ENTERPRISE MASTER DATA ARCHITECTURE: DESIGN DECISIONS AND OPTIONS

(Research-in-Progress)

Boris Otto
Institute of Information Management, University of St. Gallen
boris.otto@unisg.ch

Alexander Schmidt
Institute of Information Management, University of St. Gallen
alexander.schmidt@unisg.ch

Abstract: The enterprise-wide management of master data is a prerequisite for companies to meet strategic business requirements such as compliance to regulatory requirements, integrated customer management, and global business process integration. Among others, this demands systematic design of the enterprise master data architecture. The current state-of-the-art, however, does not provide sufficient guidance for practitioners as it does not specify concrete design decisions they have to make and to the design options of which they can choose with regard to the master data architecture. This paper aims at contributing to this gap. It reports on the findings of three case studies and uses morphological analysis to structure design decisions and options for the management of an enterprise master data architecture.

Key Words: Enterprise Master Data Architecture, Master Data Management, Design Decisions, Case Study Research

INTRODUCTION

Motivation

The management of master data on an enterprise-wide level has received increasing attention in the practitioners' community, lately [6, p. 2, 33, p. 63]. One of the reasons for this is the important role master data plays in many companies in meeting strategic business requirements, as the following examples show:

- Companies from the process industry have to meet the provisions of REACH, which is an EU guideline regulating registration, evaluation, authorization, and restriction of chemical substances. Companies affected by REACH need to gather data and give evidence of the properties of their chemical substances across the entire supply chain [8].
- With fierce competition in a highly saturated market, insurance companies need to be able to view their customers from a 360° perspective, i.e. all customer master data, contract data, and performance data must be available in a consistent, up-to-date, and complete form across the company [24, p. 330]. This requirement is enforced by regulations and promulgated under the Insurance Mediation Directive as well as the Markets in Financial Instruments Directive (MiFID) [12].
- In order to be able to conduct a comprehensive spend analysis in multi-divisional companies, the central procurement department has to have access to consistent supplier master data and product

group codes. Also, ownerships structures of suppliers and their affiliates must be transparent so that purchasing volumes of all subsidiaries can be taken into account in an evaluation [10].

Master data represents the business objects which are agreed on and shared across the enterprise [6, p. 35] whereas the management of master data is referred to as an application-independent process which describes, owns and manages those "core business data entities" [33, pp. 65-66]. It ensures quality of master data through a set of guidelines for their management [6, p. 5]. With regard to this purpose, master data management (MDM) comprises several design decisions. Among them are the identification of roles and the assignment of responsibilities in the management and use of data – often referred to as Data Governance [cp. 37] – and the design of an enterprise master data architecture. Being part of the enterprise architecture, the latter describes the company's key business objects and the relationships between them on a conceptual level as well as the application architecture comprising the entirety of a company's applications that create, store and update instances of the business object types defined in the conceptual data model. Knowledge of the essential issues a company has to deal with when designing an enterprise master data architecture as well as the corresponding design alternatives is a prerequisite for actively managing and improving the quality of master data. Only with this knowledge companies are able to initiate well-directed activities for master data quality management that best fit their requirements.

A large body of knowledge exists which deals with various aspects of enterprise architecture management. Many widely accepted frameworks such as The Open Group Architecture Framework (TOGAF) also cover data architectures and their constituents. Due to their overall scope, however, they fall short of providing details regarding design decisions and design options with regard to enterprise master data architectures. On the other hand, one can find a number of reports from business analysts, consultants, and software vendors with recommendations for MDM in general [cp. 38, 39], master data architectures in particular [36] and deployment scenarios for individual solutions [11]. But they focus on specific aspects (e.g. data distribution) of and perspectives (e.g. certain industries or master data domains) on enterprise master data architectures.

Research Question and Approach

The strategic importance of master data requires design decisions regarding enterprise-wide master data architectures to be based on solid ground. Against this background, the research question in this paper is: What are design decisions companies have to make in the design of enterprise master data architectures and which design options exist?

The paper uses three cases studies to investigate the research question. Case study research can pursue two different goals: firstly, case studies can examine, describe and explain phenomena in a given (business) context in an explorative manner, secondly, case studies allow to test and develop new theories [7, p. 533, 31, pp. 11-12]. As our study aims at the former, our case studies can be defined as explorative [cp. 41] describing and investigating a complex area of research and trying to identify and explain interdependencies or cause effect relations [41, p. 15].

The study design is characterized by multi-case studies as the cases of DB Netz, SBB Cargo and Deutsche Telekom are examined with regard to the same topic [41, pp. 38ff.]. This leads to increased generalizability of findings compared to single case studies [2, p. 58]. Data collection in the cases follows the principles of case study research in the domain of business engineering as proposed by Senger and Österle [cp. 32]. The cases are described in the third section in a brief manner due to space limitations. A more detailed presentation of the cases is available in the referenced literature. Data analysis is carried out case-by-case in a first step, followed by a morphological analysis [cp. 26] in a second step. The paper concludes with a result summary and an outlook to future research questions.

RELATED WORK

Master Data Management

Data can be divided into master data, transaction data, and inventory data. Master data refers to the characteristics of core business objects within an organization [38, p. 2]. Typical master data classes are material and product master data, supplier and customer master data, and master data regarding employees and assets [5, pp. 176-177, 17, pp. 5-6]. Transaction data describe business transactions and represents input and output of business processes. It is constantly created in the course of business activities, references master data and indicates changes to inventory data [6, 35]. Examples comprise purchase orders, invoices and shipping notes. Inventory data refers to stock and account levels, e.g. to bank account balances or reserved stock of finished goods [21, p. 107]. Table 1 compares the different types of data with regard to:

- the dependence of data from a certain point in time(time reference),
- the frequency data is updated or changed during a certain period of time (change frequency),
- the growth in volume of data over time (volume volatility), and
- the degree to which data needs to reference other data types (existential independence).

The comparison is based on analysis of the definitions found in the referenced literature.

	Time reference	**Change frequency**	**Volume volatility**	**Existential independence**
Master Data	Low	Low	Low	High
Transaction Data	High	Low	High	Low
Inventory Data	High	High	Low	Low

Table 1. Types of Data (based on [6, pp. 35f., 21, p. 107])

MDM is an application-independent process for the description, ownership and management of core business data objects [5, p. 171, 33, pp. 65-66]. In this sense, the term MDM does not primarily refer to a certain class of information systems. Instead, it comprises design activities on a strategic, an organizational and on a information systems level [20, p. 15]. While data management has been investigated for a long time, MDM has not been well established in the past. Only recently, companies and research have been working on management concepts that consider specific characteristics of master data (see Table 1) [9, pp. 17f., 33, p. 64].

Enterprise Master Data Architecture

According to ANSI/IEEE Standard 1471-2000, an architecture is defined as the “fundamental organization of a system, embodied in its components, their relationships to each other and the environment, and the principles governing its design and evolution” [13]. The information architecture as information system sub-architecture is often referred to as consisting of two parts [22, p. 198, 23, p. 39]: a conceptual master data model describing a company’s key business objects as well as the relationships between them on a conceptual level (e. g. in a data object model) [5, p. 63] and an application architecture comprising the entirety of a company’s applications that create, store, and update instances of the entity types defined in the conceptual master data model. The enterprise master data architecture is an information architecture whose scope is restricted to a specific data type (master data). It comprises both static and dynamic aspects of MDM focusing on cross-application consistency of master data definitions and distribution. As illustrated in Figure 1 this includes the exchange of data elements between different applications represented as data flows [40, p. 2].

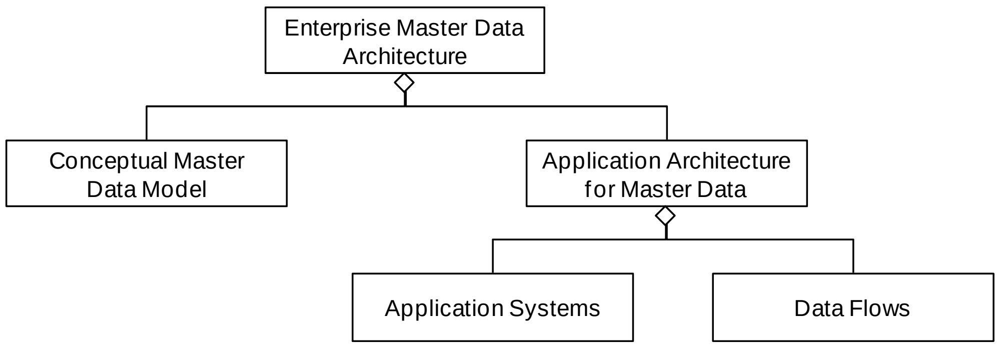

Figure 1. Components of Enterprise Master Data Architecture

Architecture Frameworks

For comparison of existing approaches for designing architectures we have taken into consideration approaches that fulfill the following requirements:

- They are widely disseminated among both researchers and practitioners (i.e. approaches for which a high number and availability of publications exist).
- They include a methodical component (e.g. procedure model, techniques).
- They address the design of information or data architectures, yet are not necessarily limited to this.

Included in the evaluation are the Zachman Framework [cp. 42], The Open Group Architecture Framework (TOGAF) [cp. 18], Enterprise Architecture Planning (EAP) [cp. 34, cp. 35], the Federal Enterprise Architecture Framework (FEAF) [cp. 4], and the Enterprise Architecture Cube (EAC) by BERNARD [cp. 3]. In addition, we incorporated the Guide to the Data Management Body of Knowledge in which the Data Management Association (DAMA) addresses challenges related to (master) data and information architectures in the context of data and data quality management [cp. 5]. Comparison of the frameworks is based on the evaluation of the referenced publications. Several approaches were left out here, because they either do not address the area of information or data architectures (e.g. the approach by LANKHORST [cp. 14]), are very similar to other approaches (e.g. the Treasury Enterprise Architecture Framework (TEAF) which is based on the Zachman Framework) or represent a modeling notation only (e.g. The Open Group's ArchiMate modeling language [cp. 15, 19] that complements TOGAF.

The different architecture frameworks were evaluated based on criteria that had been derived either from the conceptual background of our research (see section *Background*) or from the actual challenges companies are currently facing (see section *Case Studies*). The criteria include:

- Focusing on the enterprise master data architecture. An organization's architecture cannot be analyzed in its full breadth (i.e. including all of an organization's specific architectures) and with sufficient depth (i. e. to a high degree of detail for each specific architecture) at the same time [1, p. 39]. Consequently, it is advisable to restrict analysis and design efforts to a certain specific sub-architecture in order to be able to provide detailed recommendations.
- Addressing all components of an enterprise master data architecture. Each approach or framework should take into consideration all components of an enterprise master data architecture as depicted in Figure 1, i.e. they should go beyond mere data storing issues (application systems) and encompass data distribution between applications (data flows) as well as the business perspective on data objects (unambiguous understanding of business objects).
- Referring specifically to master data and its characteristics (see Table 1).

- Describing design decisions in detail. In order to allow for application in practice, it is not sufficient to just list models or architecture components and their relationships. Rather the approach needs to show which decisions a company needs to take to ensure a sustainable development of the enterprise master data architecture.
- Providing concrete guidelines and design options for each design decision.

The feature based comparison in Table 2 is based on the evaluation of publications (e.g. conference papers, literature, magazine articles) on single approaches.

	Zachman	TOGAF	EAP	FEAF	EAC	DAMA
Enterprise master data architecture focus	○	◑	◑	○	◑	◑
Coverage of all enterprise master data architecture components	◑	◑	●	◑	◑	●
Reference to master data	○	○	○	○	○	◑
Design decisions	◑	◑	◑	◑	◑	◑
Design options	○	◑	◑	○	◑	◑
Legend	● does fulfill criteria completely		◑ does fulfill criteria partly		○ does not fulfill criteria at all	

Table 2. Evaluation of Architecture Frameworks

One result that can be derived from Table 2 is that all approaches address the different sub-architectures that compose an enterprise master data architecture as defined in the background section (Criterion 2). However, only the DAMA approach has a clear focus on the design of the enterprise master data architecture (Criterion 1). This basically results from the fact that the DAMA framework concentrates on the (quality) management of data exclusively, while the other frameworks and models analyzed all present holistic approaches for designing and managing corporate architectures as a whole. Consequently, the DAMA framework is the only approach that specifically accounts for characteristics of different data types, including master data (Criterion 3). If we reconsider the statement that an organization's architecture cannot be analyzed in full breadth (i.e. including all of an organization's specific architectures) and with sufficient depth (i.e. to a high degree of detail for each specific architecture) at the same time, it becomes quite obvious that the demand for concrete guidelines and design options can only be met to a small extent (Criterion 5). While all approaches present – in one way or another – a process for designing the architecture, it is only EAP (with its implementation methodology) and TOGAF (with its Architecture Development Method) that come with a detailed procedure model allowing to design the different views of the corporate architecture systematically. Among those, EAP is the only approach offering a high degree of detail in order to provide users with practicable techniques for analysis and design including concrete design options.

CASE STUDIES

Case Study Selection and Approach

As already mentioned in the introduction, the paper relies on three case studies that are used to identify design decisions and design options for the enterprise master data architecture as the occur in business contexts. The similarity of all case studies consists in the challenges they are facing with managing their master data and that are related to architectural aspects. They slightly defer with regard to the dominant business requirement as well as to the scope of the MDM initiatives. The three case studies are characterized in Table 3.

	DB Netz	SBB Cargo	Deutsche Telekom
SIC code	40 (Railroad Transportation)	47 (Transportation Service)	48 (Communications)
Markets served	Central Europe	Central Europe	International
Business requirements	Compliance reporting, process harmonization	Cash-flow reporting	Merger of two business units
Organizational scope	Enterprise-wide	Enterprise-wide	Business unit
Master data classes	Infrastructure master data	All	All
Complete case study	---	[29]	[30]

Table 3. Categorization of Three Case Studies

Case A: DB Netz

The first case refers to Deutsche Bahn AG, a state-owned German railway operator, more precisely its subsidiary DB Netz. DB Netz is responsible for maintaining and operating Germany's railway network infrastructure, i.e. tracks, stations, tunnels and bridges. DB Netz is faced with two major challenges. Firstly, the company's yearly subsidization by the German Federal Railway Authority is directly related to the quality of infrastructure data, i.e. on the ability of the company to provide accurate, consistent and timely information on the number and maintenance status of the national infrastructure assets. For the inventory of all infrastructure assets that is reported once a year data from several applications needs to be consolidated in a single infrastructure register. This consolidation process, however, considerably lacks transparency on where certain data is stored and requires substantial manual effort leading to a loss of quality of the resulting register.

The second challenge is of operational nature. Most of the business processes go beyond the boundaries of single business units and require involvement of virtually all business functions such as construction planning, timetable planning, railway network operation, maintenance, or energy provision. One example for such a cross-functional business process is the rebuilding of a station track, which consists of seven sub-processes: basic evaluation, preplanning, design planning, implementation planning, implementation, commissioning, and documentation. Due to its cross-functional character the business process is characterized by many media breaks, missing and inconsistent data, a low level of process constancy, and long process lead times. In order to improve the situation, DB Netz has decided to develop an enterprise-wide infrastructure master data architecture. In the process of analyzing the initial situation and designing a to-be architecture, the following problems have occurred:

- What is a common definition of the business object "station track"? (master data object definition)
- Which of the business object's attributes must be used in a standardized way across different processes, and which need not? (master data validity, master data object definition)
- Which of the business object's attributes are currently stored, altered, and distributed in which application systems? (metadata management)
- How do data flows between application systems look like? (master data application topology and distribution)
- Who is responsible for which data? (master data ownership)
- What data is created, used, changed in which activity of the business process? (master data lifecycle, master data operations)
- Should data describing station tracks be stored in a central system or in several, distributed systems? (master data application topology)

Case B: SBB Cargo

SBB Cargo AG is market-leading provider for rail freight services in Switzerland. The market deregulation in 2001 caused a rapid business development, relocating the focus on cross-border transport, most notably the Alps crossing freight transport connecting the North Sea with Italy. The resulting internal re-organization, adjustment of business processes and introduction of new services virtually neglected the management of master data leading to master data redundancy, differently used object definitions, a lack of ownership of master data objects and the disruption of business processes. One of the most critical aspects was the existence of peer-to-peer data flows between an ever increasing number of (proprietary) applications that created, stored and updated master data objects redundantly. Consequently, distribution of master data on different applications was not transparent anymore. Furthermore, operational and financial data were held independently leading to a situation in which planning and reporting of the services provided were based on completely separate master data basis.

With the long-term goal of establishing an integrated, service-based master data architecture as part of the enterprise-wide MDM initiative, SBB Cargo defined several fields of action:

- Determine common uniform definitions and structures for the company’s master data objects (master data object definition, conceptual master data model) as well as unique identifiers for each master data class for unambiguous identification (master data validity).
- Establish a central organizational unit responsible for carrying out changes on master data objects (master data operations, master data ownership).
- Determine the “leading system” for each master data class and optimize architecture of applications administrating master data (master data application topology).
- Create a Master Data Map depicting assignment of master data objects to applications and the data flows between them (master data application topology and distribution).
- Design and implement tool-supported MDM processes (master data lifecycle, master data operations).

Case C: Deutsche Telekom

The third case refers to one of the leading European companies in telecommunications. Due to a merger of two formerly separate business units that managed their master data independently, the company faced the challenge of consolidating the partly redundant data on customers and infrastructure in order to allow for integrated product and service bundle offerings. With the absence of a systematic approach for preventative MDM, Deutsche Telekom was confronted with a lack of transparency on:

- Origin and distribution of master data objects on its current application architecture (master data application topology and distribution)
- Semantics of master data objects leading to ambiguous understandings and inconsistent usage (master data definitions, metadata management)
- Business requirements on enterprise-wide data quality of certain master data objects (master data validity)

For resolving the identified problems the company pursues an approach that closely interlinks data quality measurement activities (rather short-term) with techniques for the design of the enterprise master data architecture (rather long-term). The latter is based on a uniform data architecture that results from an integrated data modeling approach on all four layers of the enterprise architecture (see Table 4). Within this approach, the logical architecture layer plays a pivotal role for harmonizing different understandings of the company’s master data objects between business and information technology (IT) departments (in the Business Object Model). Moreover it interlinks the master data architecture with the application architecture by assigning master data objects to applications (in the Data Map) and functional domains (in the Group Domain Model). Deutsche Telekom uses the Data Map to define leading systems for each master data class with the goal of centralizing the formerly distributed storage of master data objects.

Architecture Layer	Architecture Model	Data Model Type	Exemplary Data Model
Business Process Architecture	Business process model	Business object glossary (master data definition)	Data Model resulting from "Processes and Data" project
Logical Architecture	Group Domain Model and Data Map	Semantic data model (conceptual master data model)	Business Object Model
Application / System Architecture	Application Landscape (master data application topology and distribution)	Logical data model of a single application	Data model of the CRM application
Infrastructure Architecture	(not applicable)	Physical data model	Database model of the application

Table 4. Interrelation Between Enterprise and Data Architecture at Deutsche Telekom

DESIGN DECISIONS AND OPTIONS

The case studies reveal the complexity of designing and developing an enterprise master data architecture. The individuals in charge had to focus on the master data perspective of architecture management and at the same time take into account the interdependencies of the latter to other architectural perspectives such as the business architecture. Therefore, the data collected in the three cases was analyzed applying the concept of morphological analysis.

Morphological analysis allows to fully capture and structure complex problems and identify alternatives for a solution [26, pp. 3-4]. RITCHEY describes it as a "general method for non-quantified modeling". The method allows to comprehensively illustrate the solution space of a multi-dimensional problem and to reduce complexity by limiting the number of design options for a solution [27, pp. 795-796]. The central element of a morphological analysis is the morphological field, which consists of several dimensions or variables. For each dimension or variable possible manifestations are entered in the respective line, which generates a matrix in the end. Alternatives for solution are produced by combining selected manifestations of different dimensions or variables (so-called configuration of the morphological field) [27, p. 794]. The morphological field is a helpful instrument to have complex issues discussed and structured by experts (e. g. during case study data collection) but also for outlining future development scenarios.

Table 5 shows a morphological field for the design of an enterprise master data architecture with each dimension standing for a necessary design decision and the according manifestations representing design options. Both the dimensions – i.e. design decisions – and the related design options emerged from the cases analysis. The open questions which were discussed in the three cases as well as potential answers to them were compared and conceptualized. The results were then reflected – as a form of triangulation – against selected other sources of knowledge, e.g. books and reports in the practitioners' domain. An example which illustrates the approach is the identification of "master data ownership" as a variable. In case A the question regarding "responsibility" for data was raised, similar to case B where the need for the assignment of responsibilities for master data was identified with regard to the lifecycle of master data. In case C the point was discussed in the context of organizational validity of the data. In the cross-case analysis these findings were conceptualized as "master data ownership". The concept was confirmed after reflection against results of current research on the topic [37, p. 6].

The morphological field has to be instantiated per master data class. In order to clarify the interrelation between the morphological field and the aforementioned use cases, we additionally included a column referencing the cases in which each design decision was considered relevant. The companies analyzed in the case study showed an “arbitrary” understanding of decisions which they had to make. However, they missed a consistent and complete conceptualization of the latter. Moreover, the morphological analysis shows that design decisions do not only relate to the enterprise master data architecture components, but do also affect related business architecture elements (depicted in italics).

Design Decision		**Case Coverage**	**Design Options**			
Master Data Ownership		*A, B and C*	*Defined enterprise-wide*		*Defined locally*	
Master Data Validity		*A, B and C*	*Enterprise-wide*	*Specific business unit*	*Single business process*	
Master Data Lifecycle	*Creation*	*A and B*	*Centrally standardized*	*Hybrid*	*Local design*	
	Update		*Centrally standardized*	*Hybrid*	*Local design*	
	Deactivation		*Centrally standardized*	*Hybrid*	*Local design*	
Master Data Operations		*A and B*	*Centralized execution (e.g. by a central organizational unit)*		*Local execution (e. g. by owner)*	
Conceptual Master Data Model		B and C	Enterprise-wide, unambiguous	Per business unit	Per project	Not defined
Master Data Object Definition		A, B, and C	Enterprise-wide, unambiguous	Per business unit	Per project	Not defined
Metadata Management		A and C	Owned and defined enterprise-wide		Not actively managed	
Master Data Application Topology		A, B and C	Central system	Leading system	Consolidation Hub	Repository
Master Data Distribution		A, B, and C	Push		Pull	
Master Data Processing		B and C	Batch		Real-time	

Table 5. Morphological Field with Design Decisions and Options

In all cases, a major concern was the definition of ownership for business objects. In the investigated cases, the unanimous conviction was that a business function has took over ownership because only business has the expertise to define the organizational validity of business objects and define and execute the business process which manipulates and uses master data objects. Typically, data governance identifies appropriate roles for ownership and also assigns responsibilities accordingly [37, p. 6].

Master data lifecycle management determines the activities within business processes which create, update/change, and deactivate master data elements as well as the corresponding responsibilities for each of the tasks. Here, the case studies reveal a common tendency of increasingly centralizing lifecycle management tasks, as in the case of customer and infrastructure data at SBB Cargo. However, currently processes for different master data lifecycle activities are either designed (and implemented) locally (as in the case of SBB Cargo) or centrally for a small number of master data objects only (as in the case of DB Netz). While the investigated companies put considerable effort on defining responsibilities for creating and updating master data objects, they most often omit the assignment of roles for the deactivation or deletion of outdated master data objects leading to increased redundancy and reduced master data quality. We therefore explicitly state this design decision within the morphological field.

Looking at the dimensions related to master data modeling (rows 5-7), it becomes evident that all aspects (master data object definitions, a conceptual master data model and metadata management) should preferably be designed on an enterprise-wide level in order to guarantee unambiguous understanding and uniform utilization of master data objects. All investigated companies are currently conducting initiatives to consolidate and harmonize project and business unit specific master data definitions as a prerequisite for consistent MDM. For the time being, master data objects have only been defined for a certain business unit (Deutsche Telekom), a certain project (SBB Cargo) or not at all (DB Netz). Moreover, as the case study of Deutsche Telekom shows, an integrated approach that interconnects different layers of the data architecture (e. g. the glossary containing common master data object definitions with the conceptual data model) should foster transfer of the master data harmonization reached on a conceptual level to the application layer and at the same time contribute to the alignment between requirements of the business and implementation by IT.

With regard to the distribution of master data to different applications as part of the application architecture, the cases indicated a distinction between a central and a distributed storage, with the former basically being better suited to ensure consistency of master data across different applications [28, p. 158]. In the case of central solutions master data can be stored either in a central system, i. e. an application set up only to keep and store master data or global attributes, or in a leading system. A leading system denotes an existing application (e. g. a CRM system) that is selected to "lead" in the management of such data [16, p. 3]. In both variants data are usually replicated into local systems, producing controlled redundancy.

If a central solution is not possible, either because of organizational or because of technological reasons, distributed keeping and storing of master data often represents a compromise. Distributed solutions can be distinguished by the fact whether there is a central instance for synchronizing locally held master data. The central instance is a consolidation hub that first extracts the globally used attributes from the local applications where they were initially created before it harmonizes these attributes [6, pp. 235ff., 25, p. 3]. In doing so, controlled data redundancy is deliberately created without any negative effects on data consistency. In a registry-style architecture in turn, only a master data index consisting of a minimal set of identifying attributes is held in a directory [17, pp. 166-168]. The registry references master data kept and stored in distributed applications. Usually, only the inquiry is directed to the registry, while the actual data exchange happens directly between the inquiring system and the application storing the data. Ideally, an architecture design pattern like that should do without data redundancy, as the existence of master data object attributes in other applications is checked in the registry before new attributes are to be created.

Finally, regarding the application architecture decisions have to be taken on how data flows between the application systems are triggered and processed. This firstly concerns the question whether the implementation is based on a push or a pull principle, specifying whether an exchange of data is triggered by the sending (push) or by the receiving (pull) system, and secondly, whether changes on master data objects are propagated directly (real-time) to all relevant systems. This last design decision is strongly dependent on the application architecture decision as for instance real-time master data processing cannot be achieved in hub- or registry-style architectures.

CONCLUSION AND OUTLOOK

Due to the ever-increasing importance of master data for companies to respond to strategic business requirements, the management of master data has become more and more vital in the past. For being able to live up to this strategic importance of master data the intention of the paper was to identify and describe major decisions companies have to make in the design of enterprise master data architectures and show corresponding design options.

The paper contributes to the advancement of the scientific body of knowledge because it investigates an area in which only little previous research is available. Apart from that, it also contributes to the practitioners' community because it allows companies, such as those presented in the case studies of this paper, to systematically develop and advance their enterprise master data architecture as a fundamental basis for MDM.

The paper consequently presents a number of relevant design decisions and design options organized in a morphological field. They were derived from three case studies of companies being faced with similar challenges regarding the management of their master data. The paper describes each design decision in more detail and also discusses possible interdependencies between the design decisions.

Validity and generalizability of our research is restricted due to the relatively low number of cases investigated. This shortcoming should be compensated by extending the sample in the future. Findings described in this paper, however, will be helpful for a wider empirical study of decisions and options for enterprise master data architecture design. Moreover, future research should focus on identifying predefined architecture patterns (i. e. configurations of the morphological field) depending on enterprise contingencies. These architecture patterns could then be evaluated based on cost and quality related criteria. The patterns could also form the foundation for practitioner guidelines when to chose which design option.

Another focus of future research addresses the question to which extent design decisions and design options depend on the specific usage scenario of enterprise master data, e.g. the reference, operational, and analytical scenario as proposed by Loshin [17, pp. 160-165]. And apart from that, an interesting direction of future research would be a longitudinal approach to investigate on the effects of enterprise master data architecture design on the quality of master data.

ACKNOWLEDGEMENT

The research presented in this paper was carried out within the Competence Center Corporate Data Quality (CC CDQ). As a consortium research project at the Chair of Prof. Dr. Hubert Österle at the Institute of Information Management at the University of St. Gallen, the CC CDQ is part of the Research Program Business Engineering (BE HSG) at the University of St. Gallen.

REFERENCES

[1] Aier, S., Kurpjuweit, S., Saat, J., Winter, R. "Enterprise Architecture Design as an Engineering Discipline." *AIS Transactions on Enterprise Systems*, 1 (1), 2009, pp. 36-43

[2] Benbasat, I. "An analysis of research methodologies." In *The information systems research challenge*, F. W. McFarlan, Ed. Harvard Business School, Boston, 1985, pp. 47-85

[3] Bernard, S. A. *An Introduction to Enterprise Architecture*, Authorhouse, Bloomington, USA, 2005

[4] CIO Council *Federal Enterprise Architecture Framework Version 1.1*, Chief Information Officer Council, Washington, DC, USA, 1999

[5] DAMA *The DAMA Guide to the Data Management of Knowledge*, Technics Publications, Bradley Beach, New Jersey, 2009

[6] Dreibelbis, A., Hechler, E., Milman, I., Oberhofer, M., van Run, P., Wolfson, D. *Enterprise Master Data Management: An SOA Approach to Managing Core Information*, Pearson Education, Boston, 2008

[7] Eisenhardt, K. M. "Building theories form case study research." *Academy Of Management Review*, 14 (4), 1989, pp. 532-550

[8] European Commission *REACH*. European Parliament and the Council, Brussels, Belgium, Regulation (EC) No 1907/2006, 2006

[9] Friedman, T., Feinberg, D., Beyer, M. A., Gassman, B., Bitterer, A., Newman, D., Radcliffe, J., Bell, T., Gilbert, M. R., Blechar, M. J., White, A. H*ype Cycle for Data Management 2007*, Gartner, Stamford, USA 2007

[10] Gordon, S. "Seven Steps To Measure Supplier Performance." *Quality Progress*, 38 (8), 2005, pp. 20-25

[11] Hechler, E., Oberhofer, M., van Run, P. *Implementing a transaction hub MDM pattern using IBM InfoSphere Master Data Management Server*. IBM Corporation 2008

[12] Heinrich, B., Kaiser, M., Klier, M. "Does the EU Insurance Mediation Directive Help to Improve Data Quality? - A Metric-Based Analysis." In *Proceedings of the 16th European Conference on Information Systems (ECIS2008)*, Galway, Ireland, 2008

[13] ISO/IEC *Systems and software engineering: Recommended practice for architectural description of software-intensive systems*. ISO/IEC, Geneva, Switzerland, Report No. 42010:2007, 2007

[14] Lankhorst, M. *Enterprise Architecture at Work: Modelling, Communication and Analysis*, Springer, Berlin et al., 2005

[15] Lankhorst, M., Proper, E., Jonkers, H. "The Architecture of the ArchiMate Language." In *Enterprise, Business-Process and Information Systems Modeling*. Springer, Berlin, Heidelberg, 2009

[16] Loser, C., Legner, C., Gizanis, D. "Master Data Management for Collaborative Service Processes." In *International Conference on Service Systems and Service Management*. Research Center for Contemporary Management, Tsinghua University, 2004

[17] Loshin, D. *Master Data Management*, Elsevier Science & Technology Books, Burlington, MA, 2008

[18] Open Group *The Open Group Architecture Framework TOGAF - 2007 Edition (Incorporating 8.1.1)*, Van Haren, Zaltbommel, 2007

[19] Open Group *ArchiMate 1.0 Specification*, Van Haren Publishing, Zaltbommel, Netherlands, 2009

[20] Österle, H. "Business Engineering: Transition to the Networked Enterprise." *Electronic Markets*, 6 (2), 1996, pp. 14-16

[21] Otto, B., Reichert, A. "Organizing Master Data Management: Findings from an Expert Survey." In *Proceedings of the 2010 ACM Symposium on Applied Computing*, 2010. pp. 106-110

[22] Periasamy, K. P., Feeny, D. F. "Information architecture practice: research-based recommendations for the practitioner." *Journal of Information Technology*, 12 (3), 1997, pp. 197-205

[23] Pienimäki, T. *A Business Application Architecture Framework in Manufacturing Industry*. Dissertation. Tampere University of Technology, Tampere 2005

[24] Pula, E. N., Stone, M., Foss, B. "Customer data management in practice: An insurance case study." *Journal of Database Marketing*, 10 (4), 2003, pp. 327-341

[25] Radcliffe, J., White, A., Newman, D. *How to Choose the Right Architectural Style for Master Data Management*. Gartner Group, Stamford 2006

[26] Ritchey, T. "General Morphological Analysis: A general method for non-quantified modeling." Swedish Morphological Society, 2006

[27] Ritchey, T. "Problem structuring using computer-aided morphological analysis." *Journal of the Operational Research Society*, 2006, pp. 792-801

[28] Schemm, J. *Stammdatenmanagement zwischen Handel und Konsumgüterindustrie - Referenzarchitektur für die überbetriebliche Datensynchronisation*. Dissertation. Institut für Wirtschaftsinformatik, Universität St. Gallen, St. Gallen 2008

[29] Schmidt, A. *Fallstudie SBB Cargo – Einführung eines unternehmensweiten Stammdatenmanagements*. Institute of Information Management, University of St. Gallen, St. Gallen, Switzerland 2009

[30] Schmidt, A., Hüner, K., Grewe, A. *Fallstudie Deutsche Telekom AG – Einheitliche Datenarchitektur als Grundlage für unternehmensweites Datenqualitätsmanagement*. Institute of Information Management, University of St. Gallen, St. Gallen, Switzerland 2010

[31] Scholz, R. W., Tietje, O. *Embedded case study methods - Integrating quantitative and qualitative knowledge*, Sage, Thousand Oaks, 2002

[32] Senger, E., Österle, H. *PROMET BECS - A Project Method for Business Engineering Case Studies*. St. Gallen, Working Paper BE HSG / BECS / 1, 2002

[33] Smith, H. A., McKeen, J. D. "Developments in Practice XXX: Master Data Management: Salvation or Snake Oil?" *Communications of the AIS*, 23 (4), 2008, pp. 63-72

[34] Spewak, S. H., Hill, S. C. *Enterprise Architecture Planning - Developing a Blueprint for Data, Applications and Technology*, John Wiley & Sons, New York, 1993

[35] Spewak, S. H., Tiemann, M. "Updating the Enterprise Architecture Planning Model." *Journal of Enterprise Architecture*, 2 (2), 2006, pp. 11-19

[36] Swanton, B., Hagerty, J., Cecere, L. *MDM Strategies for Enterprise Applications*. AMR Research, Boston 2007

[37] Weber, K., Otto, B., Österle, H. "One Size Does Not Fit All – A Contingency Approach to Data Governance." *ACM Journal of Data and Information Quality*, 1 (1), 2009, Article No. 4

[38] White, A., Newman, D., Logan, D., Radcliffe, J. *Mastering Master Data Management*. Gartner, Stamford 2006

[39] White, A., Radcliffe, J., Eschinger, C. *Predicts 2009: Master Data Management Is Applicable in Down Economies and in Times of Growth*. Gartner, Stamford, G00164023, 2008

[40] Winter, R., Fischer, R. "Essential Layers, Artifacts, and Dependencies of Enterprise Architecture." *Journal of Enterprise Architecture*, 3 (2), 2007, pp. 7-18

[41] Yin, R. K. *Case study research: design and methods*. 3 ed., Sage Publications, Thousand Oaks, 2002

[42] Zachman, J. A. "A Framework for Information Systems Architecture." *IBM Systems Journal*, 26 (3), 1987, pp. 276-292