

Business-Intelligence-Konzept auf Basis einer Event-Driven Service-Oriented Architecture

Tim Vogt, Sebastian Neuhaus, Markus Linden, Peter Chamoni

Fachbereich Betriebswirtschaft – Mercator School of Management

Lehrstuhl für Wirtschaftsinformatik und Operations Research

Universität Duisburg-Essen, Campus Duisburg

Lotharstraße 65, 47057 Duisburg

{tim.vogt | sebastian.neuhaus | markus.linden | peter.chamoni}@uni-due.de

Abstract: In Zeiten eines dynamischen Wettbewerbs ist eine schnelle Entscheidungsfindung erforderlich, die im Rahmen klassischer Data-Warehouse(DWH)-Systeme, als technische Umsetzung von Business Intelligence (BI), nur eingeschränkt ermöglicht wird [FB06, 54]. DWH-Ansätze, wie das Real-Time Data Warehousing, sollen den Anforderungen von Echtzeitunternehmen gerecht werden. Das Real-Time Data Warehousing impliziert eine schnellere Informationsversorgung als der klassische Ansatz. Neben dem Aspekt einer schnelleren Informationsversorgung gewinnt der Gedanke der aktiven Unterstützung der Endbenutzer im Prozess der Entscheidungsfindung auf operativer und strategischer Ebene an Bedeutung. Ein aktives analyseorientiertes Informationssystem, im Sinne des Active Data Warehousing, ermöglicht eine automatisierte Benachrichtigung der Endbenutzer bei wiederkehrenden Ereignissen sowie eine automatisierte Durchführung von Entscheidungen [Sc06, 425f.]. Ziel dieses Beitrags ist die Entwicklung eines BI-Konzepts zur Erfüllung der Anforderungen von Echtzeitunternehmen und zur Umsetzung von Operational Business Intelligence. Die Entwicklung des Konzepts erfolgt dabei auf Grundlage einer Event-Driven Service-Oriented Architecture und wird sukzessiv anhand einer Ebenenstruktur vollzogen.

1 Erweiterungspotenzial von Data-Warehouse-Ansätzen zur Umsetzung von Operational Business Intelligence

Klassische DWH-Systeme als technische Umsetzung von Business Intelligence sind Lösungen, die insbesondere auf die strategische Entscheidungsfindung durch eine eingeschränkte Zielgruppe von Fach- und Führungskräften ausgerichtet sind. Auf Grundlage von überwiegend retrospektiven Analysen mittels DWH-Systemen werden strategische Unternehmensentscheidungen von den Führungskräften gefällt. Eine Unterstützung für hierarchisch niedrige, operative Ebenen im Sinne von Operational Business Intelligence ist bei den klassischen DWH-Ansätzen nicht vorgesehen [Ec07, 4f.]. Jedoch besitzt auch diese Unterstützung hinsichtlich einer Integration des Right-Time-Ansatzes hohe Relevanz, da operative Entscheidungen situationsadäquat beim Auftreten von Ereignissen getroffen werden müssen und eine Reaktion in Echtzeit ermöglicht werden sollte. Eine Verlagerung von Entscheidungen auf eine niedrigere Management-Ebene geht mit einer

Verbesserung der Qualität und der Geschwindigkeit der Entscheidungsfindung einher [CK06, 1f.; Ve07, 5]. Zudem ermöglicht die Integration einer aktiven Komponente den Endanwendern auf operativer Ebene automatisiert auf unternehmenskritische Situationen hinzuweisen oder bei einer vollständigen Integration der Geschäftsprozesse Entscheidungen autark durchzuführen (Active Data Warehousing¹) [CK06, 1f.]. In diesem Kontext wird deutlich, dass eine Anpassung der analytischen Informationssysteme auf die Bedürfnisse einer Vielzahl von Nutzern auf hierarchisch niedrigeren Ebenen notwendig ist.

Mit Hilfe eines BI-Konzepts soll den oben beschriebenen Anforderungen Rechnung getragen werden. Auf Grundlage der Architekturkonzepte der Service-Oriented Architecture (SOA) und Event-Driven Architecture (EDA) soll eine Verschmelzung der operativen und analytischen Informationssysteme erreicht werden. Dazu wird die Anordnung der essentiellen DWH-Komponenten aufgelöst und in ein generisches SOA-Modell überführt.

2 Event-Driven Service-Oriented Architecture

Eine Umsetzung der Anforderungen kann durch den Einsatz einer ereignisgesteuerten, serviceorientierten Architektur (Event-Driven Service-Oriented Architecture, EDSOA) erfolgen. Eine solche Architektur stellt Services zur Verfügung, die durch Ereignisse ausgelöst werden können. Separat betrachtet, stellen die SOA und die EDA zwei orthogonale Ansätze dar. Demnach kann eine SOA unabhängig von einer EDA implementiert werden und vice versa. Erst die Kombination beider Ansätze kann das volle Potenzial der Architekturen entfalten [Za07, 66]. Eine ereignisgesteuerte, serviceorientierte Architektur beseitigt die Schwächen der jeweiligen Architekturen und bietet den effektivsten Ansatz für eine agile IT-Infrastruktur sowie zur Lösung von Integrationsproblemen [SR05, 18].

Eine SOA ebnet einen Übergang von zuvor monolithischen und zentralisierten Systemen hin zu modularen und lose gekoppelten Informationssystemen. Eine EDA stellt eine sinnvolle Ergänzung zur Abbildung von Geschäftsereignissen und -prozessen in Echtzeit dar. Auf diese Weise können notwendige Entscheidungen in kürzerer Zeit getroffen und Verbesserungspotenziale ausgeschöpft werden. Die EDSOA bildet somit die Grundlage für ein erfolgreiches Geschäftsprozessmanagement und ist unabdingbar, um den Echtzeit-Gedanken zu realisieren [Za07, 69].

Im Kontext dieses Beitrags wird unter einer EDSOA eine SOA auf Basis eines Enterprise Service Bus (ESB) verstanden, die durch die EDA nicht-invasiv erweitert wird. Die Geschäftslogik liegt in Form von Services vor, wobei die initiiierenden und resultierenden Ereignisse der Prozesskette EDA-Ereignisse darstellen. Zur Realisierung der Ereignisverarbeitung liefert der ESB die entsprechenden Funktionalitäten [Za07, 67]. Der ESB einer EDSOA unterstützt als zentrale Kommunikationsplattform die ereignisge-

¹ Ausführliche Erläuterungen zum Active Data Warehousing finden sich in [Th01].

steuerte, asynchrone Interaktion zwischen den Services. Insbesondere verfolgt der ESB den Zweck einer garantierten und sicheren Übertragung von Nachrichten, der Bereitstellung von Adaptern zur Integration unterschiedlicher Komponenten sowie der Bereitstellung von Routing-Mechanismen zur Meldungsverteilung mittels einer umfangreichen Sammlung von Instrumenten [Li07, 136; Wh06, 1]. In einer EDSOA auf der Grundlage eines ESB stellen Anwendungen und Services abstrakte Endpunkte dar, die auf Ereignisse reagieren können.

Gemäß den vorangestellten Ausführungen können EDA und SOA als ein integratives, sich ergänzendes Konzept implementiert werden. Eine Kombination dieser Konzepte zu einer EDSOA auf Basis eines ESB steigert den Wert beider Architekturkonzepte [Wo06, 3]. Der wesentliche Nutzen einer EDSOA besteht darin, dass alle Geschäftsereignisse auf dem Event Bus vorliegen und im Rahmen von EDA-Prozessen, wie bspw. Complex Event Processing (CEP)², in Echtzeit verwendet werden können. Auf diesem Wege lässt sich eine vollständige Transparenz der Prozesse herstellen. Eine Reaktion auf eintretende Ereignisse kann unmittelbar erfolgen. Darüber hinaus wird durch die Möglichkeit, mehrere Services parallel und asynchron anzustoßen, der synchrone und sequentielle SOA-Kontrollfluss deutlicher und realitätsnaher abgebildet [Za07, 68].

3 BI-Konzept auf Grundlage einer EDSOA

An das im weiteren Verlauf vorgestellte Business-Intelligence-Konzept auf Grundlage einer EDSOA werden andere Anforderungen gestellt als an die klassischen DWH-Ansätze. Es müssen nicht nur eine größere Anzahl verschiedener Endnutzer mit analytischen Informationen versorgt, sondern ebenfalls ein höheres Datenvolumen in Real-Time/Right-Time verarbeitet und übermittelt werden. Dabei darf die Performanz der Quell- und Zielsysteme nicht eingeschränkt und die Transaktionen der operativen Informationssysteme nicht negativ beeinflusst werden. Durch den in diesem Kontext zunehmend operativen Charakter der analytischen Informationssysteme sind darüber hinaus eine hohe Ausfallsicherheit sowie geringe Wiederherstellungszeitfenster unumgänglich. Zudem gilt es, die Herausforderung zu meistern, bei bspw. komplexen multidimensionalen Abfragen Analysewerte in Real-Time/Right-Time zurückzuliefern. Letztlich sind solche Abfragen dynamisch zu formulieren, um als Services wieder verwendet werden zu können [Ec07, 19].

Die Unterteilung der traditionellen Struktur eines DWH-Systems in die Schichten Datenerfassung, Datenhaltung, Datenbereitstellung/-analyse sowie Präsentation kann im Weiteren nicht beibehalten werden. Im Kontext dieser Erweiterung können die Rollen einzelner Komponenten und somit die jeweilige Zugehörigkeit zu den Schichten nicht Bestand haben, da sich der Service- und Ereignisgedanke im Hinblick auf die Geschäftsprozessorientierung nur unzureichend in dieser Form der Architektur darstellen lässt. Infolgedessen dient als Grundlage ein generisches SOA-Architekturmodell³. In diesem

² Eine ausführliche Erläuterung des Complex Event Processing findet sich in [Lu02].

³ In Anlehnung an [Li07, 27].

Zusammenhang werden Komponenten der verschiedenen DWH-Ansätze in das SOA-Architekturmodell überführt. Anschließend werden Erläuterungen für das Zusammenspiel der einzelnen Komponenten sowie eine Darstellung möglicher Service-Typen angeführt. Abbildung 1 stellt das Business-Intelligence-Konzept auf Grundlage einer ED-SOA dar.

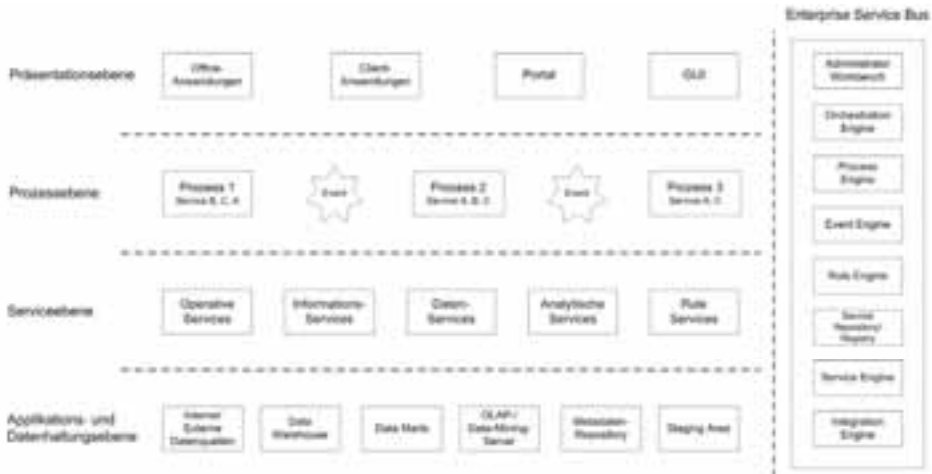


Abbildung 1: Business-Intelligence-Konzept auf Grundlage einer Event-Driven Service-Oriented Architecture

3.1 Applikations- und Datenhaltungsebene

Aufgrund der unterschiedlichen Gestaltungsziele eines Data Warehouse und einer EDSOA scheint eine Zusammenführung beider Konzepte widersprüchlich. Ein DWH dient als dauerhafter Datenspeicher, um historische Daten für Analysezwecke bereitzustellen. Zudem ist der Einsatz eines DWH auf langfristige Stabilität ausgerichtet. Mit einer EDSOA hingegen sollen Änderungen innerhalb der operativen Integration erleichtert und schneller nachvollziehbar gemacht und somit die Agilität gesteigert werden. Dieser mögliche Zielkonflikt ist insbesondere von der Beziehung zwischen operativen und analyseorientierten Informationssystemen abhängig. Die operativen Informationssysteme dienen einerseits als Datenbasis für das DWH, andererseits können die analytischen Informationssysteme aber auch als Datenquelle für die operativen Informationssysteme verstanden werden und müssen naturgemäß höheren Anforderungen gerecht werden [Sc07, 151].

Im Vergleich zu den klassischen DWH-Ansätzen ist zu erkennen, dass das DWH den Status als zentraler Kern verliert und als eine von mehreren Komponenten auf der Applikations- und Datenhaltungsebene verstanden wird. Das DWH kann somit im Rahmen der Serviceorientierung sowohl als Service-Nachfrager als auch als Service-Anbieter auftreten. Eine solche vollständige Integration ist bislang nur selten in der Praxis implementiert. Zumeist findet eine sukzessive Integration zunächst als Service-Nachfrager statt [Di07, 136f.]. Durch die Implementierung des DWH als Service-Nachfrager in eine

EDSOA werden mithilfe der zentralen ESB-Infrastruktur und den darin enthaltenen Extraktion-, Transformation-, Laden-Komponenten (ETL-Komponenten) die benötigten Daten aus den internen und externen Datenquellen in das DWH überführt. Abhängig vom Verwendungsszenario der analytischen Abfragen und der damit einhergehenden tolerierbaren Latenzzeiten kann bspw. das Konzept des Change-Data-Capture-Verfahrens (CDC-Verfahren) zur Performanzoptimierung und somit zur schnelleren Bereitstellung der Daten implementiert werden. Die benötigten Metadaten und Änderungsinformationen, um Deltas zu laden, werden mithilfe des Metadaten-Repository, das eine weitere Komponente dieser Schicht darstellt, über den ESB zur Verfügung gestellt.

Die konsequente Weiterentwicklung besteht darin, das DWH mit den aufsetzenden Analysewerkzeugen als Service-Anbieter im Sinne des Closed-Loop-Ansatzes über den ESB verfügbar zu machen und die internen und externen Datenquellen mit Analyseinformationen anzureichern. Darüber hinaus können analytische Informationen in den Ablauf von Geschäftsprozessen im Sinne von Operational Business Intelligence eingebettet werden [Di07, 137; Sc07, 152f.]. Einen ersten Schritt zur Minimierung der Latenz und somit zu einer schnelleren Bereitstellung von Analysedaten stellt die Anbindung von spezifischen Data Marts an den ESB dar, die bereits themenbezogene Daten für repetitive Abfragen vorhalten und über Services dem ESB unmittelbar zur Verfügung stehen.

Die OLAP- und Data-Mining-Server, die im klassischen DWH-System ursprünglich der Schicht der Datenbereitstellung/-analyse zugeordnet werden, stellen in diesem Konzept weitere Komponenten der Applikations- und Datenhaltungsebene dar. Ein erster Ansatz zur Zusammenführung von operativen und analytischen Daten kann über die Implementierung einer Staging Area erfolgen, um Prozesse in Echtzeit mit Analytik anzureichern. Eine temporäre Staging Area stellt eine hoch performante Datenbank zur kurzfristigen Zwischenspeicherung von operativen und analytischen Daten dar. Mithilfe von Metadaten lassen sich die eingehenden Datensätze zudem anreichern [JS07, 31].

Zusammenfassend kann konstatiert werden, dass die Applikations- und Datenhaltungsebene hierarchisch gleichwertige Anwendungen und Datenbanken umfasst, die in einer EDSOA als Backend-Systeme dienen und ihre Funktionalität den Services über den ESB zur Verfügung stellen.

3.2 Enterprise Service Bus

Der Enterprise Service Bus stellt das zentrale verbindende Element zwischen den Komponenten der verschiedenen Ebenen dar. Der ESB regelt und überwacht die Kommunikation der Services und Ereignisse. Dabei unterstützt der ESB die Datenbewirtschaftung der Systeme untereinander als asynchrone Publish-and-Subscribe-Kommunikation mithilfe der EDA-Komponente sowie als synchrone Kommunikationsform für Service-Nutzer [Di07, 138]. Aufgrund der zentralen Bedeutung zeigt Abbildung 2 den ESB im Detail.

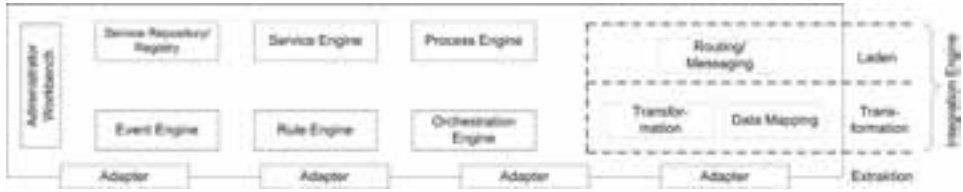


Abbildung 2: Enterprise Service Bus

Eine Interaktion der Informationssysteme und Datenbanken der Applikations- und Datenhaltungsebene untereinander ist prinzipiell multidirektional und ausschließlich über den ESB möglich.

Im Kontext von Operational Business Intelligence sind insbesondere Lösungen gefordert, die die hohen Anforderungen des Right-Time-Ansatzes erfüllen. Daher ist eine unmittelbare Datenversorgung aus den operativen Quellsystemen auf dem ESB obligatorisch, um auf Diskrepanzen im Prozessablauf sofort reagieren zu können. Damit zusätzlich analytische Abfragen für eine proaktive Geschäftsprozessüberwachung rechtzeitig verfügbar sind, müssen die operativen Daten echtzeitnah in das DWH geladen und anschließend analysiert werden. Der klassische ETL-Prozess, abhängig vom Real-Time-Anspruch, kann dahingehend als kritischer Bereich gewertet werden. Zur Realisierung müssen zunächst die ETL-Prozesse der Integration Engine im ESB schlanker formuliert werden und eine Abwendung der Verarbeitung von Massendaten zur Einzelsatzverarbeitung erfolgen, um einen kontinuierlichen Verarbeitungsstrom zu gewährleisten [Sc07a, 1]. Der beschriebene kontinuierliche ETL-Prozess der Integration Engine, als wesentliche Komponente zur Real-Time-Verarbeitung, umfasst innerhalb des ESB Adapter als Extraktionskomponente, Elemente der Transformation und des Data Mapping als Transformationskomponente sowie Routing und Messaging als Ladekomponente [JS07, 31].

Ein kontinuierlicher Verarbeitungsstrom kann mit Techniken der Datenpropagierung umgesetzt werden. In diesem Kontext ist das Verfahren der Continuous Data Integration⁴ zu favorisieren, da es zur Real-Time-Verarbeitung ereignisgesteuerte Mechanismen verwendet, die über den ESB getriggert werden können. Dadurch vollzieht sich ein Wandel von einem Pull- zu einem Push-Mechanismus. Auf diese Weise wird eine kontinuierliche und asynchrone Datenbereitstellung möglich. Die operativen Quellsysteme melden Veränderungen in den Datenbeständen als Nachricht an eine Data Change Engine, die die Änderungen sowie die geänderten Daten in eine Message Queue schreibt. Die Data Change Engine sowie die Message Queue sind in der Architektur als Routing/Messaging zusammengefasst. Mithilfe des Publish-and-Subscribe-Verfahrens der EDSOA kann die Datenintegrationsanwendung spezifische Änderungen abonnieren, diese sukzessiv verarbeiten und somit die Daten kontinuierlich in das DWH laden [LH06, 11].

Neben den zentralen ETL-Funktionen umfasst der ESB eine Orchestration, Process, Service, Rule und Event Engine. Über die Orchestration Engine werden die unterschied-

⁴ Eine ausführliche Erläuterung des Continuous-Data-Integration-Verfahrens findet sich in [TT03].

lichen Service-Abfolgen definiert, die auf der Prozessebene modelliert werden, um bspw. analytische Fragestellungen zu bearbeiten. Die eng mit der Orchestration Engine zusammenarbeitende Process Engine beinhaltet die entsprechenden Mechanismen zur Steuerung und Ausführung dieser Prozesse. Dabei ist die Process Engine insbesondere für die korrekte Abwicklung der Prozesse verantwortlich. Die Service Engine stellt Mechanismen zur Verfügung, damit Services entwickelt und integriert werden können. Die Rule Engine umfasst definierte ECA-Regeln (Event, Condition, Action) sowie analytische Entscheidungsregeln. Somit können einzelne Aktionen wie bspw. OLAP-Analysen durch definierte Szenarien automatisiert ausgeführt werden, ohne dass der Endanwender aktiv werden muss. Solche Regeln sind nicht auf Geschehnisse außerhalb des DWH-Systems anwendbar. Diesbezüglich kann die Event Engine Abhilfe schaffen. Neben den angeführten Entscheidungsregeln unterstützt der ESB mithilfe der Event Engine die Ereigniserkennung, Analyse und Verarbeitung auf dem ESB. Diese Ereignisse können unterschiedliche Services anstoßen. Die Integration von Ereignisverarbeitungsmechanismen, wie das Complex Event Processing, ermöglichen nicht nur auf Ereignisse in Echtzeit zu reagieren, sondern auch prädictiv Problemszenarien zu identifizieren und somit frühzeitig agieren zu können.

Administration- und Monitoring-Funktionalitäten zur Konfiguration aller Aktivitäten auf dem ESB sind durch die zentrale Schnittstelle Administrator Workbench implementiert. Weiterhin ist das Service Repository/Registry zur Verwaltung und Veröffentlichung der Services notwendig. Der ESB übernimmt demzufolge ebenfalls die Aufgabe eines Service-Vermittlers.

3.3 Serviceebene

Die Serviceebene umfasst unterschiedliche Service-Typen. Die Services kapseln die Funktionalitäten aus den heterogenen Systemen der Applikations- und Datenebene nach den Anforderungen der systemübergreifenden Prozesse. Die wiederverwendbaren Funktionalitäten werden über lose gekoppelte Services verfügbar gemacht. Wie zuvor beschrieben, fungiert der ESB als Service-Vermittler und veröffentlicht die definierten Services, sodass diese aufgefunden werden können. Durch die Serviceebene und die nachfolgend dargestellte Prozessebene erfolgt eine Trennung von Geschäftslogik und Prozesslogik sowie darüber hinaus von fachlichen Konzepten und der technischen Realisierung [Di07, 136]. Im Kontext klassischer DWH-Systeme geht damit ein Entwicklungsprozess von analytischen Applikationen zu analytischen Services einher. Der Serviceebene sind operative Services, Daten-Services, Informations-Services, analytische Services sowie Rule Services zugeordnet.

Operative Services können als Oberbegriff für Services verstanden werden, die eine transaktionsbezogene Geschäftslogik darstellen. Anzuführen sind bspw. das Anlegen eines neuen Kunden oder die Stornierung eines Auftrags. Daten-Services sind durch einen direkten Datenzugriff auf die Datenbestände des DWH, der Data Marts und der operativen Informationssysteme gekennzeichnet. Analytische Services können gegenüber Daten-Services umfangreiche Informationen des DWH-Systems nutzen und zu Analysen auf aggregierter Ebene verwenden. Anzuführen ist die Versorgung mit Analysen

auf dem vom Service-Nachfrager gewünschten Aggregationsniveau. Zu den analytischen Services zählen auch Analysen, die auf Real-Time-Daten angewendet und z.B. zur Prozesssteuerung eingesetzt werden [Di07, 137].

Aufgrund der zentralen Bedeutung von analytischen Services im vorgestellten BI-Konzept werden nachfolgend mögliche Ausprägungen analytischer Services dargestellt. Diese Services repräsentieren die klassischen BI-Werkzeuge, wie OLAP oder Data Mining. Eine Unterteilung von analytischen Services kann bspw. in Berichts-, Abfrage- und Analyse-Services vorgenommen werden, die jeden Daten- und Informations-Service zur Datenlieferung nutzen können und somit nicht mehr ausschließlich auf Daten aus dem DWH angewiesen sind. Weitere mögliche Service-Ausprägungen stellen Planungs- und Simulations-Services dar. Durch die Einbindung dieser Services können die Auswirkungen von einzelnen Entscheidungen simuliert und geplant werden. Somit können bspw. Individualentscheidungen geprüft und an einem übergeordneten Geschäftsziel ausgerichtet werden. Neben dieser Service-Gruppe umfassen analytische Services zudem Alerting-Services. Alerting-Services sind insbesondere im Zusammenhang mit der Ereignisorientierung unerlässlich. Bei Eintritt eines Ereignisses werden entsprechende Informationen durch das alarmauslösende Ereignis an alle Empfänger distribuiert, die diese Informationen abonniert haben, um unmittelbar Aktionen auszulösen. Darüber hinaus zählen Dashboard-Services zu den analytischen Services. Solche Services stellen Präsentationsfunktionalitäten dar, die entsprechend dem Informationsprofil des Endanwenders, z.B. als Portlet, in eine Portallösung implementiert werden. Mithilfe dieser Services werden bspw. Kennzahlen der Berichts-, Abfrage- und Analyse-Services visualisiert und personalisiert verfügbar gemacht. Letztlich umfassen die analytischen Services Key Performance Metrics (KPM) Services. Diese Services implementieren Kennzahlen, um die Leistung von Prozessen zu verwalten und zu steuern [Ma06, 16ff.].

Neben den dargelegten Daten-Services existieren sog. Informations-Services. Diese Services kombinieren neben strukturierten und unstrukturierten Daten auch operative und analytische Daten. Darüber hinaus enthält die Kategorie der Informations-Services Metadaten- und Stammdaten-Services. Die letzte Service-Kategorie in diesem Kontext bilden die Rule Services. Durch den Einsatz dieser Services werden komplexe Regeln für die Entscheidungslogik gekapselt. Auf diese Weise wird erstmals die Entscheidungslogik eine Teilmenge der Geschäftslogik. Da ein Prozess auf verschiedenen Regeln basiert und Regeln typischerweise in mehreren Prozessen verwendet werden, bietet sich eine Implementierung solcher Services im Sinne von Operational Business Intelligence an. Eine gezielte Automatisierung von menschlichen Entscheidungen kann mittels der Integration einer Entscheidungslogik via Services in Prozessen erzielt werden [Ma07, 11].

3.4 Prozessebene

Auf der Prozessebene werden die verschiedenen Geschäftsprozesse und Geschäftsregeln modelliert. Die Geschäftsfunktionalität wird unter Zuhilfenahme von Services aus den Anwendungen der Applikations- und Datenhaltungsebene nutzbar gemacht und entsprechend der Prozesslogik orchestriert. Dazu können die einzelnen Services flexibel und

dynamisch in die Prozesse eingebunden werden. Die Prozesse sind infolgedessen von den Anwendungen der Applikations- und Datenhaltungsebene unabhängig. Dementsprechend kann auf Veränderungen in den Geschäftsprozessen schnell und flexibel durch eine neue Kombination der Services reagiert werden. Durch die Orchestrierung können auch operative Geschäftsprozesse mit analytischen Informationen angereichert werden. Somit sind Geschäftsprozesse durch die Implementierung von analytischen Services plan-, überwacht- und steuerbar. Die dadurch entstandene Prozessorientierung von Business Intelligence führt zu einer Einbeziehung der an den operativen Geschäftsprozessen beteiligten Fachabteilungen und infolgedessen zu einer Ausweitung des Nutzerkreises (Operational BI) [Ma06a, 6ff.]. Darüber hinaus können bspw. Rule Services zur Automatisierung bestimmter Aktionen und unter Berücksichtigung definierter Prämissen in die Geschäftsprozesse integriert werden.

Das beschriebene Szenario berücksichtigt bisweilen keine ereignisgetriebenen Aktivitäten. Ereignisse stellen in einer EDSOA jedoch fundamentale Komponenten der Prozesskette und dementsprechend der Geschäftslogik dar. Die EDA-Komponenten können in einer SOA Szenarien erkennen sowie Ereignisse generieren und versenden. Die Abonnenten können diese entgegennehmen, auswerten und geeignete Maßnahmen parallel und asynchron in Echtzeit initiieren. Infolgedessen haben Ereignisse auslösenden und resultierenden Charakter in der Prozesskette und stoßen Services an. Services können somit nicht nur von anderen Services angestoßen werden, sondern ebenfalls von Ereignissen [Za07, 61ff.].

3.5 Präsentationsebene

Die Präsentationsebene umfasst unterschiedliche Alternativen zur Visualisierung. Durch die Verbindung von analytischen und operativen Informationssystemen in dem vorgestellten BI-Konzept sind nicht mehr nur ausschließlich die dedizierten BI-Werkzeuge für Präsentationsfunktionalitäten nutzbar. Es zeichnet sich stattdessen eine Flexibilisierung bei der Wahl der Front-End-Anwendungen ab [Ph06]. Die Darstellung der Analyseinformationen erfolgt neben Office- und Client-Anwendungen durch Portal-Lösungen. Portal-Lösungen werden über den Webbrowser verfügbar gemacht und können auf personalisierte Daten via Services zugreifen und unter Berücksichtigung entsprechender Zugriffsberechtigungen und Sicherheitsaspekte entlang der Geschäftsprozesse zur Verfügung gestellt werden. Denkbar wäre ebenfalls eine Einbeziehung von Stakeholdern durch die Distribution entsprechender Dashboard- Services zu Informationszwecken.

4 Anwendungsfall

Das vorgestellte BI-Konzept auf Basis einer EDSOA kann in der Praxis in unterschiedlichen Bereichen eingesetzt werden. Nachfolgend wird ein Anwendungsfall aus der Logistik exemplarisch aufgegriffen.

Im Bereich von Expressgut- und Schnellopaket-Zustellungen werden Lieferungen für verschiedene Kunden zu Touren zusammengefasst. In der Regel werden verspätete Zustellungen mit entsprechenden Konventionalstrafen sanktioniert. Dies erfordert eine durchgehende Überwachung der Touren mittels vordefinierter Etappenziele, um auf Probleme, wie z.B. Terminverzögerungen, unverzüglich reagieren zu können. Diese Überwachung beinhaltet eine Kontrolle, ob sich die Zustellungen im Zeitplan befinden. Im Falle einer Verzögerung im geplanten Versandablauf, die z.B. durch ein defektes Transportfahrzeug hervorgerufen wird, muss auf Grundlage einer Kosten-/Nutzenanalyse eine kurzfristige Anpassung der Tourenplanung vorgenommen werden. Hierzu sind neben operativen auch analytische Daten für eine verbesserte Entscheidungsunterstützung notwendig.

Beim Eintritt einer Terminverzögerung wird durch das auslösende Geschäftsereignis „Verzögerung bei Expresszustellung“ ein Ereignis auf dem ESB initiiert und übertragen. Ein Disponent, als Abonnent eines solchen Ereignisses, wird sofort in Kenntnis gesetzt und entsprechende Prozesse werden zur Tourenanpassung angestoßen. Die Benachrichtigung des Disponenten kann mithilfe eines Dashboard-Service erfolgen, der basierend auf einer Nachricht auf dem ESB getriggert wird. Parallel wird eine Prüfung der Daten aus einem GPS(Global Positioning System)-Tracking-System des defekten Fahrzeugs mit GPS-Daten aktuell betriebsbereiter Fahrzeuge (in regionaler Nähe) in Echtzeit ausgelöst. Über die identifizierten Fahrzeuge können operative Daten bezüglich ihrer Tour und freien Kapazitäten abgefragt werden. Weiterhin muss eine Analyse über die Art und Höhe der Konventionalstrafen und den Customer Lifetime Value (CLV) erfolgen (Kosten-/Nutzenanalyse), um eine verbesserte Grundlage für den Disponenten bezüglich der komplexen Entscheidung der Tourenanpassung/-zusammenlegung herbeizuführen. Dies kann mithilfe eines analytischen Services erreicht werden. Auf der Grundlage der gelieferten Informationen führt der Disponent eine Tourenanpassung durch. Als Ergebnis der Umplanung werden dann wiederum automatisch neue Ereignisse auf dem ESB initiiert, die z.B. einen Service anstoßen, um die von einer verspäteten Zustellung betroffenen Kunden per E-Mail zu informieren.

Die Zusammenführung der operativen und analytischen Informationen in Echtzeit, ausgelöst durch ein Ereignis, zeigt den Nutzen auf, den die Einführung eines BI-Konzepts auf Basis einer EDSOA stiften kann. Auf diese Weise können die Anforderungen von Echtzeitunternehmen zur Umsetzung von Operational Business Intelligence erfüllt werden.

5 Fazit und Ausblick

Die veränderten Anforderungen an Unternehmen ziehen Modifikationen der Systemlandschaften nach sich, die auch die analytischen Informationssysteme umfassen. Das Ziel dieses Beitrags bestand darin, durch die Entwicklung eines BI-Konzepts auf Grundlage einer EDSOA diesen veränderten Anforderungen Rechnung zu tragen. Die Implementierung einer SOA ermöglicht monolithische Systeme zu verteilten, modularen und lose gekoppelten Systemen zusammenzuführen, die eine flexible Darstellung von Geschäftsprozessen ermöglichen. Durch die Abbildung von Geschäftsereignissen im Rah-

men einer EDA besteht die Möglichkeit Geschäftsprozesse echtzeitnah auszuwerten und darzustellen. Unternehmen können sich schnell und flexibel an die veränderten Anforderungen anpassen und strategische sowie operative Entscheidungen unter Zuhilfenahme von analytischen Services treffen. Infolgedessen wird eine vollständig flexible und agile Infrastruktur realisiert. Zudem können Analysen automatisiert angestoßen und dem Endnutzer zur Entscheidungsunterstützung zur Verfügung gestellt werden. Außerdem wird eine weitreichende Automatisierung von repetitiven Entscheidungen durch eine Rule Engine ermöglicht. Es lassen sich ebenfalls mithilfe der Kombination von Complex Event Processing und einer Rule Engine prädiktive Szenarien erkennen, um darauf automatisiert reagieren zu können oder den entsprechenden Endnutzer für den manuellen Entscheidungsprozess rechtzeitig zu informieren.

Das vorgestellte Konzept stellt einen ersten Ansatz dar, den Anforderungen, wie bspw. der Echtzeitunterstützung auf operativer Ebene, die an Business Intelligence gestellt werden, gerecht zu werden. Die Überführung der Komponenten aktueller DWH-Ansätze in ein generisches EDSOA-Modell zeigt einen Paradigmenwechsel in der bislang getrennten Betrachtung von operativen und analytischen Informationssystemen auf. Vor diesem Hintergrund muss zukünftig eine spezifische Betrachtung der einzelnen Komponenten und deren Zusammenspiel sowie möglicher technischer und organisatorischer Probleme durchgeführt werden. Zukünftig werden höhere Anforderungen an Betriebskonzepte analytischer Informationssysteme gestellt, wie z.B. eine erhöhte Verfügbarkeit (24x7). Weiterhin sind Probleme beim ETL-Prozess auf Basis eines Enterprise Service Bus im Bereich der Massendatenverarbeitung zu erwarten.

Die Entwicklung eines BI-Konzepts auf Grundlage einer EDSOA zeigt, dass eine Annäherung von operativen und analytischen Informationssystemen vollzogen wird. Die Systeme treten somit zukünftig vermehrt in den Hintergrund und der Geschäftsprozess in den Mittelpunkt der Betrachtung.

6 Literaturverzeichnis

- [CK06] Chemburkar, A.; Keny, P.: Trends in Operational BI. <http://www.dmreview.com/issues/20060701/1057833-1.html>, 2006-07-01, Abruf am 2007-12-14.
- [Di07] Dittmar, C.: Latenzzeiten von Business Intelligence-Systemen. In (Gluchowski, P.; Chamoni, P.; Gersch, M.; Krebs, S.; Reinersmann, M. Hrsg.): Schlaglichter der Wirtschaftsinformatik. Verlag der Gesellschaft für Unternehmensrechnung und Controlling m.b.H., Chemnitz, 2007, S. 131-142.
- [Ec07] Eckerson, W.: Best Practices in Operational BI – Converging Analytical and Operational Processes. <http://www.tdwi.org/research/display.aspx?ID=8621>, 2007-07-01, Abruf am 2008-01-03.
- [FB06] Frank, W.; Baier, A.: Business Intelligence in SOA: Das Zusammenspiel von SOA- und BI-Konzepten – Ein Weg aus der Sackgasse. In: Javamagazin (2006) 11, S. 54-62.
- [JS07] Jonas, P.-K.; Scheuch, R.: Process Warehouse (PWH) – Metadatengetriebene Datenbewirtschaftung. In: BI-Spektrum (2007) TDWI Sonderheft, S. 29-34.

- [Li07] Liebhart, D.: SOA goes real – Service-orientierte Architekturen erfolgreich planen und einführen. Carl Hanser Verlag, München, 2007.
- [LH06] Liebhart, D.; Hänggi, F.: Does BI really needs SOA? Trivadis AG, Stuttgart, 2006.
- [Lu02] Luckham, D.: The Power of Events: An introduction to Complex Event Processing in Distributed Enterprise Systems, Addison-Wesley, Boston et al., 2002.
- [Ma06] Martin, W.: Analytics meets Enterprise SOA – Wertschöpfende Geschäftsprozesse durch Analytik. S.A.R.L. Martin, Annecy, 2006.
- [Ma06a] Martin, W.: Arcplan Information Services – Prozess- und service-orientierte Business Intelligence mit analytischen Services. S.A.R.L. Martin, Annecy, 2006.
- [Ma07] Martin, W.: SOA basierendes Geschäftsprozessmanagement. S.A.R.L. Martin, Annecy und IT Research, Sauerlach 2007.
- [Ph06] Philippi, J.: SOA trifft auf BI. <http://www.cio.de/knowledgecenter/bi/828707/>, 2006-10-25, Abruf am 2008-02-20.
- [Sc06] Schelp, J.: „Real“-Time Warehousing und EAI. In (Chamoni, P.; Gluchowski, P. Hrsg.): Analytische Informationssysteme. Springer, Heidelberg, 2006, S. 425-438.
- [Sc07] Schelp, J.: Ansatzpunkte zur Übertragung Serviceorientierter Konzepte auf das Data Warehousing. In (Gluchowski, P.; Chamoni, P.; Gersch, M.; Krebs, S.; Reinersmann, M. Hrsg.): Schlaglichter der Wirtschaftsinformatik. Verlag der Gesellschaft für Unternehmensrechnung und Controlling m.b.H., Chemnitz, 2007, S. 143-157.
- [Sc07a] Scheuch, R.: Quo Vadis Business Intelligence? Der BI-Markt erschließt Konvergenzen und entwickelt sich weiter. <http://www.apverlag.de/Online-Artikel/20070304/20070304w%20Opitz%20Consulting%20Quo%20vadis%20Business%20Intelligence.htm>, 2007-03-04, Abruf am 2007-02-19.
- [SR05] Sriraman, B.; Radhakrishnan, R.: Event Driven Architecture Augmenting Service Oriented Architectures. <http://soa.omg.org/Uploaded%20Docs/EDA/edamdasoa.pdf>, 2005-01-01, Abruf am 2008-01-28.
- [Th01] Thalhammer, T.: Active Data Warehouses – Complementing OLAP with Analysis Rules. Johannes Kepler Universität Linz, Linz, 2001.
- [TT03] Tho, N.; Tjoa, M.: Zero-Latency Data Warehousing for heterogeneous Data Sources and Continuous Data Streams. TU Wien, Wien, 2003.
- [Ve07] Ventana Research: Closing the Management Information Gap with Operational BI. <http://www.ventanaresearch.com/research/vendor.aspx?id=107>, 2007-05-01, Abruf am 2007-12-18.
- [Wh06] White, C.: A roadmap to enterprise data integration. IBM, Somers, 2006.
- [Wo06] Woolf, B.: Event-Driven Architecture and Service-Oriented Architecture. http://www.haifa.il.ibm.com/Workshops/oopsla2006/present/w06_eda_woolf.pdf, 2006-08-24, Abruf am 2008-01-28.
- [Za07] Zacharias, R.: SOA & Event Driven Architecture (EDA) – Eine perfekte Symbiose. In: Javamagazin (2007) 7, S. 60-69.