

Gesellschaft für Informatik e.V. (GI)

publishes this series in order to make available to a broad public recent findings in informatics (i.e. computer science and information systems), to document conferences that are organized in cooperation with GI and to publish the annual GI Award dissertation.

Broken down into

- seminars
- proceedings
- dissertations
- thematics

current topics are dealt with from the vantage point of research and development, teaching and further training in theory and practice. The Editorial Committee uses an intensive review process in order to ensure high quality contributions.

The volumes are published in German or English.

Information: <http://www.gi.de/service/publikationen/lni/>

ISSN 1617-5468  
ISBN 978-88579-287-1

The conference "IMDM 2011" was focusing on the potentials of in-memory data management for business information systems. With in-memory technology it is possible to store huge volumes of single documents from enterprises directly in main memory for high-speed processing. Flexible analyses of actual and historic data are then possible in real time. This leads to new potentials for existing and innovative business processes. The proceedings contain all contributions from the refereed program of the conference.



Wolfgang Lehner, Gunther Piller (Hrsg.): IMDM 2011

# GI-Edition

## Lecture Notes in Informatics

**Wolfgang Lehner, Gunther Piller (Hrsg.)**

## IMDM 2011

### Proceedings zur Tagung Innovative Unternehmensanwendungen mit In-Memory Data Management

**2. Dezember 2011  
Mainz**

## Proceedings





Wolfgang Lehner, Gunther Piller (Hrsg.)

**Innovative Unternehmensanwendungen mit  
In-Memory Data Management**

**Beiträge der Tagung IMDM 2011**

**2.12.2011**

**in Mainz**

Gesellschaft für Informatik e.V. (GI)

## **Lecture Notes in Informatics (LNI) - Proceedings**

Series of the Gesellschaft für Informatik (GI)

Volume P-193

ISBN 978-3-88579-287-1

ISSN 1617-5468

### **Volume Editors**

Prof. Dr. Wolfgang Lehner

Technische Universität Dresden

Fakultät Informatik, Institut für Systemarchitektur

01062 Dresden, Germany

Email: wolfgang.lehner@tu-dresden.de

Prof. Dr. Gunther Piller

Fachhochschule Mainz

Fachbereich Wirtschaft

55128 Mainz, Germany

Email: gunther.piller@fh-mainz.de

### **Series Editorial Board**

Heinrich C. Mayr, Alpen-Adria-Universität Klagenfurt, Austria

(Chairman, mayr@ifit.uni-klu.ac.at)

Hinrich Bonin, Leuphana Universität Lüneburg, Germany

Dieter Fellner, Technische Universität Darmstadt, Germany

Ulrich Flegel, Hochschule Offenburg, Germany

Ulrich Frank, Universität Duisburg-Essen, Germany

Johann-Christoph Freytag, Humboldt-Universität zu Berlin, Germany

Michael Goedicke, Universität Duisburg-Essen, Germany

Ralf Hofestädt, Universität Bielefeld, Germany

Michael Koch, Universität der Bundeswehr München, Germany

Axel Lehmann, Universität der Bundeswehr München, Germany

Ernst W. Mayr, Technische Universität München, Germany

Thomas Roth-Berghofer, DFKI, Germany

Sigrid Schubert, Universität Siegen, Germany

Martin Warnke, Leuphana Universität Lüneburg, Germany

### **Dissertations**

Steffen Hölldobler, Technische Universität Dresden, Germany

### **Seminars**

Reinhard Wilhelm, Universität des Saarlandes, Germany

### **Thematics**

Andreas Oberweis, Karlsruher Institut für Technologie (KIT), Germany

© Gesellschaft für Informatik, Bonn 2011

**printed by** Köllen Druck+Verlag GmbH, Bonn

# Vorwort

Technologische Innovationen können neue Geschäftsprozesse unterstützen oder auch erst möglich machen. Welches Potenzial bietet hier In-Memory Data Management? Das war die zentrale Frage der Tagung IMDM2011. Mit In-Memory Data Management ist es möglich, große Volumen von Einzelbelegen von Unternehmen direkt in Hauptspeichern zu halten und dort mit hoher Geschwindigkeit zu bearbeiten. Flexible Analysen großer Mengen aktueller und historischer Daten sind somit in Echtzeit möglich und erlauben unmittelbare Einblicke in Unternehmensprozesse.

Schwerpunkt der Tagung war die Diskussion neuer Möglichkeiten, die In-Memory Data Management für bestehende und zukünftige Geschäftsprozesse bieten könnte. Typische Fragestellungen waren:

- In welchen Geschäftsszenarien in Industrie und Handel, der Finanzbranche, im Energiesektor oder in der Telekommunikation bieten schnelle Analysen großer Datenmengen neue Potenziale?
- Was sind vielversprechende Anwendungsbereiche für Echtzeitanalysen, Simulationen oder interaktive Planungsläufe?
- Welche Arten von Geschäftsprozessen und -anwendungen können von In-Memory Data Management am meisten profitieren?

Ziel der Konferenz war es, vielversprechende Forschungs- und Anwendungsbereiche für In-Memory-Computing und innovative Geschäftsprozesse in Unternehmen zu identifizieren.

Die Veranstaltung richtete sich an Wissenschaftler im Bereich Wirtschaftsinformatik, Informatik und der Betriebswirtschaftslehre. Leitende Mitarbeiter von Softwarehersteller und Anwenderunternehmen waren auch herzlichst eingeladen.

Wolfgang Lehner  
Gunther Piller

Mainz, Dezember 2011

## **Programmkomitee**

Hans-Jürgen Appelrath, Universität Oldenburg  
Peter Chamoni, Universität Duisburg-Essen  
Carsten Felden, Technische Universität Bergakademie Freiberg  
Peter Gluchowski, Technische Universität Chemnitz  
Alfons Kemper, Technische Universität München  
Hans-Georg Kemper, Universität Stuttgart  
Wolfgang Lehner, Technische Universität Dresden  
Peter Loos, Universität des Saarlandes  
Marco Meier, Universität Augsburg  
Gunther Piller, Fachhochschule Mainz  
Robert Winter, Universität St. Gallen  
Alexander Zeier, Hasso-Plattner-Institut Potsdam

## **Organisation**

Wolfgang Lehner, Technische Universität Dresden  
Gunther Piller, Fachhochschule Mainz

## **Sponsoren**

**SAP AG**



**TDWI**



**TU Dresden**



**FH Mainz**



# Inhaltsverzeichnis

## **Reinhard Schütte**

*Analyse des Einsatzpotenzials von In-Memory-Technologien in  
Handelsinformationssystemen* .....1

## **Peter Chamoni**

*BI-Strategie zum Ausgleich von Technologie-Push und Business-Pull*.....13

## **Robert Winter, Stefan Bischoff, Felix Wortmann**

*Revolution or Evolution? Reflections on In-Memory Appliances from an Enterprise  
Information Logistics Perspective* .....23

## **Johannes Wust, Jens Krueger, Sebastian Blessing, Cafer Tosun, Alexander Zeier, Hasso Plattner**

*xSellerate: Supporting Sales Representatives with Real-Time Information in Customer  
Dialogs* .....35

## **Gunther Piller, Jürgen Hagedorn**

*Business Benefits and Application Capabilities Enabled by In-Memory Data  
Management*.....45

## **Maik Thiele, Wolfgang Lehner, Dirk Habich**

*Data-Warehousing 3.0 – Die Rolle von Data-Warehouse-Systemen auf Basis von  
In-Memory Technologie*.....57

## **Marco Pospiech, Carsten Felden**

*Realisierung einer serviceorientierten Business Intelligence Architektur anhand von  
In-Memory-Technologien*.....69

## **Henrik Mühe, Alfons Kemper, Thomas Neumann**

*The Mainframe Strikes Back: Multi Tenancy in the Main Memory Database HyPer on a  
TB-Server, .....*83

## **Andreas Emrich, Dirk Werth, Peter Loos**

*Real-Time Semantic Process Change Impact Analysis* .....91

## **Tobias Knabke, Sebastian Olbrich**

*Towards Agile BI: Applying In-Memory Technology to Data Warehouse Architectures  
.....*101

## **Marco C. Meier, Alexa Scheffler**

*Ökonomisch sinnhafte Bewertung von In-Memory-basierten betrieblichen  
Informationssystemen*.....115





# Analyse des Einsatzpotenzials von In-Memory-Technologien in Handelsinformationssystemen

Reinhard Schütte

EDEKA AG  
New-York Ring 6  
22297 Hamburg  
reinhard.schuette@edeka.de

**Abstract:** In-Memory-Technologien eröffnen vor allem bei Anwendungen, in denen große Datenmengen zu verarbeiten sind, interessante Einsatzpotentiale. In dem Beitrag wird mit dem Lunar-Programm das weltweit größte Projekt im Handel vorgestellt, um darauf aufbauend Einsatzpotentiale aus einer betriebswirtschaftlichen Perspektive in und aus einer technologischen Perspektive beim Betrieb von Handelsinformationssystemen heraus zu identifizieren.

## 1 In-Memory-Technologien

Die traditionelle Informationsverarbeitung ist geprägt von einer Dualität von Datenstrukturen und Algorithmen, wie es in der Informatik seit Jahrzehnten üblich ist. Auch wenn die Objektorientierung als technisches Konstrukt Eingang in die Informationssystemwelt gefunden hat, die konzeptionelle Analyse ist weiterhin sehr von einer prozessorientierten Sichtweise geprägt, die vor allem die Funktionen fokussiert. Die Betonung der Applikationen und die davon getrennte Datenhaltung findet ihren Ausdruck in der Art und Weise, wie Daten und Applikationen architektonisch verzahnt werden, was letztlich dazu führt, dass die Daten weitgehend auf Festplatten oder im Cache gespeichert und nicht im Arbeitsspeicher vorgehalten werden.

Die In-Memory-Technologiekonzepte, die aus der betrieblichen Praxis bzw. von Softwareunternehmen propagiert werden, lösen sich von den traditionellen Architekturgrundsätzen und nehmen eine Speicherung der Daten im Arbeitsspeicher vor [LV11, S. 384], so dass enorme Performance-Vorteile in Aussicht gestellt werden [PI09]. Einhergehend mit den Performance-Versprechungen sind die konzeptionellen Defizite nicht zu vernachlässigen, denn die Datenhaltung im Arbeitsspeicher ist momentan nur mit Snapshots, Replikationen oder Transaction Logs möglich, so dass nicht unbedingt die „elegantesten Lösungsansätze“ zur Unterstützung der Datenhaltung eingesetzt werden.

Ungeachtet der grundsätzlichen technologischen Fragestellung widmet sich der Beitrag zwei wesentlichen Analysefeldern: Der Identifikation von Einsatzgebieten der In-Memory-Technologien aus einer betriebswirtschaftlichen Perspektive im Umfeld von Handelsinformationssystemen und aus einer technologiezentrierten Perspektive beim Betrieb von Handelsinformationssystemen.

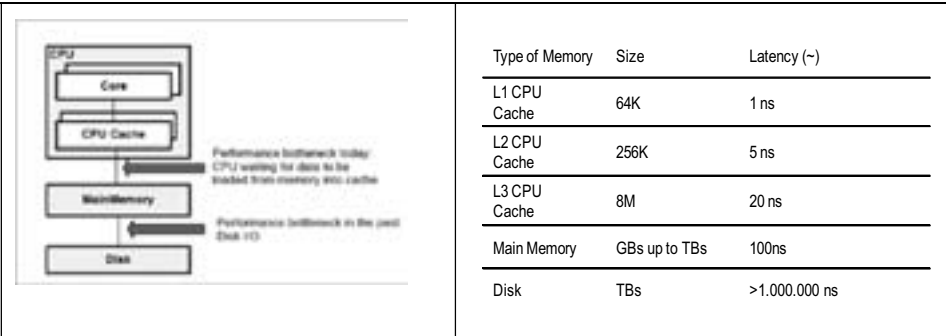


Abbildung 1: Die Grundidee einer In-Memory-Technologie [Si11]

In-Memory-Technologien werden hier verstanden als technologiegetriebene Ansätze, die die Speicherung und Verwaltung der Daten im Hauptspeicher vorsehen. Damit sind In-Memory-Ansätze vor allem dann sinnvoll, wenn an die Datenverarbeitung Echtzeitanforderungen bestehen.

## 2 Handelsinformationssysteme

Handelsinformationssysteme werden in Anlehnung an Becker/Schütte als umfassende Anwendungssysteme verstanden, die sämtliche Prozesse in Handelsunternehmen unterstützen (vgl. Abbildung 2). Im Gegensatz dazu dienen Warenwirtschaftssysteme den warenorientierten dispositiven, logistischen und abrechnungsbezogenen Prozessen, die für die Durchführung der Geschäftsprozesse eines Handelsunternehmens erforderlich sind [BS04, S. 15].

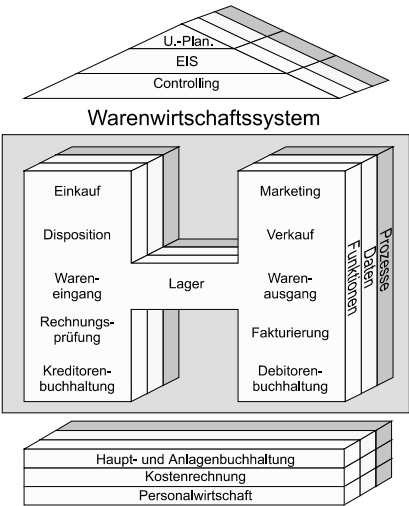


Abbildung 2: Handelsinformationssysteme und Abgrenzung zu Warenwirtschaftssystemen [BS04]

Bei Handelsunternehmen existieren aufgrund der traditionell hohen Datenvolumina besondere Anforderungen an die Performance der Applikationssysteme sowie der Hardwareeinheiten. Daher erscheint die Domäne von Handelsinformationssystemen besonders geeignet zu sein, Anwendungsfelder für den Einsatz von In-Memory-Technologien zu liefern.

## 2.1 Das Lunar-Programm bei der EDEKA: inhaltlich-funktionale Perspektive

Das Lunar-Programm, in dem die EDEKA-Gruppe bis Ende 2012 ca. 350 Mio. € in die Entwicklung von neuen Softwareprodukten investiert und sämtliche EDEKA-Einheiten adressiert werden, bildet sämtliche Anforderungen der EDEKA-Gruppe inhaltlich-funktional gemäß der dreistufigen Organisationsstruktur in sechs Programmbereichen ab (vgl. Abbildung 3).

Die Basis der Systemlandschaft bildet das EDM – EDEKA Daten Management, in dem national die Stammdaten angelegt und gepflegt werden mit sämtlichen Daten, die aus einer nationalen Perspektive für alle Subsysteme benötigt werden. Aus dem EDM werden die jeweils für die Subsysteme (die Groß- und Einzelhandelswarenwirtschaftssysteme) als relevant deklarierten Artikel und Mengeneinheiten mit den jeweiligen Initialinformationen sowie den Aktualisierungen versorgt. Das System führt aktuell ca. 1,5 Millionen Stammdaten und hat zudem diverse Mengeneinheiten je Artikel und mehrere hundert Attribute je Artikel aufzuweisen. Das alleine durch die Stammdaten induzierte Mengenvolumen ist nicht zu unterschätzen.



Abbildung 3: Die inhaltlich-funktionale Perspektive auf das Lunar-Programm

Basierend auf den Stammdaten sind die drei Blöcke EDEKA Zentrale, Großhandel und Einzelhandel als zusammenhängende, dreistufige Wertschöpfungskette zu verstehen, die in dieser Kombination die EDEKA prägt:

*EDEKA-Zentrale:* In der EDEKA-Zentrale werden aus Applikationssicht vier wesentliche Aufgaben wahrgenommen, das nationale Warengeschäft für Marken- und Eigenmarkeneinkauf, der Betrieb der Fruchtkontore, der Ein- und Verkauf von Non-Food-Ware sowie die Zentralregulierung nebst begleitenden Aufgaben.

Für den Einkauf wird ein nationales BI-System betrieben, in dem für mehr als 8.000 Betriebe artikelgenaue Abverkaufsinformationen gespeichert werden. Diese Informationen werden in einem dezidierten Kategorienreporting mit Marktforschungsdaten zusammengeführt, d. es findet die Integration von externen Daten zur Wettbewerbsentwicklung im Markt mit eigenen Daten statt. Neben den Abverkaufsinformationen sind auch die Daten aus der Zentralregulierung artikelgenau vorhanden.

Die Zentralregulierung für annähernd 25 Mrd. Euro Umsatz erfolgt artikelgenau für 23 Millionen Belege pro Jahr, so dass ca. 400 Mio. Belegpositionen pro Jahr in dem SAP Agency Business System gebucht werden. Neben der Zentralregulierung werden die rückwärtigen Konditionen in dem System artikelgenau auf Basis der eigenen Daten abgerechnet.

*Großhandel:* In den sieben Großhandelsbetrieben, die handelsrechtlich betrachtet jeweils eigene Konzerne mit einem Umsatz zwischen 3 – 7 Mrd. € sind, werden sämtliche Aufgaben eines Handelsinformationssystems unterstützt (vgl. Abbildung 2). Ausgehend von den aus der Edeka-Zentrale referenzierten Stammdaten werden diese um weitere, für die regionale Handelstätigkeit erforderliche Stammdaten angereichert. Die Aufgaben des Beschaffungsprozesses (für das Lagergeschäft) werden ebenso wahrgenommen wie die entsprechenden Aufgaben auf der Distributionsseite, um einerseits die Belieferung für die Selbständigen Kaufleute und andererseits die der eigenen Regiebetriebe vornehmen zu können. Bei den Regiebetrieben werden auch die Aufgaben des Marketings mit verantwortet, während bei den Selbständigen Kaufleuten die Ausprägungen der 4-Ps des Marketing-Mixes in der Verantwortung dieser selbst liegt. Die Logistik in der EDEKA-Gruppe umfasst dabei mehr als 50 Lagerstandorte und es werden sämtliche Belieferungsformen (Lagerlieferungen aus dem Bestand, Lagerlieferungen über Cross Docking I und II für die Belieferung von fremden und eigenen Lieferanten (Produktionsbetriebe der EDEKA Großhandlungen oder der EDEKA Zentrale)) unterstützt. Die Datenvolumina sind in der Logistik (Kommissionierung, Warenausgang) sowie den angrenzenden Prozessen der Fakturierung hoch und erfordern insbesondere in den besonders verarbeitungsintensiven Zeiträumen am Vormittag eine enorme Verarbeitungsgeschwindigkeit.

*Einzelhandel:* Im Einzelhandel werden auch die kompletten Aufgaben eines Handelsinformationssystems abgebildet, wobei die rückwärtigen Prozesse der Finanzbuchhaltung und des Controllings weniger stark ausgeprägt sind als in der Großhandlung. Es sind dabei die Besonderheiten zu beachten, die insbesondere bei einer zentralen IT-Architektur bestehen, wenn für mehrere tausend Märkte Anforderungen an die Verarbeitungsfähigkeit in einem engen Zeitintervall gestellt werden.

*Supply Chain Management:* Das Supply Chain Management wird durch eine Plattformlösung unterstützt, die für das Dispositions- und Bestandsmanagement, das Transportmanagement, das Behältermanagement, die Verfügbarkeitsprüfung, das WE-Dispatching und die USP-(Umschlagspunkt)Steuerung zuständig ist. Dabei wird eine neuartige Integration über unterschiedliche Sparten eines Handelsunternehmens (Großhandelsbelieferung, Cash+Carry sowie Fleischwerke) möglich, da die Plattform eine entscheidungsorientierte Steuerung der Logistik vornimmt.

*Business Intelligence:* Die Auswertungen über das operative Geschäft werden auf drei Ebenen unterstützt. Auf der Ebene der EDEKA Zentrale wird das nationale BI-System eingesetzt, um das nationale Warengeschäft zu unterstützen (s.o.). Auf der regionalen Ebene werden die BI-Systeme sowohl für die Großhandels- als auch die Einzelhandelsaufgaben benötigt, um das operative Geschäft zu unterstützen. Die Auswertungen reichen von Umsatz- und Artikelanalysen über Personalkostenbetrachtungen bis hin zu Ergebnisrechnungen.

## 2.2 Das Lunar-Programm bei der EDEKA: technologische Perspektive

Aus technischer Sicht ist das Lunar-Programm besonders anspruchsvoll, da die Systeme als Templates entwickelt und von dem Lunar-Rechenzentrum in Hamburg aus in die einzelnen Einheiten überführt und dort selbst betrieben werden. Aus diesem Grund sind in der EDEKA-Gruppe mehr als 300 SAP-Systeme im Einsatz, davon alleine 140 im Lunar Rechenzentrum. Die Lunar-Applikationslandschaft stellt ein komplettes System-Modell über alle Handelsstufen, Entwicklungsstufen und Programmbereiche dar.

Die Systemvielfalt, die in heutigen netzartigen Applikationslandschaften üblich sind, führt zu enormen Herausforderungen u.a. bei Releasewechseln. Die Lunar GmbH ist für die EDEKA-Gruppe der Softwarelieferant, der in Releasezyklen die Software entwickelt und in fest definierten Zeitabständen in die EDEKA-Einheiten zum Betrieb überführt, nachdem Test in der Lunar GmbH und bei den einsetzenden Einheiten erfolgt sind. Damit sind die Anforderungen an die Entwicklung und Weiterentwicklung von Produkten und Produktfamilien mit den Aufgaben eines Standardsoftwareherstellers vergleichbar.

Die Systeme werden in einem x86-Systemkonzept (vgl. Abbildung 4) betrieben, wobei die Virtualisierung der Server sowie das Oracle RAC zentrale Komponenten der technischen Konzeption bilden.

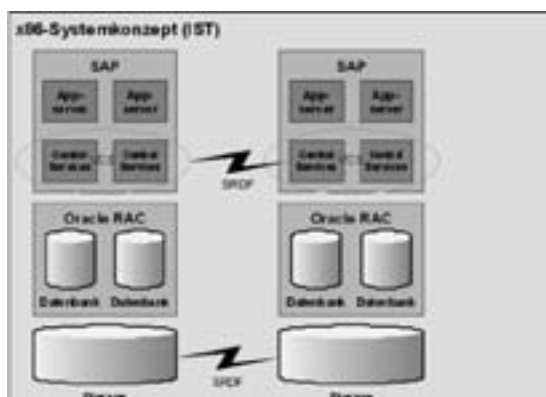


Abbildung 4: Technische Systemkonzeption bei Lunar

## 3 Einsatzpotentiale von In-Memory-Technologien im Handel

### 3.1 Inhaltlich-funktionale Perspektive

Aus betriebswirtschaftlicher Sicht bestehen vor allem dann Herausforderungen an die Geschwindigkeit der Datenverarbeitung, wenn die Anforderungen entweder nicht umsetzbar sind oder erst bei schnelleren Verarbeitungsgeschwindigkeiten von den Anwendern akzeptiert werden. Es sollen nachfolgend einige Aufgaben diskutiert werden, die sich für den Einsatz von In-Memory-Technologien besonders geeignet erweisen könnten.

#### 1. Beispiel Disposition

Bei der Disposition sind diverse Anforderungen im Rahmen der Bedarfsprognoserechnung wie die Promotions, die Abverkaufsdaten, die Marktdaten und die Profitabilität (Entwicklung der Einkaufspreise in der Zukunft) zu beachten (vgl. Abbildung 5).



Abbildung 5: Beispiel Disposition

Heutzutage werden die Optimierungsläufe in einer Forecast & Replenishment Engine im SAP-Umfeld durchgeführt, wobei diese Berechnungen erst nachts nach der Verbuchung der Bestände auf Basis von Wareneingängen und Abverkäufen erfolgen können.

Die Zeitdauer für derartige Berechnungen ist in Abhängigkeit von der Anzahl an Betrieben und Sortimenten häufig mehrere Stunden umfassend. Die Ergebnisse der Optimierungen stehen erst morgens für die Bearbeitung in den Betrieben zur Verfügung. Die Ergebnisse werden in normale Bestellvorschläge, die keine weitere Bearbeitung durch einen Disponenten bedürfen, und Exceptions unterschieden, die eine Nachbearbeitung erfordern. Bei der Nachbearbeitung sind der Bestand, die Abverkaufsausreißer, etc. zu prüfen. Nach der Rückmeldung der neu erfassten Datensituation ist es aufgrund der Zeitknappheit für die Dispositionsrückmeldung nicht mehr möglich, einen erneuten Optimierungslauf durchzuführen, sondern die Bestellmenvorschläge werden lediglich durch die Rückmeldung korrigiert. Damit erfolgt bei diesen Bestellmengen keine tatsächliche Optimierung. Durch den Einsatz von In-Memory-Technologien könnte dieser Umstand korrigiert werden und es wäre möglich, betriebswirtschaftlich bessere Ergebnisse bei der Lösung derartiger Entscheidungsmodelle zu erlangen, indem

ein erneuter Optimierungslauf nach Korrektur der Daten in einem neuen Restriktionsnetz erfolgt.

Aus diesem Beispiel lässt sich eine allgemeine Regel zur Identifikation von Einsatzpotentialen von In-Memory-Technologien ableiten: betriebswirtschaftliche Optimierungskalküle werden auch für datenintensivere Anwendungen einsetzbar, was im Handel in vielen Situationen ein erhebliches Optimierungspotential ergibt.

## *2. Beispiel Aktionen*

Bei der Gestaltung von Aktionen in Handelsunternehmen werden die in einer Aktion zu berücksichtigenden Artikel auf Basis von Wochenumsätzen auf einer taktischen Planungsebene sowie nach Thema und Warenbereich geplant.

Systemtechnisch werden die Artikel zu einer Aktion mit einem Vorlauf von einigen Wochen vor Aktionsbeginn angelegt, um die operativen Folgeprozesse – Disposition, Preisaktivierung, Werbemittelerzeugung, etc. anstoßen zu können. Für die Artikelfestlegung sind dabei Abverkaufsdaten aus der Historie erforderlich, die in heutigen Systemen im BI / DW gespeichert sind. Die Festlegung der Artikel aber wiederum erfolgt in einem transaktionalen System. An dieser Stelle wird deutlich, wie die künstliche Trennung dieser beiden Systeme ohne Integration in einem Arbeitsschritt für den Anwender die betriebswirtschaftliche Realität künstlich trennt. Es sind transaktionale Aufgaben und analytische Aufgaben simultan zu erfüllen und nicht getrennt, die Unterscheidung zwischen beiden Aufgabenarten ist das Ergebnis einer aus technischen Performance-Gründen hervorgerufenen Trennung. Als Lösungsoptionen gibt es heute nur die „künstliche“ Anordnung von operativen Aufgaben in einem Data Warehouse oder die Extraktion von Daten aus einem Business Warehouse in ein transaktionales System.

Neben dem Aktionsprozess sind analoge Anforderungen auch im Open-to-buy-Prozess vorhanden, in dem die Trennung zwischen dem Warenwirtschaftssystem und dem Data Warehouse zu Problemen führt. Auch bei der Planung der Konzernumsätze und ergebnisse, die heute ebenfalls in einem BW (z.B. SAP mit dem BPS-Systeme - Business Planning and Simulation) erfolgt und nicht integriert in den Rechnungswesensystemen. Die Entwicklungen zur Integration von OLAP und OLTP werden demzufolge in der Literatur auch bereits als OLXP bezeichnet [ZKM11].

## *3. Beispiel Marketing*

Besonders hohe Anforderungen bestehen in Handelsunternehmen, wenn die Absatz- und Umsatzanalyse entsprechend der aktuell unter dem Rubrum „Business Analytics“ diskutierten Konzepte vorgenommen werden soll. Beispielsweise gibt es in den USA Handelsunternehmen, die sämtliche Artikelpreise der Wettbewerber bis auf den einzelnen Markt in den eigenen Systemen abbilden und damit den gesamten Wettbewerb in ihren Systemen simulierbar machen. Dies führt zu einem exponentiellen Datenwachstum, die bei einer zeitgerechten Preissetzung – idealtypisch in Echtzeit – ebenfalls nur mit In-Memory-Technologien umsetzbar sein dürften.





jektbezogenen Buchungsproblematik hervorgerufenen Rechnerbelastungen bei Massendatenanforderungen nicht empfohlen wird. Die SAP empfiehlt im Handel stattdessen die Profit Center Analysis. Die beim Einsatz von In-Memory-Technologien zur Verbesserung der Performance bei der Ergebnisrechnung gewonnenen Erkenntnisse (vgl. Abbildung 7) lassen den Einsatz von diesem Konzept damit erstmalig auch für den Handel möglich werden.

	ERP w/o HANA (production Hardware)	ERP with HANA (SAP test Installation)	Acceleration factor vs. ERP
EBIT with commodity sales – initial report	260 sec	7 sec (DB 2.8 sec) <sup>7</sup>	40
EBIT with commodity sales – drilldown by alphacode	620 sec	5 sec (DB 2.8 sec) <sup>7</sup>	124
Cost allocation – initial report	45 sec	5 sec (DB 3.4 sec) <sup>7</sup>	9
Cost allocation – drilldown by sending cost center	260 sec	7 sec (DB 3.3 sec) <sup>7</sup>	37

Abbildung 7: Optimierung der Verarbeitungszeiten im SAP CO-PA durch Hana [Si11, S.15]

### 3.2 Technologische Perspektive

Aus einer technischen Perspektive heraus stellt erstens die Weiterentwicklung der Produkte und zweitens der Betrieb der Systeme Herausforderungen an die Verarbeitungsgeschwindigkeit.

#### 1. Beispiel Releasewechsel

Für Unternehmen, die in erheblichem Umfang eigene Produkte als Standardreleases einsetzen und dadurch mindestens jährliche Releasewechsel in vielen Organisationseinheiten durchführen müssen, werden die verfügbaren Zeitfenster für die Umsetzung der Releasewechsel in den Organisationseinheiten sehr eng. Bei Unternehmen wie Nestle, Unilever, Beiersdorf, EDEKA, Metro etc. sind stets in vielen Einheiten Releasewechsel durchzuführen. Es ist eine große Herausforderung in den Einheiten, die notwendigen Zeitfenster für die Releasewechsel zu finden, die mit zunehmendem Datenvolumen auch nicht weniger anspruchsvoll werden. Es wäre eine signifikante Verbesserung der Situation durch den Einsatz von In-Memory-Technologien zu erwarten, wenn es auch gelänge, die Aufgabe auf den technologischen Schichten – und nicht nur bei den Applikationsschichten – zu unterstützen.

#### 2. Beispiel Batchprozesse in der Nachtverarbeitung

Auch wenn die Literatur weitgehend von Online-Anwendungen ausgeht, so haben in der Praxis Batchprozesse weiterhin eine große Relevanz. Die Vielzahl von Batchprozessen bei großen Unternehmen ist in der gebotenen Zeit schwer darzustellen, wie die Buchung aller Wareneingänge des Tages, die Verbuchung der Warenausgänge / Fakturen eines Tages, die Automatischen-Dispositions-Läufe, etc. Angesichts der Situation in Handelsunternehmen, dass die Öffnungszeiten immer weiter ausgedehnt werden, sind Betriebs-

zeiten von 7x24 h praxisrelevante Herausforderungen für zentrale IS-Architekturen. Hier könnten In-Memory-Ansätze eine Entlastung bei dem Betrieb der Systeme mit sich bringen, die bis dato noch nicht diskutiert werden.

Außerdem sind die Performance-Anforderungen jeden Tag in engen Zeitintervallen zu erfüllen, in denen beispielsweise die Bestellungen bei Apotheken- Großhändlern zu 80 % in der Zeit von 10.00-13.00 eingehen und verarbeitet werden müssen oder die Kassendaten in Spitzenzeiten wie freitags und sonnabends im Lebensmittelhandel für eine Echtzeit-Bereitstellung zu einer Herausforderung werden.

### 3. Beispiel: Deltaload-Prozesse

Auch bei den Massendatenänderungen, die im Handel täglich erfolgen, vor allem auch durch das Aktionsgeschäft, lassen sich enorme Verbesserungen in der Änderungsversorgung erwarten, wie es aus Abbildung 8 hervorgeht.

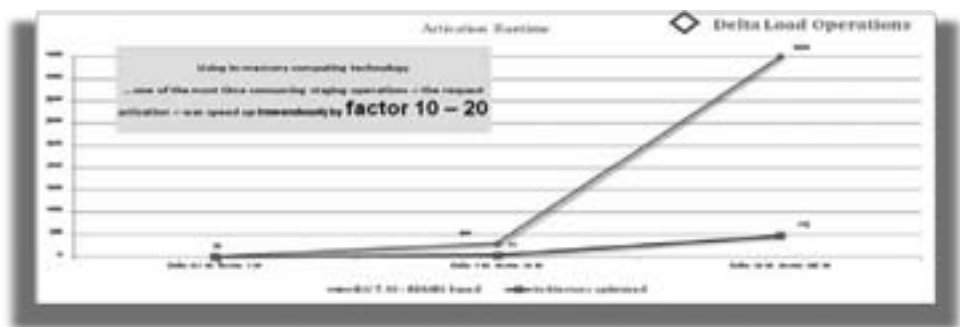


Abbildung 8: Verbesserung der Verarbeitungszeiten bei Delta-Load-Vorgängen [Si11, S. 27]

## 4 Ausblick

Die Einsatzfähigkeit der In-Memory-Technologien in Handelsunternehmen dürfte aufgrund der bereits pilotierten Projekte gegeben sein, ob damit auch die Einsetzbarkeit in derart großen Technologieumgebungen wie der EDEKA-Gruppe gegeben ist, bleibt allerdings noch offen.

Die betriebswirtschaftlichen Anforderungen, die aufgrund der Verfügbarkeit von In-Memory-Technologien umsetzbar werden, nehmen zu und hier bedarf es eines Analyse-rasters für die Ermittlung des Einsatzpotentials von In-Memory-Ansätzen.

Welche Aufgaben sich technisch besser lösen lassen, ist bislang im Kontext des neuen Technologiekonzepts wenig diskutiert worden. Dabei ist möglicherweise das technische Einsatzpotential für In-Memory-Technologien am erfolgversprechendsten, denn dort existieren bis dato weitgehend ungelöste Probleme, die nur mit In-Memory-Technologien einer Lösung zugeführt werden können.

Die größte Herausforderung bei den In-Memory-Technologien besteht darin, die konzeptionellen Anforderungen im Rahmen des Software-Engineerings nicht zu vernachlässigen [vgl. auch Le11]. Über Jahrzehnte hat die Informatik und die Wirtschaftsinformatik die fachkonzeptionelle Datenmodellierung – und seit den neunziger Jahren auch die Prozessmodellierung – vorangetrieben und es ist nun erforderlich, die neuen Technologieansätze mit den konzeptionellen Errungenschaften der Vergangenheit zu kombinieren.

## Literaturverzeichnis

- [BS04] Becker, J.; Schütte, R.: Handelsinformationssysteme. 2. Aufl., Redline-Verlag, Landsberg/Lech, 2004.
- [FG11] Fabian, B.; Günther, O.: In-Memory-Datenmanagement für Business Intelligence. In: Wirtschaftsinformatik 53 (2011) 6, S. 388.
- [Le11] Lehner, W.: In-Memory Data Management – Evolution oder Revolution. In: Wirtschaftsinformatik 53 (2011) 6, S. 386-387.
- [LV11] Lechtenbörger, J.; Vossen, G.: Hauptspeicherdatenbanktechnologie: Herausforderungen und weitere Entwicklung. In: Wirtschaftsinformatik 53 (2011) 6, S. 384-385.
- [Lo11] Loos, P.: In-Memory-Datenmanagement in betrieblichen Anwendungssystemen. In: Wirtschaftsinformatik 53 (2011) 6, S. 383-384.
- [PI09] Plattner, H.: A common database approach to OLTP and OLAP using an in-memory column database. In: Cetentemel, U.; Zdonik, SB; Kossmann, D.; Tatbul, N.: Proceedings of the ACM SIGMOD international conference on management of data (SIGMOD 2009), Providence 2009, S. 1-7.
- [PZ11] Plattner, H.; Zeier, A. (Hrsg.): In-Memory Data Management. An Inflection Point for Enterprise Applications. Berlin et al. 2011.
- [Re11] Reulmann, O.: In-Memory Evolution bei SAP. SAP Hana. Präsentation Februar 2011.
- [Si11] Sigg, S.: SAP HANA and Business Applications. Präsentation Oktober 2011.
- [ZKM11] Zeier, A.; Krüger, J.; Müller, J.: Potenzial von In-Memory-Technologien. In: Wirtschaftsinformatik 53 (2011) 6, S. 385-386.



# BI-Strategie zum Ausgleich von Technologie-Push und Business-Pull

Peter Chamoni

Lehrstuhl für Wirtschaftsinformatik, insbes. Business Intelligence  
Universität Duisburg-Essen  
Campus Duisburg  
Lotharstraße 63  
D-47057 Duisburg  
peter.chamoni@uni-due.de

**Abstract** Der vorliegende Beitrag gibt einen kurzen historischen Abriss zur Entwicklung der Business Intelligence (BI) und bezieht sich insbesondere auf das Spannungsfeld zwischen den technologischen Entwicklungen und den betriebswirtschaftlichen Anforderungen beim Aufbau von analytischen Informationssystemen. Business Intelligence stellt sich der Herausforderung, fragmentierte interne und externe Unternehmensinformationen zu einer vertrauenswürdigen Quelle der Entscheidungsfindung zusammenzuführen. Um eine adäquate Entscheidungsunterstützung zu etablieren, müssen sowohl die Prozesse und Inhalte der Unternehmensstrategie antizipiert, als auch die aktuellen Technologien der Informationsbeschaffung, Informationsspeicherung und Informationsaufbereitung implementiert werden. Zur Steuerung der BI-Prozesse werden zunehmend BI Competence Center eingerichtet, die als Organisationseinheit den Ausgleich zwischen dem wachsenden Pull der Fachabteilungen und den technologischen Angeboten (Push) der IT herstellen. Im Rahmen des Strategic Alignment werden die Ziele der BI-Strategie mit der Unternehmensstrategie abgeglichen, um effektive und effiziente BI-Systeme zu implementieren. Neben anderen aktuellen Themen der BI wird insbesondere die In-Memory-Technologie als neuer Impuls zur Wertsteigerung von BI-Lösungen diskutiert.

## 1 Einleitung

Seit dem Aufkommen der Managementunterstützungssysteme (MUS) standen die Anforderungen des Managements in Bezug auf die Planungs- und Entscheidungsprozesse sowie die unterschiedlichen Rollenkonzepte [Mi73] im Vordergrund. Aus den von SIMON postulierten Prozessphasen *Intelligence*, *Design* und *Choice* für eine Problemlösung sind direkte Anforderungsprofile für Unterstützungssysteme ableitbar [Si69]. In der Intelligence-Phase müssen Signale aufgenommen werden, um eine Statusanalyse durchzuführen. Dies ist der klassische Ansatz der BI-Systeme, die der Aufklärung und der Ursachenanalyse dienen [KBM10]. Die zweite Phase (Design) wird genutzt, um Lösungsalternativen zu generieren und anschließend, nach einer Bewertung, die beste Lösung zu wählen. Dies ist die Choice-Phase, der ein logisches Kalkül mit Restriktionen

und Zielfunktionen zugeordnet ist. Sinnvollerweise wird hierbei ein Entscheidungsmodell genutzt, welches die Basis für Decision-Support-Systeme (DSS) ist.

Damit lässt sich auch die historische Entwicklung von Managementinformationssystemen, DSS und Führungsinformationssystemen erklären, die abwechselnd den Fokus auf datengetriebene oder modellgetriebene Ansätze legten. Derzeit scheinen die datengetriebenen BI-Systeme wiederum von modellorientierten Systemen wie Predictive Analytics [SK10] unter Druck zu geraten. In regelmäßigem Turnus scheinen die Unterstützungspotenziale so von Phase zu Phase zu wechseln. Das Optimum liegt in der Koexistenz der unterschiedlichen Formen, so dass ein direkter Übergang von der frühen und zeitgerechten Analyse, über die Entscheidungsmodellierung bis zur Entscheidungsfällung und -implementierung, gewährleistet werden kann und damit alle Phasen Berücksichtigung finden.

Als Referenzarchitektur kann folgender Schichtenaufbau dienen:

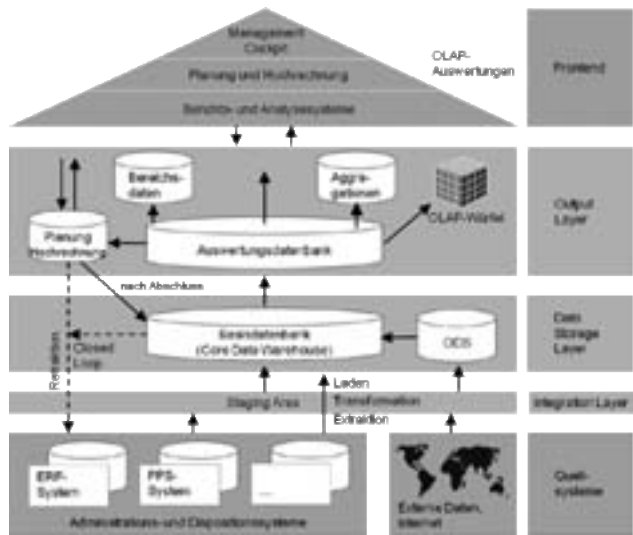


Abbildung 1: BI-Architektur

Der klassische Ansatz sieht vor, dass ein mehrstufiges System von den Quellsystemen bis zur analytischen Anwendungsschicht aufgebaut wird. Die hier skizzierte Architektur verfügt über fünf Layer, wobei die Integrationsschicht (ETL), die Basisschicht (DWH) und Auswertungsschicht (OLAP) zu den zentralen Funktionsbausteinen gehören [BG04]. Die unteren Schichten sind technologieorientiert, wohingegen die oberen Schichten der fachlichen Auswertung und betriebswirtschaftlichen Planung und Kontrolle zuzuordnen sind. Das Spannungsfeld zwischen der Informationsbereitstellung aus den Vorsystemen und der Informationsnutzung in der Auswertungsebene wird standardmä-

big durch die Pufferung der entscheidungsrelevanten Daten im Data Warehouse (DWH) ausgeglichen. Hierzu wird das Metadatenmanagement eingesetzt, welches die semantische Integration des Upstream-Datenflusses gewährleistet. Die hohe Komplexität der Datenbewirtschaftung im ETL-Prozess und die redundante Datenhaltung, welche aufgrund der Entkopplung der operativen und der analytischen Systeme notwendig erschien, führen vermehrt zur Kritik. Diese Kritik ist insoweit berechtigt, als die Rahmenbedingung bei der Einführung der Data-Warehouse-Konzepte [In05][Ki96] durch Restriktionen in Hardware und Software geprägt waren. Dies bedeutet, dass die Defizite der Informationsbereitstellung für das Management aus den operativen Systemen durch zusätzliche Verarbeitungsprozesse und Speicherstrukturen abgebaut wurden. Neue Technologien (Serviceorientierte Architektur [Di08], [GGH06]; In-Memory Data Base [Pi09], [KN11]) erwecken den Anschein, dass die Mehrschichtigkeit abgebaut und die Komplexität reduziert werden kann. Offen bleibt jedoch die Frage, ob der derzeitige Technologie-Push dem Business-Pull standhält, wenn ein Paradigmenwechsel im Data Warehousing durchgeführt wird.

## 2 Von der Strategie zum BICC

Aus Sicht des Managements sind BI-Systeme strategische Instrumente zur Unternehmensführung und haben sich dort besonders etablieren können, wo es um Fragestellungen des Corporate Performance Management geht. Hierbei handelt es sich methodisch beispielhaft um Balanced Scorecards, die den strategischen Management-Zyklus von der SWOT-Analyse bis zur Operationalisierung begleiten und aus den Performance-Kennzahlen der Umsetzung die strategische Rückkopplung ableiten. Somit können in diesem Zusammenhang Planungs- und Budgetierungsprozesse durch BI-Systeme teilautomatisiert werden, so dass eine zielgerichtete Umsetzung von Unternehmensstrategien unterstützt wird. Empirische Befunde [BMB09] belegen, dass 77% der Geschäftsprozesse aufgrund des Anpassungsdrucks unterjährig dynamisch angepasst werden müssen. Die Stabilität solcher Prozesse kann nur noch für drei Monate angenommen werden. Unter diesem Veränderungsdruck leiden nicht nur die ERP-Systeme, sondern auch die BI-Systeme, die zunehmend zur Steuerung und Adaption der Geschäftsprozesse eingesetzt werden. Erfolgversprechend scheinen die Ansätze zu sein, welche die Agilität [GH11] von BI-Systemen erhöhen und die Anpassung aus Sicht der Geschäftslogik fördern.

Zur gesamtheitlichen Konzeption von Business Intelligence in Unternehmen ist eine multiperspektivische Betrachtung unumgänglich. Business Intelligence wird durch *technologische, organisatorische* und *umweltbezogene* Faktoren beeinflusst. Die organisatorischen Rahmenbedingungen beziehen sich beispielhaft auf die verfügbaren Ressourcen, die Unternehmensgröße, den Grad der Zentralisation und Formalisierung der Abläufe. Ökonomische Umweltfaktoren lassen sich an den Branchenspezifika, den Wettbewerbsbedingungen und den regulatorischen Anforderungen festmachen. Folgt man dem Modell von TORNATZKY und FLEISCHER [TF90], so lassen sich die wichtigsten Einflussgrößen in den genannten Dimensionen auch für BI-Systeme beschreiben. Hierdurch entsteht ein BI-bezogenes Rahmenwerk, das die Gestaltungsfaktoren und den Reifegrad von analytischen Informationssystemen beschreibt.



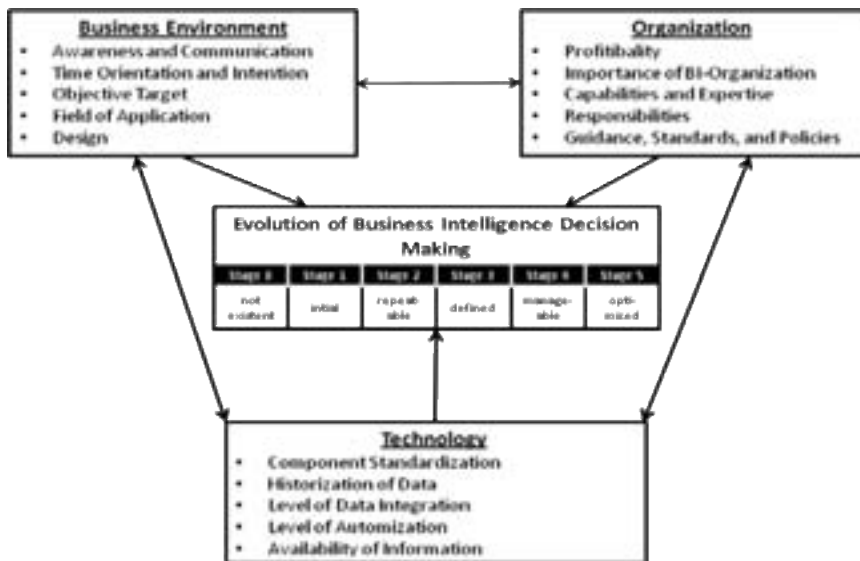


Abbildung 2: Techology-Organization-Business Environment Framework

Die Evolution der BI-Systeme vom einfachen Berichtswesen bis zur optimierten Entscheidungsunterstützung wird durch eine Vielzahl von Reifegradmodellen [BKP08], [CG04] beschrieben. Diese können als Kern des Frameworks angesehen werden, denn sie nehmen die Bedingungs-lage des jeweiligen Unternehmens durch die Einflussfaktoren auf und führen zum optimierten Design des BI-Systems.

Zur Sicherstellung dieses Anpassungsprozesses werden vermehrt organisatorische Einheiten gebildet, die als Business Intelligence Competency Center (BICC) bezeichnet werden [GTS10]. Das Aufgabenspektrum reicht vom BI-Management mit Standardisierungsvorgaben über die Ausgestaltung der fachlichen und technischen Architektur bis zur BI-Unterstützung (Personalentwicklung und Support) und zur BI-Umsetzung (Entwicklung und Betrieb).

In dieser BI-Organisation finden sich die verantwortlichen Projektmanager, Entwickler und Applikationsmanager, die einem Lenkungsausschuss unterstehen und die Schnittstellen zu den internen Kunden, dem IT-Bereich und den externen Dienstleistern koordinieren. Die Einbindung aller Anspruchsgruppen gewährleistet die Anpassung der BI-Systeme an die Unternehmensstrategie, ohne den technologischen Aspekt bei der Festlegung der passenden Architektur zu vernachlässigen.

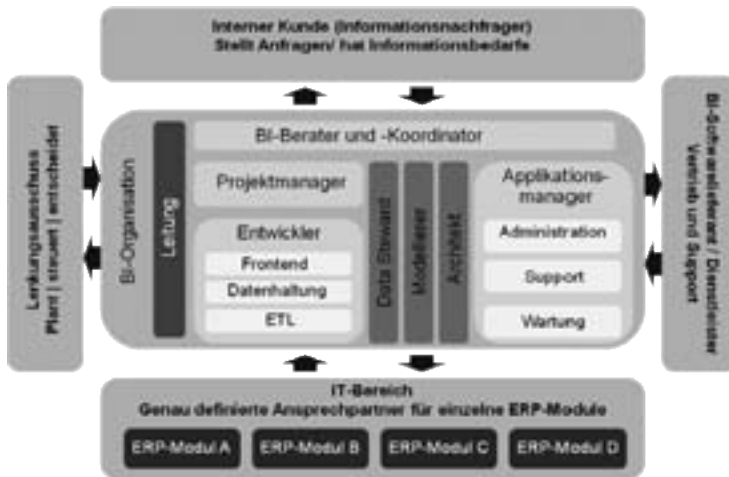


Abbildung 3: BI Competency Center

### 3 Aktuelle Trends

Den Status-quo des Einsatzes von BI-Systemen in Unternehmen zeigt die BARC-Studie 2011 [Ba11]. Die wichtigsten Ziele, die laut Studie im Vertrauen in die Daten (82%) und in der Kundenzufriedenheit (26%) liegen, werden nur unbefriedigend umgesetzt, denn nur 51% der befragten artikulieren ihr Vertrauen in die Informationen aus einem Data Warehouse. Somit ist nicht erstaunlich, dass die maßgeblichen Herausforderungen einerseits für die IT derzeit in der Datenqualität (55%) und der Performance (54%) liegen, andererseits aus der Business-Sicht die Schnelligkeit der Systemanpassung (63%) und die Datenqualität (51%) dominieren. Insgesamt ein ernüchterndes Bild, so dass neue Konzepte und Techniken auf diese Defizite zielen müssen. Die folgenden Ausführungen skizzieren die aktuellen Trends zu Big Data Analytics und Datenintegration.

Die Analyse großer Datenmengen (Big Data Analytics) hat sich zu einer der größten Herausforderungen und Trends der Business Intelligence entwickelt. Big Data Analytics definiert sich dabei nicht nur über die Größe der gespeicherten und zu analysierenden Datenmenge (*Volume*), sondern auch die Geschwindigkeit (*Velocity*), mit der Daten zur Analyse bereitgestellt werden. Eine real-time-Bereitstellung von Analysedaten ist eine ebenso große Herausforderung wie die technische Bewältigung der großen Datenmengen. Darüber hinaus spielt die Vielfalt (*Variety*) der anfallenden Daten eine entscheidende Rolle. Durch eine Vielzahl unterschiedlicher Datenquellen können strukturierte Daten ebenso zur Analyse herangezogen werden wie unstrukturierte und semistrukturierte Daten, was wiederum neue Herausforderungen bei der Verarbeitung mitbringt. Diese so genannten *Three Vs of Big Data* und deren Zusammenspiel bilden den Kern der begrifflichen Definition von Big Data Analytics.



Abbildung 4: The three Vs of big data (Quelle: [Ru11b])

Der TDWI hat in einer Studie zwei zentrale Unterscheidungskriterien zur Einordnung des Umgangs mit Big Data Analytics festgestellt: Das organisationale Commitment und die potenziellen Wachstumsmöglichkeiten. Das organisationale Commitment verdeutlicht, welche Werkzeuge und Methoden von Unternehmen heute bereits genutzt werden und/oder in Zukunft genutzt werden sollen. Die potenziellen Wachstumsmöglichkeiten werden abgeleitet aus der Diskrepanz zwischen den heute eingesetzten Werkzeugen und Methoden und denen, die nach Einschätzung der befragten Teilnehmer in drei Jahren zum Einsatz kommen werden. Durch die grafische Verbindung dieser beiden Kriterien lassen sich vier zentrale Gruppen von Werkzeugen und Methoden identifizieren:

1. Mäßiges bis starkes Commitment, hohe potenzielle Wachstumsmöglichkeiten
2. Mäßiges Commitment, gute potenzielle Wachstumsmöglichkeiten
3. Schwaches Commitment, gute potenzielle Wachstumsmöglichkeiten
4. Starkes Commitment, geringes oder rückläufiges Wachstum.

Wie in Abbildung 5 gezeigt, wird das höchste potenzielle Wachstum in den nächsten drei Jahren für *Advanced data visualization* prognostiziert, wofür sich bei der Befragung ein Wachstum von 27% ergab. Weitere wichtige Werkzeuge bezüglich des potenziellen Wachstums sind *Predictive analytics*, *Real-time dashboards*, *Text mining*, *In-memory databases* und *Visual discovery*.

Eine weitere große Herausforderung ist die Datenintegration der nächsten Generation (Next Generation Data Integration). Durch die wachsende Anzahl unterschiedlicher Datenquellen, das steigende Volumen und die vermehrte Verarbeitung unstrukturierter Daten nimmt die Komplexität der Datenintegration immer mehr zu.

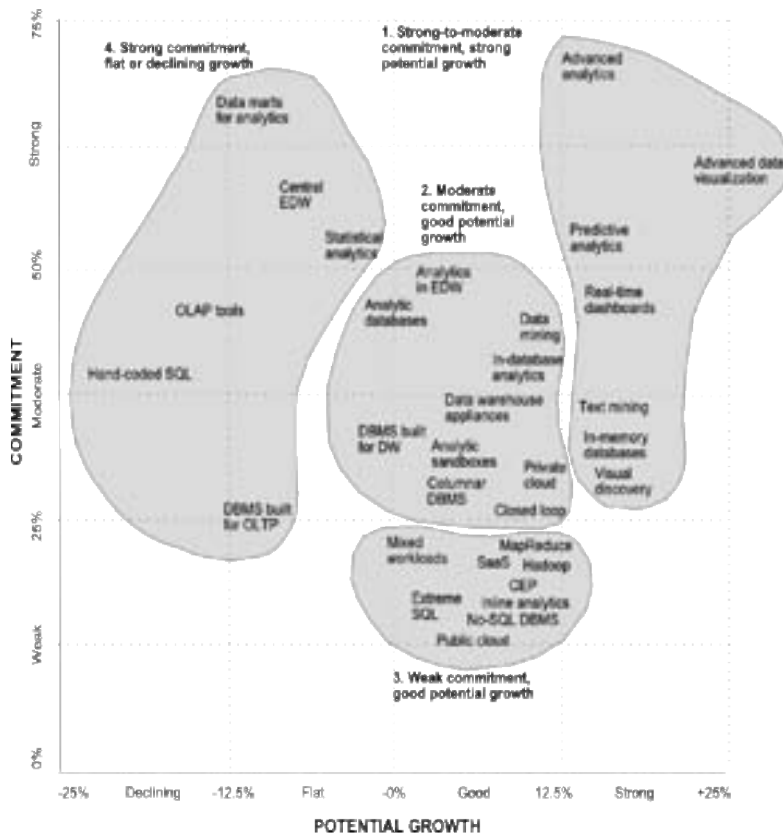


Abbildung 5: Options for Big Data Analytics Plotted by Potential Growth and Commitment (Quelle: [Ru11b])

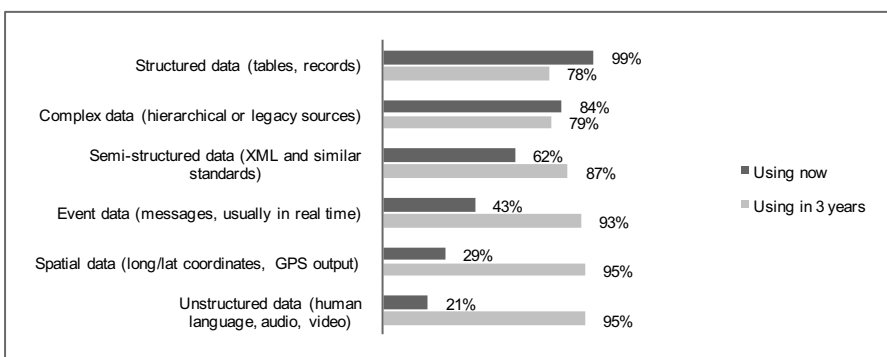


Abbildung 6: Data Types Being Integrated (Quelle: [Ru11a])

Aus Abbildung 6 wird ersichtlich, dass in den nächsten Jahren eine deutliche Verschiebung der vorhandenen Datentypen von überwiegend strukturierten und semistrukturierten Daten zu unstrukturierten Daten wie Audio- und Videomaterial zu erwarten ist. Unternehmen müssen auf diese Veränderung durch die Anpassung ihrer technischen und organisatorischen Rahmenbedingungen vorbereitet sein.

Analog zur Darstellung in Abbildung 5 wurde in einer weiteren Studie des TDWI eine Einordnung der aktuellen und zukünftigen Situation der Datenintegration nach dem organisationalen Commitment und dem potenziellen Wachstum vorgenommen. Abbildung 7 zeigt, dass sich aus der Kombination dieser beiden Kriterien fünf Gruppen möglicher Optionen ergeben:

1. Mäßiges bis starkes Commitment, hohe potenzielle Wachstumsmöglichkeiten
2. Gutes Commitment, gute potenzielle Wachstumsmöglichkeiten
3. Mäßiges Commitment, gute potenzielle Wachstumsmöglichkeiten
4. Schwaches Commitment, gute potenzielle Wachstumsmöglichkeiten
5. Starkes Commitment, geringes oder rückläufiges Wachstum

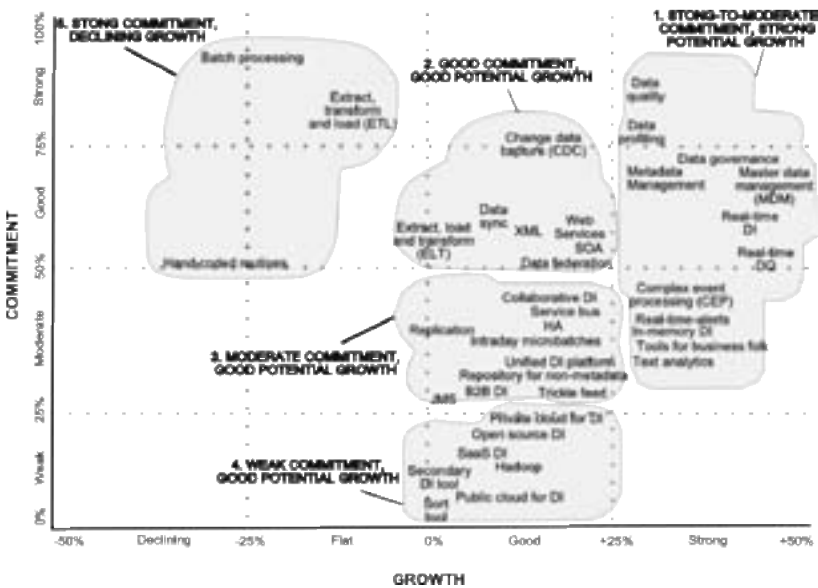


Abbildung 7: Next Generation Data Integration Plotted for Growth and Commitment (Quelle: [Ru11a])

Als wichtigste Gruppe bezüglich der Relevanz in den nächsten Jahren ist die erste Gruppe zu identifizieren. Diese ist eine Kombination aus wachsenden Real-time-Techniken, Aspekten des Datenmanagements und organisationalen Vorgehensweisen. Insbesondere *Real-time Data Integration* und *Real-time Data Quality* sowie *Complex Event Proces-*

sing werden nach der zitierten Studie als zentral und wegweisend in der BI-Entwicklung der kommenden Jahre gesehen.

## 4 Zusammenfassung

Es bleibt festzuhalten, dass die Komplexität von BI-Systemen tendenziell steigt und die gesetzten Ziele nur unzureichend erfüllt werden. Einerseits sollen diese Systeme den Anforderungen der Planungs- und Kontrollprozesse in Unternehmen genügen, andererseits auch den aktuellen Veränderungen in der Technologie Rechnung tragen. Hierzu wurden Modelle der strategischen Anpassung vorgestellt, die sicherstellen, dass alle erfolgskritischen Faktoren für die Entwicklung und den Betrieb von analytischen Informationssystemen beachtet werden. Neuere Entwicklungen wie die Beherrschung von Big Data und wachsende Integrationsansprüche führen zu Technologien, die schnelle Anpassbarkeit und hohe Performance versprechen. Die In-Memory-Speicherung ist eine dieser Schlüsseltechnologien, welche die Anforderungen aufnimmt und zu einem Paradigmenwechsel führen kann. Der Vorteil ist vornehmlich im Abbau der Architekturkomplexität und im Geschwindigkeitszuwachs zu sehen. Nicht geklärt ist die Fragestellung des Vertrauens in die Daten sowie die Historisierung der Informationen. Somit bleiben weitere Forschungsfragen offen, die noch zu bearbeiten sind. Im Sinne der Decision Analysis bleibt zu hoffen, dass die In-Memory-Technologie zur Erreichung der oberen Reifegradstufen von BI-Systemen hilft und zur substantiellen Entscheidungsunterstützung beiträgt.

## Literaturverzeichnis

- [Ba11] BARC Institut: Data Warehousing 2011 – Status quo, Herausforderungen und Nutzen. Eine unabhängige Anwenderbefragung. Würzburg, 2011.
- [BG04] Bauer, A.; Günzel, H.: Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung. Dpunkt.Verlag GmbH, Heidelberg, 2004.
- [BKP08] Becker, J.; Knackstedt, R.; Pöppelbuß, J.; Schwarze, L.: Das IT Performance Measurement Maturity Model - Ein Reifegradmodell für die Business Intelligence-Unterstützung des IT Managements, 2008.
- [BMB09] Bange, C.; Marr, B.; Bange, A.: Performance Management – Aktuelle Herausforderungen und Perspektiven. Eine weltweite Umfrage zum Reifegrad von Performance-Management-Prozessen. BARC Institut .Würzburg, 2011.
- [CG04] Chamoni, P.; Gluchowski, P.: Integrationstrends bei Business-Intelligence-Systemen. Wirtschaftsinformatik, 46 (2), 2004; S. 119-128.
- [Di08] Dinter, B.: Einsatzmöglichkeiten serviceorientierter Architekturen in der Informationslogistik. In: Töpfer, J.; Winter, R. (Hrsg.): Active Enterprise Intelligence. Unternehmensweite Informationslogistik als Basis einer wertorientierten Unternehmenssteuerung. Springer, Berlin, 2008; S. 221-241.
- [GGH06] Gordon, S.; Grigg, R.; Horne, M.; Thurman, S.: Service-Oriented Business Intelligence, in: The Architecture Journal, 2006.  
<http://msdn.microsoft.com/en-us/library/bb245659.aspx>
- [GH11] Göhl, R.; Hahne, M.: Agile Konzepte für BI. Bessere Architektur, Organisation und Methodik. In: BI Spektrum (2), 2011; S. 10-15.

- [GTS10] Gansor, T.; Totok, A.; Stock, S.: Von der Strategie zum Business Intelligence Competency Center (BICC), Hanser München, 2010.
- [In05] Inmon, W.H.; Building the data warehouse , Wiley, Indianapolis, 2005.
- [KBM10] Kemper, H.-G.; Baars, H.; Mehanna, W.: Business Intelligence - Grundlagen und praktische Anwendungen. 3. Aufl. Vieweg + Teubner, Wiesbaden, 2010.
- [Ki96] Kimball, R.: The data webhouse toolkit. Practical techniques for building dimensional data warehouses. Wiley, New York [u.a.], 1996.
- [KN11] Kemper, A.; Neumann, T.: HyPer: A hybrid OLTP&OLAP main memory database system based on virtual memory snapshots. IEEE 27th International Conference on Data Engineering (ICDE), 2011.  
<http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5767867>
- [Mi73] Minzberg, H.: The nature of managerial work, New York [u. a.], 1973.
- [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.
- [Ru11a] Russom, P.: Next Generation Data Integration, TDWI Best Practices Report, Second Quarter 2011.
- [Ru11b] Russom, P.: Big Data Analytics, TDWI Best Practices Report, Fourth Quarter 2011.
- [Si60] Simon, H.: The New Science of Management, New York, 1960.
- [SK10] Shmueli, G.; Koppius, O.: Predictive Analytics in Information Systems Research. Robert H. Smith School Research Paper No. RHS 06-138, 2010.
- [TF90] Tornatzky, L. G.; Fleischer, M.: The Processes of Technological Innovation. Lexington Books, Lexington, Massachusetts, 1990.

# Revolution or Evolution? Reflections on In-Memory Appliances from an Enterprise Information Logistics Perspective

Robert Winter, Stefan Bischoff, Felix Wortmann

Institute of Information Management

Mueller-Friedberg-Strasse 8

CH-9000 St. Gallen

{robert.winter; stefan.bischoff; felix.wortmann}@unisg.ch

**Abstract:** While the conceptual architecture of enterprise information logistics has been stable since the late 1980ies, IT hardware has been subject to radical change recently. Since in-memory appliances as a new technology might address many challenges of information logistics, we discuss its potentials first. Based on a case study of a global automotive company, we then compare potentials with actual achievements. We conclude that there are situations where in-memory appliances are a useful extension to existing IT support concepts, while other situations do not require such support. As a consequence, we regard in-memory appliances as an evolution, but not revolution of IT support from an enterprise information logistics perspective.

## 1 Introduction

Starting from a rather technical (tool) perspective in the mid 1990ies, the understanding of Business Intelligence (BI) information systems has widened and now covers “a broad category of technologies, applications, and processes for gathering, storing, accessing, and analyzing data to help its users make better decisions.” [WW10] Since BI is focused on decision-making, it usually has a ‘local’ perspective, i.e. its design is focused on a specific business domain, a specific business problem, a specific business process, or a specific organizational role. In order to cover the overall perspective of a business unit, company or even business network on data gathering, data integration, data storage and data analysis, the concept of Enterprise Information Logistics (EIL) has been proposed [Wi08]. Since the understanding of BI has developed from a technical to an information system (IS) perspective, BI and EIL do not differ in this respect. Their difference is that EIL has an enterprise-wide data provisioning perspective, while BI focuses on the ‘local’ use of provisioned data.

Rapidly growing amounts of data and more complex analyses are only two of the current EIL challenges. In times of rapid change and volatile economy, decision makers rely on timely information [Sc10]. Different approaches exist for reducing data entry latency, data analysis latency and decision latency. While some improvements focus on the manual parts of data provisioning and decision making, new technology in the form of in-memory appliances promises to reduce time for data analysis significantly [PI09].



IT vendors claim that in-memory is the end of classical EIL [E111]. They argue on the basis of radical hardware changes and foresee a new era of business analytics. While there is certainly evidence that radical hardware innovation has an impact on EIL, we want to challenge this statement from two lenses. First, hardware innovation has to be put into the context of people and tasks – in order to create impact, changes of one IS component type need to be matched by changes of the others to create improvements. Second, an enterprise wide focus might put specific hardware innovations into perspective, even if they are fundamental for a certain business domain, business problem, or business process. Both lenses are combined in a case study from a large automotive company that recently introduced in-memory appliances for business analytics.

The paper is structured into five sections. Section 2 discusses the state of the art of EIL, EIL challenges, fundamentals of in-memory appliances and the potentials of in-memory appliances in relation to EIL architecture challenges. Section 3 introduces the case study and compares the potentials of in-memory appliances to actual achievements. These findings are discussed in section 4. A summary and outlook concludes the paper in section 5.

## **2 Technology challenges of EIL and potentials of in-memory appliances**

### **2.1 Conceptual EIL architecture**

EIL IS are based on a data warehousing infrastructure that collects, stores, and integrates relevant data that is further modified and enhanced within specific reporting processes. The resulting conceptual (hub and spoke) EIL architecture usually is comprised of five layers [BG09; DM88; PR03]:

- Source systems: These software systems contain operational data and other data that is important for the reporting and analysis process. Usually internal and external information systems are used as sources [In05].
- Data transformation (ETL - extract, transform, load processes): In order to move data from different heterogeneous sources into one central data warehouse, different transformation steps are required in order to normalize, harmonize and integrate data.
- Data warehouse (DWH): The DWH is a logically centralized database, which consolidates all data that could be useful for analytical applications as well as decisions and management tasks [In05; KR02].
- Data Marts (DM): Based on the DWH, data marts are copies of data that relate to a specific domain. For example, a purchasing DM contains only relevant data for purchasing analyses. DMs usually store data in a form that supports interactive data analyses [Ba09; KBM10].
- Reporting systems: These software systems access DMs (or sometimes the DWH directly) in order to present data to users. There are different reporting approaches like standard reporting [GGD08], ad-hoc reporting [GK06] or advanced, interactive reporting like OLAP (online analytical processing) [CC93].

Recent BI innovations combine technical and organizational improvements regarding overall responsiveness of BI (e.g. real-time BI [Di09] or active BI [Di07]). Although new trends have emerged, looking back at the first data warehouse publications (e.g. [DM88]) one has to conclude that the fundamental conceptual EIL architecture has been stable for more than 20 years.

## 2.2 EIL challenges

While the stability of conceptual EIL architecture indicates a high level of maturity, the depicted conceptual EIL architecture poses three key challenges [Wi11]:

- **Speed:** Today, data processing and enrichment (extraction, transformation, preparation, integration, provisioning and analysis) cannot be performed in real-time in larger organizations, even when using powerful organizational concepts and powerful technology. Underlying inhibitors are rising data volumes, increasing needs for data integration and ever more diverse data analysis possibilities. For efficiency reasons, analyses have to be optimized carefully and data updates are often organized as batches. Iterative, interactive analytics of live data, as required by many decision makers, are therefore only inadequately supported [Fi10].
- **Integration:** The many integration and processing steps that are caused by specialized tools and infrastructures, lead to high complexity as well as high operating and development costs [Ec09]. In order to achieve substantial complexity encapsulation and cost reductions, data processing and enrichment functions have to be brought together in integrated IT systems. However, with existing technologies this seems to be possible only to a very limited extent, as numerous processing steps are indispensably simple because of performance reasons (e.g. pre-computed aggregates in DMs).
- **Flexibility:** Due to the need for optimizations in today's IT landscapes, only certain specific analysis functions and paths can be supported efficiently [Fi10]. However, in increasingly dynamic decision situations, decision makers ask more and more for solutions that enable them to analyze any desired business object from any perspective, e.g. on the basis of individual and spontaneous merge of data.

## 2.3 In-memory appliances – fundamentals and potentials

While conceptual EIL architecture has been stable for a long time, hardware is driven by radical change. During the last decades Moore's law [Mo65], which predicts the expansion rate of available processing speed, has been valid at all times. The processing performance has increased by more than the factor 4.000 [NP11] over the last 20 years. Even with the processor's clock speed having reached its peak, new concepts like massive parallel processing enable new improvements that help to increase processing speed even further [Pi09]. Additionally, the exponential price drop of processing power and main memory fosters breakthrough innovation. In-memory databases leverage these developments in order to optimize performance of analytical systems. Therefore, they

build upon massive parallel processing (MPP), in-memory technology and a new type of data storage [PI09].

In the environment of data processing, multi-core processors and MPP offer huge acceleration possibilities. In order to exploit this potential, software can use multiple processors in parallel to speed up calculations and data manipulation. In order to take full advantage of such capabilities, special programming techniques as well as processor specific coding need to be used. Thus programs need to be adapted to the corresponding processing platforms [PI09]. Compared to hard drives, main memory – that is used as the storage media in in-memory databases – has much faster access times (20-50 ns vs. 8-15 ms) [BU10]. Moreover, columnar databases have huge advantages in analytical environments because usual reports are rather set oriented than record oriented [PI09]. In order to calculate the overall number of sold products in a classical, record / row store database, every sales order has to be processed and the overall amount of accessed data is calculated as  $D1 = \text{number of records} * \text{size of all columns}$ . In contrast, in a column store database only the column with the corresponding quantity has to be read from the storage device. Thus the amount of accessed data is only  $D2 = \text{number of records} * \text{size of column "quantity"}$ .

In-memory appliances are understood as IT systems which combine software (i.e. in-memory databases) and hardware (i.e. in-memory technology). On the basis of the depicted innovations, they promise to address the above mentioned EIL challenges [PI09]:

- **Speed:** Data updates are propagated incrementally and in real-time into the analytical environment. The asynchronous actualization of persistent aggregates (pre-computed interim results) can be omitted. Thus, iterative, interactive analyses can be supported much better: Actual data is available at any point in time.
- **Integration:** Transactional and decision-related data is managed in an integrative manner. Data redundancy, which has to be actively managed, is eliminated and possible inconsistencies are thereby avoided. Processing stages can be reduced and the architectural complexity can be cut down.
- **Flexibility:** Analysis paths are not limited by pre-fabricated aggregates, so that data can be integrated and analyzed from any perspective. Thus an individual and spontaneous fusion of data is possible.

In order to assess whether these promises can be met, a case study is presented in the next section.

### 3 An in-memory appliance at an international automotive company

#### 3.1 Company characteristics and current situation

Our case study is based on the analysis of a large Germany-based, globally present automotive company. The company can be classified as an early adaptor of in-memory

appliances. On both, the organizational and the technical level the corporation prepared itself for a “first class BI environment.” On an organizational level they established an internal SAP competence center that is responsible for any operational and analytical SAP-based system. In addition an BI department<sup>1</sup> has been created with company-wide responsibility that not only supports the operations of analytical systems, but also starts initiatives to optimize and improve the BI system. From a technical perspective, the company has clearly an innovate attitude: The BI department is always willing to implement new technology and acts regularly as a showcase or ramp-up partner in case a technology promises significant improvements.

The current EIL landscape of our case study is visualized in figure 1. At the moment the EIL landscape’s structure conforms to the traditional architecture introduced in section 2. There are various source systems on the basis of SAP and non-SAP applications that feed data into the analytical components. The company runs two data warehouses. While one is based on the SAP Business Information Warehouse (SAP BW) that is used for SAP-based source systems, another DWH (based on Oracle technology) integrates all other source systems. For different analytical purposes, various data marts (SAP BW-based) exist that contain business scenario specific extractions from the two main DWHs. For reporting and analysis purposes, different applications and tools are used in the various functional areas. Analytical tools like SAP BW Query, Xcelsius Dashboards, Crystal Reports access not only the DMs, but also other data sources.

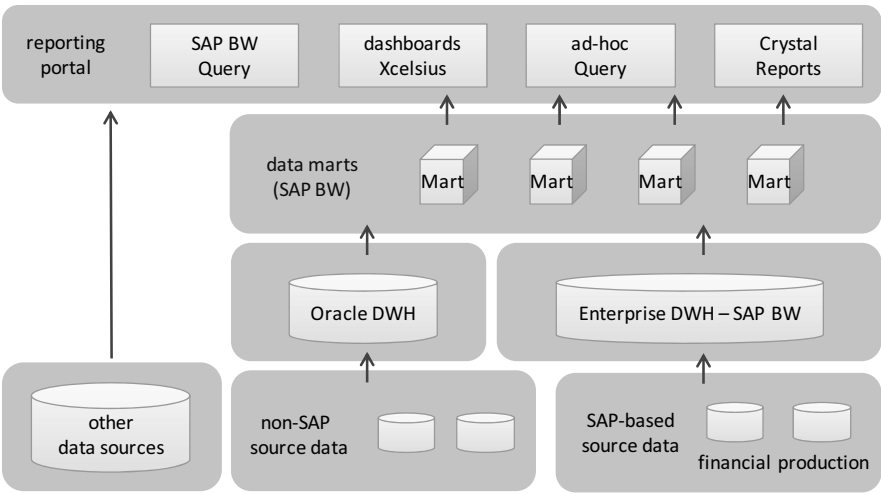


Figure 1: Basic structure of the current EIL landscape

In order to improve EIL and offer improved services for the business users, the company is in the process of implementing an in-memory appliance. In the following, we focus on the in-memory initiative as well as its business improvements.

<sup>1</sup> The BI department has an enterprise perspective. In the terminology of this paper, we would designate it as EIL department.

Within our analysis we were faced with two major usage scenarios that are currently about to be implemented. On the one hand, the in-memory appliance will be used for improving operational analytics. On the other hand, it will be used to increase the overall performance of the DWH landscape. According to our IS lens, we differentiate between the business perspective (task) and the technological solution for these problems (technology) for each of the usage scenarios.

### 3.2 Operational analytics

The first usage scenario for the in-memory appliance is the support of different business functions in the production, purchasing and financials areas.

Today, transactional ERP systems have significant performance problems with specific tasks like *quality and fault analysis* or *calculation of procurement commissions*. In order to be effective, both analyses are rather complex and require a full bill of material explosion. Current ERP systems are not capable of providing this information within an appropriate time period. Furthermore, due to long calculation times only static decision making is possible. Iterative recalculation of reports to explore different scenarios and alternatives is restricted by query run times. Existing BI systems could be used to accelerate query times. However, in order to leverage these BI systems, data has to be extracted from the underlying ERP systems and stored redundantly in the BI environment. The resulting solution would have such a complexity and incur so much higher costs, that business is reluctant to realize this scenario.

In the *financial environment* the monthly closing of accounts is a complex task that is not supported efficiently by today's EIL landscape. In this case relevant account data needs to be analyzed and the results of the analysis need to be posted back (after performing manual checks and adjustments) into the source systems (closed-loop approach). Today, the complete process is performed in the ERP system. Long calculation times prevent a fast close of books and limit the quality of the overall process (static process without iterations). An easy integration of existing BI systems into this scenario is not possible, as the data has to be queried, manipulated and finally posted back into the ERP system. In order to accomplish such a fully integrated solution that perfectly supports the accountants' tasks, a new technological solution is needed. This solution needs to increase reporting speed and thereby support iterative decision making.

In order to solve the previously described problems and to enhance the operational decision processes, the in-memory appliance can be used for reducing the access time to operational data.

Figure 2 illustrates the technical solution for the first task (quality and fault analysis, calculation of procurement commissions). To improve decision making, the operational ERP database is replicated into an in-memory appliance, so that a new record is instantly available in both databases. Now, reports can be retrieved from the high performance in-memory appliance via BI reporting frontend tools. As a consequence, query times can be reduced significantly.

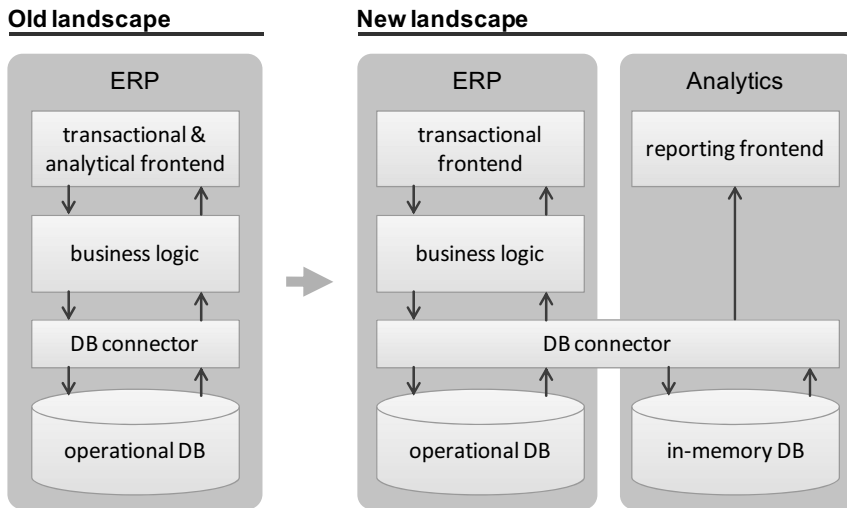


Figure 2: Enabling operational analytics on the basis of an in-memory appliance [on the basis of SA11]

Figure 3 illustrates the technical solution for the second task (financial environment). Again, the operational ERP database is replicated into an in-memory appliance. To enable closed-loop analytics, query results have to be available in the ERP system where they can be checked, modified and processed in a next business transaction. Thus the analyzed data needs to be transferred from the in-memory appliance back to the operational ERP system.

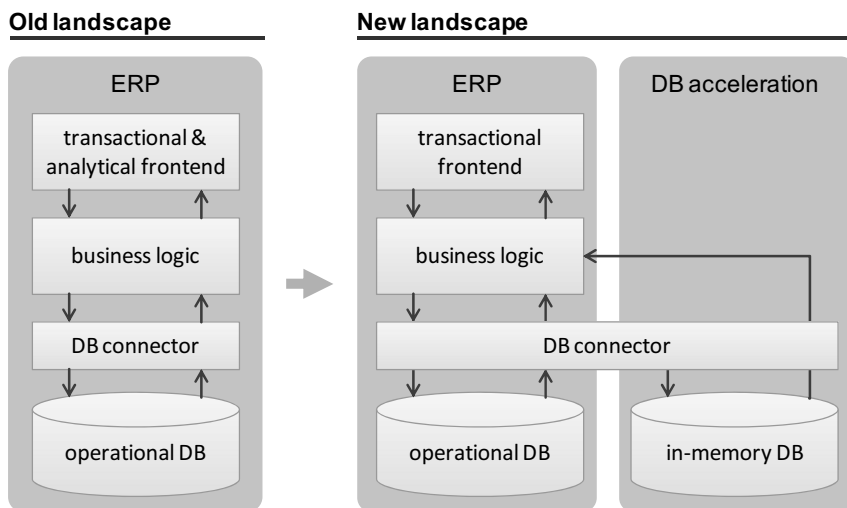


Figure 3: Enabling closed loop analytics on the basis of an in-memory appliance [on the basis of SA11]

### 3.3 Data warehousing

In addition to the mentioned weaknesses of current operational systems, there are also disadvantages that classical EIL architectures reveal. In analytical tasks the analysts and business users of the studied company demand fast and flexible analyses without restrictions of analyzable key performance indicators (KPIs) or pre-defined drill-down paths. Due to performance problems and slow response times for complex reports, the international automotive company used DMs and materialized cubes for analyses. These concepts extract a previously defined amount of data and KPIs from the data warehouse and pre-aggregate it into DMs or cubes that can be used for analyses. Thus the possible dimensions of reports are fixed, based on the ones that are copied into the DMs, only these are accessible at an acceptable speed. This configuration is not satisfying for business users because they are limited in the scope of their analysis. They demand more flexibility and better response times when it comes to analyzing data. In order to transfer data from the DWH into DMs and cubes, specific ETL processes are necessary. These processes have to be executed after the load process of the DWH has been completed successfully. In the complex environment of our case study partner, the runtime of both ETL processes (source systems to DWH, DWH to DMs) almost exceeds the amount of time that is available every night. Thus their aim is also to reduce the runtime of ETL processes.

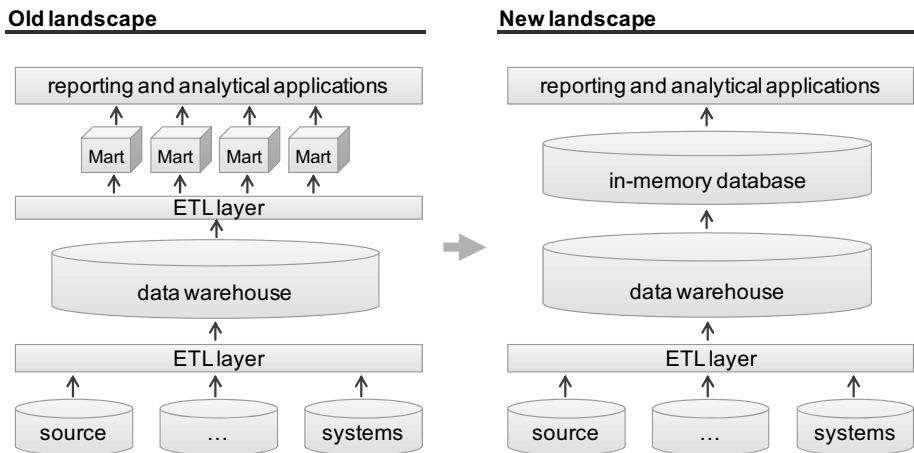


Figure 4: Enabling in-memory data warehousing [on the basis of SA11]

In order to fulfill these requirements, the case study company is about to implement an in-memory-based application that speeds up the DWH queries. Based on the fact that this solution offers much faster data access, there is no need for pre-aggregation and data extraction into DMs or other materialized components that limit reporting flexibility. In addition the elimination of DMs obviates the need for the second ETL level and reduced the overall ETL runtime. Figure 4 visualizes the architecture of that solution where an in-memory appliance replaces traditional DM technology. In this architecture the classical EIL stack (see section 1) is slightly modified, i.e. the DM layer is replaced by an in-

memory appliance. The whole DWH that is stored in a classical, hard drive-based database, is copied into an in-memory database that now represents the basis for analysis and reporting. The response time is now significantly reduced because data can be retrieved from the in-memory database much faster than it could be from a traditional DM.

In order to assess the impact of in-memory appliances from an EIL perspective, the following section discusses our findings.

## 4 Discussion

In our case study we discovered that in-memory appliances indeed can have a significant impact on the way people use IT systems to support their tasks. Static decision making can become dynamic, i.e. iterative analysis and lightweight simulation are facilitated by these technologies. The case study reveals two scenarios which should be differentiated:

- **Operational analytics** is about enhancing the performance of a single transactional system. Operational analytics is often performed in ERP systems, which are not able to process more sophisticated queries in a short period of time. Established BI solutions can be used to solve this problem. However, huge investments are necessary to bring that operational data into the existing BI/DWH environments. Therefore, in-memory appliances are perceived as a lightweight and easy to implement/maintain technology to increase operational query performance.
- **Data warehousing** is about integrating data from multiple sources in order to support decision making. While in-memory appliances can replace the performance function of DMs, in-memory appliances do not address the fundamental issue of integration. Therefore, in-memory databases complement the DWH and, by replacing DMs, slightly change the traditional hub-and-spoke architecture.

So how do in-memory appliances change the EIL landscape? On the basis of the case study two essential characteristics can be identified: To a large extent, the underlying IS landscape is determined by the need for integration (*required degree of integration*). If the performance of a single IS has to be improved, other approaches should be implemented than in the case of optimizing the performance in a domain where several IS are tightly integrated. The *data volume*, which has to be analyzed, is a major driver for the use of in-memory appliances. However, there are scenarios where data volumes are moderate so that no advanced, high performance technology is necessary. On the basis of these characteristics four stereotype patterns can be identified (see Fig. 5):

- **Low/moderate data volume, no need for integration:** This pattern is about operational analytics. The transactional system at hand has no performance issues. There is no significant need for an in-memory appliance.
- **Hugh data volume, no need for integration:** This pattern is also about operational analytics. High amounts of data can be handled by introducing an in-memory appliance (see section 3.2).



- **Low/moderate data volume, need for integration:** The need for integration drives the application of BI/DWH. There is no significant value added by an in-memory appliance.
- **Hugh data volume, no need for integration:** This pattern is also about BI/DWH. High amounts of data can be handled by introducing an in-memory appliance (see section 3.3).

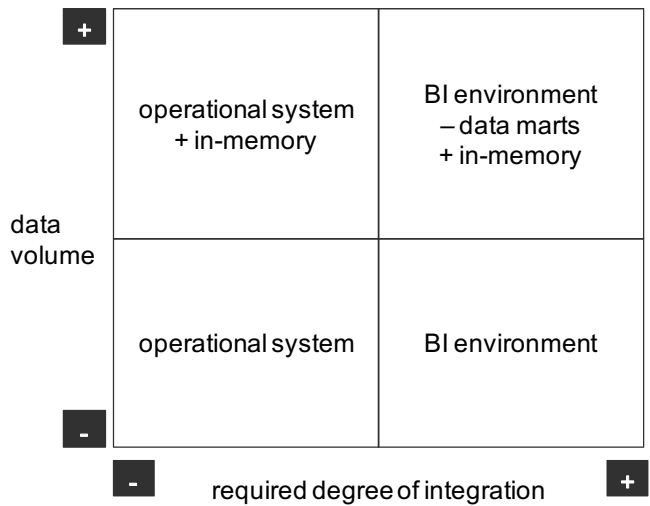


Figure 5: Four system architectures stereotypes

Summing up, in-memory appliances complement traditional operational software systems as well as BI/DWH systems. Therefore, in-memory appliances can be regarded as an incremental evolution of EIL architecture rather than a revolutionary change.

### 5 Summary and outlook

In this article we conducted a case study that helped us to assess the impact of in-memory appliances on the EIL landscape of a global automotive company. We identified two usage scenarios where in-memory appliances enhance the functionality and speed of EIL. Reflecting the case study results, we proposed to differentiate four architecture patterns. While in-memory appliances promise significant improvements in two of these architecture stereotypes, their impact is very limited in the other two. Therefore, in-memory should be regarded not to be disruptive or as “the end of” existing EIL architectures and approaches, but rather as an evolution that creates significant progress under certain circumstances.

The current discussion of the potentials and challenges of in-memory appliances fits to the increasingly important role of analytics, that is being reflected especially in the U.S.

under the label “Big Data.” However, this technology driven analysis has its risks: IT-driven transformations often fail exactly because of their focus on technology [LWU11]. Therefore, further research should investigate in scenarios where decision making is significantly improved by exploiting the full potential of all available data assets. On the basis of these scenarios, reference models and methods can be build, which finally can be leveraged by software companies in order to develop innovative business applications. Moreover, there is a need for (1) examining the effects of user groups with distinct preferences on how systems are constructed and (2) moving from pure requirements orientation to more differentiated constructional considerations.

## References

- [Ba09] Bauer, A. et al.: Referenzarchitektur. Aspekte einer Referenzarchitektur. In: (Bauer, A.; Günzel, H. eds.): Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung. 3. Ed., dpunkt.verlag, Heidelberg, 2009, pp. 33-38.
- [BG09] Bauer, A.; Günzel, H. (eds.): Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung. dpunkt.verlag, Heidelberg, 2009.
- [BU10] Brinkschulte, U.; Ungerer, T.: Mikrocontroller und Mikroprozessoren. 3. Ed., Springer, Heidelberg, 2010.
- [CC93] Codd, E. F.; Codd, S. B.; Salley, C. T.: Providing OLAP to User-Analysts: An IT Mandate. E.F. Codd & Associates, 1993.
- [Di07] Dittmar, C.: Integration von Real Time Business Intelligence in eine SOA – Auf dem Weg zum Active Business Intelligence. In: (Gluchowski, P. et al. eds.): Schlaglichter der Wirtschaftsinformatik. 1. Ed., guc, Chemnitz, 2007, pp. 129-142.
- [Di09] Dittmar, C.: Phasen des Data Warehousing. Exkurs: Realtime Data-Warehouse-Systeme. In: (Bauer, A.; Günzel, H. eds.): Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung. 3. Ed., dpunkt.verlag, Heidelberg, 2009, pp. 101-104.
- [DM88] Devlin, B. A.; Murphy, P. T.: An Architecture for a Business and Information System. In: IBM Systems Journal 27 (1988) 1, pp. 60-80.
- [Ec09] Eckerson, W. W.: 2009 TDWI BI Benchmark Report. TDWI, Chatsworth, 2009.
- [EI11] Elliott, T.: Prepare to be blown away: The 4.0 releases of BI & EIM solutions have landed!, 2011.
- [Fi10] Finucane, B. et al.: The BI-Survey 9. Business Application Research Center - BARC GmbH, Würzburg, 2010.
- [GGD08] Gluchowski, P.; Gabriel, R.; Dittmar, C.: Management Support Systeme und Business Intelligence. 2. Ed., Springer, Berlin, Heidelberg, 2008.
- [GK06] Gluchowski, P.; Kemper, H.-G.: Quo Vadis Business Intelligence? In: BI-Spektrum 1 (2006) 1, pp. 12-19.
- [In05] Inmon, W. H.: Building the Data Warehouse. 4. Ed., Wiley Computer Publishing, Indianapolis, IN, 2005.
- [KBM10] Kemper, H.-G.; Baars, H.; Mehanna, W.: Business Intelligence - Grundlagen und praktische Anwendungen. Eine Einführung in die IT-basierte Managementunterstützung. 3. Ed., Vieweg + Teubner, Wiesbaden, 2010.
- [KR02] Kimball, R.; Ross, M.: The Data Warehouse Toolkit. 2. Ed., John Wiley & Sons Inc., New York et al., 2002.
- [LWU11] Lahrmann, G.; Winter, R.; Uhl, A.: Transformationsmanagement – Aktueller Stand der Praxis und Potenziale. In: Wirtschaftsinformatik Und Management 3 (2011) 3, pp. 62-67.

- [Mo65] Moore, G. E.: Cramming more components onto integrated circuits. In: *Electronics* 38 (1965) 8.
- [NP11] Nambiar, R.; Poess, M.: Transaction Performance vs. Moore's Law: A Trend Analysis. In: (Nambiar, R.; Poess, M. eds.): *Performance Evaluation, Measurement and Characterization of Complex Systems*. Springer, Berlin, 2011, pp. 110-120.
- [PI09] Plattner, H.: A common database approach for OLTP and OLAP using an in-memory column database. In: (Binnig, C.; Dageville, B. eds.): *Proceedings, SIGMOD/PODS '09 International Conference on Management of Data*, Providence, RI, (USA) 2009.
- [PR03] Propach, J.; Reuse, S.: Data Warehouse: ein 5-Schichten-Modell. In: *WISU – Das Wirtschaftsstudium* 32 (2003) 1, pp. 98-106.
- [SA11] SAP AG (ed.): *SAP In-Memory Computing and SAP High-Performance Analytical Appliance (HANA)*. <http://www.slideshare.net/SAPanalytics/sap-hana-overview-nd-update-dec-1-2010>, Access on 23.10.2011.
- [Sc10] Schelp, J.: Near Real-Time Warehousing. In: (Chamoni, P.; Gluchowski, P. eds.): *Analytische Informationssysteme. Business Intelligence-Technologien und -Anwendungen*. 4. Ed., Springer, Berlin, 2010, pp. 463-480.
- [Wi08] Winter, R. et al.: Das St. Galler Konzept der Informationslogistik. In: (Dinter, B.; Winter, R. eds.): *Integrierte Informationslogistik*. Springer, Berlin, 2008, pp. 1-16.
- [Wi11] Winter, R.: In-Memory-Appliances – ein Fortschritt in der Informationslogistik? wird erscheinen in: *Wirtschaftsinformatik* 53 (2011) 6.
- [WW10] Wixom, B. H.; Watson, H. J.: The BI-Based Organization. In: *International Journal of Business Intelligence Research* 1 (2010) 1, pp. 13-28.

# **xSellerate: Supporting Sales Representatives with Real-Time Information in Customer Dialogs**

Johannes Wust<sup>1</sup>, Jens Krueger<sup>1</sup>, Sebastian Blessing<sup>1</sup>  
Cafer Tosun<sup>2</sup>, Alexander Zeier<sup>1</sup>, Hasso Plattner<sup>1</sup>

<sup>1</sup>Hasso Plattner Institute, 14440 Potsdam

<sup>2</sup>SAP AG, 69190 Walldorf

johannes.wust@hpi.uni-potsdam.de, jens.krueger@hpi.uni-potsdam.de  
sebastian.blessing@hpi.uni-potsdam.de, cafer.tosun@sap.com  
alexander.zeier@hpi.uni-potsdam.de, hasso.plattner@hpi.uni-potsdam.de

**Abstract:** The introduction of 64 bit address spaces in commodity operating systems and the constant drop in hardware prices make large capacities of main memory in the order of terabytes possible. Storing the entire ERP data of large companies in main memory becomes technically feasible and economically viable. Especially column-oriented in-memory databases are a promising platform for enterprise applications to run even complex reports in merely seconds. Response-times in the order of seconds mean that we can use enterprise applications in completely new ways, for example, on mobile devices. In this paper, we demonstrate how mobile applications backed by in-memory data management can support mobile workers. We illustrate this with xSellerate, a running prototype of an application that supports sales representatives with real-time product recommendations and availability checks during customer dialogs. This way, organizations can leverage their operational data to support their sales force in the field and thus, achieve a competitive advantage over rival companies.

## **1 Introduction**

With an increased address space in commodity operating systems, as well as dropping prices for memory, servers with large capacities of main memory become available at a competitive price. Combined with increased computing power due to multicore CPUs, this change enables to store and process entire data sets of even the largest companies in main memory and opens the way for in-memory data management in enterprise computing.

Especially column-oriented in-memory databases have the potential to boost the performance of data-intensive enterprise applications by orders of magnitude and reach sub-second response time for most analytical queries. First installations in large companies demonstrated a reduction of particularly long running queries from several hours on traditional row-oriented and disk-based databases down to merely seconds [PZ11]. This increase in performance has the potential to run analytical queries on the latest operational data, while simultaneously processing the transactional load on a single database system,

making analytics in real-time become true. This so-called real-time business promises significant cost reductions and revenue gains for companies: As analyzed in [Eco11], companies that have implemented real-time business techniques and were able to estimate revenue impact reported revenue gains of over 20%, and cost reduction of nearly 20%. Future gains are expected to reach even higher revenue increases of up to 28%.

Beyond the speed-up of existing applications, this step change in database performance based on in-memory data management has the potential to enable applications in completely new ways, such as analyzing enterprise data from mobile devices. Working with these devices is expected to be even more common in the future: According to a study by IDC, 1.19 billion workers worldwide will be using mobile technology in 2013, accounting for 34.9% of the total workforce [IDC11]. Providing these mobile workers with the information they need in their current context of work will allow them to deliver a distinctive experience to their customers. We demonstrate this potential based on a scenario we developed in cooperation with construction tools producer HILTI. Sales representatives at the construction site of customers can query the availability of products in real-time to tell their customers instantly when a product will be available. In addition, our application xSellerate supports the sales representative with product recommendations based on historic sales information. With the help of this information, the sales representative can offer related products more targeted to the customer.

The remainder of this paper is structured as follows: Section 2 gives a brief overview of column-oriented in-memory databases. In Section 3, we motivate the access of enterprise data from mobile devices and demonstrate the benefits with xSellerate, a mobile application in customer dialogs in Section 4. Section 5 closes the paper with some concluding remarks.

## 2 Column-oriented In-Memory Databases

This section describes the system model of the relational in-memory database on which xSellerate is based on.

An In-Memory Database (IMDB) is a database system where the primary persistence resides entirely in the main memory [GMS92]. The application prototype described in this paper is based on a column-oriented in-memory database, following the system model described in [PZ11]. All columns are stored dictionary-compressed to utilize memory efficiently. While column-orientation typically favors read-mostly analytical workloads, updating and inserting data into dictionary-compressed column structures is challenging. To achieve high read and write performance, a common concept in column-oriented databases is to split the data store in two parts [SAB<sup>+</sup>05, BZN05]: a read optimized main partition and a write optimized differential store.

Figure 1 illustrates the overall architecture, which is described in more detail in [PZ11] and [Pla11]. The database system is designed to run on a cluster of blades to allow scale-out. An interface service keeps track of client sessions and the distribution layer coordinates the working nodes that hold the actual data in main memory. The tasks of the distribution layer are distributing queries, metadata synchronization, as well as managing

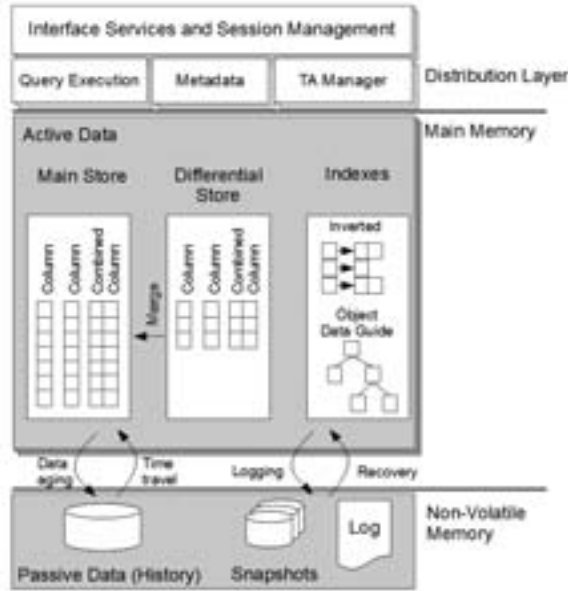


Figure 1: Conceptual overview of the underlying in-memory database [PZ11]

global transactions.

Each node that holds primary data consists of a main store, a differential store, and a collection of indexes. The read optimized main store operates on a sorted dictionary, whereas the write-optimized store appends new values to its dictionary, without sorting the dictionary each time. Each column illustrated in Figure 1 is stored physically as a dictionary vector holding the mappings of values to value IDs and an attribute vector holding the value IDs corresponding to the values of the stored records. If attributes of several columns are accessed mostly together, we can store these columns together to leverage data locality for fast read access. Indexes can be defined to further reduce access time. As enterprise applications access data mostly in terms of business objects, a special index called object data guide is provided, which is a join index for querying the tree-shaped data of business objects [PZ11].

To achieve durability in case of a system failure, the in-memory database writes log information to persistent memory. This log information is used to recover the latest consistent state in-memory in case of a failure and thus guarantees durability. Furthermore, historic data that is not accessed frequently anymore, so-called passive data, can also be shifted to non-volatile memory and loaded if needed.

### **3 A Case for Mobile Applications with In-Memory Database Backend**

Reducing the query response time of even complex analytical queries to seconds opens the way of providing mobile workers with ad-hoc information targeted to their current context. In discussions with several companies, we have identified use cases for mobile applications, ranging from negotiations with business partners to supporting sales representatives during customer dialogs. In these situations, the right information can make the difference of a better informed decision or a better customer experience, and thus a competitive advantage for a company that can provide this information to its mobile workers. For customer dialogs, all fixed information, such as customer history or the product portfolio can be provided up front. However, not all required information can be anticipated prior to a meeting and some information might also be out of date if it is pre-compiled. An example for outdated information is the current stock level to determine whether a customer order can be delivered at a given date. With mobile frontends for applications that are based on in-memory databases, we can compute queries on the fly. This enables mobile workers to specify the information they actually need in their context of work and prevents the application to show potentially outdated information. Hence, we see a huge potential in mobile applications that allow accessing a company's data in an ad-hoc manner from mobile devices.

Response time is an important factor for the acceptance of a mobile application. However, further factors such as the user interface or the context of use as described in [SW03, PD11] heavily influence the adoption of mobile applications. Especially the user interface needs to be designed in a way that users can quickly define the information they require in the current situation. Furthermore, the results need to be presented in a way that considers form factors such as the smaller screen. Complex results need to be presented in an aggregated form with the possibility of further drilling down if detailed information is required. In the following Section we describe xSellerate, an application that supports sales representatives in customer dialogs.

### **4 Case Study xSellerate: Product Recommendations in Real-Time**

In this section, we give an overview of xSellerate, an application that supports a sales representative in a customer dialog. The scenario is based on discussions we had with our project partner HILTI, a Liechtenstein based company selling top-quality products for professional customers in the construction and building maintenance industries. We first give an overview of the scenario in which a mobile application can support the sales representative while visiting a construction site. We then present an overview of the prototypical implementation in detail and focus on the use of in-memory technology to realize this scenario.

## 4.1 Real-Time Information in Customer Dialogs

While speaking to our project partner HILTI, we have learned that a well-informed sales representative is key to a successful customer dialog. Good knowledge of the construction industry in general and the customer in specific is required to understand the challenges a customer faces; furthermore, a good knowledge of the product portfolio is necessary to offer correct products and parts. This information is rather static and can be gathered in preparation of an interview. In most cases the sales representative knows the type of construction for which the customer needs support in advance and can prepare a list of potential products to offer up front. However, the exact requirements, as well as the quantity are often not known in advance. Therefore, product offerings during the customer dialog need to be adapted to the specific needs of a customer. Once a customer decides to place an order, the sales rep needs to check whether the products are available at the requested time, which is typically a process that happens after the customer interaction, potentially leading to an additional iteration with the customer.

We prototypically implemented a mobile application to support a sales representative in such a customer dialog. On the one hand, the application provides the information whether a product is available at a specific time. Furthermore, it supports the sales rep in recommending further products. Once the customer has chosen a product, the application generates a list of further products that have been bought by comparable customers that bought the same product. To make these product recommendations more meaningful for the specific customer, the sales rep has the possibility to adjust the list of comparable customers by specifying the industry, as well as the region. These two characteristics have been implemented in our prototype, but any other characteristic of an order, such as the total order volume or the size of the customer, are conceivable parameters to be considered when selecting meaningful product recommendations.

In the following, we give an overview of the prototypical application. We focus on the description of the product recommendations; the underlying concepts of the product availability check have been presented in [TMK<sup>+</sup>11].

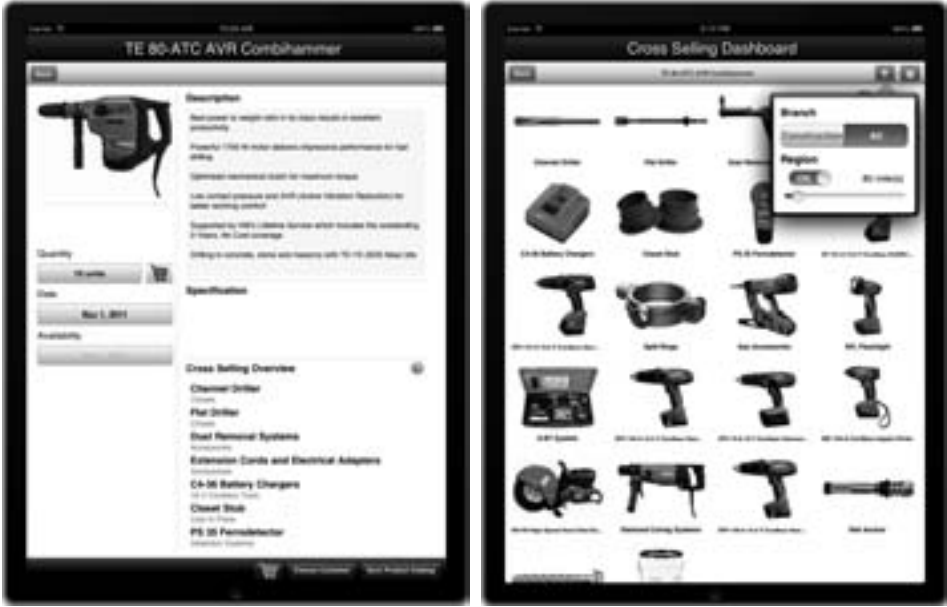
## 4.2 The xSellerate Prototype

We describe our Prototype xSellerate in this Section. We first give an overview of the features of xSellerate and then describe the architecture and the underlying data model, focusing on the application of in-memory technology.

### 4.2.1 The Frontend of xSellerate

The frontend of xSellerate is implemented on Apple's iPad device, so that a sales rep can easily take it to its customer interviews. Figure 2 shows the user interface of the xSellerate frontend. Figure 2(a) demonstrates the availability check: for a given product, the sales rep can determine if a product can be delivered at an expected delivery date in the required quantity. Furthermore this view shows a number of potential product recommendations





(a) Selecting the leading product

(b) Product Proposal Dashboard with Filters

Figure 2: The xSellerate user interface

in the Cross Selling Overview in the bottom right. To further narrow down the product recommendations, the sales rep can limit the number of proposals and filter the data by region and industry in the Cross Selling Dashboard shown in Figure 2(b). The products are ranked based on the strength of the association between the product chosen by the customer and the recommended product, as described in the next section. Operating on 40 million records, or 45 gigabytes of raw data, our prototype calculates the product recommendations on the fly in sub-second response time, including time for network transmission.

#### 4.2.2 Product Recommendations

In this section, we briefly introduce the basic definitions our product recommendations or so-called cross-selling is based on.

Given there are association rules of the form  $A \rightarrow B$  describing that customers who bought a leading product  $A$  also bought a dependent product  $B$ , then the strength of this association is rated with reference to the key parameters “Confidence”, “Support”, and “lift” [BKKN08].

Confidence determines how much an order of product  $B$  depends on ordering product  $A$  at the same time. It is defined as the conditional probability of an order of  $B$  given that  $A$

was ordered as well:

$$\text{confidence}(A \rightarrow B) = \frac{|\text{transactions}(A \cap B)|}{|\text{transactions}(A)|}$$

A product recommendation of product  $B$  given that  $A$  was already ordered is rated strong if the association  $A \rightarrow B$  has a high confidence.

Support determines the overall probability that two products  $A$  and  $B$  have been bought together. It is defined as:

$$\text{support}(A \rightarrow B) = \frac{|\text{transactions}(A \cap B)|}{|\text{transactions}|}$$

However, confidence and support together are not sufficient to rate strong associations, as the relative frequency of the order of  $B$  is not taken into account. If the confidence of  $A \rightarrow B$  is below the probability that  $B$  was ordered, we do not have a strong association. Therefore, a new parameter, called lift, is introduced which exactly addresses this problem. The lift factor takes the number of total orders that contain the dependent product into account and is defined as:

$$\text{lift}(A \rightarrow B) = \frac{\text{support}(A \rightarrow B)}{\text{support}(A) \times \text{support}(B)}$$

In case the order of products  $A$  and  $B$  is independent of each other the lift is equal 1, otherwise the lift is  $> 1$ . However lift alone is also not sufficient: if the support is pretty low, this means that overall, not many customers have ordered the product, and thus, that it is a less attractive recommendation.

Hence, we search our historic sales data for associations of product orders that have high scores in all three parameters, and provide these as recommendations to the sales rep using xSellerate.

### 4.2.3 The Architecture of xSellerate

Figure 3 gives an overview of the main components of the xSellerate Prototype. The primary sales data our application operates on is generated by enterprise applications, such as an ERP system, and stored in an in-memory database. A web server connects our mobile frontend to the database. As we calculate the parameters for product recommendations as described in the previous Section 4.2.2 on the fly, transferring all historic order information from the database to an application server would be prohibitively expensive. Therefore, we calculate the product recommendations directly in the database, bringing application logic closer to the data, which is a key design decision for fast in-memory applications [PZ11]. This leads to a very slim web server that merely establishes a secure communication channel with the mobile frontend and serves as a mediator between frontend and database.

In our prototypical implementation, we have used the SAP In-Memory Computing Engine [SAP11b] running on a 32 core server with 256 gigabyte of main memory as in-memory database. The web server is based on a slim python framework and the frontend application is implemented for Apple's iOS.

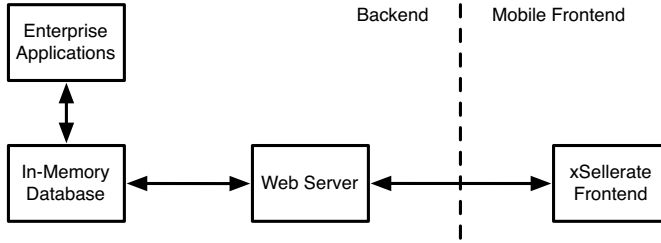


Figure 3: Architecture overview of xSellerate Prototype

#### 4.2.4 In-Memory Data Management of xSellerate

We have adopted the SAP ERP data schema [SAP11a] for the historic sales data that is analyzed to calculate product recommendations. In total, we have loaded around 40 million records of transactional sales, as well as material and customer master data in the in-memory database, summing up to a total of 45 gigabytes of data volume. Product recommendations are calculated on the fly, as we face too many combinations of potential query parameters (products, regions, branches or customers). Furthermore, this leads to a less complex data schema, as we do not have to store any pre-calculated values in form of materialized views.

In the following, we illustrate the computation of the key parameters for product recommendations to a given leading product. The calculations of confidence, support, and lift as described in Section 4.2.2 are directly implemented in SQL and executed in the database server.

We illustrate the calculation by the example of the *support* parameter for two products. Given a customer is interested in a leading product  $A$ , we determine all products  $B$  that are included in a relation  $A \rightarrow B$ , meaning that if a past order that included  $A$ , also included  $B$ . Then, the support of the leading product  $A$  and the dependent product  $B$  is calculated as follows:

```

SELECT enumerator.value / denominator.value
FROM
    (SELECT COUNT(DISTINCT left.order_ID) as value
     FROM sales_items left ,
          sales_items right
     WHERE
         left.order_ID = right.order_ID
     AND
         left.item_ID = lead_product_ID
     AND
         right.item_ID = dependent_product_ID) as
     enumerator ,
    (SELECT COUNT(*) as value FROM sales_headers) as
     denominator;

```

In an SAP ERP data schema, the order transactions are stored in a table with all the relevant order information and the ordered items are stored in a separate item table that holds an *order.ID* as foreign key to the header table. In the SQL query above, we call these two tables *sales\_headers* and *sales\_items*. To calculate all orders that contain the leading and dependent product, we count all distinct *order.IDs* in the *sales\_items* table that contain both, the leading and the dependent product. To calculate the number of all orders, we count the number of records in the *sales\_headers* table.

The calculation of the support parameter requires a self join of the *sales\_items* table, which holds 33 millions records in our scenario. Narrowing down the result set by specifying the region and the industry further increases this complexity, as we create a view containing a join of *sales\_headers*, *sales\_items* and the *customer* to obtain a table containing all completed transactions, filtered by region and industry. We have shown this simplified query for the purpose of illustration; note that the actual implementation in our prototype calculates confidence, support and lift for one leading product and all its dependent products in one single query.

As discussed in Section 3, long response times for these analytical queries are not acceptable in a mobile context. In our prototype, the SAP In-Memory Computing Engine returned all product proposals under a second, scanning more than 45 gigabytes of table data; this makes xSellerate a great use case for in-memory data management

In traditional ERP installations finding product recommendations is implemented with association rules which can be configured by the user. The calculation is then implemented as a long running batch job and the results are stored in materialized views, which require additional memory and lead to a more complex data schema. These batch jobs and the materialized view are not required in our proposed application architecture

## 5 Conclusion

In this paper we have presented xSellerate, a mobile application that supports sales representatives with real-time information in customer dialogs. The key technology to provide the information such as availability checks and product recommendations with response-time in the order of merely seconds is an in-memory database in the backend. Although we have not verified the added value to sales representatives in the field, we have demonstrated the technical feasibility of mobile applications that analyze a company's operational data in real-time. As the amount of mobile workers steadily grows, we see a huge potential for such mobile applications as they allow companies to leverage their enterprise data in completely new ways.

## 6 Acknowledgments

We especially would like to thank our project partner HILTI. The interaction with HILTI helped us to jointly identify this use case for in-memory technology. The application

XSellerate has been implemented as part of a student project. Therefore, we would like to further thank Markus Dreseler, Kai Hoewelmeyer, Christoph Mueller, Tim Berning, Henning Lohse, Uwe Hartmann, and Maximilian Schneider for their contribution to this work.

## References

- [BKKN08] Robert Blattberg, Byung-Do Kim, Pyng-do Kim Kim, and Scott Neslin. *Database marketing: analyzing and managing customers*. Springer, 2008.
- [BZN05] Peter A. Boncz, Marcin Zukowski, and Niels Nes. MonetDB/X100: Hyper-Pipelining Query Execution. In *CIDR*, pages 225–237, 2005.
- [Eco11] Oxford Economics. Real-time Business - Playing to win in the new global marketplace. <http://www.oxfordeconomics.com/free/pdfs/real-time.pdf>, 2011. [Online; accessed 19-October-2011].
- [GMS92] H. Garcia Molina and K. Salem. Main memory database systems: an overview. *IEEE Transactions on Knowledge and Data Engineering*, 4(6):509–516, 1992.
- [IDC11] IDC. Worldwide Mobile Worker Population 2009-2013 Forecast. <http://www.workshifting.com/downloads/2010/07/29/Worldwide2011>. [Online; accessed 19-October-2011].
- [PD11] Kenny Phan and Tugrul Daim. Exploring technology acceptance for mobile services. *Journal of Industrial Engineering and Management*, 4:339–360, 2011.
- [Pla11] Hasso Plattner. SanssouciDB: An In-Memory Database for Processing Enterprise Workloads. In *BTW*, pages 2–21, 2011.
- [PZ11] Hasso Plattner and Alexander Zeier. *In-Memory Data Management: An Inflection Point for Enterprise Applications*. Springer Verlag, Heidelberg, 2011.
- [SAB<sup>+</sup>05] Michael Stonebraker, Daniel J. Abadi, Adam Batkin, Xuedong Chen, Mitch Cherniack, Miguel Ferreira, Edmond Lau, Amerson Lin, Samuel R. Madden, Elizabeth J. O’Neil, Patrick E. O’Neil, Alexander Rasin, Nga Tran, and Stan B. Zdonik. C-Store: A Column-Oriented DBMS. In *VLDB*, pages 553–564, Trondheim, Norway, 2005.
- [SAP11a] SAP. ERP Software From SAP. <http://www.sap.com/solutions/business-suite/erp/index.epx>, 2011. [Online; accessed 19-October-2011].
- [SAP11b] SAP. SAP HANA 1.0 Enables Breakthrough Data Management With In-Memory Computing Engine. <http://www.sap.com/press.epx?pressid=14464>, 2011. [Online; accessed 19-October-2011].
- [SW03] Suprateek Sarker and John D. Wells. Understanding mobile handheld device use and adoption. *Commun. ACM*, 46:35–40, December 2003.
- [TMK<sup>+</sup>11] Christian Tinnefeld, Stephan Müller, Helen Kaltegärtner, Sebastian Hillig, Lars Butzmann, David Eickhoff, Stefan Klauck, Daniel Taschik, Björn Wagner, Oliver Xylander, Alexander Zeier, Hasso Plattner, and Cafer Tosun. Available-To-Promise on an In-Memory Column Store. In *BTW*, pages 667–686, 2011.

# Business Benefits And Application Capabilities Enabled By In-Memory Data Management

Gunther Piller<sup>1</sup>, Jürgen Hagedorn<sup>2</sup>

<sup>1</sup>University of Applied Sciences Mainz  
Lucy-Hillebrand-Straße 2  
55128 Mainz  
gunther.piller@fh-mainz.de

<sup>2</sup>SAP AG  
Dietmar-Hopp-Allee 16  
69190 Walldorf  
juergen.hagedorn@sap.com

**Abstract:** Current developments in the area of in-memory data management can significantly change the way business applications will be used in the future. Thus, it will be possible to store huge volumes of single documents directly in main memory for high-speed processing. To introduce and evolve in-memory data management successfully, it is necessary to understand which types of applications benefit most from this new technology. To address this question, we develop typical application patterns. They help to identify promising domains for this innovative technology. We also introduce parameters which support a systematic assessment of corresponding benefits. Our approach is illustrated with examples from different industries.

## 1. Introduction

In-memory data management (IMDM) is a strongly technology driven innovation: Powerful multi-core processors, availability of massive capacity in main memory and new developments in data organization allow storage of large data volumes in main memory for high-speed processing [PZ11]. From a business perspective, the possibility to access actual and historic data from the entire value chain of an enterprise for interactive real-time analyses has been an idealistic goal for a long time. As information technology comes closer to this goal, it supports not only better decisions but also enables new, innovative business processes (see e.g. [KO07, SAB03, Gar02]).

In this respect in-memory data management is a big step forward. Expectations on its potential for business information systems are high [Gar11]. First applications have been already delivered by software vendors or were implemented as pilot solutions [PZ11, SS11]. To successfully introduce and advance in-memory concepts in enterprises, it is necessary to understand which application scenarios could benefit most from this new technology. This is the focus of our discussion. We sketch how one can identify business domains and processes where IMDM could add significant value. In particular, we focus on the question: Which kind of analyses and decision processes fit best to the capabilities of in-memory technology? Besides an investigation of the improvement potential in existing use cases, it is important to find out how IMDM may enable new business processes. These questions seem to be of interest for software vendors,

consulting companies, as well as enterprises, who consider the use of in-memory solutions on their own.

The usage of IMDM in business application systems is apparently still a relatively new field. A comprehensive or even final assessment is not yet possible. Therefore, this contribution focuses mainly on the introduction of current ideas and trends, as initially discussed in [PH11]. As compared to earlier more technically oriented publications, see e.g. [PZ11], this article concentrates entirely on application aspects for this new technology.

## 2. Capabilities of In-Memory Data Management

In practice the response times for elaborate computations in current ERP or CRM systems can still amount to several minutes. Consequently, interactive analyses, planning runs or simulations are in many cases not possible. The separation of transaction and analytical processing is one way to deal with this problem in traditional information system architectures: Data, which are usually distributed across several transactional applications, are transferred into analytic systems using cyclical batch processes. As a consequence data available in data warehouse systems are not up-to-date but rather of historic nature. A combined analysis of actual and historic data is, therefore, often difficult (an overview about traditional analytic information systems can be found in e.g. [CG10, MM09]).

IMDM can help in many cases to overcome or reduce the sketched restrictions. In the following we briefly summarize the essential characteristics of this new technology. A detailed discussion can be found in [PZ11].

- **Short response times:** Huge data volumes for analyses, simulations or planning runs can be processed very quickly due to short access and read times, as well as high computation rates. In current performance tests IMDM could easily handle 10,000 queries per hour against 1.3 terabytes of data, returning results within seconds. Here commercially available hardware with 32 cores, 0.5 terabytes of memory and a RAID 5 disk system was used [SI11].  
Essential for this progress are: Increase in clock-speed, the use of multi-core processors, high bandwidth between CPU and main memory through a clever utilization of caches, keeping data exclusively in main memory, as well as column-oriented data storage.
- **Unified transactional and analytic data processing:** The separation of transactional and analytic data processing in traditional application architectures can be overcome with IMDM. All data, including records from latest documents, can be stored in main memory for both usage types. Delays due to cyclical batches for conventional data transfers into data warehouses are absent as already indicated. In

addition the redundant storage of data in operational and analytic systems becomes obsolete.

Crucial for these improvements is the huge, still strongly increasing capacity of main memory. Currently 2 terabyte are often used in practice. Future blade server architectures will allow storage volumes in the petabyte range. Furthermore, with column-oriented data storage efficient compression algorithms are feasible.

- **Analysis of line item data:** Aggregation for the acceleration of response times is no longer necessary. Analyses and planning runs can be based on original data records. Current restrictions due to pre-defined aggregates are obsolete.

The hardware trends sketched above are likely to continue with similar dynamics in the foreseeable future (Moore's Law). Therefore, one can anticipate that adequate powerful computers for IMDM will be available at reasonable costs.

These advantages could give the impression that IMDM may replace current data warehousing concepts completely. However, in our opinion IMDM can improve traditional data warehousing mainly in the areas of response time optimization and data storage efficiency. Multidimensional and relational OLAP implementations and corresponding specific data models may thus become less important. Data warehouse functionality around extraction, cleansing, harmonization and integration of heterogeneous data sets will still continue to be relevant.

Practical experience shows that data warehousing and IMDM can be viewed as complementary approaches. As an example, SAP has been using in-memory technology already for several years to accelerate OLAP-based reporting in their product "Business Warehouse". The next version of the Business Warehouse will optionally be delivered entirely on top of SAP's in-memory platform called HANA (High Performance Analytical Appliance). Main advantages are the reduction of operating costs through a consolidation of data base technologies, as well as an acceleration and simplification of reporting. Here the omission of aggregates and a simplification of data models are important. As a consequence the transfer of data into the Business Warehouse becomes significantly more efficient. In this sense data warehousing can be viewed - in anticipation of Section 3 - as a useful "horizontal" area of application for IMDM.

### 3. Typical Application Areas for IMDM

In this section we characterize business processes and application categories with respect to capabilities and potential of IMDM. We describe typical application patterns where the usage of IMDM provides additional benefits.



### 3.1 Selection of Business Processes

As already emphasized, the concept of IMDM is still in a quite early stage of development. A comprehensive assessment is, therefore, difficult. Nevertheless, we formulate general criteria for the classification of application areas for IMDM. In this way we point out future domains that can presumably leverage in-memory technology. We deductively derive meaningful application areas from the described technical capabilities of IMDM. In addition we inductively obtain common characteristics from already existing in-memory applications.

In this sense, promising areas for the usage of IMDM can be identified with the help of the following criteria for business processes:

- **Data dynamics:** How often and predictably do data from a business process change?
- **Range of variation:** To what extent do key figures typically change and how severe is the corresponding influence on corporate success?
- **Number of analysis options:** How many alternatives or working hypotheses shall ideally be compared?
- **Urgency of analysis results:** To what extent is it urgent to obtain analysis results, e.g. to meet given deadlines, carry out subsequent and dependent process steps or to exploit results while still meeting with specialists and managers?
- **Complexity of analysis:** Is an analysis comparatively complex due to underlying algorithms or data structures?
- **Data volume:** Are huge data volumes processed in the considered business activities?

Apparently, the use of IMDM is promising in areas with a high level of data dynamics and variation, many analysis options, urgently needed results, high complexity and huge data volumes. Therefore, the listed characteristics help to judge, whether a business process may benefit from IMDM. Of course, in real use cases all conditions will rarely exist in a single use case.

In business processes, which qualify based on the criteria described above, one can in principle achieve the following effects by using IMDM:

- **Massive increase in analysis frequency** through a drastic reduction of response times
- **Higher analysis flexibility** as predefined reporting hierarchies become obsolete and a multitude of options can be explored
- **Lower data latency**, up to and including evaluations in real-time
- **Enhancement of data bandwidth**, e.g. historic and current data can be analyzed together
- **Higher level of detail** through an access to single documents and line item data

Of course, an assessment of the overall benefit depends on the business impact resulting from changes of individual parameters in a specific use case.

### 3.2 Application Patterns

Based on the considerations in Section 3.1 one can identify a list of “ideal-typical” application patterns for the use of IMDM. These are explained in more detail below. A summary is shown in Figure 1.

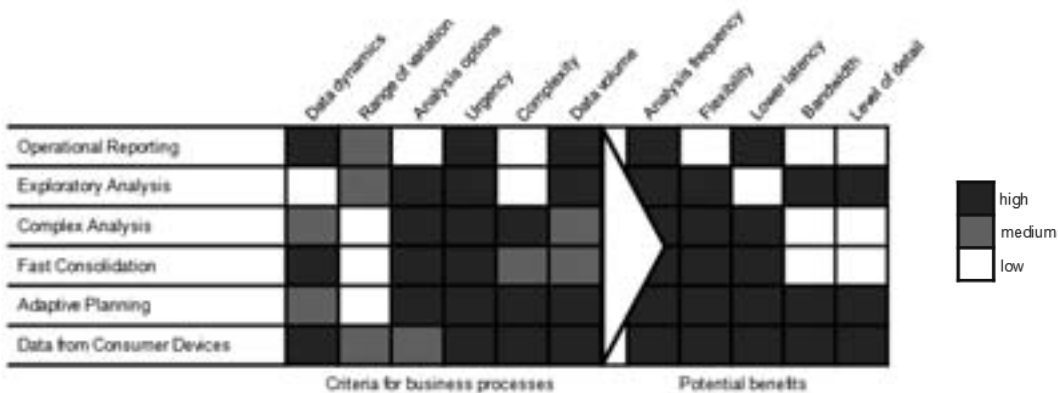


Figure 1: Typical application patterns for the use of IMDM: Business process characteristics and potential benefits.

#### 3.2.1 Operational Reporting

Day-to-day business requires fast and well informed decision making. For this purpose it is often critical to be aware of recent changes in operational transactions and to understand their business context. Here early alerts based on the analysis of many single documents and individual data records are of particular interest.

Based on the criteria of Section 3.1 the pattern “Operational Reporting” can be classified as follows:

- High level of dynamics and variation for data due to their operational character
- Great urgency, as critical situations have to be identified immediately, to rapidly take countermeasures
- In many cases very large volumes of data, as single transactions have to be considered

The potential benefits of IMDM result primarily through:

- Reduced data latency, so that operational decisions can be taken in due consideration of the most recent facts
- Increased frequency of analysis, up to continuous ongoing evaluation. Reports can then be used anytime in operational processes without causing delays.

A striking example for the application of this pattern is the monitoring of sales and inventory data in the retail industry to avoid out-of-stock situations. Corresponding data volumes are typically very high: For leading retailers 100 million data records per day are common. Huge variations of sales figures can be observed in particular during promotions where they reach up to 300%. At present analyses are carried out in larger time intervals. As calculations usually last several hours, batch processing is used. As a consequence, data records, on which new reports are based, are mostly outdated. This often leads to a poor alignment of orders and deliveries. With IMDM most current analysis results can be made available on an ongoing basis. This allows a significant reduction of out-of-stock risks. Replenishment quantities, which are commonly delivered up to five times per day, can then be adjusted to most current sales forecasts.

Quality monitoring is another case for future, IMDM-based operational reporting. For example, in the semiconductor industry wafer fabrication facilities place a great deal of importance on the increase of yields. During production a large amount of material data and equipment parameters are captured continuously. Information from several hundred process steps have to be evaluated and brought into context. The sooner one can identify patterns with a high probability for defects, the quicker countermeasures can be triggered to reduce damaging yield excursions. Besides the operational monitoring of production data, systematic development and improvement of algorithms for early failure forecast and diagnosis is also important here (see e.g. CHH07, Mer11]).

Altogether, it seems that IMDM opens up a considerably more effective operational utilization of information already available today: For example, companies are interested in always up-to-date reports on the payment behavior of customers, e.g. measured in “Days-Sales-Outstanding”. If available, they allow rapid interventions in operational processes, like shipments of goods or sales activities. This example clearly shows how the usage of operational data (here for incoming payments) can be extended beyond traditional areas (in this case: dunning) to directly control operational activities.

Another natural domain for the use of in-memory technology seems to be the management of complaints. Here it is obviously critical that a current, complete and consistent picture of a specific customer problem is available without major delay for all involved persons in a company. In this case IMDM can serve as the “single source of truth”. In addition it is important to put complaints about deficiencies into a statistic and historic context. Then, in specific customer situations, appropriate actions can be taken, e.g. through accommodating concessions. With the help of the analytic capabilities of IMDM systems, the required analyses could be realized relatively quickly and easily.

### **3.2.2 Exploratory Analysis of Mass Data**

In many cases predefined standard reports are not sufficient for an appropriate assessment of business situations. Often exploratory investigations with a free choice of selectable analysis criteria are required.

Profitability analysis can serve as an almost classical use case: In investigations of contribution margins data are usually aggregated in advance to improve response times. Profitability fluctuations can then be evaluated only for predefined product or market segments. In this case calculations of contribution margins down to the level of customer-product-combinations are not possible. Course grained data also limit detailed evaluations of deviations in contribution margins related to changes in quantity, price, costs, discounts or structural variations. IMDM allows investigations without any predefined aggregation. An exploratory and sophisticated assessment of influencing factors – e.g. in the context of a gross margin flow analysis [Lin88] – is then possible. In this context it is worth to mention that IMDM also offers great potential for the efficient implementation of statistical analysis methods, as well as data mining algorithms. Partially automated top-down-navigation is a corresponding example. Here one aims at a detailed understanding of main causes for aggregated deviations from planed, target or historic figures.

Customer segmentation is another common example for an exploratory analysis of mass data. The idea is to identify customer segments which can be targeted in marketing campaigns successfully and efficiently – often measured through conversion rates. For this purpose huge amounts of customer and market data have to be explored with respect to flexibly chosen combinations of characteristic values. This extremely interactive type of investigation is possible only if response times are short.

The following criteria from Section 3.1 seem obvious for this application pattern:

- Large variations of key figures
- Many analysis options
- High urgency of results
- Large data volumes

Potential improvements through the use of IMDM are:

- Increased analysis frequency
- Increase of analysis flexibility, e.g. following variable reporting structures
- Extension of data bandwidth
- Greatest level of detail

### **3.2.3 Complex Analysis**

There is still a whole series of reports in today's ERP systems which cannot be used interactively due to their long response times. In practice they are therefore often run in batch mode. Examples are material availability checks, complex pricing, cost allocation and dunning runs. It seems natural to use IMDM for a drastic reduction of response times for these analyses and embed them better into operative processes.

Typical for this pattern are:

- High level of data dynamics
- Many decision options with significantly different business consequences

- High urgency to continue operational processes quickly and seamlessly
- Partially very complex analyses
- High data volume

The following improvements can be obtained with IMDM:

- Increased analysis frequency
- Increase of analysis flexibility; different strategies can be explored, e.g. for the allocation of material stock to customer orders
- Reduced data latency; reports carried out in an ad hoc way accurately reflect a momentary business situation

Within the mentioned examples IMDM can be used in practice in the following scenarios:

Available-to-Promise checks provide available quantities of requested products and corresponding delivery dates (see e.g. [KMZ+09]). The sequence of planned deliveries can be changed according to the priority of customers or customer orders. The consequence of changes in the sequence of production or shipment with respect to key figures, like revenue, penalties or customer satisfaction, can be evaluated through corresponding simulations. IMDM allows you to commit to delivery dates immediately, e.g. in conversations with customers. In addition different strategies for the allocation of material can be played through and compared on short notice.

The use case “flexible pricing in sales” seems to be quite similar. In many industries prices depend on a complex set of rules – e.g. in consumer industries. In ERP or CRM systems this is typically realized through condition techniques [HH10]. In particular for sales representatives, who access business applications through mobile solutions, response times in the order of a few seconds – as possible with IMDM – are desirable. In negotiations with customers different options for prices and discounts can then be investigated directly.

### **3.2.4 Fast Consolidation**

The term “Fast Consolidation” summarizes all use cases, where documents created in the past have to be consolidated according to given rules up to a fixed deadline, or on request. Typical are attempts to accelerate the financial closing for the filing of monthly, quarterly or annual reports – often referred to as “Fast Close” (see e.g. [Bra09, Sch06]). Amongst other steps it is necessary here, to reconcile internal payables as quickly as possible. In large enterprises this often involves the evaluation of several million invoice documents. For many companies it is also important to create consolidated interim reports with most current data for different reporting levels on short notice.

Altogether this field of application is characterized through:

- Huge data dynamics due to documents created on short notice
- Different analysis options, e.g. for the allocation of costs

- High urgency through deadline constraints, e.g. as a consequence of legal requirements or stakeholder demand
- High level of complexity though a comprehensive body of rules and regulations or elaborate calculation methods
- Large data volumes

IMDM can lead to the following improvements:

- Increase of the number of consolidation runs
- Increased flexibility; different scenarios for consolidation can be compared
- Low data latency; short term postings can be accounted for immediately

One can imagine that the use of IMDM will massively influence the way consolidations are carried out: Batch procedures - which still dominate today - will evolve towards interactive and iterative practices. In this way different calculation schemes can be run through and compared. In addition one could work with original documents instead of replicated data. Problems with data consistency and efforts for corresponding adjustments would be reduced and eventually become obsolete.

A similar use case is the consolidation of up-to-date sales figures, like expected or closed deals: In many industries the closing of deals is emphasized towards the end of quarters - sometimes driven by budget cycles. As a consequence there are many late transactions - just before quarter closing. To allow for short-term interventions of sales management teams and to support the preparation of quarterly reports, it is necessary to provide consolidated results with most current data as fast as possible.

### **3.2.5 Adaptive Planning**

The use of IMDM for planning scenarios has been discussed by Sinzig and Sharma in [SS11]. For completeness we provide a short classification with respect to the criteria from 3.1.

Planning processes often show the following characteristics:

- Partly high level of data dynamics, e.g. in cases where the latest customer requests need to be included
- High number of planning options which potentially have to be considered
- High level of urgency to trigger subsequent business activities
- Complex planning algorithms
- Large data volumes, in particular for company-wide planning runs

Potential benefits from IMDM arise from:

- Plenty of planning runs with different variables
- Reduced latency of data, so that planning results can be adjusted better to momentary circumstances

- Increase of data bandwidth, to enable company wide, integrated planning - avoiding “planning-silos”
- High level of detail for planning objects

A classical example can be found in production planning [KMZ+09, SK05]. Planning runs using MRPII or APS usually need many hours. Therefore, necessary computations are typically carried out in batch mode and repeated in intervals of several days only. To save on computing power, planning runs are often restricted to master scheduled items. The consequences are inaccurate results with partly outdated information. Due to reduced computing time, planning runs can be completed with IMDM in short time intervals with always up-to-date starting parameters. Then realistic and exact results on the level of materials can be expected. This enables flexible reactions to changes in market conditions – e.g. short term fluctuations in customer demand.

### 3.2.6 Analysis of Data from Consumer Devices

For providers of traditional business software, like SAP, the use of IMDM seems to open up a new domain: In-memory technology enables the processing of extreme volumes of highly dynamic consumer data from embedded systems, as they appear e.g. in domestic homes.

“Smart Meter Analytics” can serve as a first use case for which a solution is currently in development at SAP. Consumer data from electric meters located in households and businesses are central to this application. These data are brought together using intelligent electric supply networks - so called smart grids. With IMDM they are then analyzed in nearly real time. On the one hand, this allows consumer-oriented solutions, which e.g. benchmark the energy consumption of individual households and serve consumers with suggestions for improvements. On the other hand, it is planned to provide components which support utility companies in their efforts to dynamically adjust capacities and prices to actual consumption profiles.

According to the classification from Section 3.1, large potential for the usage of IMDM can be expected:

- Apparently extremely high dynamics of consumer data
- Potentially large range of variations, e.g. for electricity consumption
- Many alternatives, e.g. different pricing policies for electricity
- High urgency, e.g. to immediately react to particular consumption patterns of consumers
- High complexity of analyses, e.g. for simulations
- Extremely large data volumes

Expected benefits are:

- Massive increase of analysis frequency, continuous data access for consumers
- Significant rise of analysis flexibility: The consequences of different measures, e.g. for pricing, can directly be simulated.

- Reduction in data latency: All analyses are based on the most recent consumer behavior.
- Maximal bandwidth of data: In an ideal case all actual and historic data are available.
- High level of detail: For example, one can utilize data for electricity consumption, which are typically measured in time intervals of a few minutes.

## 4. Summary

This article highlights actual and future key areas for the use of IMDM in business information systems. Our discussion should be seen as a current snapshot rather than a final assessment. This is mainly due to the fact that the use of IMDM in business applications is still in an early stage of development. Therefore, a detailed evaluation of particular application scenarios is left to future investigations.

In our opinion, it already becomes apparent that in-memory technology will have a massive impact on the redesign of existing information systems and the preferred way of their use within business processes. Applications, which today can only be operated in batch mode due to extensive computing times, will - through IMDM – evolve towards interactive, exploratory, ad hoc usable solutions.

Furthermore, IMDM opens up a completely new type of business applications: Solutions which involve consumers seem to be of particular interest here. The aim is to leverage the corresponding huge data volumes in “consumer apps”, as well as applications for business purposes.

## References

- [CHH07] Chien, C.; Hsu, S.; Hsu C.: Enhancing Competitive Advantages and Operational Excellence for High-Tech Industry through Data Mining and Digital Management. In: Liao, W.; Triantaphyllou, E. (Eds.): Recent Advances in Data Mining of Enterprise Data: Algorithms and Applications. World Scientific, 2007.
- [Bra09] Bragg, S.: Fast Close. John Wiley & Sons, 2009.
- [CG10] Chameni, P.; Gluchowski, P. (Eds.): Analytische Informationssysteme. Springer, 2010.
- [Gar02] Gartner: Now is the Time for Real-Time Enterprise. 2002. <http://www.gartner.com/pages/story.php.id.2632.s.8.jsp>. Retrieved June 15th 2011.
- [Gar11] Gartner: Gartner Identifies Nine Key Data Warehousing Trends for the CIO in 2011 and 2012. Gartner Press Release, February 2011. <http://www.gartner.com/it/page.jsp?id=1542914>. Retrieved June 15th 2011.
- [HH10] Hirn, M.; Herhuth, W.: Preisfindung und Konditionstechnik in SAP ERP. SAP PRESS, 2010.
- [KMZ+09] Knolmayer, G.; Mertens, P.; Zeier, A.; Dickersbach, J.: Supply Chain Management Based on SAP Systems. Springer, 2009.
- [KO07] Kagermann, H.; Oesterle, H.: Geschäftsmodelle 2010. FAZ, 2007.



- [Lin88] Link, J.: Verbreitung und Einsatzformen der Deckungsbeitrags-Flussrechnung in der Industrie. Die Betriebswirtschaft, Nr. 6, 1988.
- [Mer11] Mertens, P.: Integrierte Informationsverarbeitung I, Gabler, 2011.
- [MM09] Mertens, P.; Meier, M.: Integrierte Informationsverarbeitung II, Gabler, 2009.
- [PH11] Piller, G.; Hagedorn, J.: Einsatzpotenziale für In-Memory Data Management in betrieblichen Anwendungssystemen. Wirtschaftsinformatik und Management, Nr. 5, 2011.
- [PZ11] Plattner, H.; Zeier, A.: In-Memory Data Management. Springer, 2011.
- [SAB03] Scheer, A.; Abolhassan, F.; Bosch, W. (Eds.): Real-Time Enterprise. Springer, 2003.
- [Sch06] Schulte, O.: Fast-Close-Abschlüsse und Schadenrückstellungen nach HGB, IAS/IFRS und US-GAAP. Deutscher Universitätsverlag, 2006.
- [SI11] SAP Inside: First HANA Performance Results Released. March 2011. [http://www.insidesap.com.au/\\_blog/News/post/First\\_HANA\\_performance\\_results\\_released](http://www.insidesap.com.au/_blog/News/post/First_HANA_performance_results_released). Retrieved June 15th 2011.
- [SK05] Stadtler, H.; Kilger, C.: Supply Chain Management and Advanced Planning. Springer, 2005.
- [SS11] Sinzig, W.; Sharma, K.: In-Memory-Technologie: Verbesserung bei Planung, Simulation und Entscheidungsunterstützung. Wirtschaftsinformatik und Management, Nr. 2, 2011.

# Data-Warehousing 3.0 – Die Rolle von Data-Warehouse-Systemen auf Basis von In-Memory-Technologie

Maik Thiele, Wolfgang Lehner, Dirk Habich

Lehrstuhl Datenbanken, Institut für Systemarchitektur  
Technische Universität Dresden, Fakultät Informatik  
01069 Dresden

{maik.thiele, wolfgang.lehner, dirk.habich}@tu-dresden.de

**Abstract:** In diesem Beitrag widmen wir uns der Frage, welche Rolle aktuelle Trends der Hard- und Software für Datenbanksysteme spielen, um als Enabler für neuartige Konzepte im Umfeld des Data-Warehousing zu dienen. Als zentraler Schritt der Evolution im Kontext des Data-Warehousing wird dabei die enge Kopplung zu operativen Systemen gesehen, um eine direkte Rückkopplung bzw. Einbettung in operationale Geschäftsprozesse zu realisieren. In diesem Papier diskutieren wir die Fragen, wie In-Memory-Technologie das Konzept von Echtzeit-DWH-Systemen unterstützt bzw. ermöglicht. Dazu stellen wir zum einen eine Referenzarchitektur für DWH-Systeme vor, die insbesondere push- und pull-basierte Datenversorgung berücksichtigt. Zum anderen diskutieren wir die konkrete Rolle von In-Memory-Systemen mit Blick auf konkrete Aspekte wie der Frage optionaler Persistenzschichten, Reduktion der Batchgröße, Positionierung von In-Memory-Techniken für den Aufbau eines Corporate Memorys und die schnelle Bereitstellung externer Datenbestände zur Unterstützung situativer BI-Szenarien.

## 1 Einleitung

Data-Warehouses (DWH) haben seit ihrer ersten Erwähnung 1988, damals noch unter dem Begriff „Information Warehouse“, einen enormen Entwicklungsprozess durchlaufen, der durch aktuelle Trends wie beispielsweise hin zur Integration semi-strukturierter Datenbestände oder die Versorgung Echtzeit zunehmend verstärkt wurde. Hatten in der Vergangenheit Data-Warehouse-Systeme lediglich den Auftrag der Informationsversorgung von Management und Wissensarbeitern, entwickeln sie sich schrittweise zur zentralen Plattform für die integrierte Informationsversorgung eines Unternehmens. Oftmals wird in diesem Zusammenhang daher bereits von „Corporate Memory“ oder „Information Hub“ gesprochen. Diese Erweiterung schließt sowohl strategische und taktische Analysen basierend auf vorberechneten Berichten oder multidimensionalen Würfeln als auch die Unterstützung operativer Geschäftsprozesse mit ein, so dass die Feedback-Schleife von den DWH-Infrastrukturen zu operationalen Systemen realisiert wird. Letzteres macht insbesondere die Integration von Daten in Echtzeit notwendig, was jedoch im Konflikt zur Philosophie klassischer Data-Warehouse-Systeme steht, d.h. einen konsistenten Snapshot zu liefern, der als Gegenstand einer ausführlichen Analyse dient.

Der Begriff Echtzeit im Kontext von DWH-Systemen beschreibt dahingehend auch die Forderung, Änderungen in der Diskurswelt zeitnah im Data-Warehouse abzubilden. Aufgrund der Mehrdeutigkeit des Echtzeit-Begriffs wird in der Literatur auch von Near-Real-Time-, Right-Time-, On-Time- oder Living-DWHs gesprochen. Dabei muss man sich die Frage stellen, inwieweit die klassischen aus der Literatur bekannten DWH-Referenzarchitekturen [Le02,BG09,KR02] noch Gültigkeit besitzen und welchen Einfluss ein Design auf Ebene der Infrastruktur auch Auswirkung auf die technische Grundlage hat. All diese Aspekte wurden noch nicht, durch das von Inmon et al. beschriebene DWH 2.0 [ISN08] adressiert. Inmon et al. haben in ihrem Buch lediglich das Konzept des DWH um die Integration von semistrukturierten Datenquellen erweitert und den Begriff des „DWH als lebendes Archiv“ postuliert. Im Sinne einer Evolution wollen wir mit der zusätzlichen Betrachtung von In-Memory-Technologien sowie der Integration von Daten in Echtzeit das sogenannte DWH 3.0 positionieren.

Mit der Frage nach dem Data-Warehouse „der nächsten Generation“ müssen auch mögliche Ausprägungen einer Architektur diskutiert werden. Galt bisher üblicherweise eine homogene technische Plattform (mit unterschiedlichen Datenbanken etc.) als Standardlösung, so werden aktuell vermehrt zum Beispiel Map-Reduce-Systeme zur Vorverarbeitung insbesondere zur Entity-Extraktion im Rahmen einer Textanalyse positioniert; relationale Ansätze kommen erst für die späteren Schichten des DWH-Stacks zum Einsatz. Spielen Map-Reduce-Systeme eine Rolle im Kontext der funktionalen Erweiterung von DWH-Ansätzen durch Integration semi-strukturierter Datenbestände, fokussiert sich dieser Beitrag auf die Erweiterung mit Blick auf Echtzeit-DWHs auf Basis von main-memory-zentrischer Datenbanktechnologie.

Den Trend, spezielle (Teil-)Systeme als Lösung spezifischer Probleme aufgreifend, kritisieren Stonebraker et al. [SBC+07, SMA+07] die Produktstrategie der kommerziellen Datenbankanbieter, die dem Paradigma „one size fits all“ folgend, ihre Altsysteme stetig erweitern, um neuen Anforderungen gerecht zu werden. Stattdessen wird ein „rewrite from scratch“ gefordert, indem für spezifische Probleme wie Datenstromverarbeitung, Textsuche, datenintensive Analysen oder Informationsintegration angepasste Lösungen geschaffen werden. Diese Entwicklung hin zu neuen Lösungen für die Datenverarbeitung und -analyse wird zusätzlich noch durch aktuelle Entwicklungen im Hardwarebereich verstärkt. Die zunehmende Anzahl von Rechenkernen in Many-Core-Architekturen, der Einsatz spezialisierter Prozessoren wie GPU's (engl. Graphic Processing Unit) oder FPGA (engl. Field Programmable Gate Array) und vor allem die stetig wachsenden Hauptspeichergroßen haben eine Vielzahl neuer Datenbanksysteme hervorgebracht. Ein zusätzlicher Treiber auf der Hardware-Seite ist der Trend hin zu energieverbrauchsoptimierten Systemen, im Zuge dessen neue Prozessortypen und damit einhergehend neue Datenbanklösungen entstanden sind. Im Kontext dieser sehr Hardware-zentrischen Softwareentwicklung spricht man auch von einem „Hardware-Software Co-Design.“

Die enorme Bedeutung der Datenanalyse in vielen Bereichen des Lebens hat zusätzlich dazu geführt, dass ehemals kleinen Nischen groß genug geworden sind, so dass sich Anbieter speziell angepasster Datenbanklösungen behaupten können. Allerdings ist zu

bemerken, dass in den letzten beiden Jahre getrieben durch mehrere Übernahmen (z. B. Sybase durch SAP, Netezza durch IBM, Vertica durch HP) bereits ein deutlich sichtbare Konsolidierung eingesetzt hat.

Der Schritt weg von General-Purpose- hin zu Speziallösungen ist im Bereich der DWH Systeme keineswegs neu. Die Trennung operationaler und analytischer Datenverarbeitung wurde durch das Konzept des Data-Warehousing seit jeher als Konzept postuliert. Inwiefern diese Unterscheidung jedoch angesichts der Forderungen nach Datenanalysen in Echtzeit und den zur Verfügung stehenden technischen Möglichkeiten, insbesondere der main-memory zentrischen Datenverarbeitung noch sinnvoll erscheint, ist im Folgenden zu diskutieren. Dazu sollen in Abschnitt 3 die Kernaspekte einer DWH-Architektur der „nächsten Generation“, welche die Informationsversorgung in ihrer ganzen Breite – von historischen bis hin zu aktuellen Daten – abdecken kann, betrachtet werden. Gleichmaßen ist die Instanziierung potentieller DWH-Architekturen durch konkrete Data-Management-Technologien zu diskutieren. Insbesondere wird untersucht inwiefern In-Memory-Datenbanken den Anforderungen moderner DWH-Anwendungen gerecht werden und welche Implikationen sich daraus für eine DWH-Infrastruktur ergeben. Dieser Diskussion vorangestellt werden zunächst in Abschnitt 2 aktuelle Architekturansätzen zur Datenanalyse vorgestellt. Der Artikel endet mit einer Zusammenfassung und einem Ausblick in Abschnitt 4.

## **2 Trends in Hardware- und DBMS-Architekturen**

Aus den zunehmenden Anforderungen an Systeme zur Datenanalyse sind in den letzten Jahren eine Reihe von Architekturansätzen entstanden. Die wesentlichen Techniken – Parallelisierung, Einsatz spezieller Hardware und die massive Nutzung vom Hauptspeicher – werden in diesem Kapitel in aller Kürze vorgestellt und diskutiert.

### **Parallelisierung**

Ein wichtiger Ansatz, um trotz steigender Anzahl von Anfragen und größer werdenden Datenvolumen Antwortzeiten für einen interaktiven Betrieb zu erreichen, ist die durchgängige Parallelisierung innerhalb von DBMS-Architekturen. Hier werden die folgenden drei Ansätze unterschieden: Shared-Memory, Shared-Disk und Shared-Nothing. Der einfachste aber zugleich auch am wenigsten leistungsfähige Ansatz ist die Shared-Memory-Architektur (z. B. Microsoft SQL Server). Alle Prozessoren teilen sich gemeinsamen Haupt- und Plattenspeicher, wodurch die Implementierung dieser Systeme vereinfacht wird, da auf verteilte Sperrprotokolle verzichtet werden kann. Die Skalierung wird dadurch beeinträchtigt, da sich alle Prozessoren den gleichen Bus für E/A-Operationen und Speicherzugriffe teilen müssen. Ähnlichen Beschränkungen unterliegt die Shared-Disk-Architektur, wobei voneinander unabhängige Verarbeitungsknoten mit jeweils eigenem Hauptspeicher auf einen gemeinsamen Plattenspeicher zugreifen. Oracle RAC und darauf aufbauend die Exa\*-Serie bilden ein Beispiel für diese Architektur.

Ein höheren Grad an Skalierung wird üblicherweise mit einer Shared-Nothing-Architektur, auch als Massive-Parallel-Processing- oder kurz MPP-Architektur bezeichnet, erzielt. In diesem Ansatz besitzen alle Verarbeitungsknoten ihren eigenen Haupt- und üblicherweise auch Festplattenspeicher.

Die Datentabellen werden horizontal partitioniert und auf die Verarbeitungsknoten verteilt. Pufferspeicher und Sperrtabellen werden lokal für jeden Verarbeitungsknoten gehalten. Bekannte Vertreter dieser Architektur sind zum Beispiel Teradata, Netezza (mittlerweile IBM) und Greenplum (mittlerweile EMC<sup>2</sup>). Zusätzlich interessant werden MPP-Systeme beim Einsatz einfacher Commodity-Hardware, wodurch sich kostengünstig sehr leistungsfähige Systeme konfigurieren lassen. In extremer Form wird dies durch das Map-Reduce-Framework von Google umgesetzt womit nebenläufigen Berechnungen, wie zum Beispiel die Vorverarbeitung unstrukturierter Dokumente, auf viele tausend Knoten skaliert werden können.

### **Hardwarebeschleunigung**

Einen anderen Weg beschreiten Systeme, die durch den Einsatz spezieller und meist proprietärer Hardware Datenbankoperationen beschleunigen. Xtreme Data Appliance ergänzt beispielsweise einen Standardprozessor um einen FPGA-Chip. Der FPGA-Chip dient dabei der Beschleunigung von Kernroutinen bei der Anfrageverarbeitung. Netezza verwendet ebenfalls FPGAs und ergänzt mit diesen den Festplatten-Controller zur Datenvorverarbeitung. Weiterhin existieren Forschungsprojekte zur Nutzung spezieller Mehrkernarchitekturen, wie etwa Grafikkartenprozessoren oder kombinierte Systeme wie die Intel Sandybridge oder AMD Bulldozer-Architektur.

Aus Sicht datenintensiver Data-Warehouse-Anwendungen in Kombination mit Echtzeit-Anforderungen, werden jedoch vor allem Hauptspeicherorientierte Ansätze relevant, die neben einer leseoptimierten Datenablage auch schreiboptimiert mit einer Historisierung der Datenbestände ermöglichen. Entsprechende Systeme werden im folgenden Abschnitt diskutiert.

### **Hauptspeicherdatenbanken und spaltenbasierte Speicherung**

Die Optimierung von Data-Warehouse-Anfragen sowohl für Lese- als auch für Schreiboperationen wird vor allem durch die Reduzierung des E/A-Aufwandes erreicht. Auf Seiten der Algorithmen und Software wird dies vor allem durch Materialisierung von Aggregationen, Reduzierung von Zwischenergebnissen durch den Optimierer, spaltenorientierte Speicherung, Kompression usw. erreicht. Auf Hardwareseite ist der E/A-Aufwand durch die extensive Nutzung vom Hauptspeicher reduzierbar, indem die Daten vollständig im Arbeitsspeicher abgelegt und somit Festplattenzugriffe überflüssig werden. Die effiziente Persistierung von Schreiboperationen wird dabei durch effiziente Datenstrukturen und Gruppen-Commits unterstützt. Bei bis zu zwei Terabyte Hauptspeicher, die heute schon für einen einzelnen Server verfügbar sind, können bereits viele kleinere Data-Warehouse-Szenarien komplett im Hauptspeicher abgebildet werden.

In Anlehnung an einen Vortrag von Jim Gray aus dem Jahr 2006 mit dem Titel „Tape is Dead, Disk is Tape, Flash is Disk, RAM Locality is King“ wurde aufgrund der wachsenden Hauptspeichergößen die Phrase um „Memory is the new disk“ erweitert. Im Vergleich zu dem rasanten Wachstum bei den Hauptspeichergößen hat sich allerdings die Speicherbandbreite nur relativ gering verbessert. Um trotzdem die Prozessoren optimal auszulasten, ist es notwendig die vorhandenen mehrstufigen Cache-Architekturen nutzbar zu machen und spezielle Datenbankoperatoren und Datenstrukturen zu entwickeln.

Durch den Einsatz von Kompressionsalgorithmen, die eine signifikante Reduzierung des Datenvolumens erlauben, ist es möglich auch sehr große Datenbanken komplett im Hauptspeicher abzulegen. Damit einhergehend werden in OLAP-Szenarien die Daten häufig nicht mehr zeilen-, sondern spaltenweise abgelegt. Erfolgt der Zugriff auf Tabelleneinträge in OLTP-Anwendungen meist bezogen auf einen ganzen Datensatz, was insbesondere auch für Einfüge-, Löscho- und Aktualisierungsoperationen gilt, so ist die zeilenorientierte Speicherung im Vorteil. Wird jedoch, wie für OLAP-Anfragen typisch, meist nur auf wenige Spalten einer Tabelle zugegriffen, so reduziert die spaltenorientierte Speicherung den E/A-Aufwand und damit die Kosten der Anfrageverarbeitung erheblich. Ohne effiziente Datenstrukturen sind Änderungsoperationen in spaltenorientierten Systemen in der Regel teurer, da die Daten eines Datensatzes nicht hintereinander gespeichert, sondern auf mehrere Spalten verteilt werden müssen.

Die aufeinanderfolgende Speicherung von Daten gleichen Typs bei spaltenorientierten Datenbanksystemen erlaubt den Einsatz vielfältiger Kompressionstechniken, wie zum Beispiel Lauflängenkodierung, Delta-Kodierung oder Wörterbuchkompression. Dadurch wird der E/A-Aufwand im Vergleich zu zeilenorientierten Systemen nochmals um ein Vielfaches reduziert. Die Anfrageverarbeitung wird durch den Einsatz von Kompressionstechniken jedoch nicht notwendigerweise verzögert. Oftmals kann die Datenverarbeitung direkt auf den komprimierten Daten durchgeführt werden. Implementiert wurde die spaltenorientierte Speicherung und Anfrageverarbeitung sowohl durch Forschungsprototypen wie C-Store (wurde in Vertica kommerzialisiert und 2011 von HP übernommen) und Monet-DB sowie in kommerziellen Systemen wie Sybase IQ (diskbasiert, mittlerweile zu SAP gehörend). Für sehr große Data-Warehouse-Systeme ist durch die zusätzliche Anwendung einer Shared-Nothing-Architektur (z.B. SAP BW Accelerator bzw. SAP HANA) eine nahezu beliebige Skalierung möglich.

### **3 Die Data-Warehouse-Architektur 3.0**

Das klassische DWH-Referenzmodell beschreibt die Rolle der einzelnen Datenschichten einer DWH-Architektur (Arbeitsbereich, Konsolidierte Basisdatenbank, DWH-Datenbank, Data Marts) sowie den Datenfluss zwischen diesen Komponenten. Aktuelle Fragestellungen hinsichtlich des Aspekts der Echtzeit und damit einhergehend zu Batchgrößen, optionalen Persistenzschichten, alternativen Zugriffssemantiken usw. bleiben jedoch unberücksichtigt.

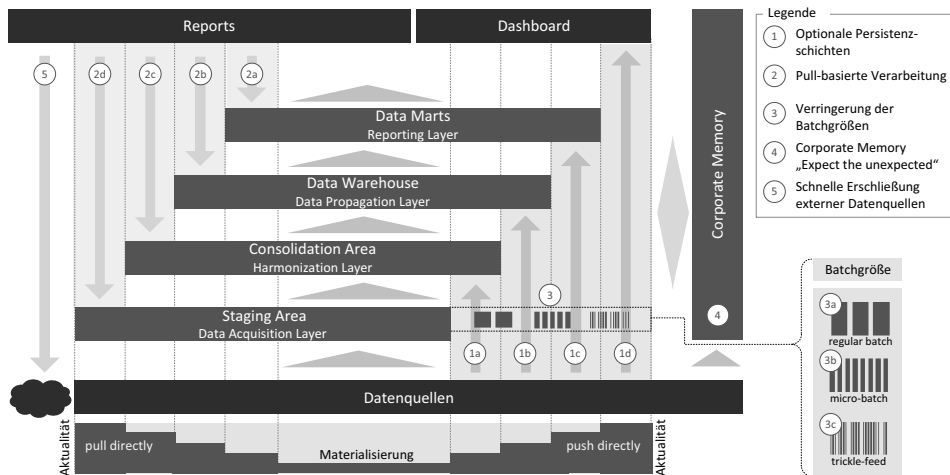


Abbildung 1: Die fünf Aspekte einer Echtzeit-Data-Warehouse-Architektur.

Unter Berücksichtigung der Trends im Bereich der DBMS-Architekturen (siehe Abschnitt 2) sowie den zunehmenden Anforderung nach Datenanalysen in Echtzeit wollen wir im Folgenden das klassische DWH-Referenzmodell auf seine Schwächen untersuchen und schrittweise erweitern.

### Optionale Persistenzschichten

Ein DWH-System speist sich aus einer großen Menge üblicherweise heterogener Quellsysteme, deren Daten in einem aufwändigen Prozess bereinigt, transformiert, konsolidiert und in eine für die Analyse optimierte Darstellung gebracht werden müssen. An dieser schrittweisen Verfeinerung der Daten sind eine Vielzahl von Transformationen und Repräsentationen in den jeweiligen Persistenzschichten beteiligt, die aufeinander aufbauend jeweils einen weiteren Verfeinerungszustand der Ursprungsdaten materialisieren. Diese von der klassischen Referenzarchitektur vorgeschrieben Materialisierung der Zwischenstände führt, zusammen mit den notwendigen Transformationsprozessen und Qualitätssicherungsmaßnahmen, zu Verzögerung beim Laden und läuft damit der Echtzeitanforderung zuwider. Daher ist fallbasiert zu entscheiden, für welche Daten die Implementierung des gesamten DWH-Stacks und die damit verbundene redundante Datenhaltung wirklich adäquat ist oder ob nicht einzelne Schichte lediglich virtuell vorhanden sind (Abbildung 1, 1a bis 1d).

In welchem Umfang und welche Zwischenstände persistiert werden müssen, gibt die Echtzeit-DWH-Architektur nicht vor. Diese Entscheidung wird vielmehr anhand gegebener technischer oder juristischer Anforderungen im Unternehmen, wie etwa Governance-Bestimmungen, Möglichkeit der Datenrückverfolgung (engl. data lineage) oder bestimmte Gesetzgebungen, gefällt. Eine gute Übersicht der Kriterien für Datenpersistenz in DWH-Systemen findet sich in [WK11].

Wird in den Datenquellen dasselbe Datenmodell wie im DWH verwendet, so kann die technische Harmonisierung der Daten entfallen und damit die Schicht der konsolidierten Basisdatenbank übersprungen werden (Fall 1b in Abbildung 1). Ebenso verhält es sich mit dem Bereich der Staging Area zur temporären Zwischenablage in das DWH eingebrachter Quelldatenbestände. Diese Schicht kann ebenfalls entfallen, falls keine Entkoppelung der Quellsystemen und kein Vorhalten der Daten zur Rekonstruktion des Datenverfeinerungsprozesses im Fehlerfall des DWHs notwendig ist (Fall 1a) und die Datenbestände direkt aus den Quellsystemen übernommen werden können. Des Weiteren sind Szenarien realisierbar (zum Beispiel „Dashboards“), in denen die Daten der Quellsysteme aufgrund ihrer Beschaffenheit keiner weiteren Transformationen bedürfen, so dass sie direkt an die Data Marts bzw. die Reporting-Schicht propagiert werden können (Fall 1c und 1d).

**Rolle von main-memory zentrischen Systemen:** DWH-Architekturen sind in der Praxis nicht nur auf die vier, in Abbildung 1 dargestellten Schichten beschränkt. Stattdessen ist in vielen Szenarien, zur Beschleunigung analytischer Data-Warehouse-Anfragen, eine vielstufige Data-Mart-Schicht, bestehend aus einer Reihe von einander abgeleiteter Materialized Views, zu finden. Diese unterliegen hohen Wartungskosten und verringern je nach Wartungsstrategie die Aktualität der Daten. Solche klassischen Datenbank-Tuning-Mechanismen sind einem In-Memory-DB-System obsolet. Sobald die Daten vollständig im Hauptspeicher vorgehalten werden können, sind auch datenintensive Operationen direkt auf den Basisdaten berechenbar. Somit kann durch den Einsatz von In-Memory-Techniken der DWH-Stack vereinfacht und die Kosten für das mehrfache Kopieren und Materialisieren von Daten reduziert werden. Zusätzlich dazu wird auch die Administration von Data-Warehouse-Systemen vereinfacht und damit ihre Gesamtbetriebskosten gesenkt.

Im Moment sind In-Memory-Datenbanken noch auf eine zusätzliche festplattenbasierte Backupplattform angewiesen, um Persistenz zu gewährleisten. Durch aktuelle Weiterentwicklungen bei den nichtflüchtigen Speichern (NVRAM, Non-Volatile Random-Access Memory) ist jedoch absehbar, dass in Zukunft diese ebenfalls redundante Datenhaltung entfallen und sich der DWH-Stack noch stärker konsolidieren wird.

### **Pull- vs. Push-basierte Verarbeitungssemantik**

Unter Echtzeitfähigkeit eines DWH-Systems wird die Fähigkeit verstanden, Daten schnell in das DWH zu integrieren und hinsichtlich der Datenanalysen aufzubereiten. Die zeitliche Verzögerung vom Eintreten eines Ereignisses in der Diskurswelt bis hin zur Datenbereitstellung im DWH oder im Data Mart wird als Datenverzögerung bezeichnet. Jede Schicht und jede Verarbeitungsstufe des klassischen DWH-Stacks wirkt sich negativ auf die Datenverzögerung aus. Diese kann, wie bereits skizziert, durch das Überspringen von Persistenzschichten minimiert werden.



Neben dieser push-basierten Verarbeitungsemantik von Daten ist für ein echtzeitfähiges DWH-System auch ein pull-basierter Datenzugriff notwendig und strukturell zu unterstützen (Abbildung 1, 2a bis 2d). Dies setzt voraus, dass die Nutzer des DWH die verschiedenen Verarbeitungszustände der Daten kennen und sich somit bewusst für noch nicht vollständig verfeinerte, dafür aber aktuelle Daten entscheiden können. Abhängig von der adressierten Datenschicht, vom Arbeitsbereich (Fall 2d in Abbildung 1) bis hin zu den Data Marts (Fall 2a), kann somit mit abnehmender Abhängigkeit vom Datenproduktionsprozess frühzeitig auf aktuelle Daten zugegriffen und natürlich im Rahmen des DWH mit anderen Datenbeständen aus dem Produktionsprozess verknüpft werden.

**Rolle von main-memory zentrischen Systemen:** Betrachtet man die pull-basierte Verarbeitung im Zusammenhang mit den im vorherigen Abschnitt vorgeschlagenen optionalen Persistenzschichten so resultiert dies in einer schrittweisen Aufhebung der Trennung zwischen OLTP- und OLAP-Systemen (z. B. die Kombination der Fälle 1b und 2b in Abbildung 1). Statt der bisher durch den Batchbetrieb getrennten Verarbeitung von Anfragen und Aktualisierungen ist im Kontext eines Echtzeit-DWHs eine gleichzeitige Verarbeitung von lesenden und schreibenden Transaktionen notwendig. Die Konsolidierung beider Welten in einem System und die daraus resultierenden gemischten Workloads erfordern eine leistungsfähige Anfrageverarbeitung wie sie nur durch In-Memory-Datenbanken bereitgestellt werden kann. Allerdings müssen weiterhin die klassischen ACID-Eigenschaften gewährleistet und innerhalb einer OLAP-Session ein konsistenter Datenzustand garantiert werden. Dabei sollen jedoch weder die schreibenden OLTP-Zugriffe, noch die komplexeren OLAP-Anfragen verzögert werden. Die In-Memory-Datenbank HyPer [KN11] gewährleistet dies beispielsweise durch einen hardware-unterstützten Replikationsmechanismus der jeweils einen konsistenten Snapshot der transaktionalen Daten vorhält. Die wesentliche Idee dieses Ansatzes besteht darin, dass die Snapshots durch Anwendung des Fork-Betriebssystemaufrufs auf den OLTP-Prozess erzeugt werden können. Durch die entsprechend gegebene Prozessorunterstützung bei der Verwaltung des virtuellen Speichers, sind hohe Durchsatzraten seitens des OLTP-Workloads aber auch geringe Antwortzeiten bei den OLAP-Anfragen erreichbar. Ein anderer Ansatz wird durch das Hyrise-System [GKP+09] implementiert. Mit Hilfe einer automatisierten vertikalen Partitionierung der Daten können gemischte OLTP/OLAP-Workloads jeweils optimal unterstützt werden. Dazu werden die Partitionsgrößen basierend auf einem Modell über die Cache-Misses und den entsprechenden Zugriffscharakteristika bestimmt. Schaut man sich die beiden Ansätze an so ist festzustellen, dass zur Unterstützung gemischter OLTP/OLAP-Workloads jeweils hybride Ansätze erforderlich sind um beiden Anfragentypen optimal zu unterstützen.

Diese Entwicklung hin zu hybriden Datenbankarchitekturen ist auch anhand der Produktstrategie der großen Datenbankhersteller absehbar. So existieren mit FlexStore von Vertica, der NewDB von SAP bzw. mit Oracle 11g Release 2 bereits Systeme die hybriden Datenstrukturen unterstützen.

## Reduktion der Batchgröße

Eine Reduzierung der Datenverzögerung kann neben der optionalen Persistierung und dem pull-basierten Datenzugriff vor allem durch die Optimierung der beteiligten Transformationsprozesse erreicht werden. Dazu sind die beteiligten ETL-Prozesse (Extraktion, Transformation und Laden) zu optimieren [TVS07] und mit ausreichenden Hardware-Ressourcen zu provisionieren. Diese Maßnahmen vorausgesetzt, ist eine entscheidende Verringerung der Datenverzögerung vor allem durch kürzere Ladezyklen zu erzielen (Abbildung 1, 3a bis 3c). Die klassischen Ladeprozesse basierend auf täglichen Batches („regular batches“, Fall 3a) werden dabei in Abhängigkeit der Aktualitätsanforderung der Anwendung durch kleinere Batchgrößen („micro batches“, Fall 3b) bzw. durch einen kontinuierlichen Strom von Aktualisierungen („trickle feed“, 3c) ersetzt. Das Data Warehouse der Zukunft ist damit in der Lage die komplette Informationsversorgung eines Unternehmens, vom klassischen ETL bis hin Datenstromsystemen, abzudecken. Dabei geht es nicht um die Ablösung existierender Techniken sondern vielmehr um die Schaffung einer generalisierten DWH-Architektur anhand derer entsprechend konforme Installationen instanziiert werden können. Mit der in Abbildung 1 dargestellten Verringerung der Aktualisierungsperiodizität müssen sich allerdings auch die auf die Verarbeitung von Massendaten spezialisierten ETL-Prozesse ändern. Dies schließt etwa die Entwicklung nicht-blockierender ETL-Operatoren wie zum Beispiel für die Erzeugung von Surrogatschlüsseln ein [PSV+08]. Der Strom kontinuierlicher Aktualisierungen kann darüber hinaus einer Ablaufplanung unterworfen werden. Dazu sind die Anforderungen der DWH-Anwender hinsichtlich der Aktualität der Daten zu ermitteln und die Aktualisierungen entsprechend zu priorisieren [TL09].

**Rolle von main-memory zentrischen Systemen:** Die Verringerung der Datengranularität für Aktualisierungen bis hin auf Tupelebene schließt die Verwendung effizienter Bulk-Load-Mechanismen aus wodurch Ladeprozess extrem verlangsamt werden. Um diesen Nachteil zu umgehen wurde mit RiTE [TPL08] eine Middleware entwickelt die den Ladeprozess anfragegetrieben optimiert. RiTE wurde unter Verwendung einer Hauptspeicherbasierten und schreiboptimierten Datenstruktur umgesetzt und ist dadurch in der Lage eine große Menge von Änderungen zwischen zu speichern und diese bedarfsgesteuert per Bulk-Load in das DWH laden. Mit Hyper [KN11] und Hyrise [GKP+09] existieren darüber hinaus schon hybride, d.h. sowohl schreib- als auch leseoptimierte Datenstrukturen. Wie sich solche Ansätze allerdings unter extrem hohen Aktualisierungsraten, etwa bei der Integration von Sensordatenströmen oder Click-streams in ein DWH verhalten bleibt zu untersuchen.

## Corporate Memory

Im Kontext von Echtzeit-DWH-Systemen wird häufig der Begriff der situativen Datenanalyse (engl. situational BI) verwandt. Die zentrale Eigenschaft der situativen Datenanalyse im Vergleich zu Echtzeit-DWH-Systemen besteht darin, dass der Umfang der Datenbereitstellung dem DWH-Betreiber a priori nicht bekannt ist. Dennoch muss es möglich sein, die Bedürfnisse der DWH-Anwender zeitnah zu befriedigen.

Derartige Szenarien kommen klassischerweise aus dem Bereich der wissenschaftlichen Datenanalyse, wobei erst schrittweise die unterschiedlichen Muster in den Daten aufgedeckt werden. Zur Unterstützung solcher Anforderungen ist eine zusätzliche Datenschicht, der sogenannte Corporate Memory (Abbildung 1, Fall 4) notwendig.

Ein Corporate Memory (CM) ist innerhalb der Referenzarchitektur so positioniert, dass die Daten in dem gleichen Umfang und in der gleichen Granularität wie sie von den Datenquellen zur Verfügung gestellt, gespeichert und mittelfristig archiviert werden. Getreu dem Motto „expect the unexpected“ werden zunächst unnötige Attribute nicht ausgeblendet, sondern proaktiv für zukünftige Anforderungsänderungen vorgehalten. Das CM erlaubt damit höchstmögliche Flexibilität bei sich ändernden und nicht vorhersehbaren Nutzeranforderungen und erfüllt gleichzeitig die Rolle einer „Recovery-Schicht“, aus welcher Transformationsprozesse im Fehlerfall wiederholt werden können. Neben den Daten aus den Quellsystemen werden im CM auch aufbereitete Datenbestände aus den anderen Schichten archiviert, so dass oftmals auch zwischen „Aquisition Memory“ und „Reporting Memory“ unterschieden wird.

**Rolle von main-memory zentrischen Systemen:** *In Hinblick auf den Umfang der Daten die in einem CM gespeichert werden, ist der Einsatz von In-Memory-Datenbanken für diesen Zweck innerhalb den nächsten Jahre, ausgeschlossen. Zusätzlich sind die Antwortzeiten bei einem CM weniger relevant so dass Hauptspeicherlösungen nicht erforderlich sind. Mittelfristig wird der CM daher als Backbone einer DWH-Landschaft, bestehend aus einer oder mehrere In-Memory-Datenbanken, fungieren. Schaut man jedoch ein wenig in die Zukunft und extrapoliert die Entwicklung der Hauptspeichergrößen insbesondere bei nichtflüchtigen Speichertechnologien ist eine technische Migration des CM in eine Hauptspeicherdatenbank durchaus denkbar.*

### **Schnelle Erschließung externer Datenquellen**

Neben der hinsichtlich Umfang und Detailgrad proaktiven Speicherung von Daten besteht im Kontext von Echtzeit-DWH des Weiteren die Anforderung bisher unbekannte, zum großen Teil externe Datenquellen zeitnah zu erschließen (Abbildung 1, Fall 5). Ein typischer Anwendungsfall ist etwa die Einbindung offener Daten (engl. open data) bei der Erstellung von Geschäftsanalysen. Klassische DWH-Architekturen sind aufgrund der Forderungen nach hoher Datenqualität und Konsistenz an wohldefinierte Entwicklungszyklen gebunden und daher zu wenig agil um situativen Bedürfnisse nach zusätzlichen Datenquellen zu befriedigen. Zur Unterstützung situativer Datenanalysen existieren diverse Methoden und Lösungsansätze sowohl auf organisatorischer als auch auf technischer Ebene. So müssen externe Datenquellen mit einer Dienstschnittstelle beschrieben sein, so dass eine einheitliche Abstraktionsschicht existiert, auf die dann aufgesetzt werden kann. Für weiterführende Betrachtungen sei an dieser Stelle auf [TL11] verwiesen.

**Rolle von main-memory zentrischen Systemen:** *Durch den Einsatz von hauptspeicherbasierten Systemen, können externe Datenquellen ohne weiteren Einsatz von Datenbank-Tuning-Mechanismen zur Nutzung bereitgestellt werden.*

*Durch den Wegfall technischer Hilfsstrukturen wie beispielsweise Indexstrukturen, die für einen effizienten Zugriff auf Plattensysteme notwendig sind, reduziert sich die „time to consumption“ beträchtlich. Anders formuliert ist eine situative Integration und Nutzung externer Datenquellen mit klassischen Systemansätzen nur sehr aufwändig realisierbar.*

## 4. Zusammenfassung

Eine klare Trennung zwischen operativen und analytischen Systemen, wie sie bisher aus unterschiedlichen, insbesondere auch technischen Gründen propagiert wurde, hat sich überlebt. Nur durch Kombination beider Welten ist eine echte Rückkopplung der aus DWH-Infrastrukturen gewonnenen Erkenntnisse zurück in die operativen Geschäftsprozesse möglich. Die wesentlichen Unterscheidungsmerkmale von Echtzeit-Data-Warehouse-Systemen im Vergleich zur klassischen DWH-Architektur bestehen zum einen in kürzeren Aktualisierungszyklen und zum anderen durch die Möglichkeit, dass Zugriffe auf die Datenbestände nicht nur auf die Data-Marts beschränkt sind, sondern entlang des Datenverfeinerungsprozess in jeder Prozessstufe ausgeführt werden können. Die gleichzeitige Verarbeitung von lesenden und schreibenden Transaktionen stellt hohe Anforderungen an die Leistungsfähigkeit eines Datenbanksystems und ist nur durch hauptspeicherbasierte Techniken zu gewährleisten. In-Memory-Datenbanken spielen somit eine Schlüsselrolle bei der Umsetzung der Vision eines DWH 3.0. Weiterhin gilt, dass dem Aspekt der situativen Datenanalyse („Self-service-BI“) bereits beim Entwurf des Systems Rechnung sowohl aus technischer als auch aus organisatorischer Perspektive getragen werden muss. Dem „Corporate Memory“ kommt im „DWH der Zukunft“ dabei eine wesentlich prominentere Stellung zu als in den klassischen Ansätzen.

## Literaturverzeichnis

- [BG09] Bauer, A.H., Günzel, H.H.: Data-Warehouse-Systeme. dpunkt, Heidelberg (2009).
- [GKP+09] Grund, M., Krüger, J., Plattner, H., Zeier, A., Cudre-Mauroux, P., Madden, S.: Hyrise: a main memory hybrid storage engine. In: Proc. VLDB Endow., vol. 4, pp. 105–116, November 2010.
- [ISN08] Inmon, W.H., Strauss, D., Neushloss, G.: DW 2.0: The Architecture for the Next Generation of Data Warehousing. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (2008).
- [KN11] Kemper A., Neumann T.: HyPer – A Hybrid OLTP&OLAP Main Memory Database System Based on Virtual Memory Snapshots. In: ICDE 2011.
- [KR02] Kimball, R., Ross, M.: The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling, 2nd edn. John Wiley & Sons, Inc., New York, NY, USA (2002).
- [Le02] Lehner, W.: Datenbanktechnologie für Data-Warehouse-Systeme: Konzepte und Methoden, 1. Auflage edn. dpunkt-Verlag (2003).
- [PSV+08] Polyzotis, N., Skiadopoulos, S., Vassiliadis, P., Simitsis, A., Frantzell, N.E.: Meshing streaming updates with persistent data in an active data warehouse. IEEE Trans.Knowl. Data Eng. 20(7), 976-991 (2008).

- [SBC+07] M. Stonebraker, C. Bear, U. Cetintemel, M. Cherniack, T. Ge, N. Hachem, S. Harizopoulos, J. Lifter, J. Rogers, and S. B. Zdonik. One size fits all? Part 2: Benchmarking studies. In Third Biennial Conference on Innovative Data Systems Research (CIDR 2007), pages 173-184, 2007.
- [SMA+07] M. Stonebraker, S. Madden, D. J. Abadi, S. Harizopoulos, N. Hachem, and P. Helland. The end of an architectural era: (it's time for a complete rewrite). In VLDB '07: Proceedings of the 33rd international conference on Very large data bases, pages 1150-1160. VLDB Endowment, 2007.
- [TBL09] Thiele, M., Bader, A., Lehner, W.: Multi-objective scheduling for real-time data warehouses. In: In Proceedings der 12. GI-Fachtagung für Datenbanksysteme in Business, Technology und Web, pp. 307-326. GI (2009).
- [TL11] Thiele, M., Lehner, W.: Real-Time BI and Situational Analysis (Chapter). In BUSINESS INTELLIGENCE APPLICATIONS AND THE WEB: MODELS, SYSTEMS AND TECHNOLOGIES, IGI-Global, Editors: Marta Zorrilla, Jose-Norberto Mazón, Óscar Ferrández, Irene Garrigós, Florian Daniel and Juan Trujillo, ISBN-13: 9781613500385.
- [TPL08] Thomsen, C., Pedersen, T.B., Lehner, W.: Rite: Providing on-demand data for right-time data warehousing. In: ICDE '08: Proceedings of the 2008 IEEE 24th International Conference on Data Engineering, pp. 456-465. IEEE Computer Society, Washington, DC, USA (2008).
- [TVS07] Tziovara, V., Vassiliadis, P., Simitsis, A.: Deciding the physical implementation of ELT workflows. In: DOLAP '07: Proceedings of the ACM tenth international workshop on Data warehousing and OLAP, pp. 49-56. ACM, New York, NY, USA (2007).
- [WK11] Winsemann, T., Köppen, V.: Kriterien für Datenpersistenz bei Enterprise Data Warehouse Systemen auf In-Memory Datenbanken. In: In Proceedings of the 23. GI-Workshop on Foundations of Databases, pp. 97-102. GI (2011).

# Realisierung einer serviceorientierten Business Intelligence Architektur anhand von In-Memory-Technologien

Marco Pospiech, Carsten Felden

Professur für Informationswirtschaft/Wirtschaftsinformatik

TU Bergakademie Freiberg

Lessingstraße 45, 09599 Freiberg

marco.pospiech@bwl.tu-freiberg.de

carsten.felden@bwl.tu-freiberg.de

**Abstract:** Business Intelligence (BI) verspricht eine verbesserte Entscheidungsfindung. [Gl08] Um den veränderten Unternehmensanforderungen gerecht zu werden, ist das traditionelle Konzept zu erweitern. Infolgedessen sind Real Time, Active, Operational, Embedded oder prozessorientierte BI entstanden, um den steigenden Bedürfnissen nachzukommen. Diese verlangen jedoch eine technologische Umsetzung. In diesem Zusammenhang gehen aktuelle Ansätze dazu über, den wechselnden BI-Paradigmen durch einen serviceorientierten Ansatz zu begegnen [Di08]. Störend erweist sich hierbei, dass dieses Konzept dem Datenaufkommen nicht gewachsen ist. [Vo08] Der vorliegende Beitrag adressiert diese Lücke, indem unter Verwendung der Referenzmodellierung eine serviceorientierte Business Intelligence (SoBI) erarbeitet wird, die mit Hilfe von In-Memory-Technologien den aufkommenden Bedürfnissen gerecht werden soll. Als erstes Artefakt entsteht ein Konzept, welches für spätere Realisierungen die entsprechende Grundlage bietet.

## 1 Einleitung

Die Anforderungen an den Produktionsfaktor Information sind über die letzten Jahre gestiegen. [CG06] In diesem Zusammenhang erfolgt seit 1958 eine evolutionäre Entwicklung, des Business-Intelligence-(BI)-Konzeptes. Seitdem sind neue Anforderungen aufgekommen, die im traditionellen Ansatz nicht relevant waren und nun einer Lösung bedürfen. [Ma11] Fokussierte BI früher strategische und taktische Fragestellungen, rücken heute zusätzlich operative Belange ins Spektrum. [KB09] Insbesondere weitet dies den Anwenderkreis, die Problemstellung und die Forderung nach Echtzeitentscheidungen aus [Wh06]. Auch gehen diese Veränderungen mit einer Prozessorientierung [Da09] und einer unternehmensweiten Datenintegration [Ma06] einher. Zusätzlich sind Bestrebungen zu beobachten, deren Bemühung die Einbettung von BI-Funktionalitäten in Prozessen [Vo08] oder Fremdsystemen [NO09] ist. Viele Problematiken gehen aus mangelnder Systemintegration und überlasteten Infrastrukturen hervor. [DB06; Ma11] Ziel dieses Beitrages ist ein Ansatz, der den erweiterten Anforderungen an eine umfassende BI gerecht wird.

Aktuelle Bemühungen gehen dazu über, den wechselnden BI-Paradigmen durch einen serviceorientierten Ansatz zu begegnen. Infolgedessen sowohl Real Time, Active [Ng05], Embedded [Pa08] und Operational BI [Go06], als auch die unternehmensweite Datenintegration [Mü10] und Prozessorientierung [Ng05] verwirklicht werden kann. Schwachstelle des Ansatzes ist bis dato die eingeschränkte Fähigkeit, dem hohen Datenaufkommen in Echtzeit gerecht zu werden. Ergänzend kommt hier die zu erwartende Überlastung des Service-Busses oder der beteiligten Systeme hinzu. [Vo08] In diesem Zusammenhang bildet die Kombination von SoBI und In-Memory-Technologien einen vielversprechenden Ansatz. Allerdings ist dies in der Literatur bisher nicht umfassend adressiert, wobei dieser Beitrag einen Diskussionsansatz in diesem Umfeld liefert.

Zur Bearbeitung der Thematik legt Kapitel 2 den gegenwärtigen Forschungsstand der Thematik SoBI dar, um auf Grundlage dessen Defizite in den bestehenden Ansätzen aufzuzeigen. Um die Umsetzung des Konzeptes voranzutreiben, behandelt Kapitel 3 die Referenzmodellierung als geeignete Herangehensweise und die Erarbeitung einer SoBI-Referenzarchitektur auf Grundlage der In-Memory-Technologie. Abschluss findet der Beitrag in Kapitel 4.

## **2 Status Quo**

Dynamische Umgebungen zwingen Unternehmen, die richtigen Informationen zum richtigen Zeitpunkt am richtigen Ort und zum richtigen Zweck bereitzustellen. [Gl08] Auf deren Grundlage sind zeitnah Entscheidungen zu treffen. Die Herausforderung besteht in einer umfassenden Systemintegration in Echtzeit. Hierbei ist die unternehmensweite Systemzusammenführung komplex. [Ma06] Regelmäßig sind 80 Prozent aller Datenintegrationsvorhaben der Datenzuführung in BI- und Data Warehouse-Systemen (DWH) gewidmet [Ru09], von denen die meisten durch handkodierte Punkt-zu-Punkt-Verbindungen verknüpft sind. [Ma11] Hierdurch wird die Integration von neuen Vorhaben, wie Active, Embedded, Operational oder prozessorientierter BI erschwert und ein Verlust an Flexibilität in Kauf genommen. Ein Verlust der insbesondere bei Reorganisationsvorhaben zum Vorschein kommt. Problematisch erweist sich zudem die Datenversorgung in Echtzeit. Gemäß TDWI gaben 92 Prozent aller Befragten an, Real Time BI bis 2013 anzustreben. [Ru10] Kritisch wirkt in diesem Zusammenhang das steigende Datenvolumen, das nachweislich die Ladeoperationslatenz verlängert. [EW03] So verwenden nahezu 91 Prozent der Unternehmen langwierige Stapel-Verarbeitungsverfahren. [Ru11] Demnach besteht die Herausforderung in der Etablierung von neuen Techniken, die eine Real Time Datenintegration in einer kosteneffizienten Weise ermöglichen und somit BI zu Real-Time-Analysen befähigen. Eine mögliche Lösung ist die Kombination einer serviceorientierten Architektur (SOA), In-Memory-Technologien und BI.

Hierbei ist SOA als „... a way of designing and implementing enterprise applications that deals with the intercommunication of loosely coupled, coarse grained (business level), reusable artifacts (services). Determining how to invoke these services should be through a platform independent service interface ...“ [WH04], zu verstehen. Infolgedessen ist eine Verschiebung weg von der monolithischen BI-Lösung, hin zur vernetzten IT-Architektur, zu verzeichnen [Kl08]. Der aufkommende Nachrichtenverkehr und die einhergehenden Zugriffszeiten können anhand von In-Memory-Technologien verbessert werden [CB08].

Der wesentliche Integrationsansatz einer SoBI ist der Service-Bus. Jener bildet das zentrale Element, welcher BI-Komponenten den universelle Zugriff zu allen an den Service-Bus angeschlossenen Systemen erlaubt und folglich jeglichen System den Zugang zur BI. [Vo08] Die wesentlichen Aufgaben bestehen in der Nachrichtenübermittlung, der Datentransformation und im intelligenten Routing von Nachrichten. [FZ09] Gemäß Dittmar beinhaltet der Service-Bus sowohl die asynchrone ereignisgesteuerte Kommunikation mittels Publish-and-Subscribe, als auch die synchrone Service-Kommunikation, welche einem Anfrage-Antwort-Model folgt. [Di07] Hierbei fungiert der Bus als Mittler zwischen Service-Nachfrager und -Anbieter und ermöglicht die Kommunikation zwischen Service und Ereignis [Za07]. Somit liegen alle Geschäftsereignisse im Service-Bus vor und lassen sich in Echtzeit verarbeiten. [Ne09]

Services repräsentieren Endpunkte, welche auf Ereignisse reagieren oder diese hervorbringen [Vo08] In diesem Sinn ermöglicht die Kombination beider Kommunikationsmodelle eine zeitnahe Reaktion auf eintretende Ereignisse, indem beispielsweise mehrere Services zeitgleich und asynchron ausgelöst werden können. [Ne09] Nach Dinter und Stroh sind im Kontext von SoBI BI-Funktionalitäten in Services zu kapseln. So kann BI als Service-Nachfrager oder Service-Anbieter fungieren. [DS09] Analytische Services sind gekapselte Module, die analytische Geschäftslogik bereitstellen und entsprechen klassischen BI-Werkzeugen [Ma11; Di08]. Die Grundlage für Analysen bilden Software-Services [Wi08; Gl08], welche aktuelle Daten aus den operativen, taktischen und strategischen Bereichen bereitstellen. In diesem Zusammenhang sind diese in Datenzugriffs-, Datentransformations- und Infrastruktur-Services zu unterteilen und entsprechen vornehmlich den gekapselten ETL-Funktionalitäten [Di08]. Der Datenzugriffs-Service beinhaltet vier Basisoperationen: Schreiben, Lesen, Update und Löschen. Hierdurch wird der Service befähigt, sowohl Systeme auszulesen als auch zu füllen. [Ma11] Datentransformations-Services repräsentieren die verkapselte Transformationsphase des ETL-Prozesses. Ergänzend zu den vorgestellten Services bestehen Querschnittaufgaben, die durch Infrastruktur-Services realisiert werden. [Di08] Diese Services beinhalten Datensicherheits- und Datenschutzfunktionen sowie Aspekte des Datenqualitäts-, Master-Data- und Meta-Data-Management. [Go06] In diesem Zusammenhang verspricht SoBI die Einbindung von BI-Services in Prozesse und Anwendungen, vermeidet redundante Implementierungen, vollzieht eine problemlose Integration, verbessert die Skalierbarkeit und erlaubt eine offene Kommunikation zwischen allen beteiligten Komponenten. [FZ09; Di08] Problematisch erweist sich das Paradigma der Real Time BI.



Um eine Datenbereitstellung in Echtzeit zu ermöglichen, benötigt es des Einsatzes von Publish-and-Subscribe-Mechanismen. Bei diesen erzeugen datenhaltende Systeme bei Datenveränderungen Ereignisse, welche durch eine Publisher-Komponente an den Service-Bus übergeben werden. [Vo08] Kritisch ist zu beachten, dass Services für kleine Datenmengen konzipiert sind und die aufkommenden Volumina einer BI-Umgebung nicht gerecht werden. So beinhalten Services zumeist immense Beschreibungsstrukturen, welche die eigentliche Nachricht vergrößern und zu Leistungsverlusten führen. [FZ09] Demnach ist eine Substitution des klassischen ETL-Prozesses und der damit verbunden Realisierung von Real Time BI anzuzweifeln. [Di08; Vo08]

In diesem Zusammenhang lässt sich die Grundlagenforschung von [Ch11; Di08; Vo08] nutzen, um diese im Rahmen dieses Beitrages zu verfeinern. Infolgedessen besteht die Notwendigkeit der Erarbeitung eines Konstrukts, welches die vorgestellten Theorien vereint, den neuzeitigen BI-Anforderungen gerecht wird und eine Realisierung ermöglicht. Dabei bietet sich die Referenzmodellierung als Methode an. [DE98]

### 3 Referenzmodell zur serviceorientierten Business Intelligence

Die Referenzmodellierung ist eine Technik, die eine formale Beschreibung von Wissen erlaubt und unter anderem in der Systementwicklung eingesetzt wird. [DE98] In diesem Kontext nennt Hars Allgemeingültigkeit, Anpassbarkeit und Anwendbarkeit als zentrale Referenzmodellanforderungen. [Ha94] Ein möglichst abstraktes Modell lässt sich somit für eine Vielzahl von Unternehmen nutzen. In diesem Zusammenhang verwendet diese Arbeit das Klassendiagramm der Unified Modeling Language (UML), das in der Referenzmodellierung eine große Akzeptanz genießt. So ermöglicht diese Darstellungsform die Abbildung von Beziehungen, Attributen und Operationen von Elementen, bietet Erweiterungsmöglichkeiten und ist als Illustration geeignet. [FL04]

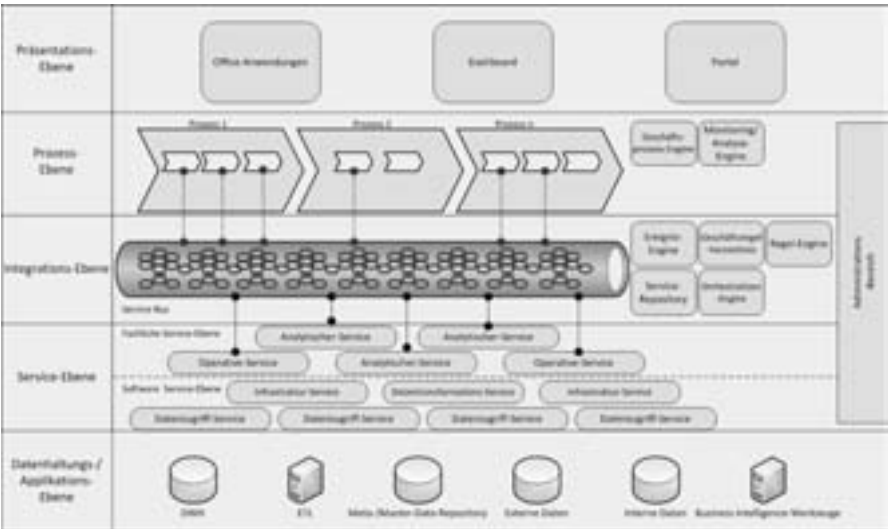


Abbildung 1: Serviceorientierte Business Intelligence Referenzarchitektur

In Abbildung 1 erfolgt eine semistrukturale Darstellung der SoBI-Referenzarchitektur, die in weiteren Schritten durch schematische Klassendiagramme verfeinert wird. Aus pragmatischen Gründen wird allerdings von einer Darstellung der Datenhaltungs-/Applikations- und Präsentations-Ebene mittels Klassendiagrammen Abstand genommen.

Das vorgeschlagene Referenzmodell orientiert sich nicht an der klassischen BI-Architektur. In diesem Zusammenhang diene eine generische SOA-Architektur als Orientierung [Ch11; Vo08; Mu08]. Die Datenhaltungs-/Applikations-Ebene fungiert als zentrales Fundament für die nachfolgenden Ebenen. In diesem Zusammenhang sind die Klassen der Ebene als Informationssystem zu verstehen, welche als Teilnehmer die Rolle eines Service-Nachfrager bzw. Service-Anbieter einnehmen können. Des Weiteren verfügt jedes Informationssystem über Operationen, die das Empfangen und Senden von Ereignissen ermöglichen. Ein Ereignis ist ein *ModelElement* und beinhaltet den *EreignisParameter* als Attribut. In diesem Zusammenhang entspricht *SetEreignisParameter* einer abstrakten Operation, die zum Ausdruck bringt, dass jede Operation zu einem Ereignis führt. Die Ereignis-Engine ist Bestandteil des Service-Busses und sendet bzw. empfängt Ereignisse der Informationssysteme. Das DWH verliert seine Rolle als zentrale Datenhaltungskomponente und wird als eine von vielen Komponenten der Datenhaltungs-/Applikations-Ebene aufgefasst. [Di07] Infolgedessen erscheint es nachteilig, das klassische BI-Architekturkonzept weiter zu verfolgen, da das DWH in einer SoBI nicht länger als monolithische Säule vorliegt, sondern in die IT-Infrastruktur eingebettet ist. [Ma11] Demnach müssen historische Daten mit Echtzeit-Informationen aus den operativen Systemen kombiniert, synchronisiert und in Prozesse eingebracht werden. [Fi07] Eine redundante und verteilte Speicherung kann zu individuellen Applikationsterminologien führen, was wiederum in Inkonsistenzen und Dubletten endet. Dies erfordert ein zentrales Meta- und Master-Data-Management, das insbesondere im Zusammenhang mit SOA unerlässlich wird und über Datenzugriff-Services und Service-Bus angesprochen wird. [Ma11]

Die Service-Ebene spiegelt alle lose gekapselten Funktionalitäten der Applikations- und Datenhaltungsebene wider, die den Anforderungen der systemübergreifenden Prozesse entsprechen [Vo08]. Demnach ist die Sinnhaftigkeit zu hinterfragen, sodass insbesondere Wiederverwendbarkeit zu prüfen ist. [Mü10; Di08] In diesem Zusammenhang konnte dieses Merkmal für Datenzugriff-, Datentransformations- [Go06], Infrastruktur- [Di08], Analytische- [Mu08] und Operative-Services [Me10] nachgewiesen werden. [Mü10] Wie in Abb. 2 zu sehen ist, ist zunächst das Rollenkonzept einer SOA in die Referenzarchitektur zu integrieren. Demnach kann ein Teilnehmer die Rolle eines Service-Nutzers oder Service-Anbieters einnehmen. In diesem Zusammenhang stellt der Anbieter dem Nutzer einen Service zur Verfügung. Hierfür wird der Service-Vertrag benötigt, der Bestandteil des Services ist und die für den Nachrichtenaustausch erforderliche Service-Beschreibung enthält. Nachrichten erweitern im Verständnis der Arbeit das Modellelement *DataTyp* und beinhalten den Container, welcher die Nachricht beherbergt und empfangen und gesendet werden kann.

Der Service selbst stellt eine technische Aktivität dar. Im Verständnis von Winter, ist die Service-Klasse in fachliche- und Software-Services zu spezialisieren. [Wi08] Anhand der enthaltenen Service-Operationen lassen sich weitere Spezialisierung vornehmen. Die Analytischen-Services bilden das Herzstück des vorgestellten Referenzmodells und beziehen den transformierten Inhalt der unterliegenden Services. Sie beinhalten analytische- und grafische Operationen. Diese sind nicht auf eine Datenquelle beschränkt, sondern greifen auf verschiedene Systeme zurück. Infolgedessen ist eine Kombination von taktischen und strategischen Daten aus dem DWH, sowie operativen Daten möglich. [Ma11] Zudem sind die operativen Systeme im Rahmen der Referenzarchitektur serviceorientiert zu gestalten [Vo08]. Dadurch werden deren Funktionalitäten als Services verfügbar. Die Service-Einbindung findet über die Integrations-Ebene statt. Daraus folgt, eine Trennung zwischen Geschäfts- und Prozesslogik sowie der fachlichen Konzeption und technischer Implementierung [Di07].

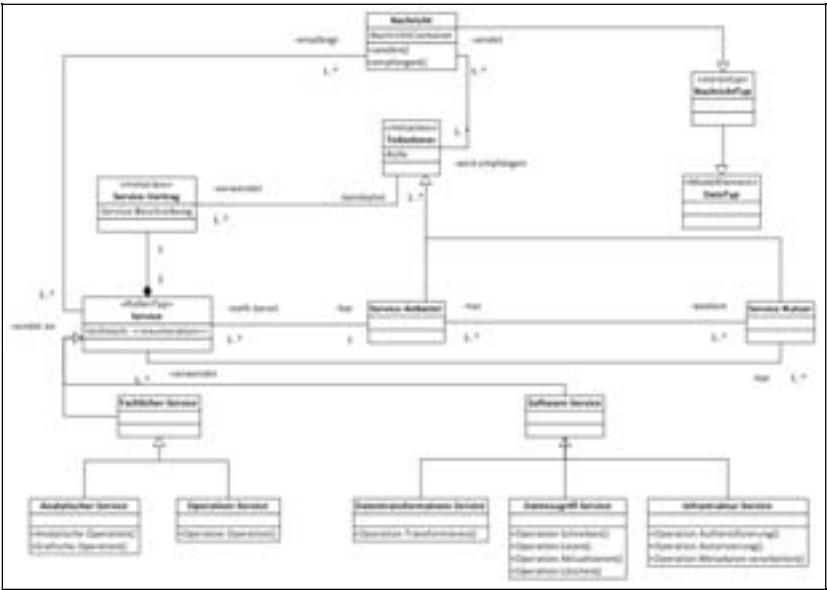


Abbildung 2: Service-Ebene

Die Integrations-Ebene in Abb. 3 realisiert die Koordination aller Services und Ereignisse im Gesamtsystem. Der Service-Bus besteht aus Service-Repository, Orchestrations-Engine, Regel-Engine, Ereignis-Engine, Geschäftsregelverzeichnis, Administrationsbereich und dem unterliegenden SOA-Grid, letzteres soll später betrachtet werden. Das Service-Repository verwaltet und veröffentlicht alle im System befindlichen Service-Beschreibungen, die in Form von Service-Verträgen vorliegen [Vo08]. In diesem Zusammenhang trifft das Service-Repository in Kooperation mit der Orchestrations-Engine eine Service-Auswahl auf Grundlage der Service-Beschreibung und stellt diese der Orchestrations-Engine bereit. Die Orchestrations-Engine nutzt die Operation Orchestrieren, um eine Service-Abfolge auszuführen. Hierbei eine Abfolge aus verschiedenen Service-Arten bestehen kann und in Fremdsysteme oder Prozesse eingebettet werden kann (Embedded BI).

Diese Abfolgen sind im Verständnis der Arbeit in Geschäftsprozesse und rein technische Prozesse zu unterteilen. In diesem Zusammenhang ermöglicht die Orchestrations-Engine die Orchestration rein technischer Prozesse. [Mu08] Zusätzlich obliegt der Orchestrations-Engine die Statusverwaltung, Protokollierung und Überwachung der Abfolgen. Angesprochen wird die Orchestrations-Engine über Ereignisse, die von der Ereignis-Engine ausgelöst werden. Hierbei basiert die Ereignis-Engine auf Publish-and-Subscribe und empfängt und verarbeitet Ereignisse aus allen Ebenen, die an registrierte Konsumenten versendet werden. Des Weiteren verfügt die Ereignis-Engine über analytische Operationen. [Vo08]

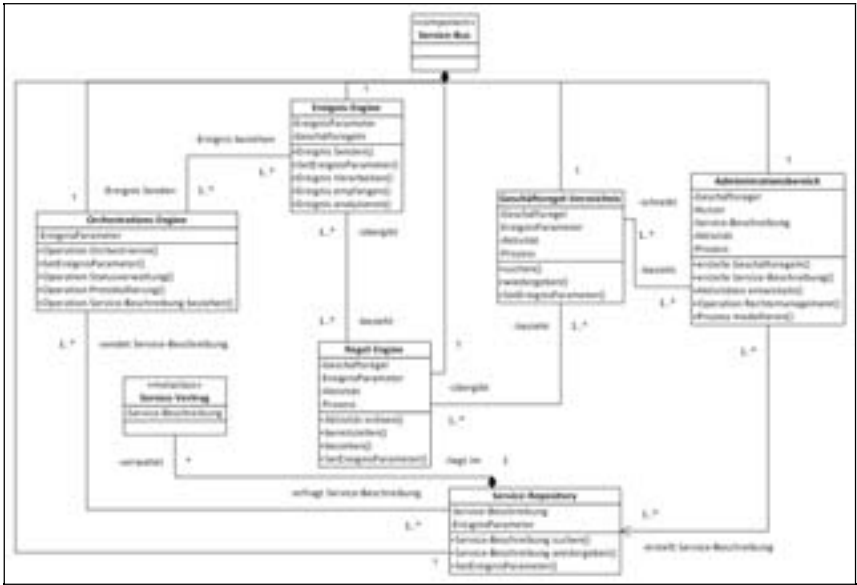


Abbildung 3: Integrations-Ebene

Im Weiteren sind definierte Geschäftsregeln erforderlich, die von der Regel-Engine bezogen werden. In diesem Zusammenhang ist eine zeitgleiche Verarbeitung von mehreren Ereignissen denkbar, um kausale, temporale oder räumliche Beziehungen zwischen Ereignissen aufzuspüren, [BD10] um so prädiktiv Problemszenarien zu identifizieren. Zusätzlich erfolgt die Versorgung von geschäftsprozessrelevanten Ereignissen über die Monitoring/Analyse-Engine, welche ebenfalls Aktionen zur Folge haben. Diese Funktionalitäten folgen der *Event-Condition-Action*-Regel und erlauben eine automatisierte Entscheidungsfindung und Ausführung. [Vo08] Die Regel-Engine stellt Geschäftsregeln bereit, die vom zentralen Geschäftsverzeichnis bezogen werden. Dieses speichert unternehmensweit Geschäftsregeln, sodass eine redundante Implementierung von Geschäftsregeln verhindert werden kann und deren Wiederverwendbarkeit steigt. Im Verständnis dieses Beitrages sind Geschäftsregeln als Fachwissen, Managementpolitik [Ma11] und Entscheidungslogik für Prozesse aufzufassen [Di07]. Somit wird die Prozesslogik von der Entscheidungslogik getrennt. [Ma11] Zusätzlich verwaltet das Geschäftsregelverzeichnis bereits modellierte Prozesse, die von der Regel-Engine abfragbar sind.

Der Administrations-Bereich ist die verwaltende Komponente der SoBI-Referenzarchitektur. Demnach werden die Geschäftsregeln und Service-Beschreibungen erstellt und vom Geschäftsregelverzeichnis bzw. Service-Repository bezogen. In diesem Zusammenhang erfolgen die Service-Entwicklung und die Definition von menschlichen Tätigkeiten. Zudem verfügt der Administrationsbereich über Möglichkeiten, Prozesse zu modellieren, die im Geschäftsregelverzeichnis abzulegen sind.

Das zentrale Element der Prozess-Ebene (s. Abb. 4) ist der Prozess. Dieser besteht aus Aktivitäten, welche technisch oder manuell sein können. Liegt eine technische Aktivität vor, handelt es sich um einen Service. Eine manuelle Aktivität ist von einem menschlichen Aufgabenträger auszuführen. Der zweite Bestandteil eines Prozesses sind Geschäftsregeln. In diesem Zusammenhang ordnet die Regel-Engine die Aktivitäten anhand von Geschäftsregeln. Das Resultat dieser Ordnung ist ein Prozess. Infolgedessen ist festzustellen, um welche Prozessart es sich handelt. Im Verständnis des Beitrages wird zwischen technischen Prozess und Geschäftsprozess unterschieden. Demnach kann ein Geschäftsprozess aus menschlichen (manuellen) und technischen Aktivitäten bestehen, hingegen sich ein technischer Prozess ausschließlich aus technischen Aktivitäten zusammensetzt. Dem Folgend, verantwortet die Orchestrations-Engine die Orchestrierung technischer Prozesse. Die Geschäftsprozess-Engine verantwortet die Ausführung von Geschäftsprozessen. Demnach wird eine Folge von manuellen und technischen Aktivitäten realisiert. Im Falle eines Geschäftsprozesses kooperiert die Orchestrations-Engine mit der Geschäftsprozess-Engine, sodass die technischen Aktivitäten des Geschäftsprozesses über Services bereitgestellt werden.

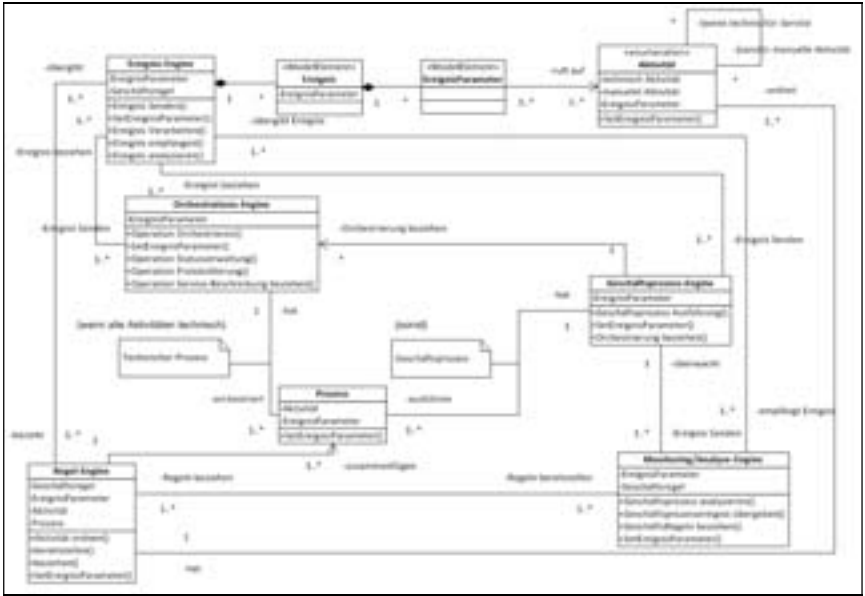


Abbildung 4: Prozess-Ebene

Ausgelöst werden Prozesse durch Vorfälle in der Ereignis-Engine. Demnach sendet die Ereignis-Engine ein Ereignis an die Regel-Engine, welche mit Hilfe des Geschäftsregel-Verzeichnisses passende Geschäftsregeln sucht. Infolgedessen entsteht ein Prozess, der von der Orchestrations-Engine orchestriert oder von der Geschäftsprozess-Engine ausgeführt wird. Alternativ kann der Prozess vorab im Administrations-Bereich modelliert werden. Infolgedessen die Regel-Engine den Prozess vom Geschäftsregel-Verzeichnis bereitstellt und übergibt. In diesem Verständnis wird der Prozess mittels Ereignis in der Orchestrations- oder Geschäftsprozess-Engine ausgelöst. Die Monitoring/Analyse-Engine arbeitet eng mit der Geschäftsprozess- und Ereignis-Engine zusammen und dient der Überwachung aller Geschäftsprozesse. Demnach analysiert die Engine kontinuierlich Geschäftsprozessereignisse in Echtzeit. [DB06] Die benötigten Prozessmeldungen bezieht die Monitoring/Analyse-Engine aus der Geschäftsprozess-Engine. Hierbei lösen jegliche Prozessveränderungen in der Monitoring/Analyse-Engine Ereignisse aus, welche der Ereignis-Engine übergeben und an etwaige Konsumenten weitergereicht werden. Besondere Prozesszustände oder Vorkommnisse, haben ebenfalls die Versendung eines Ereignisses zur Folge, auf welches die Ereignis-Engine reagieren kann. [Ma11] Hierzu sind Geschäftsregeln erforderlich, die von der Regel-Engine bezogen werden. Dadurch erfolgt eine automatisierte Entscheidung in kürzester Zeit. [Za07] Infolgedessen ermöglicht die Monitoring/Analyse-Engine der BI eine Echtzeitanalyse und Überwachung von kritischen Prozesskennzahlen. In diesem Zusammenhang die SoBI Referenzarchitektur die entsprechenden Geschäftsfunktionen in Form von Services aus der Applikations- und Datenhaltungsebene nutzbar macht und entsprechend der Prozesslogik orchestriert.

Die oberste Ebene bildet die Präsentations-Ebene. Diese ermöglicht eine flexible Einbindung von analytischen Informationen durch Anwendung unterschiedlicher Visualisierungsmöglichkeiten. Die Grundlage hierfür schaffen Analytische-Services, welche grafischen Darstellungen in Portale, Dashboards oder Office-Anwendungen einfließen lassen. [Vo08] In diesem Zusammenhang erben die Klassen Portal, Dashboard und Office-Anwendung vom Informationssystem. Demnach kann der Benutzer mittels Ereignis über Vorkommnisse informiert werden bzw. seine Entscheidung direkt an die Ereignis-Engine senden.

Die Real Time Datenversorgung im Umfeld von SoBI stellt ein Problem dar. Neuste Erkenntnisse offerieren in diesem Zusammenhang die Verwendung eines erweiterbaren In-Memory SOA-Grids [CB08]. In-Memory ist hierbei als Ansatz zu verstehen, der entgegen des traditionellen Konzeptes, große Datenvolumina im Arbeitsspeicher vorhält, anstatt sie in persistenten Speichern abzulegen [PH11]. Im Kern ist das SOA-Grid als eine Ansammlung von miteinander verbundenen Software-Prozessen zu verstehen, die zusammen in einem gemeinsamen In-Memory-Netzwerk Daten in sogenannten Nodes speichern und pflegen. Diese Nodes sind sich über den Status der angrenzenden Nodes bewusst und balancieren das Datenaufkommen zwischen den verbleibenden, sodass stets ein Gleichgewicht gewährleistet wird (Load Balancing). Dieser Ausgleich geschieht zur Echtzeit.

Zudem gewährleistet das SOA-Grid im Falle eines Datenausfalles die Konsistenz, indem die beschädigte Primary Node von einer Backup Node ersetzt wird, welche zur neuen Primary Node mutiert. Infolgedessen BI ein performanter In-Memory Zugang gewährleistet wird, der jederzeit horizontal erweiterungsfähig ist und eine verbesserte Verfügbarkeit verspricht. [CB07]

Zusätzlich leistet diese Lösung einen schnelleren Datenzugriff, als traditionelle Lösungen, in denen auf persistente Speicher zurückgegriffen wurde [CB08]. In diesem Zusammenhang bildet das SoBI-Grid einen wichtigen Bestandteil des Service-Busses (s. Abb. 1). Hierbei arbeitet das Netzwerk eng mit der Ereignis-Engine zusammen, was am Beispiel von Real Time BI veranschaulicht werden soll. Dabei empfängt die Ereignis-Engine ein Ereignis, welches durch eine Datenveränderung in einer der Ebenen erzeugt wurde. Auf Grundlage dessen wird mit Hilfe der Regel-Engine und Orchestrations-Engine ein Datenbewirtschaftungsprozess angestoßen der mit Hilfe eines Datenzugriffs-Service die Datenveränderungen im SOA-Grid einpflegt. In einem weiteren Schritt wird ein eindeutiger Referenz-Identifizier erzeugt, welcher auf die datenhaltende Node verweist und als Ereignis allen Konsumenten zugeht. Jeder Service der das Ereignis empfängt, bezieht je nach Bedürfnis die Daten in Echtzeit aus dem Grid. Infolgedessen Analytische-Services jederzeit der Zugang zu hochbrisanten Information gewährt wird und der Datentransfer auf das Mindeste reduziert wird. Zusätzlich bietet diese Lösung die Möglichkeit Datenquellen wie das DWH asynchron zu Aktualisieren, um Konsolidierungen zu ermöglichen. In diesem Zusammenhang löst die Befüllung des SOA-Grids einen weiteren Datenbewirtschaftungsprozess aus, welcher Echtzeitdaten in das DWH einpflegt.

Neben den datenhaltenden Funktionalitäten bietet das In-Memory-Netzwerk Möglichkeiten Geschäftslogik zu verarbeiten. [CB07] Demnach erfolgt die Ausführung des Service in der Node selbst, sodass der Service mit verminderten Nachrichtenverkehr und beschleunigter Zugriffszeit transformierende oder analytische Aufgaben auf datenhaltenden Nodes verwirklichen kann. Infolgedessen wird SoBI eine erhöhte Leistungsfähigkeit beschert. Hierbei auf Service Oriented Programming (SOP) verwiesen werden soll, welches den Bedürfnissen von In-Memory Services nachkommt. [Si02]

## 4 Fazit

Zielstellung des Beitrages war die Erarbeitung einer serviceorientierten SoBI-Referenzarchitektur und die Lösung der innewohnenden Real Time Problematik. In diesem Zusammenhang wurden sowohl die aufgezeigten BI-Anforderungen erfüllt, als auch ein Ansatz präsentiert, welcher Real Time BI im Umfeld von SoBI ermöglicht.

Zusammenfassend lässt sich festhalten, dass die Verwendung eines In-Memory basierenden SOA-Grids die Tragfähigkeit von SoBI erhöht. Die unnötige Versendungen von aufgeblähten Nachrichten wird vermieden. Es erfolgt eine Reduzierung der Belastung des Service-Busses und der beteiligten Systeme und im Weiteren wird eine verbesserte Zugriffszeit anhand von In-Memory-Technologien erzielt. Infolgedessen sich die Entscheidungslatenz vermindert. Zusätzlich erweist sich diese Lösung insbesondere für Cloud Computing Anbieter als vorteilhaft. Demnach ermöglicht die Service-Orientierung die Bereitstellung von verkapselten Funktionalitäten, In-Memory Technologien eine verbesserte Zugriffszeit und das SOA-Grid eine erhöhte Erweiterbarkeit.

Resümierend kann konstatiert werden, dass SoBI weiterer Forschung bedarf [Di08], jedoch bereits jetzt ein vielversprechendes Konzept darstellt, welches den wechselnden BI-Paradigmen gerecht wird und in der Summe die Systemvitalität von BI erhöht [Gl08]. Hierbei liegt der nächste Schritt in der Realisierung eines Prototyps, um wertvolle Erkenntnisse für den Aufbau zukünftiger Systeme gewinnen zu können.

## Literaturverzeichnis

- [BD10] Bruns, R.; Dunkel, J.: Event-Driven Architecture - Softwarearchitektur für ereignisgesteuerte Geschäftsprozesse, 1. Auflage, Berlin, Springer, 2010.
- [CB08] Chappell, D.; Berry, D.: Next-Generation Grid-Enabled SOA: Not Your MOM's Bus, in: Erl, T., (Hrsg.): The SOA Magazine, Arcitura Education, Nr. 14, 2008; S. 1-8.
- [CB07] Chappell, D.; Berry, D.: SOA-Ready for Primetime: The Next-Generation, Grid-Enabled Service-Oriented Architecture, in: Erl, T., (Hrsg.): The SOA Magazine, Arcitura Education, Nr. 10, 2007; S. 1-10.
- [CG06] Chamoni, P.; Gluchowski, P.: Analytische Informationssysteme - Einordnung und Überblick, in: Chamoni, P.; Gluchowski, P., (Hrsg.): Analytische Informationssysteme, 3. Auflage, Berlin usw., Springer, 2006; S. 3-22.
- [Ch11] Chan, L. K.; Sim, Y. W.; Yeoh, W.: A SOA-Driven Business Intelligence Architecture, in: Communications of the IBIMA, 2011; S. 1-7.
- [Da09] Dayal, U.; Wilkinson, K.; Simitis, A.; Castellanos, M.: Business Processes Meet Operational Business Intelligence, in: Proceedings of IEEE Data Eng. Bull., Vol. 32, 3, 2009; S. 35-41.
- [DB06] Dinter, B.; Bucher, T.: Business Performance Management, in: Chamoni, P.; Gluchowski, P., (Hrsg.): Analytische Informationssysteme. Business Intelligence-Technologien und Anwendungen. 3. Auflage, Berlin usw., Springer, 2006; S. 23-50.



- [DE98] Dietzsch, A.; Esswein, W.: Wissensaustausch auf der Grundlage von Referenzmodellen - Anforderungen und Möglichkeiten, in: Zimmermann, H. H.; Schramm, V., (Hrsg.): Knowledge Management und Kommunikationssysteme, Konstanz, UVK Verlagsgesellschaft, 1998; S. 201-209.
- [Di08] Dinter, B.: Einsatzmöglichkeiten serviceorientierter Architekturen in der Informationslogistik, in: Töpfer, J.; Winter, R., (Hrsg.): Active Enterprise Intelligence, 1. Auflage, Berlin, Springer, 2008; S. 221-242.
- [Di07] Dittmar, C.: Latenzzeiten von Business Intelligence-Systemen, in: Gluchowski, P.; Chamoni, P.; Gersch, M.; Krebs, S.; Reinersmann, M., (Hrsg.): Schlaglichter der Wirtschaftsinformatik, 1. Auflage, Chemnitz, Verlag der Gesellschaft für Unternehmensrechnung und Controlling, 2007; S. 131-142.
- [DS09] Dinter, B.; Stroh, F.: Design Factors for Service-oriented Architecture Applied to Analytical Information Systems: an Explorative Analysis, in: Proceedings of the 17th European Conference On Information Systems, 2009; S. 1-12
- [EW03] Eckerson, W. W.; White, C.: TDWI Report Series - Evaluating ETL and Data Integration Platforms, unter: <http://www.evolve.mb.ca/dw/etlreport.pdf>, (05.07.2011).
- [Fi07] Fischer, J. H.: (24.09.2007), Data Warehousing Stammdaten – Der Business Case für SOA, <http://www.computerwoche.de/software/bi-ecm/544282/index.html>, (23.08.2011).
- [FL04] Fettke, P.; Loos, P.: Referenzmodellierungsforschung – Langfassung eines Aufsatzes, in: Working Papers of the Research Group Information Systems & Management, 16, 2004.
- [FZ09] Finger, P.; Zeppenfeld, K.: SOA und Web Services, 1. Auflage, Berlin, Springer, 2009.
- [Gl08] Gluchowski, P.; Gabriel, R.; Dittmar, C.: Management Support Systeme und Business Intelligence, 2. Auflage, Berlin, Heidelberg, Springer, 2008
- [Go06] Gordon, S.; Grigg, R.; Horne, M.; Thurman, S.: Service-Oriented Business Intelligence, in: The Architecture Journal, Vol. 6, 2006; S. 23-32.
- [Ha94] Hars, A.: Referenzdatenmodelle – Grundlagen effizienter Datenmodellierung, Wiesbaden, 1994.
- [KB09] Kemper, H. G.; Baars, H.: From Data Warehouse to transformation Hubs - A conceptual Architecture, in: Newell, S.; Whitley, E.; Pouloudi, N.; Wareham, J.; Mathiassen, L., (Hrsg.): Proceedings European Conference on Information Systems, 2009; S. 2-13.
- [KI08] Klawans, B.: Embedded or Conventional BI: Determining the Right Combination of BI for Your Business, in: Business Intelligence Journal, Vol. 13, Nr. 1, 2008; S. 30-36.
- [Ma11] Martin, W., (03.2011), Performance Management und Analytik – Business Intelligence trifft Business Process Management, <http://www.pmone.de/fileadmin/documents/studien/CPM-WhitePaper-7.1-D-2011-03-03-final.pdf>, (06.06.2011).
- [Ma06] Martin, W.: Analytics meets Enterprise SOA, (07.2006), [http://www.solutionsscout.com/fileadmin/downloads/experten/SAP-Analytics\\_whitepaper\\_de\\_finsec.pdf](http://www.solutionsscout.com/fileadmin/downloads/experten/SAP-Analytics_whitepaper_de_finsec.pdf), (16.06.2011).
- [Me10] Melzer, I.: Service-orientierte Architekturen mit Web Services, 4. Auflage, Heidelberg, Spektrum Verlag, 2010.
- [Mü10] Müller, R. M.; Linders, S.; Pires, L. F.: Business Intelligence and Service-oriented Architecture, in: Information Systems Management, Vol. 27, 2, 2010; S. 168-187.
- [Mu08] Mueller, W.: (08.06.2008), SOA Referenzarchitektur, <http://www.e-services.admin.ch/dokumentation/00162/00167/>, (30.06.2008).
- [Ne09] Neuhaus, S.: Konzeption einer Referenzarchitektur für Operational Business Intelligence auf Basis einer ereignisgesteuerten, serviceorientierten Architektur, in: Forschungskolloquium Business Intelligence 2009 der GI Fachgruppe 5.8. CEUR Workshop Proceedings, Vol. 542, 2009; S. 1-13.

- [Ng05] Nguyen, T. M.; Schiefer, S.; Tjoa, A. M.: Sense & response service architecture (SARESA): An approach towards a real-time business intelligence solution and its use for a fraud detection application, in: Proceedings International Workshop on Data Warehousing and OLAP (DOLAP), 2005; S. 77-86.
- [NO09] Nijkamp, E.; Oberhofer, M.: Embedded Analytics in Front Office Applications, in: Gesellschaft für Informatik BTW, Vol. 144, 2009; S. 449-459
- [Pa08] Panian, Z.: How to Make Business Intelligence Actionable through Service-oriented Architecture, in: Transactions on Business and Economics, Vol. 5, 5, 2008; S. 210-221.
- [PH11] Piller, G.; Hagedorn, J.: Einsatzpotenziale für In-Memory Data Management in betrieblichen Anwendungssystemen. In: Wirtschaftsinformatik, Vol. 53, 5, 2011; S. 18-25.
- [Ru11] Russom, P., (2011), TDWI Next Generation Data Integration, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2011/TDWI\\_BPReport\\_Q211\\_NGDI\\_Web.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2011/TDWI_BPReport_Q211_NGDI_Web.ashx), (05.07.2011).
- [Ru10] Russom, P., (2010), TDWI best practices Report operational data warehousing, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2010/TDWI\\_BPReport\\_Q410\\_OpDW.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2010/TDWI_BPReport_Q410_OpDW.ashx), (05.07.2011).
- [Ru09] Russom, P., (04.2009), TDWI best practices Report operational data integration A New Frontier for Data Management, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2009/TDWI\\_BPReport\\_Q09\\_ODI%20pdf.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2009/TDWI_BPReport_Q09_ODI%20pdf.ashx), (07.07.2011).
- [Si02] Sillitti, A.; Vernazza T.; Succi, G.: Service Oriented Programming: A New Paradigm of Software Reuse, in: Lecture Notes in Computer Science, Vol. 2319, 2002, 39-47.
- [Vo08] Vogt, T.; Neuhaus, S.; Linden, M.; Chamoni, P.: Business-Intelligence-Konzept auf Basis einer Event-Driven Service-Oriented Architecture, in: Dinter, B.; Winter, R.; Chamoni, P.; Gronau, N.; Turowski, K., (Hrsg.): Synergien durch Integration und Informationslogistik Proceedings DW 2008, 2008; S. 217-229.
- [Wh06] White, C., (06.2006), The Next Generation of Business Intelligence: Operational BI – BI Research, [http://certification.sybase.com/content/1041416/Sybase\\_OperationalBI\\_WP-071906.pdf](http://certification.sybase.com/content/1041416/Sybase_OperationalBI_WP-071906.pdf), (06.06.2011).
- [WH04] Wilkes, S.; Harby, J., (06.2004), A move to drive industry standardization of SOA, <http://xml.coverpages.org/SOA-BlueprintsConceptsV05.pdf>, (03.06.2011).
- [Wi08] Winter, R., (21.01.2008), SOA braucht eine klare Definition, <http://www.computerwoche.de/software/soa-bpm/1853224/index3.html>, (21.07.2011).
- [Za07] Zacharias, R.: SOA & Event Driven Architecture (EDA) – Eine perfekte Symbiose, in: Javamagazin, Vol. 7, 2007; S. 60-69.



# The Mainframe Strikes Back: Multi Tenancy in the Main Memory Database HyPer on a TB-Server

Henrik Mühe, Alfons Kemper, Thomas Neumann

TU München  
Boltzmannstr. 3  
85748 Garching, Germany  
muehe@in.tum.de kemper@in.tum.de neumann@in.tum.de

**Abstract:** Contrary to recent trends in database systems research focussing on scaling out workloads on a cluster of commodity computers, this presentation will break grounds for scale-up. We show that an elastic multi-tenancy solution can be achieved by combining a many-core server with a low footprint main memory database system. Total transactional throughput for TPC-C like order-entry transactions reaches up to 2 million transactions per second on a 32 core server while the number of tenants sharing a single server can be varied from a few to hundreds of separate tenants without diminishing total throughput. Contrary to common belief, a scale-up solution provides high flexibility for tenants with growing throughput needs and allows for simple sharing of common resources between different tenants while minimizing hardware and computing overhead. We show that our approach can handle changes in tenant requirements with minimal impact on other tenants on the server. Additionally, we prove that our architecture provides sufficient per-tenant throughput to handle big tenants and scales well with database size.

## 1 Introduction

Over the last years – perhaps fueled by observing the scale-out strategy of Google and other web service providers – the database community has largely focused on scale out using large numbers of low-cost commodity computers. An example of a provider offering this kind of hardware infrastructure is Amazon’s cloud offering called EC2, which allows customers to obtain resources in the form of predefined machine specifications – just as if they rented a commodity computer hosted inside Amazon’s data center<sup>1</sup>. In the area of database systems, cloud services typically offer two types of data stores: On the one hand, a key-value store implementation for data with low consistency requirements, like for instance click logs for a company’s website. On the other hand, a regular relational database system hosted inside one of the predefined machines which can be rented by their customers. Unfortunately, the latter implementation of a relational database in the cloud is in stark contrast to the widespread idea of cloud computing. According to wikipedia,

---

<sup>1</sup><http://aws.amazon.com/ec2/>

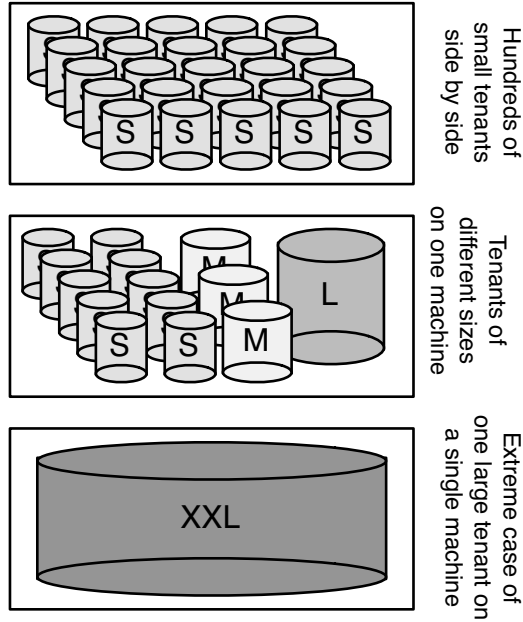


Figure 1: A range of usage scenarios – no explicit reconfiguration is required to adapt the cloud database server.

cloud computing provides a service – in this case a relational data store – instead of a server hosting a DBMS. The latter idea of a database in the cloud leads to multiple drawbacks:

**Scalability:** Since computing resources specified by a fixed machine profile are rented and used as a database server, no additional flexibility is offered over an actual computer hosted in a data center. If migration to a different, more powerful machine profile is at all possible, the process is not seamless and requires substantial user involvement which can lead to user error<sup>2</sup>.

**Sharing of resources:** With many applications, data can be shared between multiple tenants [ASJK11]. Using virtualization to share hardware between multiple tenants effectively prohibits this kind of resource sharing.

**Overhead:** Virtualization as well as the high footprint of traditional database systems incur substantial overhead. Without this additional layer between data and hardware, existing resources can be used more efficiently.

In this paper, we advocate combining multi-core server hardware with a low-footprint main memory database system to create an elastic and low-cost multi-tenancy setup. Commodity servers with 32 cores and main memory sizes of up to 1 Terabyte are widely available today for less than \$35,000 (c.f. Figure 2). Combined with a state of the art transactional

<sup>2</sup>An example of a user error when managing cloud resources in this context is described in <http://bit.ly/nZcX84>



Figure 2: A commodity server capable of a combined TPC-C like transaction throughput of up to 2 million transactions per second. Notice the keyboard on top of the machine for scale.

main memory database system like VoltDB [Vol10] or our prototype database system HyPer [KN11], an overall throughput in the order of millions of transactions per second for TPC-C like order-entry transactions can be achieved.

Instead of designing complex mechanisms to separate different tenants in one database system, our approach relies on the operating system to map different tenants' databases to specific cores to achieve maximum throughput. This keeps the overhead of the database system to a minimum and allows for features like different scheduling strategies and prioritization implemented in the OS to be used without cost.

## 2 Technical Realization

We have developed the HyPer database system prototype [KN11] which reduces the memory footprint of traditional database systems to a minimum. Our prototype consumes less than 100kb of memory for an empty database with the TPC-C schema and an additional 10MB of memory for the statically linked executable. Despite the low memory overhead and small executable size, HyPer is capable of running OLTP transactions comparable to

those found in the TPC-C at a rate of 100,000 transactions per seconds per thread. Additionally, HyPer is capable of executing long-running OLAP queries on an arbitrarily recent snapshot without severely impacting transaction throughput as we showed in [KN11] and demonstrated in [FKN11].

The high transaction throughput is achieved by reengineering a main memory database system from scratch, removing superfluous components formerly required in disk-based database systems and by using compilation over interpretation. Therefore, HyPer makes extensive use of the operating system’s virtual memory management, removing the need for a buffer manager and executes transactions serially as pioneered by H-Store [KKN<sup>+</sup>08], thus eliminating the need for a lock manager. To further increase performance, HyPer compiles pre-canned transactions as well as OLAP queries to machine code using the LLVM compiler infrastructure as illustrated in [Neu11]. All schema specific parts – like reading an attribute from a tuple or finding a tuple using an index – are generated as relation specific machine code, whereas general fragments like for instance a join operator are written in C++ and invoked from the generated code. Moreover, HyPer is fully ACID compliant, as atomicity and isolation directly follow from employing single threaded execution for transactions.

#Tenants	Average TPS	Overall TPS
1	80,729	81k
2	80,767	162k
4	81,060	324k
8	67,018	536k
16	58,605	938k
32	54,087	1731k
64	34,319	2197k
128	16,812	2152k
256	7,927	2029k
512	3,746	1919k

Table 1: Example configurations with a varying number of homogeneous tenants.

To support multiple tenants, multiple HyPer instances are launched, one for each tenant. Because of the compact footprint of the DBMS, even several hundred instances run on the server displayed in Figure 2 cause only a minimal memory overhead of less than 0.1%. When an instance does not execute any transactions or queries, no background tasks or batch processes are run, effectively reducing the CPU cycles required to run an idle instance to zero. Allocating tenants to a specific processor or migrating them to another one in case of a situation with unbalanced load is done by the operating system using the integrated scheduler. Resource sharing between different instances is achieved using shared memory segments mapped into the virtual memory of multiple instances. That way, read-only data can be shared with minimal implementation overhead. For data that needs to be updated infrequently, regular mutual exclusions as provided by the operating system ensure consistency in shared segments.

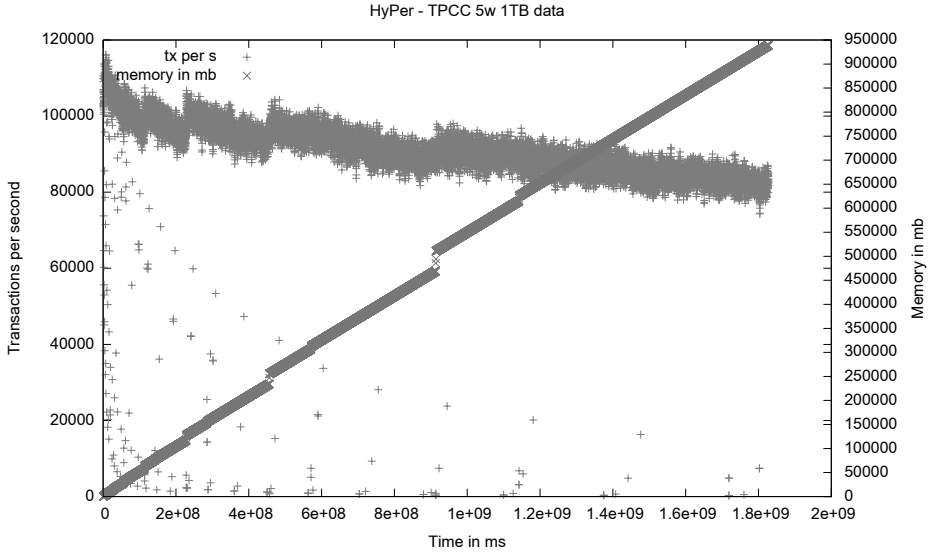


Figure 3: HyPer main memory database system growing form 500MB to 950GB of data with almost constant throughput.

Since resource distribution is dynamically controlled by the operating system, a tenant with growing throughput or memory demands will receive more resources without any explicit intervention. Table 1 shows average throughput and overall throughput for the same benchmark with a varying number of homogeneous clients. First, it can be observed that the overall throughput reaches a peak at 64 tenants which is the number of available hardware threads. Second, the sustained total throughput does not significantly diminish in an overload situation which is the case for 128, 256 and 512 tenants. Average throughput halves with each doubling of the number of tenants, but total throughput stays the same meaning that sharing a hardware context between multiple tenants does not incur unacceptable overhead.

The peak performance for a single instance caused by single- threaded serialized transaction execution as employed by current main memory database systems is not a limiting factor here. A single threaded instance for a tenant can execute about 80,000 transactions per second (c.f. Table s1) even when OLAP queries run simultaneously on a snapshot. A back-of-the-envelope calculation shows, that even big retailers like Amazon – on average – have orders of magnitude less orders than can be processed in 80,000 TPC-C transactions: Amazon has a yearly revenue of about 15 billion Euros [KN11]. Assuming that an individual order line is valued at 15 Euros and each order contains an average of 5 items, the average number of orders is less than 7 per second – significantly less than the 80,000 order related transactions achievable in a single thread.

The transaction throughput can be sustained over time and does not degrade for customers with a high amount of data. Figure 3 shows a single instance run at peak performance until



the database reached a size of about 900 gigabytes of data. Note that this amount of data can cover orders totaling about 10 billion items which were processed in less than 8 hours.

When the resource requirements of all tenants on a server outgrow the available resources, tenants can be moved easily by leveraging the snapshot mechanism integrated in HyPer, which can be applied to any main memory database system. A consistent snapshot of the DBMS can be created with minimal overhead as described in [KN11], transferred to the target server and be updated using the redo log. That way, migrating a tenant requires only one visible interruption in the order of milliseconds for snapshot creation before clients can start running transactions against the target server.

### 3 Preliminary results

Our evaluation showing the implementation of multi-tenancy on a many core server uses our HyPer main memory database system. HyPer was originally engineered to show the feasibility of executing both OLTP and OLAP on the same database state by using a virtual memory snapshot (forked process) for the OLAP queries [KN11]. Because HyPer has been built from scratch to specifically work in main memory, its computational as well as memory footprint is small, making it an ideal candidate for the evaluation of a low overhead multi-tenancy approach.

In order to show different multi-tenancy scenarios, we built a management component which can automatically deploy groups of clients with different usage patterns in terms of throughput, memory usage and change in resource consumption over time. We were able to show the viability of different predefined scenarios with changing number of tenants and workload characteristics. In Figure 4, the graphical interface of the management component is shown. The scenario displayed is a server initially hosting 50 tenants with comparable throughput requirements. The fact that one tenants steadily increasing throughput demands have no visible influence on other tenants on the same machine can be observed, proving the high elasticity provided by our setup.

We were able to test the full spectrum of flexibility that a TB server provides: Sharing of the resources by hundreds of small tenants, dedicating the entire server to a large Amazon-style tenant as well as several scenarios in between.

The CH-benCHmark [CFG<sup>+</sup>11], a combination of the TPC-C transactional workload and the queries specified in the TPC-H, will be used to simulate each tenants workload. The workload can be parameterized to exhibit different characteristics, both in initial throughput and memory consumption as well as their variance over time. This allows for simulating both, OLTP transactions as well OLAP queries executed using HyPer’s snapshotting mechanism, therefore simulating the full repertoire of client needs.

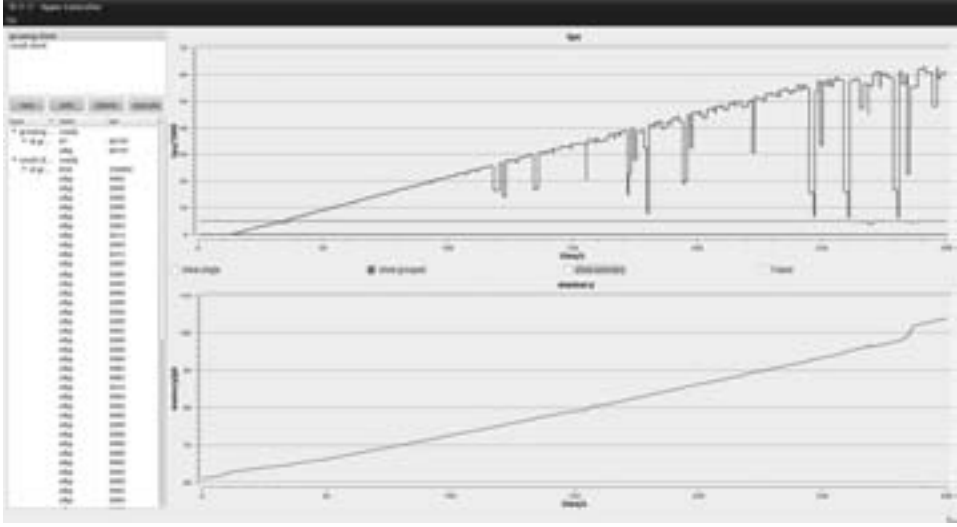


Figure 4: Screenshot displaying the management component of our approach. The diagram in the top-right shows a live view of the transaction throughput of each tenant. Note that the constantly rising transaction rate of the blue tenant does not influence the other 50 instances pictured in red.

## 4 Summary

Our presentation will show that there are elastic scaling solutions beyond massive deployment of cheap low-end machines. The approach introduced in this paper allows high transaction throughput by collocating many tenants on a multi-core server, allowing cost to be split among tenants and common resources to be shared. Our setup behaves predictably even when workloads change drastically and allows a high degree of elasticity for many small tenants with growing data management needs as well as big tenants having to process ten-thousands of transactions.

The advantages of our approach are achieved by combining a multitude of factors:

- Our state of the art, small footprint main memory database system, HyPer. By combining recent research results like serial execution without a lock- or buffermanager, transaction and query compilation in a database system specifically designed for use in main memory, we constructed a low footprint high performance DBMS that can be individually deployed for each tenant.
- By allowing the operating system to schedule each tenant separately, system resources are used efficiently without explicitly managing all DBMS instances. Furthermore, the architecture automatically benefits from improvements in the operating system, for instance adjustments due to changing hardware properties like - for instance - non uniform memory access.

- Simplified resource sharing like – for example – read-only database content or memory temporarily used for query execution by relinquishing strict separation in contrast to virtualized environments.

All in all, we could show how main memory will not only change the database systems landscape in terms of processing speed but additionally in how solutions for well known problems – like multi-tenancy – can be crafted.

## References

- [ASJK11] Stefan Aulbach, Michael Seibold, Dean Jacobs, and Alfons Kemper. Extensibility and Data Sharing in evolving multi-tenant databases. In *ICDE*, pages 99–110, 2011.
- [CFG<sup>+</sup>11] Rick Cole, Florian Funke, Leo Giakoumakis, Wey Guy, Alfons Kemper, Stefan Krompaß, Harumi Kuno, Raghunath Nambiar, Thomas Neumann, Meikel Poess, Kai-Uwe Sattler, Michael Seibold, Eric Simon, and Florian Waas. The Mixed Workload CH-benCHmark. In *DBTest*, 2011.
- [FKN11] Florian Funke, Alfons Kemper, and Thomas Neumann. HyPer-sonic Combined Transaction AND Query Processing. In *PVLDB*, 2011.
- [KKN<sup>+</sup>08] Robert Kallman, Hideaki Kimura, Jonathan Natkins, Andrew Pavlo, Alex Rasin, Stanley B. Zdonik, Evan P. C. Jones, Samuel Madden, Michael Stonebraker, Yang Zhang, John Hugg, and Daniel J. Abadi. H-store: a high-performance, distributed main memory transaction processing system. *PVLDB*, 1(2):1496–1499, 2008.
- [KN11] Alfons Kemper and Thomas Neumann. HyPer: A Hybrid OLTP&OLAP Main Memory Database System Based on Virtual Memory Snapshots. In *ICDE*, 2011.
- [Neu11] Thomas Neumann. Efficiently Compiling Efficient Query Plans for Modern Hardware. In *VLDB*, 2011.
- [Vol10] VoltDB LLC. VoltDB Technical Overview. [http://voltdb.com/\\_pdf/VoltDBTechnicalOverviewWhitePaper.pdf](http://voltdb.com/_pdf/VoltDBTechnicalOverviewWhitePaper.pdf), 2010.

# Real-time Semantic Process Change Impact Analysis

Andreas Emrich, Dirk Werth und Peter Loos

Institut für Wirtschaftsinformatik (IWi) am  
Deutschen Forschungszentrum für Künstliche Intelligenz (DFKI)  
Stuhlsatzenhausweg 3, Campus D3.2  
66123 Saarbrücken  
{andreas.emrich | dirk.werth | peter.loos}@iwi.dfki.de

**Abstract:** Today's environment and business is constantly changing on a rapid pace. Models have to be adapted in order to fit the changing situations. In contrast, the complexity of enterprise models is a huge problem wrt the management of such complex models. High-performance computing technologies offer a great opportunity to leverage increased computational power to cope with that complexity. In this paper we will present an approach for determining the impact of process changes using a semantic context model for BPM that enables semantic querying on complex business process models. Based on this context models large ontologies such as Cyc and process repositories such as the MIT process handbook are queried in complex scenarios of our evaluation. In-memory databases such as Couch DB and Hadoop serve as technological basis for this evaluation. The paper concludes that such queries can be performed nearly in real-time.

## 1 Introduction

Today's business world is rapidly changing. One cause for this trend is the established correlation between economics and IT [SAE05]. This leads to growing enterprise models and with the size of these models their complexity increases as well [AG04]. The complexity of these models may also be too high to be understood by an employee who is only responsible for order processing. In some cases such an employee may even not be aware changes in the business model or strategy took place [GRC04]. That means when the management of an enterprise changed the business model in a specific way that affects operational processes, these changes have to be made accessible and understandable for the employees who execute the business processes. Otherwise the impact of these changes on the enterprise's performance may fade. So as to handle and communicate these changes properly, analytical data concerning the executed operational processes has to be used for management information as well as for the information of employees who work on a lower organizational level. To ensure that this data is up-to-date the required analyses have to be made for every process in real-time. This means that for example if an employee receives a customer order and starts to process it, the system would have to display information about the customer, about the product he or she ordered and what this order means in general for the enterprise. For

example if the employee sees that the customer he or she has to deal with is a major client of the enterprise and generates a major part of its turnover and that the ordered product is only available in a limited number, the employee may suggest that it would be better for the enterprise to satisfy the needs of the important client first. Additional information could be the known behavior of every client who ordered the product when it comes to delivery delays. Maybe there are clients that respond to such disappointments by reducing their share-of-wallet for the enterprise whereas other clients do not change their purchasing behavior at all. This information could assist the employee deciding which client should be delivered the product immediately and which shall not.

These analyses need a lot of computational power and are therefore usually not available in real-time [NTT05]. This power can be provided by High Performance Computing (HPC)[PI09] technologies. Using this technology, it is also feasible to simulate business process changes to be made in real-time. The goal of this work is to provide the results of the explained computations in less than one second to be able to present the information fast enough to support the respective employees. Furthermore, a software architecture is needed to enhance the speed of the necessary computations. In-memory databases [PI09] are one way to do so. With their ability to store data in the main memory of the server, they can shorten data access very easy. Through such a high performance system the impact on the business processes themselves or on related resources can be assessed and the simulated changes can be approved. To provide the relationships between artifacts the business logic has to be modeled and assessed properly. Furthermore the data has to be stored in a way that ensures cross application and even cross enterprise access. In order to enable this, the data structure has to be very generic. Therefore a semantic context model for BPM is needed that is capable of describing the relationships between resources and business processes in a generic way. Moreover such a model could be used to map every interaction in the enterprise to an ontology and by that ensure cross application and cross enterprise traceability of every piece of information.

In this paper such a model is presented in section 3. Later the performance evaluations of a test scenario that is supposed to show the feasibility of the computations in less than one second are presented including a description of the prototype which is meant to run representative queries on a data set that is large enough to simulate business reality such as the MIT process handbook, the test design that describes the structure of the test and the expected results in detail and the presentation of the evaluation results. Overall, the paper should show, that semantic process change impact analysis can be performed nearly in real-time, thus enabling real-time business intelligence scenarios.

## **2 Related Work**

Because of the problem of fast changing and heterogeneous software landscapes in today's enterprises approaches that are able to deal with the upcoming complexity of information flow are needed and developed [BH95]. Especially the traceability of relevant information has to be ensured [PBM07]. This is trivial as long as the information is passed in a structured way inside a single application. But it becomes

more and more complex as media disruptions usually occur especially if the supply chain partners try to integrate their applications [TP07].

Business Process Intelligence is another widely discussed method to address the problem [MR06] [Gr04]. It not only analyses data that has been collected in the past but it also tries to make conclusions about future process paths and issue spots using techniques like data or process mining [AW04]. A real time status of running processes can be computed harnessing the prediction of process paths.

All contemporary approaches are flawed by the fact that they are not able to put the tracked information in a semantic context that would enable the system to not only clear the provenance of the information but also provide the knowledge what processes could need it afterwards and what OLAP queries will be affected by it. That is where the Semantic Context Model for BPM as developed in the following section is needed.

To ensure the presence of current data at the start of every business process OnLine Analytical Processing (OLAP) [CCS93] and OnLine Transaction Processing (OLTP) have to be linked in order to trigger an OLAP query when a transaction starts. In that way, every transaction can be enriched with relevant data in order to optimize the business process. In addition to that, further analyses can be triggered that will support or prepare transactions that follow the one that just started. Thereby it is possible to ensure an optimal business process execution. Furthermore consequences of started transactions can be forecasted ensuring the relevance to the current situation and the accuracy of later analyses. Because of this accuracy the business process itself can be analyzed and modified based on real-time information.

Another problem is the analysis of the already mentioned business process changes which are also widely discussed [DHC04]. It is important to be able to measure the success of these changes in terms of its impact on perceived productivity and the related needs for information technology changes [Gr98]. To enhance the use of such changes it is also necessary to analyze the business process itself in advance [Bi00]. For all these analyses it is also important to model the business and its processes in a structured way in order to detect relationships between resources. Moreover the simulation of changes is useful to predict their impact in advance [Tu96]. In order to simulate changes properly the concerned processes have to be modeled. Such a model has to be updated every time a change to a process is applied in order to make sure to consider these changes for further simulations and computations [LBS08]. In this paper the Semantic Context Model for BPM (section 3) is used to solve this problem. Using this model it is possible to model the relationship in a business properly and to generate information from operational data and these relationships.

But in order to harness this information in an efficient way it is necessary to offer the generated information in real-time [NST05]. Traditionally business analyses are made monthly or yearly for marketing or finance departments. Therefore the data has to be loaded into a data warehouse which aggregates operational data to enhance the speed and the significance of queries that aim to generate relevant information [SR08]. The obvious disadvantage of this procedure is the stale data base on which the queries are

run. Decisions derived from such data could be wrong or at least not as efficient as they could be if they used real-time data [Az06]. Another disadvantage of the traditional approach is the long period between the computations. The cause for this is the expensiveness of such queries in terms of runtime. But as already mentioned the impact of the availability of such information could be maximized if every operational transaction could be enriched with the resulting data of these queries in real-time. Overall real-time business analyses have a huge impact on the enterprise's success [An04]. These analyses are mostly done by business intelligence (BI) systems. These systems gather and store data and manage knowledge in order to present it in an edited way to planners and decision makers [Ne04]. In general there are three scopes of BI: strategic, tactical and operational. Strategic BI is used to plan and define an enterprise's goals over a long period such as months or years. These goals are specified to be measured with certain Key Performance Indicators (KPIs). If these indicators do not apply the right values, tactical BI has to come up with measures that will improve the values of the KPIs. These measures are mostly planned over days or weeks or months. The most relevant variant of BI for this paper is operational BI. It delivers business analyses at right-time in order to support the daily business [Wh06]. BI can also be used to avoid operational risks in an enterprise but for this purpose it is also more efficient to use real-time business intelligence [Az06][Az07].

A related issue is the definition of the term "real-time". There are several different understandings of this term. For some systems it means that any latency within a process should be avoided. Another point of view is that "real-time" means that information should be accessible when it is needed. It could also be used to describe the requirement for a process to provide information to the management on demand or the ability to provide key performance indicators that reflect the current situation of the business [Az06]. In this paper the term "real-time" is used to describe the necessity to provide the result of every aggregating query in less than one second, i.e. to provide impact analysis on process changes, once these occur. Thereby every definition of the term is satisfied to some extent. The user will not experience any latency if the results to a specific query are presented in below one second. When the computation is finished during this time the information will always be available when it is needed. Because the repository should be built with current data the results will show the as-is situation of the queried section of the enterprise. This is also a requirement which is quite measureable and which will expose the difference between traditional hardware systems and HPC technology [PI09]. By making data mining analyses available on a transactional level in real-time business intelligence can be used as a tool for business management [GRC04].

### **3 Conceiving Impact Analysis through a Semantic Context Model for BPM**

This section describes the Semantic Context Model for BPM developed to save the knowledge about connected resources in an enterprise in order to be able to compute and forecast impacts of transactions or changes to the business processes on every related resource. This Semantic Context Model has been developed to improve structured, semi-structured and unstructured scenarios. It allows a formal, multi-view-enabled view on

various aspects. In addition to that, search and recommendations as well as semantic relationship discovery support less structured scenarios. The Semantic Context Model is based on the EPC and tries to make conclusions about the connectedness of artifacts using transitive relationships. This ability may also be used to trace structural equivalences between processes and these equivalences can serve to simulate and eventually adapt changes made to analyzed processes to similar ones.

The model consists of three layers: the meta layer, the model layer and the instance layer. The meta layer describes process types, different kinds of documents and possible user roles. The model layer describes concrete classes. As the name already suggests concrete instances of these can be found on the instance layer.

Figure 1 in [Em10] shows the structure and the connection of the Semantic Context Model to the EPC. The EPC is represented by elements of the instance layer of the ontology. These elements are instances of the class level. In this case they are directly assigned to the functional elements “Function”, “Connector”, “Organization” and “Resources”.

Because this partition of the concept is very similar to the object oriented paradigm, it can be realized with OWL. The meta layer is represented by the class level and the instance layer can be displayed using the activities and peculiarities.

Because the ontology representation of business logic is already targeted by many researches it is as desirable as hard to integrate them to the Semantic Context Model. This can be done using the presented ontology as a wrapper ontology and connect it to others using similarTo, subClass and superClass relationships to depict similar concepts as well as subordinated and superordinated concepts.

Figure 2 from [Em10] explains how a connection of the developed ontology with an external ontology, in this example the BMO [Je11], could look like in the scope of the Semantic Context Model. In this manner, arbitrary aspects from the context can be integrated with the semantic context model in a lightweight manner. This is especially helpful in highly unstructured scenarios, where it is not feasible or efficient to model relationships to external aspects.

For explicating, how this semantic context model can be used to enable real-time semantic process change analysis, a sample business process is being reviewed. As an EPC, it includes the basic events and functions of a business process, but also associated stakeholders, risks, systems and documents. Each of these aspects can be interlinked with external ontologies, e.g. describing the spatial and temporal characteristics of an entity in the process.

## **4 Evaluation**

In this chapter we evaluate the introduced model and the mechanism to detect changes and their impact on the model by using different existing ontologies. To simulate the impact of a change, a sufficient amount of data is needed which is similar to the amount



used in a real world environment. At this stage of work we had to attach existing sometimes not-business related ontologies to our proposed upper ontology to get the information if we can provide results in a reasonable time.

In our prototypical environment we assume that concepts are highly interlinked with other concepts through subclass, object property, and datatype relations. This abstract view refers to a business use-case where a business process or any artifact has several connections to other processes and instances. We describe the impact of a change as possible consequences to other connected artifacts. The connection is not only limited to concepts and individuals connected to the source artifact where the change was initiated but also to artifacts with a higher logical distance to it. We follow the impact of a change by following all connections from the changed artifact itself. Our reasoner is able to detect and eliminate cycles in the reasoning path and also to introduce heuristics to stop after certain steps of following a too deep path.

#### **4.1 Subjects of the evaluation**

Is it feasible to get results (the impact of a certain change) in a certain amount of time (real time or nearly real time) to handle tasks needed by the imitated change. Three different classifications of time were considered:

- *Short-term impact:* Operational tasks and actions, e.g. the impact of a sickness of an employee and subsequent delegation of work to other employees.
- *Medium-term impact:* Tactical tasks and actions, e.g. changes in the supply chain such as bankrupt of a supplier, etc.
- *Long-term impact:* Strategic tasks and actions, e.g. changed consumer behaviour because of the financial crisis forces companies to rethink their product and service portfolio.

#### *Expected Results*

For the given set of data we achieved good results in nearly real time but more business related data set is needed as well as the forecasting itself must be researched. How does a change and the impact in the context model really affect the real world?

In this scenario two major problems arise. The first problem is the high data size of the used ontologies which either have to be saved on the disk or in an in-memory database. The second problem is the complex reasoning process which needs strong process power and good memory performance as several intermediate reasoning steps have to be saved. We try to solve these challenges with a high-performance infrastructure provided by the Hasso-Plattner Institut Future SOC Lab.

## **4.2 Test Case Description**

In order to check the performance of the architecture against a sufficiently large variety of data the test is run over three different ontologies: The first ontology is the OWL version of the MIT Process Handbook with a size of 16Mbytes and about 8000 artifacts like resources and processes which are interlinked with each other. The second one is the openCyc ontology with a size of 160Mbytes. It contains general knowledge and is therefore not limited to business related topics. The third used ontology is the YAGO ontology with a size of 20 Gbytes and nearly two million entities. These ontologies were chosen because it is a goal of the whole test case to find the right technology and storage method of an ontology to get query results which are (almost) real-time which means a runtime shorter than one second. Therefore large datasets have to be queried during the tests. The ontologies have to be loaded in the repository. Because it is interesting in which amount the performance of In-Memory databases excels the performance of traditional disk databases the two technologies will be compared. After the data is loaded incoming events like changes and actions that affect a resource are analyzed. Affected artifacts are highlighted and available to be used for further computations about the process trail. Furthermore, simulations could be executed in order to determine how changes to the resource would affect other artifacts. Sesame was used to create the repository. It is possible to use file based storage layer as well as in-memory databases and traditional relational database management systems.

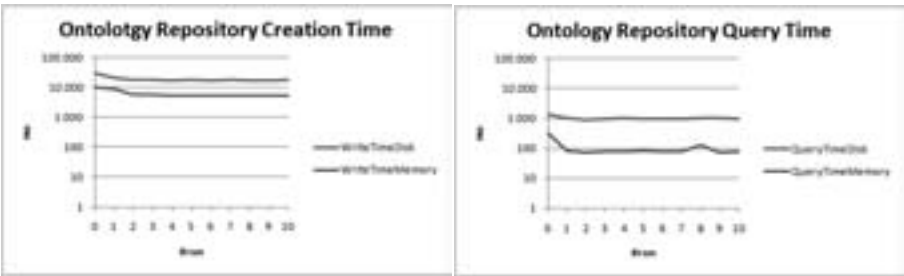
## **4.3 Test Design**

The test that targets the comparison of disk and In-Memory persistence is only run on more than one ontology if the run on the first ontology will not deliver results that expose a significant improvement when it comes to In-Memory databases. To minimize the effort of this comparison it is run on the smallest data set. In order to check how the use of HPC technology affects the runtime of the queries made on the databases, the tests are run as well on an ordinary workstation with a dual core processor and 4Gbyte of main memory as in an HPC environment. To build the repository in the main memory the heapsize was increased to 3Gbytes on the workstation and 32Gbytes on the HPC environment.

## **4.4 Evaluation Results**

In order to evaluate the tests run as described in the previous sections the time has been measured to check whether the runtime fulfills the criterion of being below one second. Additionally this enables comparisons between the several test runs. Thereby it is possible to show if the HPC approach is really able to increase performance significantly in order to enable real-time analyses. In the following the results are explained textually as well as graphically. The diagrams show the measured runtime for every executed test run. For each test there is one diagram that compares the required time for writing access and on that compares the required time for reading access. The time has been measured in milliseconds which is why the scale goes up to 100,000. In order to make the graphs

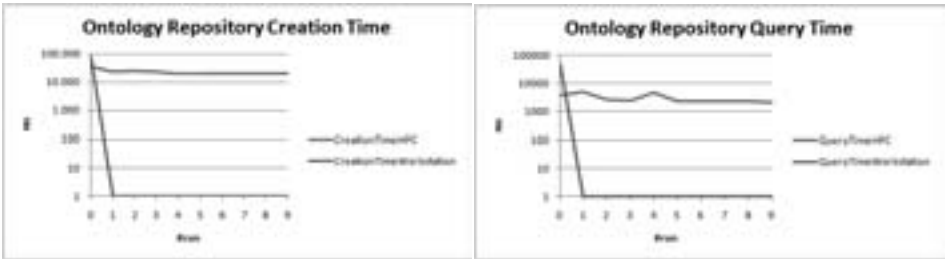
comparable and readable the diagrams are presented using a logarithmic scale. The first important result is the difference between disk and In-Memory persistence.



**Fig. 1.** Comparison of disk and In-Memory persistence

As expected the performance of the In-Memory database was very fast. When it came to write access it excelled the disk database in terms of speed by an average factor of three. When it came to read access it was even nine times as fast as the disk database. Because this confirmed our expectations there is no need for further comparisons of disk and In-Memory persistence. Nevertheless these results prove that an In-memory database should be used to achieve optimal results.

In the next step the In-Memory database was used to test the performance of queries all the datasets.



**Fig. 2.** Performance Data HPC Cluster

Ten runs were triggered as well on the workstation as on the HPC environment but the workstation was only able to create the repository once. After that the java virtual machine crashed due to the limited capacity. Besides the workstation needed in average 80 seconds to query the whole dataset which is far too long to be considered as real-time. Therefore, only the HPC environment with In-memory databases seems to capable of processing such huge amounts of data.

### 5 Conclusion & Outlook

This paper presented an approach for semantics-enabled impact analysis of business process changes to support business intelligence support in real-time. A semantic context

model covers the basic BPM artifacts and enables the interlinking with other aspects in the context. The evaluation based on large ontology sets and process repositories has shown, that impact analysis can be performed very quickly for different use cases. High-performance computing infrastructure helps to provide this near real-time. However, the depicted data could be extended with simulation scenarios, also determining the costs of different impacts, thus helping to assess different change management alternatives. Moreover, structural workflow patterns could be used to support a structural automatic analysis of large repositories.

## References

- [AG04] Arteta, B.M., Giachetti, R.E.: A measure of agility as the complexity of the enterprise system. *Robotics and Computer-Integrated Manufacturing*. 20, 495–503 (2004).
- [An04] Anderson-Lehman, R., Watson, H.J., Wixom, B.H., Hoffer, J.A.: Continental airlines flies high with real-time business intelligence. *MIS Quarterly Executive*. 3, 163–176 (2004).
- [AW04] van der Aalst, W.M., Weijters, A.: Process mining: a research agenda. *Computers in Industry*. 53, 231–244 (2004).
- [Az06] Azvine, B., Cui, Z., Nauck, D.D., Majeed, B.: Real time business intelligence for the adaptive enterprise. In *Proceedings of the The 8th IEEE International Conference on E-Commerce Technology and The 3rd IEEE International Conference on Enterprise Computing, E-Commerce, and E-Services (CEC-EEE '06)*. IEEE Computer Society, Washington, DC, (2006).
- [Az07] Azvine, B., Cui, Z., Majeed, B., Spott, M.: Operational risk management with real-time business intelligence. *BT technology Journal*. 25, 154–167 (2007).
- [BH95] Bettis, R.A., Hitt, M.A.: The new competitive landscape. *Strategic Management Journal*. 16, 7–7 (1995).
- [Bi00] Biazzo, S.: Approaches to business process analysis: a review. *Business Process Management Journal*. 6, 99–112 (2000).
- [CCS93] Codd, E., Codd, S., Salley, C.: Providing OLAP to User-Analysts: An IT Mandate. (1993).
- [DHC04] Davenport, T.H., Harris, J.G., Cantrell, S.: Enterprise systems and ongoing process change. *Business Process Management Journal*. 10, 16–26 (2004).
- [Em10] Emrich, A., Ganz, F., Werth, D., Loos, P.: Enabling Cross-Application Traceability of Semi-Structured Business Processes. In: *Proceedings of the 1st International Workshop on Traceability and Compliance of Semi-Structured Processes. (TC4SP-2010)*, co-located with BPM 2010, September 13, Hoboken, NJ, United States, o.A. 2010. (2010).
- [Gr98] Grover, V., Teng, J., Segars, A.H., Fiedler, K.: The influence of information technology diffusion and business process change on perceived productivity: the IS executive's perspective. *Information & Management*. 34, 141–159 (1998).
- [Gr04] Grigori, D., Casati, F., Castellanos, M., Dayal, U., Sayal, M., Shan, M.C.: Business process intelligence. *Computers in Industry*. 53, 321–343 (2004).
- [GRC04] Golfarelli, M., Rizzi, S., Cella, I.: Beyond data warehousing: what's next in business intelligence? *Proceedings of the 7th ACM international workshop on Data warehousing and OLAP*. S. 1–6 (2004).
- [Je11] Jenz & Partner: The Open Source Business Management Ontology (BMO), [http://www.bpiresearch.com/Resources/RE\\_OSSOnt/re\\_ossont.htm](http://www.bpiresearch.com/Resources/RE_OSSOnt/re_ossont.htm). (2011)
- [LBS08] Lautenbacher, F., Bauer, B., Seitz, C.: Semantic business process modeling-benefits and capability. *AAAI 2008 Stanford Spring Symposium-AI Meets Business Rules and Process Management (AIBR)*, Stanford University, California, USA (2008).

- [MR06] Mutschler, B., Reichert, M.U.: Aktuelles Schlagwort: Business Process Intelligence. (2006).
- [Ne04] Negash, S.: Business intelligence. *Communications of the Association for Information Systems* (Volume13, 2004). 177, 177 (2004).
- [NST05] Nguyen, T.M., Schiefer, J., Tjoa, A.M.: Sense & response service architecture (SARESA): an approach towards a real-time business intelligence solution and its use for a fraud detection application. *Proceedings of the 8th ACM International Workshop on Data warehousing and OLAP*. S. 77–86 (2005).
- [NTT05] Nguyen, T.M., Tjoa, A.M., Trujillo, J.: Data warehousing and knowledge discovery: A chronological view of research challenges. *Data warehousing and knowledge discovery*. 530–535 (2005).
- [PBM07] Panetto, H., Baína, S., Morel, G.: Mapping the IEC 62264 models onto the Zachman framework for analysing products information traceability: a case study. *J Intell Manuf.* 18, 679-698 (2007).
- [PI09] 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 (SIGMOD '09)*, Carsten Binnig and Benoit Dageville (Eds.). ACM, New York, NY, USA, 1-2. (2009).
- [SAE05] Scheer, A.W., Adam, O., Erbach, F.: Next generation business process management. *Von Prozessmodellen zu lauffähigen Anwendungen*. 1–15 (2005).
- [SR08] Sahay, B., Ranjan, J.: Real time business intelligence in supply chain analytics. *Information Management & Computer Security*. 16, 28-48 (2008).
- [TP07] Theuvsen, L., Plumeyer, C.H.: Certification Schemes, Quality-Related Communication in Food Supply Chains and Consequences for IT-Infrastructures. *Environmental and Rural Sustainability through ICT*. *Proceedings of EFITA/WCCA*. S. 2–5 (2007).
- [Tu96] Tumay, K.: Business process simulation. *Proceedings of the 28th conference on Winter simulation*. S. 93–98 (1996).
- [Wh06] White, C.: *The Next Generation of Business Intelligence: Operational BI*. BI Research, Ashland, Oregon. (2006).

# 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](mailto:tobias.knabke@stud.uni-due.de)  
[sebastian.olbrich@uni-due.de](mailto: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 proof/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].



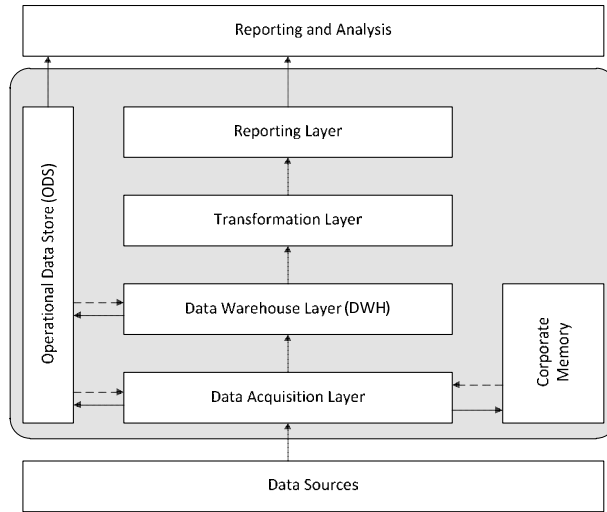


Figure 1: Multi-layer BI architecture (adapted from [Ha10])

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 re-loaded 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 deduces 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 deduced "*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 VDPWH 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, VDPWHs do not store any data physically but access data directly on the source systems [SBM99]. Nevertheless, VDPWHs 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 VDPWH.

### 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, PI09, 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 [PI09, 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.

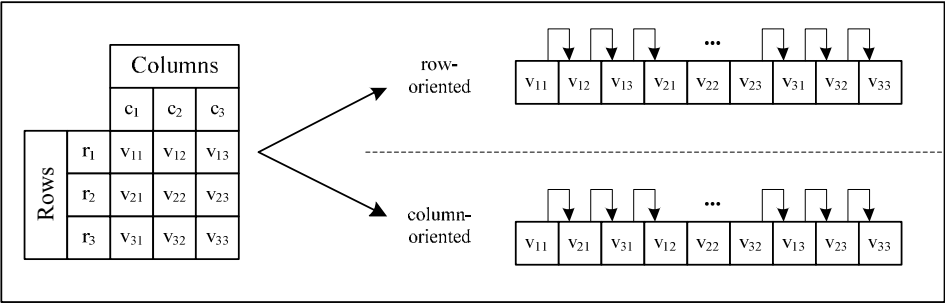


Figure 2: Data storage alternatives (adapted from [PZ11])

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 [PI09]. Because BI systems are OLAP applications the focus lies on the column-oriented storage in this paper.

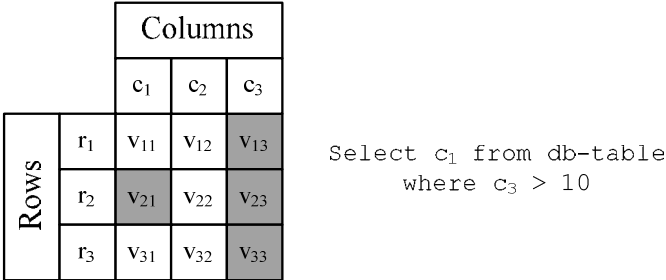


Figure 3: Example query using column-orientation (adapted from [PI09])

### 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.

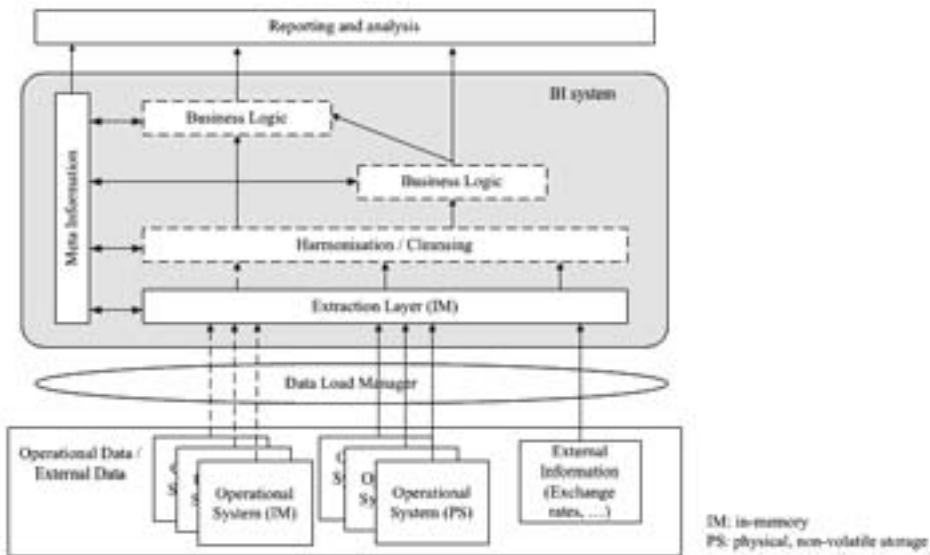


Figure 4: Semi-virtual DWH using IMDB

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 [PI09, 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, PI09, 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 [PI09]. 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](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. Vieweg, Wiesbaden, 2006.
- [Ki96] Kimball, R.: The data warehouse toolkit: Practical techniques for building dimensional data warehouses. Wiley, New York [u.a.], 1996.
- [La11] Lahrman, 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 systems*, 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.



# **Ökonomisch sinnhafte Bewertung von „In-Memory-basierten betrieblichen Informationssystemen“**

Marco C. Meier, Alexa Scheffler

Kernkompetenzzentrum Finanz- und Informationsmanagement  
Fraunhofer Projektgruppe Wirtschaftsinformatik  
Universitätsstr. 12  
86159 Augsburg  
marco.meier@wiwi.uni-augsburg.de  
alexa.scheffler@wiwi.uni-augsburg.de

**Abstract:** In-Memory Data Management bietet das Potenzial, sich im Sinne eines nachhaltigen Trends in der Informationstechnologie zu etablieren, der die integrierte betriebliche Informationsverarbeitung erheblich beeinflussen könnte. Gleichsam existiert noch so viel Unsicherheit hinsichtlich eines ökonomisch sinnvollen Einsatzes, dass auf betrieblicher und gesellschaftlicher Ebene Fehler drohen, wenn es gilt, zu entscheiden ob, wo und wie In-Memory Data Management praktisch eingesetzt und erforscht werden sollte. Um dieses Fehlentscheidungsrisiko zu verringern, ist die systematische Entwicklung von geeigneten Methoden der ökonomischen Bewertung von „In-Memory-basierten Informationssystemen“ Gegenstand dieses Beitrags.

## **1 Motivation**

Ein grundlegendes Dilemma, mit dem Entscheidungsträger im IT-Bereich naturgemäß konfrontiert sind, wird derzeit mit dem Thema In-Memory Data Management und der damit verbundenen Vision des „Real-time-Unternehmens“ [ACN09] wieder akut. Einerseits gilt es, die Chancen, die das Potenzial neuer Technologien birgt, schnell zu nutzen um sich Wettbewerbsvorteile zu verschaffen. Andererseits sind die Verantwortlichen mit einem vergleichsweise hohen Maß an Komplexität und Unsicherheit konfrontiert, sodass Fehlinvestitionen drohen.

In Forschung und Praxis mehren sich Einschätzungen, wonach In-Memory Data Management mittlerweile einen Stand erreicht hat, der es zu einer Schlüsseltechnologie für eine neue „Ära“ der betrieblichen Informationsverarbeitung macht [Pl09, ZP11].

In der Vergangenheit gab es immer wieder Themen, beispielsweise virtuelle Welten, die ähnliche Begeisterung entfachten, dann aber im Sinne einer „Modeerscheinung“ auch schnell wieder aus dem Fokus verschwanden und sich nicht zu einem nachhaltigen „Trend“ entwickelten.

Der typische zeitliche Verlauf bei der Entwicklung und Verbreitung einer neuen (Informations-)Technologie ist geprägt von schnell wachsender Euphorie sowie einem nahezu ebenso rasanten Abschwung, wenn sich herausstellt, dass die (zu) hohen Erwartungen in der Praxis nicht erfüllt werden können. In einigen Fällen stellt sich mit einer zeitlichen Verzögerung von mehreren Jahren dann ein sinnhaftes Nutzen- und damit verbunden auch Nutzungsniveau ein [SRR09].

In Praxis und Wissenschaft besteht also gleichermaßen ein akuter Bedarf, schnell und verlässlich einschätzen zu können, für welche Anwendungsbereiche und unter welchen Voraussetzungen Investitionen in den Einsatz sowie in die weitere Erforschung und Entwicklung von In-Memory Data Management ökonomisch sinnvoll sind.

Sachziel dieses Beitrags ist es dementsprechend zu untersuchen, welche Anforderungen geeignete Methoden der ökonomischen Bewertung in diesem Kontext erfüllen müssen sowie im Sinne der gestaltungsorientierten Wirtschaftsinformatik-Forschung konzeptionell einen Lösungsbeitrag zu skizzieren. Als Forschungsfragen ergeben sich daraus:

1. Welchen Anforderungen muss eine ökonomische Bewertung von In-Memory-basierten betrieblichen Informationssystemen genügen?
2. Inwieweit erfüllen bekannte Methoden der ökonomischen Bewertung diese Anforderungen?
3. Wie lässt sich ggf. auf Basis existierender Methoden und Erkenntnisse eine Bewertungsmethodik konstruieren, welche die Anforderungen an eine ökonomische Bewertung von In-Memory-basierten betrieblichen Informationssystemen erfüllt?

## **2 Abgrenzung des Forschungsgegenstands**

### **2.1 In-Memory-basierte Informationssysteme**

Unter In-Memory-Datenbanken versteht man Datenbankmanagementsysteme, die im Gegensatz zu herkömmlichen Datenbankmanagementsystemen nicht Festplattenlaufwerke, sondern den Arbeitsspeicher eines Computers zur Datenspeicherung nutzen [GS92]. Informationssysteme, die In-Memory-Datenbanken nutzen (In-Memory-basierte Informationssysteme (fortan mit IMIS abgekürzt)) können Daten sehr schnell verarbeiten, weil die Zugriffsgeschwindigkeiten auf den Arbeitsspeicher signifikant höher sind als auf Festplattenlaufwerke. Hinzu kommen spaltenbasierte Datenstrukturen und spezielle Kompressionsverfahren, welche die Zugriffszeiten noch weiter verkürzen.

Das Thema ist grundsätzlich nicht neu. Neuen Antrieb erhält es jedoch durch technologische Entwicklungen, insbes. bessere Kosten-Nutzen-Verhältnisse von Arbeitsspeicherelementen durch Fortschritte bei der Produktentwicklung und in der Produktion sowie einer besseren Ausnutzung der Kapazitäten durch Grid bzw. Cloud Computing. Zudem investieren führende Softwareanbieter viel in die entsprechende Produktentwicklung und Vermarktung von IMIS, wie z. B. SAP HANA, IBM solidDB oder Oracle TimesTen.

## **2.2 Ökonomische Bewertung**

Wertorientierte Unternehmensführung hat sich in marktwirtschaftlich geprägten Wirtschaftsordnungen als Leitbegriff durchgesetzt. Vereinfacht ersetzt die (nachhaltige) risikoadäquate Kapitalverzinsung dabei eine (kurzfristige) Gewinnorientierung. Im Mittelpunkt stehen der Marktwert des Eigenkapitals und damit verbunden das langfristige Überleben eines Betriebs. In den letzten Jahren entwickelten sich viele wertorientierten Steuerungskonzepte, die sich im Wesentlichen hinsichtlich ihrer Periodizität und den Rechengrößen (buchwert- oder zahlungsorientiert) unterscheiden [CS07, Di06].

Auf eine ausführliche Diskussion dazu verzichtet der Beitrag, um den vorgegebenen Rahmen nicht zu sprengen. Für das hier behandelte Anliegen reicht es an dieser Stelle zunächst festzuhalten, dass das Bewertungskriterium für IMIS aus den oben genannten Gründen in einen Wertbeitrag münden sollte. Wie dieser genau berechnet wird ist Gegenstand der weiteren Operationalisierung.

## **3 Anforderungen an Methoden zur ökonomischen Bewertung von In-Memory-basierten Informationssystemen**

Aus dem in den beiden vorhergehenden Kapiteln geschilderten Problemhaushalt ergeben sich folgende Anforderungen im Sinne von Formalzielen an Methoden zur ökonomischen Bewertung von IMIS.

### **(A1) Wertorientierung als Prinzip ökonomischer Bewertung**

Wie eben erläutert sollte Ziel der Bewertung eine nachhaltigen (Gesamt-) Unternehmenswertsteigerung sein. Somit ist eine Zielformulierung, die den Unternehmenswert und Risiken adäquat abbildet nötig. Hinzu kommt die Nebenbedingung, dass bei aller Bedeutung der langfristigen Perspektive die kurzfristige Liquidität und damit die Überlebensfähigkeit des Betriebs gewährleistet bleiben muss.

### **(A2) Berücksichtigung dynamischer Effekte**

Die Forderung nach Wertorientierung führt zu einer mehrperiodigen Analyse verschiedener Faktoren, die sich aus betrieblicher Perspektive mehr oder weniger

beeinflussen lassen. Symptomatisch dafür sind Rückkopplungseffekte, die sich im Zeitverlauf selbst (ohne direkte Handlung des betrachteten Betriebs) bis zu bestimmten Grenzen verstärken oder abschwächen können. Ein Beispiel dafür sind Mund-zu-Mund-Propaganda-Effekte. Je mehr zufriedene Kunden ein Unternehmen hat, desto mehr tragen diese über positive Erfahrungsberichte dazu bei, dass die Zahl der Kunden – auch ohne direktes Marketing – wächst. Dies kann sich in mehreren Perioden bis zu einer gewissen durch den Markt vorgegebenen Grenze wiederholen. Ebenso besteht die Gefahr des gegenteiligen Effekts, wenn unzufriedenen Kunden – insbesondere etwa in Sozialen-Online-Netzwerken – ihren Unmut mit Produkten und Dienstleistungen einer breiten Masse an potenziellen Kunden mitteilen.

**(A3) Transparenz i.S.v. Offenlegung von Unsicherheiten und Annahmen**

Aufgrund der Größe, Dynamik und Komplexität der potenziellen Einsatzfelder für IMIS im Kontext betrieblicher Informationsverarbeitung und somit naturgemäß auch den damit verbundenen Kosten- sowie Nutzeffekten erscheint eine objektiv vollständige und eindeutig richtige Bewertung aufgrund der inhärenten Unsicherheit unerreichbar. Diese Unsicherheit umfasst primär vier Felder, die Abb. 1 zeigt.

<b>Kosten</b>	(A)	(C)
<b>Nutzen</b>	(B)	(D)
	<b>Prognoseunsicherheit</b>	<b>Zuordnungsunsicherheit</b>

Abb. 1: Relevante Dimensionen der Unsicherheit bei der ökonomischen Bewertung von In-Memory-basierten betrieblichen Informationssystemen

Zu unterscheiden sind Prognose- und Zuordnungsunsicherheit, die sowohl die Kosten als auch den Nutzen betreffen. Die Prognoseunsicherheit bezieht sich darauf, dass es bei der Einschätzung zu künftigen Kosten ((A) etwa Energiepreissteigerung für Klimaanlage zur Rechenzentrumskühlung) sowie erwarteten Nutzen ((B) z. B. erhöhte Zahlungsbereitschaft und damit Umsatzsteigerung durch schnelleren Service) zu Abweichungen von der Realität kommen kann.

Verschärft wird diese Unsicherheit durch die Problematik einer verursachungsgerechten Zuordnung von Kosten- und Nutzeffekten. In der Kostenrechnung ist die Suche nach dem „richtigen“ Verursachungsprinzip ein Dauerthema, insbes. zwischen den Fraktionen der Befürworter von Voll- und Teilkostenrechnung (C). Bei der Fokussierung auf die Kostenseite tritt häufig in den Hintergrund, dass es bei den oft nur indirekt beobachtbaren Nutzeffekten von IT-Investitionen noch schwieriger erscheint, diese fundiert verursachungsgerecht zuzuordnen (D). Wesentliche Nutzeffekte von IMIS verspricht man sich beispielsweise durch Geschwindigkeitssteigerungen. Doch gerade bei diesen ist es unklar, inwiefern sie ursächlich sind für Veränderungen des Unternehmenswerts.

#### **(A4) Operationalisierung für In-Memory-basierte Informationssysteme**

Es gibt – wie die folgende Analyse des erreichten Standes zeigt – viele Ansätze zu Themen im Feld der Nutzenbewertung von Informationstechnologie bzw. Economics of Information Systems. Teilweise sind diese systemneutral, teilweise systemspezifisch, z. B. fokussiert auf Enterprise-Resource-Planning-Systeme (ERP-Systeme). Ziel des Beitrags ist es, Wissenschaftlern und Praktikern eine möglichst konkrete Hilfestellung bei der Bewertung von IMIS zu geben. Daher ist es zielführend, wenn die Methodik konkret für diese Systeme operationalisiert wird, d. h. spezifische direkte und indirekte Kosten (Total Cost of Ownership) sowie damit einhergehende Nutzeffekte explizit berücksichtigt.

### **4 Überblick zum erreichten Stand**

Eine Literaturrecherche in gängigen Datenbanken wie AIS Electronic Library, EBSCOhost, EmeraldInsight, IEEEExplore, ProQuest, ScienceDirect, SpringerLink ergab, dass dort zwar rund 20 fundiertere wissenschaftliche Publikationen zu IMIS zu finden sind. Sie alle legen den Fokus jedoch eher auf technische Themen (v. a. Antwortzeitverhalten, Datensicherung und Datenwiederherstellung) und behandeln ökonomische Aspekte allenfalls als Randthema bzw. bestätigen, dass hier noch Herausforderungen liegen [PI09, DM11, PZ11].

Im nächsten Schritt galt es somit, den Suchraum zu erweitern. Schnell gelangt man zu einer Fülle genereller Ansätze zur ökonomischen Bewertung von Informationssystemen (Economics of Information Systems), deren umfängliche Behandlung ein Feld für eigene Aufsätze bis hin zu größeren Monographien eröffnet. Da dies nicht Kerngegenstand dieses Beitrags ist, dient hier als Basis eine aktuelle Meta-Analyse von „Literatur-Reviews“ sowie Primärliteratur zum Thema „Ökonomischer Wert von Informationssystemen“ von Schryen [Sc10]. Zentrales Ergebnis dieser Studie ist, dass es noch eine Reihe von Forschungslücken in Bezug auf Wettbewerbswirkungen, Verzögerungswirkungen, Unsicherheit gibt.

Es stellt sich also die Frage, welche der existierenden Ansätze das Potenzial bieten, so erweitert bzw. angepasst zu werden, dass die im vorhergehenden Kapitel entwickelten Anforderungen erfüllt werden können.

Eine vielversprechende Alternative in diesem Sinn, weil sie als eine der wenigen im Vergleich zu vielen anderen explizit die im Rahmen der Anforderung 3 thematisierten Unsicherheiten bei Prognose sowie Zuordnung von Kosten- und Nutzeffekten thematisiert fand sich mit dem Rahmenkonzept „Wirtschaftlichkeitsanalyse mit Risikostufen (WARS)“ von Ott [Ot93]. Er teilt den Nutzen von Informationstechnologie in folgende Kategorien ein:

1. Direkter Nutzen  
entsteht durch Einsparung von bisher anfallenden direkten Kosten und ist damit leicht bewertbar



2. Indirekter Nutzen  
entsteht durch Einsparung von zukünftigen – auch indirekten – Kosten und ist somit schwerer bewertbar als direkter Nutzen
3. Schwer fassbarer Nutzen  
entsteht primär durch zusätzliche Erlöse und wird auch oft als „strategischer Nutzen“ bezeichnet. Ursache-Wirkungsbeziehungen sind hier mit sehr hoher Unsicherheit behaftet, sodass diese Nutzenkategorie sehr schwer bewertbar ist

Zusätzlich unterteilt Ott den Nutzen hinsichtlich der Realisierungswahrscheinlichkeit in drei weitere Kategorien, sodass sich letztlich eine 3x3-Matrix ergibt, in der die Felder mit den entsprechenden monetären Äquivalenten nach Risikostufen 1-9 sortiert sind. Analog behandelt der Ansatz die Kostenseite, sodass auch hier eine entsprechende Reihenfolge von monetären Äquivalenten entsteht. Grundgedanke ist, dass ein pessimistischer Entscheider einerseits nur Nutzeffekte akzeptieren würde, die sicher eintreten und eindeutig der neuen Informationstechnologie zuordenbar sind, andererseits dagegen auch Kosten berücksichtigen würde, deren Anfall unsicher und deren Zuordnung zur Investitionsalternative nur vage möglich ist. Dies entspräche den jeweils kumulierten Werten der Risikostufen 1 von 9. Angewendet und validiert hat Ott sein Modell anhand eines Computer-Aided-Software-Engineering-Systems (CASE-Systems) in einem mittelständischen Unternehmen [Ot93].

## **5 Grundkonzept einer ökonomischen Bewertung von In-Memory-basierten Informationssystemen auf Basis des WARS-Ansatzes von Ott**

Das eben erläuterte WARS-Modell bietet grundsätzlich die Möglichkeit, neben der Forderung nach Transparenz bez. Unsicherheiten auch die Forderungen nach Wertorientierung, Berücksichtigung dynamischer Effekte sowie Operationalisierung für IMIS zu erfüllen; auch wenn Ott dies selbst noch nicht explizit realisierte. Deshalb nutzt es dieser Beitrag als theoretisches Fundament für die Entwicklung eines Ansatzes zur ökonomischen Bewertung von IMIS. Im Unterschied zum originären Modell schlagen die Autoren dieses Beitrags vor, es wie folgt weiterzuentwickeln:

1. Im Sinne der Wertorientierung werden die pagatorischen Größen Ein- und Auszahlungen statt der Begriffe Kosten und Nutzen verwendet. Damit wird das Problem der periodengerechten Abgrenzung entschärft. Aufgrund der Mehrperiodigkeit ist mit Barwerten zu rechnen.
2. Exemplarisch werden im Folgenden den verschiedenen Risikostufen Ursachen für Ein- und Auszahlungen zugeordnet, die im Rahmen von IMIS relevant sind.
3. Zur Kalkulation der monetären Äquivalente können als Erweiterung dort, wo dynamische Verzögerungs- und Rückkopplungseffekte auftreten Verfahren der dynamischen Simulation (z. B. System Dynamics) dazu beitragen, die Prognosegüte für die Barwerte der relevanten Ein- und Auszahlungen zu verbessern.

Somit ergibt sich folgende Systematik, die im Rahmen dieses Beitrags nur stark vereinfacht dargestellt werden kann. Die in den Tabellen 2 und 4 aufgeführten Ein- und Auszahlungsursachen sind hier lediglich exemplarisch den Kategorien aus den Matrizen aus den Tabellen 1 und 3 zugeordnet. Eine Zuordnung zu anderen Risikostufen ist abhängig von den realen Gegebenheiten ebenso denkbar. Existiert beispielsweise ein eigener Server-Raum für das IMIS, so wären die Auszahlungen für Klimatisierung direkt zurechenbar (Stufe 7/Tab. 4). Wird in dem Raum auch Hardware für andere Informationssysteme betrieben, wären die Auszahlungen indirekt zurechenbar (Stufe 5/Tab. 4).

**Bewertung positiver ökonomischer Effekte**

		Realisierungswahrscheinlichkeit		
		hoch	mittel	niedrig
Zurechenbarkeit	direkt	(1)	(3)	(6)
	indirekt	(2)	(5)	(8)
	vage	(4)	(7)	(9)

Tab. 1: Risikostufen positiver ökonomischer Effekte

Risiko-stufe	Exemplarische Ursachen für Einzahlungen bzw. Wegfall von Auszahlungen
1	Wegfall von Auszahlungen für nicht mehr benötigten Festplattenbetrieb, geringere Auszahlungen/Einzahlungsverluste durch schnellere Reaktion bei Unternehmenskrisen
2	schnellere Verfügbarkeit analytischer Informationen am Point of Sale, z. B. Call-Center oder Flughafen-Gate und so höhere Ausschöpfung von Kundenwertpotenzial
3	Wegfall von Auszahlungen für Datenqualitätssicherung bei Extraktions-Transformations-Lade-Prozessen (ETL) sowie für Maßnahmen zur Verbesserung von Abfragezeiten (z. B. materialisierte Sichten), geringere Architektur-Komplexität für Informationstechnologie, schnellere Kompensation von Ausfällen im Supply Chain Management (Supply Chain Event Management mit Real-time RFID)
4	schnellere Verfügbarkeit analytischer Informationen bei Reklamationsbearbeitung und so Erhöhung der Kundenbindung bei Kunden mit hohem Wertbeitragspotenzial
5	schnellere Sperrung von Konten bei Missbrauchsverdacht/Verdacht auf Forderungsausfall und so Verringerung von Einzahlungsverlusten, zusätzliche Einzahlungen durch neue Geschäftsmodelle im Energiesektor (Smart Metering)
6	schnellere Reaktion auf Preisschwankungen an Finanz- und Rohstoffmärkten, Verringerung von Out-of-Shelf-Problemen im Handel (in Kombination mit RFID)
7	höhere Motivation der IMIS-Nutzer durch weniger Frust bei Wartezeiten auf Antwort
8	schnellere Reaktion auf Gerüchte („Word of Mouth“) in Sozialen-Online-Netzwerken
9	positives Image durch schnellere Verarbeitung von Kundenanfragen

Tab. 2: Zuordnung konkreter Beispiele positiver Effekte im Kontext von In-Memory-basierten Informationssystemen [PZ11]

### Bewertung negativer ökonomischer Effekte

		Realisierungswahrscheinlichkeit		
		hoch	mittel	niedrig
Zurechenbarkeit	direkt	(9)	(7)	(4)
	indirekt	(8)	(5)	(2)
	vage	(6)	(3)	(1)

Tab. 3: Risikostufen negativer ökonomischer Effekte

Risiko- stufe	Exemplarische Ursachen für Auszahlungen bzw. Wegfall von Einzahlungen
9	Lizenzgebühren für IMIS, Hochleistungs-Hardware (Arbeitsspeicher), Datenmigration
8	Beratungsleistungen für Schnittstellenanpassungen in Quellsystemen
7	komplexere Systemadministration, häufigere Datensicherungen, Zusatzschulungen
6	Bedienfehler aufgrund mangelnder Kenntnisse/Akzeptanz des IMIS
5	leistungsstärkere Klimatisierung, höherer Energieeinsatz beim IMIS-Betrieb
4	Erfüllung zusätzlicher regulatorischer Auflagen beim Einsatz von IMIS
3	verstärktes Risikomanagement in der Anlaufphase
2	komplexe Datenwiederherstellung nach Systemausfall
1	Fluktuation von Mitarbeitern

Tab. 4: Zuordnung konkreter Beispiele negativer Effekte im Kontext von In-Memory-basierten Informationssystemen

Trägt man die kumulierten Barwertreihen aus beiden Risikostufen-Matrizen grafisch ab (siehe Abb. 2) wird die ökonomische Sinnhaftigkeit intuitiv sichtbar. Sinnvoll ist eine Alternative grundsätzlich, wenn die Funktion negativer Effekte bei allen 9 Risikostufen unter der Funktion positiver Effekte liegt und vice versa. Gibt es einen Schnittpunkt im pessimistischen Bereich (niedrige Risikostufe), würden auch risikoaverse Entscheider zustimmen. Liegt er dagegen bei einer hohen Risikostufe, würden nur noch sehr optimistische, risikofreudige Entscheider investieren [Ot93].

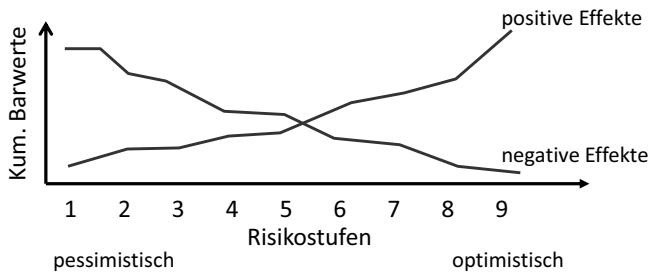


Abb. 2: Grafische Auswertung (in Anlehnung an [Ot93])

## 6 Zusammenfassende Evaluation und Ausblick

Gemäß den im ersten Kapitel entwickelten Forschungsfragen wurden Anforderungen an eine ökonomische Bewertung von IMIS entwickelt und festgestellt, dass v. a. in Bezug auf die Transparenz der Prognose- und Zurechnungsverlässlichkeit sowie dynamische Effekte noch Forschungsbedarf besteht. Das Rahmenkonzept WARS von Ott trat bei der Literaturrecherche als vielversprechendes Fundament für einen Lösungsansatz hervor und bildete daher die Basis für einen Weiterentwicklungsvorschlag für IMIS.

Aufgrund der noch vergleichsweise geringen Verbreitung von IMIS im betrieblichen Produktiveinsatz fehlt aktuell eine ausreichend breite Grundgesamtheit für eine praktische ex-post Evaluation des Ansatzes. Somit bleibt aus wissenschaftlicher Sicht im Moment nur eine Evaluation in Bezug auf Formalkriterien (Anforderungen A1 bis A4 aus Kapitel 3).

Die Anpassung bez. barwertiger Ein- und Auszahlungen trägt zur Wertorientierung bei (A1). Noch nicht explizit berücksichtigt ist jedoch die Nebenbedingung der kurzfristigen Liquiditätssicherung.

Die Transparenz bez. Prognose- und Zurechnungsverlässlichkeit (A3) wird durch die von Ott vorgeschlagenen Risikostufen-Matrizen geschaffen. Während die Position der Risikostufen (1) und (9) jeweils unstrittig erscheint, ist die Reihung der übrigen Kategorien durchaus ein Diskussionsthema. Ebenso problematisch ist die eindeutige Zuordnung der mit IMIS verbundenen barwertigen Ein- und Auszahlungen zu den jeweiligen Risikostufen (A3 und A4). Im Vergleich zu klassischen dynamischen Investitionsrechnungsverfahren, bei denen ggf. noch im Rahmen einer Szenarioanalyse „Best“ und „Worst Cases“ kalkuliert werden, bietet die hier höhere Transparenz aber einen gewissen Mehrwert.

Die Berücksichtigung dynamischer Verzögerungs- bzw. Rückkopplungseffekte (A2) konnte im begrenzten Rahmen dieses Beitrags nur angedeutet werden – durch den obigen Vorschlag, die Prognosegüte für die Ein- und Auszahlungsbarwerte durch System-Dynamics-Simulationen zu erhöhen. Beispiele zur Operationalisierung, etwa

bez. „Word-of-Mouth-Effekte“, die man mit IMIS schneller analysieren könnte, finden sich jedoch in der Literatur, etwa bei [MR10].

Insgesamt bleibt festzuhalten, dass dies die Skizze eines ersten, noch sehr rudimentären Ansatzes zur ökonomischen Bewertung von IMIS ist, bei dem noch viele Fragen offen bleiben. Es fehlt noch eine systematische Erhebung der in den Tabellen 2 und 4 skizzierten Beispiele sowie tiefergehende Analysen, etwa zur künftigen Rolle von ETL-Prozessen. Hinzu kommt, dass wegen der geringen Anzahl produktiv genutzter IMIS noch eine breite Basis zur praktischen Evaluation fehlt.

Aufgrund der eingangs beschriebenen Problematik, Relevanz und Dynamik des Forschungsfelds erscheint es dennoch sinnvoll, sich bereits jetzt intensiv mit Methoden der ökonomischen Bewertung von IMIS zu beschäftigen, um dazu beizutragen, diese schneller in einen nachhaltig sinnvollen Einsatz zu bringen, was sowohl im Interesse der Forschung, der Nutzer sowie auch der Softwarehersteller liegen dürfte.

## 7 Literaturverzeichnis

- [ACN09] Azvine, B.; Cui, Z.; Nauck, D. D.; Majeed, B.: Real Time Business Intelligence for the Adaptive Enterprise. In: Proceedings of the 8th IEEE International Conference on E-Commerce Technology and the 3rd IEEE International Conference on Enterprise Computing, E-Commerce and E-Services, Los Alamitos, 2006, S. 29-39.
- [CS07] Coenenberg, A.; Salfeld, R.: Wertorientierte Unternehmensführung, Schäffer Poeschel, Stuttgart, 2007.
- [DM11] Danciu, A.; Mladenova, V.: In-Memory Database Technology in the Context of Enterprise Applications. In (Krcmar, H. Hrsg.): 7th Workshop on Information Systems and Services Sciences, München. 2011; S. 10-18.
- [Di06] Dillerup, R.: Wertorientierte Unternehmensführung. In (Ernst, D.; Häcker, J.; Moser, U.; Auge-Dickhut, S., Hrsg.): Praxis der Unternehmensbewertung und Akquisitionsfinanzierung, Olzog, 2006; S. 8.1.1-8.1.34.
- [GS92] Garcia-Molina, H.; Salem, K.: Main Memory Database Systems: An Overview. In: IEEE Transactions on Knowledge and Data Engineering 1992 (6); S. 509-516.
- [MR10] Meier, M. C.; Reinwald, D.: A System Dynamics Approach to Value-Based Complaint Management Including Repurchase Behavior and Word of Mouth. In: Proceedings of the 18th European Conference on Information Systems, Pretoria, 2010; S. 1-13.
- [Ot93] Ott, H. J.: Wirtschaftlichkeitsanalyse von EDV-Investitionen mit dem WARS-Modell am Beispiel der Einführung von CASE. In: WIRTSCHAFTSINFORMATIK 1993 (6); S. 522-531.
- [PI09] Plattner, H.: A Common Database Approach for OLTP and OLAP Using an In-Memory Column Database. In: Proceedings of the 35th International SIGMOD Conference, Providence, 2009; S. 1-7.
- [PZ11] Plattner, H.; Zeier, A.: In-Memory Data Management: An Inflection Point for Enterprise Applications. Springer-Verlag, Heidelberg, 2011.
- [Sc10] Schryen, G.: Ökonomischer Wert von Informationssystemen - Beitrag von Literatur-Reviews zum Wissenserhalt. In: WIRTSCHAFTSINFORMATIK 2010 (4); S. 225-237.
- [SRR09] Steininger, K.; Riedl, R.; Roithmayr, F.; Mertens, P.: Moden und Trends in Wirtschaftsinformatik und Information Systems - Eine vergleichende Literaturanalyse. In WIRTSCHAFTSINFORMATIK 2009 (6); S. 478-495.

## *GI-Edition Lecture Notes in Informatics*

- P-1 Gregor Engels, Andreas Oberweis, Albert Zündorf (Hrsg.): Modellierung 2001.
- P-2 Mikhail Godlevsky, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications, ISTA'2001.
- P-3 Ana M. Moreno, Reind P. van de Riet (Hrsg.): Applications of Natural Language to Information Systems, NLDB'2001.
- P-4 H. Wörn, J. Mühling, C. Vahl, H.-P. Meinzer (Hrsg.): Rechner- und sensorgestützte Chirurgie; Workshop des SFB 414.
- P-5 Andy Schürr (Hrsg.): OMER – Object-Oriented Modeling of Embedded Real-Time Systems.
- P-6 Hans-Jürgen Appelrath, Rolf Beyer, Uwe Marquardt, Heinrich C. Mayr, Claudia Steinberger (Hrsg.): Unternehmen Hochschule, UH'2001.
- P-7 Andy Evans, Robert France, Ana Moreira, Bernhard Rumpe (Hrsg.): Practical UML-Based Rigorous Development Methods – Countering or Integrating the extremists, pUML'2001.
- P-8 Reinhard Keil-Slawik, Johannes Magenheimer (Hrsg.): Informatikunterricht und Medienbildung, INFOS'2001.
- P-9 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Innovative Anwendungen in Kommunikationsnetzen, 15. DFN Arbeitstagung.
- P-10 Mirjam Minor, Steffen Staab (Hrsg.): 1st German Workshop on Experience Management: Sharing Experiences about the Sharing Experience.
- P-11 Michael Weber, Frank Kargl (Hrsg.): Mobile Ad-Hoc Netzwerke, WMAN 2002.
- P-12 Martin Glinz, Günther Müller-Luschnat (Hrsg.): Modellierung 2002.
- P-13 Jan von Knop, Peter Schirmbacher and Viljan Mahni\_ (Hrsg.): The Changing Universities – The Role of Technology.
- P-14 Robert Tolksdorf, Rainer Eckstein (Hrsg.): XML-Technologien für das Semantic Web – XSW 2002.
- P-15 Hans-Bernd Bludau, Andreas Koop (Hrsg.): Mobile Computing in Medicine.
- P-16 J. Felix Hampe, Gerhard Schwabe (Hrsg.): Mobile and Collaborative Business 2002.
- P-17 Jan von Knop, Wilhelm Haverkamp (Hrsg.): Zukunft der Netze –Die Verletzbarkeit meistern, 16. DFN Arbeitstagung.
- P-18 Elmar J. Sinz, Markus Plaha (Hrsg.): Modellierung betrieblicher Informationssysteme – MobiS 2002.
- P-19 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3. Okt. 2002 in Dortmund.
- P-20 Sigrid Schubert, Bernd Reusch, Norbert Jesse (Hrsg.): Informatik bewegt – Informatik 2002 – 32. Jahrestagung der Gesellschaft für Informatik e.V. (GI) 30.Sept.-3. Okt. 2002 in Dortmund (Ergänzungsband).
- P-21 Jörg Desel, Mathias Weske (Hrsg.): Promise 2002: Prozessorientierte Methoden und Werkzeuge für die Entwicklung von Informationssystemen.
- P-22 Sigrid Schubert, Johannes Magenheimer, Peter Hubwieser, Torsten Brinda (Hrsg.): Forschungsbeiträge zur "Didaktik der Informatik" – Theorie, Praxis, Evaluation.
- P-23 Thorsten Spitta, Jens Borchers, Harry M. Sneed (Hrsg.): Software Management 2002 – Fortschritt durch Beständigkeit
- P-24 Rainer Eckstein, Robert Tolksdorf (Hrsg.): XMIDX 2003 – XML-Technologien für Middleware – Middleware für XML-Anwendungen
- P-25 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Commerce – Anwendungen und Perspektiven – 3. Workshop Mobile Commerce, Universität Augsburg, 04.02.2003
- P-26 Gerhard Weikum, Harald Schöning, Erhard Rahm (Hrsg.): BTW 2003: Datenbanksysteme für Business, Technologie und Web
- P-27 Michael Kroll, Hans-Gerd Lipinski, Kay Melzer (Hrsg.): Mobiles Computing in der Medizin
- P-28 Ulrich Reimer, Andreas Abecker, Steffen Staab, Gerd Stumme (Hrsg.): WM 2003: Professionelles Wissensmanagement – Erfahrungen und Visionen
- P-29 Antje Düsterhöft, Bernhard Thalheim (Eds.): NLDB'2003: Natural Language Processing and Information Systems
- P-30 Mikhail Godlevsky, Stephen Liddle, Heinrich C. Mayr (Eds.): Information Systems Technology and its Applications
- P-31 Arslan Brömmke, Christoph Busch (Eds.): BIOSIG 2003: Biometrics and Electronic Signatures

- P-32 Peter Hubwieser (Hrsg.): Informatische Fachkonzepte im Unterricht – INFOS 2003
- P-33 Andreas Geyer-Schulz, Alfred Taudes (Hrsg.): Informationswirtschaft: Ein Sektor mit Zukunft
- P-34 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 1)
- P-35 Klaus Dittrich, Wolfgang König, Andreas Oberweis, Kai Rannenberg, Wolfgang Wahlster (Hrsg.): Informatik 2003 – Innovative Informatikanwendungen (Band 2)
- P-36 Rüdiger Grimm, Hubert B. Keller, Kai Rannenberg (Hrsg.): Informatik 2003 – Mit Sicherheit Informatik
- P-37 Arndt Bode, Jörg Desel, Sabine Rathmayer, Martin Wessner (Hrsg.): DeLFI 2003: e-Learning Fachtagung Informatik
- P-38 E.J. Sinz, M. Plaha, P. Neckel (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2003
- P-39 Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.): IT-Incident Management & IT-Forensics – IMF 2003
- P-40 Michael Rebstock (Hrsg.): Modellierung betrieblicher Informationssysteme – MobIS 2004
- P-41 Uwe Brinkschulte, Jürgen Becker, Dietmar Fey, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle, Thomas Runkler (Edts.): ARCS 2004 – Organic and Pervasive Computing
- P-42 Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Economy – Transaktionen und Prozesse, Anwendungen und Dienste
- P-43 Birgitta König-Ries, Michael Klein, Philipp Obreiter (Hrsg.): Persistence, Scalability, Transactions – Database Mechanisms for Mobile Applications
- P-44 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): Security, E-Learning. E-Services
- P-45 Bernhard Rumpe, Wolfgang Hesse (Hrsg.): Modellierung 2004
- P-46 Ulrich Flegel, Michael Meier (Hrsg.): Detection of Intrusions of Malware & Vulnerability Assessment
- P-47 Alexander Prosser, Robert Krimmer (Hrsg.): Electronic Voting in Europe – Technology, Law, Politics and Society
- P-48 Anatoly Doroshenko, Terry Halpin, Stephen W. Liddle, Heinrich C. Mayr (Hrsg.): Information Systems Technology and its Applications
- P-49 G. Schiefer, P. Wagner, M. Morgenstern, U. Rickert (Hrsg.): Integration und Datensicherheit – Anforderungen, Konflikte und Perspektiven
- P-50 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 1) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm
- P-51 Peter Dadam, Manfred Reichert (Hrsg.): INFORMATIK 2004 – Informatik verbindet (Band 2) Beiträge der 34. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 20.-24. September 2004 in Ulm
- P-52 Gregor Engels, Silke Seehusen (Hrsg.): DELFI 2004 – Tagungsband der 2. e-Learning Fachtagung Informatik
- P-53 Robert Giegerich, Jens Stoye (Hrsg.): German Conference on Bioinformatics – GCB 2004
- P-54 Jens Borchers, Ralf Kneuper (Hrsg.): Softwaremanagement 2004 – Outsourcing und Integration
- P-55 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): E-Science and Grid Ad-hoc-Netze Medienintegration
- P-56 Fernand Feltz, Andreas Oberweis, Benoit Otjacques (Hrsg.): EMISA 2004 – Informationssysteme im E-Business und E-Government
- P-57 Klaus Turowski (Hrsg.): Architekturen, Komponenten, Anwendungen
- P-58 Sami Beydeda, Volker Gruhn, Johannes Mayer, Ralf Reussner, Franz Schweiggert (Hrsg.): Testing of Component-Based Systems and Software Quality
- P-59 J. Felix Hampe, Franz Lehner, Key Pousttchi, Kai Ranneberg, Klaus Turowski (Hrsg.): Mobile Business – Processes, Platforms, Payments
- P-60 Steffen Friedrich (Hrsg.): Unterrichtskonzepte für informatische Bildung
- P-61 Paul Müller, Reinhard Gotzhein, Jens B. Schmitt (Hrsg.): Kommunikation in verteilten Systemen
- P-62 Federrath, Hannes (Hrsg.): „Sicherheit 2005“ – Sicherheit – Schutz und Zuverlässigkeit
- P-63 Roland Kaschek, Heinrich C. Mayr, Stephen Liddle (Hrsg.): Information Systems – Technology and its Applications

- P-64 Peter Liggesmeyer, Klaus Pohl, Michael Goedicke (Hrsg.): Software Engineering 2005
- P-65 Gottfried Vossen, Frank Leymann, Peter Lockemann, Wolfried Stucky (Hrsg.): Datenbanksysteme in Business, Technologie und Web
- P-66 Jörg M. Haake, Ulrike Lucke, Djamshid Tavangarian (Hrsg.): DeLFI 2005: 3. deutsche e-Learning Fachtagung Informatik
- P-67 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 1)
- P-68 Armin B. Cremers, Rainer Manthey, Peter Martini, Volker Steinhage (Hrsg.): INFORMATIK 2005 – Informatik LIVE (Band 2)
- P-69 Robert Hirschfeld, Ryszard Kowalczyk, Andreas Polze, Matthias Weske (Hrsg.): NODE 2005, GSEM 2005
- P-70 Klaus Turowski, Johannes-Maria Zaha (Hrsg.): Component-oriented Enterprise Application (COAE 2005)
- P-71 Andrew Torda, Stefan Kurz, Matthias Rarey (Hrsg.): German Conference on Bioinformatics 2005
- P-72 Klaus P. Jantke, Klaus-Peter Fähnrich, Wolfgang S. Wittig (Hrsg.): Marktplatz Internet: Von e-Learning bis e-Payment
- P-73 Jan von Knop, Wilhelm Haverkamp, Eike Jessen (Hrsg.): "Heute schon das Morgen sehen"
- P-74 Christopher Wolf, Stefan Lucks, Po-Wah Yau (Hrsg.): WEWoRC 2005 – Western European Workshop on Research in Cryptology
- P-75 Jörg Desel, Ulrich Frank (Hrsg.): Enterprise Modelling and Information Systems Architecture
- P-76 Thomas Kirste, Birgitta König-Riess, Key Pousttchi, Klaus Turowski (Hrsg.): Mobile Informationssysteme – Potentiale, Hindernisse, Einsatz
- P-77 Jana Dittmann (Hrsg.): SICHERHEIT 2006
- P-78 K.-O. Wenkel, P. Wagner, M. Morgens-tern, K. Luzi, P. Eisermann (Hrsg.): Land- und Ernährungswirtschaft im Wandel
- P-79 Bettina Biel, Matthias Book, Volker Gruhn (Hrsg.): Softwareengineering 2006
- P-80 Mareike Schoop, Christian Huemer, Michael Rebstock, Martin Bichler (Hrsg.): Service-Oriented Electronic Commerce
- P-81 Wolfgang Karl, Jürgen Becker, Karl-Erwin Großpietsch, Christian Hochberger, Erik Maehle (Hrsg.): ARCS'06
- P-82 Heinrich C. Mayr, Ruth Breu (Hrsg.): Modellierung 2006
- P-83 Daniel Huson, Oliver Kohlbacher, Andrei Lupas, Kay Nieselt and Andreas Zell (eds.): German Conference on Bioinformatics
- P-84 Dimitris Karagiannis, Heinrich C. Mayr, (Hrsg.): Information Systems Technology and its Applications
- P-85 Witold Abramowicz, Heinrich C. Mayr, (Hrsg.): Business Information Systems
- P-86 Robert Krimmer (Ed.): Electronic Voting 2006
- P-87 Max Mühlhäuser, Guido Röbling, Ralf Steinmetz (Hrsg.): DELFI 2006: 4. e-Learning Fachtagung Informatik
- P-88 Robert Hirschfeld, Andreas Polze, Ryszard Kowalczyk (Hrsg.): NODE 2006, GSEM 2006
- P-90 Joachim Schelp, Robert Winter, Ulrich Frank, Bodo Rieger, Klaus Turowski (Hrsg.): Integration, Informationslogistik und Architektur
- P-91 Henrik Stormer, Andreas Meier, Michael Schumacher (Eds.): European Conference on eHealth 2006
- P-92 Fernand Feltz, Benoît Otjacques, Andreas Oberweis, Nicolas Poussing (Eds.): AIM 2006
- P-93 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 1
- P-94 Christian Hochberger, Rüdiger Liskowsky (Eds.): INFORMATIK 2006 – Informatik für Menschen, Band 2
- P-95 Matthias Weske, Markus Nüttgens (Eds.): EMISA 2005: Methoden, Konzepte und Technologien für die Entwicklung von dienstbasierten Informationssystemen
- P-96 Saartje Brockmans, Jürgen Jung, York Sure (Eds.): Meta-Modelling and Ontologies
- P-97 Oliver Göbel, Dirk Schadt, Sandra Frings, Hardo Hase, Detlef Günther, Jens Nedon (Eds.): IT-Incident Mangament & IT-Forensics – IMF 2006



- P-98 Hans Brandt-Pook, Werner Simonsmeier und Thorsten Spitta (Hrsg.): Beratung in der Softwareentwicklung – Modelle, Methoden, Best Practices
- P-99 Andreas Schwill, Carsten Schulte, Marco Thomas (Hrsg.): Didaktik der Informatik
- P-100 Peter Forbrig, Günter Siegel, Markus Schneider (Hrsg.): HDI 2006: Hochschuldidaktik der Informatik
- P-101 Stefan Böttinger, Ludwig Theuvsen, Susanne Rank, Marlies Morgenstern (Hrsg.): Agrarinformatik im Spannungsfeld zwischen Regionalisierung und globalen Wertschöpfungsketten
- P-102 Otto Spaniol (Eds.): Mobile Services and Personalized Environments
- P-103 Alfons Kemper, Harald Schöning, Thomas Rose, Matthias Jarke, Thomas Seidl, Christoph Quix, Christoph Brochhaus (Hrsg.): Datenbanksysteme in Business, Technologie und Web (BTW 2007)
- P-104 Birgitta König-Ries, Franz Lehner, Rainer Malaka, Can Türker (Hrsg.) MMS 2007: Mobilität und mobile Informationssysteme
- P-105 Wolf-Gideon Bleek, Jörg Raasch, Heinz Züllighoven (Hrsg.) Software Engineering 2007
- P-106 Wolf-Gideon Bleek, Henning Schwentner, Heinz Züllighoven (Hrsg.) Software Engineering 2007 – Beiträge zu den Workshops
- P-107 Heinrich C. Mayr, Dimitris Karagiannis (eds.) Information Systems Technology and its Applications
- P-108 Arslan Brömme, Christoph Busch, Detlef Hühnlein (eds.) BIOSIG 2007: Biometrics and Electronic Signatures
- P-109 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 1
- P-110 Rainer Koschke, Otthein Herzog, Karl-Heinz Rödiger, Marc Ronthaler (Hrsg.) INFORMATIK 2007 Informatik trifft Logistik Band 2
- P-111 Christian Eibl, Johannes Magenheimer, Sigrid Schubert, Martin Wessner (Hrsg.) DeLFI 2007: 5. e-Learning Fachtagung Informatik
- P-112 Sigrid Schubert (Hrsg.) Didaktik der Informatik in Theorie und Praxis
- P-113 Sören Auer, Christian Bizer, Claudia Müller, Anna V. Zhdanova (Eds.) The Social Semantic Web 2007 Proceedings of the 1<sup>st</sup> Conference on Social Semantic Web (CSSW)
- P-114 Sandra Frings, Oliver Göbel, Detlef Günther, Hardo G. Hase, Jens Nedon, Dirk Schadt, Arslan Brömme (Eds.) IMF2007 IT-incident management & IT-forensics Proceedings of the 3<sup>rd</sup> International Conference on IT-Incident Management & IT-Forensics
- P-115 Claudia Falter, Alexander Schliep, Joachim Selbig, Martin Vingron and Dirk Walther (Eds.) German conference on bioinformatics GCB 2007
- P-116 Witold Abramowicz, Leszek Maciszek (Eds.) Business Process and Services Computing 1<sup>st</sup> International Working Conference on Business Process and Services Computing BPSC 2007
- P-117 Ryszard Kowalczyk (Ed.) Grid service engineering and management The 4<sup>th</sup> International Conference on Grid Service Engineering and Management GSEM 2007
- P-118 Andreas Hein, Wilfried Thoben, Hans-Jürgen Appelrath, Peter Jensch (Eds.) European Conference on health 2007
- P-119 Manfred Reichert, Stefan Strecker, Klaus Turowski (Eds.) Enterprise Modelling and Information Systems Architectures Concepts and Applications
- P-120 Adam Pawlak, Kurt Sandkuhl, Wojciech Cholewa, Leandro Soares Indrusiak (Eds.) Coordination of Collaborative Engineering - State of the Art and Future Challenges
- P-121 Korbinian Herrmann, Bernd Bruegge (Hrsg.) Software Engineering 2008 Fachtagung des GI-Fachbereichs Softwaretechnik
- P-122 Walid Maalej, Bernd Bruegge (Hrsg.) Software Engineering 2008 - Workshopband Fachtagung des GI-Fachbereichs Softwaretechnik

- P-123 Michael H. Breitner, Martin Breunig, Elgar Fleisch, Ley Pousttchi, Klaus Turowski (Hrsg.)  
Mobile und Ubiquitäre Informationssysteme – Technologien, Prozesse, Marktfähigkeit  
Proceedings zur 3. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2008)
- P-124 Wolfgang E. Nagel, Rolf Hoffmann, Andreas Koch (Eds.)  
9<sup>th</sup> Workshop on Parallel Systems and Algorithms (PASA)  
Workshop of the GI/ITG Special Interest Groups PARS and PARVA
- P-125 Rolf A.E. Müller, Hans-H. Sundermeier, Ludwig Theuvsen, Stephanie Schütze, Marlies Morgenstern (Hrsg.)  
Unternehmens-IT: Führungsinstrument oder Verwaltungsbürde  
Referate der 28. GIL Jahrestagung
- P-126 Rainer Gimnich, Uwe Kaiser, Jochen Quante, Andreas Winter (Hrsg.)  
10<sup>th</sup> Workshop Software Reengineering (WSR 2008)
- P-127 Thomas Kühne, Wolfgang Reisig, Friedrich Steimann (Hrsg.)  
Modellierung 2008
- P-128 Ammar Alkassar, Jörg Siekmann (Hrsg.)  
Sicherheit 2008  
Sicherheit, Schutz und Zuverlässigkeit  
Beiträge der 4. Jahrestagung des Fachbereichs Sicherheit der Gesellschaft für Informatik e.V. (GI)  
2.-4. April 2008  
Saarbrücken, Germany
- P-129 Wolfgang Hesse, Andreas Oberweis (Eds.)  
Sigsand-Europe 2008  
Proceedings of the Third AIS SIGSAND European Symposium on Analysis, Design, Use and Societal Impact of Information Systems
- P-130 Paul Müller, Bernhard Neumair, Gabi Dreö Rodosek (Hrsg.)  
1. DFN-Forum Kommunikationstechnologien Beiträge der Fachtagung
- P-131 Robert Krimmer, Rüdiger Grimm (Eds.)  
3<sup>rd</sup> International Conference on Electronic Voting 2008  
Co-organized by Council of Europe, Gesellschaft für Informatik and E-Voting, CC
- P-132 Silke Seehusen, Ulrike Lucke, Stefan Fischer (Hrsg.)  
DeLFI 2008:  
Die 6. e-Learning Fachtagung Informatik
- P-133 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)  
INFORMATIK 2008  
Beherrschbare Systeme – dank Informatik Band 1
- P-134 Heinz-Gerd Hegering, Axel Lehmann, Hans Jürgen Ohlbach, Christian Scheideler (Hrsg.)  
INFORMATIK 2008  
Beherrschbare Systeme – dank Informatik Band 2
- P-135 Torsten Brinda, Michael Fothe, Peter Hubwieser, Kirsten Schlüter (Hrsg.)  
Didaktik der Informatik – Aktuelle Forschungsergebnisse
- P-136 Andreas Beyer, Michael Schroeder (Eds.)  
German Conference on Bioinformatics GCB 2008
- P-137 Arslan Brömme, Christoph Busch, Detlef Hühnlein (Eds.)  
BIOSIG 2008: Biometrics and Electronic Signatures
- P-138 Barbara Dinter, Robert Winter, Peter Chamoni, Norbert Gronau, Klaus Turowski (Hrsg.)  
Synergien durch Integration und Informationslogistik  
Proceedings zur DW2008
- P-139 Georg Herzwurm, Martin Mikusz (Hrsg.)  
Industrialisierung des Software-Managements  
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschaftsinformatik
- P-140 Oliver Göbel, Sandra Frings, Detlef Günther, Jens Nedon, Dirk Schadt (Eds.)  
IMF 2008 - IT Incident Management & IT Forensics
- P-141 Peter Loos, Markus Nüttgens, Klaus Turowski, Dirk Werth (Hrsg.)  
Modellierung betrieblicher Informationssysteme (MobIS 2008)  
Modellierung zwischen SOA und Compliance Management
- P-142 R. Bill, P. Korduan, L. Theuvsen, M. Morgenstern (Hrsg.)  
Anforderungen an die Agrarinformatik durch Globalisierung und Klimaveränderung
- P-143 Peter Liggesmeyer, Gregor Engels, Jürgen Münch, Jörg Dörr, Norman Riegel (Hrsg.)  
Software Engineering 2009  
Fachtagung des GI-Fachbereichs Softwaretechnik

- P-144 Johann-Christoph Freytag, Thomas Ruf, Wolfgang Lehner, Gottfried Vossen (Hrsg.)  
Datenbanksysteme in Business, Technologie und Web (BTW)
- P-145 Knut Hinkelmann, Holger Wache (Eds.)  
WM2009: 5th Conference on Professional Knowledge Management
- P-146 Markus Bick, Martin Breunig, Hagen Höpfner (Hrsg.)  
Mobile und Ubiquitäre Informationssysteme – Entwicklung, Implementierung und Anwendung  
4. Konferenz Mobile und Ubiquitäre Informationssysteme (MMS 2009)
- P-147 Witold Abramowicz, Leszek Maciaszek, Ryszard Kowalczyk, Andreas Speck (Eds.)  
Business Process, Services Computing and Intelligent Service Management  
BPSC 2009 · ISM 2009 · YRW-MBP 2009
- P-148 Christian Erfurth, Gerald Eichler, Volkmar Schau (Eds.)  
9<sup>th</sup> International Conference on Innovative Internet Community Systems  
I<sup>2</sup>CS 2009
- P-149 Paul Müller, Bernhard Neumair, Gabi Dreö Rodosek (Hrsg.)  
2. DFN-Forum  
Kommunikationstechnologien  
Beiträge der Fachtagung
- P-150 Jürgen Münch, Peter Liggesmeyer (Hrsg.)  
Software Engineering  
2009 - Workshopband
- P-151 Armin Heinzl, Peter Dadam, Stefan Kirn, Peter Lockemann (Eds.)  
PRIMIUM  
Process Innovation for Enterprise Software
- P-152 Jan Mending, Stefanie Rinderle-Ma, Werner Esswein (Eds.)  
Enterprise Modelling and Information Systems Architectures  
Proceedings of the 3<sup>rd</sup> Int'l Workshop EMISA 2009
- P-153 Andreas Schwill, Nicolas Apostolopoulos (Hrsg.)  
Lernen im Digitalen Zeitalter  
DeLFI 2009 – Die 7. E-Learning Fachtagung Informatik
- P-154 Stefan Fischer, Erik Maehle, Rüdiger Reischuk (Hrsg.)  
INFORMATIK 2009  
Im Focus das Leben
- P-155 Arslan Brömmе, Christoph Busch, Detlef Hühnlein (Eds.)  
BIOSIG 2009: Biometrics and Electronic Signatures  
Proceedings of the Special Interest Group on Biometrics and Electronic Signatures
- P-156 Bernhard Koerber (Hrsg.)  
Zukunft braucht Herkunft  
25 Jahre »INFOS – Informatik und Schule«
- P-157 Ivo Grosse, Steffen Neumann, Stefan Posch, Falk Schreiber, Peter Stadler (Eds.)  
German Conference on Bioinformatics 2009
- P-158 W. Claupein, L. Theuvsen, A. Kämpf, M. Morgenstern (Hrsg.)  
Precision Agriculture  
Reloaded – Informationsgestützte Landwirtschaft
- P-159 Gregor Engels, Markus Luckey, Wilhelm Schäfer (Hrsg.)  
Software Engineering 2010
- P-160 Gregor Engels, Markus Luckey, Alexander Pretschner, Ralf Reussner (Hrsg.)  
Software Engineering 2010 – Workshopband  
(inkl. Doktorandensymposium)
- P-161 Gregor Engels, Dimitris Karagiannis, Heinrich C. Mayr (Hrsg.)  
Modellierung 2010
- P-162 Maria A. Wimmer, Uwe Brinkhoff, Siegfried Kaiser, Dagmar Lück-Schneider, Erich Schweighofer, Andreas Wiebe (Hrsg.)  
Vernetzte IT für einen effektiven Staat  
Gemeinsame Fachtagung  
Verwaltungsinformatik (FTVI) und  
Fachtagung Rechtsinformatik (FTRI) 2010
- P-163 Markus Bick, Stefan Eulgem, Elgar Fleisch, J. Felix Hampe, Birgitta König-Ries, Franz Lehner, Key Pousttchi, Kai Rannenberg (Hrsg.)  
Mobile und Ubiquitäre Informationssysteme  
Technologien, Anwendungen und Dienste zur Unterstützung von mobiler Kollaboration
- P-164 Arslan Brömmе, Christoph Busch (Eds.)  
BIOSIG 2010: Biometrics and Electronic Signatures  
Proceedings of the Special Interest Group on Biometrics and Electronic Signatures

- P-165 Gerald Eichler, Peter Kropf, Ulrike Lechner, Phayung Meesad, Herwig Unger (Eds.)  
10<sup>th</sup> International Conference on Innovative Internet Community Systems (I<sup>2</sup>CS) – Jubilee Edition 2010 –
- P-166 Paul Müller, Bernhard Neumair, Gabi Dreö Rodosek (Hrsg.)  
3. DFN-Forum Kommunikationstechnologien  
Beiträge der Fachtagung
- P-167 Robert Krimmer, Rüdiger Grimm (Eds.)  
4<sup>th</sup> International Conference on Electronic Voting 2010  
co-organized by the Council of Europe, Gesellschaft für Informatik und E-Voting.CC
- P-168 Ira Diethelm, Christina Dörge, Claudia Hildebrandt, Carsten Schulte (Hrsg.)  
Didaktik der Informatik  
Möglichkeiten empirischer Forschungsmethoden und Perspektiven der Fachdidaktik
- P-169 Michael Kerres, Nadine Ojstersek, Ulrik Schroeder, Ulrich Hoppe (Hrsg.)  
DeLFI 2010 - 8. Tagung der Fachgruppe E-Learning der Gesellschaft für Informatik e.V.
- P-170 Felix C. Freiling (Hrsg.)  
Sicherheit 2010  
Sicherheit, Schutz und Zuverlässigkeit
- P-171 Werner Esswein, Klaus Turowski, Martin Juhrisch (Hrsg.)  
Modellierung betrieblicher Informationssysteme (MobIS 2010)  
Modellgestütztes Management
- P-172 Stefan Klink, Agnes Koschmider, Marco Mevius, Andreas Oberweis (Hrsg.)  
EMISA 2010  
Einflussfaktoren auf die Entwicklung flexibler, integrierter Informationssysteme  
Beiträge des Workshops der GI-Fachgruppe EMISA (Entwicklungsmethoden für Informationssysteme und deren Anwendung)
- P-173 Dietmar Schomburg, Andreas Grote (Eds.)  
German Conference on Bioinformatics 2010
- P-174 Arslan Brömmel, Torsten Eymann, Detlef Hühnlein, Heiko Roßnagel, Paul Schmücker (Hrsg.)  
perspeGKtive 2010  
Workshop „Innovative und sichere Informationstechnologie für das Gesundheitswesen von morgen“
- P-175 Klaus-Peter Fährnrich, Bogdan Franczyk (Hrsg.)  
INFORMATIK 2010  
Service Science – Neue Perspektiven für die Informatik  
Band 1
- P-176 Klaus-Peter Fährnrich, Bogdan Franczyk (Hrsg.)  
INFORMATIK 2010  
Service Science – Neue Perspektiven für die Informatik  
Band 2
- P-177 Witold Abramowicz, Rainer Alt, Klaus-Peter Fährnrich, Bogdan Franczyk, Leszek A. Maciaszek (Eds.)  
INFORMATIK 2010  
Business Process and Service Science – Proceedings of ISSS and BPSC
- P-178 Wolfram Pietsch, Benedikt Krams (Hrsg.)  
Vom Projekt zum Produkt  
Fachtagung des GI-Fachausschusses Management der Anwendungsentwicklung und -wartung im Fachbereich Wirtschaftsinformatik (WI-MAW), Aachen, 2010
- P-179 Stefan Gruner, Bernhard Rumpe (Eds.)  
FM+AM'2010  
Second International Workshop on Formal Methods and Agile Methods
- P-180 Theo Härder, Wolfgang Lehner, Bernhard Mitschang, Harald Schöning, Holger Schwarz (Hrsg.)  
Datenbanksysteme für Business, Technologie und Web (BTW) 14. Fachtagung des GI-Fachbereichs „Datenbanken und Informationssysteme“ (DBIS)
- P-181 Michael Clasen, Otto Schätzel, Brigitte Theuvsen (Hrsg.)  
Qualität und Effizienz durch informationsgestützte Landwirtschaft, Fokus: Moderne Weinwirtschaft
- P-182 Ronald Maier (Hrsg.)  
6<sup>th</sup> Conference on Professional Knowledge Management  
From Knowledge to Action
- P-183 Ralf Reussner, Matthias Grund, Andreas Oberweis, Walter Tichy (Hrsg.)  
Software Engineering 2011  
Fachtagung des GI-Fachbereichs Softwaretechnik
- P-184 Ralf Reussner, Alexander Pretschner, Stefan Jähnichen (Hrsg.)  
Software Engineering 2011  
Workshopband  
(inkl. Doktorandensymposium)

- P-185 Hagen Höpfner, Günther Specht,  
Thomas Ritz, Christian Bunse (Hrsg.)  
MMS 2011: Mobile und ubiquitäre  
Informationssysteme Proceedings zur  
6. Konferenz Mobile und Ubiquitäre  
Informationssysteme (MMS 2011)
- P-186 Gerald Eichler, Axel Küpper,  
Volkmar Schau, Hacène Fouchal,  
Herwig Unger (Eds.)  
11<sup>th</sup> International Conference on  
Innovative Internet Community Systems  
(I<sup>2</sup>CS)
- P-187 Paul Müller, Bernhard Neumair,  
Gabi Dreö Rodosek (Hrsg.)  
4. DFN-Forum Kommunikations-  
technologien, Beiträge der Fachtagung  
20. Juni bis 21. Juni 2011 Bonn
- P-188 Holger Rohland, Andrea Kienle,  
Steffen Friedrich (Hrsg.)  
DeLFI 2011 – Die 9. e-Learning  
Fachtagung Informatik  
der Gesellschaft für Informatik e.V.  
5.–8. September 2011, Dresden
- P-189 Thomas, Marco (Hrsg.)  
Informatik in Bildung und Beruf  
INFOS 2011  
14. GI-Fachtagung Informatik und Schule
- P-190 Markus Nüttgens, Oliver Thomas,  
Barbara Weber (Eds.)  
Enterprise Modelling and Information  
Systems Architectures (EMISA 2011)
- P-191 Arslan Brömme, Christoph Busch (Eds.)  
BIOSIG 2011  
International Conference of the  
Biometrics Special Interest Group
- P-192 Hans-Ulrich Heiß, Peter Pepper, Holger  
Schlingloff, Jörg Schneider (Hrsg.)  
INFORMATIK 2011  
Informatik schafft Communities
- P-193 Wolfgang Lehner, Gunther Piller (Hrsg.)  
IMDM 2011  
Proceedings zur Tagung  
Innovative Unternehmensanwendungen  
mit In-Memory Data Management

The titles can be purchased at:

**Köllen Druck + Verlag GmbH**

Ernst-Robert-Curtius-Str. 14 · D-53117 Bonn

Fax: +49 (0)228/9898222

E-Mail: [druckverlag@koellen.de](mailto:druckverlag@koellen.de)