



California
DEPARTMENT OF TECHNOLOGY

California Enterprise Architecture Framework

Master Data Management (MDM) Reference Architecture (RA)

Version 1.0 Final
January 2, 2014



This Page is Intentionally Left Blank



TABLE OF CONTENTS

1	Introduction.....	1
1.1	Purpose	1
1.2	Limitations.....	1
1.3	Intended Users.....	1
1.4	Document Organization	2
1.5	Future Directions	2
2	Master Data Management Overview	3
2.1	Definitions.....	3
2.2	Business Benefits	6
2.3	MDM Usage Scenarios	7
2.4	MDM Implementation Styles.....	8
2.4.1	<i>Registry Style</i>	8
2.4.2	<i>Consolidation Style</i>	8
2.4.3	<i>Transaction Style</i>	9
2.4.4	<i>Coexistence Style</i>	9
2.5	Key Capabilities of MDM Solution	9
2.5.1	<i>Master Data and Metadata-Related Capabilities</i>	9
2.5.2	<i>Integration- and Interoperability-related Capabilities</i>	10
2.5.3	<i>Supporting Capabilities</i>	10
2.6	Components of MDM Solution	10
2.6.1	<i>Master Data Repositories</i>	11
2.6.2	<i>Master Data Management Services</i>	11
2.6.3	<i>MDM Analysis and Discovery Services</i>	12
2.6.4	<i>Enterprise Information Integration (EII)</i>	13
2.6.5	<i>MDM Presentation Services</i>	14
2.6.6	<i>MDM Integration Services</i>	15
3	MDM Reference Architecture Description	16
3.1	MDM RA Conceptual View.....	16
3.2	MDM RA Logical View	18
3.3	MDM RA Deployment View	21
4	MDM Implementation Guidelines.....	22



4.1	MDM Implementation Lessons.....	22
4.2	A Step-wise Approach for Introducing MDM.....	23
5	Glossary	25
6	References and Bibliography.....	26
7	Document History	27





LIST OF FIGURES

<i>Figure 2-1 Master Data and Other Types of Organizational Data</i>	3
<i>Figure 2-2 Enterprise Context of Master Data Fragmentation and Duplication</i>	4
<i>Figure 2-3 Organizational and Technical Facets of MDM</i>	5
<i>Figure 2-4 MDM Components Overview</i>	10
<i>Figure 2-5 Master Data Repositories in MDM RA</i>	11
<i>Figure 2-6 Master Data Management Services in MDM RA</i>	12
<i>Figure 2-7 Analysis and Discovery Services In MDM RA</i>	13
<i>Figure 2-8 Enterprise Information Integration in MDM and BI RA</i>	14
<i>Figure 2-9 Presentation in MDM RA</i>	15
<i>Figure 2-10 Integration Services in MDM RA</i>	15
<i>Figure 3-1 MDM Reference Architecture – Conceptual View</i>	17
<i>Figure 3-2 Service and Component Interactions when Accessing Master Data</i>	19



LIST OF TABLES

<i>Table 2-1 Aspects of Quality in Master Data</i>	6
<i>Table 3-1 Accessing Master Data, Application’s Perspective</i>	20
<i>Table 4-1 Common MDM Implementation Lessons</i>	22
<i>Table 4-2 Step-wise Approach for Introducing MDM</i>	23
<i>Table 7-1 Document History</i>	27



This Page is Intentionally Left Blank

1 Introduction

Of all important types of data in the enterprise, there is a type of data that is more important for the organization than others. This group of data is called “Master Data”, and is widely believed to be the most important data assets owned by an organization. It is this Master Data on which enterprise business transactions are executed. Enterprise business analytics often involve analysis of other data in relation to this Master Data. Master Data is also involved in information sharing within an organization and with partners and is crucial to interoperability.

Today’s organizations are faced with ever growing abundance of data. The data comes from various sources (external and internal to the organization, involving potentially multiple systems of record); unfortunately, the data often are of uncertain quality. Not being able to deal with abundant yet scattered and uncertain data is likely to lead to major operational challenges and faulty decisions.

Master Data Management (abbreviated as “MDM”) is an effort to tame problems related to Master Data in an organization in a reliable and repeatable way, and to provide for clean and authoritative source of Master Data. This document presents a Reference Architecture (RA) for an SOA-based enterprise-level architectural approach to MDM, as part of CEAF 2.0.

1.1 Purpose

The Master Data Management (MDM) Reference Architecture (RA) document provides guidelines and options for making architectural decisions when implementing MDM solutions.

The objectives for the document include the following:

- To introduce key terms and distinctions relevant for the topic
- To provide inputs for creating or evaluating architectures for MDM from enterprise perspective
- To identify building blocks (architectural layers, services, components) for integrating elements of an MDM solution
- To communicate the key architectural decisions relevant for creating or evaluating MDM solutions
- To communicate opportunities for solution and/or platform sharing at agency, cross-agency and/or state levels

1.2 Limitations

The document focuses on MDM and related concepts at the enterprise architectural level in the context of CEAF 2.0. It is not intended to serve as the MDM-related product guide, or a guide for *organizational* aspects of MDM solutions. It is also not intended as an endorsement of any product. Its overall perspective remains CEAF-centric.

1.3 Intended Users

The primary intended users of this document are Enterprise Architecture practitioners and other architects that contribute to enterprise architecture. This broad group includes architects from other domains/disciplines such as Security, Application, Information, Business, Technology,

Infrastructure, and Solution Architects. It is also beneficial to Managers, at senior or operational levels, who are involved with MDM or related areas, such as Enterprise Application Integration, Business Intelligence, SOA, and similar areas.

1.4 Document Organization

The MDM Reference Architecture documentation is organized as follows:

- Section “MDM Overview” provides background for the MDM RA by introducing descriptions and definitions of MDM, discusses the main usage scenario types found in MDM implementations, and identifies architectural components for respective usage scenarios.
- The section “MDM Reference Architecture Description” elaborates RA for MDM using the following architectural views:
 - The Conceptual View (in the subsection “MDM RA Conceptual View”) introduces the necessary capabilities for an MDM architecture and how they are supported by Architectural Building Blocks (ABBs)
 - The Logical View (in the subsection “MDM RA Logical View”) describes key interactions among Layers and/or ABBs to realize functionality specific to MDM systems
 - The Deployment View (in the subsection “MDM RA Deployment View”) focuses on system topologies and deployment facets of MDM in the state.
- The section “Glossary” provides description of the terms and abbreviations used in the document
- The section “References” lists publications used for preparation of the document. Note that literals in square brackets that appear in the document (e.g., [A], [5], [d]) are references to publications listed in this section

1.5 Future Directions

Future evolution of the document includes the following steps:

- Addition of existing best-practice-based realizations of the MDM RA
- Identification and elaboration of solution sharing opportunities
- Formulation of implementation guidelines for MDM RA

2 Master Data Management Overview

This section provides a description of Master Data Management (MDM), including clarification of key terms and concepts. It identifies MDM's intended business benefits and summarizes its main usage scenarios. A set of key capabilities of an MDM solution are identified and key components of the solution are described at a high level.

2.1 Definitions

In the context of the enterprise, **data** is understood as raw information required for everyday operations and for adequate decision making. Broadly, this information is about the domain in which the enterprise operates.

Master Data (further abbreviated as “MD”) can be defined as data “referring to core business entities an organization uses repeatedly across many business processes and systems” [5]. MD “captures the key things that all parts of an organization must agree on, both in meaning and in usage” [4]. Master data includes the business objects, definitions, classification, and terminology that constitute business information; consequently, master data forms the basis for business processes.

MD is sometimes referred to as “the single source of truth” – that is, the authoritative source or the “system of record”; given that function, MD is critical business information that contains details of internal and external business entities involved in business transactions and business analytics of the enterprise. MD also explains the context within which an organization does its business and holds the business rules.

The following figure shows Master Data among other types of organizational data:

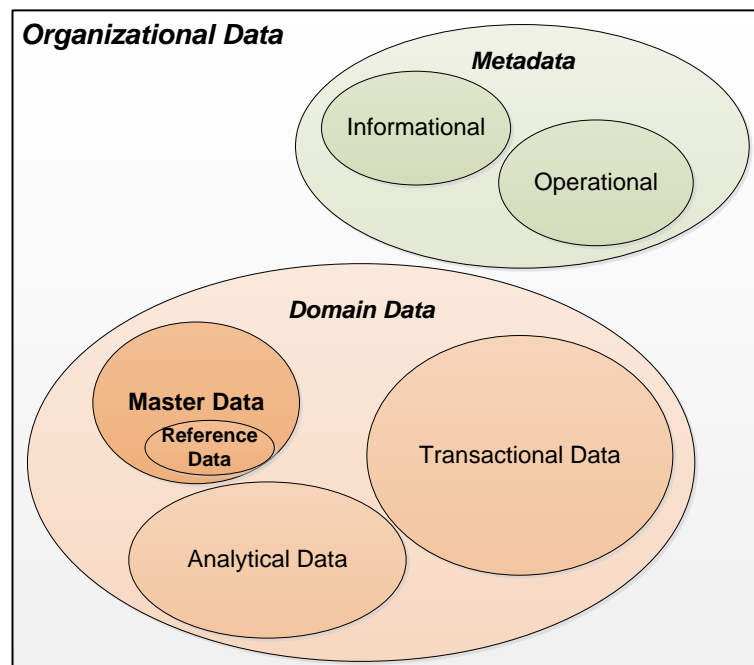


Figure 2-1 Master Data and Other Types of Organizational Data

In the above figure, “Metadata” are the data about other data, typically about Domain Data – such as database catalogs or XML schemas. The area of Domain Data contains Transactional Data, analytical data, and Master Data. Master Data are different from Transactional Data in a number of respects; the following list main differences:

- Master Data objects are independent of other objects (for example, a customer is independent of a sale, but not vice versa), whereas transactional data contains dependent data objects
- Master Data have low volatility compared to Transactional Data in terms of their schema – the structure of master data objects rarely changes over their lifetime, and the structure of Transactional Data can change significantly depending on the transactions they represent
- Master Data have constant or limited changes in the volume, at least compared to Transactional Data, with quickly growing volumes.

In addition to above characteristics, specific data entities can serve as Master Data depending on the needs of a specific domain. However, there are commonalities that can be pointed out. As a rule of thumb, Master Data involve the following types of data objects:

- Data related to *Parties* of the relationships in the enterprise – such as customers, citizens, suppliers, or employees
- Data related to *Things* that are present in the relationships with the Parties, such as services, products, assets, and similar. Things can form hierarchies and groupings
- Data related to *Locations* involved in *Parties* and *Things* – such as places, sites, regions, etc.
- Data representing cross-domain relationships (for example, between an employee, a location and a service to be rendered).

The following figure depicts a typical context of multiple data sources, data definitions, quality rules and governance, which typically create fundamental challenges with respect to master data maintenance and its quality:

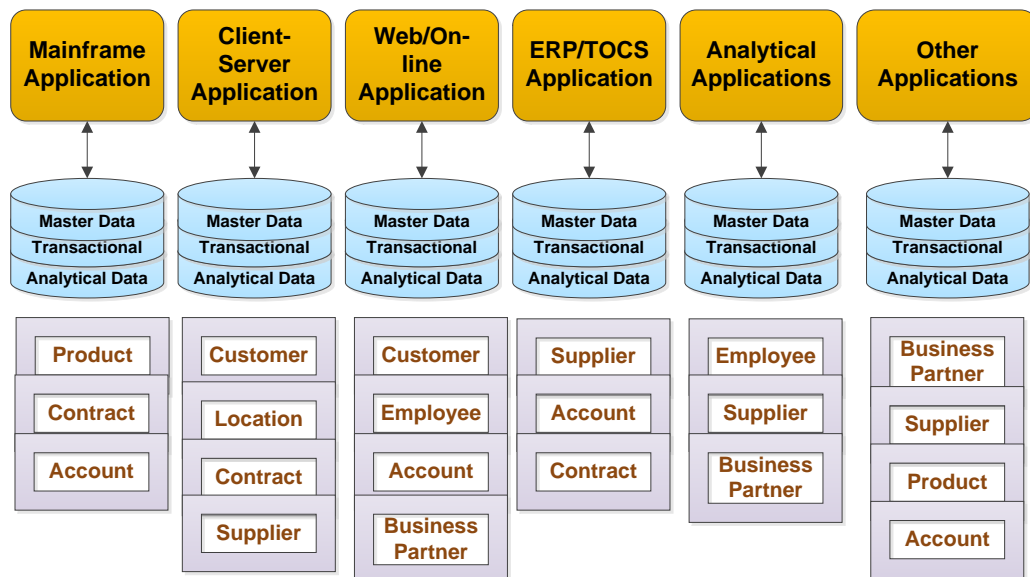


Figure 2-2 Enterprise Context of Master Data Fragmentation and Duplication

As suggested in the above figure, today’s organizations operate using high volumes of data coming from various sources, ranging from mainframe applications, client-server systems to

web and on-line applications. In functioning of many of those systems, the overlap in using core business data is unavoidable, and if unchecked it leads to multiple formats and/or versions of core business data objects. It is not uncommon for a single data entity (such as customer) to be present in a dozen or more of separate systems.

Master Data Management (MDM) is a collection of best data management practices that orchestrate key stakeholders, participants, and business clients in incorporating the business applications, information management methods, and data management tools to implement the policies, procedures, services, and infrastructure to support the capture, integration, and subsequent shared use of accurate, timely, consistent, and complete master data. [1]

The first main task for MDM is to address data quality problems, typically caused by fragmented system and application environments. In turn, data quality problems pose challenges for business, not only on the operational level, but also on the strategic level, where analytics on reliable data are required to make right decisions.

In addition to Master Data quality, other areas that need to be taken into account when introducing an MDM solution to the organization include Master Data structure and processes, Master Data systems architecture, and Masters Data governance. The following figure schematically shows these areas:

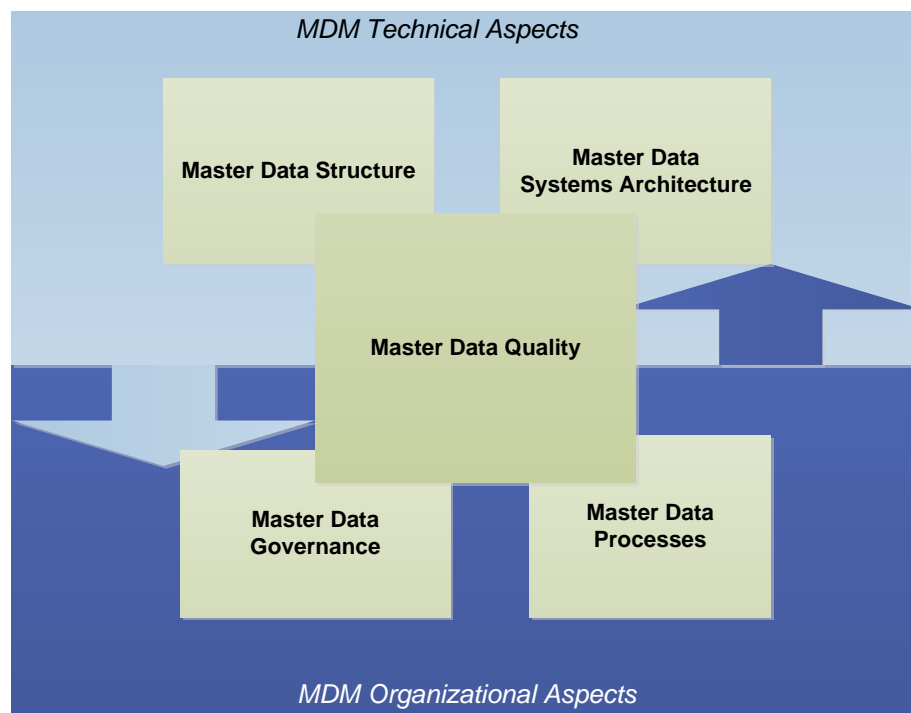


Figure 2-3 Organizational and Technical Facets of MDM

As shown in the above figure, both organizational and technical aspects of MDM contribute to a successful implementation. The central place in the facets of MDM belongs to quality of Master Data. The following table summarizes attributes of quality of Master Data [3].

Table 2-1 Aspects of Quality in Master Data

Attribute	Description
Accuracy	Data are accurate when a stored piece of information is conformant with its actual value
Completeness	Data are complete when it contains all relevant entities, attributes, and values required to represent the real-life master constructs such as customers, products, or accounts
Consistency	Data are consistent when it has the same values for the same entities regardless of the location or manner of their retrieval.
Timeliness	Data are timely if the propagation of changes to data values across organization does not affect quality of the master data.
Relevance	Data are relevant when it satisfies the needs of their consumer(s) in a given context.

The concept of trust in case of Master Data is typically defined in terms of the above attributes. When the data has all above attributes (or their practically acceptable approximations), they are recognized as trust-worthy.

2.2 Business Benefits

The value of MDM is *not just* in the establishment of sound data management practices. MDM is a means by which other strategic and operational objectives can be successfully accomplished. MDM can be justified in support of business initiatives that rely on any of following benefits enabled by the unified view of key business information objects provided by MDM:

- Increase information quality:** Master data made up of standardized models, and business rules enables organizations to have a solid data foundation which helps to more effectively monitor conformance to information quality expectations across lines of business applications
- Improve business productivity and operational efficiency:** MDM helps remove duplication and manual entry in multiple systems across the enterprise. It helps organizations understand how the same data objects are represented, manipulated, or exchanged across applications within the enterprise and how those objects relate to business process flows. This understanding gives organizations an opportunity to explore how effective the organization is in automating its business processes and to identify business process improvements by exploiting the information asset. High quality master data the business depends on can also enable a higher level of straight-through processing resulting in fewer process errors and rework

- **Improve interoperability:** Quality master data enables efficient cross-communication between disparate systems and improves interoperability
- **Improve governance, regulatory compliance, and risk management:** Master data fragmentation negatively impacts governance, compliance and risk management processes by making data hard to reconcile and business process reporting difficult to extend across the lines of business. MDM eliminates inconsistencies in master data and enables strong process controls
- **Simplify and accelerate application development:** When master data objects across multiple systems are consolidated into a master repository, there is a corresponding opportunity to consolidate the application functionality associated with the master data life cycle. New applications only need to integrate with the MDM application for sharing master data across the existing application portfolio. MDM also allows the introduction of a technical service layer for data life cycle functionality providing the abstraction necessary for deploying a service-oriented architecture (SOA). A standardized view of the information asset reduces the delays associated with extraction and transformation of data, speeding the implementation of application migrations, application modernizations, and construction of a data warehouse
- **Enable consistent reporting and improve decision-making:** MDM delivers high-quality data for better decisions. Reliance on the reports generated from governed processes using master data reduces inconsistencies. The information consistency provided by MDM across applications reduces data variability, which in turn improves data quality in business intelligence and data warehousing systems allows for clearer and faster business decisions.
- **Enable comprehensive customer knowledge.** Legacy application infrastructures often support the same type of customer data functionality in different ways. Customer records may be created, updated, or retired through any of these applications in an uncoordinated way. Master data enables a true customer centric view of enterprise information.

2.3 MDM Usage Scenarios

There are a number of ways Master Data can be used in an organization. The main usage scenarios for MDM are as follows:

- **Collaborative scenario**, in which multiple users participate in the same process on a master data entity. This scenario has a number of prerequisites: a workflow support with check-in/check-out functions; support for relationships and hierarchy management; and attribute-level granularity of authorization privileges.
- **Operational scenario** takes place when an MDM system functions as an Online Transaction Processing (OLTP) server. In this scenario, a large number of applications and users require quick access to master data to retrieve and change master data through MDM services invoked by business processes. In this scenario, MDM services are often used in the context of an SOA and need to be accessible through a variety of interfaces.
- **Analytical scenario** in which analytical operations on Master Data are the main focus. This includes Identity Analytics (e.g., to verify an identity and discover hidden relationships) and requires support for reporting and ad hoc queries on Master Data. The scenario is also used when integrating analytics with data warehouses, in order to provide master data to the data warehouses for accuracy improvements. The insights gained in the data warehouse are made available and actionable by feeding them back to the MDM System for use.

- **Enterprise scenario** represents a combination of Collaborative, Operational and/or Analytical MDM usage scenarios. Although this scenario is often desirable because it can deliver maximum business value, in practice it requires an incremental approach to successful, low-intrusion implementation.

2.4 MDM Implementation Styles

There are a number of MDM implementation styles, as follows:

- **Registry** style, applicable for read-only view of master data in the MDM solution
- **Consolidation** style for full materialization of all master data attributes in the MDM solution
- **Transaction** style for full materialization and also for completeness and consistency of master data
- **Co-existence** style for hybrid solutions involving more than one solution style.

These implementation styles differ in what they provide and in their area of applicability and trade-offs. The subsections that follow provide short characterizations.

2.4.1 Registry Style

The Registry Style of MDM implementation is applicable when a read-only view of master data is required and sufficient for subscribing systems. This style helps to identify and remove duplicates and to provide consistent access to master data to subscribing systems.

In this style, the MDM system holds only a thin slice of master data - mainly the data required to enforce uniqueness and for cross-referencing. Outside of the MDM-managed slice, other data attributes remain in application systems without harmonization. MDM system uses federated query mechanisms to access these attributes.

Given the limited scope of the MDM system in this style, master data is neither consistent nor complete regarding all attributes in the MDM system. At the same time, the MDM system is quicker to deploy and has lower costs compared to other implementation styles. It also involves less intrusion into application systems compared to other styles.

2.4.2 Consolidation Style

The Consolidation Style of MDM implementation fully materializes all master data attributes in the MDM system. The authoring of master data happens in the application systems. The MDM system is updated from those sources.

This style results in convergent consistency due to multiple source application systems updating master data, delays in the synchronization of master data updates in the MDM, and resolution of conflicting updates in the MDM system. Over time, the delays in the distribution of updates to the MDM system and the delays for conflict resolution and synchronization of master data in the MDM system decrease, marking the move towards absolute consistency of master data.

Consolidation Style also involves higher deployment costs compared to the Registry Style because all master data attributes need to be harmonized and cleansed before loading into the MDM system. To offset the deployment costs, the style offers improved master data quality and faster access to master data due to the absence of federation.

The Consolidation Style is often used to support reporting and analytical MDM. For those areas, this style has an important advantage over other styles: as all attributes of Master Data are in one place and are harmonized, reporting gets simplified.

2.4.3 Transaction Style

The Transaction Style of MDM implementation aims not only at full materialization of all master data attributes in the MDM System, but also at maintaining consistency, completeness, and accurateness of that data at all times. To that end, both read and write operations on master data are done through MDM system (utilizing SOA techniques). Moreover, all applications with the need to change master data invoke MDM services to do so.

In this style, an absolute consistency on master data can be achieved because propagation of changed master data causing delay no longer exists.

However, with the important gains come some downsides:

- Deployment of the Transaction Style requires deep intrusion into the application systems
- It requires intercepting business transactions so they interact with the MDM System for master data changes
- It also requires adequate infrastructure such as a global transaction mechanism (e.g., a two-phase commit infrastructure).

2.4.4 Coexistence Style

The Coexistence Style of MDM implementation is marked by coexistence of more than one style of implementation. In this style, authoring of master data can happen either in the MDM system or in the application systems. Updates between the applications systems and the MDM system are performed selectively. Similarly, publishing of the MDM data to subscribing systems is done selectively, too.

Consolidation Style MDM implementations usually evolve into coexistence style implementations before further evolving into Transaction Style MDM implementations.

2.5 Key Capabilities of MDM Solution

The key capabilities of MDM are grouped here as follows:

- Master data and metadata-related capabilities and characteristics
- Integration- and interoperability-related capabilities
- Supporting capabilities

Each group of the above capabilities is presented in the subsections that follow.

2.5.1 Master Data and Metadata-Related Capabilities

Master Data and Metadata related capabilities of MDM solutions include the following:

- A flexible, extensible and open enterprise data model to hold the master data and all needed attributes; the data model must be application-neutral. Built-in Data Models/Hubs packaged with MDM products are desirable
- Support for Enterprise Business Objects, including:
 - Fully cross-referenced business objects
 - Metadata management of business entity relationships and hierarchies
- Support for common enterprise-wide metadata
- Support global identification, linking and synchronization of master data across heterogeneous data sources through semantic reconciliation of master data
- Support for OLTP workloads and connected/subscribing applications

- Built-in analytics and discovery functions for directly analyzing master data and discovering hidden relationships.

2.5.2 Integration- and Interoperability-related Capabilities

Integration- and interoperability-related capabilities of MDM solutions include the following:

- Supporting multiple implementation styles and multiple methods of use
- Capability to leverage Enterprise Information Integration (EII) Platform
- High availability
- Scalability for mission critical data access supporting heavy mixed workloads.

The interoperability capabilities with platform components include the following:

- Integration with Enterprise Portal
- Integration with Enterprise Identity and Access Management
- Role based fine grained attribute level access control
- Support for providing MDM Services through integration with Enterprise Application Integration (EAI) Bus/Hub
- Support for SOA technologies.

2.5.3 Supporting Capabilities

Supporting capabilities of MDM solutions include the following:

- User interfaces for business users, master data authors and data stewards
- Providing for ongoing master data stewardship and governance through workflow and event based monitoring.

2.6 Components of MDM Solution

The following figure provides an overview of main groups of components in an MDM solution:

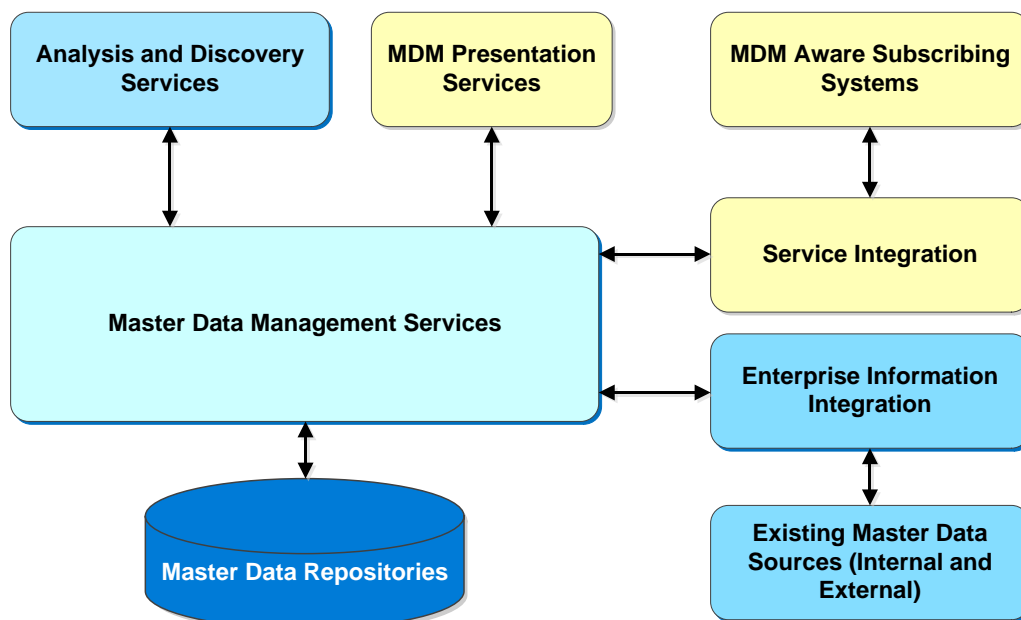


Figure 2-4 MDM Components Overview

The above figure shows the following components:

- Master Data Repositories
- Master Data Management Services
- MDM Presentation Services
- Service Integration with MDM-Aware Subscribing Systems
- Enterprise Information Integration (EII) and Existing Master Data Sources.

The above components are described in subsections that follow.

2.6.1 Master Data Repositories

The following figure shows Master Data Repositories:

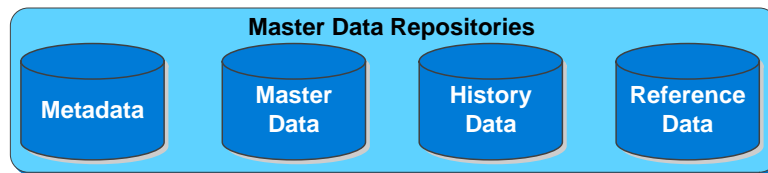


Figure 2-5 Master Data Repositories in MDM RA

The components shown in the above figure are as follows:

- **Metadata** are data *about* Master Data (such as origin of the data or transformation applied). Metadata are also used for validations, cleansing, searches, matching and executing business rules
- **Master Data** include definitions of data entities (schema, attributes, relationships, hierarchies, default values, formats etc.) and Instance Data
- **History Data** capture changes to Master Data using updates produced by Audit/Logging Services
- **Reference Data** are low-volatility data that typically originate outside the organization but are still relevant for the functioning of the organization. This data typically contains codes and their descriptions (such as currency codes). Recognizing, harmonizing and sharing this data as *master reference data* for “reference” by multiple consumers (systems, other data sets, and people) is a rapidly evolving industry trend. Managing reference data as a special type of master data by leveraging MDM hub functionality is widely accepted as a best practice.

2.6.2 Master Data Management Services

The following figure shows Master Data Management Services:

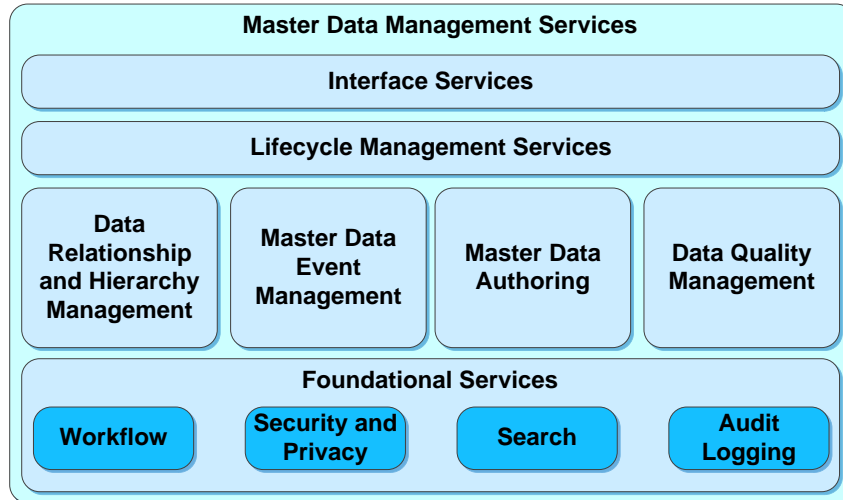


Figure 2-6 Master Data Management Services in MDM RA

The components shown in the above figure are as follows:

- **Foundational Services** include Workflow, Security and Privacy, Search and Audit logging
- **Data Relationship and Hierarchy Management** is responsible for the following:
 - Managing of the defined master data hierarchies, groupings, and relationships
 - Discovering non-obvious relationships through other components (e.g., Identity Analytics Services) and storing that information in the MDM System
- **Data Quality Management** is responsible for enforcing data quality rules and for performing data cleansing, standardization and reconciliation
- **Master Data Event Management** is responsible for detecting and triggering actions based on business rules and data governance policies
- **Master Data Authoring** is responsible for the following:
 - Providing services to author, approve, manage, customize and extend the definition of master data
 - Providing ability to add, modify instance master data
 - Supporting the MDM collaborative style of use
 - Servicing invocations as part of a collaborative workflow
- **Life Cycle Management Services** are responsible for supporting CRUD (create, read, update, and delete) operation, and for maintaining consistent business logic that applies depending on the data context
- **Interface Services** provide consistent entry point to MDM services, and support messaging, method calls, web services, and batch processing capabilities

2.6.3 MDM Analysis and Discovery Services

The following figure shows MDM Analysis and Discovery Services:

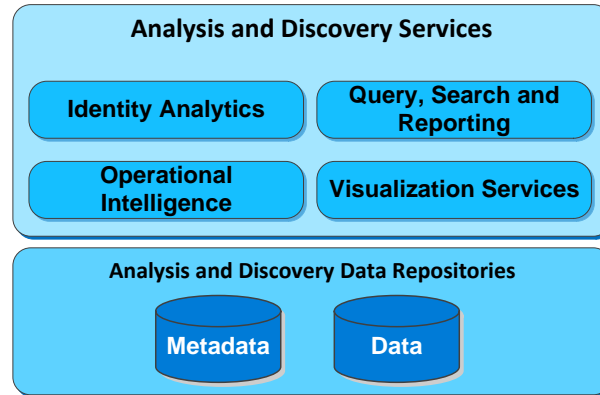


Figure 2-7 Analysis and Discovery Services In MDM RA

The components shown in the above figure are as follows:

- **Identity Analytics**, including identity resolution capability to uniquely identify an entity (e.g., a person or a party)
- **Operational Intelligence**
- **Query, search and report** creation services, usually centered around a reporting engine
- **Visualization services**, which provide functionality to represent potentially complex information in ways suitable for human consumption
- **Analysis and Discovery data repositories** to provide persistence and CRUD capabilities for analysis and discovery services.

2.6.4 Enterprise Information Integration (EII)

Enterprise Information Integration (EII) provides common set of data integration capabilities across enterprise and is a critical component of overall Enterprise Information Management. EII appears both in the Business Intelligence RA and in this MDM RA. EII typically needs to support multiple styles of data integration, including the following:

- Traditional ETL-based integration (bulk extract, transform, and load)
- Capturing of changed data, typically involving a form of subscribe and publish (“pub/sub”) mechanism
- Data cleansing and quality management, related to Master Data Management (MDM) when such solutions are present
- Summarization capabilities
- Information integration process flow management that allows for automation of data integration processes
- Message-based data integration, which typically involve an Enterprise Service Bus (ESB) together with Adaptors and/or Connectors to access various data sources.

Key capabilities of EII include the following:

- Strong support for meta-data management, in order to support identification, classification, and effective querying of information
- Extensibility, in order to easily accommodate modifications to existing data integration styles or addition of the new required styles
- Scalability, in order to sustain increasing data volumes and loads.

The following figure shows EII components:

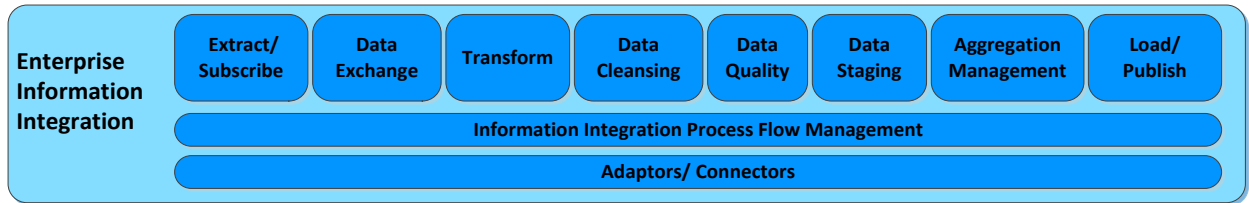


Figure 2-8 Enterprise Information Integration in MDM and BI RA

The components shown in the above figure are as follows:

- Information Integration Process Flow Management component, which allows for defining business processes, workflows or automated data integration jobs relevant to integration of information
- Adaptors/Connectors components, which allow for interoperability between differing access protocols, and with various data sources
- Load/Publish components, which provide for active disseminating of data entities to subscribing systems or applications
- Extract/Subscribe components, which allow for consumption of disseminated data entities by subscribing systems or applications
- Data Exchange component
- Transform component, which is responsible for predefined transformations between various data formats and values
- Data Cleansing component , which provides for identification of violations of applicable data standards, and for standardized transformations of data values
- Data Quality component , which helps identify and measure quality-related attributes of the data being processed
- Data Staging component , which supports persistence needs during information integration processes
- Aggregation Management component, which supports definition of dimensions/views and corresponding aggregations of data to be pre-calculated or obtained on demand.

2.6.5 MDM Presentation Services

MDM Presentation Services provide users with interfaces (preferably thin/browser-based GUI) to interact with various areas of the MDM solution, typically in function of the user role (e.g., typical user role of MDM versus administrator role). The following figure shows Presentation components in MDM solution:

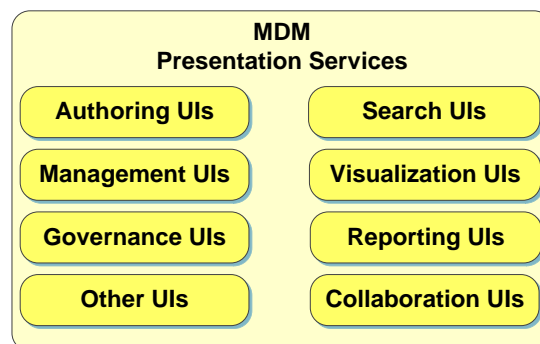


Figure 2-9 Presentation in MDM RA

The user interface components shown in the above figure are as follows:

- UIs for Master Data querying and reporting
- UIs for Master Data visualization (of relationships, aggregations, etc.) using various representational approaches and algorithms
- UIs for data and metadata authoring and collaboration
- UIs for management and governance.

User Interfaces are typically responsible for organizing direct interaction of human users with MDM services, including collecting input, performing first-level validations, or presenting output data. It is not, however, the responsibility of presentation components to specify business rules that are applicable to Master Data or to define workflows/business processes that control the interactions between the system and the user.

2.6.6 MDM Integration Services

The following figure shows Integration components in MDM solution:

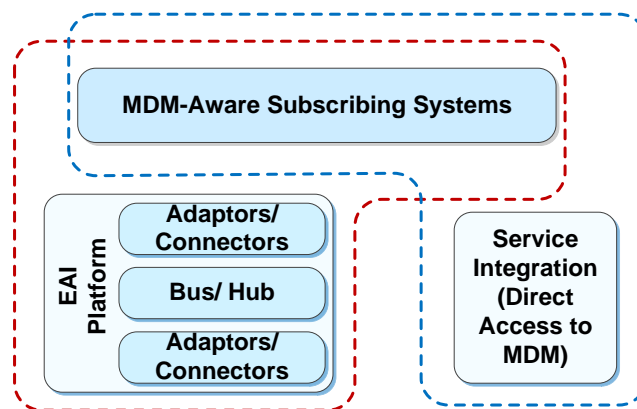


Figure 2-10 Integration Services in MDM RA

The components shown in the above figure fall into two groups, marked by different colors in dotted lines:

- *Direct* MDM Access for MDM-aware systems, using *Service Integration* component of the MDM solution
- *Indirect* MDM Access for MDM-aware systems, using integration through Enterprise Application Integration (EAI) platform (Bus/Hub), or into enterprise SOA. This type of access allows providing master data as Data-as-a-Service, integration with Business Process Management systems, and building Composite Applications.

The portrayed Integration services provide for real-time, high volume, high performance access to MDM-aware subscribing systems. MDM Integration Services are required for dissemination of changes in Master Data across applications and systems, and consequently they are crucial for preserving the quality of Master Data in the enterprise.

3 MDM Reference Architecture Description

This section describes MDM Reference Architecture (RA) using three views:

- Conceptual View, which provides a summary of logical-level building blocks for MDM as presented in the Section 2 above
- Logical View, which provides an overview of relationships and interactions between components in an MDM solution for specific usage scenarios
- Deployment View, which illustrates the distribution of components and processing across nodes in the system.

Each of the above views is presented in the subsections that follow.

3.1 MDM RA Conceptual View

The following Conceptual View figure brings together all major components of an MDM solution that have already been described in the section “Components of MDM Solution”. The diagram shows key MDM components organized in subsequent horizontal layers. The layers placed vertically (such as Enterprise Information Integration) represent components that can be used in more than one horizontal layer.

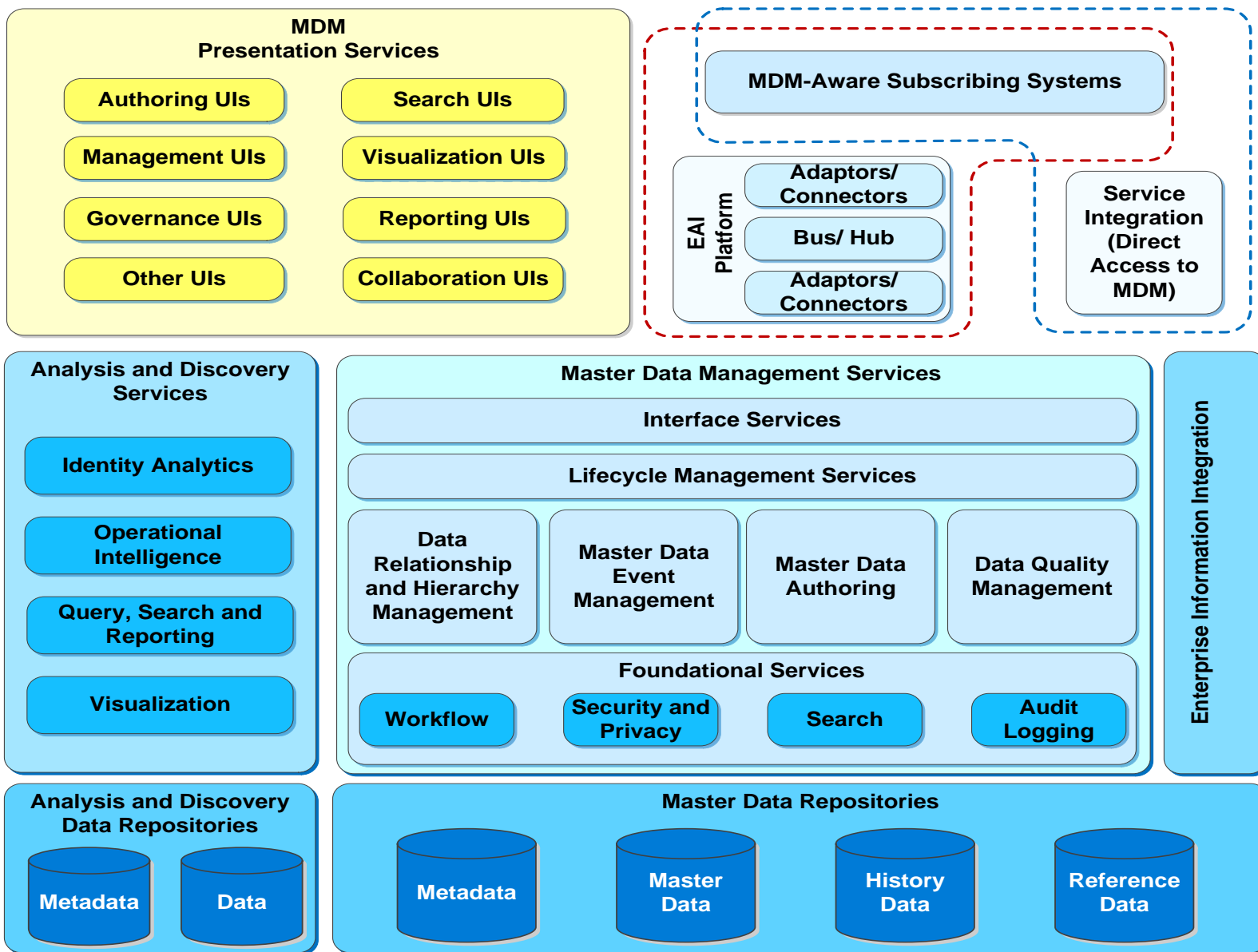


Figure 3-1 MDM Reference Architecture – Conceptual View

3.2 MDM RA Logical View

This section shows the relationships and main interactions among the components of an MDM solution in a sample scenario: *Accessing Master Data in a Registry Style MDM Implementation*.

In this scenario, a user working with an *MDM-aware subscribing system* accesses master data through the MDM system. It is important to distinguish between the interactions as they are visible to the user and the sequence of interactions between applications and their components. The interactions from the user's perspective appear simple because the interactions between the MDM-aware subscribing system and the MDM system are not visible to the user. Figure 3-2 shows the typical interactions among the application components in this scenario.

An overview of these interactions is provided in Table 3-1. The interaction numbers shown in the table correspond to the interaction numbers (shown in circles) shown in Figure 3-2.

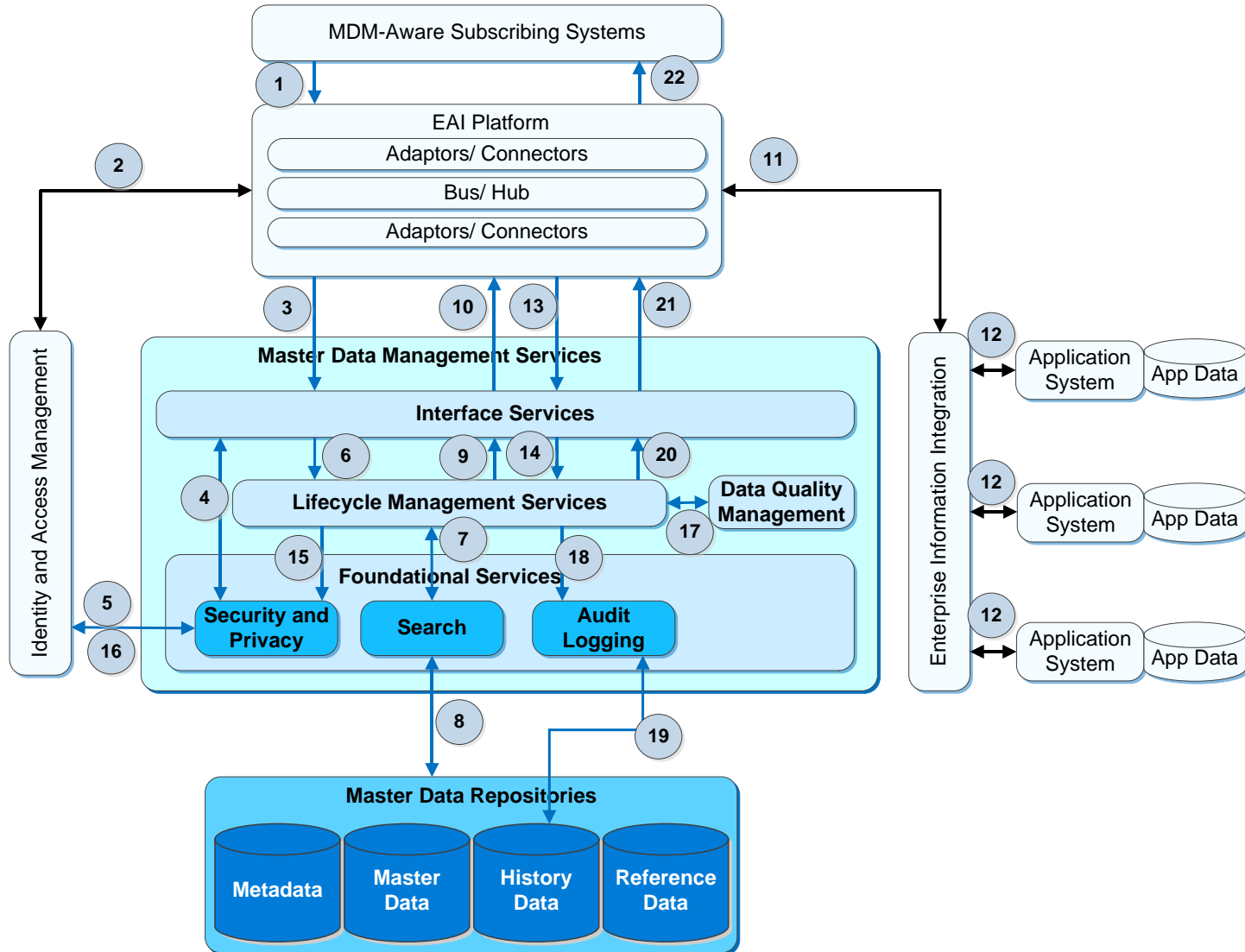


Figure 3-2 Service and Component Interactions when Accessing Master Data

Table 3-1 Accessing Master Data, Application's Perspective

Interaction	Description
1	An <i>MDM-aware subscribing system</i> sends a <i>web service request</i> to access master data.
2	The <i>Enterprise Application Integration (EAI) platform</i> (which includes an Enterprise Service Bus) checks the service registry (not shown in the figure) to identify the service and policy details. It verifies user's authorization to access the service by interacting with the <i>Identity and Access Management platform</i> .
3	The <i>Interface Services</i> component of the MDM receives the request for master data.
4	The Interface Services component invokes the <i>Security and Privacy</i> component to verify user's authorization to access the specific master data.
5	The Security and Privacy component verifies user's authorization to access the specific master data by interacting with the <i>Identity and Access Management platform</i> .
6	The Interface Services component invokes the <i>Lifecycle Management Services</i> component to service the request.
7	The Lifecycle Management Services component invokes the <i>MDM Search</i> service to query the Master Data Repository.
8	The MDM Search service queries the <i>Master Data Repository</i> to retrieve the cross-reference information. This cross-reference information contains the mapping between the subscribing system's attributes and the master data attributes, and the mapping between master data attributes and the source application system's attributes. In a Registry Style MDM implementation, this cross-reference information is populated when a "slice" of master data is loaded into the MDM system.
9	The Lifecycle Management Services component uses the information returned from the MDM Search service, constructs <i>Federated Query Service request(s)</i> and passes it to the Interfaces Services component.
10	The Interface Services component sends the Federated Query Service request(s) to the EAI Platform.
11	The EAI Platform checks the service registry (not shown in the figure) to identify the service and policy details, verifies authorization to access the service, and sends the service request to the <i>Enterprise Information Integration (EII) Platform</i> .

Interaction	Description
12	The EII Platform sends the query to the appropriate source <i>Application System</i> , retrieves the query results, and sends the response back to the EAI platform. The EII Platform here virtualizes the source Application Systems and/or Databases and provides an entry point to access enterprise information from multiple sources. It should be noted that the EII platform may directly access the source database, or use other mechanisms provided by the applications to execute the query.
13	The EAI Platform sends the query results to the MDM Interface Services.
14	The Interface Services component sends the query results to the Lifecycle Management Services component.
15	The Lifecycle Management Services component invokes the Security and Privacy component as part of post-processing to filter the query results based on the rules of data visibility.
16	The Security and Privacy component verifies user's authorization related to the rules of data visibility with the <i>Identity and Access Management platform</i> .
17	After filtering the results based on the rules of visibility, the Lifecycle Management Services component invokes the <i>Data Quality Management</i> component to apply quality rules to the result set.
18	The Lifecycle Management Services component invokes the <i>Audit Logging</i> component to log the transaction details.
19	The Audit Logging component logs the transaction details to the <i>MDM History Database</i> .
20	The Lifecycle Management Services component sends the formatted result set to the Interface Services.
21	The Interface Services component sends the result set (response) to the EAI platform.
22	The EAI platform sends the <i>response to the original web service request to the MDM-aware subscribing system</i> .

3.3 MDM RA Deployment View

This subsection is to be completed in a future release. It is intended to show best-practice-based system topologies and implementation patterns of MDM, based on existing realizations of the MDM RA in the state.

4 MDM Implementation Guidelines

This section is intended to provide description of challenges and pitfalls in the MDM domain and basic guidelines that can help to successfully implement an MDM solution. The section is expected to grow over time in subsequent releases of the document, based on internal feedback and available research and surveys.

4.1 MDM Implementation Lessons

A number of practitioners shared their lessons learned when implementing MDM solutions. The following table summarizing common MDM implementation-related lessons learned is based on [2] and [6].

Table 4-1 Common MDM Implementation Lessons

Area	Lesson
Use MDM to Support Business Needs	<ul style="list-style-type: none"> • Remember that MDM is about the implementation of a business strategy - it's not an academic IT exercise. • Build a business case based on a tangible return on investment that gets support from the executives. • Institute the proper data stewardship and management policies and procedures at corporate and line-of-business levels to ensure a high-quality master data asset. • Assess the use of commonly used information objects, collections of valid data values, and explicit and implicit business rules in the range of applications across the enterprise. • Identify core information objects relevant to business success that are used in different application data sets that would benefit from centralization.
Thinking Big, Starting Small	<ul style="list-style-type: none"> • Think and plan for big MDM, but start small and deliver incremental benefits quickly. • Implement a data governance and data stewardship regime that is led by people with business experience. • Identify and compare the business needs of different parts of the enterprise.
Analyzing Data	<ul style="list-style-type: none"> • Don't assume anything about your data - perform a data audit to gain a complete understanding and plan remedial action. • Be alert to data quality issues such as erroneous, missing, default or dummy data. • Create business rules to validate and improve the data • Instantiate a standardized model for integrating and managing key information objects. • Manage collected and discovered metadata as an accessible,

	browsable resource, and use the metadata to facilitate consolidation. <ul style="list-style-type: none"> • Collect data from candidate data sources, evaluate how different data instances refer to the same real-world entities, and create a unique, consolidated view of each one. • Make use of external reference data and taxonomies to enrich your data and accelerate delivery.
Providing Access to Data	<ul style="list-style-type: none"> • Provide methods for transparent access to the unified view of real-world data objects for both existing and newly developed business applications. • Deploy real-time data quality and duplicate checks on all data feeds - both batch and real time (via SOA). • Monitor key data quality performance indicators on an ongoing basis and publish results internally.

4.2 A Step-wise Approach for Introducing MDM

A number of reports and surveys have emphasized that introducing MDM solution is not just about technology, but rather it involves broader scope and activities that involve non-technical elements. The following table summarizes a step-wise approach to introducing MDM (as advocated in [5]), which emphasizes non-technical elements in successful adoption of an MDM solution.

Table 4-2 Step-wise Approach for Introducing MDM

Step	Notes
1. Identify the need and the objectives for MDM	Specific objectives may include: providing processes for data collection, integration, consolidation, quality assurance, and distribution to ensure data integrity, maintenance, and the application of information usage control mechanisms
2. Identify the organization's core data and processes that use it	<ul style="list-style-type: none"> • What are Master Data sets across all subsystems/applications? • What are the processes and services associated with the data?
3. Define the governance	Governance includes regulations, practices, procedures, data and concept ownerships, responsibilities and roles, and the descriptions of the roles – all of those at distinct levels: organizational level, support function level and data set level.
4. Define the maintenance	The processes that are needed for administrating and



processes needed	<p>maintaining master data include:</p> <ul style="list-style-type: none">• Responsibilities, methods and tools for collecting data, defining workflows and guidelines for reviewing data in the workflows, and appropriate instructions for users and administrators.• Common operational models (e.g. service level agreements) between responsible units had to be created. <p>An important facet is reduction of costly and error-prone manual administration and maintenance procedures.</p>
5. Define data standards	<p>Data standards define the content <i>and</i> the model of a master data set on an attribute level. The data model was perceived as an enabler for making changes in the business environment. Data standards should be defined for every master data set considered.</p>
6. Define metrics for MDM	<p>Identify both repeatable and non-recurring measures for MDM.</p>
7. Plan an architecture model for MDM	<p>MDM architecture contains information about the applications involved, data flows between them, systems and data administration practices and points (centralized vs. decentralized), potential new acquisitions, and data security and data privacy issues.</p>
8. Plan training and communication with all stakeholders	<p>Motivation, objectives, master data criteria and the common data sets are typically recognized as being the most important issues which need to be communicated.</p>
9. Create a road-map for MDM development	<p>The road-map should be in agreement with longer term organization's strategy and long term strategy pertaining to MDM.</p>
10. Define MDM application characteristics	<p>For all master data sets involved, consider functional and operational characteristics of all components involved in supporting a given function or a business process, in order to determine optimal MDM usage scenarios – which may differ depending on the function or data set.</p>

5 Glossary

Authoritative Source is a trusted source of master data (or information in general). An authoritative source may be an acknowledged system of record or a system of reference.

Data – in context of the MDM, this is information about the domain in which the enterprise operates or information required for everyday operations and for adequate decision making.

Master Data are data referring to core business entities a company uses repeatedly across many business processes and systems, which captures the key things that all parts of an organization must agree on, both in meaning and in usage.

Master Data Domains specifies the attributes, relationships, and hierarchies that are relevant to one or more master data domains.

Master Data Management is a combination of processes, applications and technologies that consolidates, cleans, augments, organizes and distributes enterprise master data to applications, business processes, and analytical tools with widespread use in the organization.

Reference Architecture models the abstract architectural elements in the domain independent of the technologies, protocols, and products that are used to implement the domain.

Service Component is an actual application, program or subsystem providing implementation of a Service treated as a contract and performing specific responsibilities.

6 References and Bibliography

State and Federal Documents

- A. State of California, California State Information Technology Strategic Plan, 2013 Update
- B. State of California, California Enterprise Architecture Framework, Version 2.0

Books and Papers

- 1. D. Loshin, *Master data management*. Morgan Kaufmann, 2010.
- 2. A. Berson and L. Dubov, *Master Data Management and Data Governance, 2/E*. McGraw-Hill Osborne Media, 2010.
- 3. Dreibelbis et al., *Enterprise Master Data Management: An SOA Approach to Managing Core Information*, IBM Press, 2008.
- 4. Cleven, A.; Wortmann, F., "Uncovering Four Strategies to Approach Master Data Management," *System Sciences (HICSS), 2010 43rd Hawaii International Conference*, vol., no., pp.1,10, 5-8 Jan. 2010, doi: 10.1109/HICSS.2010.488
- 5. Vilminko-Heikkinen, R. ; Pekkola, S., Establishing an Organization's Master Data Management Function: A Stepwise Approach, *System Sciences (HICSS), 2013 46th Hawaii International Conference*, Publication Year: 2013, pp. 4719 – 4728, doi: 10.1109/HICSS.2013.205
- 6. S. Tuck, Is MDM the route to the Holy Grail?, *Journal of Database Marketing & Customer Strategy Management*, vol. 15, no. 4, pp. 218–220, 2008.
- 7. Dreibelbis, Information service patterns, Part 4: Master Data Management architecture patterns, 2007.
- 8. L. Menet, M. Lamolle, and A. Zerdazi, Managing master data with XML schema and UML, in *Advanced Information Systems for Enterprises, 2008. IWAISE'08. International Workshop on*, 2008, pp. 53–59.
- 9. Key Issues for Master Data Management, 7 February 2011, Gartner.
- 10. The Relationship Between Master Data Management and Enterprise Information Architecture, 19 February 2010, Gartner.
- 11. Magic Quadrant for Master Data Management of Product Data Solutions, 21 November 2012, Gartner.

Referenced Web Sites and Resources

- a. Gartner IT Glossary, <http://www.gartner.com/it-glossary>



7 Document History

Table 7-1 Document History

Release	Description	Date
Version 1.0 Draft	Initial creation	6/11/2013
Version 1.0 Second Draft	Revised based on internal review comments	6/21/2013
Version 1.0 Final Draft	Addressed EAC review comments	10/21/2013
Version 1.0 Final	Final version	01/02/2014