



Managed Self Service BI mit ODI und OBI

Michael Klose, CGI Deutschland Ltd. & Co. KG, Vorstand DOAG BI Community

In den letzten Jahren sind die Anforderungen an die Flexibilität von BI-Systemen durch Fachanwender ständig gestiegen. Dadurch erhielten viele Anbieter von Spezialanwendungen Zugang zu den Fachbereichen, um dort eigenständige Produkte für Self Service BI zu platzieren. Oftmals sind diese nicht Bestandteil der Unternehmensstrategie und führen zu einem großen Portfolio an eingesetzten Produkten in einem Unternehmen.

Die Spezialanwendungen zeichnen sich durch eine einfache Bedienbarkeit aus und erfordern keine oder nur wenige IT-Kenntnisse. Klassische BI-Tool-Anbieter wie Oracle, IBM, SAP oder MicroStrategy haben auf diesen Trend reagiert und das eigene Produkt-Portfolio dahingehend erweitert. Oracle ermöglicht mit der aktuellen Business-Intelligence-Version sowohl klassisches Reporting als auch Visual Analytics innerhalb eines Produkts. Um die Daten möglichst schnell Fachbereichen für Auswertungen über das Data Warehouse bereitstellen zu können, unterstützt der Oracle Data Integrator den Managed-Self-Service-BI-Ansatz.

Anforderungen an Self Service BI

In den Fachbereichen besteht schon lange der Wunsch, schnell und selbstständig Ad-hoc-Berichte, Dashboards und Analysen auf Data-Warehouse-Datenbeständen erstellen zu können. Problematisch oder hinderlich sind dabei häufig etablierte Prozesse im IT-Bereich, die es oft nicht erlauben, neue Datenbestände schnell in das (unternehmensweite) Data Warehouse zu integrieren und für Auswertungszwecke bereitzustellen. Zudem existieren meist Release-Zyklen durch Abhängigkeiten von Quellsystemen sowie fest definierte Testschritte wie Systemintegrations- und Endbenutzer-Tests.

Standardberichte und Analysen können gut durch die IT bereitgestellt werden, doch wie verhält es sich im Bereich „Visual Analytics“? Hier möchten Fachanwender die Datenbestände selbstständig erkunden, um unbekannte Zusammenhänge zu erkennen und daraus neue, verfeinerte Analysen ohne Wartezeit auf die IT umzusetzen.

In vielen Fällen sind auch die im Data Warehouse bereitgestellten Daten nicht ausreichend. Dies kann beispielsweise der Fall sein, wenn externe Daten von Drittanbietern (wie GfK für Marktforschungszwecke) mit den bestehenden Daten verknüpft werden müssen, um etwa neue

Marketing-Kampagnen zu erstellen. Hier ist es zwingend notwendig, die Daten ad hoc zusammenführen zu können. Gerade die Integration eigener Daten(quellen) kann es notwendig machen, bestehende Datenmodelle dynamisch anzupassen, wie es bei der Aufbereitung und Ergänzung von Hierarchien der Fall sein könnte. Auch Daten, die zum Beispiel ergänzende Informationen für einen Kunden liefern, müssen an die bestehenden Tabellen angehängt werden.

Die aus Berichten und Visual Analytics gewonnenen Erkenntnisse sollten über eine gemeinsame Collaboration-Plattform geteilt werden können. Dies trägt erheblich zur Reduzierung des Gesamtaufwands bei, da keine Auswertungen doppelt erstellt werden müssen. Zudem kann eine Analyse als Basis für andere Fachbereiche verwendet werden, wenn beispielsweise eine Kundenbasis für eine spezielle Kampagne bereits vorselektiert wurde.

Eine immer größere Bedeutung erlangt auch der Einsatz mobiler Endgeräte wie Tablets, um Ergebnisse zu präsentieren oder auch Analysen zu erstellen. Dies ermöglicht eine Erstellung am Arbeitsplatz-Rechner und die Fortsetzung oder Präsentation der Ergebnisse in einem Workshop oder Meeting.

Herausforderungen von Self Service BI

Viele Applikationen erfüllen die beschriebenen Anforderungen an Self-Service-BI-Systeme. Allerdings ergeben sich daraus auch Anforderungen, die bedient werden müssen. So sind für die Erstellung von Berichten und Analysen Prozesse im Unternehmen einzuhalten. Dies betrifft beispielsweise die Produktivsetzung eines neuen Release, das die Phasen „Entwicklung“, „Integrationstest“ und „Endbenutzertest“ durchlaufen muss. Sofern im Unternehmen ein Business Intelligence Competence Center (BICC) oder eine ähnliche Organisationseinheit existiert, ist es notwendig, dass Kennzahlen dort zur Vereinheitlichung abgestimmt und genehmigt werden.

Im Bereich „Visual Analytics“ gibt es eine Vielzahl von Anbietern, die meist nur diesen speziellen Bereich abdecken und somit eher als Insellösungen gelten. Oft sind diese auf Arbeitsplatz-Rechnern und nicht zentralisiert auf einem dafür vorgesehenen und ausreichend mit Arbeitsspeicher und CPU-Leistung versehenen Server installiert.

Daten-Integration wird bei Self-Service-BI-Systemen häufig durch die Fachbereiche

durchgeführt. Hier stellt sich allerdings die Frage, wie und durch wen garantiert werden kann, dass die Daten korrekt sind. Nicht-automatisierte Datenlieferungen wie spezielle Ad-hoc-Extrakte aus Quellsystemen können falsche oder nicht vollständige Daten liefern, die zu irreführenden oder falschen Analyse-Ergebnissen führen.

Auch unvollständige oder manuell bearbeitete Daten können Datenqualitäts-Probleme hervorrufen. Hier muss überlegt werden, was beispielsweise mit falschen oder veralteten Adressdaten geschieht oder wie mit im falschen Format manuell eingetragenen Datumswerten oder Zahlen mit falschem Dezimal- und Tausender-Trenner umgegangen wird.

Wertet man die Daten auf lokalen Arbeitsplatz-Rechnern aus, stellt das Datenvolumen oft eine weitere Herausforderung dar. Zwar sind aktuelle Rechner durchaus mit 8,16 oder auch 32 GB Speicher zu bekommen, doch wird dies schnell durch die komplette Haltung der Daten „In-Memory“ verbraucht.

Im Bereich der Daten-Modellierung besteht die Herausforderung, neue (externe) Daten mit bestehenden Daten verknüpfen zu müssen. Ein gutes Beispiel dafür ist die Anreicherung von Kundendaten im Marketing. Hier stellt sich die Frage, wie diese verknüpft werden, da Schlüsselwerte sicherlich nicht übereinstimmen. Namens- und Wohnorts-Gleichheit ergeben noch keine hundertprozentig sichere Zuordnung. Oftmals sind hier komplexe Business-Regelwerke notwendig, um die Daten zu verknüpfen.

Werden Self-Service-BI-Systeme als Insellösungen auf Arbeitsplatz-Rechnern eingesetzt, ist eine Zusammenarbeit mit anderen Mitarbeitern oft schwierig beziehungsweise Daten oder Ergebnisse werden dann häufig per E-Mail versendet. Dies ist allerdings in Bezug auf Security-Anforderungen höchst riskant, im Unternehmen verboten oder durch entsprechende Mailserver-Filter gar nicht möglich.

Gleiches gilt bei der mobilen Verwendung der Daten. Einige Self-Service-BI-Applikationen besitzen Apps für Tablets, um die Daten ebenfalls zu exportieren und zu speichern. Was passiert im Falle eines Diebstahls? Reicht hier die Code-Eingabe als Sicherungsmechanismus aus?

Managed Self Service BI als Königsweg

Wenn man die Anforderungen und Herausforderungen gegenüberstellt, kristallisieren sich konträre Aspekte zwischen Fachbereich

und IT heraus. Während die IT auf Prozesse und Qualität bedacht ist, fordern Fachbereiche schnelle, flexible Lösungen, die teilweise auch nur für kurze Zeiträume (temporär) benötigt werden. Die Anforderungen an Managed Self Service BI, um obige Punkte für Fachbereich und IT gleichermaßen zu erfüllen, sind:

- *Schnell*
 - Einfache Datenmodellierung ist gewährleistet
 - Keine beziehungsweise wenig Business-Logik ist zu implementieren
 - Schnelle Entwicklung der Ladeprozesse und Daten-Integration
- *Flexibel*
 - Variables, leicht erweiterbares Datenmodell
 - Einfache Einbindung neuer Datenquellen und Systeme
- *Temporär*
 - Spielweise für den Fachbereich
 - Kurzfristige Spezialauswertungen (wie Data Mining) möglich
 - Einmalige Daten-Analyse ohne dauerhafte Integration in das DWH
- *Prozesse*
 - Einhaltung des Deployment-Verfahrens durch DEV, SIT, UAT, PROD
 - Sicherstellung des Betriebs
 - Einbindung in die Security-Landschaft
- *Qualität*
 - Nur Integration von verifizierten Daten
 - Vollständige Prüfung auf Integrität

Man greift diese Anforderungen auf und versucht, eine geeignete Maßnahme zur Realisierung zu identifizieren. Im Mittelpunkt eines Data Warehouse steht die konsistente und historisierte Speicherung der Daten in einem vereinheitlichten Datenmodell. In der Vergangenheit wurde das Core Data Warehouse häufig mit einem (nahezu) Dritte-Normalform-Datenmodell erstellt. Durch die Redundanz-Freiheit entstehen viele Abhängigkeiten zwischen den Tabellen, was ebenfalls bedingt, dass verschiedenste Ladeprozesse implementiert werden müssen. All dies steigert den Komplexitätsgrad, reduziert die Entwicklungsgeschwindigkeit und erhöht den Testaufwand.

Gerade im Bereich der Datenmodellierung und Methodik hat sich in den letzten fünf Jahren mehr und mehr die Data-Vault-Technik durchgesetzt. Durch die geringe Anzahl verschiedener Tabellen-Typen (Hub, Link, Satellite) ist es möglich, die Ladeprozesse auf diese Anforderungen zu begrenzen. Dies führt letztendlich dazu, dass nur drei verschiedene Arten nach einem einheitlichen Muster entwickelt werden müssen.

Was bedeutet das nun im Kontext von „Managed Self Service BI“, bezogen auf die Anforderungen und Herausforderungen von IT und Fachbereichen? Durch einen Generator-Ansatz müssen Ladeprozesse (Mappings) nicht mehr manuell codiert, sondern können durch Konfiguration generiert werden. Dies ermöglicht die schnelle Entwicklung von Ladeprozessen und die leichte Einbindung neuer Datenquellen. Zudem muss der Generator einmalig intensiv getestet werden, die generierten Mappings allerdings nicht mehr, da hier dann keine Fehler mehr auftreten können. Das erlaubt ein schnelles Deployment durch die verschiedenen Umgebungen, ohne viel Aufwand und Zeit in Tests investieren zu müssen.

Die Nachteile des stark normalisierten und starren Dritte-Normalform-Datenmodells lassen sich mit einem Raw-Data-Vault-Ansatz umgehen. Raw Data Vault ermöglicht eine einfache Datenmodellierung (es gibt nur drei verschiedene Tabellen-Typen), ist sehr variabel und leicht erweiterbar. Zudem werden keine Transformationen oder Business-Logik auf die Daten angewendet. Details zu Data Vault stehen in diversen Fachbüchern.

Nachdem die Daten in das Core Data Warehouse in Raw-Data-Vault-Strukturen geladen wurden, stehen sogenannte „Raw Data Marts“ für Fachanwender bereit. Diese dienen sowohl als Spielwiese für Spezialauswertungen als auch zum Kennenlernen der Daten und Analysen im Visual-Analytics-Bereich. Oftmals ist es in diesem Umfeld leichter, mit Fachanwendern zukünftige dauerhafte Reports zu spezifizieren.

Umsetzung mit ODI und OBI

In den vorherigen Kapiteln wurden ausgiebig Anforderungen, Herausforderungen und ein möglicher Lösungsweg dargelegt. Nachfolgend ist ein möglicher Implementierungsweg mit Oracle-Produkten aufgezeigt und außerdem, welche Voraussetzungen dafür notwendig sind:

- **Datenmodellierung**
Die Stage-Tabellen im Data-Warehouse entsprechen „1:1“ den Tabellen aus dem Quellsystem, angereichert um technische Attribute wie Ladezeitpunkt und Schlüssel. Im Core Data Warehouse wird das Raw-Data-Vault-Modell nach strikten Namens-Konventionen erstellt. Dies hat zur Folge, dass keinerlei Typ-Konvertierungen wie „CHAR“ nach „NUMBER“ oder „CHAR“ nach „DATE“ durchgeführt werden. Business-Regeln wie Berechnungen oder Data Cleansing werden ebenfalls nicht umgesetzt.
- **Mapping Generierung**
Durch die strikten Konventionen bei der Daten-Modellierung und der Data-Vault-Methode ist es möglich, die Ladeprozesse mit keiner oder nur geringer Konfiguration zu generieren. Stage-Ladeprozesse können ohne weitere Konfiguration durch den Oracle Data Integrator unter Einsatz des API und Implementierung entsprechender Generatoren in Groovy automatisiert erstellt werden. Gleiches gilt für die Core-Ladeprozesse in die Raw-Data-Vault-Strukturen. Da hier im Bereich von Links nicht alles durch Namens-Konventionen abbildbar ist, müssen für Sonderfälle Tabellen und Attributs-Zuordnungen für den Generator erstellt werden.
- **„Star“-View-Generierung**
Raw-Data-Vault-Strukturen sind für einen Fachbereich durch die fremd wirkenden IDs und Tabellen-Konstrukte schwer verständlich und nicht für den direkten Zugriff geeignet. Allerdings ist es wiederum möglich, über das Data Dictionary automatisiert Views auf Basis der Links zu generieren. Letztendlich sind Links und die angebotenen Hubs nichts anderes als „Mini-Star“-Schemata. Wichtig ist auch, dass hier keine Typ-Konvertierungen oder Business-Regeln implementiert werden.
- **Quick Präsentation**
Die „Mini-Star“-Schemata, die durch die „Star“-View-Generierung entstehen, bilden die Basis für den Aufbau und die Integration in die Oracle-BI-Metadaten-Schicht. Um Daten schnell auswertbar bereitstellen zu können, sollte auch hier auf Business-Logik verzichtet werden. Da die Daten bereits im Star-Schema vor-

liegen, können sie einfach in die Metadaten integriert werden. Dies kann manuell oder auch hier durch den Einsatz eines Generators geschehen. Lediglich ein logisches Verbinden der Stars (Links) erfordert gegebenenfalls fachliches/technisches Know-how. Bis eine vollständige Integration in das Data Warehouse erfolgt, können die Präsentationen in einem getrennten Spielbereich bereitgestellt werden.

Fazit

Bei Self-Service-BI-Applikationen verwenden Fachanwender häufig die angelieferten Rohdaten, um selbst Bereinigungen und Zusammenführungen durchzuführen sowie Kalkulationen zu erstellen. Gleiches wird durch den oben beschriebenen Ansatz ebenfalls gewährleistet.

Durch den gewählten Managed-Self-Service-BI-Ansatz werden die Ladeprozesse nicht mehr manuell implementiert, sondern vom Datenmodell generiert. Sind Änderungen beispielsweise an Tabellen-Strukturen notwendig oder kommen neue Systeme und Tabellen hinzu, sind diese schnell umsetzbar.

Im Falle der Generierung sind kurze Release-Zyklen realisierbar, was auch im Data-Warehouse-Bereich den agilen Projektansatz unterstützt. Ein Einsatz von Insel-Lösungen für Analyse-Zwecke ist durch die Integration in die Enterprise-BI-Plattform nicht mehr notwendig. Ein wichtiger, nicht zu vernachlässigender Aspekt ist auch die vollständige Gewährleistung von Security- und Datenschutz-Anforderungen, da die Daten auf zentralen Servern und nicht auf lokalen Arbeitsplatz-Rechnern oder Mobile Devices liegen.

Michael Klose
michael.klose@cgi.com