

# Mit GOBIA zu zielgerichteten Business-Intelligence-Architekturen

David Fekete und Gottfried Vossen, Universität Münster

*Traditionelle Data-Warehouse-Architekturen [2,4], bestehend von unten nach oben etwa aus Datenquell-Ebene, Staging Area, Data-Storage sowie Anwendungs-Ebene, werden heutzutage durch eine Vielzahl neuartiger Big-Data-Tools wie Apache-Hadoop-Packages oder NoSQL-Datenbanken [3] infrage gestellt. Die neuartigen Tools machen es möglich, Daten zu verarbeiten und zu analysieren, die vorher nicht effizient oder überhaupt nicht betrachtet werden konnten.*

Zu Big Data [4,5] zählen nicht nur viel größere Datenmengen, sondern auch vielfältigere Daten (wie Bilder, Videos, Social-Media-Beiträge) und eine schnellere Datenverarbeitung (beispielsweise Sensordaten aus dem Auto oder Smart Home). Nicht selten überfordern die Herausforderungen um Big Data die eigenen organisatorischen Analyse-Kapazitäten eines Unternehmens. Verschärft wird das Problem durch die Vielzahl von verfügbaren Big-Data-Toolsets. Insbesondere Hadoop-Distributionen wie MapR oder Hortonworks enthalten sehr viele Tools für stark heterogene Einsatzszenarien (wie Spark für die Datenstrom-Analyse oder MapReduce für parallele Verarbeitung). Dadurch wird die Auswahl des passenden Technologie-Mix für die eigene Business-Intelligence-Architektur eine viel komplexere Aufgabe als in der Vergangenheit. Auch stellt sich bei der traditionellen Data-Warehouse-Referenz-Architektur die Frage, wie sich diese in Zukunft verändern muss oder sollte, um auch Big-Data-Fragestellungen angehen zu können.

## Data Warehouse + Big Data?

Diverse Ansätze, die versuchen, Data Warehouses (DWH) mit Big-Data-Tools zu kombinieren (etwa indem diese für die ETL-Schicht eingesetzt werden), bilden lediglich ausgewählte Anwendungsfälle ab, sodass daraus keine allgemeingültige Architektur abgeleitet werden kann. Bei allgemeineren Ansätzen, die die klassische DWH-Schichten-Architektur mit Big-Data-Tools erweitern [4], bleibt die Herausforderung, den richtigen Technologie-Mix zu wählen, denn diese Tools bilden sehr viele Technologien und

mögliche Szenarien ab. Verstärkt wird dies dadurch, dass viele Tools für mehrere Zwecke eingesetzt werden können und deren Rolle in der eigenen Architektur nicht von vornherein klar ist. Beispielsweise lässt sich MapReduce als generisches Tool sowohl für Datenvorverarbeitung im ETL-Prozess als auch für Analysen einsetzen.

## Referenz-Architektur

Ein Ansatz zum Umgang mit dieser Herausforderung ist die von der Universität Münster entwickelte GOBIA-Methode („Goal-oriented Business Intelligence Architectures“, zu Deutsch „zielorientierte BI-Architekturen“, siehe *Abbildung 1*). Diese beinhaltet eine Technologie-unabhängige Referenz-Architektur („GOBIA.REF“), die auf dem Grundprinzip einer Schichten-Architektur basiert, einem bekannten Konzept der Informatik [1]. Ergänzt wird sie durch einen Entwicklungsprozess („GOBIA.DEV“), der sicherstellen soll, dass die Big-Data-/BI-Funktionalität und damit letztlich auch die verwendete Technologie zu den Zielen und Anforderungen einer Organisation an ihr jeweiliges BI-System passend gewählt wird. Damit soll der Kernaspekt von BI eingehalten werden, dass BI-Systeme wie ein DWH an der Unternehmung an den für sie relevanten Use Cases und deren Zielen ausgerichtet sind.

Bei der GOBIA-Methode entsteht zunächst eine konzeptionelle BI-Architektur. In einem zweiten Schritt erfolgt die Auswahl der Technologien, basierend auf der konzeptionellen BI-Architektur, für die angestrebte angepasste Architektur. Diese bildet die Blaupause für das physisch zu implementierende BI-System.

„GOBIA.REF“ ist als einfache Schichten-Architektur entworfen, da sich die übrigen Architekturen im Wesentlichen auf dieses Grundkonzept zurückführen lassen. Um die Brücke zwischen diversen Technologien zu schlagen, gibt „GOBIA.REF“ keine Technologien vor und fokussiert sich stattdessen auf BI-Funktionalität (wie „Kunden in Segmente clustern“ oder „Produkt-Sentiments ermitteln“) und die Daten, die dafür erforderlich sind.

## Development-Methode

In „GOBIA.DEV“ sollen ausgehend von konkreten Geschäftszielen an das BI-System über Business Cases zunächst BI-Anforderungen abgeleitet werden. Da die Formulierung von Business Cases und Geschäftszielen in vielen Unternehmen zum Tagesgeschäft gehört, macht „GOBIA.DEV“ keine Vorgaben zu deren Herleitung.

Die BI-Anforderungen sollen insbesondere funktionale („Welche Funktionen soll das System bieten?“) und informationelle („Welche Informationen sollen vom BI-System an wen bereitgestellt werden?“) Anforderungen abdecken. Ergänzt werden diese beispielsweise durch rechtliche, organisationale und sonstige projektübliche Anforderungen. Eine BI-Anforderung könnte beispielsweise „Bereitstellung einer Analyse zum Kundenverhalten, um sein Verhalten auf Eigenschaften möglicherweise für ihn interessanter Produkte abzubilden“ sein.

Im weiteren Verlauf werden BI-Anforderungen, BI-Funktionalitäten und BI-Daten iterativ aufeinander abgestimmt (sogenanntes „Co-Alignment“). Ausgehend von den BI-Anfor-

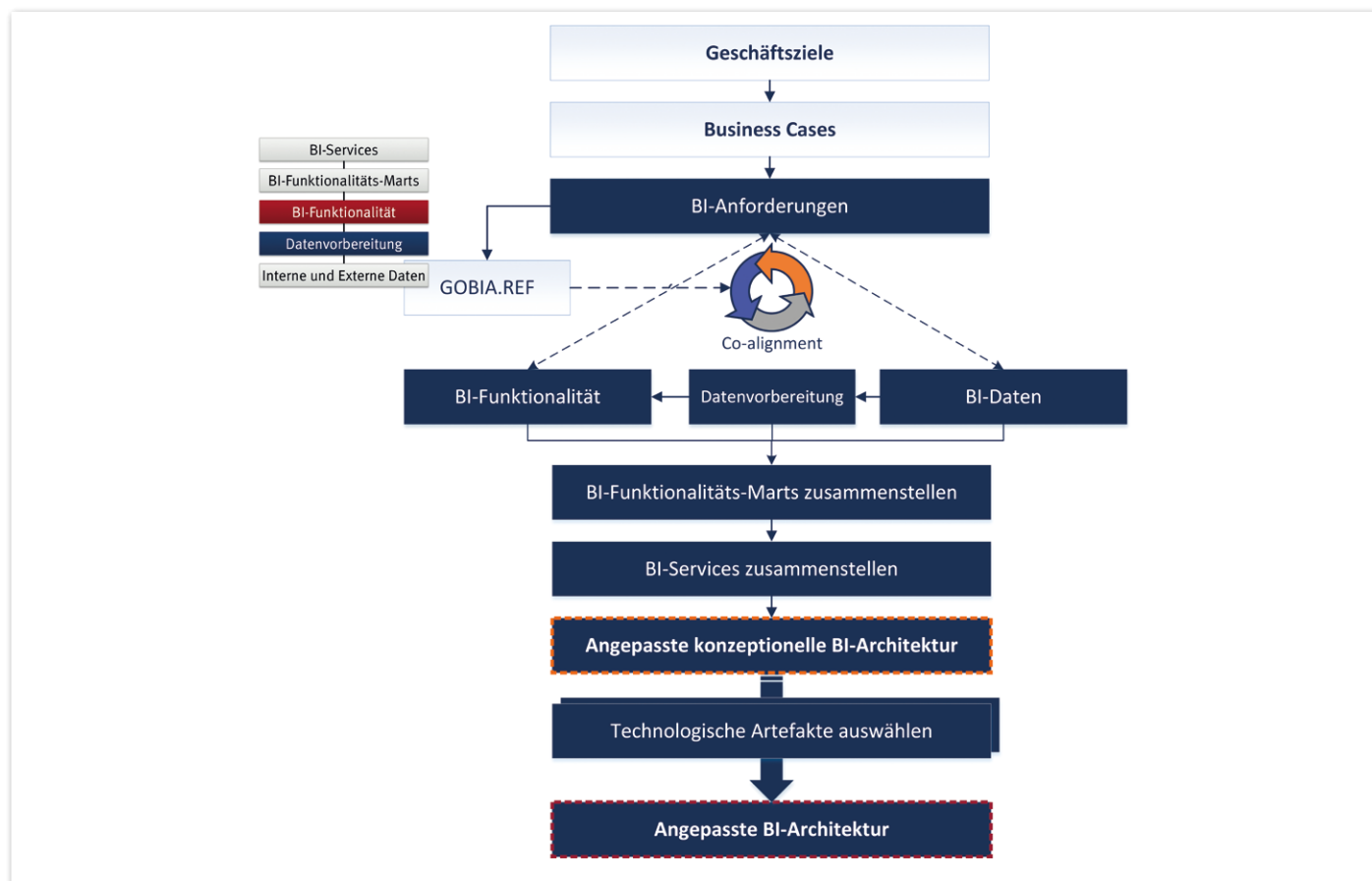


Abbildung 1: Die GOBIA-Methode

derungen werden die nötigen BI-Funktionalitäten und die passenden BI-Daten bestimmt, die diese Anforderungen umsetzen können. Beispielsweise könnten die BI-Funktionalitäten „Kunden nach Verhalten clustern“ und „Kundencluster auf Produkteigenschaften mappen“ die obigen Anforderungen erfüllen. Die BI-Funktionalitäten konsumieren dabei die bereitzustellenden Daten. Um die Eigenschaften der gewählten BI-Daten festzuhalten, können die aus der Big-Data-Diskussion bekannten „V“s (wie Volume, Variety oder Velocity) eingesetzt werden.

Ein Rückfluss zu den BI-Anforderungen ist ebenfalls vorgesehen. Wenn etwa die obige Anforderung mangels Daten nicht erfüllt wird, lässt sich die Anforderung anpassen. Sollten sich durch bestehende Daten unerkannte Potenziale ergeben, können auch ambitioniertere Anforderungen formuliert werden. Die ermittelten Elemente werden den Schichten von „GOBIA.REF“ zugewiesen.

Für die Zukunft ist vorstellbar, dass domänenspezifische Varianten von „GOBIA.REF“ in den Prozess einfließen; so könnte für die Finanz-Industrie ein spezieller Baukasten, mit üblichen Fraud-Detection-Algorithmen als BI-Funktionalitäten, bereitgestellt werden.

Falls erforderlich, lassen sich Datenvorbereitungsschritte dazu wählen. Wenn beispielsweise die zugrunde liegenden Kundendaten für die BI-Funktionalität „Kunden nach Kaufverhalten clustern“ nicht die erforderliche Qualität aufweisen, können im Sinne des bekannten ETL-Prozesses qualitätssteigernde Maßnahmen erfolgen.

Im Anschluss wird gemäß den Anforderungen die passende Bereitstellungsform der BI-Funktionalität bestimmt, etwa nach Datenausschnitt wie bei Data Marts im Data Warehouse oder prozessbasiert. Damit werden letztlich die BI-Services aufgespannt, die später durch konkrete Applikationen umgesetzt werden. Damit ist die konzeptionelle Architektur als Zwischenschritt der GOBIA-Methode vollständig.

#### Fazit

Die GOBIA-Methode soll mit der strukturierten Erfassung von BI-Funktionalitäten und BI-Daten sowie deren Zusammenhänge die anschließende Technologieauswahl erleichtern. Die aktive Forschung der Universität Münster zur GOBIA-Methode konzentriert sich derzeit auf die Transformation dieser konzeptionellen BI-Architektur mit ihren einzelnen Ele-

menten in eine mit den geeigneten Technologien gefüllte, individuelle BI-Architektur.

#### Literatur

- [1] Abeck, S., Lockemann, P.C., Schiller, J., Seitz, J.: Verteilte Informationssysteme: Integration von Datenübertragungstechnik und Datenbanktechnik. dpunkt, Heidelberg (2003)
- [2] Bauer, A., Günzel, H.: Data Warehouse Systeme. dpunkt, Heidelberg, 3rd edn. (2009)
- [3] Lechtenböcker, J., G. Vossen: NoSQL, NewSQL, MapReduce und Hadoop; Kapitel 10 in: P. Gluchowski, P. Chamoni (Hrsg.): Analytische Informationssysteme — Business Intelligence-Technologien und -Anwendungen, 5. Auflage, Springer-Verlag, Berlin 2016, 205–223
- [4] Vossen, G.: Big data as the new enabler in business and other intelligence. Vietnam Journal of Computer Science 1(1), 3-14 (Feb 2014)
- [5] Vossen, G.: Big Data: Daten sammeln, aggregieren, analysieren, nutzen; in T. Schwarz (Hrsg.): Big Data im Marketing — Chancen und Möglichkeiten für eine effektive Kundensprache; Haufe-Verlag, Freiburg, 2015, 35–54

David Fekete  
david.fekete@wi.uni-muenster.de

Gottfried Vossen  
g.v@www.de