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. [Gi08] 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

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]

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 [KJ08]. Der aufkommende Nachrichtenverkehr und die einhergehenden Zugriffszeiten können anhand von In-Memory-Technologien verbessert werden [CB08].


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 neuzzeitigen BI-Anforderungen gerecht wird und eine Realisierung ermöglicht. Dabei bietet sich die Referenzmodellierung als Methode an. [DE98]

3 Referenzmodell zur serviceorientierten Business Intelligence


Abbildung 1: Serviceorientierte Business Intelligence Referenzarchitektur
In Abbildung 1 erfolgt eine semistрукturelle 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.


Abb. 2: Service-Ebene

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 prädictiv 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.


Abbildung 4: Prozess-Ebene


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]


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 besichert. 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 innenwohnenden 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.


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

Literaturverzeichnis


