Framework and Research Agenda for Master Data Management in Distributed Environments

Tomi Dahlberg, Jukka Heikkilä and Marikka Heikkilä
University of Jyväskylä
P.O. Box 35
FI-40014 UNIVERSITY OF JYVÄSKYLÄ
FINLAND, EUROPE
Tomi.K.M.Dahlberg@jyu.fi, Marikka.Heikkila@jyu.fi, jups@jyu.fi

Abstract. Master data is the foundation for relating business transactions with business entities such as customers, products, locations etc. In master data literature these entities are also referred to as domains. The integrity, availability and timeliness of master data both as single-domain and as multi-domain combinations is crucial for business transactions in general and especially to distributed eBusiness transactions in the network of stakeholders over the Internet, or in the cloud. Distributed environments set additional challenges for the management of master data. In this idea paper, we first describe master data in the context of various data types and processes, the responsibilities on master data management as well as other typical master data management practices used to improve master data quality. We combine them into a framework, which provides a generic background description over the master data phenomenon, that is, a classification over practical means to solve shortcomings in master data. We argue that although these means are able to support the improvement of master data quality and master data management they are insufficient to capture the underlying causes of master data problems. We then and therefore investigate master data management from the IS theoretical viewpoint and finally propose a research agenda for most critical issues in master data management. We suggest that enterprise architecting could serve as a coherent framework in identifying common and specific master data management research themes for global businesses and networked governments.

Keywords: Data Governance, Master Data management, MDM, master data organization, Enterprise Architecture.

Introduction

A recent survey conducted in Finland revealed several shortcomings in master data management (Dahlberg, 2010). This survey and other studies (e.g. Silvola et al, 2011) indicate that master data required to carry out business transactions is spread over multiple systems and databases in an organization and has become more fragmented with each new information system implementation.

 Typically, the organization becomes aware of the master data challenge when the organization engages in enterprise architecture work, in large scale IT projects such as an ERP implementation, in major migrations of the organization’s core information systems, or wants to increase the usage of eBusiness (Dahlberg 2010). The legacy systems are found to consist of numerous partly or totally overlapping data sets that were developed for some limited or specific purpose resulting in a tangled web of incoherent databases and data coding schemes. In addition to dominantly used narrow project focus, over the years conducted mergers and acquisitions with uncompleted integrations have added to the fragmentation of systems, databases and coding schemes of master data. For example, in Dahlberg’s (2010) survey one global company had 54 overlapping ERP systems in five continents. It is not rare that an organization’s systems have multiple instances of the same customer or product entity, or variants with only slight variations in the data. Multiple entries of the same data, missing, erroneous and conflicting data values cause inefficiencies into operational processes, cast doubts on the reliability of managerial reporting and analytics and lead to excess stocks, inability to gain economies of scale, poor customer relationship management etc. When organizations become involved in various kinds of business networks or consider moving their IT to clouds, there is even more urgent need for holistic management of master data.

With master data management (MDM) organizations aim to first improve and then to ensure the quality, consistency and accuracy of the master data. This requires that both technical solutions and organizational processes are managed (Cleven and Wortmann, 2010). Both are demanding, but there are perhaps even more challenges in the latter (Loshin,

---

1 Survey consisted of a series of interviews in 10 large organizations with cumulatively over 100 interviewees from senior executives to master data maintainers.
It appears that in many organizations nobody has responsibility over the content and hence quality of customer, product, vendor, etc. data or lacks the means to execute such a responsibility. In addition to technical solutions and organizational processes also master data development, master data quality improvement and information architecture appear as typical MDM measures. Poor master data management results in missing business and performance objectives, inadequately agreed data ownership, fragmented data management processes and lack of continuous data quality assurance (Dahlberg, 2010; Silvola et al., 2011).

Even though master data management is one of the most topical issues in information system discipline at the moment, research on MDM has only started (Silvola et al., 2011) and consequently vagueness characterizes concepts used. At the moment, there are a lot of shortcomings in master data quality and hence some studies describe these challenges (e.g. Silvola et al., 2011) and efforts taken to solve them (e.g. Otto and Reichhardt, 2011). Due to these reasons we felt necessary to describe master data in the context of various types of data and processes as well as the responsibilities on master data management and other typical master data management practices. We then combine these two dimensions – types of data and master data management perspectives - into a MDM framework to provide a generic background over the phenomenon and approaches used improve master data quality. This is done the next three sections.

Although the framework sums up the MDM phenomenon and approaches applied to remove the consequences of bad quality master data, we argue that a deeper understanding is needed to understand the underlying cause of the problems. Against this background we then describe and analyze master data management from the IS theoretical viewpoint and propose a research agenda for most critical issues in master data management. We argue that the ability to first understand and then to hopefully solve master data management challenges requires that one goes back to the origin of these challenges. It appears that master data – and data in general - started to fragment when the number of information systems grew and new systems were predominantly installed with narrow project, technical etc. focus. Consequently the information systems architecture – the ontological structure (Wand and Weber 1989, 1993, 2002) of information systems – became fragmented as well. We specifically propose that enterprise architecting and (ontological) information architecting could serve as a coherent basis in identifying common and specific master data management research themes for global businesses and networked governements.

The contributions of this paper are twofold. Firstly, the framework presented as the background for the IS theoretical analysis is a step forward in formalizing understanding on master data management issues and master data concepts. Secondly, the proposed research agenda based on ontological analysis of master data management sets a theoretically solid path for future research.

Types of data and processes

The types of data used in organizations consist of transactional data, master data, meta data and reference data (Cleven and Wortmann, 2010). Master data consists of data items that describe the core entities of an organization. They are typically persistent items of independent business domains, the status of which does not change too often. For example, the master data attributes of vendor, product and customer in an organization tend to change little over time. The weight, size and other attributes of a product are typically considered to remain unchanged throughout the lifecycle of a product. Even those master data attributes that change from time to time, such as standard unit price, are unchanged between the updates. The idea of master data is to enter and maintain data once and to transfer needed attributes to all tasks where such data is needed. This is called data inheritance. The total number of master data records is also usually rather stable when compared to the seasonal and other fluctuations of business transaction volumes.

Typical examples of master data domains are: parties (customers, employees, vendors), places (customer locations, office etc. sites) and things (accounts, contracts, documents products and services) (White et al, 2006; Cleven and Wortmann, 2010), where party is an abstract high level domain and customer is a concrete lower level domain. Most organizations have a limited number lower of level domains, usually around a dozen (Dahlberg, 2010). Each domain may have several data objects. For example a typical SAP ERP system installation contains approximately 150 master data objects in the domain of Management Accounting (MA) alone, such as currencies and payment terms (Dahlberg, 2010). Practitioners typically address master data quality on concrete domain and object levels.

In addition to persistence and rather constant volumes, master data is differentiated from transactional data by its independency of the entities. As an example, consider sales order (transactional data), which cannot exist without customer (master data), product (master data) and payment term (master data) (Cleven and Wortmann, 2010). Since key master data attributes typically act as the identifiers of data queries and the basis of sorting transactional data to perform various aggregations and calculations for reporting, the quality of master data has the highest quality requirements and should therefore be devoted a lot of attention to (Loshin, 2009). Since most business transactions are linked to several master data attributes at the same time, one of the challenges of master data management is the simultaneous
management of multiple domains – called multi-domain or domain neutral MDM - as opposed to single domain MDM such as Product Information Management (PIM).

Master data is typically also used across multiple business processes and functions. For example sales, delivery logistics, after sales and services, spare parts business, billing, accounts receivable and finance, and management through managerial and analytical reporting may all rely on customer data but have at the same time different needs and priorities. Furthermore, some processes may be cross-functional, for example order to cash, whereas other processes or activities are functional, for example recruiting of employees. Balancing the needs of cross-functional and single function activities is another master data management challenge closely related to the domains of master data.

**Master Data Management perspectives**

Master data management is defined by Smith and McKeen as: “*Master data management (MDM) is an application-independent process which describes, owns and manages core business data entities. It ensures the consistency and accuracy of these data by providing a single set of guidelines for their management and thereby creates a common view of key company data, which may or may not be held in a common data source*” (2008, pp. 65-66).

Joshi (2007) proposed that the eight steps outlined below should be followed to execute MDM successfully:

- Define the master data flow
- Identify the sources and consumers of master data
- Collect business metadata
- Define the master data model
- Define the needed functional and operation characteristics of the MDM tool
- Merge the source data to create a master data list or element
- Collect and maintain the technical and business rules metadata
- Publish the master data or modify the consuming applications

Otto and Reichert (2011) listed the activities of the topmost concerns of MDM shown in Figure 1 and called their figure “MDM tasks”. The activities outlined by them concentrate on managing data assets strategically, agreeing upon and maintaining standards and guidelines for design and on handling changes as projects - all this in line with the support from the management.

<table>
<thead>
<tr>
<th>Response Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application management for a master data management software</td>
</tr>
<tr>
<td>Business user support</td>
</tr>
<tr>
<td>Development and maintenance of the master data strategy</td>
</tr>
<tr>
<td>Development and maintenance of standards and guidelines</td>
</tr>
<tr>
<td>Master data lifecycle activities (e.g. creation, maintenance, deactivation)</td>
</tr>
<tr>
<td>Measurement and reporting of master data quality</td>
</tr>
<tr>
<td>Project support</td>
</tr>
<tr>
<td>Training of users</td>
</tr>
<tr>
<td>Other</td>
</tr>
</tbody>
</table>

*Figure 1. MDM Tasks by Otto and Reichert (2011)*

As a whole, master data management appears to break down into many perspectives. Dahlberg (2010) has identified the following five perspectives, which are used to improve MDM and master data quality:

**Management perspective:** This perspective addresses the governance and management aspects of master data. For
example, Dahlberg (2010) classified 28 master data governance and management issues identified by him into seven managerial task categories by adapting Gartner Group’s (2006) “Seven Building Block Model for Enterprise Information Management”. Categories, shown also in Figure 2 in the Framework section, are: MDM objectives, MDM road-map, MDM governance, MDM organization, MDM processes, MDM infrastructure and MDM reporting.

**Information architecture perspective**: Ability to use and manage master data requires that master data is modeled. A standard data modeling approach with the overall model, conceptual model, logical model and physical model levels could probably be applied as master data appears similar enough to other types of data. Since same master data entities are used in multiple processes and/or organizational functions it is also necessary to model the data flows of master data to cover the inheritance of this data. When a new master data record is created some of the data attributes could be controlled against accepted reference data values. For example, the country of a customer could be selected from a drop list of countries. The creation of some other data attributes could be controlled with reference data rules. Reference data management controls and meta data related to master data needs also to be modeled and managed (Cleven and Wortmann, 2010).

**Master data quality perspective**: The consequences of badly executed MDM typically appear as the poor quality of data. Bad data appears as duplicates, missing attribute values and data value conflicts. Improvements of master data quality require that data is analyzed and cleaned up in a planned way. Migration and harmonization may also be used to improve data quality. When quality-improving changes are made to master data, it is necessary to secure the continuity of data, for example via audit trail and data inheritance checks. One mean to accomplish this is to use so called delta file approach. Setting up rules and other controls to ensure the quality of master data is another data quality management measure. Furthermore, it is necessary to manage the information security of master data.

**Technology management perspective**: Ideally a specific master data item, such as a customer, is entered and maintained only once and made available to all SOA components / WEB services, legacy applications and other IT components, which use that data. At the moment, very few organizations, however, appear to have clarity on which of their alternative databases serves as the master database – even when the database has a MDM module label on it.

**The Framework Combining Data Types and MDM Perspectives**

When we combine the categories of data types and the types of processes where master data is used with MDM perspectives, the result is the framework shown in Figure 2. The columns of the framework categorize the number of involved master data domains into single and multi-domain categories and functions where master data is used into single-function and multi-function process categories. The rows of the framework describe the various MDM perspectives on how to manage and improve the quality of master data. Thus the framework summarizes and organizes research findings and approached discussed above and provides a holistic description of master data management as a whole.

It is noteworthy, that both the columns and the rows of the framework describe and describe what master data is and how the quality of master data could be improved. The framework, however, is not able to explain why the quality of master data has become poor and thus how to prevent that happening again. In other words, the framework describes what means in various contexts one could adopt to remove the symptoms of master data problems but does not explain what are the underlying reasons for those problems.
In the light of literature reviewed, MDM appears to be a topic of its own (Smith & McKeen, 2008; Otto & Reichert, 2010) like master data is ideally separate from the transactional data and other concerns of an IS (Cleven and Wortmann, 2010). Yet, any data set or database serves a bigger whole, information systems representing and tracking the behaviour of a purposeful realworld system. The definition of an information system (Iivari et al., 1998) underlines that this realworld system is typically a human activity system, representing vested interests and dynamic interplay of the socially constructed concepts about the stakeholders and their behaviour for the purpose which the IS is to serve. This can be also considered from a more realist stance (ibid.): An IS describes facts, especially the relevant facts of the technical system derived from the stationary and stable real world. It is evident that the present way of looking into master data management and master data quality builds on the realist stance, and at the same time omits largely the idealist, or constructivist interpretation (ibid.). We believe and claim that this is the root cause for those master data problems we have depicted in previous chapters. That is, efforts to model and solve master data quality and master data management problems are done as if these problems were stemming from a stable, predictable, uniform and causal world, when the actual problems are more deeply and profoundly related to the fundamental changes in the real world and our perceptions and representations of it.

In master data management practice, the realist stance could lead to harmonization efforts in the name of rationality where the resulting compromise satisfies nobody and cannot be used. For example, a company could try to harmonize processes and master data used in them to the extent that it serves none of the actual processes. Similarly adopting only the extreme reflective stance could lead to a situation where master data is systematically overruled. For example, the unit price of a product inherited from master data could always be overruled in use situations by users, which will make that data useless very rapidly.

As Goldkuhl and Lyytinen stated already in (1982), “...information systems can be viewed as “technical systems with social implications” or “social systems only technically implemented”. Both views are valid, eventhough the contemporary IS researchers distinguish themselves from the rest of the computing and engineering researcher community with the previous statement, as they emphasize the socially constructed nature of the organizational complexities including information systems (Iivari et al., 1998).
We illustrate our argumentation in more detail with the help of the ontological description of the structure of an IS (Wand & Weber, 1989) and its requirements for an information systems methodology, including the subset of databases. According to this approach any IS methodology must represent the ontology of the system with a minimum set of constructs and to map and track the state changes for various purposes of the real world system (Wand & Weber, 1993), including the reporting requirement for various stakeholders of an IS for their anticipated and ad-hoc purposes. In master data contexts, this means that the various user needs of master data in relation to other data types as well as other relevant characteristics of the real world are understood. Only then is an IS development model fulfilling its role properly, that is, it has the necessary representational means for expressing the ontology of the domain completely and clearly but also without overloading constructs and extra constructs which result in redundancy. The goodness of this representational model does not exist in vacuum, but must be able to serve first the ‘needs’ of the state-tracking model, against which the states of the real world are reflected in the IS, and secondly, the ‘needs’ of the reporting, where the states of an information system can be reported for the stakeholders to reflect the corresponding state changes of the real world. The first requirement refers to creating and maintaining all the attributes of master data. Correspondingly, the second requirement describes how the relevant attributes of master data are inherited to the various use situations of master data. The attribute needs of use situation could be totally different. These two principles are able to capture both the holistic nature of master data management and the specific use situations of this data.

Some of the master data problems described in the previous chapters can be restated according to the IS theoretical explanation as follows:

- There are cases where the **constructs of the ontology are incomplete** because some fundamental concepts are missing. A real case example of this problem is a company that had multiple identities for its vendors, each new bank contact creating a new vendor record (Dahlberg, 2010). The data model or its implementation did not contain sufficient structure to capture this feature in vendor data.

- The **constructs of the ontology are overloaded** because planned and implemented concepts cannot distinguish the subtle differences of meanings in the real world constructs. For example, a customer may have multiple roles depending on who is in contact with the customer. If the attributes of customer master data are not able to capture these different roles, the content of the data could reflect the most typical role or could lead to duplicates.

- The **constructs of the ontology are redundant**, or become redundant, e.g., when information system instances are connected to serve wider geographical activities or entire global business. For example, it is not uncommon that the same product has two different codes in two different markets. This might just reflect differences in local coding practices forced by local authorities or voluntarily followed by all parties in that local market. Very few data items have globally standardized codes. In this kind of a situation, merging the instances with simultaneous harmonization is not a working option. The consequences could be quite severe if the product is a component in a machine, which consists of tens of thousands of components and those data structures are lost.

- **Mappings are incomplete, or cannot reflect changes in the real world**, because the real world is not or is no longer steady and stable, but is in constant turmoil, at the same time as master data does not change accordingly, is not managed properly or meta data does not reflect changes in a systematic manner. For example, standard unit price is a typical product master data attribute. Standard unit price is often checked at certain intervals and may include a rough fixed cost element. Should the environment change suddenly, so that the fixed cost element changes significantly or prices fluctuate constantly, these practices become insufficient to cope with the real world.

- **Reporting needs are unanticipated or not taken into account** in the original design of master data coding, data repositories, information systems etc. There can be several reasons for this. Necessary classification codes were not considered or registered and hence are not connected to transactional data. The design methodology could also have been inappropriate. For example, aims to make public master data repositories available to open use with simultaneous protection of privacy data may lead to a dead-end.

- **Forced limitations on ontological clarity and increasing construct redundancy** This may happen if the constructs cannot be designed freely, but must follow socially constructed conventions, standards and regulations limiting the ontological clarity and increasing construct redundancy. things are almost, but not quite the same, to put it colloquially.

Fundamentally, the separation of master data concerns from the real world social systems in master data management practices – as summarized in Figure 2 in the previous section - leads to the dilemma of not describing master data in the the real world context, but rather trying to improve the management of “socially implemented technical systems”.

The backdrop provided above provides a solid foundation for a research agenda in the area of master data quality and its management. As a whole we propose a more strategic approach into the design and active maintenance of the master data. Furthermore, we urge that in the requirements crafting and design of master data constructs as well as in the problem solving of current challenges researchers and practitioner should not just look at the logical technical designs, but also at the foreseeable changes in socially constructed concepts and at the effects of such changes. In this context, management means those structures, process and mechanisms which are needed to ensure the quality of
designs, state-tracking and reporting requirements collected from socially relevant sources, that is from all stakeholders of purposeful systems with related master data.

**Research Agenda for Master Data Management**

We started this article by looking at various types of data and processes where master data is used. We then discussed the current meaning of master data management and its various perspectives, which are used to improve master data quality and MDM. We noticed that MDM practices are good to describe current status but insufficient to understand the underlying nature of the master data management. The necessary depth of analysis was found from the IS theoretical viewpoint, especially from the ontological structure of information systems. Against this background we suggest that enterprise architecting is able to serve as a coherent framework in identifying common and specific master data management research themes for global businesses and networked governments. Below we offer selected research questions for future research:

- How should interoperability requirements be derived from stakeholder analysis, business process modeling and enterprise architecting?
- How could security, access rights, and integrity be ensured? What kinds of consents are needed to use and combine master data from various sources? What kinds of Opt-in or opt-out arrangements are relevant?
- Data ownership is a business issue but has not been considered in MDM practice clearly enough. What are the means to accomplish that and to secure the participation of all relevant stakeholders?
- From enterprise architecture perspective componentization is the way to go. (Master) data cannot, however, be encapsuled when it is processed. What is the role of master data in SOA and other component based architectures?
- How to adapt IT governance principles and build best practices for MDM? How could the Evaluate, Direct and Monitor (EDM) presented in ISO/IEC 38500 model and Plan, Do, Check and Act (PDCA) be applied in MDM? What project management practices are relevant for MDM?
- How should we treat master data in clouds, outsourcing and off-shoring? What are the best ways to to map 'good' designs, business needs and the views of the stakeholders includig privacy and security?

**References**


