An introduction to the Master Data Management Reference Architecture

Martin Oberhofer
Allen Dreibelbis

April 24, 2008

Get a short introduction to the Master Data Management Reference Architecture for the enterprise which supports implementing multiform Master Data Management (MDM). Learn about the key concepts that drive the design of the MDM Reference Architecture and Logical System Architecture and see how to map the relevant IBM® Information Management software products to the core components of the Master Data Management Reference Architecture. Each product is briefly introduced, and in the Resources section of this paper you will find a wealth of additional information for reference. Finally, you will also learn about an upcoming book, Enterprise Master Data Management: An SOA Approach Managing Core Information, that describes the MDM Reference Architecture in full detail as well as other topics related to MDM.

Introduction

In many relevant business processes, entities like customers, products, accounts, contracts and locations play a central role. These entities are known as master data and many companies suffer today from low-quality master data scattered across the enterprise in various application silos. Improving master data quality and managing it more efficiently to optimize business processes is known as Master Data Management (MDM). An example would be the optimization of the New Product Introduction (NPI) business process by applying MDM for the product master data domain.

A great deal of material is available that describes what MDM is and why it is useful. Therefore we shift our attention to something which is of particular relevance for IT architects designing a MDM solution: The MDM Reference Architecture. However, before diving into the MDM Reference Architecture, it is important to define a bit of terminology first.
Terminology
First, it is important to understand methods of use. For the various master data domains such as customer, product, account, contract, or location, there are different patterns that describe how these master data entities are created, maintained, and used. The different, distinct types of use are called the *methods of use*. The different methods of use include:

- **Collaborative method of use**: Collaboration means that multiple users, usually in different roles, participate in the same process on a master data entity. A typical example would be the collaborative authoring of product master data where item specialists, brand category managers, pricing specialists and translators collaborate to author the definition of a new product. Key requirements of collaborative method of use are workflow support with check-in/check-out functions, support for relationships, and product hierarchy management. From a security perspective, attribute-level granularity of authorization privileges across all functions such as workflow, relationship and hierarchy management must be available for implementation.

- **Operational method of use**: This method of use is important when an MDM System has to function as an Online Transaction Processing (OLTP) server. Typically, 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 such as "New Account Opening". The MDM services are often used in the context of an SOA and need to be accessible through a variety of interfaces. MDM Systems supporting this method of use might have the need to support several hundred transactions per second on millions of master data records. For more information about the operational method of use, please see the Related topics section.

- **Analytical method of use**: This method of use has three known sub-types:
  - **Identity analytics**: This sub-type is usually encountered when there is a need to determine or verify an identity and discover hidden relationships.
  - **Analytics on master data**: Here, an MDM System needs to answer questions such as "How many new customers did I receive over the last day?" or "How many customers changed their address in the last week?"
  - **Analytics integration with data warehouses**: First, an MDM System provides master data to the data warehouse for accuracy improvements in the data warehouse environment. In a second step in this sub-type of the analytical method of use, insight gained in the data warehouse is made actionable by feeding it back to the MDM System for use in the IT landscape. An example of this analytical method of use is to persist computed customer profitability metrics and customer potential metrics in the MDM System, so that, from there, this insight can be leveraged in all front and back office systems.

Master data typically is scattered throughout the enterprise in multiple source systems which makes it inconsistent, incomplete and of poor quality. An MDM System is implemented to resolve these types of problems and requires the ability to distribute master data to various applications in a heterogeneous environment. The distribution of master data to other systems must fulfill a myriad of different business and technical requirements. These requirements also change over time, so the introduction of MDM within an enterprise tends to be, in many cases, a journey rather
than a one-time project. These are the reasons why, in addition to the various methods of use described above, there are also different implementation styles to accommodate the variety of requirements. Often an enterprise starts with one style and evolves their implementation to continue driving business value to the organization. The three styles are:

- **Registry style**: This style provides a read-only view to master data for downstream systems which need to read but not modify master data. This implementation style is useful to remove duplicates and provide (in many cases federated) a consistent access path to master data. The data in the MDM System is often only a thin slice of all the master data attributes which are required to enforce uniqueness and cross-reference information to the application system that holds the complete master data record. In this scenario, all attributes of the master data attributes remain with low quality without harmonization in the application systems except for the attributes persisted in the MDM System. Thus, the master data is neither consistent nor complete regarding all attributes in the MDM System. The advantage of this style is that it is usually quick to deploy and with lower cost compared to the other styles. Also, there is less intrusion into the application systems providing read-only views to all master data records in the IT landscape.

- **Coexistence style**: This style fully materializes all master data attributes in the MDM System. Authoring of master data can happen in the MDM System as well as in the application systems. From a completeness perspective, all attributes are there. However, from a consistency perspective, only convergent consistency is given. The reason for this is that there is a delay in the synchronization of updates to master data in the application systems distributed to the MDM System. This means, consistency is pending. The smaller the window of propagation, the more this implementation style moves towards absolute consistency. The cost of deploying this style is higher because all attributes of the master data model need to be harmonized and cleansed before loaded into the MDM System which makes the master data integration phase more costly. Also, the synchronization between the MDM Systems and application systems changing master data is not free. However, there are multiple benefits of this approach that are not possible with the Registry Style implementation: The master data quality is significantly improved. The access is usually quicker because there is no need for federation anymore. Workflows for collaborative authoring of master data can be deployed much easier. Reporting on master data is easier – now all master data attributes are in a single place.

- **Transaction style**: With this style, master data is consistent, accurate and complete at all times. The key difference to the Coexistence Style is that both read and write operations on master data are now done through the MDM System. Achieving this means that all applications with the need to change master data invoke the MDM services offered by the MDM System to do so. As a result, absolute consistency on master data is achieved because propagation of changed master data causing delay no longer exists. Deploying an MDM solution with this style might require deep intrusion into the application systems intercepting business transactions in such a way that they interact with the MDM System for master data changes or the deployment of global transaction mechanism such as a two-phase commit infrastructure.
A Reference Architecture for MDM needs to be able to support all methods of use and all implementation styles described. These three implementation styles are described in greater detail in other documents (see Related topics).

What is a reference architecture?

Reference architectures are an abstraction of multiple solution architectures that have been designed and successfully deployed to address the same types of business problems. Reference architectures incorporate the knowledge, patterns, and best practices gained from those implementations into the reference architecture. There are reference architectures that are cross-industry and others that might be industry-specific. Reference architectures provide detailed architectural information in a common format such that solutions can be repeatedly designed and deployed in a consistent, high-quality, supportable fashion. Reference architectures describe the major foundational components such as architecture building blocks for an end-to-end solution architecture. Early in the analysis and design stage of a solution, it is common for IT architects to search for reference architectures that can be used as input to design the solution architecture. They provide a framework for scope identification, gap assessment, and risk assessment to develop a roadmap to design and implement a solution.

Using a reference architectures has the following benefits:

- **Separation of concerns**: A good reference architecture uses components which are built with the separation of concern principle. This separation of concerns means you can change one component with zero to minimal impact on other components. As a result, you have a flexible and extensible infrastructure.
- **Risk mitigation**: Many implementations and deployments are done without the availability of a reference architecture for guidance. Thus, if a new project in the same domain needs to be done and a reference architecture can be used as guidance to codify the key concepts and capabilities, risks are mitigated because a proven architectural foundation can be re-used and adopted to the current project needs.
- **Cost reduction**: Since the development of the solution architecture doesn’t need to start from scratch, solution development costs are reduced. Usually, critical architectural decisions require time, several rounds of requirement discussion, alternative consideration, and the like. A lot of this time can be saved by using a reference architecture as guidance thus reducing costs.
- **Simplify decision making**: The business view on a reference architecture outlines what benefits could be derived by selecting a solution based upon the reference architecture.
- **Improved deployment speed**: The description of a reference architecture also outlines key principles, architecture decisions, deployment scenarios and guidance for developing a solution. It provides examples of architecture building blocks and components that can assist in the selection of software products and interoperability requirements between products or applications. This improves the overall speed for the deployment of a solution.
MDM Reference Architecture

This section gives you a brief introduction to the MDM Reference Architecture (RA). A full description with a complete and detailed component descriptions followed by component interaction walkthroughs can be found in the Related topics section.

The MDM Reference Architecture is an industry- and product-agnostic reference architecture that supports implementing the multiple methods of use (collaborative, operational, analytical) for MDM and multiple implementation styles (registry, coexistence, transaction style). It enables the ability to design business solutions incorporating MDM capabilities. By industry-agnostic, we mean that the reference architecture incorporates the knowledge, best practices, and patterns discovered through review and analysis of how MDM was deployed in multiple Customer Data Integration and Product Information Management Solutions spanning multiple industry sectors. We have abstracted the knowledge and patterns from those implementations to develop the Master Data Management RA as a way to describe MDM as a MDM Solution within the enterprise that implements MDM capabilities.

An MDM solution derived from the MDM RA enables an enterprise to govern, maintain, use, and analyze complete, contextual, and accurate master data for all stakeholders, users, and applications, across and beyond the enterprise. The identification of stakeholders and users will be different based upon the industry sector and background of the reader. But for the purpose of this discussion, stakeholders consist of line of business (LOB) system users and data analysts within the enterprise, and stakeholders such as trading partners and agents that have extended relationships with the enterprise. A key concept about MDM is that implementing an MDM solution is more than maintaining a central authoritative repository of master data within the enterprise. MDM provides the following benefits to the enterprise:

- Establishes the ability to generate operations to implement master data governance policies to manage and control the quality of master data
- Establishes data standards and provides for the cleansing of master data being used in current operations to improve data quality and improve consistency for use in operational environments across the enterprise
- Delivers business value by standardizing the way that master data is used across an enterprise, treating master data as a unique corporate asset bridging structured and unstructured data
- Provides the authoritative source of master data for new and existing applications, and establishes guidelines for the lifecycle management of master data
- Provides high-value actionable services over the data, delivering business value by detecting and generating business operations derived from changes that occur to a master data entity during its lifecycle

Architecture principles

An architecture principle is a comprehensive and fundamental law, doctrine, or assumption that provides overarching guidance for the development of a solution. A good architecture principle will not be outdated by advancing technology and has objective reasons for advancing it instead of
alternatives. The following principles are core architecture principles that should be considered for guiding the development of an MDM solution. The principles are:

- The MDM solution should provide the ability to decouple information from enterprise applications and processes to make it available as a strategic asset for use by the enterprise. This is a fundamental concept of Information on Demand founded upon Service Oriented Principles to deliver information at the right time in the right context to the right application or user.
- The MDM solution should provide the enterprise with an authoritative source for master data that manages information integrity and controls the distribution of master data across the enterprise in a standardized way that enables reuse. The primary motivation for this principle is to centralize the management of master data to reduce data management costs and improve the accuracy and completeness of that data.
- The MDM solution should provide the flexibility to accommodate changes to master data schema, business requirements and regulations, and support the addition of new master data. This improves the ability of a business to quickly respond to business changes that may require the addition of new master data elements or changes to existing master data.
- The MDM solution should be designed with the highest regard to preserve the ownership of data, integrity and security of the data from the time it is entered into the system until retention of the data is no longer required. The objective of this principle is to ensure that core business data that is critical to the success of the enterprise will be secure and to comply with privacy laws and regulations.
- The MDM solution should be based upon industry-accepted open computing standards to support the use of multiple technologies and techniques for interoperability with external systems and systems within the enterprise. This will guide development of the architecture to remain open and flexible so it can easily integrate with a variety of vendor software that may already exist within the enterprise and any future unknown technologies.
- The MDM solution should be based upon an architectural framework and reusable services that can leverage existing technologies within the enterprise. This principle guides the architectural decisions to leverage existing investments in technologies such as those that facilitate connectivity and interoperability or information integration where it makes sense in order to implement a MDM Solution.
- The MDM solution should provide the ability to incrementally implement an MDM solution so that a MDM solution can demonstrate immediate value.

MDM Logical Systems Architecture

MDM enables the ability to implement solutions that span many industry sectors, such as banking, insurance, retail, health care, telecommunications and government. The MDM Reference Architecture for the enterprise represents the capabilities to implement a resilient, adaptive architecture to enable and ensure high-performance and sustained value for the enterprise. The reference architecture provides a framework of components that can manage the lifecycle of master data, manage the quality and integrity of the data, make master data actionable, and provides stateless services to control the consumption and distribution of data. The design is guided by the core architecture principles described earlier, identifies key software components
within the architecture building blocks, and describes the basic responsibilities for each of those software components.

The MDM Logical Systems Architecture as seen in the following graphic is also designed with the flexibility to provide the capabilities needed to support multiple implementation styles and multiple MDM Architecture Hub patterns to support:

- A Transaction implementation style that may use SOA techniques to access master data services as part of an application transaction.
- A Coexistence implementation style that uses techniques such as publish and subscribe to harmonize data across the enterprise.
- A Registry implementation style that maintains a minimal amount of master data for each master data record and provides links to master data in the source systems.

The MDM Logical System Architecture is shown in Figure 1:

**Figure 1: MDM Logical System Architecture**

External participants may access and update master data through multiple delivery channels. Customers might access and update master data through business systems that provide self-service capabilities for shopping and online banking or through the use of telephony systems to access and update personal information. Supply chain data from suppliers, trading partners, and business partners participate in business-to-business transactions that involve the exchange of core master data entities such as customer and product data. Agents from multiple branch locations that conduct business on behalf of a company may access and update master data through a business system provided by that company or through a business-to-business transaction. Business system users update and query master data typically through the use of their respective business systems. Business systems request MDM Services as part of a business transaction or after the transaction has completed based upon the MDM method of use and implementation style. The decision to access MDM Services as part of a business transaction or after the system has completely processed the transaction is an implementation decision.
that should be based upon analysis of nonfunctional requirements such as performance and availability. Business systems and partner systems would request MDM Services to access master data through capabilities provided in the connectivity and interoperability layer.

Third-party data service providers such as Dun and Bradstreet, Acxiom, and Lexis Nexis can be accessed for additional information about a person or organization information to enrich master data maintained in the MDM System. Data from these organizations may be used to support the initial loading of master data into the MDM System or periodic updates, or may be used on a transactional basis depending on business requirements. Government agencies also provide watch lists required to support regulatory compliance, the war against terror and anti-money laundering.

The connectivity and interoperability layer facilitates business-to-business communications with trading and business partners, system-to-system communications within the enterprise, and communications to external data providers. Many IT organizations have realized the need to reduce the number of point-to-point interfaces between systems in order to reduce complexity and improve maintainability of the enterprise. They have implemented this layer using application integration techniques such as Enterprise Application Integration Hubs that support communications through the use of messaging, or have adopted the use of an enterprise service bus. MDM and Information Integration Services provide information services that can be invoked and choreographed through this layer. The connectivity and interoperability layer represents the enterprise service bus architectural construct or it can simply be thought of as a layer that provides choreography services, and synchronous and asynchronous integration capabilities such as message mediation and routing, publish and subscribe, FTP, and service-oriented integration through the use of Web services. Service integration represents that MDM and Information Integration Services can be requested directly from any business system without going through the connectivity and interoperability layer.

Just below the connectivity and interoperability layer in the center of the figure resides the MDM Services Architecture Building Bock. It consists of a set of MDM Services that are grouped into the following software components:

- **Interface Services** support a consistent entry point to request MDM Services through techniques such as messaging, method calls, Web services, and batch processing. The same MDM service should be invoked during batch processing that may be requested as part of a transaction in order to maintain and apply consistent business logic.
- **Lifecycle Management Services** manage the lifecycle of master data, provide CRUD (create, read, update, and delete) support for master data managed by the MDM System, and apply business logic based upon the context of that data. Data Quality Management Services are called by Lifecycle Management Services to enforce data quality rules and perform data cleansing, standardization, and reconciliation. MDM Event Management Services are called to detect any actions that should be triggered based upon business rules or data governance policies.
- **Hierarchy and Relationship Management Services** manage master data hierarchies, groupings, and relationships that have been defined for master data. These services may also
request Identity Analytics Services to discover relationships, such as those between people that are not obvious, and then store that information in the MDM System.

- **MDM Event Management Services** are used to make information actionable and trigger operations based upon events detected within the data. Events can be defined to support data governance policies, such as managing changes to critical data, based upon business rules or time and date scheduled.

- **Authoring Services** provide services to author, approve, manage, customize, and extend the definition of master data as well as the ability to add or modify instance master data, such as product, vendor, and supplier. These services support the MDM collaborative style of use and may be invoked as part of a collaborative workflow to complete the creation, updating, and approval of the information for definition or instance master data.

- **Data Quality Management Services** validate and enforce data quality rules, perform data standardization for both data values and structures, and perform data reconciliation. These services may request Information Integrity Services that are available from the Information Integration Services architecture building block.

- **Base services** are available to support security and privacy, search, audit logging, and workflow. Base services can be implemented to integrate with common enterprise components that support workflow, security, and audit logging.

- **The Master Data Repository** consists of master data, both instance and definition master data, metadata for the MDM System, and history data that records changes to master data. MDM Services can also be used to maintain and control the distribution of reference data that should be maintained at the global level for an organization.

Information Integration Services provide Information Integrity Services, ETL services, and EII services for federated query access to structured and unstructured data distributed over disparate data sources. Information Integrity Services include data profiling, analysis, cleansing, data standardization, and matching services. Data profiling and analysis services are critical for understanding the quality of master data across enterprise systems, and for defining data validation, data cleansing, matching, and standardization logic required to improve master data quality and consistency. MDM Data Quality Management Services can request Information Integrity Services to standardize, cleanse, and match master data updates received by the MDM System from a business system. ETL services support the initial and incremental extract, transform, and load of data from one or more source systems to meet the needs of one or more targets, such as a Data Warehouse and MDM System. The initial and incremental ETL processing to load large volumes of data is represented in the MDM Logical System Architecture Diagram at the bottom of the figure. Synchronous and asynchronous communication techniques to support the transporting of low volumes of changed data could occur within the Connectivity and Interoperability Layer.

The Analysis and Discovery Services architecture building block contains an Identity Analytics component that has analytical services that can determine the true identity of a person that might be trying to hide his or her identity. These services can also be used to discover non-obvious relationships between people, such as those that are part of the same household but have different names and address information, and between people and organizations. MDM Hierarchy and Relationship Management Services can request these services and then store the
results in the MDM Data Repository. In order for Identity Analytics Services to effectively discover relationships and a person's true identity, it may be necessary to load and analyze data from external data sources along with data from within the enterprise. Information Integration Services can be used to load data into the Identity Analytics component.

The Analysis and Discovery Services architecture building block also contains additional components that enable businesses to adapt to changing market dynamics and everyday operational disruptions. The Operational Intelligence component consists of services that provide event-based analytic functionality, the ability to perform scenario analysis, and sense and respond capability. It may utilize information and process models as input to implement the analytics capabilities for these services. The Query, Search, and Reporting component provides services that support ad hoc queries, reporting services, and online analytical processing (OLAP) capabilities for the reporting, analysis, and multidimensional modeling of business data. The Visualization component provides charting and graphing functionality, spatial dashboard reporting services such as for scorecard reporting, spatial analysis services, and rendering services for interaction with components that provide user presentation services.

The Content Management Services architecture building block provides services to capture, aggregate, and manage unstructured content in a variety of formats such as images, text documents, Web pages, spreadsheets, presentations, graphics, e-mail, video, and other multimedia. Content Management Services provide the ability to search, catalog, secure, manage, and store unstructured content and workflow services to support the creation, revision, approval, and publishing of content. Classification Services are used to identify new categories of content and create taxonomies for classifying enterprise content. Records Management Services manage the retention, access control and security, auditing and reporting, and ultimate disposition of business records. Storage Management Services provide for the policy-driven movement of content throughout the storage lifecycle and the ability to map content to the storage media type based on the overall value of the content and context of the business content. MDM Services would refer to content managed by Content Management Services and request these services to access unstructured content associated with master data, such as a customer, product, or account. For example, an application could request an MDM Service to get product master data from the MDM Data Repository, and then use the reference data returned from the MDM System to request a content management service to retrieve image data about the product.

The MDM Reference Architecture is designed to support the multiple MDM methods of use for multiple master data domains, to maintain cross-domain relationships, and to provide the required functionality to maintain an authoritative source of master data for the enterprise. The architecture is structured to be scalable, highly available, and extensible, and will provide the flexibility to integrate technology from a variety of vendors and integrate with future unknown systems.

Product mapping

When an IT architect defines a component model and proceeds to developing the operational model for a solution architecture, the IT architect analyzes which of the components need to be built and which can be taken of the shelf (bought). Thus, part of the operational modelling exercise includes a step to map software products. This section provides help for the IT architect
by mapping the MDM Logical System Architecture to IBM market leading software products. The products suggested can be used to implement the MDM Reference Architecture. We limit the scope of our discussion to core areas for building MDM Solutions which represent center stage examples for Information on Demand Solutions. The scope this article covers is shaded in yellow in Figure 2:

**Figure 2: Product mapping**

Below is a short summary of the capabilities of these products shown by component.

**Analysis and Discovery Services**

For this component, we cover only the group of services related to identity analytics. For identity analytics services, there are the products IBM Entity Analytic Solutions and IBM Global Name Recognition available (see the Related topics section).

The product IBM Entity Analytic Solutions covers the following areas:

- **Identity resolution**: Identity resolution is a task to resolve ambiguous, inconsistent information regarding an identity answering the question "Who is Who?". It determines if multiple records with name variations belong to the same or different identity. This is helpful for fraud detection, in homeland security environments for safety and security, or if a need to comply with privacy regulations is given.
- **Anonymous resolution**: The key feature of anonymous resolution is to share data on customers, organizations, citizens or employees between companies in such a way that privacy is protected at all times. Anonymous resolution determines “Who is Who” and “Who knows Who” anonymously. This allows companies to check identities against watch lists or to compare customer lists to determine the overlap in the customer base in case of a merger – all anonymously.
- **Relationship resolution**: Relationship resolution is useful to complete a party record regarding relationship information. It helps to compute “Who knows Who” to open new
business potential in uncovering relationships among customers, vendors, citizens or employees. In addition, potentially harmful relationships can be detected before the damage is done reducing losses and mitigating business risks leveraging alerts if a suspicious relationship is detected.

The IBM Global Name Recognition product addresses the following areas:

- **IBM Global Name Analytics**: This technology enables the sensitive management of multicultural data sets. By leveraging best-of-breed name recognition techniques the cultural background of names can be identified. Additionally, the gender can be determined if the name is primarily used for one sex.
- **IBM Global Name Reference Encyclopedia**: This is a huge database of names and knowledge about names collected from around the world. Name processing is based on global linguistic expertise using information about names, their meanings, typical spelling variations and gender associations in a culture-aware context in a fully automated fashion.
- **IBM Global Name Scoring**: In essence, identity checks on names are done more effectively with fuzzy clear text and phonetic search support. The technology enables multi-cultural name searches and eliminates vagueness and other inconsistencies in name transliterations.
- **IBM Global Name Management**: This bundle includes, for example, the components Global Name Analytics and Global Name Scoring. It has a foundation of nearly 1 billion names supporting parse, genderize and classify operations of multi-cultural names from over 200 countries. With this, the number of false positives can be reduced by measuring the degree of similarity between names.

**Master Data Management Services**

For this component, there are two products available (see the Related topics section for further details):

- IBM InfoSphere Master Data Management Server
- WebSphere Product Center

The IBM InfoSphere MDM Server (see Related topics) is the strategic platform to manage critical master data in various domains such as customer, product, account and location centrally with high master data quality. This product supports Multiform MDM (see Related topics). The MDM Server enables consistent, complete and accurate master data across the enterprise. As an authoritative source of master data, MDM Server delivers the single version of the “truth” to all channels and front and back office systems through multiple interfaces. The interfaces are designed for seamless integration with existing systems and the MDM Server delivers an industry leading MDM solution for performance and scalability supporting millions of master data records in a high-transaction environment.

Within MDM, multiple implementations styles such as Registry, Coexistence and Transaction implementation styles can be used for deployment (see Related topics for more information on these styles). Compared to other software solutions which provide only a view of master data in a registry style implementation, the MDM Server supports all three implementation styles. The MDM Server provides more than 800 business services supporting simple and complex operations
to manage and maintain master data out-of-the-box. This dramatically reduces processing costs and creates economies of scale to administer your master data effectively. Furthermore, Data Warehouse (DW), Customer Relationship Management (CRM) or Enterprise Resource Planning (ERP) deployment costs can be reduced dramatically since MDM Server can hide the complexity of the back office by providing a single integration point for all relevant master data entities. This streamlines middleware integration, reduces maintenance and upgrade costs, and simplifies architectural complexity.

Managing master data centrally provides a single location to track and manage customer privacy. Compliance with legal requirements can be managed in this central location with reduced costs and improved efficiency as well. Other benefits derived from a central MDM are:

- The ability to track complex hierarchy and relationship information in Business-to-Customer (B2C) and Business-to-Business (B2B) environments
- Cost savings by having a single place to enforce business rules and master data integrity rules on master data
- The ability to prevent duplicates in real-time, thus reducing operational costs by avoiding pricey, manual cleanup

The IBM InfoSphere MDM Server is based upon open standards and is designed to be implemented within an SOA. It has, for example, the right infrastructure to help companies to move to a customer-centric business model improving customer service. Finally, IBM InfoSphere MDM Server has a proven implementation record with large scale deployments in leading companies across all industries.

The WebSphere Product Center is a product information management solution linking product, location and trading partner (that is, supplier and resellers) in terms of product and trade information, which is typically scattered across the enterprise. By harmonizing and materializing this master data in WebSphere Product Center, various benefits can be derived, such as:

- Efficient distribution of product master data across countless customer, trading partner and employee touch points
- Deliver rich product information to Web sites and e-commerce applications, printing solutions used to generate product fliers and product catalogs
- Single point of integration with Global Data Synchronization Network further optimizing the exchange of product information with trading partners
- Enables category managers to more accurately report on product categories
- With leading features such as attribute inheritance in product hierarchies, product maintenance efforts can be reduced and use cases like micro-merchandising in a Direct Store Delivery Environment supported more efficiently

The WebSphere Product Center is designed with support for open standards such as JMS to simplify integration with other systems in an SOA environment. For easy import and export of product information, support for formats such as Microsoft® Excel is provided as well.

For both products, there are a number of industry-tailored solutions available — see Related topics for links to the MDM Solutions and WebSphere® Product Center industry solutions pages.
When deciding whether to use IBM InfoSphere MDM Sever or WebSphere Product Center, you should consider the following:

- If the solution requires the operational or analytical method of use in addition to any of the three implementation styles across multiple master data domains, then IBM InfoSphere MDM Server should be considered.
- If the solution is based upon the collaborative method of use with check-in and check-out services, full authoring capability, and a Coexistence implementation style, WebSphere Product Center should be considered.

**Information Integration Services**

The process of information integration requires the capabilities to understand, integrate, cleanse and transform data and content to deliver authoritative, consistent, timely and complete information to applications with support for data governance throughout the information life cycle. There is one product suite for information integration addressing the needs of the information integration services component: This is the IBM Information Server (more details can be found in the Related topics section). IBM Information Server is a revolutionary information integration platform that enables seamless information integration in complex IT environments. Furthermore, traditional information integration functions such as address standardization can now be exposed as services and woven into processes easily to improve data quality when data enters the system as part of a create customer business process for example. This enables SOA participation.

The IBM Information Server has many components as Figure 3 shows. We introduce the components in light green because they are of particular relevance when implementing MDM Solutions. The other components may or may not be used, such as the connectors to SAP and Siebel, which are obviously only needed if an application requires integration with those systems, but they are part of the base infrastructure in addition to the parallel processing engine.

**Figure 3: IBM Information Server**
The components are:

- **WebSphere Information Services Director**: With SOA adoption spreading, the demand to integrate information into business processes more efficiently rises. This component enables information integration functions to participate in an SOA world by delivering these services in an “always on” fashion. Delivering these services in this fashion allows the ability to integrate information integrity functions as services in complex business processes such as customer or product creation. The key features are:
  - It exposes functions as services with a couple mouse clicks with various bindings such as SOAP/HTTP for web services or Enterprise Java™ Beans (EJBs) for high-speed, direct Java Integration.
  - It provides a resilient infrastructure that supports fault tolerance, load balancing and true parallel execution satisfying the most demanding high availability requirements.
  - Due to the design architecture exploiting a pure J2EE infrastructure, it is capable to deliver secure and resilient data integration services to WebSphere Process Server and WebSphere Portal, other applications and databases.
  - A services catalog maintains all available services built on the MetaData Server infrastructure. The services catalog has an out-of-the-box integration with WebSphere Service Registry and Repository to seamlessly publish all services into the enterprise wide service repository.
  - This component is compliant with Web service standards and other open standards from organizations like W3C and the Java Community Process.

- **WebSphere Information Analyzer**: Before integrating master data from a variety of sources into an MDM System, the quality of the data and the source system data models need to be understood. For this task and also for similar tasks in other projects like data warehousing, ERP instance consolidation, and the like, WebSphere Information Analyzer can be used. Understanding data quality upfront enables a more accurate sizing of data integration projects thus reducing project risks. The key features are:
  - Data Model discovery and seamless storage of discovered data models in the shared MetaData Server infrastructure so that WebSphere QualityStage and WebSphere DataStage can leverage them.
  - Value distribution analysis
  - Cross-column analysis for foreign key discovery
  - Deep profiling capabilities

- **WebSphere Business Glossary**: Creating, managing and sharing an enterprise vocabulary and classification of terms is the key foundation that business users and technical users can efficiently communicate with each other. The key features are:
  - Management of terms and categories through a Web-based user interface. The terms organized in hierarchies represent the major information concepts in the enterprise.
  - Operational data stewards are usually members of a data governance team in an enterprise responsible for information assets. Managing profiles with support for import and export from external sources for operational data stewards is simplified with the leading administration capabilities of the WebSphere Business Glossary.
  - The WebSphere Business Glossary awakens metadata to life by enabling collaborative use of it across business users, data modelers, data profilers and ETL developers.
• It supports easy-to-use browsing with no need for training.

• **WebSphere QualityStage**: This component provides a powerful framework for developing and deploying data investigation, standardization, enrichment, probabilistic matching and survivorship operations. For example, name and address standardization for improving quality of customer master data are available. It has as key features:
  • A standardization engine
  • Embedded data dictionaries and rule sets
  • Interface to seamlessly integrate with postal verification service such as WAVES
  • Linear scalability
  • State of the art probabilistic matching and survivorship decision engine

• **WebSphere DataStage**: This component provides an industrial strength Extraction, Transformation and Loading (ETL) engine and is a core component of IBM Information Server. It enables the integration of enterprise information, regardless of the sources, targets and timeframes. When deploying an MDM Solution during the Master Data Integration (MDI) phase, master data harmonization from in heterogeneous IT environment is needed. Transforming the source data to the master data model used by the MDM System is usually a task for which WebSphere DataStage is well-suited with the following key features and benefits:
  • It is a powerful ETL solution which collects, integrates and transforms large volumes of data with simple and complex data structures.
  • Its parallel processing engine scales helps it process faster growing, large data volumes in continuously shrinking batch windows. It provides massive throughput of data through the transformations in typical batch processing operations.
  • The component supports real-time data integration. Thus WebSphere DataStage operates in real-time using the ability to capture messages from Message Oriented Middleware (MOM) queues using JMS or WebSphere MQ adapters to seamlessly combine data into conforming operational and historical analysis perspectives.
  • With the rich connectivity layer, this component can extract data from virtually any data source (for example, SAP, Siebel, relational databases and business intelligence systems such as SAS) and load it into virtually any target after transformation.
  • The component provides advanced support for development and administration. This reduces development efforts for new transformation jobs and simplifies administration -- thus reducing operational costs.

• **WebSphere MetadaServer and IBM MetaData Workbench**: The WebSphere MetaData Server is a shared infrastructure component for all components of IBM Information Server managing metadata. The IBM MetaData Workbench is a tool to work with metadata. It supports:
  • Data lineage to understand where data is originating from
  • Impact analysis to find out which functions in cleansing or transformation routines would be impacted if a data model is changed
  • Seamless integration of design and operational metadata enabling to see where data comes from and what processing was applied to it. If compliance with regulations such as Sarbanes-Oxley or Basel II is required, this is a key capability.
Conclusion

Every MDM project will have its own unique set of challenges and risks that need to be considered for the selection of software and the implementation strategy. The MDM Reference Architecture provides the basis for developing an MDM Solution for the enterprise that is based upon architecture patterns and best practices. Selecting the right software to meet both the tactical and long-term strategic business objectives is critical for achieving both the immediate and long-term business value of an MDM Solution.

The MDM Reference Architecture should be referenced as input for developing an MDM Solution for the enterprise. The MDM RA is designed to support the evolution of an MDM Solution to implement one or more MDM methods-of-use and accommodate multiple master data domains. Designing an architecture evolves through multiple stages of elaboration and specification, taking into account system distribution; nonfunctional requirements such as performance, reliability, and high availability; the use of specific products; the choice of middleware; and other technologies. One of the key drivers for the design of the MDM Reference Architecture is that it should be a scalable, highly available, adaptive and capable of supporting high performance. The implementation of an MDM Solution will always need to consider the existing IT environment, IT standards, enterprise architecture policies, and choice of software for the MDM System and for implementing Information Integration Services such as Information Integrity, ETL, and EII.

It is important to consider the long-term MDM strategy when selecting software for your MDM System. Designing and implementing an MDM System that will continue to deliver sustained business value to the enterprise requires the ability to support the multiple implementation styles. Therefore, open, easy to extend software products like the ones we described in the previous product mapping section ensure that the investment is well-done.

Book preview

Related topics

- **Operational MDM**: Discover how to use the operational method of use in MDM.
- "**Implementing a Transaction Hub MDM pattern using IBM InfoSphere Master Data Management Server**" (developerWorks, March 2008): Learn to use the Transaction Hub Master Data Management (MDM) pattern to implement your MDM systems.
- **Enterprise Master Data Management: An SOA Approach for Managing Core Information** (IBM Press, June 2008): This book covers many of the key aspects for understanding what is meant by MDM, the business value of MDM and how to architect an Enterprise Master Data Management Solution.
- **IBM Entity Analytics Solution homepage**: Access the IBM Identity Resolution, IBM Anonymous Resolution and IBM Relationship Resolution technologies.
- **Global Name Recognition homepage**: Get the IBM Global Name Analytics, IBM Global Name Reference Encyclopedia, IBM Global Name Scoring, and IBM Global Name Management technologies.
- **Introduction to Multiform Master Data Management**: Harness more value from your master data.
- **Master Data Management Solutions**: Get information solutions you can trust!
- **WebSphere Product Center**: Get this product information management solution for building a consistent central repository.
- **WebSphere Product Center industry solutions**: Learn about retail and consumer packaged goods industry solutions from IBM and our business partners.
- **IBM MDM for SAP ERP using SOA** (whitepaper): Extend the value of SAP ERP with SOA and IBM Master Data Management.
- **IBM Information Server homepage**: Get the IBM Information Server and learn all about this new technology.
- **WebSphere Information Services Director**: Publish tasks as reusable services in a SOA. Take data integration logic built using IBM Information Server or IBM Information Server for System z and publish it as an "always on" service in a SOA.
- **WebSphere Information Analyzer**: Understand the structure, content and quality of your data sources.
- **IBM Metadata Workbench**: See a visual Web-based exploration of metadata generated and used by IBM Information Server.
- **WebSphere Business Glossary**: Create, search and manage a common vocabulary for business and IT users to collaborate with using Business Glossary, Business Glossary Browser and Business Glossary Anywhere.
- **WebSphere QualityStage**: Unlock the value of your technology investment by ensuring that decisions are based upon trusted information.
- **WebSphere DataStage**: Integrate data across multiple and high volumes of data sources and target applications.
- **Information Server Data Quality Module for SAP**: Learn about the Information Server Data Quality module for SAP.
• **SOA solutions using IBM Information Server - IBM Redbook**: Get an overview of IBM Information Server, its architecture, and processing flow. Learn about the development and implementation of a financial services business scenario that showcases the SOA capabilities of the IBM Information Server product and WebSphere Integration Developer.

• "**Information service pattern series**": Learn more about how to use the data federation, data consolidation, data cleansing and MDM architecture patterns in an SOA context.

• "**Configure and invoke Web services for WebSphere Customer Center**" (developerWorks, October 2007): Use Web services with security and global transactions and download a sample Web services client.

• **IBM Industry Models**: Leverage deep expertise and best practice by capturing industry knowledge in a form useable by both business and IT communities.

© Copyright IBM Corporation 2008

**Trademarks**
(www.ibm.com/developerworks/ibm/trademarks/)