

Schneller Einstieg in OBI 11g für den Mittelstand

Michael Weiler
PROMATIS software GmbH
Pforzheimer Strasse 160, 76275 Ettlingen

Schlüsselworte:

Oracle Business Intelligence (OBI), Modellierung, Star Schema, Referenzmodelle, ETL-Prozesse, BI-Projekte, Oracle Warehouse Builder (OWB), BI Answers, BI Interactive Dashboards.

Einleitung

Immer mehr mittelständische Unternehmen stehen vor der Herausforderung, ihre über lange Zeiträume hinweg gesammelten operativen Daten einfach und aussagekräftig auswerten zu können. Oracle bietet speziell dem Mittelstand mit der Standard Edition One seiner Business Intelligence Suite 11g (OBI SE1 11g) eine wirtschaftlich interessante Einstiegsmöglichkeit in eine leistungsfähige Auswertungsplattform.

Der Beitrag zeigt am Beispiel mehrerer konkreter Kundenprojekte, wie es gelingen kann, auf Basis OBI SE1 einen raschen Einstieg in das Thema Business Intelligence für mittelständische Unternehmen zu schaffen. Dabei werden die Einschränkungen der Standard Edition One gegenüber der Enterprise Edition bewertet und Lösungsmöglichkeiten zur Überwindung dieser Restriktionen aufgezeigt. In der Analysephase werden Vorlagen für Kennzahlen und Dimensionen eingesetzt. Für die Designphase werden Beispiele für Tabellenstrukturen in der Oracle Datenbank und Oracle Answers Beispielberichte mit entsprechenden OBI-Dashboards vorgestellt. Es existieren Beispiele aus den Bereichen Lagerhaltung, Produktion, Verkauf und Betriebsdatenerfassung (BDE). Mittels dieser Best Practice-Vorlagen ist ein rascher Einstieg gewährleistet.

Zur Unterstützung des ETL-Prozesses (Extraktion Transformation Laden) liefert Oracle den Oracle Warehouse Builder (OWB) in der Basisversion mit. Der Beitrag zeigt auf, wie die fehlende „Scheduling-Komponente“ über Skripte und Datenbankfunktionen ersetzt werden kann.

Abschließend werden Frage- und Problemstellungen aus der täglichen Praxis behandelt. Dabei werden Lösungsansätze beim Erstellen kundenspezifischer Datenmodelle und technische Fragestellungen diskutiert, wie z.B. der Zugriff auf ein ERP-System oder das Einlesen von BDE- oder Excel-basierten Daten.

Warum BI-Projekte scheitern

Im Kontext der Implementierung von Business Intelligence Projekten werden oftmals ähnliche Ursachen für das Scheitern angeführt. Im März 2011 veröffentlicht die Computer Woche unter dem Titel „Zehn Gründe – Woran BI-Projekte scheitern“ die Top 10 der Gründe für einen Fehlschlag:

1. Unklare Anforderungen
2. Falsche oder fehlende Daten
3. Zu späte Einbindung des Endanwenders
4. Zu späte vorzeigbare Ergebnisse
5. Fehlendes Change Management
6. Gesetzliche Anforderungen
7. Schlechte Dokumentation
8. Falsche Dimensionierung der Ressourcen
9. „Excel-minded“ Mitarbeiter

10. Zu geringes Budget

Viele dieser Punkte können durch vordefinierte Lösungen auf Basis der SE1 11g deutlich entschärft werden.

OBI Standard Edition One versus Enterprise Edition

Die Oracle Business Intelligence-Lösung in der Version 11g wird in zwei Ausprägungen, der Standard Edition One (SE1) und der Enterprise Edition, angeboten. Die SE1 beinhaltet neben dem BI Server mit Administrationskomponente eine Standard Edition One-Datenbank nebst ETL-Werkzeug Oracle Warehouse Builder core ETL und die Oracle Anwenderwerkzeuge Answers, Dashboards und BI Publisher. Mit diesen Komponenten ist der Aufbau einer umfassenden BI-Lösung möglich.

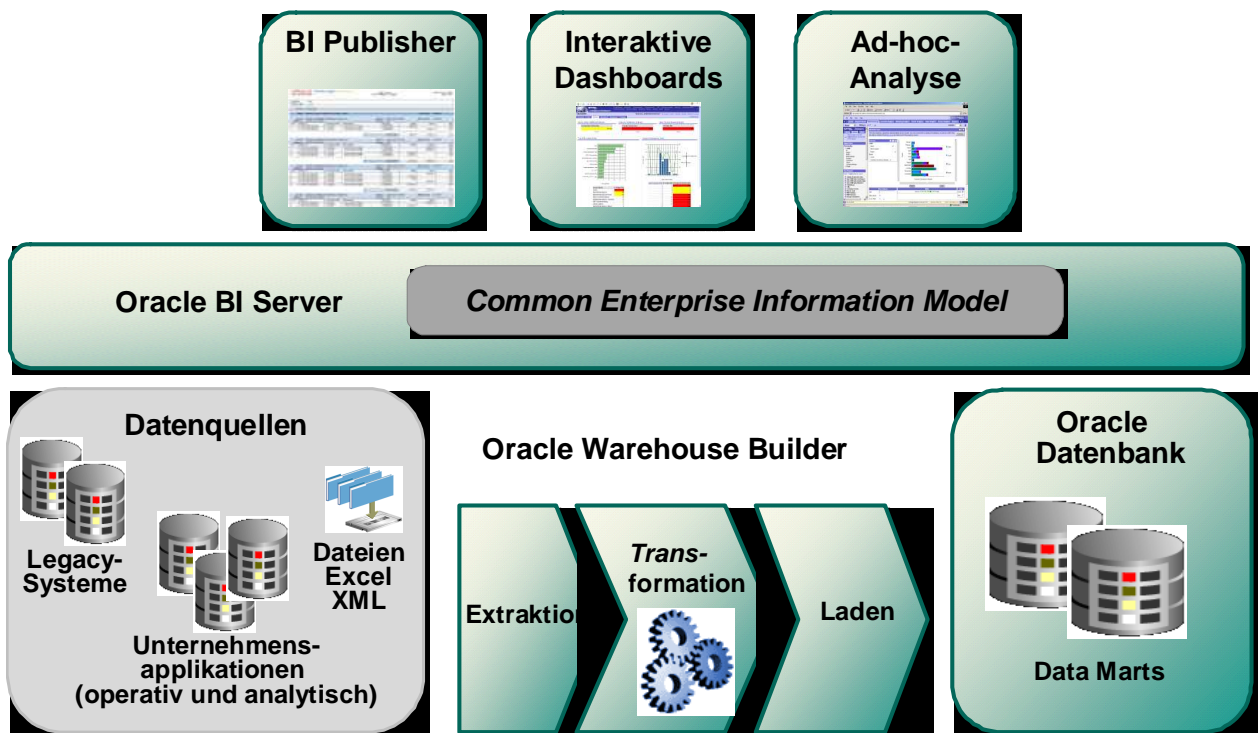


Abb. 1: BI-Lösung auf Basis der Oracle BI Suite Standard Edition One

Die wesentlichen Einschränkungen bei der Datenbank und dem BI Server sind die Benutzeranzahl und die Einschränkung der Hardware auf Systeme mit maximal 2 Sockeln. Im BI-Server können die Funktionen BI Delivers (Action Framework), BI Microsoft Office Add-In, BI Performance Management, BI Marketing und weitere nicht genutzt werden. Die grundlegenden Funktionen zum Aufbau der Dashboards sind jedoch mit BI Answers, BI Interactive Dashboards und BI Publisher möglich. Die Einführung bei Kunden hat gezeigt, dass mit diesen Grundfunktionen ein professioneller Einstieg zu erreichen ist. Über die Scheduling-Komponente des BI Publishers können Berichte automatisch per E-Mail versendet werden. Die Möglichkeit des Herunterladens von Tabellen aus dem Dashboard in Microsoft Excel ist zunächst ausreichend, da das Microsoft Office Add-In in einer späteren Ausbaustufe integriert werden kann.

Die Restriktion der Datenquellen auf die Oracle Datenbank und maximal eine weitere Datenbank schränkt den Aufbau umfassender BI Lösungen deutlich ein, da oftmals mehrere Quellen in den Unternehmen existieren. Als Lösung kann der Oracle Warehouse Builder (OWB) eingesetzt werden, mittels dessen man die Daten über einen ETL-Prozess in die Oracle Data Mart Datenbank überführen kann. Diese Architektur hat den Vorteil, dass das analytische System physikalisch von den operativen Systemen getrennt werden kann. Dies bringt Vorteile hinsichtlich Performance, Ausfallsicherheit und Betrieb mit sich. In dieser Umgebung kann der BI Server so konfiguriert werden, dass er lediglich auf die Data Marts in der Oracle Datenbank zugreift. Ein direkter Zugriff auf die operativen Systeme, eventuell vorhandene Altsysteme (Legacy Systeme) oder Excel-Dateien ist nicht notwendig.

Rasche Einführung durch vordefinierte Komponenten

In der Analysephase kommen vordefinierte logische Datenmodelle zum Einsatz. Dabei werden Kennzahlen, wie in Business Intelligence-Projekten üblich, über unterschiedliche Dimensionen strukturiert. Für jede Dimension kann es eine oder mehrere Hierarchien geben. Für jede Dimension und die entsprechenden Kennzahlen existieren vordefinierte CSV-Vorlagen, die Beispieldaten enthalten. Als Werkzeug wird das auch als Freeware-Version verfügbare Modellierungswerkzeug Horus eingesetzt (siehe www.horus.biz). Aus der Modellpalette wurden zur Definition der Kennzahlen und Dimensionen im Stil von Star- oder Snowflake-Schemata die Horus Objektmodelle verwendet. Die angehängten CSV-Dateien und die Screenshots aus einem Beispiel-Portal mit OBI 11g erleichtern die Diskussion mit dem Fachbereich. Aus mehreren dieser Fragmente entsteht ein kundenspezifisches Datenmodell, das Grundlage für die Implementierung ist. Die CSV Dateien stellen dabei eine direkte Vorlage für die Datenextraktion dar.

The screenshot displays the Horus Business Modeler interface. On the left, a project tree shows a hierarchy of objects. The main window shows a data table with columns for 'Maschinenlaufezeit' and 'Maschinenruehzeit'. Below the table, a diagram illustrates the relationships between 'Maschinenlaufezeit' and 'Maschinenruehzeit' using terms like 'ist zugeordnet', 'für', and 'zum'. A dialog box titled 'Eigenschaften für BDE Daten' is open, showing configuration options for the data source. At the bottom, an Excel spreadsheet shows the data extracted from the CSV files, with columns for 'MASCHINE', 'BEZEICHNUNG', 'GRUPPE', 'DATUM', 'RUNTIME', 'DOWNTIME', and 'DOWNTIME_SETUP'.

MASCHINE	BEZEICHNUNG	GRUPPE	DATUM	RUNTIME	DOWNTIME	DOWNTIME_SETUP
1	MASCHINE					
2	K350	G1	10.03.2011	20	4	1
3	K222	G1	10.03.2011	19	5	2
4						

Abb. 2: Vordefiniertes Datenmodell für den Bereich Betriebsdatenerfassung

Basierend auf unterschiedlichen Fragmenten wird ein kundenspezifisches Datenmodell entworfen, das in der Regel aus mehreren Objektmodellen besteht. Die Objektmodelle können im Anschluss direkt als

OBI-Themengebiete mit den entsprechenden Kennzahlen und Dimensionen abgebildet werden. Es existieren solche Fragmente für folgende Kennzahlen und Dimensionen

- BDE – Zeit, Maschinen
- Lagerwert – Zeit, Artikel, Lagerorganisation
- Umsatz – Zeit, Artikel, Kunde, Lieferadresse, Vertriebsmitarbeiter, Organisation
- Kontenwerte – Zeit, Konto, Kostenstelle, Kostenträger, Geschäftspartner, Budget (Planung)
- Lagertransaktionen – Zeit, Artikel, Lagerorganisation, Transaktionstyp

Neben den Datenmodellen hat sich eine Übersicht über den ETL-Prozess als Geschäftsprozessdarstellung bewährt. In diesem Prozess werden die Quellen und einige wichtige Rahmenbedingungen dokumentiert.

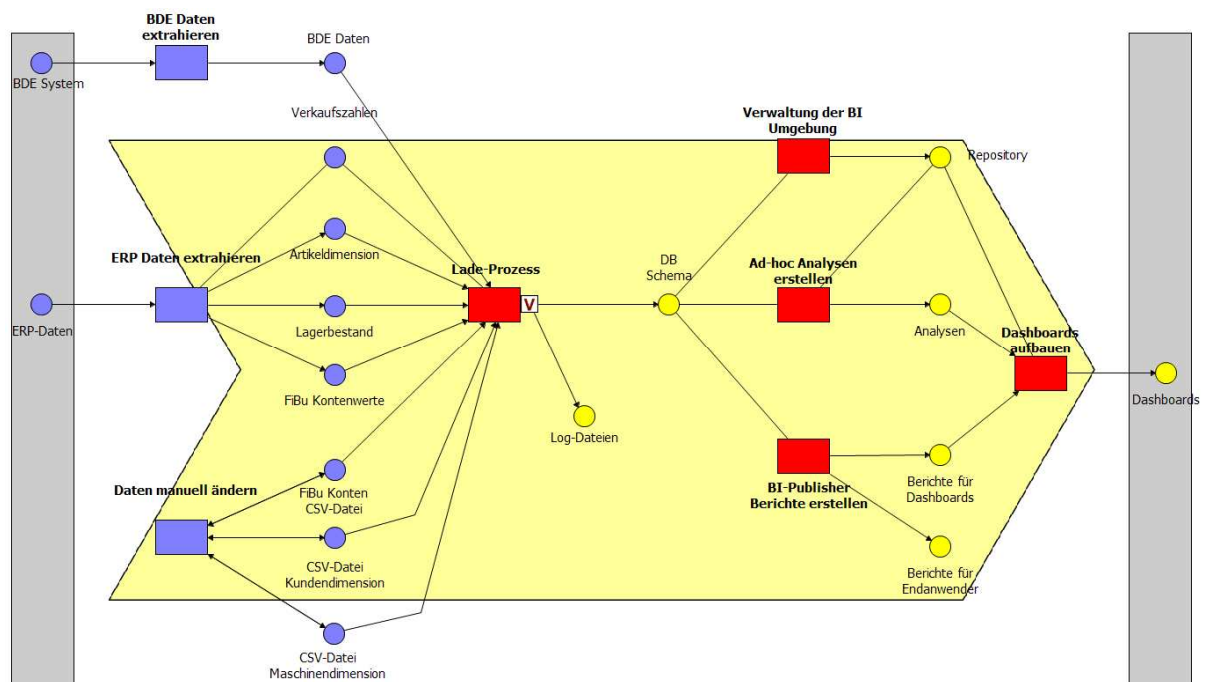


Abb. 3: Beispiel eines ETL-Prozesses

Noch einen Schritt weiter gehen die BI-Applikationen von Oracle, die ein vollständiges Datenmodell sowie implementierte ETL-Prozesse für unterschiedliche Vorsysteme beinhalten. Aufgrund des Umfangs der Datenmodelle und der Ladeprozesse sind diese Komponenten in der Einführung aufwändiger, bieten allerdings einen deutlich größeren Funktionsumfang.

Ersatz der OWB Scheduling-Komponente

In der Regel wird der Datentransfer täglich über Nacht erfolgen. Der Scheduler des OWB zum zeitgesteuerten automatischen Ablauf des Ladeprozesses darf in der Standard Edition One nicht genutzt werden. Auf Linux-basierten Systemen wurde diese Einschränkung mittels Shell-Skripten und des Betriebssystem-Befehls „crontab“ umgangen. Dabei werden die vom OWB erzeugten PL/SQL-Programme über sqlplus aufgerufen. Ein Beispiel für ein SQLPLUS Skript zeigt folgender Quelltext:

```
SET LINESIZE 500
SET PAGESIZE 0
```

```

SET ECHO OFF
SET FEEDBACK OFF
SET TERMOUT OFF
SET VERIFY OFF
SET TRIMSPOOL ON
SET TRIMOUT ON
SET SERVEROUTPUT ON
SPOOL &1

```

```

DECLARE
  P_STATUS VARCHAR2(200);
  P_MAX_NO_OF_ERRORS VARCHAR2(200);
  P_COMMIT_FREQUENCY VARCHAR2(200);
  P_OPERATING_MODE VARCHAR2(200);
  P_BULK_SIZE VARCHAR2(200);
  P_AUDIT_LEVEL VARCHAR2(200);
  P_PURGE_GROUP VARCHAR2(200);
  P_JOB_AUDIT VARCHAR2(200);
BEGIN
  P_MAX_NO_OF_ERRORS := NULL;
  P_COMMIT_FREQUENCY := NULL;
  P_OPERATING_MODE := NULL;
  P_BULK_SIZE := NULL;
  P_AUDIT_LEVEL := NULL;
  P_PURGE_GROUP := NULL;
  P_JOB_AUDIT := NULL;

  LOAD_F_ACCOUNT_VALUE.MAIN(
    P_STATUS => P_STATUS,
    P_MAX_NO_OF_ERRORS => P_MAX_NO_OF_ERRORS,
    P_COMMIT_FREQUENCY => P_COMMIT_FREQUENCY,
    P_OPERATING_MODE => P_OPERATING_MODE,
    P_BULK_SIZE => P_BULK_SIZE,
    P_AUDIT_LEVEL => P_AUDIT_LEVEL,
    P_PURGE_GROUP => P_PURGE_GROUP,
    P_JOB_AUDIT => P_JOB_AUDIT
  );
  DBMS_OUTPUT.PUT_LINE('STATUS = ' || P_STATUS);
END;

```

Dieses Ladeskript kann wiederum in einem Shell-Skript aufgerufen werden, das letztendlich über crontab zeitgesteuert die Beladung des Data Warehouse durchführt.

```

#!/bin/bash
# Umgebungsvariablen setzen
. /u01/scripts/setdb.env
ABSPFAD="$(cd "$(dirname "$0")";pwd)"

# BDE Daten laden
sqlplus ${BI_LOGIN} @${ABSPFAD}/load_bde.sql ${ABSPFAD}/load_bde.log

```

```
# Alle Logfiles verschieben
mv ${ABSPFAD}/*.log ${IMPORTFILES}/Logfiles
```

Der entsprechende Crontab Eintrag sieht beispielsweise wie folgt aus:

```
# BDE Tabellen um 6:30 Uhr laden
30 6 * * * /u01/scripts/import_6.30.sh
```

Die einzelnen Parameter der Crontab-Tabelle haben folgende Bedeutung:

1. Minute (eine Zahl zwischen 0 und 59)
2. Stunden (eine Zahl zwischen 0 und 23)
3. Tag des Monats (eine Zahl zwischen 1 und 31 oder * bedeutet jeden Tag)
4. Monat (eine Zahl zwischen 1 und 12 oder * bedeutet jeden Monat)
5. Wochentag (eine Zahl zwischen 0 und 6 wobei 0 der Sonntag ist oder * bedeutet jeden Wochentag)
6. Skript oder Kommando, das zum angegebenen Zeitpunkt ausgeführt wird

Die Erfahrung zeigt, dass mit geringem Aufwand ein zeitgesteuerter Ladeprozess mit den Linux-Standardfunktionen umgesetzt werden kann. Die Ladeprogramme werden über entsprechende Log-Dateien überwacht.

Projekterfahrungen

Bei einem mittelständischen Hersteller von Kunststoffspritzgussteilen wurde die OBI Standard Edition One in 15 Personentagen eingeführt. Solch eine rasche Einführung stellt natürlicherweise lediglich einen Einstieg in Business Intelligence dar. Aufgrund der vordefinierten Modelle konnten als Ergebnis erste Analysen und Dashboards bereitgestellt werden, die durch eigene Mitarbeiter des Kunden weiter ausgebaut werden können. Die Vorsysteme sind ein SQL Server-basiertes ERP-System und ein ebenfalls auf SQL Server basierendes BDE-System. Daneben existieren noch Excel-Dateien zur Anreicherung der Kunden- und Konteninformationen. Wesentliche Dimensionen sind Kunden, Lieferanten, Konten, Artikel, Maschinen, Zeit, Aufträge und Werkzeuge. Wesentliche Kennzahlen sind die Maschinenlaufzeiten, Maschinenrüstzeiten, Umrüsthäufigkeit, Soll und Haben, Transaktionswerte, Lagerbestand, Einstandspreis.

Bei der Anbindung von SQL Server-basierten Systemen stellt der Zugriff via SQL eine Herausforderung dar. Anders als bei Windows-Systemen existieren ODBC- oder JDBC-Treiber von Drittherstellern, die jedoch aufgrund der Lizenzkosten für eine Einstiegslösung ungeeignet sind. Eine Alternative stellt das Unix ODBC-Paket von Oracle im Zusammenspiel mit dem FreeTDS-Treiber dar. Bei der Konfiguration der Treiber muss beachtet werden, ob ein 32 Bit SQL Server oder ein 64 Bit SQL Server angebunden werden soll. Die einfacheren Treiber, die direkt mit der Linux Distribution mitgeliefert werden, verursachen oftmals Probleme bei langen Tabellennamen oder komplexen Abfragen. Die Lösungen sind zum einen Views, die auf der Microsoft-Seite einfach definiert werden können, und zum anderen die Einführung von Staging-Tabellen auf der Oracle-Seite. Dies ist ein gängiges und einfaches Konzept, bei dem die Daten zunächst auf die Oracle Datenbank kopiert und danach weiter verarbeitet werden. Die weitere Verarbeitung erfolgt mit dem OWB ausschließlich auf Oracle-Seite. Zur Analyse, zum Anlegen von Views und zum Test von Abfragen hat sich der Einsatz des SQL Developer 3.0 bewährt, der sowohl auf die Oracle Datenbanken als auch mittels eines Plug-Ins auf die SQL-Datenbanken problemlos zugreifen kann.

Beim Einsatz des OWB in der Standard Edition One darf die Funktionalität zum automatischen Aufbau von „Slowly Changing Dimensions“ nicht genutzt werden. Die Veränderungen in den Dimensionstabellen zu historisieren ist ein gängiges Konzept beim Aufbau von multidimensionalen

Datenmodellen. Für einen Einstieg in Business Intelligence kann auf dieses Konzept verzichtet werden beziehungsweise der Typ1 – eine einfache Aktualisierung des Datensatzes durchgeführt werden. Es sollte jedoch darauf geachtet werden, dass alle Dimensionen bereits mit Spalten wie „Gültig von“ und „Gültig bis“ ausgestattet sind, so dass eine nachträgliche Historisierung (im Konzept der Slowly Changing Dimensions als Typ2 oder Typ3 bezeichnet) über eigenentwickelte PL/SQL-Programme möglich ist.

Zusammenfassend ist nach Ansicht des Autors ein schneller Einstieg in Business Intelligence mit den hier vorgestellten Konzepten möglich. Wichtig ist jedoch, dass Dienstleister und Kunde gemeinsam einen engen Rahmen für das Einstiegsprojekt verabreden und diesen auch konsequent einhalten.

Kontaktadresse:

Dipl.-Wirtschaftsing. Michael Weiler

PROMATIS software GmbH

Pforzheimer Strasse 160

D-76275 Ettlingen

Telefon: +49 (0) 7243-2179-0
Fax: +49 (0) 7243-2179-99
E-Mail: michael.weiler@promatis.de
Internet: <http://www.promatis.de>
<http://www.horus.biz>