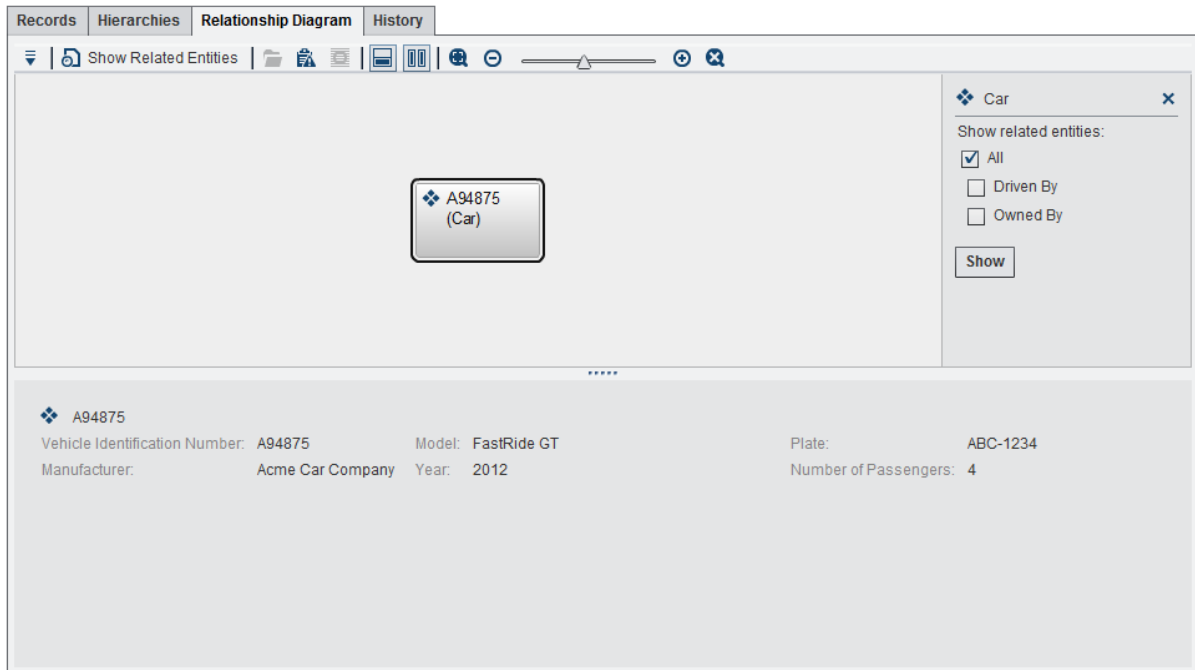
New Relationship

MDM Entity Type	MDM Cluster ID	Full Name	Address Line 1 (Standardized)	City
Company	6023	COLBURN BANKING	4009 Summitview Ave	MERION

Explore Entity Relationships

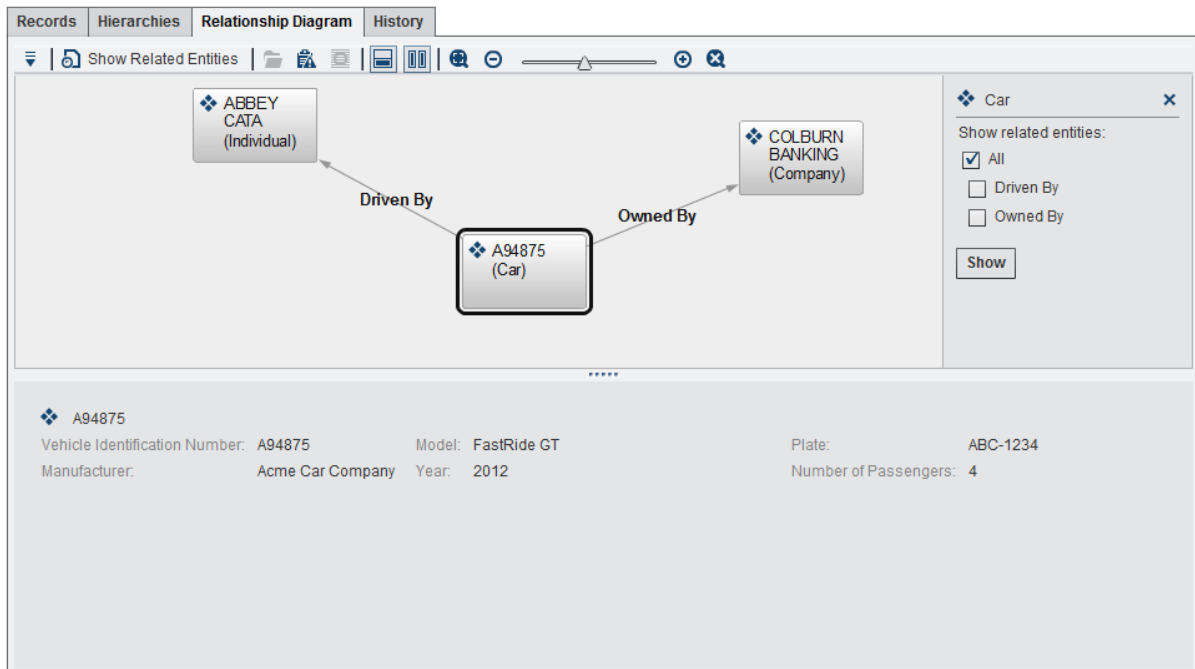
1. Click the **Relationship Diagram** tab in the entity editor to explore the previously created entity relationships graphically.
2. Select the A94875 entity node in the entity graph on the **Relationship Diagram** tab. Leave the **All** check box selected.

Figure 11.24 Initial Relationship Tab



3. Click **Show**.
4. The entity graph pane now displays nodes for the Abbey Cata and Colburn Banking entity records. The arrows between the nodes indicate the Driven By and Owned By relationships with the A94875 entity. Clicking each node displays additional details for the entity in the details pane.

Figure 11.25 Populated Relationship Tab



Note: You can click and drag the nodes to rearrange them on the entity graph. To move the entire graph, press Shift while clicking and dragging the graph.

Glossary

authorization

the process of determining the permissions that particular users have for particular resources. Authorization either permits or denies a specific action on a specific resource, based on the user's identity and on group memberships.

business rule

a statement of business logic that specifies conditions to be evaluated and actions to be taken if those conditions are satisfied.

database

an organized collection of related data. A database usually contains named files, named objects, or other named entities such as tables, views, and indexes.

database management system

a software application that enables you to create and manipulate data that is stored in the form of databases. Short form: DBMS.

DBMS

See database management system

ETL

See extract, transform, load

extract, transform, load

a data warehousing process in which data is extracted from outside sources, transformed according to operational and quality needs, and loaded into a target database. Short form: ETL.

locale

a setting that reflects the language, local conventions, and culture for a geographic region. Local conventions can include specific formatting rules for paper sizes, dates, times, and numbers, and a currency symbol for the country or region. Some examples of locale values are French_Canada, Portuguese_Brazil, and Chinese_Singapore.

match

a set of values that produce identical match codes or identical match code components. Identical match codes are assigned to clusters.

match code

an encoded representation of an input character variable that is used to cluster and compare data.

match definition

a part of a locale that is referenced during the creation of match codes. Each match definition is specific to a category of data content. In the ENUSA locale, for example, match definitions are provided for names, e-mail addresses, and street addresses, among others.

metadata

descriptive data about data that is stored and managed in a database, in order to facilitate access to captured and archived data for further use.

repository

a storage location for data, metadata, or programs.

rule

See business rule

Simple Object Access Protocol

See SOAP

SOAP

a standard, web-based protocol that enables applications written in various programming languages to communicate. Short form: SOAP.

SQL

See Structured Query Language

Structured Query Language

a standardized, high-level query language that is used in relational database management systems to create and manipulate objects in a database management system. SAS implements SQL through the SQL procedure. Short form: SQL.

Index

Special Characters

.met files [36](#)
.NET applications, Web service calls from [8](#)

A

abstract entity types
 defining [85](#)
 example of creating [153](#), [173](#)
 generating jobs for [28](#), [82](#)
 overview [23](#)
access
 control [39](#)
 managing [43](#)
applications, integrating with SAS MDM [8](#)
architecture [4](#)
archive files, generating [34](#)
attribute properties [85](#)
attributes
 constraints [86](#)
 creating [85](#)
 data types [24](#), [86](#)
 display order [24](#)
 encrypting [26](#), [87](#)
 flux [14](#), [86](#)
 grouping [86](#)
 inherited [22](#), [87](#)
 label attribute [30](#), [85](#), [158](#)
 length [86](#)
 modifying [85](#)
 read-only [25](#), [87](#)
 regular expression constraints [86](#)
 required [87](#)
 retiring [86](#)
 retiring required [26](#), [86](#)

B

batch jobs [7](#)

Boolean data type [24](#)
business tier [5](#)

C

cleansing, data [13](#)
cluster conditions [26](#)
 creating [88](#)
 modifying [88](#)
clustering [14](#)
clustering editor [88](#)
clustering tab, entity type editor [88](#)
clusters
 specifying conditions [88](#)
command line interface [5](#)
concrete entity types
 example of creating [166](#)
constraints, attribute [86](#)
country code, in language definitions [110](#)

D

data cleansing [13](#)
data clustering [14](#)
data encryption [26](#), [87](#)
data model [77](#)
data profiling [10](#), [13](#), [23](#), [36](#)
data quality [5](#), [13](#)
data remediation
 issue view [131](#)
 item view [134](#)
 managing issues [130](#)
 overview [113](#)
 package view [137](#)
 portlet [128](#)
 searches [129](#)
 understanding [127](#)
data remediation administration
 issue types properties [122](#)
 properties tab [117](#)
 roles [126](#)

- subject areas permissions 120
- subject areas properties 119
- subject areas task templates 123
- understanding 115
- data standardization 5, 13
- data tier 5
- data types, supported 24, 86
- data, trimming 34
- DataFlux Data Management Server 7, 8
 - licensing 14
 - permissions 43
- DataFlux Data Management Studio 6, 18
 - licensing 14
 - Match Codes node 18
- date data type 24
- deep linking 59
- default attributes
 - grouping 86
- definition editor 83
- definition tab
 - entity type editor 83
- definitions
 - language 109
 - source system 107
- deploying SAS MDM 21
- derived entity types 23
- dfwfproc 30
- display order, attribute 24

E

- editing entities 62
- encrypting data 26, 87
- end-user applications, integrating with
 - SAS MDM 8
- enterprise applications, integrating with
 - SAS MDM 8
- Enterprise Resource Planning 10
- entities
 - construction of 12
 - overview 11
 - relationships 12
- ENTITY (top-level entity type) 22, 84, 89
- entity relationships
 - example of creating 173
- entity type editor 80
 - attributes properties 85
 - clustering tab 88
 - definition tab 83
 - relationship types tab 93
 - tools tab 89
- entity types 80
 - abstract 23, 85
 - cluster conditions 88
 - concrete 85
 - definition 83

- derived 23
- example of creating 153, 173
- importing 36
- parent 84
- publishing 81, 82
- relationship types 93, 94, 99
- retiring 80
- sample 27
- entity types editor
 - relationship types tab 94, 99
- ERP 10
- ETL 8, 11, 33
- examples
 - creating a concrete entity type 166
 - creating an abstract entity type 153, 173
 - creating entity relationships 173
- Extraction, Transformation, and Loading
 - 8, 11, 33

F

- flux attributes 14
 - creating 86
 - grouping 86

G

- generate jobs dialog box 81, 82
- global preferences 49
- groups, defining 41

H

- history tab 70
- home page, interface 48

I

- importing entity type definitions from
 - Master Data Management Foundation 36
- inherited metadata 87
- integer data type 24
- integrating other applications with SAS
 - MDM 8
- interface
 - home page 48
 - overview 51
- IS_MAX_LOAD 30

J

- Java applications, Web service calls from
 - 8
- jobs
 - designing 43

generating for entity types 28

L

label attribute 30, 85, 158
 language definitions 109
 licensing, QKB 14, 109
 list data type 24, 86
 load problems, recovering from 33
 locales 109
 adding 14
 licensing 14

M

Master Data Dashboard report
 enabling 36
 master data management 55, 105
 Master Data Management Foundation 35, 36
 Master Data Management user interface 7
 match codes 5, 14, 17
 Match Codes node, DataFlux Data Management Studio 18
 match conditions, for entity relationship types 96, 102
 match definitions 18
 match sensitivity 18
 matching records 14, 17
 metadata
 adding 21
 attributes 23
 entity type definitions 22
 entity types 80
 importing 36
 inherited 22
 overview 11
 presentation 24
 sample 27
 uses in hub 11

N

new entities, creating 70

P

parent entity type 84
 password 48
 permissions, DataFlux Data Management Server 43
 preferences 49
 global 49
 SAS Data Management Console 50
 presentation metadata 24
 presentation tier 5

process jobs 6

publishing entity type definitions 81, 82

Q

QKB 5, 6
 licensing 14
 QKB, licensing 109

R

read-only attributes 25, 87
 real data type 24
 real-time services 5, 6, 7
 records
 matching 14, 17
 records tab 62
 regular expression constraints, attribute 86
 relationship diagram tab 67
 relationship type definitions 93, 98
 relationship type editor 93, 94, 99
 relationship types tab, entity type editor 93
 relationship types tab, entity types editor 94, 99
 relationship types, entity 27
 creating 93, 94, 99
 creating match conditions 96, 102
 modifying 93, 94, 99
 relationships, entity
 overview 12
 remediation issue, creating 72
 reports
 configuring 36
 Master Data Dashboard 36
 required attributes 87
 retiring 26, 86
 retiring
 attributes 86
 entity type definitions 80
 required attributes 26, 86

S

sample metadata 27
 SAP 8
 SAS Data Management Console 48
 preferences 50
 SAS MDM deployment models 8
 scripting 5
 search results 59
 searching entities 56
 searching hierarchies 74
 sensitivity, matching 18
 Service Oriented Architecture 5

- Servlet 2.5 [5](#)
- Siebel [8](#)
- SOA [5](#)
- SOAP [5](#)
- source system definitions [107](#)
- staging tables [33](#)
- standardization [5](#), [10](#), [13](#), [23](#), [158](#), [163](#)
- string data type [24](#)
- survivor records [11](#)
- survivorship [10](#), [18](#), [83](#)

T

- task manager
 - managing tasks [146](#)
 - overview [141](#)
 - portlet [144](#)
 - roles [148](#)
 - searches [145](#)

- understanding [144](#)
- text data type [24](#)
- TIBCO [8](#)
- timestamp data type [24](#)
- tools editor [89](#)
- tools tab
 - entity type editor [89](#)
- transpose tables [37](#)
- trimming data [34](#)

U

- URL data type [24](#)
- user id [47](#)
- user interface [47](#)

W

- webMethods [8](#)