Towards agile BI: applying in-memory technology to data warehouse architectures

Tobias Knabke, Sebastian Olbrich

University of Duisburg-Essen, Germany
Mercator School of Management (MSM)
Chair of Information Systems and Operations Research
Lotharstraße 63
47057 Duisburg
tobias.knabke@stud.uni-due.de
sebastian.olbrich@uni-due.de

Abstract: Confronted with increased market dynamics and hence frequently changing system environments, today's decision support systems face the demand to respect such developments. Developing and maintaining so called agile business intelligence (BI) systems is a major challenge for information technology and organizations, since the underlying assumption of BI is to support mostly long term decisions in a non-volatile and integrated way. Hence, current approaches towards agility often focus on shortened implementation times using agile methods like Extreme Programming (XP) or Scrum. Given the existing BI architectures and environments, these methods are not fully applicable. Thus, this paper focuses on the architecture of agile BI. To achieve this goal, we derive criteria for agile BI. Then, a semi-virtual data warehouse architecture with an in-memory database (IMDB) as a technology enabler for agile BI is proposed conceptually.

1 Introduction and motivation

With the use of computer based information systems (IS) the amount of data to be incorporated in management decisions increased heavily in the last decades [Cu10]. Frequently changing internal and external requirements as well as globalization forces managers to react flexible on environmental changes [Wi10, CG04, KMU06]. Therefore Business Intelligence (BI) Systems are widely used within strategic and operational business processes. However, it turned out that current BI systems behave rigid and inflexible in terms of readjustment to fulfil changed or new requirements [Ba10]. This characteristic of BI contradicts with the earlier statement of frequently changing requirements and big data [Cu10, Be11]. In order to make BI applicable for volatile environments and requirements, the usage of agile process or management models like Scrum [Sc95] or Extreme Programming (XP) [Be01] has been in the centre of recent discussions [Ba10, GH11]. However, agile methods focus on the process of creating or changing a BI system. Hence, they potentially improve that process of creation and change by means of time and flexibility. While this is with no doubt an important issue, the resulting (layered and scalable) data warehouse architecture (DWH) remains the same - rigid and inflexible in terms of readjustment to fulfil changed or new requirements. Many argue whether agile development methods can be applied to BI projects at all [e.g. Mo09].

The assumption of this article is that the whole BI architecture has to become more agile to create agile BI systems. We therefore neglect the agile development and project management approaches and focus on the architecture of the BI system. Earlier approaches to achieve this goal – like virtual data warehouses (VDWHs) - could not establish itself in the last years [SBM99] for multiple reasons like performance when drilling to granular level or assessing historical data [VBP10]. Recently, Schmidt-Volkmar [Sc08], Schaffner et al. [Sc09] and Plattner [Pl09] have proposed promising in-memory (IM) solutions to enable operational reporting in a timely manner. Using in-memory databases (IMDBs), also known as main memory databases (MMDB), as technology foundation reduces response time of queries involving mass data significantly [PZ11]. However, the promises made by applying IMDB are based on studies that are executed in one-system (mostly operational) environments [Sc08]. Nevertheless, the application landscapes of most companies are heterogeneous. The contribution of this paper is therefore to propose an agile BI architecture in heterogeneous system environments by using IMDB as technology enabler in conjunction with VDWH technology and a logical data model. By creating that reference architecture we try to answer the following research question:

- Do BI architectures become more agile by applying IM-technology in conjunction with VDWH?

And two further sub-questions:

- What are the changes on current state-of-the-art BI architectures?
- What are the effects on current BI projects and environments?

To answer these questions we use logical deductive argumentation in order to poof/disproof our hypothesis. Therefore, the paper is structured as follows: In the next section we conduct background research on underlying technology. First, we look at current BI architectures, then derive the need for agility and last introduce the IMDB technology. The third section introduces our reference BI architecture applying the requirements from the second section. We close this paper with a discussion of our results, the limitations of this research and an outlook to our research agenda.

2 Background and definitions

BI is a broad category of IS that support decision makers through business analyses on the basis of internal and external data [CCN05, WW07, AC08]. BI can be defined as a set of technologies, applications, and processes for gathering, storing, accessing, and analyzing data that helps users to make better decisions [CG04]. BI supports problem and opportunity identification, decision-making, and alignment of operations with the corporate strategy [MH07] and, thus, contributes to the organization's competitiveness and sustainable development. Advanced BI systems include unstructured data [AC08], integrate external data sources (e.g. via remote servers) [CCN05], trigger (real-time) actions [Sh02], and enable data mining techniques. They support the intelligent exploration, integration, aggregation, and multidimensional analysis of data originating from a diverse set of information resources [OZ07].

The potential of BI systems to contribute to corporate success is considered enormous and, therefore, many organizations have launched BI initiatives with the intention to implement or to improve these systems [WW10]. Recently, a worldwide survey of 1500 CIOs even identified BI as the number one technology priority [Ga09a]. There is evidence however, that a significant number of organizations have failed to realize the expected benefits of BI [JC99, Sh03, HH05, CCD06]. For instance, the Cutter Consortium Report [Cu03] revealed that only 15% believed their BI initiative was a success and 41% of respondents had experienced at least one BI project failure. After all, BI implementation projects are expensive, time-consuming and risky undertakings [WW01, Ga09b].

2.1 State of the art data warehouse architecture

To fulfil the above mentioned goals of BI, the underlying architecture currently often consists of a DWH with separated layers [KMU06, Ha10]. Figure 1 depicts a generalized architecture to illustrate the concepts of DWHs exemplarily. Data is loaded from the sources (operational systems like ERP, flat files, external information such as exchange rates, etc.) into the business warehouse to the acquisition layer first. The data contained in this stage is not transformed, i.e. raw. Data cleansing, harmonisation and consolidation is the target of the DWH layer. It builds the single source of truth and is application-independent. The DWH layer forms the basis for the enterprise's business specific BI applications [Ha02, Ha10]. For large companies the data volume contained in a DWH can be up to several terabytes (TB) or even more, depending on the required data granularity or industry for the BI applications [KMU06]. During processing, these records are even stored several times (partly or completely) in the layered architecture.

As the functional requirements of the business can be diverging, i.e. financial departments usually have different requirements than their colleagues in production, business logic transforms and enriches the "general" DWH data into functional and application specific data. These steps take place in the transformation layer. Especially strategic business decisions are not made depending on detailed data granularity, e.g. sales order line items as contained in the lower layers. Hence, to meet performance and response time requirements during analysis operations, data is aggregated during loads to the reporting layer. In addition, many BI tools use a de-normalized approach (e.g. star schema) [Ki96] which allows for efficient read operations on big data volumes. Within the reporting layer data is often allocated in data marts to serve a specific application domain [CG98]. The corporate memory stores all data loaded to the BI system without transformation or cleansing. Thus, it can be used for reconstructing the data model after change implementations or if issues require a reload.

BI systems are more and more used to support operational tasks, e.g. in operational BI (OBI) systems, and thus are not limited to decision support. Hence, the concept of operational data stores (ODS) has been introduced [Wi10]. An ODS allows for analysis of operational data on a high granularity and it must be possible to load the data from different layers to the ODS [In98].

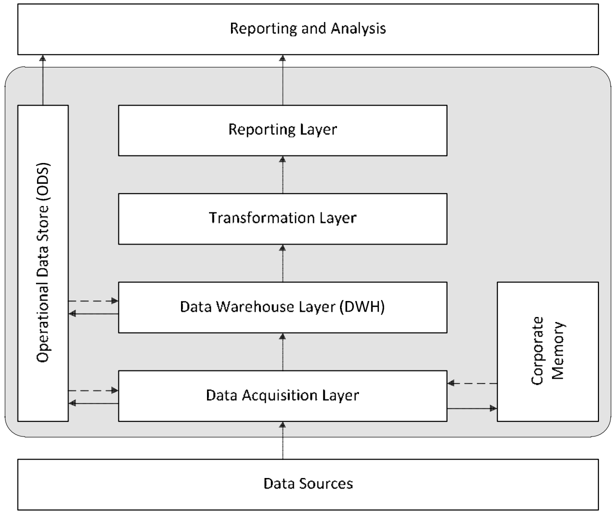

Figure 1: Multi-layer BI architecture (adapted from [Ha10])

The introduction and implementation of new or changed requirements with current BI architectures as described above is a time consuming process. This holds particularly true if e.g. new dimensions or key figures are not even contained in the lower levels of the BI architecture as this results in the restructuring or adaption of the data model within all affected layers. If such requirements impact historical data, data needs to be reloaded to the corresponding layers that store data physically to make changes available. To ensure a companywide consistency and keep systems maintainable and efficient, standard governance processes such as alignment of requirements need to be followed. This organizational and process compliance often extends the time frame to fulfil the business requirements – at least the perceived time frame by business users. In any case, a conflict in the goals of efficiency and agility can be observed [Ba10].

2.2 Agility in context of BI systems

Current businesses are facing a world of "agility" that has been shifted from traditional management [Hu99]. This often yields in new or changed requirements for their supporting tool sets and for BI in particular because BI applications have a close connection to business departments. In research literature, the term agility is ambivalent and almost no "*conceptual development of agility*" [Co09] p.330 took place in information system development (ISD). Conboy took a "first-principles" approach by examining the concepts of flexibility and leanness which build the basis of agility. After evolutionary defining flexibility and leanness based on a structured literature review of domains like management and other related business areas, he deducts his definition of agility in an ISD context as "*the continual readiness of an ISD method to rapidly or inherently create change, proactively or reactively embrace change, and learn from change while contributing to perceived customer value (economy, quality, and simplicity), through its collective components and relationships with its environment*" [Co09], p.340.

Following Conboy's deducted "*Taxonomy of ISD Agility*" [Co09] p. 341 and transferring it to BI architectures, such architectures must at least contribute to "creation of", "proaction in advance of", "reaction to" or "learning from" change to be agile. On top of that the taxonomy talks of "perceived economy, quality and simplicity" that the ISD agility should achieve as evaluation criterion. Although Conboy's definition originated in an ISD context, it is supported by other definitions of IS agility, e.g. the one by Pankaj et al. who focus on real-time aspect defining agile IS as *"one that can sense a change in real-time, diagnose the change in real-time, select a response in real-time, and execute the response in real-time"* [Pa09], p. 30. Since BI systems are often faced with the implementation of new or changed requirements in today's dynamic environments, these claims are relevant to examine if a BI architecture can be called agile. As illustrated in section 2.1, current BI architectures are optimized for storing and analyzing mass data rather than adapting to changing environments. Of course, applying agile process methods may also contribute to faster adaption of a BI. However, in this paper we focus on the technological aspects. A VDWH can be considered as a first technological approach to enable agile BI architecture.

DWHs – whether they are organized in a central, local or distributed way - store data physically to optimize performance or to clean, transform and enrich data to fulfil and enable (management) decisions. In contrast, VDWHs do not store any data physically but access data directly on the source systems [SBM99]. Nevertheless, VDWHs could not been widely established due to several reasons [In00, In04, KMU06]: For example, relevant information can be contained in several heterogeneous source systems which often do not store historical data. But historical data is needed for several strategic decisions, e.g. planning. Queries executed for analytical purposes are often based on mass data. Thus, they are resource intensive and have an enormous impact on operational systems. Moreover, in order to get reliable and consistent information operational data needs to be cleaned, consolidated, transformed and enriched. These tasks can be very complex and require many resource capacities. Given the technological developments and price reduction of main memory in recent years, IMDBs might overcome the shortcomings and might possibly serve as a technology enabler for VDWH.

2.3 In-memory database systems

Conventional database systems like relational database management system (RDBMS) usually use physical hard drives to store data. If data is accessed by an application it is loaded to the main memory for processing. Although data can be cached in the main memory in an RDBMS, the primary storage location remains a magnetic hard disk. Instead, an in-memory database system (IMDBS) keeps its data permanently in main memory of the underlying hardware. Main memory is directly accessible by the CPU(s) and the access is orders of magnitudes faster [GS92]. However, main memory is volatile. Therefore, data recovery strategies are critical regarding IM applications. Such strategies are available [PZ11] but not topic of this paper. Due to recent price reductions for main memory and the usage of dedicated compression techniques it is now possible to even hold the entire data of large-size companies in-memory [PZ11].

Depending on the organization's background and the target situation of the resulting IS, different data association approaches fit best for the appliance of IM-technology [De84, St05, Sc08, Pl09, Sc09, PZ11]. Online transactional processing (OLTP) systems like enterprise resource planning (ERP) systems prefer write-optimized structures whereas online analytical processing (OLAP) applications like BI need read-optimized data organization to achieve better query performance. Thus, row-oriented data organization is common in operational systems, i.e. OLTP. In contrast, column-oriented storage better suits OLAP systems. If one system yields to combine OLTP and OLAP functionalities, hybrid approaches that combine row- and column-oriented storage have been introduced recently [Pl09, Sc09, PZ11]. The difference of row- and column-oriented storage is shown in the example depicted in Figure 2. The database contains three tuples (rows *r*) with three attributes each (columns *c*). In row-oriented databases each row is stored in adjacent blocks whereas column-orientation keeps each column together.

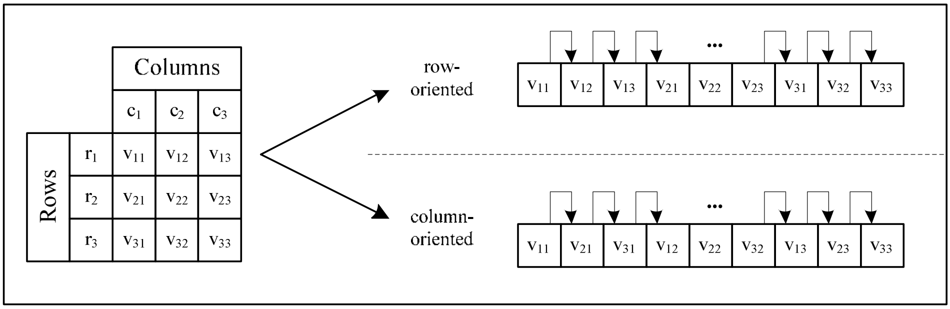

Figure 2: Data storage alternatives (adapted from [PZ11])

Most queries only use a very small amount (about 10%) of the attributes (columns) of a database table. In a column-oriented design only the relevant tuples are accessed in a query and not the complete set of rows as in row-oriented models. Moreover, for most analytical operations only a small set of rows is relevant when a certain condition for an attribute is fulfilled. The accessed attributes of a column-oriented database are shown in Figure 3. Column-oriented storage also allows for better suited compression techniques and gains huge performance impacts – up to factor 1000 with praxis data [Pl09]. Because BI systems are OLAP applications the focus lies on the column-oriented storage in this paper.

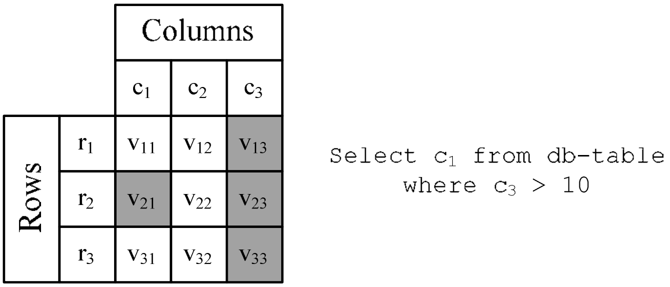

Figure 3: Example query using column-orientation (adapted from [Pl09])

3 In-memory databases as technology enabler for agile BI

3.1 Semi-virtual data warehouse architecture using in-memory technology

The proposed agile BI architecture shown in Figure 4 utilizes a semi-virtual DWH that is based on an IMDB. In addition, the presented architecture respects heterogeneous application landscapes as they can be usually found in most organizations. Like a common DWH the semi-virtual DWH is based on a layered and scalable architecture. In contrast to BI systems as described in section 2.1 the data is kept completely in-memory. Only the extraction layer stores data in this approach and the data is stored in-memory. The depending layers, i.e. harmonisation / cleansing and business logic are modelled logically. The models are kept within metadata storage. Thus, query execution triggers data transformation and aggregation during runtime without actually storing any data physically in the upper layers. Our suggested agile DWH-architecture is semi-virtual in the sense that it is not required to store the data in the BI system if a source system is already IM based. Instead, the data can be accessed directly from the source system like in a VDWH. But, it has to be ensured then that historical data is available in the source system in order to meet the requirements of a DWH. In addition, appropriate storage architectures like column-oriented storage or hybrid approaches [Sc09], i.e. row storage for OLTP and column storage for OLAP are required for direct source system access. On the other hand, data from BI relevant source systems which use non-volatile, physical storage is loaded to the IM based extraction layer of the BI system. This does not contradict to the concept of leanness as a basis of agility (cf. section 2.2). The focus of this research is the construction of an agile BI architecture where the term "lean" is understood in the context of the ability of a BI architecture to adapt to change.

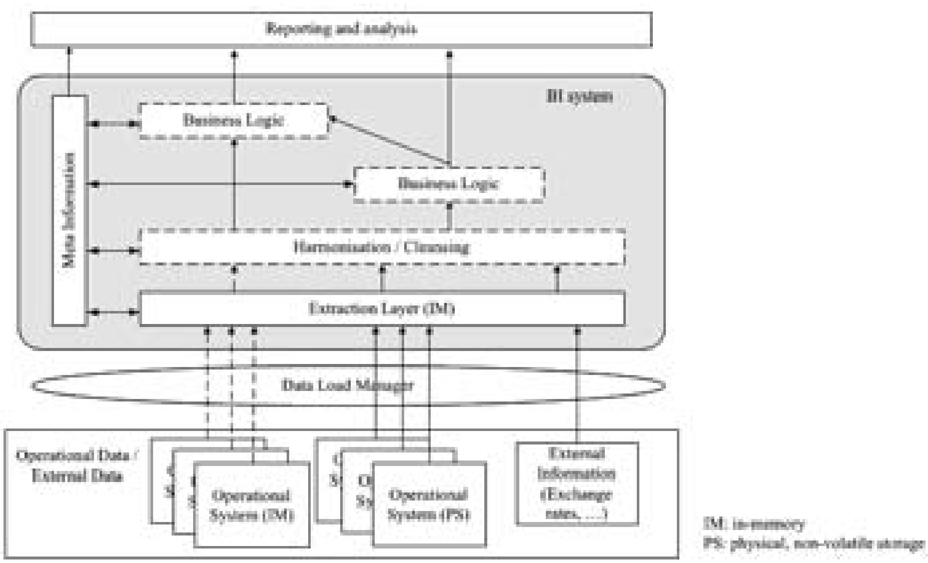

Figure 4: Semi-virtual DWH using IMDB

As stated in section 2, the process of extraction, transformation and loading (ETL) is a critical, usually time consuming task in current DWH architectures. In the approach proposed in this paper the source data is loaded completely and untouched to the extraction layer if the sources are not accessed directly (see above) and is transformed "virtually". Changes or inserts of new records in the underlying source systems can be copied to a delta index [Sc08] or differential buffer [PZ11] which is then merged with the main data store in the extraction layer in the BI system at an appropriate time, e.g. periodically or if a defined threshold is reached. The data load manager is responsible for these activities. To allow for historical tracking an insert-only approach can be followed that does not update existing records but only conducts inserts tagged with a timestamp [Sc08]. Thus, the extraction layer serves as a corporate memory for BI relevant source systems in the presented architecture. Nevertheless, despite economic considerations it is already possible to store the entire data of the world´s largest companies in-memory according to Plattner and Zeier [PZ11]. Recent research shows that on average only a very small percentage of records is changed over time which results in acceptable additional memory and delta operations for the data load manager [Pl09, PZ11].

It is necessary to integrate data in order to get useful and meaningful results as argued by Inmon [In09]. This is an important aspect when taking data from different source systems with varying data models into account. Therefore a VDWH has to provide methods and tools for data integration, which is a key challenge in data warehousing [Ca98]. As the proposed architecture has no physical data storage like magnetic disks these integration steps can be processed logically using meta-data and semantics. Because the focus on recent research activities lay on (operational) one-system environments [Sc08, Pl09, Sc09] there was no need to address this so far. To sum up, a few impacts and changes to previous concepts should be considered and respected.

3.2 Impact on data modelling, data provisioning and analysis

Data modelling: One of the key challenges of BI is the consideration of new or changed requirements [Ba10] and thus their incorporation in the data model. The layered and scalable architecture has for several reasons its right to exist. Nevertheless, it stretches the data model via several layers and especially stores data physically in each layer. However, this is at odds with the shortened response time that businesses are faced with in today's globalized markets. The presented architecture overcomes these shortcomings by utilizing a logical modelling approach. Schmidt-Volkmar has suggested one opportunity for implementing an IM based architecture for operational reporting in a one-system landscape [Sc08]. Adoptions to an existing data model can be incorporated faster as in traditional architectures if logical modelling with real-time transformation is applied. In the upper layers the logical models have a close relationship to business processes and can therefore be modelled by business users. This allows for ad-hoc reporting with new or changed requirements without reloading the complete layered architecture including historical data in the worst case. Hence, applying in-memory technology to BI architecture together with logical data modelling results in a big step forward towards an agile BI as such a system can quickly react to change - a key factor for an agile BI architecture [Co09].

Data analysis: Even the data models of BI systems with strategic focus usually have to be performance optimized to meet response times for analytical queries that often operate on mass data. Nevertheless, if parameters on a query or planning run are changed, the results can often not be expected in justifiable time (within seconds) to be incorporated in a meeting for instance. By utilizing a BI system based on an IMDB using column-oriented data storage the analysis performance significantly increases. This is based on several facts: First, data located in the main memory does not need to be loaded from slower hard disks in order to be processed by the system. Especially when working on high data volumes as in BI systems this creates a huge delay in response time [PZ11]. The second advance is that most queries only use a very small amount of the attributes of a database table (see section 2.3): Column-oriented data organization allows for accessing only relevant tuples during querying. As better compression techniques exist, this even increases performance. The performance increase is even augmented as recent research showed that logically modelled real-time calculation of aggregates perform relatively better for a high number of aggregates in column oriented storage as pre-built, i.e. physically stored, aggregates in row-oriented disk systems [Pl09]. Hence, these improvements allow for a quick reaction to change. Nevertheless, by using semi-virtual DWH with IMDB as technology enabler it even creates change if the deducted actions based on the analysis results lead to new or changed management decisions. Reaction to and creation of change are two important factors of agile BI [Co09].

Data provisioning: Information disposal in "real time" is a critical aspect if BI systems are not only used for strategic reasons but also for operational purposes. To influence a business process, an automated action could be triggered within an operational BI system if a certain condition is fulfilled on an incoming event. The ETL process from source systems through different layers and the physical storage of (partly) redundant information is, however, time consuming for current DWHs. Hence, the common DWH approach is not suited for time-critical BI operations. The slow process of storing data physically in several layers is omitted in the presented approach. In this case the source system is disk based, the data is stored only in the extraction layer of the semi-virtual DWH and, furthermore, the storage is in-memory, not disk. If the source system is already in-memory based with column- or hybrid storage even the latency for transferring the data to the extraction layer is avoided. Besides a reduction of storage space this reduces the time until data is available for reporting or analysis. Transformations can be calculated without the usual step of storing the result in order to offer cleaned, harmonized and consolidated data. As mentioned above, logical modelling together with IM technology can reduce the latency of providing data for analysis. Thus, an IM based, semi-virtual BI system can proactively or reactively embrace change. Especially, if mechanisms are implemented that self-adapt the applied rules or actions, such a system can even learn from and create change. Of course, BI systems as described in section 2.1 can implement and apply such event-condition-action rules as well. But, the latter do not fulfil the time constraints and miss the overall results. Hence, agile BI cannot be achieved by using architectures as mentioned in section 2.1 but by applying IMDB technology in conjunction with VDWH. The proactive reaction on change as well as learning from and creation of change are aspects impacting the agility of a BI system [Co09].

Virtual data warehousing: A VDWH without dedicated data storage shows drawbacks in terms of resource intensive access to the required data in operational systems. Depending on the duration of data storage in the source system, analysis of historical data is often not sufficiently possible in VDWH environments. The presented architecture overcomes these deficiencies of by storing historical information in the extraction layer. As the data is organized in columns in-memory, quick response time even for mass data operations can be expected [PZ11]. The challenge of integrating data from several source systems is addressed by loading the heterogeneous data of non IM source systems to the extraction layer of the proposed architecture. Afterwards, the data is consolidated and integrated in the upper layer by applying logical business transformation rules without storing results physically. Hence, IMDBs can serve as a technology enabler for establishing of VDWHs.

4 Contributions, limitations and outlook

This paper's main contribution is the introduction of a semi-virtual DWH architecture enabled by applying an in-memory database. It identified these aspects as key factors for BI to become technologically more agile. After investigating current BI architectures in the outset, the definition of agility in a BI context exposed shortcomings in current approaches towards agile BI. Mainly since these approaches (e.g. [Sc95, Be03, Hu08]) focus on methods for implementation or management processes that are not always sufficient or applicable to BI [e.g. Mo09] and do not take architectural aspects into account. To extend the current literature, we transferred fundamental components of ISD agility to BI architectures. Afterwards, we evaluated the proposed semi-virtual BI architecture against these components in a qualitative manner. Figure 5 briefly summarizes the evaluation.

Criteria for agility	DWH	Semi-virtual DWH
Creation of change	Partly	Full
Proaction in advance of change	Partly	Full
Reaction to change	Partly	Full
Learning from change	Full	Full
Perceived economy, quality and simplicity	Partly	Future research

Figure 5: Support for criteria concerning agility

Nonetheless, the presented results in Figure 5 should be carefully reviewed in the light of the study's limitations. As stated before, the concept of VDWH is not well established on the market. Hence, our proposal will likely face some skepticism – particularly since the criterion of "perceived economy, quality and simplicity" is still to be evaluated in future research. This leaves us to admit, that the second sub-question of the research - regarding the effects on current BI projects and environments - could not be sufficiently answered to date. Focusing on the enabling technology, we acknowledge that the applied techniques disregard the organizational context.

Hence, our future activities will address these limitations. First, we will apply our architecture to an organizational context in practice. We expect many insights to the application from different perspectives – e.g. the question “to what degree is the prototype really more agile in the field?” can be answered from an organizational (e.g. project- and demand management) as well as technological (eclipsed time on on-the-fly calculations, data transformations, etc.) perspective. We are also keen to investigate whether or not agile process methods correlate with agile BI architectures. On top of that, we plan to investigate how agile BI systems do in terms of economy, quality and simplicity.

5 References

[AC08] Abbasi, A.; Chen, H.: CyberGate: A System and Design Framework for Text Analysis of Computer-Mediated Communication. In: MIS Quarterly, 32 (4) 2008, pp. 811-837.

[Ba10] Baars, H.: Business Intelligence im Spannungsfeld von Agilität und Effizienz. In: Controlling, 22 (12) 2010, pp. 664–671.

[Be01] Beck, K., et al.: Manifesto for Agile Software Development, 2001. http://agilemanifesto.org/. Accessed at 2011-09-18.

[Be03] Beck, K.: Extreme Programming: Die revolutionäre Methode für Softwareentwicklung in kleinen Teams ; [das Manifest]. Addison-Wesley, München, Boston [u.a.], 2003.

[Be11] Beyer, M.: Gartner Says Solving 'Big Data' Challenge Involves More Than Just Managing Volumes of Data, 2011. http://www.gartner.com/it/page.jsp?id=1731916. Accessed at 2011-10-25.

[Ca98] Calvanese, D., et al.: Description Logic Framework for Information Integration. In: Proc. of the 6th Int. Conf. on the Principles of Knowledge Representation and Reasoning (KR’98), 1998, pp. 2–13.

[CG98] Chamoni, P.; Gluchowski, P.: Analytische Informationssysteme - Einordnung und Überblick. In: Analytische Informationssysteme, Springer, Berlin [u.a.], 1998, pp. 3–25.

[CG04] Chamoni, P.; Gluchowski, P.: Integrationstrends bei Business-Intelligence-Systemen. In: WIRTSCHAFTSINFORMATIK, 46 (2) 2004, pp. 119-128.

[CCD06] Chenoweth, T.; Corral, K.; Demirkan, H.: Seven Key Interventions for Data Warehouse Success. In: Communications of the ACM, 49 (1) 2006, pp. 114-119.

[CCN05] Chung, W.; Chen, H.; Nunamaker, J.F.: A Visual Knowledge Map Framework for the Discovery of Business Intelligence on the Web. In: Journal of Management Information Systems, 21 (4) 2005, pp. 57-84.

[Co09] Conboy, K.: Agility from first principles: Reconstructing the concept of agility in information systems development. In: Information systems research, 20 (3) 2009, pp. 329–354.

[Cu10] Cukier, K.: Data, data everywhere. In: The Economist 2010. http://www.economist.com/specialreports/displaystory.cfm?story_id=15557443. Accessed at 2011-10-25.

[Cu03] Cutter Consortium: Cutter Consortium Report on Corporate Use of BI and Data Warehousing Technologies, 2003.

[De84] DeWitt, D.J., et al.: Implementation techniques for main memory database systems. In: Proceedings of the 1984 ACM SIGMOD international conference on Management of data, ACM, New York, NY, USA, 1984, pp. 1-8.

[GS92] Garcia-Molina, H.; Salem, K.: Main Memory Database Systems: An Overview. In: IEEE Trans. on Knowl. and Data Eng., 4 (6) 1992, pp. 509-516.

[Ga09b] Gartner: Gartner Says Organisations Can Save More Than $500,000 Per Year by Rationalising Data Integration Tools, 2009. http://www.gartner.com/it/page.jsp?id=944512. Accessed at 2009-11-19.

[Ga09a] Gartner: Gartner EXP Worldwide Survey of More than 1,500 CIOs Shows IT Spending to Be Flat in 2009, 2009.

[GH11] Göhl, R.; Hahne, M.: Agile Konzepte für BI: Bessere Architektur, Organisation und Methodik. In: BI-Spektrum, 6 (2) 2011, pp. 10–16.

[Ha02] Hahne, M.: Logische Modellierung mehrdimensionaler Datenbanksysteme Deutscher Universitätsverlag, Wiesbaden, 2002.

[Ha10] Hahne, M.: Design des Enterprise Data Warehouse: Modellieren mehrschichtiger Architekturen. In: BI-Spektrum, 5 (4) 2010, pp. 8–12.

[HL04] Haupt, J.; Lehmann, P.: Aspekte des unternehmensweiten Data Warehousing am Beispiel SAP Business Information Warehouse. In Proceedings of the Multikonferenz Wirtschaftsinformatik (MKWI), 2004.

[HS10] Hawking, P.; Sellitto , C.: Business Intelligence (BI) Critical Success Factors. In Proceedings of the Australasian Conference on Information Systems (ACIS), Vol. Paper 4, Brisbane, Australia, 2010.

[Hu99] Huang, C.-C.: An agile approach to logical network analysis in decision support systems. In: Decision Support Systems, 25 (1) 1999, pp. 53–70.

[Hu08] Hughes, R.: Agile Data Warehousing: Delivering World-Class Business Intelligence Systems Using Scrum and XP. iUniverse, New York, 2008.

[HH05] Hwang, M.I.; Hongjiang, X.: A Survey of Data Warehousing Success Issues. In: Business Intelligence Journal, 10 (4) 2005, pp. 7-14.

[In98] Inmon, B.: The Operational Data Store: Designing the Operational Data Store: In Information Management Magazine, 1998. http://www.information-management.com/issues/19980701/469-1.html. Accessed at 2011-09-02.

[In00] Inmon, B.: ODS Types: Information Management: Charting the Course. In: Information Management Magazine 2000. http://www.information-management.com/issues/20000101/1749-1.html. Accessed at 2011-09-02.

[In04] Inmon, B.: The Virtual Data Warehouse – Transparent and Superficial: Information Management: Charting the Course. In: Information Management Magazine 2004. http://www.information-management.com/issues/20040301/8205-1.html. Accessed at 2011-09-17.

[In09] Inmon, B.: The Elusive Virtual Data Warehouse, 2009. http://www.b-eye-network.com/view/9956. Accessed at 2011-09-16.

[JC99] Joshi, K.; Curtis, M.: Issues in Building a Successful Data Warehouse. In: Information Strategy, 15 (2) 1999, pp. 28-35.

[KMU06] Kemper, H.-G.; Mehanna, W.; Unger, C.: Business Intelligence - Grundlagen und praktische Anwendungen: eine Einführung in die IT-basierte Managementunterstützung. 2., erg. Aufl. Aufl. Vieweg, Wiesbaden, 2006.

[Ki96] Kimball, R.: The data warehouse toolkit: Practical techniques for building dimensional data warehouses. Wiley, New York [u.a.], 1996.

[La11] Lahrmann, G., et al.: Business Intelligence Maturity: Development and Evaluation of a Theoretical Model. In Proceedings of the Hawaii International Conference on System Sciences (HICSS), Kauai, Hawaii, 2011.

[MH07] March, S.T.; Hevner, A.R.: Integrated Decision Support Systems:A Data Warehousing Perspective. In: Decision Support Systems, 43 (3) 2007, pp. 1031-1043.

[Mo09] Moss, L.: Beware of Scrum Fanatics On DW/BI Projects. In: EIMInsight MAGAZINE, 3 (3) 2009. http://www.eiminstitute.org/library/eimi-archives/volume-3-issue-3-march-2009-edition/beware-of-scrum-fanatics-on-dw-bi-projects. Accessed at 2011-09-17.

[OZ07] Olszak, C.M.; Ziemba, E.: Approach to Building and Implementing Business Intelligence Systems. In: Interdisciplinary Journal of Information, Knowledge, and Management, 2 2007, pp. 135-148.

[Pa09] Pankaj, P., et al.: Revisiting agility to conceptualize information systems agility. In: Emerging topics and technologies in information sytems, Information Science Reference, Hershey, Pa. [u.a.], 2009, pp. 19–54.

[Pl09] Plattner, H.: A Common Database Approach for OLTP and OLAP Using an In-Memory Column Database. In: Proceedings of the 35th SIGMOD International Conference on Management of Data, Providence, Rhode Island, 2009.

[PZ11] Plattner, H.; Zeier, A.: In-Memory Data Management: An Inflection Point or Enterprise Applications. Springer, Berlin, 2011.

[Sc09] Schaffner, J., et al.: A hybrid row-column OLTP database architecture for operational reporting. In: Business intelligence for the real-time enterprise, Springer, Berlin [u.a.], 2009, pp. 61–74.

[SBM99] Schinzer, H.D.; Bange, C.; Mertens, H.: Data Warehouse und Data Mining: Marktführende Produkte im Vergleich. 2., völlig überarb. und erw Aufl. Vahlen, München, 1999.

[Sc08] Schmidt-Volkmar, P.: Betriebswirtschaftliche Analyse auf operationalen Daten. Betriebswirtschaftlicher Verlag Gabler, Wiesbaden, 2008.

[Sc95] Schwaber, K.: SCRUM Development Process. In: Proceedings of the 10th Annual ACM Conference on Object Oriented Programming Systems, Languages, and Applications (OOPSLA), 1995, pp. 117-134.

[Sh02] Shim, J.P., et al.: Past, Present, and Future of Decision Support Technology. In: Decision Support Systems, 33 (2) 2002, pp. 111-126.

[Sh03] Shin, B.: An Exploratory Investigation of System Success Factors in Data Warehousing. In: Journal of the Association for Information Systems, 4 (1) 2003, pp. 141-170.

[St05] Stonebraker, M., et al.: C-Store: A Column-oriented DBMS. In: Proceedings of the 31st international conference on Very large data bases, VLDB Endowment, 2005, pp. 553-564.

[VBP10] Virparia, P.V.; Buch, S.H.; Parabia, R.F.: Trade and Tricks: Traditional vs. Virtual Data Warehouse. In: International Journal of Advanced Engineering & Application 2010, pp. 220–224.

[WW07] Watson, H.J.; Wixom, B.H.: The Current State of Business Intelligence. In: IEEE Computer, 40 (9) 2007, pp. 96-99.

[Wi10] Winter, R.: Zur Positionierung und Weiterentwicklung des Data Warehousing in der betrieblichen Applikationsarchitektur. In: Data-Warehousing-Strategie, Springer, Berlin [u.a.], 2000, pp. 127–139.

[WW01] Wixom, B.H.; Watson, H.J.: An Empirical Investigation of the Factors Affecting Data Warehousing Success. In: MIS Quarterly, 25 (1) 2001, pp. 17-41.

[WW10] Wixom, B.H.; Watson, H.J.: The BI-Based Organization. In: International Journal of Business Intelligence Research, 1 (1) 2010, pp. 13-28.