

Henning Baars (Hrsg.)

**Revolution oder Evolution –
Neue Trends in der Business Intelligence**

Vierter Workshop „Business Intelligence“

der GI-Fachgruppe Business Intelligence
in Zusammenarbeit mit der Fachgruppe
Wirtschaftsinformatik der Fachhochschule Mainz

am 27. und 28. September 2012 in Mainz

Tagungsband

Vorwort zum 4. Workshop „Business Intelligence“

Der vierte Workshop „Business Intelligence“ der gleichnamigen GI-Fachgruppe hatte das Ziel, innovative Forschungsansätze und Forschungsergebnisse aus dem Bereich der Business Intelligence (BI) zu präsentieren, zu diskutieren und in Bezug zu setzen. Vorge stellt wurden neben originären Forschungsergebnissen bewusst auch Zwischenergebnisse aus Forschungsprojekten (*Research in Progress*), konkrete Forschungsideen sowie neue methodische Herangehensweisen.

Dieser Tagungsband enthält die vier angenommenen und auf dem Workshop vorgestellten Beiträge. Diese zeigen exemplarisch die Breite der deutschsprachigen BI-Forschung. Inhaltlich reicht das Spektrum von domänenspezifischen Arbeiten etwa zu BI in der Logistik, über architektonisch-infrastrukturelle Fragestellungen bis hin zu Entscheidungsmodellen für die BI-Steuerung:

Der Beitrag „Erstellung eines Konzepts für ETL-Prozesse zur Befüllung von produktorientierten Datawarehouses“ stammt von Clemens Hausmann (Universität Stuttgart) und liefert ein Beispiel für die Ausweitung der BI auf neue Anwendungsbereiche sowie die damit verbundenen Herausforderungen. Konkret wird aufgezeigt, wie konstruktionsorientierte Produktdaten mit betriebswirtschaftlichen Transaktionsdaten zusammengeführt werden können, um so eine integrierte Entscheidungsunterstützung für Aufgaben etwa in der Konstruktion oder der Qualitätssicherung zu ermöglichen.

Stärker auf die infrastrukturelle Basis der BI sowie dessen Betrieb ausgerichtet ist die Thematik „Business Intelligence in the Cloud“. In „Assisted migration of enterprise applications to the Cloud“ diskutiert Adrián Juan-Verdejo (CAS A.G.), wie Enterprise-Systeme (insbes. aus dem BI-Bereich) partiell über eine Cloud und partiell inhouse bereitgestellt werden können. Er schlägt hierfür die Bereitstellung eines Entscheidungsunterstützungssystems vor, das relevante Optionen zur Verteilung von Systemkomponenten selektiert und bewertet.

Die Einsatzmöglichkeiten der BI im Umfeld der Kontraktlogistik sowie die damit verbundenen Potentiale und Lösungsansätze diskutieren Axel Klarmann, Martin Roth und Bogdan Franczyk (Universität Leipzig) in ihrem Beitrag „Entscheidungsunterstützung in logistischen Netzwerken“. Hierbei heben sie insbesondere die Bedeutung des Complex Event Processing und von Mobile BI hervor.

Sarah Otyepka, Benjamin Mosig, Marco C. Meier (Universität Augsburg) präsentieren in „Towards an Economic Foundation for the Decision between Agile and Plan-driven Project Management in a Business Intelligence Context“ ein von ihnen entwickeltes, ökonomisch fundiertes Entscheidungsmodell für die Auswahl eines Vorgehensmodells für die BI-Entwicklung. Das Modell basiert auf dem Net-Present-Value-Ansatz und berücksichtigt mit der Wahrscheinlichkeit unzureichend berücksichtigter Anforderungsänderungen und Integrationsmängeln zwei konträre Risikofaktoren.

Inhalt

Clemens Haußmann: Erstellung eines Konzepts für ETL-Prozesse zur Befüllung von produktorientierten Datawarehouses	1
Adrián Juan-Verdejo: Assisted migration of enterprise applications to the Cloud – A hybrid Cloud approach	14
Axel Klarmann, Martin Roth, Bogdan Franczyk: Entscheidungsunterstützung in logistischen Netzwerken – Ansätze für eine Umsetzung	28
Sarah Otyepka, Benjamin Mosig und Marco C. Meier: Towards an Economic Foundation for the Decision between Agile and Plan-driven Project Management in a Business Intelligence Context	37

Erstellung eines Konzepts für ETL-Prozesse zur Befüllung von produktorientierten Datawarehouses

Clemens Haußmann

*Lehrstuhl für Wirtschaftsinformatik I
Universität Stuttgart*

Abstract

Um im Wettbewerb bestehen zu können, ist es für industrielle Unternehmen von zentraler Bedeutung, die Auswirkungen produktbezogener Entscheidungen möglichst zum Zeitpunkt der Entscheidung antizipieren zu können. Betriebswirtschaftliche Analysen müssen dazu neben Daten aus transaktionsorientierten und betriebswirtschaftlichen Systemen auch Produkteigenschaften beschreibende Daten des technischen Bereichs wie der Konstruktion berücksichtigen. Gegenwärtig existiert hier jedoch keine ausreichende Integration. Dieser Beitrag thematisiert die Ausgangssituation und zeigt auf, wie Daten aus den verschiedenen Systemen der Bereiche in einem produktorientierten Datawarehouse zusammengeführt und für Auswertungen entlang des gesamten Produktlebenszyklus verfügbar gemacht werden können, um dadurch die Notwendigkeit eines Konzepts für ETL-Prozesse zur Befüllung von produktorientierten Datawarehouses als Ziel des dargestellten Forschungsvorhabens bzw. den Forschungsbedarf abzuleiten.

1 Einleitung

In industriellen Unternehmen werden bereits in frühen Phasen des Produktlebenszyklus bzw. Produktentstehungsprozesses (Vgl. Abb. 1), insb. in der Konstruktion, zentrale Entscheidungen bzgl. des Bestehens des Produkts bzw. des Unternehmens am Markt getroffen. Die Auswirkungen dieser Entscheidungen sind jedoch gegenwärtig nur schwer antizipierbar, oftmals nur durch Erfahrungswissen abschätzbar. (Kemper, et al., 2011)

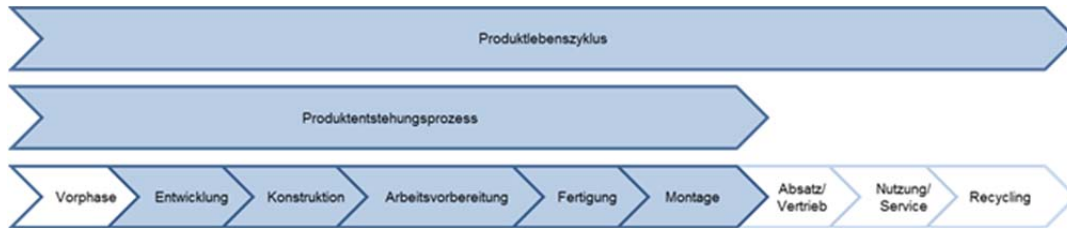


Abbildung 1: Der Produktentstehungsprozess im Produktlebenszyklus (Eigene Darstellung, basierend auf Eigner & Stelzer, 2009; Kemper, et al., 2011; Vajna, 2009; Westkämper, 2006; Wiendahl, 2010)

Dies liegt darin begründet, dass in industriellen Unternehmen eine Trennung zwischen den eher dem betriebswirtschaftlichen Bereich zuzuordnenden transaktionsorientierten bzw. betriebswirtschaftlichen Systemen, wie Enterprise Resource Planning (ERP)-Systeme, und den produktorientierten Systemen, wie Computer-Aided Design (CAD)-Systeme, die in den technischen Bereichen wie der Konstruktion eingesetzt werden, vorherrscht. Dabei existieren Schnittstellenproblematiken und Medienbrüche sowohl innerhalb der einzelnen Bereiche zwischen den dort eingesetzten Systemen, bspw. zwischen CAD-System und Product Data Management (PDM)-System bzw. Product Life Cycle Management (PLM)-System im technischen Bereich, als auch bereichsübergreifend, bspw. bei der Übertragung der Produktstruktur vom PDM-System ins ERP-System (Lasi & Kemper, 2011). Dies wurde durch eigene Voruntersuchungen bestätigt.

Für bereichsübergreifende Analysen entlang des Produktlebenszyklus ist somit eine Integration der relevanten Daten der in den jeweiligen Bereichen eingesetzten Systeme notwendig. Ein Lösungsansatz hierfür ist die Zusammenführung dieser Daten in einem sogenannten produktorientierten Datawarehouse (pDWH). (Lasi & Kemper, 2011; Kemper, et al., 2011)

2 Forschungsdesign, Forschungsfrage und Themeneinordnung

Der vorliegende Beitrag ist Teil eines umfassenden Forschungsprojekts im Kontext der Digitalen Fabrik mit dem Ziel, Analysen auf Basis technischer und betriebswirtschaftlicher bzw. transaktionsorientierter Daten über den gesamten Produktlebenszyklus zu ermöglichen (Vgl. Abb. 2). Zunächst wurde mittels Literaturrecherche und empirischen Voruntersuchungen in Form von leitfadenbasierten Experteninterviews in mittelständischen Unternehmen die Notwendigkeit für bereichsübergreifende Analysen über den gesamten Produktlebenszyklus hinweg festgestellt. Im Anschluss wurden relevante Use Cases erarbeitet mit dem Ziel der Bestimmung der jeweiligen Informationsbedarfe. Mögliche Use Cases sind bspw. der Qualitätsmanager (Lasi 2012a) und der Wissensingenieur

(Lasi 2012b). Ziel des Qualitätsmanagers ist es, die Qualität eines Produktes sicherzustellen und dabei insb. Ursachen für mögliche bzw. akute Qualitätsschwankungen zu identifizieren (Lasi 2012a). Die primäre Aufgabe des Wissensingenieurs besteht in der Systematisierung konstruktiven Wissens und der Aufstellung von Konstruktionsregeln (Lasi 2012b).

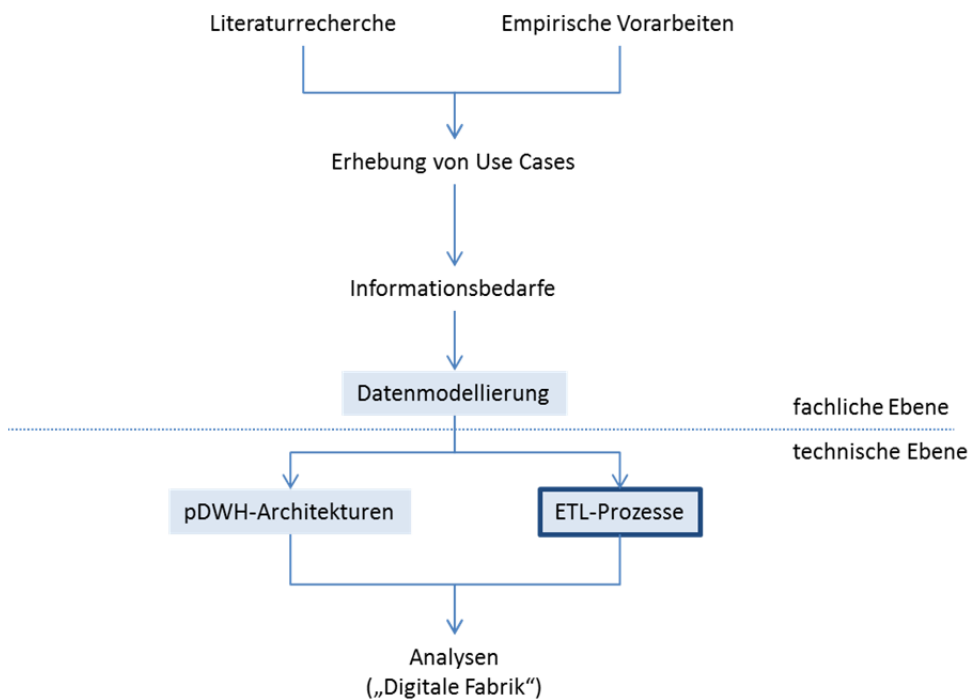


Abbildung 2: Forschungsdesign

Mit dem Ziel der Deckung der identifizierten Informationsbedarfe werden derzeit mögliche Konzepte der Datenmodellierung evaluiert. Im Anschluss an diese fachliche Konzeption werden auf technischer Ebene einerseits mögliche Konzepte für – aufgrund der Voruntersuchungen anzunehmende mehrstufige – pDWH-Architekturen erarbeitet. Andererseits stehen die für die Befüllung des pDWHs notwendigen ETL-Prozesse im Fokus der Forschung. Hier setzt der vorliegende Beitrag an. Ziel dieses Beitrags ist es, anhand der Voruntersuchungen zu klären, inwiefern Forschungsbedarf im Bereich der ETL-Prozesse für pDWHs besteht und diesen grob abzugrenzen. Im folgenden Kapitel wird nun zunächst auf die vorliegende Problemstellung eingegangen und diese anhand eines Beispiels erläutert. Im Anschluss werden ein möglicher Lösungsansatz und die daraus resultierenden Folgen für ETL-Prozesse thematisiert.

3 Problemstellung

3.1 Getrennte Welten – transaktionsorientierte bzw. betriebswirtschaftliche und produktorientierte Systeme

Die im Rahmen der Entscheidungsunterstützung durchgeführten betriebswirtschaftlichen Analysen nutzen lediglich Daten aus Systemen des betriebswirtschaftlichen Bereichs. Daten, die bspw. technische Produkteigenschaften beschreiben, wie sie in einem 3D-Modell (Digital Mock Up, DMU) bzw. 2D-Modell (Zeichnung) hinterlegt sind, werden nicht berücksichtigt. Die (vornehmlich technische) Produktsicht wird somit nicht mit einbezogen. Daten aus CAD-, PDM- und PLM-Systemen finden keine Berücksichtigung. Die Granularität der Daten ist nicht einheitlich, denn die Systeme der technischen Bereiche enthalten wesentlich ausführlichere Informationen bspw. bzgl. der Geometrie, wohingegen im ERP-System lediglich die Produktstruktur bis auf Einzelteilebene vorhanden ist, wie sie bspw. in Stücklisten zu finden ist. So ist es kaum möglich, die Auswirkungen von technischen Veränderungen an Produktbestandteilen (Baugruppen bzw. Einzelteile) in der Konstruktion auf betriebswirtschaftliche Kennzahlen zu ermitteln. Daher können bspw. im Bereich der Angebotskalkulation nur schwer verlässliche Aussagen bzgl. der Auswirkungen von kundenspezifischen Änderungen an Produktbestandteilen auf die Kostenstruktur gemacht werden. (Lasi & Kemper, 2011)

Die erhobenen Use Cases zeigen, dass sowohl technische als auch betriebswirtschaftliche bzw. transaktionsorientierte Daten aus verschiedenen Phasen des Produktlebenszyklus für Analysen im Vorfeld von Entscheidungen benötigt werden (Lasi 2012a; Lasi 2012b):

- Der Qualitätsmanager muss bspw. für Ursachenanalysen im Fall von Reklamationen konstruktive Informationen sowie Daten zu den verwendeten Fertigungsverfahren mit Informationen bzgl. des Anwendungsfalles beim Kunden kombinieren (Lasi 2012a).
- Der Wissensingenieur benötigt bspw. Kosteninformationen (insb. historische), um die Auswirkungen verschiedener Materialalternativen für Komponenten bewerten zu können (Lasi 2012b).

Die durchgeführten Voruntersuchungen in einzelnen Unternehmen ergaben, dass Bestrebungen vorhanden sind, die in den jeweiligen Phasen des Produktlebenszyklus zum Einsatz kommenden operativen Systeme durchgängig miteinander zu verknüpfen und die Produktstruktur (wie in Stücklisten tabellarisch dargestellt) nach der Erstellung in CAD-Systemen über bspw. PDM-/PLM-Systeme in ERP-Systeme zu überführen. In den einzelnen Phasen existieren jedoch unterschiedliche Anforderungen an die Produktstruktur. Somit kann die Gliederung eines Endprodukts in Baugruppen und Einzelteile und deren

Stufenzuordnung über den Produktentstehungsprozess hinweg variieren. Bspw. kann die aus konstruktiver Sicht sinnvolle Gliederung in Funktionsgruppen einer Gliederung nach optimalen Fertigungsabläufen in der Produktion weichen. Gliederungsänderungen werden auch nicht zwangsläufig in die in vorangegangenen Phasen eingesetzten Systeme rückübertragen, sodass für ein und dasselbe Endprodukt unterschiedliche Gliederungen parallel existieren können.

Im technischen Bereich der Konstruktion findet verstärkt die Feature-Technologie Anwendung, bspw. in CAD-Systemen in der Konstruktion (VDI, 2003). Features beschreiben „[...]Bereiche von besonderem (technischem) Interesse[...]“ (VDI, 2003, S.10). Ein Feature stellt die abhängig von einer gewissen Sicht relevanten Eigenschaften und deren Beziehungen eines Objektes dar. Es ist somit kontextabhängig und kann sowohl ausschließlich geometrische (Form, Abmessungen etc.) oder semantische Informationen (bspw. Qualitätsinformationen), aber auch beides beinhalten (VDI, 2003). Somit können bspw. dem Feature „Kernlochbohrung mit Durchmesser 3,5mm“ Qualitätsinformationen wie zulässige Toleranzen hinzugefügt werden. Die Summe der geometrischen Features bildet bei feature-basierten Systemen gemeinsam mit dem Grundkörper das dreidimensionale digitale Produktmodell (Lasi & Kemper, 2011).

In ERP-Systemen ist das Konzept der Sachnummer bei der Identifizierung von Komponenten bzw. Endprodukten von zentraler Bedeutung. Sie identifiziert dabei lediglich eine Klasse von Objekten, bspw. einen bestimmten Schraubentyp. Sie ist also keine Seriennummer, die genau ein bestimmtes Objekt einer Klasse identifiziert. Jeder Knoten in der Produktstruktur wird anhand einer Sachnummer eindeutig identifiziert und nach dem Baukastenprinzip im System hinterlegt. (Grupp, 1987)

Somit unterscheidet sich die Art der Hinterlegung von Produktstrukturen in den Systemen der jeweiligen Bereiche deutlich.

3.2 Verdeutlichung der Problemstellung anhand eines Beispiels

Im Folgenden soll nun der im vorangegangenen Abschnitt angeführte Sachverhalt anhand eines Beispiels erläutert werden. Abbildung 3 stellt die Produktstruktur für ein beispielhaftes Produkt mit 4 Hierarchiestufen exemplarisch dar.

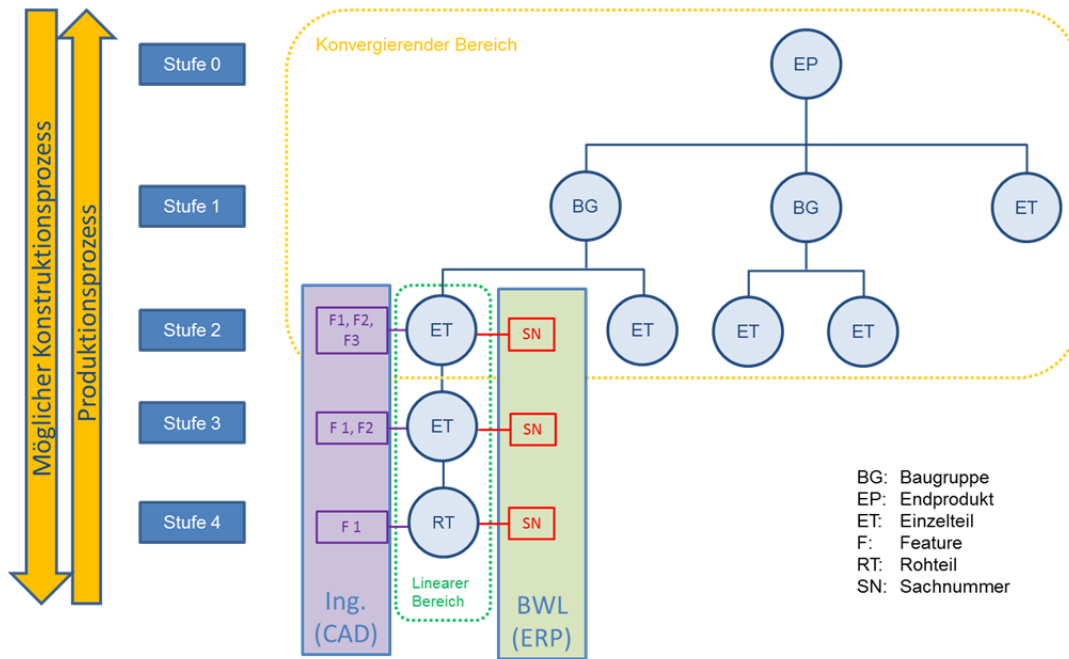


Abbildung 3: Eine beispielhafte Produktstruktur
(Eigene Darstellung, basierend auf Binner, 2003)

Rohteile, Einzelteile, Baugruppen und das Endprodukt werden in dieser Baumstruktur als Knoten dargestellt. Wird aus einem Rohteil ein Einzelteil gefertigt bzw. ein Einzelteil in ein anderes Einzelteil umgearbeitet, wird von einer linearen Struktur gesprochen. Findet eine Kombination von Einzelteilen oder Baugruppen zu anderen Baugruppen statt, wie in der Montage, wird dies als konvergierend bezeichnet. (Günther & Tempelmeier, 2009)

Beispiel: Aus einem Rohteil auf Stufe 4 wird ein Winkel gefräst. Diesen stellt das Einzelteil auf Stufe 3 dar. Während des Bearbeitungsprozesses werden somit Features hinzugefügt, hier die Form des Winkels (Geometrie). Im nächsten Fertigungsschritt wird der Winkel mit einem Loch und dazugehörigem Gewinde versehen, technisch gesehen ein weiteres Hinzufügen von Features. Das resultierende Einzelteil befindet sich auf Stufe 2. Im weiteren Produktionsprozess wird dann im Rahmen der Montage ein extern beschaffter Bolzen angeschweißt, wieder ein Hinzufügen von weiteren Features. Die resultierende Baugruppe befindet sich auf Stufe 1. An dieser Baugruppe wird dann im weiteren Montageablauf mittels einer Schraube (das dargestellte Einzelteil auf Stufe 1) die andere auf Stufe 1 dargestellte Baugruppe angeschraubt. Auf eine Darstellung des Montagevorgangs für diese zweite Baugruppe wird an dieser Stelle verzichtet. Am Ende steht das Endprodukt, das in diesem Fall bspw. eine Maschinenkomponente sein kann, die an ein Maschinenbau-

unternehmen weiterverkauft wird. Eine Betrachtung der im Produktionsablauf eingesetzten Systeme zeigt deutlich die Schnittstellenproblematik und die unterschiedliche Granularität der Daten: Im ERP-System sind lediglich die einzelnen Knoten aus Abbildung 3 (Rohmaterial, Einzelteile, Baugruppen, Endprodukt) als Objekt unter der jeweiligen sogenannten Sachnummer hinterlegt. Abbildung 3 verdeutlicht die hinterlegten Informationen im CAD- bzw. ERP-System exemplarisch für den linearen Bereich. Auf eine Darstellung im konvergierenden Bereich wurde aus Komplexitätsgründen verzichtet. Hier ist der Sachverhalt jedoch analog zum linearen Bereich.

Zu einer Sachnummer sind bspw. Kosten- und Termininformationen bzw. Arbeitspläne hinterlegt, unter Umständen auch technische Zeichnungen oder Abbildungen von dreidimensionalen Modellen. Die technischen Informationen, die in diesen Dokumenten enthalten sind, sind jedoch im Gegensatz zu Kosten- oder Termininformationen somit nicht im ERP-System auswertbar. Im obigen Beispiel sind dem Einzelteil auf Stufe 3 die benötigte Fertigungszeit und die zu verwendende Maschine bzw. das benötigte Werkzeug zugeordnet. Es besteht eine Verknüpfung, die besagt, dass das Einzelteil aus dem Rohteil aus Stufe 4 gefertigt wird. Des Weiteren sind je Sachnummer auch bspw. das Datum des Anlegens im ERP-System, der Name des anlegenden Sachbearbeiters und der Versionsstand hinterlegt.

Im CAD-System der Konstruktion hingegen (im Folgenden wird von einem 3D-fähigen CAD-System ausgegangen) existiert am Ende des Konstruktionsprozesses für das gesamte Endprodukt ein digitales dreidimensionales Modell. Der Konstruktionsprozess kann im Beispiel also so aussehen, dass zunächst mit einem viereckigen Stück Material begonnen wird (Rohmaterial), aus dem dann im CAD-System die Form des Winkelstücks erzeugt wird (Einzelteil auf Stufe 3). Es wird hier also ein geometrisches Feature hinzugefügt. Analog verhält es sich mit der Bohrung und dem Gewinde. Auch diese Features werden im Konstruktionsprozess hinzugefügt. Im Beispiel folgt der Produktionsprozess also dem Konstruktionsprozess.

Das Beispiel zeigt, dass eine Baugruppe nicht die Summe der in sie eingehenden Einzelteile und Rohteile bzw. Baugruppen ist. Dies wird insbesondere im linearen Bereich deutlich. Vielmehr ist eine Baugruppe die Summe der in sie eingehenden Features, also die Summe sämtlicher Features, die in den hierfür verwendeten Einzelteilen bzw. Baugruppen enthalten sind. Die Features sind in den DMUs der CAD-Systeme enthalten. Betriebswirtschaftliche Auswertungen bedienen sich aber der Daten aus den ERP-Systemen. Zusammenfassend lässt sich also sagen, dass die im Rahmen der Entscheidungsunterstüt-

zung benötigten Informationen zwar im Unternehmen durch die Daten der operativen Systeme vorhanden sind, jedoch für Auswertungen verfügbar gemacht werden müssen.

4 Lösungsansatz

Die oben beschriebene Schnittstellenproblematik zeigt, dass Analysen direkt mittels der in den operativen Systemen hinterlegten proprietären Daten bei der Umsetzung zu erheblichen Problemen, insb. aufgrund unterschiedlicher Granularitäten, führen würden. Die benötigten Daten sind somit bereits in operativen Systemen vorhanden, müssen jedoch noch für bereichsübergreifende Zwecke nutzbar gemacht werden.

Um nun hier durchgehende betriebswirtschaftliche Analysen über die gesamte Entstehung eines Produktes zu ermöglichen, ist somit zwingend eine durchgehende integrierte dispositive Datenhaltung entlang des gesamten Produktentstehungsprozesses notwendig. Dies kann durch die Zusammenführung der Daten aus den eingesetzten operativen Systemen in einem pDWH geleistet werden (Lasi & Kemper, 2011; Kemper, et al., 2011). Dort werden die gesamten Erzeugnisstrukturen mit den dazugehörigen Hierarchien hinterlegt. Dieses pDWH ermöglicht somit eine integrierte, einheitliche Datenbasis entlang des gesamten Produktentstehungsprozesses. Dabei wird die Produktstruktur um technische Informationen ergänzt, um so die Informationslücke zwischen ingenieurwissenschaftlichen und betriebswirtschaftlichen Bereichen zu schließen und Transparenz zu ermöglichen. Insbesondere die technischen Informationen sind zwar in DMUs bzw. Zeichnungen in CAD- oder PDM-/PLM-Systemen enthalten, im Rahmen von Analysen anhand der Produktstruktur sind sie jedoch nicht direkt auswertbar. Eine Möglichkeit, technische Informationen für betriebswirtschaftliche Analysezwecke im pDWH verfügbar zu machen, ist mittels der sog. Feature-Technologie (Lasi & Kemper, 2011). Wie in Kapitel 3 erläutert, entspricht ein Einzelteil der Summe aller Features, analog gilt dies für Baugruppen und Endprodukte. Mittels Features werden also die technischen Eigenschaften voll beschrieben und es können zudem weitere Informationen in Form von semantischen Features hinterlegt werden.

Die Feature-Technologie ermöglicht es zudem, in den einzelnen Phasen des Produktlebenszyklus neue Informationen in Form von Features zu ergänzen und gewährleistet somit eine durchgängige Flexibilität (VDI, 2003). Diese Skalierbarkeit ist Grundvoraussetzung für eine bereichsübergreifende Nutzung.

Jedoch variieren die Features bezüglich der beinhalteten Informationen entlang des Produktentstehungsprozesses. Die Feature-Arten (z.B. Konstruktionsfeature, Fertigungsfeature) in den Feature-Bibliotheken der eingesetzten Systeme können sich unterscheiden. Dies erfordert bei der Übertragung des digitalen Produktmodells von einem zum anderen

System auch eine tendenziell aufwendige und problembehaftete Transformation der Features. (VDI, 2003)

Operative Quellsysteme sind hierbei transaktionsorientierte Systeme des betriebswirtschaftlichen Bereiches und produktorientierte Systeme des technischen Bereichs. Das Grundgerüst der Daten im pDWH wird dabei durch die klassische Produktstruktur gebildet, wie sie in Stücklisten zu finden ist. Die in einer Stückliste hinterlegte Hierarchie wird im pDWH abgebildet. Nur so kann sichergestellt werden, dass das Datenmodell während des gesamten Produktlebenszyklus skalierbar ist. Eine reine Beschränkung auf Features würde zwar im Bereich der Konstruktion unter Umständen ausreichen, jedoch bauen transaktionsorientierte und betriebswirtschaftliche Systeme wie ERP-Systeme auf dem Konzept der Sachnummern auf (Vgl. Abb. 3). Diese Sachnummern sind somit auch zentraler Gegenstand betriebswirtschaftlicher Analysen. Eine reine Beschränkung auf Features würde sogar zu einem Informationsverlust führen, denn Informationen wie Versionsstand können nur dem jeweiligen Knoten und keinen Features zugeordnet werden. Die für Auswertungen essentielle Historisierung der Daten wäre somit nicht möglich. Daher ist zwingend die Produktstruktur als Grundgerüst für das Datenmodell notwendig.

Mittels Feature-Technologie kann ein Einstieg an jedem Knoten der Produktstruktur stattfinden, wobei jeweils, wie oben gezeigt, sämtliche Informationen über die eingehenden Einzelteile und Rohteile bzw. Baugruppen voll verfügbar sind.

Die Feature-Technologie ermöglicht es somit, sowohl Informationen aus dem betriebswirtschaftlichen Bereich als auch aus dem technischen Bereich in einem pDWH zusammenzuführen und für Auswertungen entlang des gesamten Produktlebenszyklus bereitzustellen. Die Produktstruktur wird hierfür mit Features angereichert bzw. die Produktstruktur wird über die Einzelteilebene hinaus um Features erweitert. Hierfür sind verschiedene Modellierungskonzepte möglich. Vorarbeiten führen zu der Annahme, dass eine multidimensionale Modellierung hier geeignet sein könnte. Zu jeder Sachnummer werden die Informationen in Form von Features als Fakten einer multidimensionalen Modellierung hinterlegt. Features unterschiedlicher Sichtweise (bspw. Qualität oder Revisionsstand) stellen unterschiedliche Dimensionen dar. Eine zentrale Herausforderung stellt dabei die Harmonisierung unterschiedlicher Feature-Arten dar. Die Evaluation der Modellierungsmethoden ist Gegenstand eines separaten Forschungsvorhabens im übergeordneten Forschungsprojekt (Vgl. Kapitel 2).

Durch diesen Ansatz wird es möglich, Analysen unter Berücksichtigung sowohl technischer als auch betriebswirtschaftlicher Daten durchzuführen. Somit können die Auswirkungen technischer Änderungen auf betriebswirtschaftliche Größen antizipiert werden.

Zur Überführung von Produktstruktur und Produkteigenschaft beschreibenden Daten aus den operativen Quellsystemen ins pDWH sind umfangreiche ETL-Prozesse notwendig. ETL-Prozesse umfassen die aufeinanderfolgenden Teilbereiche der Filterung, Harmonisierung, Aggregation und Anreicherung der Daten (Chamoni, et al., 2005; Gluchowski, et al., 2008; Kemper, et al., 2010). Im betriebswirtschaftlichen Bereich ist dies etablierte Praxis (Chamoni, et al., 2005; Gluchowski, et al., 2008; Kemper, et al., 2010), für produktspezifische Daten aus dem ingenieurwissenschaftlichen Umfeld der Entwicklung bzw. Produktion mangelt es jedoch an einer konzeptionellen Ausarbeitung für ETL-Prozesse. Ein Grobkonzept für diese ETL-Prozesse wurde in einer ersten Phase erarbeitet. Hierbei hat sich gezeigt, dass bereits existierende Konzepte aus dem betriebswirtschaftlichen Bereich nicht direkt übertragbar sind, da sich die Ausgangssituation, wie oben erläutert, grundlegend unterscheidet: Quellsysteme für die zu extrahierenden Daten sind in den technischen Bereichen geometrieorientierte Systeme wie CAD- bzw. PDM/PLM-Systeme. Dort sind die benötigten Daten in DMUs in Form von Features eingebettet. Die Daten müssen somit aus diesen digitalen Modellen zunächst extrahiert werden. Dies unterscheidet sich deutlich von den im betriebswirtschaftlichen Kontext etablierten ETL-Prozessen. Eine weitere wesentliche Herausforderung stellt die Harmonisierung und Zusammenführung der aus den DMUs extrahierten Daten mit den betriebswirtschaftlichen und transaktionsorientierten Daten aus der ERP-Welt dar. Das Konzept sieht daher vor, die aus den DMUs extrahierten Features anhand der Feature-Arten zu aggregieren, bis sie konsistent zu den aus den ERP-Systemen übernommenen Produktstrukturen sind. Damit können betriebswirtschaftliche und transaktionsorientierte Daten (bspw. Plankosten) ebenfalls im pDWH den jeweiligen Knoten zugeordnet werden. Ziel eines weiteren Forschungsvorhabens muss es daher sein, ein entsprechendes Konzept zu entwickeln, um zunächst die relevanten Daten in den operativen Quellsystemen zu identifizieren, anschließend zu extrahieren und aufbereitet im pDWH zur Verfügung zu stellen. Das erarbeitete Konzept soll dann der Forschungskonzeption der Gestaltungsorientierten Wirtschaftsinformatik (Österle, et al., 2010) folgend prototypisch implementiert und evaluiert werden.

5 Zusammenfassung

Dieser Beitrag erläutert die Notwendigkeit eines Forschungsvorhabens, dessen Ziel es ist, ein Konzept für die Extraktion und Aufbereitung von Produktdaten (ETL-Prozesse) sowohl aus operativen transaktionsorientierten und betriebswirtschaftlichen als auch produktorientierten Systemen zur Bereitstellung in einem produktorientierten Data Warehouse zu entwickeln, prototypisch umzusetzen und zu evaluieren. Diese ETL-Prozesse beinhalten

- technische (Zugang zu den relevanten Quellsystemen der verschiedenen Bereiche),
- semantische (Filterung und Harmonisierung der relevanten Daten) und
- syntaktische (Integration auf Basis der Feature-Technologie)

Herausforderungen.

Das übergeordnete Ziel ist es, durch die fokussierten ETL-Prozesse Daten aus produktorientierten Systemen, wie bspw. Produkteigenschaften, in einem pDWH für betriebswirtschaftliche Analysen bereitzustellen und so bereits bestehende Analysemöglichkeiten um Aussagefähigkeiten bzgl. direkter Ursache-Wirkungsbeziehungen, bspw. von konstruktiven Änderungen einzelner Teile, zu erweitern.

Aufgrund der unterschiedlichen Ausgangssituation in den beiden Bereichen, der Form in der die benötigten Daten in den jeweiligen Systemen vorliegen, sind jeweils abgestimmte ETL-Prozesse notwendig:

- Die Nutzung der Feature-Technologie im technischen Bereich lässt den Schluss zu, dass ETL-Prozesse des betriebswirtschaftlichen Bereichs hier nicht übertragbar sind, um die in Form von Features vorliegenden Informationen in einem pDWH zur Verfügung zu stellen.
- Zudem erfordert die feature-basierte Konzeption des pDWH hierauf abgestimmte ETL-Prozesse im betriebswirtschaftlichen Kontext, um betriebswirtschaftliche bzw. transaktionsorientierte Daten ins pDWH zu überführen und die erforderliche Integration mit Daten aus technischen Systemen zu leisten.

Daher besteht im Bereich der ETL-Prozesse sowohl für transaktionsorientierte bzw. betriebswirtschaftliche als auch produktorientierte Daten weiterer Forschungsbedarf.

6 Literaturverzeichnis

Binner, H.F. (2003). Prozessorientierte Arbeitsvorbereitung (2. ed.). München Wien: Hanser.

Chamoni, P., Gluchowski, P. & Hahne, M. (2005). Business Information Warehouse: Perspektiven betrieblicher Informationsversorgung und Entscheidungsunterstützung auf der Basis von SAP-Systemen. Berlin Heidelberg: Springer.

- Eigner, M. & Stelzer, R. (2009). *Product Lifecycle Management: Ein Leitfaden für Product Development und Life Cycle Management* (2. ed.). Berlin Heidelberg: Springer.
- Gluchowski, P., Gabriel, R. & Dittmar, C. (2008). *Management Support Systeme und Business Intelligence: Computergestützte Informationssysteme für Fach- und Führungskräfte* (2. ed.). Berlin Heidelberg: Springer.
- Grupp, B. (1987). *Optimale Verschlüsselung bei Online-Datenverarbeitung: Aufbau moderner Nummernsysteme für Sachnummern jeder Art, Personennummern und Auftragsnummern*. Köln: TÜV Rheinland.
- Günther, H.-O. & Tempelmeier, H. (2009). *Produktion und Logistik* (8. ed.). Berlin Heidelberg: Springer.
- Kemper, H.-G., Baars, H. & Lasi, H. (2011). *Business Intelligence - Innovative Einsatzfelder in der Industrie*, in: Felden, C., Krebs, S. & Stock, S. (Eds.): *Perspektiven der Business Intelligence - Festschrift anlässlich des 60. Geburtstages von Peter Chamoni*. München, Hanser.
- Kemper, H.-G., Baars, H. & Mehana, W. (2010). *Business Intelligence - Grundlagen und praktische Anwendungen: Eine Einführung in die IT-basierte Managementunterstützung* (3. ed.). Wiesbaden: Vieweg + Teubner.
- Lasi, H., & Kemper, H.-G. (2011). *Integrationsansätze zur Verbesserung der Entscheidungsunterstützung im Innovationsmanagement*. HMD - Praxis der Wirtschaftsinformatik (278), 94-103.
- Lasi, H. (2012a). *Industrial Intelligence - a BI-based approach to enhance manufacturing engineering in industrial companies*. *Proceedings of the 8th CIRP Conference on Intelligent Computation in Manufacturing Engineering (CIRP ICME)*. 18 - 20 July 2012. Gulf of Naples, Italy.
- Lasi, H. (2012b). *Decision Support within Knowledge-Based Engineering - a Business Intelligence-Based Concept*. *Proceedings of the 18th Americas Conference on Information Systems (AMCIS)*. 09.08.-11.08.2012. Seattle, USA.
- Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., Loos, P., Mertens, P., Oberweis, A. & Sinz, E.J. (2010). *Memorandum zur gestaltungsorientierten Wirtschaftsinformatik*. *Zeitschrift für betriebswirtschaftliche Forschung*, 62 (6), 664-672.
- Vajna, S., Weber, C., Bley, H. & Zeman, K. (2009). *CAX für Ingenieure: Eine praxisbezogene Einführung* (2. Ed.). Berlin Heidelberg: Springer.

VDI (2003). VDI-Richtlinie 2218: Informationsverarbeitung in der Produktentwicklung - Feature-Technologie. Düsseldorf: Verein Deutscher Ingenieure (VDI).

Westkämper, E. (2006). Einführung in die Organisation der Produktion. Berlin Heidelberg: Springer.

Wiendahl, H.-P. (2010). Betriebsorganisation für Ingenieure (7. ed.). München: Hanser.

Assisted migration of enterprise applications to the Cloud

– A hybrid Cloud approach

Adrián Juan-Verdejo

CAS Software A.G

Abstract

Cloud Computing is a relatively new paradigm with the potential to transform how IT hardware and software are designed and purchased. Computing is no longer purchased as typical products but delivered as a service over the Internet from large data centres. However, despite the potential benefits associated with the migration of enterprise applications from an in-house data centre into a Cloud infrastructure, there are still some issues that hinder the process. In this respect, a migrated application should satisfy specific enterprise policies related to privacy and security, as well as provide an acceptable quality of service. Because of the complexity of enterprise applications today with respect to the large number of components deployed in multi-tier architectures, the complexity of the interactions between them and the components' dependence on stored data, the migration process has to be assisted. Therefore it is presented a framework able to automate the decision-making process related to the migration of different components to a virtualized Cloud system based on a model which factors in benefits and disadvantages associated with the migration subject to the fulfilment of the SLA and the respect of policy constraints. The presented approach envisions an application hosted both partly on-premise and on the Cloud. Resulting migrated applications need to be compliant with enterprise and security policies' constraints while minimizing costs and ensuring performance in terms of wide area network communications and response times. The framework relies on a model to efficiently decide on the component placement. A model which takes into account several factors such as enterprise policies, dynamic performance bottlenecks, data sensibility, performance, cost savings from migration, data flows between application components, and the spread and variability of users. On the one hand the migration could lower the SLA (Service Level-Agreement) while on the other hand the deployment on the Cloud would provide the ability to handle peaks in workload and a higher reliability due to the higher level of replication, the existence of multiple fault domains and the deployment of the application on multiple Clouds. With respect to the

latter, it will be assessed the deployment on different Clouds depending on the constraints imposed by them. The presented framework relies on algorithms and a formal decision model in order to assist developers in migrating existing applications. With the aim of empirically evaluate the overall framework, a prototypical implementation of the framework will be used in order to support the migration to a Cloud-based environment of an existing product configurator. The empirical validation will be used to further improve the input model on which the decision-making process was founded.

1 Introduction

Due to the fast-paced development of both processing and storage technologies, and the great adoption of the Internet, computing resources have become ubiquitously available at a cheaper price while offering powerful solutions. This trend has enabled the delivery of computing as a utility, with its resulting potential to significantly transform the information technology (IT) world (Fox & Griffith, 2009). Over the last few years, not only early adopters but an increasing number of enterprises are being attracted to Cloud-based deployments due to the promises of reducing the costs associated with their IT-related activities while maintaining, if not increasing, the quality of the service provided. Both the potential advantages and the initial success stories of Cloud Computing adoption inspire enterprises to migrate their existing applications to a Cloud-based architecture. However beneficial, the adoption of this new paradigm by enterprises implies a fundamental change in information technology provision that affects how the services they offer are devised, developed, deployed, scaled, tested, improved, updated, and even charged for (Armbrust et al., 2010). Furthermore, in addition to the technical implications of the embracement of Cloud Computing, the socio-technical and organizational effects of the adoption have to be considered (Khajeh-Hosseini, Sommerville, & Sriram, 2010). Enterprise decision makers should take a holistic approach to analyze the different implications related to the adoption of this relatively new technology. This approach considers both the technical and business-related consequences of the adoption.

From a socio-technical viewpoint, there are legal and regulatory constraints — related to enterprise-specific policies, industry-specific laws and regulations, and national privacy requirements — that have to be respected after the application and enterprise data have been migrated. On the other hand, from a more technical standpoint, the Quality of Service (QoS) requirements of the end-users have to unarguably be met in order to conform to the Service-level agreement (SLA). As a result of this service contract, developers usually have to cope with stringent requirements with regard to availability, performance and delay, while at the same time they are required to respect the aforementioned data privacy regulations. Moreover, costs in terms of wide-area communications resulting

from the actual migration have to be contemplated as well. All these issues call for a framework to simplify the task of migration of legacy applications to the Cloud.

2 Motivation

In an effort to provide a solution for these concerns, the architecture is based on a hybrid deployment in which some parts of the application are migrated to the Cloud whereas some others are hosted on-premises within the boundaries of the enterprise to which the service is being delivered. (Calheiros, Ranjan, Beloglazov, De Rose, & Buyya, 2011). A framework is supplied in order to help professionals by providing them with a mechanism to increase application deployment flexibility, as components and data can be migrated to the Cloud while some other parts of the application are kept locally. The decision on what to migrate hinges on a strategy that strives for balance in performance, cost reduction, availability, data privacy, and enterprise policies.

As an example, let us assume that an enterprise aiming for cost reduction plans the migration to the Cloud of an application accessing patients' healthcare data. Due to national privacy requirements, databases which store sensitive data (back-end tier) have to be kept locally. This design decision might affect the performance of those application components which frequently retrieve data from local databases. The system could suffer from increased delays and response times because of the existence of interdependent parts of the application which run at separated locations. Therefore, in this case the wiser solution might be to keep locally both the data and the components which access them very often, in order to enhance performance. On the other hand, less sensitive components would be migrated to the Cloud and can be remotely accessed. As evidenced by this example, the decision of what components to migrate or to keep locally is a challenging one. Hence, the presented framework will base the deployment decision on a component placement model.

The described model for component positioning applies to typical enterprise applications today, which usually present a multi-tiered architecture. These applications are usually composed of lots of components which interact and depend on each other in a non-trivial manner. These facts motivate the delivery of a model to improve the migrating decision by taking into account many factors, namely data privacy, legal and regulatory constraints (e.g., enterprise-specific constraints), dynamic performance bottlenecks, cost reduction due to migration, data flows between application components (transactions could suffer from increased delays), increased reliability due to the presence of multiple fault domains, wide-area communication costs of migrations, and spread and variability of users.

The model and algorithms are evaluated through the application of the introduced framework to realistic migration scenarios. Specifically, the case study uses a product configurator as the enterprise application targeted to be migrated to the Cloud.

At an early stage, the migration process takes into account two locations, local versus Cloud data-centre. However, the described approach can be extended to the migration to multiple Cloud locations running in different Cloud providers' architectures. Optionally, the different characteristics of the Clouds available in the market can be incorporated into the model. The different set of characteristics offered by Cloud providers can be used to decide which Cloud suits the application to be migrated better.

3 Related Work

The related work section will start with a general description of cloud computing together with references to some positioning papers. Next, the reasons for a company to adopt this relatively new paradigm will be described in detail using case studies and best practices. Finally, some frameworks and simulation tools related to the migration of legacy applications to the cloud will be listed.

For some years now Cloud Computing has been a promising area for both scientists and professionals. Cloud Computing emerged as a natural evolution of a combination of virtualization, utility computing and distributed computing (Armbrust et al., 2010). Despite the initial mismatch and overabundance of Cloud Computing definitions, an agreement has been reached as to specify Cloud Computing based on the US National Institute of Standards and Technology definition of Cloud Computing (US NIST, 2009). According to their definition, Cloud Computing is *a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction*. A system offering a Cloud Computing service should present five essential characteristics; namely, on-demand self-service, broad network access, resource pooling, rapid elasticity, and measured service. In literature new Cloud-based service models are presented as everybody strives to offer *everything* as a Service. Nevertheless, only three service models are commonly accepted, namely Software as a Service, Platform as a Service, and Infrastructure as a Service. Nowadays research institutions tend to build upon the NIST definition striving for standardization (Vaquero, Rodero-Merino, Caceres, & Lindner, 2008) (Vouk, 2008). Some of these surveys base their line of argument on companies' reports which aim at providing a global overview of the opportunities for IT Cloud services (IDC,

2010). Those company reports state how Cloud Computing is an appealing topic not only for academia but for enterprises as well.

Enterprises are attracted by the benefits of migrating to a Cloud-based architecture. However, Cloud Computing not only represents a chance to technically improve modern data centres but it also entails an important change in how services are both provisioned and used (Fox & Griffith, 2009). Therefore, professionals have to consider both the benefits and risks of migrating to a Cloud-based deployment. Moreover, more subtle changes have to be considered as well, such as the organizational changes caused once the responsibilities of the IT department are shifted outside of the organization to external companies, e.g. Amazon (Khajeh-Hosseini, Sommerville, et al., 2010). In literature researchers strive to illustrate the real-life implications of the migration of existing applications to the Cloud by performing case studies with different approaches either technically oriented, more business-oriented or something in between.

On the one hand, some take a technical approach on how to tackle the challenges related to the carrying out of the actual migration or the implications for the software architecture of planning the migration of an application. Some authors (Babar & Chauhan, 2011) focus on the analysis of the requirements of the application to be migrated in order to identify the architectural modifications needed to Cloud-enable an existing application. They state that current architecture evaluation methods do not effectively assess the architecture decisions before their implementation as they do not draw attention to Cloud-related quality features such as scalability and accessibility. Likewise, security aspects have to be borne in mind and integrated in the migration process of legacy systems (Rosado, Gómez, Mellado, & Fernández-Medina, 2012). Due to the fact that each service provider has its own identity management system, a user needs to have multiple digital identities. Therefore, they suggest the use of federated identity management in order to uniquely identify users for each of the services accessed (Seung, Lam, Li, & Woo, 2011).

On the other hand, some authors stress the economic and operational implications of including Cloud Computing in an organization, yet ignoring the inherent technical challenges (Marston, Li, Bandyopadhyay, Zhang, & Ghalsasi, 2011). Following this trend, (Hajjat et al., 2010) study how the overall migration cost is the result of a complex combined effect of applications characteristics in terms of workload intensity, storage capacity, growth rate, and the cost of software licenses. They conclude that horizontal partitioning between in-house and Cloud deployments can be very beneficial for certain applications. (Khajeh-Hosseini, Greenwood, & Sommerville, 2010) shows a 37% system infrastructure cost reduction. Moreover, they exemplify the operational implications of the migration as approximately 1 out of 5 support calls for the analyzed system were avoided.

Furthermore, some authors avoid taking sides and have adopted a more holistic approach as they abandon the typical reductionist approach and try to emphasize the study of complex systems. (Mohagheghi & Sæther, 2011) have developed an agile, model-driven, tool-supported methodology for the migration to the service Cloud paradigm. They built upon SMART (Lewis, Morris, Smith, & Simanta, 2008) and included business processes into their methodology. Nevertheless, they did not incorporate legal and regulatory constraints. Following this holistic trend, some researchers supply frameworks which consider multiple criteria in order to assess the migration to the cloud.

The *CloudGenious* framework takes into account a large set of heterogeneous criteria and their interdependencies. However, they apply their framework to single-tiered applications. Therefore, they leave out a lot of current enterprise applications which are more complex than that (Menzel & Ranjan, 2011). A similar approach is taken in *Cloudward Bound* as (Hajjat et al., 2010) present a modelling technique for the migration of enterprise applications to the Cloud. Nevertheless, they forgot to include the socio-technical implications of the migration. As validation, they evaluate their algorithms in the migration of an ERP system to the Cloud. *Cloud Motion (CMotion)* is another framework devised to help researchers to solve the issues related to the migration of heterogeneous composite applications. The very nature of these applications, which consist of heterogeneous components noncompliant with any specific interface, hinders the migration process. *CMotion* effectively addresses this problem whereas it does not incorporate enterprise or government policy support into the decision system (Binz, Leymann, & Schumm, 2011). *CloudMIG* is a framework which automates parts of the migration process, such as the extraction of application requirements, the selection of a Cloud-provider and finally the generation of the target architecture (Frey & Hasselbring, 2010). However, they still have to improve their framework to enable it to evaluate migrated applications. Following the same trend, *CloudFlex* focuses on the technical issues related to the migration of a typical three-tiered enterprise application. *CloudFlex* uses load balancing to solve problems related to dynamic performance bottlenecks which appear after the migration but disregards the socio-technical effects of the migration (Seung et al., 2011). The different frameworks explained are often validated using ad-hoc validation methods. In an attempt at facilitating these tasks, some researchers have presented their own framework for modelling, simulation, and experimentation.

With the aim of assisting researchers in the evaluation of the performance of Cloud provisioning policies, application workload models and resources performance, there have been some attempts to develop simulators of Cloud Computing architectures. *CloudSim* (Calheiros et al., 2011) models VM allocation, network and data centre energy consumption but they focus on a lower level of abstraction, the Infrastructure as a service (IaaS)

level. On the other hand, the presented approach works on the Software as a Service (SaaS) level.

4 Approach

4.1 Overview

As stated in the Motivation section, the main goal of this work is to deliver a framework able to automate the decision-making process related to the migration of different components to a virtualized Cloud system. Given the complexity of the migration strategy selection, the process is based on a model (*see* 4.2)4.2 which factors in benefits and costs associated with the migration subject to the fulfillment of the SLA and the respect of policy constraints. It will help developers to perform the efficient migration of existing enterprise applications by adopting a hybrid migration approach. According to this approach, some of the applications' components are kept locally while others are migrated to a Cloud-based infrastructure. The prototypical implementation of the framework determines a migration strategy by using a decision support tool based on a multi-criteria model for component placement. Those criteria consider not only applications' technical characteristics but they also take into account business-related constraints and requirements. The applicability, validity and effectiveness of both the overall framework and the presented model are evaluated.

In order to validate the described approach the migration of an enterprise will be evaluated. The framework will be employed for the migration of a legated Product Configurator system to a hybrid cloud deployment. The Product Configurator is the real application intended to be migrated to a new Cloud-based deployment and the findings obtained during the validation will provide feedback in order to enhance the component placement model and the decision support part of the framework.

Additionally, this work might consider the specific characteristics of the different Cloud providers in order to incorporate them into the component placement model. The migration of a component to a certain Cloud provider instead of to another can be more beneficial due to the characteristics offered by that specific provider. Moreover, this functionality offers different alternatives in order to avoid the vendor lock-in problem.

4.2 Modelling of components placement

The proposed model can be further explained by the description of the criteria taken into account for the modelling process. Namely, the following:

Legal and regulatory constraints: often disregarded in literature but very relevant to the approach taken by this work. In this classification, enterprise-specific constraints, industry-specific laws and regulations, and national privacy requirements are contemplated.

Dynamic performance bottlenecks (*see Dynamic Performance Bottlenecks*, also called choke points (Seung et al., 2011)).

Cost reduction thanks to the migration because of the migration of both computation and storage to a Cloud environment.

Data flows between application components Increased transactional delays between components which interact very often. The transaction sizes and frequency will be taken into account together with the transaction delays means and variances. Moreover, the transaction origin and destination will be considered (internal users, external users, different components).

Wide-area communications cost of the actual migration arising from the migration of both data and application state to the Cloud nodes.

Spread and variability of users

1 Within or outside the enterprise premises

Data privacy and sensibility. Due to the impossibility of migrating sensitive data (e.g. credit card numbers), those components highly dependent on these data could be forced to be kept locally in the environment of the frequently accessed data.

The model performs application discovery tasks in order to detect dependencies and interactions between components. Moreover, the model needs input related to the predicted response times of these interactions and how much traffic will happen between the application components.

1.1 Example of the migration of a typical enterprise system

With the aim of clarifying the approach taken, an example of the migration of a typical enterprise system will be presented in two figures. This example is not founded on any empirical result but it intends to illustrate the presented approach by showing a specific scenario. Figure 1, depicts a typical example of an enterprise system running off-premises, everything runs on the client side and the provider must deploy the application on the client's infrastructure. As in typical enterprise systems today, the system consists of many interdependent components, namely the *migration target component 1, 2, and 3*,

the *s1.1* and *s1.2* components, and the *client-side component*. Moreover, the system interacts with two database management systems, namely *DBMS1* and *DBMS2*. The *migration target components 1, 2, and 3* are the components intended for migration and they depend on other components and databases. The *client-side component* presents the data to the user. Finally, the *s1.1* and *s1.2* components are two components which do not belong to the application to be migrated but to system 1 (*s1*). Nevertheless, the *migration target component 2* interacts with them and therefore *s1.1* and *s1.2* are of relevance to the migration strategy selection. The system does not use virtualization and is not highly scalable.

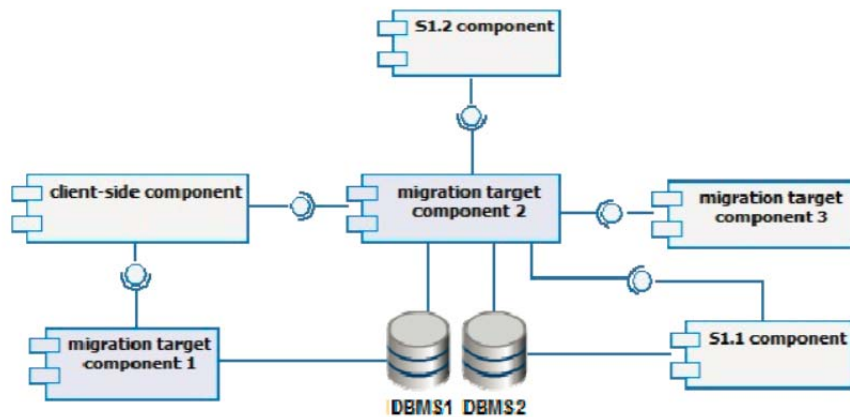


Figure 1: Typical Enterprise System today, on-premises deployment²

In Figure 2 a new node is included, namely the Cloud provider. The Cloud provider runs the migrated components, namely the *s1.2 component*, the *migration target components 1* and *3*. On the left side, i.e. the client side, a new component has been created, namely a *broker*, as a result of the migration of parts of the system to the Cloud. Part of the knowledge previously on the migrated components has been transferred to the *broker*. Surprisingly enough, the *migration target component 3* could not be migrated due to its high reliance on database *DBMS1*. It might be because the *migration target component 3* accesses the database (*DBMS1*) too often or because the traffic load between them is very high. Therefore, moving this component out of the local infrastructure could harm the overall system performance. On the other hand, with respect to the migrated components, a component which was not targeted (*s1.2 component* in Figure 2) has been shifted to the Cloud provider along with *migration target components 2* and *3*. This decision could be taken in order to avoid an SLA infringement due to an increased transactional delay in case the *s1.2 component* would have been kept on the client side.

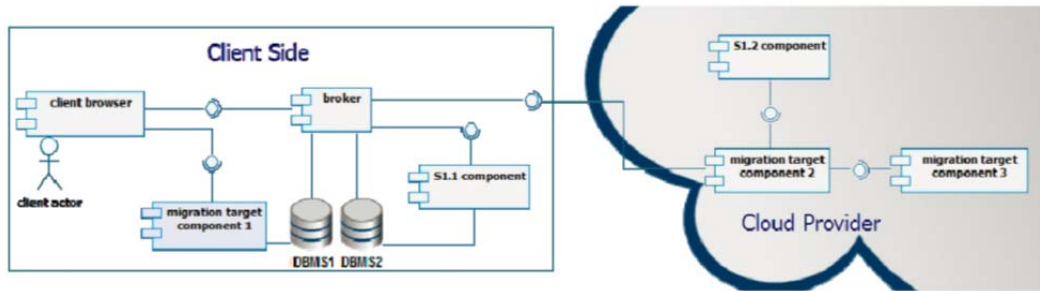


Figure 2: Hybrid migration approach

Dynamic Performance Bottlenecks

Typical enterprise applications today, like the one presented, employ multi-tiered architecture with lots of components which depend on each other. This fact entails some design challenges which have the potential to hinder the migration process. In this respect, it is important to bear in mind applications' performance bottlenecks and their dynamic nature. As an application scales in a horizontal fashion, a dynamic performance bottleneck might move from one part of the system to another.

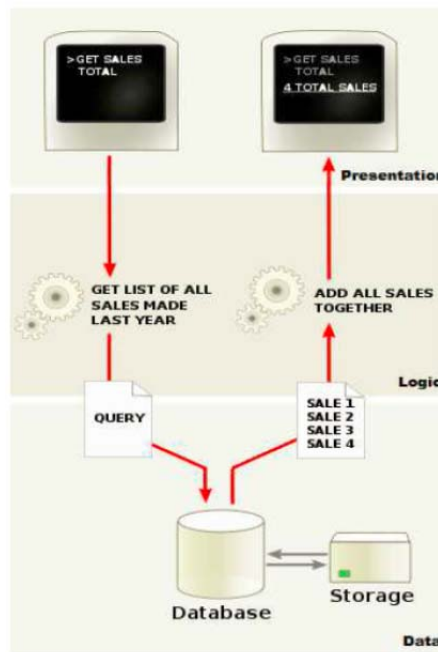


Figure 3: Typical three-tiered enterprise application

In Figure 3, a typical three-tiered enterprise application is depicted in order to exemplify the concept of dynamic performance bottleneck and how it affects a migrated application. It does therefore not arise from any empirical finding but it is just an illustrative example.

At the bottom of Figure 3, the back-end layer (Tier-1) consists of a database which interacts with the components in the immediately superior level (Tier-2), which belong the business logic layer. The business logic components query the database in order to get information related to the sales made the previous year. Finally, the layer at the top, the front-end layer, presents the data to the user (Tier-3).

Such a deployment has some effects on the dynamic performance bottlenecks as they might shift to different parts of the system as the system scales out. Let us assume that in Figure 3 the components of the business logic layer (Tier-2), which get their data from the bottom layer, are the current bottleneck of the application. As a result of the utilization of the proposed framework, Tier-2 components might be selected as target components to be migrated to the Cloud in order to keep up with the SLA. Let us assume that due to this decision the system performance keeps on improving until so many requests are sent to the back-end layer that its components cannot respond timely to the requests of the upper-layer. At this point, the dynamic performance bottleneck has been shifted from where it initially was, Tier-2, to the back-end layer. The depicted framework might then decide, by using the presented model (*see* Modelling of components placement), that the bottom layer should be scaled out in an attempt to improve the system performance. However, the requirements related to enterprise policies and data privacy are also factored in the presented model. Therefore, the decision support system could bring about a different migration strategy in order to meet those requirements too.

1.1 Evaluation

Aiming at evaluation, a legated Product Configurator will be migrated to a virtualized environment by using the provided framework. As a result the application will be hosted partly on the Cloud and locally. It will be used as a case study to show the advantages of a hybrid approach as a cloudenabling. Moreover the validation will show that the model for component placement effectively works and respect the SLA and enterprise-specific constraints. Firstly, the application used as a case study to exemplify the advantages of the presented approach will be modelled in order to describe it in terms of the nature of the users (internal, external) and its components (back-end, business logic, front-end). Moreover the transactions from the users to the front-end components (thin clients) or business logic components (thick clients), and the transactions between components will be analyzed. Once this model is ready, the typical communications happening in the system will be measured on an end-to-end fashion as well as the transactions between users and components and between different components will be assessed. As a result, communication delays between local components, migrated components, internal users, and external users will be estimated. In addition to the overall transaction sizes and their frequency, the transactions between components will be carefully taken into account as they

could have a large effect on the migration strategy. As an example, the migration of a business logic component to a Cloud environment might not be beneficial whereas its migration together with the back-end it deeply rely on in order to satisfy users' requests, may imply great economical gain at little transactions cost. The presented framework will use the model of the application in order to estimate the migration benefits and communication costs in order to recommend a migrated deployment (in case it is beneficial) depending on the maximum delays allowed for the application and their variance. On the one hand, the benefits will be calculated from the migration to the Cloud of both computation and storage (Khajeh-Hosseini, Greenwood, et al., 2010). On the other hand, the communication costs are related to the migration of data and application state to the Cloud (Armbrust et al., 2010). Additionally, it will be verified that the traffic is routed accordingly to the final migrated deployment in order to profit from the new architecture. For example, users external to the architecture are routed to the components on the Cloud as far as it is possible.

2 Research Method

It is argued for a design-oriented information systems approach (Österle et al., 2010), in which a system is build up bearing the conceptual design in mind while taking into account both applicable restrictions and limitations. However, the precise steps might either change as this work evolves and some steps may be executed iteratively until an appropriate outcome is reached. Nine steps have been identified: 1. Conduct a literature review on: Cloud Computing and Internet data centres, case studies, frameworks and approaches to the migration of legacy applications to the Cloud, descriptive modelling techniques, application discovery, and distributed system economics. 2. Conduct interviews or have informal meetings with other member of the scientific and professional communities in order to clearly identify and further define the relevant problems. 3. Choose the correct Hybrid Cloud approach to create a framework for the migration of legacy enterprise applications to the Cloud. 4. Clearly define the modelling of components' placement. 5. Use the migration of a product configurator to the Cloud as a real-world use case to exemplify and evaluate the framework and defined models. 6. Conduct experiments based on a real-world use case to be deployed in an enterprise environment with the resulting realistic evaluation. 7. Statistically analyse the performed measurements and use those results in order to improve the modelling of component placement. 8. (Optional) Study the migration to different Cloud providers depending on the specific requirements of the application. 9. Gather all the experience, observations and experimental results in a doctoral dissertation.

3 Conclusions and Acknowledgements

Although the Cloud offers many benefits, the migration to a cloud-based architecture needs to be carefully planned. Organizations need to take many criteria into account in order to adopt an adequate migration strategy to select the appropriate parts of the application and target infrastructure for the migration. This paper calls for a framework to assist in the migration following a hybrid Cloud deployment in which parts of the application are kept locally and which parts are migrated. Decision making is supported by a model for component placement. Finally, both the framework and models are validated in a real-case scenario, namely a Product Configurator. This research has been supported by the FP7 Marie Curie Initial Training Network “RELATE” for research on Engineering and Provisioning of Service Based Cloud Applications. Grant Agreement No. 264840. 7

4 References

- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., . . . Stoica, I. (2010). A view of cloud computing. *Communications of the ACM*, 53(4), 50-58.
- Babar, M. A., & Chauhan, M. A. (2011). *A tale of migration to cloud computing for sharing experiences and observations*. Paper presented at the SECCLOUD '11 Proceedings of the 2nd International Workshop on Software Engineering for Cloud Computing.
- Binz, T., Leymann, F., & Schumm, D. (2011). *CMotion: A Framework for Migration of Applications into and between Clouds*.
- Calheiros, R. N., Ranjan, R., Beloglazov, A., De Rose, C. A. F., & Buyya, R. (2011). CloudSim: a toolkit for modeling and simulation of cloud computing environments and evaluation of resource provisioning algorithms. *Software: Practice and Experience*, 41(1), 23-50.
- Fox, A., & Griffith, R. (2009). Above the clouds: A Berkeley view of cloud computing. *Dept. Electrical Eng. and Comput. Sciences, University of California, Berkeley, Rep. UCB/EECS*, 28.
- Frey, S., & Hasselbring, W. (2010). *Model-Based Migration of Legacy Software Systems to Scalable and Resource-Efficient Cloud-Based Applications: The CloudMIG Approach*.
- Hajjat, M., Sun, X., Sung, Y. W. E., Maltz, D., Rao, S., Sripanidkulchai, K., & Tawarmalani, M. (2010). *Cloudward bound: planning for beneficial migration of enterprise applications to the cloud*.

- IDC. (2010). Cloud Computing. An IDC Update. Khajeh-Hosseini, A., Greenwood, D., & Sommerville, I. (2010). *Cloud migration: A case study of migrating an enterprise it system to IaaS*.
- Khajeh-Hosseini, A., Sommerville, I., & Sriram, I. (2010). Research challenges for enterprise cloud computing. *Arxiv preprint arXiv:1001.3257*.
- Lewis, G. A., Morris, E. J., Smith, D. B., & Simanta, S. (2008). Smart: Analyzing the reuse potential of legacy components in a service-oriented architecture environment: DTIC Document.
- Marston, S., Li, Z., Bandyopadhyay, S., Zhang, J., & Ghalsasi, A. (2011). Cloud computing—The business perspective. *Decision Support Systems*, 51(1), 176-189.
- Menzel, M., & Ranjan, R. (2011). CloudGenius: Automated Decision Support for Migrating Multi- Component Enterprise Applications to Clouds. *Arxiv preprint arXiv:1112.3880*. Mohagheghi, P., & Sæther, T. (2011). *Software Engineering Challenges for Migration to the Service Cloud Paradigm: Ongoing Work in the REMICS Project*.
- Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., Sinz, E. J. (2010). Memorandum on design-oriented information systems research. *European Journal of Information Systems*, 20(1), 7-10.
- Rosado, D. G., Gómez, R., Mellado, D., & Fernández-Medina, E. (2012). Security analysis in the migration to cloud environments. *Future Internet*, 4(2), 469-487.
- Seung, Y., Lam, T., Li, L. E., & Woo, T. (2011). *CloudFlex: Seamless scaling of enterprise applications into the cloud*. US NIST, M., P. Grance, T.,. (2009). The NIST definition of cloud computing *National Institute of Standards and Technology* (Vol. 53, pp. 50). Vaquero, L. M., Rodero-Merino, L., Caceres, J., & Lindner, M. (2008). A break in the clouds: towards a cloud definition. *ACM SIGCOMM Computer Communication Review*, 39(1), 50-55.
- Vouk, M. (2008). *Cloud computing—Issues, research and implementations*.

Entscheidungsunterstützung in logistischen Netzwerken

– Ansätze für eine Umsetzung

Axel Klarmann, Martin Roth, Bogdan Franczyk

*Wirtschaftswissenschaftliche Fakultät
Universität Leipzig*

Abstract

Die Logistik stellt als Bindeglied zwischen den Teilnehmern einer zunehmend globalisierten Wertschöpfungskette einen wesentlichen Teil der Wertschöpfungskette dar. Aufgrund einer Spezialisierung auf die unternehmensinternen Kernkompetenzen, durch immer stärkere Schmälerung der Gewinnmargen und globalen Wettbewerb, ergibt sich die Notwendigkeit logistische Funktionen vermehrt an spezialisierte Dienstleister abzugeben. In diesem Umfeld ist in den letzten Jahren das Konzept des Kontraktlogistikern erwachsen, welcher die Aufgabe der Planung, Kontrolle und Steuerung des zumeist hochdynamischen Dienstleisternetzwerks übernimmt. Insbesondere die Kontrolle und Steuerung stellt das Konzept des Kontraktlogistikern vor eine Herausforderung. Hierzu wird in diesem Beitrag die Anwendung von aktuellen Konzepten der Business Intelligence diskutiert, sowie ein möglicher Ansatz für eine Umsetzung zur Entscheidungsunterstützung, sowie deren Voraussetzungen vorgestellt.

2 Einleitung

Im Zuge eines weitgehend globalisierten und volatilen Wirtschaftsumfelds, ist die Logistik, im Sinne des Umschlags, Lagerns und Transports von Gütern, zu einer der Schlüsselfaktoren der Wettbewerbsfähigkeit von Unternehmen erwachsen. Aufgrund der zunehmenden Variabilität der Märkte stellt dazu die Planung, Kontrolle und Steuerung der teils komplexen Netzwerke, Unternehmen vor große Herausforderungen. Dies gerade im Zuge einer zunehmenden Orientierung auf Rentabilität, Flexibilität und Agilität, im Sinne einer schnellen Adaption an veränderte Umweltbedingungen (Pfohl, 2010).

In diesem Licht entstand in den letzten Jahren der Begriff des Fourth-Party-Logistikern (4PL) bzw. Kontraktlogistikern (LLP) durch das Beratungsunternehmen Andersen/Accenture (Hoek,Chong 2001). Aufgabe eines solchen ist die Gestaltung und der Betrieb eines firmenspezifischen Netzwerkes und die Integration in die bestehende Pro-

zesslandschaft des Auftraggebers. Als Besonderheit des Geschäftsmodells des 4PL/LLP wurde in (Bretzke, 2008) herausgestellt, dass dieser wiederum auf die Ressourcen von weiteren Logistikdienstleistern zurückgreift, um die letztliche Dienstleistung eines Logistikers zu erbringen, ohne eigene Logistik-bezogene Ressourcen zu besitzen. Der Schwerpunkt der Dienstleistung ist damit nach (Bretzke, 2008) die Kontrolle und Steuerung verschiedener, operativ tätiger Dienstleister. Das Ressourcen-freie Geschäftsmodell soll dabei die Flexibilität und die Unabhängigkeit des Dienstleisters sicherstellen, welcher damit als fokale Schnittstelle zwischen einer Vielzahl an verketteten Dienstleistern und dem Auftraggeber fungiert.

Im Zuge der Umsetzung der Dienstleistung eines Logistikers, insbesondere hinsichtlich einer verteilten Entscheidungsfindung durch die eingebundenen Dienstleister, werden im Folgenden das Konzept des 4PL/LLP näher erläutert, sowie die Herausforderung bzgl. einer Entscheidungsunterstützung formuliert, Ansätze für eine Entscheidungsunterstützung diskutiert, sowie eine Ausblick auf eine mögliche Umsetzung eines Systems zur verteilten Entscheidungsunterstützung gegeben.

3 Kontraktdienstleister und Anforderungen an Monitoring/Steuerung

Zunächst soll eine Einordnung des LLP in die Organisationstypologie vorgenommen werden, welches die besonderen Charakteristika des Geschäftsmodells betont. Anschließend sollen die Möglichkeiten einer Steuerung und Kontrolle aufgezeigt werden.

Das Geschäftsmodell des LLP konzentriert sich auf die Einbindung von logistischen Dienstleistern, welche per Vertrag bzw. Service Level Agreement (SLA) in ein Netzwerk, zur Erbringung einer Gesamtdienstleistung gegenüber einem Auftraggeber gebunden werden. In diesen SLA werden bestimmte Eigenschaften bzgl. der Erbringung der Dienstleistung, als Service Level Objective (SLO) bezeichnet, definiert und durch den Dienstleister garantiert. Organisationstypologisch kann das entstehende, vertragliche Netzwerk nach (Hess, 2002), welcher zwischen den extremen Ausprägungen „Markt“ und „Hierarchie“ eine Vielzahl an Unternehmenstopologien benennt und unter dem Oberbegriff Unternehmensnetzwerk subsummiert, eingeordnet werden. Je nach Organisationstyp variiert in diesen Netzwerken die Möglichkeit der Einflussnahme und Stabilität, sodass nach (Alt, Legner, Österle, 2005) bzw. (Sydow, 2006) der LLP als ein strategisches Netzwerk aufgefasst werden kann. Dieses ist gekennzeichnet durch eine fokale Steuerung, in (Sydow, 2006) als „bounded autonomy“ bezeichnet, mehr als zwei, enger als marktüblich, verbundenen Unternehmen und einer stabilen Aufgabenzuordnung.

Die klassischen Aufgaben eines Logistikers können mit Lagerung, Umschlag und Transport von Gütern umrissen werden. Durch Mehrwertlogistiker werden diese Basisaufgaben durch weitere Dienstleistungen ergänzt, die bspw. die Verpackung, Aufbau und Konfiguration von Transportgut umfassen können. Diese Aufgaben können mit zunehmender Spezifität nicht mehr durch den einzelnen Dienstleister erbracht werden, welches zur Notwendigkeit der zentralen Planung, Kontrolle und Steuerung einer Vielzahl von Dienstleistern führt. In diesem Aufgabenfeld, mit Fokus auf die logistische Dienstleistung, findet sich das Geschäftsfeld des LLP. Der Auftraggeber für den Betrieb einer logistischen Wertschöpfungskette beauftragt zu diesem Zweck den LLP mit der Planung sowie den Betrieb der Kette und vereinbart die entsprechenden Parameter, wie Qualität, Flexibilität und Zuverlässigkeit der Ausführung in einem SLA mit entsprechendem Reporting. Der LLP übernimmt die Analyse der logistischen Kette und vereinbart, als „manufacturer[s] without factories“ (Braham, 1985), mit Dienstleistern aus seinem Pool entsprechende Verträge/SLA. Durch Simulation werden die Rahmenbedingungen für die Erbringung der Dienstleistung eines jeden Teilnehmers in Form von SLO ermittelt und ausgehandelt, sodass im Zuge der Implementierung die IT-Systeme der Dienstleister integriert werden können. In der operativen Phase ist es Aufgabe des LLP die Aufträge des fokalen Unternehmens in einen Warenstrom in dem konfigurierten, logistischen Netzwerk im Rahmen des ausgehandelten SLA zu überführen und die Dienstleistungen zu koordinieren und zu steuern.

Die Steuerung und Koordination erfolgt zunächst nur über die ausgehandelten SLA anhand der ursprünglichen Grobplanung und stellt damit eine Koordination auf Basis von Verträgen dar. Diese Verträge sind allerdings aufgrund der langfristigen Ausrichtung, insbesondere unter der Berücksichtigung der Principal-Agent-Theory, nicht ausreichend, eine vollständige Koordination im operativen Umfeld zu gewährleisten. In (Placzek, Kohler, 2003) wird die Erfahrung von Miebach Logistik (Miebach, 2003) aufgezeigt, welche den Einbezug von operativer Kennzahlen zur verbesserten Beurteilung der Kundenbeziehung, Verbesserung der operativen Umsetzung der Strategie und zur präziseren Messung der Zielerreichung vorschlägt. Dieser Vorschlag wird durch (Haasis, 2008) bestätigt. In diesem Licht ist die Erkenntnis von (Seufert, Lehmann, 2006), dass nur 50% der Logistikunternehmen ein Werkzeug zur Verfolgung der entsprechenden Kennzahlen einsetzen, entsprechend zu werten. Für den LLP, als zentralen Integrator und Koordinator ist es entsprechend schwer, die für eine Steuerung notwendigen Kennzahlen aus den Systemen der extern eingebundenen Dienstleister zu extrahieren. Hinzu kommt die in (Scholz, Schieder, Kurze, et al, 2010) und (Schulze, Dittmar, 2010) als problematisch beschriebene, fehlende Standardisierung von Kennzahlen auf fachlicher Ebene im Unternehmen, damit auch über Unternehmensgrenzen hinweg. Als Alternative zur direkten Extraktion von Kennzahlen aus den Systemen der Netzwerkpartner über Extract-Transfer-Load

(ETL) bietet der Ansatz einer ereignis- bzw. prozessgeführten Ermittlung dieser, anhand von Ereignissen die während eines Prozessdurchlaufs auftreten, eine mögliche Alternative. Der in (Roth, 2011) skizzierte Ansatz einer Event-getriebenen Aggregation von Ereignissen durch Complex Event Processing (CEP), siehe auch (Chandy, Schulte, 2010), (Luckham, 2010) und Ermittlung von Performance-Kennzahlen, sowie deren Prüfung auf Einhaltung von SLO, kann hierbei als Basis für eine Extraktion der entsprechenden Kennzahlen für einen Dienstleister genutzt und in ein Informationssystem zur Entscheidungsunterstützung (Kap. 3) integriert werden. Mit diesem technischen Ansatz und der fachlichen Integration über ein unternehmensübergreifendes Kennzahlensystem kann der LLP seiner Rolle als Integrator in einem Unternehmensnetzwerk gerecht werden und die Forderungen auf der Prozessebene (Alt, Legner, Österle, 2005) erfüllen. Diese umfassen zum einen Transparenz, zum Beispiel durch den Einsatz von Dashboards und Portale, sowie Steuerung, manuell oder semi-automatisch mittels eines Regelsystems, und Optimierung, durch abgestimmte Heuristiken und Prognosemethoden.

4 Entscheidungsunterstützung im Unternehmensnetzwerk

Entscheidungen müssen in dem in Kap. 2 beschriebenen Unternehmensnetzwerk über alle Ebenen des Geschäftsbetriebs, strategisch bis operativ, getroffen werden. Hierbei ist festzustellen, dass operative Entscheidungen zumeist dezentral durch die Dienstleister, strategische Entscheidungen maßgeblich durch den LLP getroffen werden. Zur Unterstützung von Entscheidungsträgern haben sich entsprechende IT- und Kommunikationssysteme unter der generellen Bezeichnung Management Support Systeme (MSS) entwickelt. Die zu treffenden Entscheidungen variieren hierbei maßgeblich bzgl. ihrer Dringlichkeit und Höhe der Auswirkung. Speziell der Bereich der Führungsinformationssysteme mit den Systemen, welche sich unter der Dachbezeichnung Business Intelligence (BI), sammeln, fand in den letzten Jahren intensive Anwendung im wirtschaftlichen Kontext, insbesondere zur Unterstützung der Analysephase in Vorbereitung einer Entscheidung. Diese als „klassische BI“ (Gluchowski, 2009) bezeichneten Systeme die gekennzeichnet sind durch Prozesse der Datenextraktion (ETL), Datenkonsolidierung in einem Data Warehouse (DW), dem explorativem bzw. informativen Zugriff über Dashboards bzw. Online-Analytical-Processing (OLAP) sind dabei in Bezug auf die Entscheidungsunterstützung stark auf das einsetzende Unternehmen ausgerichtet, beziehen kein unternehmensübergreifende Vermittlung ein und sind reaktiv ausgerichtet. Dazu werden die Reporte und Extraktionen im ETL-Prozess in groben Intervallen (Stunden bis Tage) generiert bzw. durchgeführt, was einen Einsatz auf operativer und taktischer Ebene aufgrund der damit verbundenen Latenzen verhindert. Der durch (Gluchowski, 2009) beschriebene Ansatz einer operational BI (opBI) verspricht dahingegen einen zeitnahen und prozessintegrierten Zugriff auf Informationen zur Entscheidungsunterstützung bis in den operativen Be-

reich. (Gluchowski, Gabriel, Dittmar, 2008) zeigen dazu den unternehmensübergreifenden Aspekt auf und skizzieren eine mögliche Architektur für ein solches System. Diese Architektur empfiehlt sich für eine Integration des in Kap. 2 benannten CEP-Ansatzes, um die Prozessinformationen aus dem Netzwerk für eine Entscheidungsunterstützung bei den Netzwerkteilnehmern einzusetzen bzw. Daten aus den Systemen der Netzwerkteilnehmer durch einen Vergleich mit den aus ETL-Prozessen extrahierten Daten zu validieren. Hierbei ist insbesondere der temporale Aspekt eines Events zu beachten und dessen Integration in ein BI-Umfeld entsprechend zu konzipieren, sodass aggregierte Kennzahlen aus dem CEP-Prozess und aus dem ETL-Prozess konsistent in dem DW gehalten werden. Weiterhin ist ein einfacher, situationsgerechter und verteilter Zugriff auf die Daten sicherzustellen, welches die Integration des Konzeptes der mobile BI empfiehlt. Insbesondere die aktuellen Anforderungen an den Zugriff auf Anwendungssysteme über mobile Endgeräte, gerade im Logistikumfeld, wie diese in (Hompele, 2012) beschrieben werden, und im Mittelstand allgemein, welche die Studie von (Seidler, Mack, Bange, 2012) aufzeigen, stützen diesen Anspruch. Durch die kurze Reaktionszeit bei operativen Entscheidungen ist dabei zu prüfen, inwiefern Entscheidungen bereits automatisch bzw. semiautomatisch erfolgen können. Hierzu ist eine Einbindung von Regeln bzw. Regelsystemen, wie bspw. durch (Halle, Goldberg, 2010) beschrieben in die Gesamtarchitektur zu berücksichtigen, um bspw. das Eskalationsmanagement zu automatisieren bzw. Dienste im Sinne einer Serviceorientierten Architektur (SOA) auszuführen. Um die Dringlichkeit und Notwendigkeit einer Reaktion zu ermitteln, ist der Kontext einer Entscheidung im System abzubilden, um valide Entscheidungen zu ermöglichen.

5 Ansätze für eine Umsetzung für Kontraktdienstleister

Um eine in Kap. 3 skizzierte Architektur zu ermöglichen ist es notwendig ein Kennzahlensystem einzusetzen, welches von allen potentiellen Netzwerkteilnehmern getragen und verstanden wird. In diesem Zuge wurden Referenzmodelle zum Aufbau logistischer Dienstleistungen wie bspw. das Supply Chain Operations Reference-Modell (SCOR) des Supply Chain Councils (SCC) analysiert, um ein entsprechendes Kennzahlensystem, welches operative bis strategische Kennzahlen und deren Aggregationsregeln beinhaltet, aufzubauen. Die Analyse zeigt, dass eine integrierte Sicht auf ein strategisches Unternehmensnetzwerk eines LLP derzeit fehlt. Insbesondere die Aggregation und Transformation in eine Sicht für das fokale Unternehmen, mit entsprechend definierten Reportingintervallen und -inhalten bedarf der Entwicklung eines geeigneten Modells. Im Zuge der durch (Stein, 2010) erfolgten Analyse bzgl. einer konfigurierbaren Erweiterung von Referenzmodellen, bspw. des SCOR-Modells, soll ein entsprechendes Modell im Weiteren aufgebaut werden. Dazu wird in Anlehnung an (Otto, 2003), (Nguyen, Schiefer, Tjoa, 2005), (Wu, Barash, Bartolini, 2007) eine Architektur für ein Event-getriebenes Informa-

tionssystem entworfen, welches die Ansätze einer operational BI und einer zunehmenden Entscheidungsunterstützung auf Basis von Geschäftsregeln integriert (Abbildung 1). Durch den Einsatz des CEP-Ansatzes wird eine zeitnahe Information des Entscheiders und durch die entsprechenden Geschäftsregeln eine einfache Auswahl von Entscheidungen, unter Beachtung der strategischen Ausrichtung des Gesamtnetzwerks, ermöglicht.

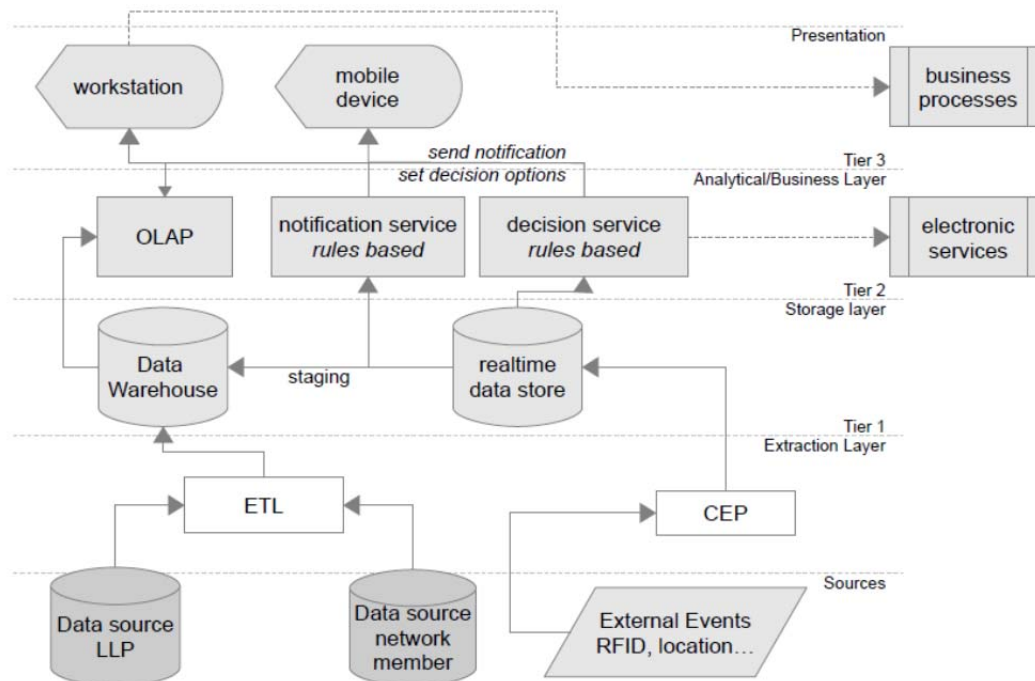


Abbildung 1: Skizze der avisierten Architektur, in Anlehnung an (Otto, 2003)

6 Einordnung in die bestehende Literatur

Neben der in (Roth, 2011) diskutierten Möglichkeit zur Nutzung von CEP-basierten Aggregation von Daten in logistischen Netzwerken aus dem Netzwerk, existiert noch kein entsprechender Ansatz für eine verteilte Entscheidungsunterstützung auf operativer bis strategischer Ebene im logistischen Bereich, wobei (Nguyen, Schiefer, Tjoa, 2005) zumindest eine Architektur entwerfen, welche Events in eine service-orientiertes BI-Konzept integrieren. Allerdings gehen die Autoren nicht auf die verteilte Entscheidungsfindung und das Problem der Vereinheitlichung der Kennzahlen ein. Einen Ansatz zur Service-Orientierung der BI-Architektur liefern (Wu, Barash, Bartolini, 2007), wobei kein Bezug auf eine operative Entscheidungsfindung und ein einheitliches Kennzahlensystem genommen wird. (Taylor, Raden, 2007) beschreiben auf operativer Ebene die Notwendigkeit Entscheidungen zu automatisieren und weisen in diesem Zusammenhang insbesondere auf die Hebung von versteckten Entscheidungen hin. Diese Automatisie-

rung ist aufgrund der Menge an notwendigen Entscheidung durch den LLP, als auch durch die Dienstleister, welche wiederum im gesamten Netzwerk abgestimmt sein müssen, relevant. In die Modellierung eines Regelsystems führen dabei (Halle, Goldberg, 2010) ein und (Rosen, 2010) berücksichtigt in seinem Modell die Integration in eine Service-orientierte Infrastruktur, allerdings ohne Bezug auf eine verteilte Entscheidungsfindung, respektive der Integration in eine BI-Lösung. (Werner, et al, 2010) zeigen Ansätze auf, wie eine Wertschöpfungskette, im Speziellen an der Schnittstelle zwischen Supply Chain Management (SCM) und Customer Relationship Management (CRM), unternehmensübergreifend mittels eines Cockpit-Konzepts visualisiert werden und wie das Konzept der collaborative BI in die Anwendungslandschaft integriert werden kann. (Otto, 2003) ermittelt aus der Perspektive des SCM drei Sichten auf den Einsatz eines Supply Chain Event Managements (SCEM), wobei dieser Events als Status/Meilensteine im Ablauf von Geschäftsprozessen definiert werden und schlägt eine Architektur für die Umsetzung eines solchen Systems vor. Hierbei wird allerdings nicht auf die Möglichkeiten eines CEP eingegangen. (Stefanovic, 2011) weist insbesondere auf die Integration von Prozesskennzahlen aus einem Business Activity Monitoring (BAM) hin und stellt eine entsprechende Architektur für eine integrierte Betrachtung im Rahmen eines Performance Management Portals vor. Bei der Betrachtung fehlt allerdings der direkte Bezug zur Logistik und der Anspruch auf die Ubiquität der Anwendung wird nicht berücksichtigt. Bzgl. der Notwendigkeit eines generellen Event- Managements, im Sinne von Hinweisen auf Problemen im logistischen Netzwerk, sei auf (Werner, 2008) verwiesen, wobei sich der beschriebene Ansatz ausschließlich auf die Hilfsmittel „Alert-Management“, „Workflow-Management“ und „Tracking and Tracing“ beschränkt, dabei also keine Lösung für eine verteilte Entscheidungsunterstützung geboten wird.

Auf Seiten der Referenzmodellierung für logistische Systeme kann auf das SCOR gebaut werden, sowie auf die abgeleitete Arbeit zum Industrial Services Reference Model in (Milano, 2009), dem Ansatz von (Schuh, Schmidt, Rinis, et al., 2008) und den im Rahmen des InCoCo-S Projekts (Kleinert, 2011) erstellten Modell, welches Mehrwertdienste berücksichtigt. Die Konfigurierbarkeit von Referenzmodellen wird in (Stein, 2010) untersucht und ein konfigurierbarer Ansatz für das SCOR-Modell vorgeschlagen, welcher für die weitere Anpassung mit unternehmensübergreifenden Kennzahlen, speziell in der Logistik, herangezogen werden kann und diese um Logistik-spezifische Kennzahlen erweitert.

7 Quellen

Alt, R., Legner, C., Österle, H. (2005). Virtuelle Organisation – Konzept, Realität und Umsetzung. Virtuelle Organisation – Konzept, Realität und Umsetzung

- Braham, P. (1985). Marks & Spencer: a technological approach to retailing. Implementing new technologies
- Bretzke, W.-R. (2008). Logistische Netzwerke. Springer
- Chandy, K. M., Schulte, W. R. (2010). Event processing: Designing IT Systems for Agile Companies. McGraw Hill
- Gluchowski, P. (2009). Operational Business Intelligence. BI-Spektrum
- Gluchowski, P., Gabriel, R., Dittmar, C. (2008). Management Support Systeme und Business Intelligence.
- Haasis, H.-D. (2008). Produktions- und Logistikmanagement. Hess, T. (2002). Netzwerkcontrolling.
- Kleinert, A. (2011). InCoCo-S Publishable Final Activity Report.
- Luckham, D. (2010). The Power of Events. Addison-Wesley
- Miebach Logistik (2012). Eine Balanced Scorecard - maßgeschneidert für die Logistik. <http://www.mylogistics.net/de/news/themen/key/news40936/jsp>
- Milano, P. (2009). Industrial Services Reference Model. Proceedings of the 1st CIRP Industrial Product-Service Systems (IPS2) Conference
- Nguyen, T. M., Schiefer, J., Tjoa, A M. (2005). Sense & Response Service Architecture (SARESA): An Approach towards a Real-time Business Intelligence Solution and its use for a Fraud Detection Application. DOLAP 2005
- Otto, A. (2003). Supply Chain Event Management: Three Perspectives. International Journal of Logistics Management
- Pfohl, H.-C. (2010). Logistiksysteme: Betriebswirtschaftliche Grundlagen. Springer
- Placzek, T., Kohler, U. (2003). Balanced Scorecard für die Logistik, Uni-Duisburg und Miebach entwickeln die „Miebach Logistik Scorecard“. Deutsche Logistikzeitung
- Rosen, M. (2010). The decision model; Service-Oriented Architectures. CRC Press
- Roth, M., Donath, S. (2011). Applying Complex Event Processing towards Monitoring of Multi- Party Contracts and Services for Logistics - A Discussion. 5th International Workshop on Event-Driven Business Process Management
- Scholz, P., Schieder, C., Kurze, C., Gluchowski, P., Böhringer, M. (2010). Benefits and Challenges of Business Intelligence Adoption in Small and Medium-Sized Enterprises. 18th European Conference on Information Systems
- Schuh, G., Schmidt, C., Rinis, M., Garg, A., Kleinert, A., Quick, J., Salhani, O. (2008). Integriertes prozess- und kennzahlenbasiertes Referenzmodell für Logistikanbieter.

- Schulze, K.-D., Dittmar, C. (2010). State of the Art von Business Intelligence – Ergebnisse der biMA® - Studie 2009. BI & BPM Kompendium 2010
- Seidler, L., Mack, M., Bange, C. (2012). Business Intelligence im Mittelstand 2011/2012 - Status quo, Ausblick und Empfehlungen.
- Seufert, A., Lehmann, P. (2006). Business Intelligence – Status quo und zukünftige Entwicklungen. HMD Praxis der Wirtschaftsinformatik
- Stefanovic, N., Stefanovic, D. (2011). Supply chain performance measurement system based on scorecards and web portals. Computer Science and Information Systems
- Stein, A. (2010). Erweiterung des Supply Chain Operations Reference-Modells.
- Sydow, J. (2006). Management von Netzwerkunternehmen. Gabler Verlag.
- t. Hompel, M. (2012). IT in der Logistik.
- Taylor, J., Raden, N. (2007). Smart (Enough) Systems. Prentice Hall
- v. Halle, B., Goldberg, L. (2010). The decision model. CRC Press
- Van Hoek, R. I., Chong, I. (2001). Epilogue: UPS Logistics – Practical Approaches to the Esupply Chain. International Journal of Physical Distribution & Logistics Management
- Werner, A., Hrach, Ch., K. Heyden, Alt, R., Franczyk, B. (2010). Value Chain Cockpit auf Basis betrieblicher Anwendungssysteme. MKWI 2010 - MKWI 2010 – Enterprise Resource Planning und Transformation von ERP-Systemen
- Werner, H. (2008). Supply Chain Management. Gabler
- Wu, L., Barash, G., Bartolini, C. (2007). A Service-oriented Architecture for Business Intelligence. Service-Oriented Computing and Applications, 2007. SOCA '07 9

Towards an Economic Foundation for the Decision between Agile and Plan-driven Project Management in a Business Intelligence Context

Sarah Otyepka, Benjamin Mosig, Marco C. Meier

*FIM Research Center
University of Augsburg*

Abstract

Lacking a formal yet practical decision model, nowadays decision makers mostly follow corporate guidelines or their intuition when it comes to the decision between agile and plan-driven project management in Business Intelligence projects. As one size does not fit all, using management methods hyped by temporary fashion or other management methods not adapted to the situation bears the risk of project failure. Thus, this paper proposes a risk-adjusted net present value-based model to support decision makers in the selection of the appropriate management method for Business Intelligence projects. We focus on two decisive risk parameters – the likelihood of environmental changes and the peril of improper system integration – and a project’s estimated cash flows. As a result, the tradeoff between different characteristics of risks and cash flows in a specific project is formalized and made transparent. In summary, this research-in-progress paper sketches the idea of a practical decision model that improves the foundation for the selection of the appropriate management method.

1 Motivation

Although studies suggest that non-technical factors such as project management dominate a Business Intelligence (BI) project’s success (Adamala & Cidrin, 2011), the decision on a project management method in BI projects, such as Scrum, Extreme Programming, the waterfall model or Sandboxing, is often not based on rational considerations (Project Management Methodologies. n. d.). Examples include German authorities that require any in-house information technology (IT) project to use the V-model XT (Höhn & Höppner, 2008, p. IX). In other cases, the decision on the project management method is the result of nothing more than “this is what we’ve always done” (Project Management

Methodologies. n. d.). Thus, many companies or authorities seem to give either mandatorily “one-size-fits-all” instructions or apply project management methods inconsiderately.

The Project Management Institute defines project management as “the application of knowledge, skills, tools and techniques to project activities to meet the project requirements” (Project Management Institute, 2008, p. 6). While project management describes the necessity to use knowledge, skills and tools, a project management method(ology) describes more precisely how this knowledge, skills and tools look like: “A methodology is a set of guidelines or principles that can be tailored and applied to a specific situation. In a project environment, these guidelines might be a list of things to do” (Charvat, 2003, p. 3). As the terms “method” and “methodology” are often used interchangeably in IT-related literature (Boehm & Turner, 2003; Charvat, 2003; Larman & Basili, 2003; Moss, 2009), we will consistently use “method” in the following.

There are important differences between BI and other IT or software development projects that need to be addressed. While BI projects are rather data-centric business integration projects, being closely linked to a company’s strategy, software development projects are rather code-centric (Moss, 2009). This is the reason why non-technical success factors dominate technical success factors in BI projects (Adamala & Cidrin, 2011). Hence, the selection of an appropriate project management method is critical. Instead of an ex ante determined project management method that might ultimately result in project failure, a situation-based decision prevents the application of a potentially inappropriate one (Charvat, 2003, pp. 18-20). Therefore, a situation-based decision seems more promising. BI projects – as any kind of projects – tie up capital and accordingly should be seen as investments supposed to increase shareholder value. For the required evaluation of suitable project management methods, a value-based and future-oriented approach is adequate (Coenenberg, Mattner, & Schultze, 2003, p. 3; Mertens, 1999, p. 11).

Surprisingly, “there are few [researchers] that compare agile method projects with those using traditional approaches, which one would expect when a new range of methods [...] claims to be superior” (Conboy, 2009, p. 331). Furthermore, decisions on project management methods lack a fact-based and comparable approach considering significant parameters and, therefore, bear the risk of incorrect decisions. After an initial literature review (including: Black, Boca, Bowen, Gorman, & Hinchey, 2009; Boehm & Turner, 2003; Charvat, 2003; Conboy, 2009; Fowler & Highsmith, 2001; Kerzner, 2009; Moss & Atre, 2003; Wysocki, 2011), there seems to be no approach of a quantitative decision model for the selection of the most appropriate project management method in a specific BI project setting. This leads us to following research question: “What would be an appropriate value-based decision model that supports decision makers in their selection of a specific project management method for a given BI project?”

This research-in-progress paper first identifies requirements for a practical decision model and provides necessary theoretical background. The subsequent section sketches the idea of a value-based decision model using a simplified example for demonstration. The model enables the determination of the most appropriate project management method in a particular BI project setting. The last section discusses limitations of the current model sketch and describes the further research agenda to enhance both maturity and generalizability of the decision model.

2 Theoretical Background

As indicated above, the decision on an appropriate project management method in a BI context lacks a comprehensible and practical decision model. From a scientific point of view, we expect a decision model to fulfill the following requirements: It has to (1) be applicable to a class of problems, (2) contribute new findings to the body of knowledge, (3) be comprehensible and reproducible, and (4) generate value for a user now or in future (Österle et al., 2010, p. 666). Furthermore, we additionally expect the model to (5) be economically feasible to ensure its applicability, i.e., the effort of application should be justified by its benefits, and (6) adopt a value-based approach using quantitative metrics as indicated in section 1 (Coenenberg et al., 2003, p. 3; Mertens, 1999, p. 11).

As object of research, we limit the considered project management methods in a first step to two extremes: a purely agile project management method (A) and a purely plan-driven project management method (P). Regarding the former, we need to distinguish agile project management methods from iterative and incremental approaches. While iterative development is “a rework scheduling strategy in which time is set aside to revise and improve parts of the system”, incremental development is “a staging and scheduling strategy in which various parts of the system are developed at different times or rates and integrated as they are completed” (Cockburn, 2008, p. 27). A does not necessarily include iterative or incremental elements, although changes during project runtime might suggest including one or both of the two approaches – corresponding with the principle to “value responding to change over following a plan” (Fowler & Highsmith, 2001, p. 29).

Both extremes, A and P, can be realized differently, for example with Scrum or Extreme Programming in case of A and CMMI (Capability Maturity Model Integration) or other plan-driven project management methods such as the waterfall method in case of P (Boehm & Turner, 2003, p. 14 ff.). Since we refer to the major characteristics of A and P, the key findings of this paper can be applied to any realized A or P. P promises planning reliability, strong documentation, comparability and repeatability (Boehm & Turner, 2003). However, the major shortcoming associated with P has been widely criticized: It is

inappropriate to handle rapidly changing environments and customer requirements (Abrahamsson, Conboy, & Wang, 2009, p. 281). Consequently, dynamism is the major risk to be addressed in projects applying P (Maruping, Venkatesh, & Agarwal, 2009, p. 377). On the other hand, A is characterized as lightweight processes with close customer collaboration and short iterative cycles resulting in independently running modules (Boehm & Turner, 2003, p. 17; Fowler & Highsmith, 2001). The trend to value working software over documentation, individuals and interactions over processes and tools and responding to change over following a plan bear the risk of delivering perfectly working individual modules which are not thoroughly working as an integrated system (Black et al., 2009, p. 39; Fowler & Highsmith, 2001, p. 29). Therefore, the risk of improper integration is the major risk to be addressed when applying A. In summary, environmental dynamism and improper system integration are the two major risks to be taken into account in the proposed basic decision model.

Literature shows that there are few researchers who compare projects applying one project management method or the other (Conboy, 2009, p. 331). Therefore, we seek for support on the decision between A and P in a specific project setting. We are aware that not all relevant project management methods can be classified into one of the two extremes. However, to identify key issues in a first step, we will not take into account those as well as the possibility of merging two project management methods – like already proposed, for example, in Boehm and Turner’s five step risk-based approach (Boehm & Turner, 2003). While their approach evaluates the project’s characteristics in detail, they do not take into account the project’s expected cash flows, and hence lack the desired quantitative nature.

Since there is no decision model that meets the aforementioned requirements, we suggest a value-based and future-oriented Net Present Value (NPV) model taking into account the project’s planned cash flows and integrating the risk of dynamism and the risk of improper system integration.

3 Proposition of a Decision Model and Demonstration Example

The example project has the objective to set up three reports (R1, R2, and R3) based on a data warehouse (DW) that integrates data from three operational databases (ODB1, ODB2, and ODB3) (see Figure 1). The project has to be finished within the next nine months.

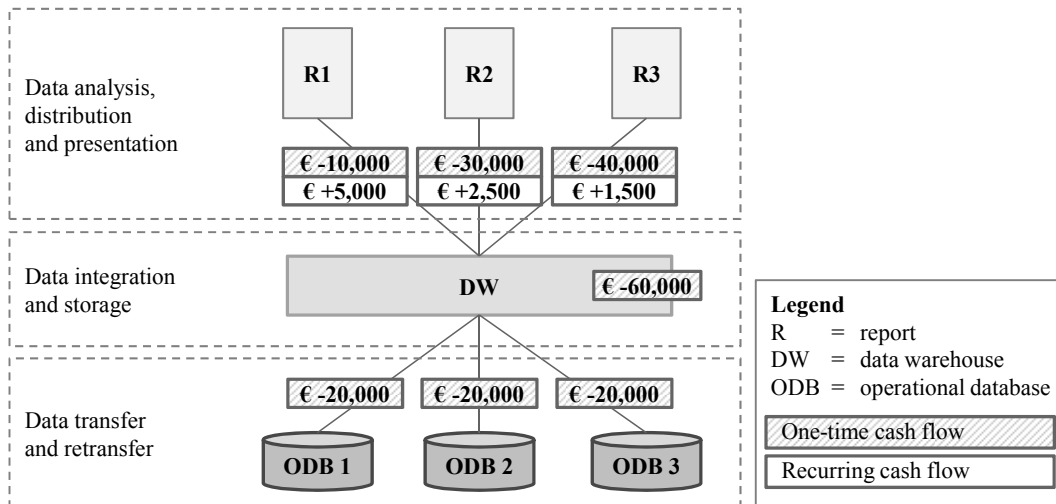


Figure 1: Target architecture including component-based cash flows

The three reports provide different value to the company. Their implementation costs depend on the reports' complexity. The most important report R1 fulfills new mandatory regulatory requirements and, by preventing sanctions, provides the highest value. It requires the integration of two operational databases (ODB1 and ODB2). R2, a better CRM (Customer Relationship Management) report, has the second priority and also requires ODB1 and ODB2. R3 shows the comparatively smallest value contribution. It offers more detailed analyses for controlling purposes and requires all three source systems to be available.

Costs and returns are supposed to be based on expert's estimation. For the simplified example we assume following one-time costs that are identical irrespective of the selected project management method: Developing R1 causes costs of €10,000, R2 €30,000, and R3 €40,000. Once the reports are in place, they generate recurring returns of €5,000 (R1), €2,500 (R2), and €1,500 (R3) in each period. A period thereby equals a quarter (three months). Integrating an ODB causes costs of €20,000 each. The development of the DW costs €60,000, which will be – in case of applying A – assigned mostly to the first period t_1 , taking into account that the data model has to be defined at the beginning. The costs of the remaining modules are assigned to the period when they are implemented and tested (see Figure 2). In case of applying P, we assume that the costs in period t_1 are slightly higher than in the following periods due to extensive project planning activities (Black et al., 2009, p. 42).

	Period t_1	Period t_2	Period t_3	Periods $t_4 \dots t_n$
Agile method A				
Benefits	€ ---	€ 5,000	€ 7,500	€ 9,000
Costs	€ 100,000	€ 40,000	€ 60,000	€ ---
Plan-driven method P				
Benefits	€ ---	€ ---	€ ---	€ 9,000
Costs	€ 80,000	€ 60,000	€ 60,000	€ ---
Legend Design Implementation Testing Productive System				

Figure 2: Comparison of benefit and cost cash flows for A and P along the project roadmap

The risk situation is characterized as follows. In case of a changing environment, we assume additional costs of €30,000 for A and €60,000 for P since the latter one is not able to adjust to changes as flexible. The likelihood for the occurrence is identical for both project management methods. In case of improper system integration, we assume additional costs of €180,000 whereby the likelihood of this peril is three times higher for A than for P.

The project roadmap has the milestones as depicted in Figure 2. Note that we only take into account three periods in the proposed model. We additionally show the final BI system (periods t_4 to t_n) for the sake of completeness. In case of applying A, in the first iteration the most important steps are taken, namely developing the major parts of the DW, integrating ODB1 and ODB2, and establishing R1. In the second iteration, the DW is completed and R2 is established. The missing ODB3 and R3 are integrated in the last iteration. In case of applying P, the first period is used to design the system architecture, the second period is used for the initial implementation of all system components, and in the last period the system is tested.

Formally, the decision model is based on following assumptions:

- The decision to implement a BI system is final. The company only chooses between two project management methods, agile (A) or plan-driven (P). Both are capable of delivering the same functionality within the same time (Black et al., 2009, p. 42). Due to its distinct strengths and weaknesses, each project manage-

ment method has a different cost/benefit structure that is captured in aggregated cash flows per period.

- All considered cash flows are discounted to the period the project starts.
- The project is affected by two types of risks only. These are the likelihood of environmental changes within a project period $\delta \in [0; 1]$ and the likelihood of a lack of integration at the end of the project $\varepsilon \in [0; 1]$. Both risks are independent.
- δ is constant for all periods and independent of the chosen project management method. ε is inherent to the chosen project management method and hence differs. δ occurs in each period, ε only occurs in the last one.
- If a risk occurs, the company will fix the failure. The costs associated with the correction are considered within the cash flow of the period the risk occurred (again discounted to the period of project start).
- The assumptions required for applying a NPV approach are fulfilled. That is, the company reinvests all cash flows in other value-contributing projects and acts within a perfect capital market.

Due to the risk structure of the project, we model the associated risks by a random experiment with three periods. Figure 3 shows an exemplary decision tree for A. In each of the three periods, two outcomes regarding the environmental risk are possible, “no change” with a likelihood of $1 - \delta$, and “change” with a likelihood of δ . Additionally, in the last period the risk of improper integration occurs with a likelihood of ε while “good” integration occurs with a likelihood of $1 - \varepsilon$. According to A5, we consider both risks by adding their expected costs to the cash flow of the period the risk occurs.

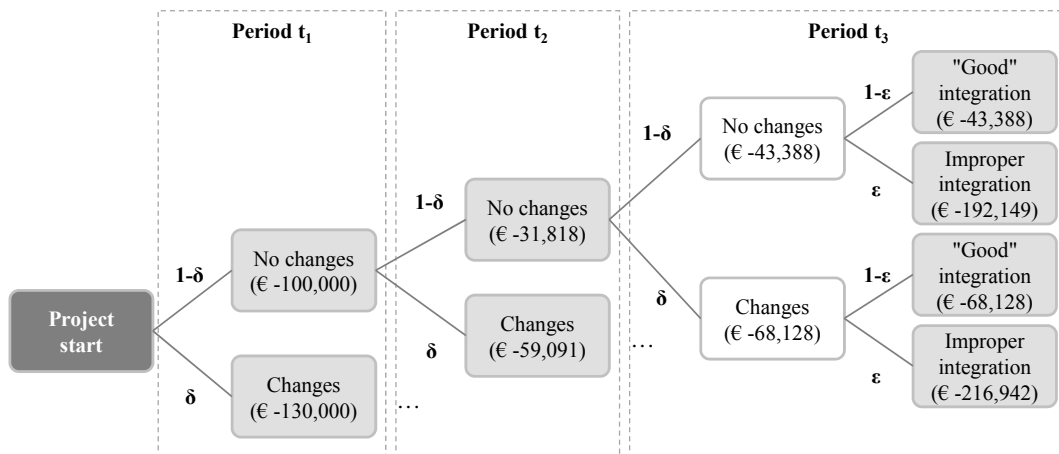


Figure 3: Decision tree showing the discounted cash flows for A

To calculate the NPV, the likelihoods have to be multiplied along all possible paths in the decision tree. The expected cash flow is this likelihood of a path multiplied by the sum of the corresponding cash flows along the path (already discounted based on an internal interest rate of 10%). The total NPV is the sum of these expected cash flows of all possible paths.

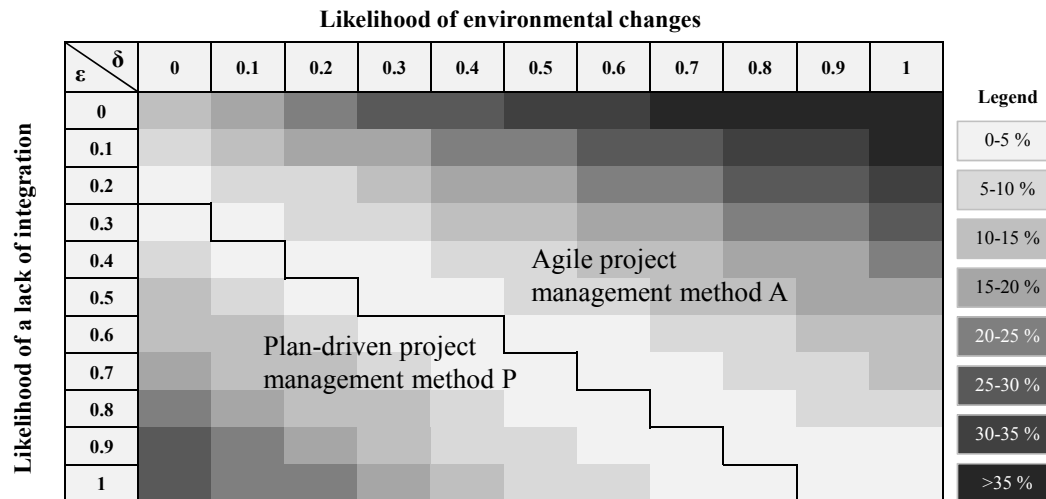


Figure 4: Result of the decision model applied to the demonstration example

According to the common NPV approach, one compares the NPVs of two alternatives and chooses the higher one – given it is positive (Ross, 1995, p. 96). In our demonstration example, the resulting NPV is expected to be negative since we only take into account the cash flows during project runtime but do not consider returns that are generated after the end of the project. This is possible since A1 ensures that the project is economically feasible in the long term. Furthermore, we extend this NPV approach by a comparison of the NPVs of both project management methods. We simulate all combinations of δ and ϵ in discrete steps of 0.1 in order to identify the risk parameter combinations that mark the turning point in the decision. Figure 4 visualizes the resulting recommendations for any parameter combination.

We see that the model's recommendation is not contrary to an intuitive estimation. For highly agile environments with $\delta \geq 0.8$, A is superior regardless of the value of ϵ . A is also superior in case of low integration risks with $\epsilon < 0.3$. For $\epsilon \geq 0.6$ and $\delta \leq 0.5$, P is the most appropriate project management method. For all other parameter combinations, the turning point is located in an area of indifference between both project management methods. The shading of Figure 4 indicates the deviation of the resulting NPV in percent if applying the less appropriate project management method instead of the recommended one. While the decision's turning point is located in a corridor of marginal differences, it reaches up to 40% in the boundary area. This result shows that applying project manage-

ment methods inconsiderately might result in significant discrepancies in the resulting NPVs. Even though the model does not yet offer surprising findings concerning a recommendation, we appreciate that its results conform to intuitive decisions in this simple demonstration example. This indicates that a more advanced model might give useful recommendations in more complex decision situations when an intuitive decision is no longer reliable.

The proposed decision model requires a sufficient clarity and preciseness of parameter values that might demand great efforts to be provided. To analyze the necessary degree of data accuracy, future research will include sensitivity analyses addressing the parameters of the model. These sensitivity analyses should further strengthen the fact base on which decision makers can derive a recommendation for the appropriate project management method – instead of relying on “one-size-fits-all” instructions or gut feeling. In this respect, this paper already provides a rough model sketch and proves the principle feasibility of an approach based on estimating two risk parameters and a project’s cash flows.

4 Limitations and Outlook

The presented model fulfills or is able to fulfill all six initially stated requirements mentioned in section 2. (1) It is applicable to a class of problems, namely to decisions on the project management method in a BI project. (2) It contributes to the body of knowledge since no formal quantitative model on the decision between agile and plan-driven project management methods could be found in literature. (3) The underlying NPV approach is established and results are easily reproducible. (4) The model claims to add value to the decision process by supporting decision makers in the selection of the most appropriate project management method. (5) Since estimating the project’s cash flows is necessary for project planning anyway, the model is easily applicable. Decision makers have to only estimate two additional risk parameters. (6) Relying on a NPV-based decision model, we propose the way towards a quantitative, value-based and future-oriented approach.

Nevertheless, the proposed model is beset with limitations that need to be taken into account when applying it in project settings.

- The proposed model is only applied to a three period case study.
- The proposed model only takes into account two decisive parameters.
- The peril of improper integration is assumed to occur only in the last period, while other arguments suggest this risk could be prevented earlier.

- The proposed model only takes into account a “pure-blood” agile and plan-driven approach and neglects other project management methods and hybrids of both extremes.
- The proposed model claims to offer a clear recommendation for a decision, yet might underlie errors regarding the estimation of cash flows and risk parameters.

Despite the named limitations, the proposed NPV model represents an initial decision model comparing the impact of selected project management methods in a BI context. Based on the presented model sketch, we currently work on a more detailed approach.

We will address L1 by the development of a general analytical model that can be applied to any number of project periods. The limitation to two parameters (L2) might be considered a far reaching simplification. Therefore, based on a more detailed analysis it might be an interesting option to add more risk parameters as, for example, the five parameters identified by Boehm and Turner (2003). Referring to L3, we will consider the possibility that decision makers detect an integration problem not at the end, but earlier during the project. Thus, an extended model might consider both risks in each period. Additionally, further research should address L4 by proposing a general analytical model that can be applied to any project management method. To strengthen the reliability of the decision model and to examine the effects of estimation errors, we plan to address L5 by conducting sensitivity analyses to test the impact of deviations of estimated cash flows and risk parameters. Additionally, it might be interesting to evaluate the model using case studies from different industries in order to identify if – and in which aspects – the model needs to be adapted. Summarizing, since there seems to be a lack of theoretical foundation for a decision on the appropriate project management method in a BI project, we presented a research-in-progress model already giving first recommendations and an outlook on future research.

References

- Abrahamsson, P., Conboy, K., & Wang, X. (2009). ‘Lots done, more to do’: the current state of agile systems development research. *European Journal of Information Systems*, 18(4), 281-284.
- Adamala, S., & Cidrin, L. (2011). Key Success Factors in Business Intelligence. *Journal of Intelligence Studies in Business*, 1(1), 107-127.
- Black, S., Boca, P. P., Bowen, J. P., Gorman, J., & Hinchey, M. (2009). Formal versus Agile: Survival of the Fittest. *IEEE Computer*, 42(9), 37-45.

- Boehm, B., & Turner, R. (2003). *Balancing Agility and Discipline: A Guide for the Perplexed* Addison-Wesley Longman Publishing Co., Inc.
- Charvat, J. (2003). *Project Management Methodologies: Selecting, Implementing, and Supporting Methodologies and Processes for Projects* John Wiley & Sons Inc.
- Cockburn, A. (2008). Using Both Incremental and Iterative Development. *STSC CrossTalk*, 21(5)
- Coenberg, A. G., Mattner, G. R., & Schultze, W. (2003). Wertorientierte Steuerung: Anforderungen, Konzepte, Anwendungsprobleme. In A. Rathgeber, H. J. Tebroke & M. Wallmeier (Eds.), *Finanzwirtschaft, Kapitalmarkt und Banken, Festschrift für Professor Dr. Manfred Steiner* (1st ed., pp. 1-24). Stuttgart: Schäffer-Poeschel.
- Conboy, K. (2009). Agility from first principles: reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329-354.
- Fowler, M., & Highsmith, J. (2001). The Agile Manifesto. *Software Development*, 9(8), 28-35.
- Höhn, R., & Höppner, S. (2008). *Das V-Modell XT: Grundlagen, Methodik und Anwendungen* (1st ed.) Springer Publishing Company, Incorporated.
- Kerzner, H. (2009). *Project Management: a Systems Approach to Planning, Scheduling, and Controlling* (10th ed.) John Wiley & Sons.
- Larman, C., & Basili, V. R. (2003). Iterative and Incremental Development: A Brief History. *IEEE Computer*, 36(6), 47-56.
- Maruping, L. M., Venkatesh, V., & Agarwal, R. (2009). A Control Theory Perspective on Agile Methodology Use and Changing User Requirements. *Information Systems Research*, 20(3), 377-399.
- Mertens, P. (1999). Operiert die Wirtschaftsinformatik mit den falschen Unternehmenszielen?-15 Thesen. In J. Becker, W. König, R. Schütte, O. Wendt & S. Zelewski (Eds.), *Wirtschaftsinformatik und Wissenschaftstheorie: Bestandsaufnahme und Perspektiven* (pp. 379-392) Gabler Verlag.
- Moss, L. (2009). Beware of Scrum Fanatics On DW/BI Projects. *EIMI*, 3(3)
- Moss, L., & Atre, S. (2003). *Business Intelligence Roadmap: The Complete Project Lifecycle for Decision-Support Applications* Addison-Wesley Professional.
- Österle, H., Becker, J., Frank, U., Hess, T., Karagiannis, D., Krcmar, H., et al. (2010). Memorandum zur gestaltungsorientierten Wirtschaftsinformatik. *Schmalenbachs Zeitschrift Für Betriebswirtschaftliche Forschung*, 62(9), 662-672.

Project Management Institute. (2008). A Guide to the Project Management Body of Knowledge (4th ed.) Project Management Institute.

Project Management Methodologies. (n.d.). Retrieved July 15, 2012, from <http://infolific.com/technology/methodologies/>

Ross, S. A. (1995). Uses, Abuses, and Alternatives to the Net-Present-Value Rule. *Financial Management*, 24(3), 96-102.

Wysocki, R. K. (2011). *Effective Project Management: Traditional, Agile, Extreme* (6th ed.) John Wiley & Sons.