

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

Marco Pospiech, Carsten Felden

Professur für Informationswirtschaft/Wirtschaftsinformatik  
TU Bergakademie Freiberg  
Lessingstraße 45, 09599 Freiberg  
marco.pospiech@bwl.tu-freiberg.de  
carsten.felden@bwl.tu-freiberg.de

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

## 1 Einleitung

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

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

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

## **2 Status Quo**

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

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

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

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

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

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

### 3 Referenzmodell zur serviceorientierten Business Intelligence

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

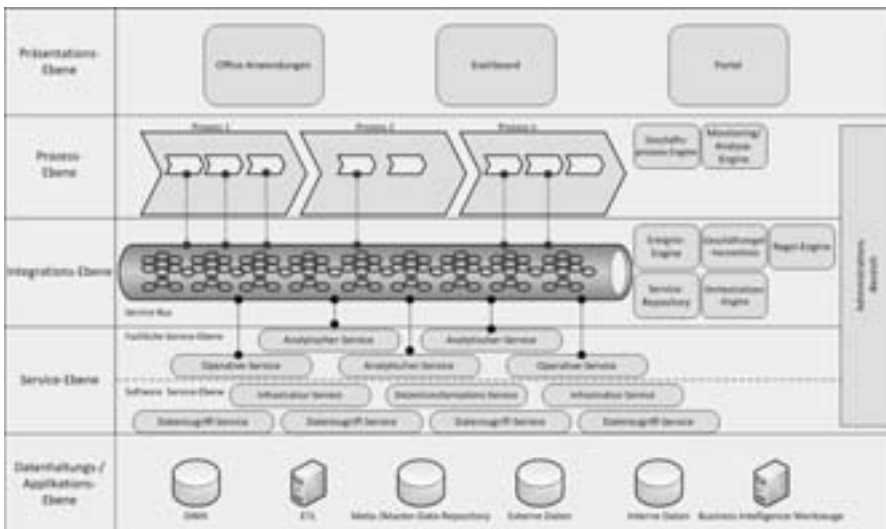


Abbildung 1: Serviceorientierte Business Intelligence Referenzarchitektur

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

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

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

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

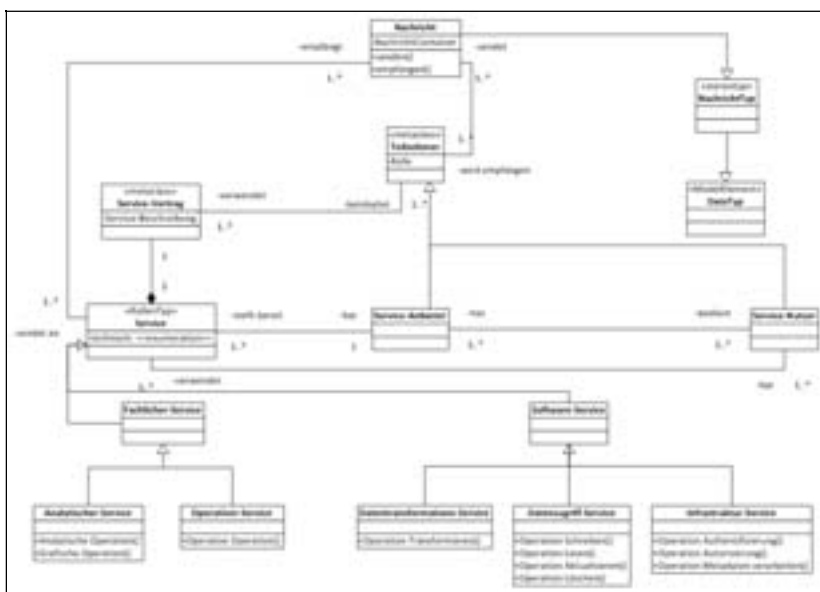


Abbildung 2: Service-Ebene

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

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

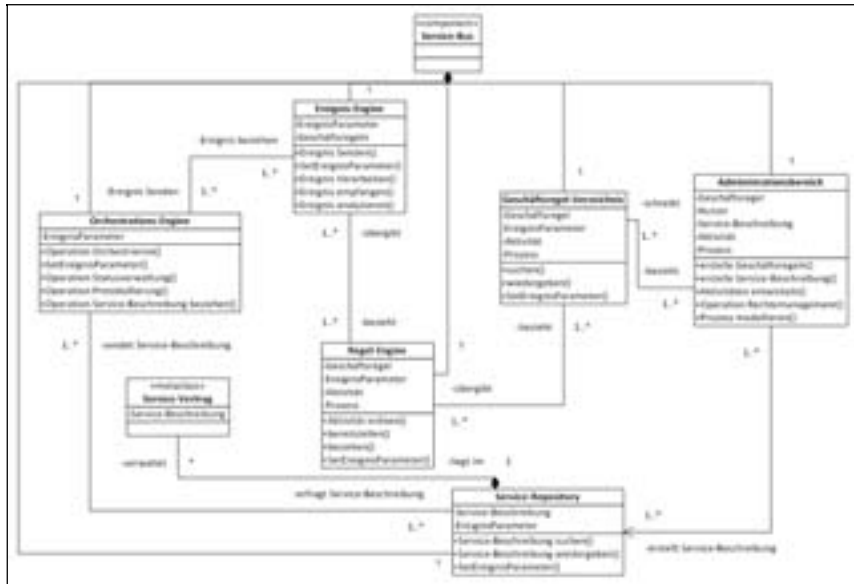


Abbildung 3: Integrations-Ebene

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

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

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

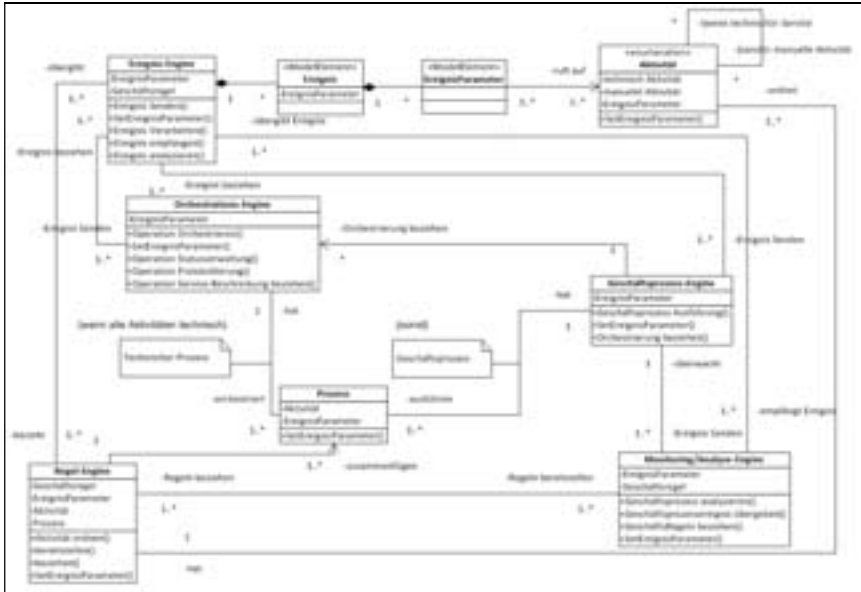


Abbildung 4: Prozess-Ebene



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

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

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

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

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

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

## 4 Fazit

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

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

Resümierend kann konstatiert werden, dass SoBI weiterer Forschung bedarf [Di08], jedoch bereits jetzt ein vielversprechendes Konzept darstellt, welches den wechselnden BI-Paradigmen gerecht wird und in der Summe die Systemvitalität von BI erhöht [GI08]. Hierbei liegt der nächste Schritt in der Realisierung eines Prototyps, um wertvolle Erkenntnisse für den Aufbau zukünftiger Systeme gewinnen zu können.

## Literaturverzeichnis

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

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

- [Ng05] Nguyen, T. M.; Schiefer, S.; Tjoa, A. M.: Sense & response service architecture (SARESA): An approach towards a real-time business intelligence solution and its use for a fraud detection application, in: Proceedings International Workshop on Data Warehousing and OLAP (DOLAP), 2005; S. 77-86.
- [NO09] Nijkamp, E.; Oberhofer, M.: Embedded Analytics in Front Office Applications, in: Gesellschaft für Informatik BTW, Vol. 144, 2009; S. 449-459
- [Pa08] Panian, Z.: How to Make Business Intelligence Actionable through Service-oriented Architecture, in: Transactions on Business and Economics, Vol. 5, 5, 2008; S. 210-221.
- [PH11] Piller, G.; Hagedorn, J.: Einsatzpotenziale für In-Memory Data Management in betrieblichen Anwendungssystemen. In: Wirtschaftsinformatik, Vol. 53, 5, 2011; S. 18-25.
- [Ru11] Russom, P., (2011), TDWI Next Generation Data Integration, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2011/TDWI\\_BPRReport\\_Q211\\_NGDI\\_Web.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2011/TDWI_BPRReport_Q211_NGDI_Web.ashx), (05.07.2011).
- [Ru10] Russom, P., (2010), TDWI best practices Report operational data warehousing, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2010/TDWI\\_BPRReport\\_Q410\\_OpDW.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2010/TDWI_BPRReport_Q410_OpDW.ashx), (05.07.2011).
- [Ru09] Russom, P., (04.2009), TDWI best practices Report operational data integration A New Frontier for Data Management, [http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2009/TDWI\\_BPR\\_2Q09\\_ODI%20pdf.ashx](http://tdwi.org/~media/TDWI/TDWI/Research/BPR/2009/TDWI_BPR_2Q09_ODI%20pdf.ashx), (07.07.2011).
- [Si02] Sillitti, A.; Vernazza T.; Succi, G.: Service Oriented Programming: A New Paradigm of Software Reuse, in: Lecture Notes in Computer Science, Vol. 2319, 2002, 39-47.
- [Vo08] Vogt, T.; Neuhaus, S.; Linden, M.; Chamoni, P.: Business-Intelligence-Konzept auf Basis einer Event-Driven Service-Oriented Architecture, in: Dinter, B.; Winter, R.; Chamoni, P.; Gronau, N.; Turowski, K., (Hrsg.); Synergien durch Integration und Informationslogistik Proceedings DW 2008, 2008; S. 217-229.
- [Wh06] White, C., (06.2006), The Next Generation of Business Intelligence: Operational BI – BI Research, [http://certification.sybase.com/content/1041416/Sybase\\_OperationalBI\\_WP-071906.pdf](http://certification.sybase.com/content/1041416/Sybase_OperationalBI_WP-071906.pdf), (06.06.2011).
- [WH04] Wilkes, S.; Harby, J., (06.2004), A move to drive industry standardization of SOA, <http://xml.coverpages.org/SOA-BlueprintsConceptsV05.pdf>, (03.06.2011).
- [Wi08] Winter, R., (21.01.2008), SOA braucht eine klare Definition, <http://www.computerwoche.de/software/soa-bpm/1853224/index3.html>, (21.07.2011).
- [Za07] Zacharias, R.: SOA & Event Driven Architecture (EDA) – Eine perfekte Symbiose, in: Javamagazin, Vol. 7, 2007; S. 60-69.

