

Vollständig generisches DWH für kleine und mittelständische Unternehmen

Marc Werner
Freiberufler
Berlin

Schlüsselworte:

Wirtschaftlichkeit, Kostenreduzierung, Metadaten,
Core Data Warehouse, Slowly Changing Dimensions (SCD I - III) ,
Oracle Warehouse Builder, Codegenerierung von: DDL, DML, OWB – Mappings,
Navision, Microsoft Axapta, Oracle eBusiness Suite,
Tcl, OMBPLUS, SQL, PL/SQL,
Change Data Capture (CDC)

Einleitung

Data Warehouse (DWH) - Systeme sind groß angelegte Projekte mit sehr hohem Analyse, Design - und Implementationsaufwand.

Aufwende von **1 bis 2 Mann - Jahren** bis zur Fertigstellung der ersten beiden Ausbaustufen sind keine Seltenheit und gestatten es in der Regel nur großen Unternehmen diese Vorhaben umzusetzen.

In dem hier life demonstrierten Framework zur Generierung/Befüllung der zentralen, technischen Schicht (**Core Layer**: normalisiertes Datenmodell, inkl. künstlicher Schlüssel und verschiedener Historisierungsverfahren) eines Data Warehouse (DWH) ist mittels konsequenter Metadatenanreicherung -und Auswertung, Codegeneratoren und einigen vereinfachenden Designmaßnahmen eine Inbetriebnahme nach **1-2 Mann - Monaten** unabhängig von der Anzahl der anzubindenden Quell – Tabellen erreicht.

Auch im laufenden Betrieb entstehen durch diesen Ansatz - bei den meisten der im DWH nachzuziehenden Änderungen (durch Datenmodelländerungen der OLTP – Systeme: Navision, Axapta, SAP, Oracle eBusiness Suite, etc. aus denen geladen wird) - lediglich marginale Aufwende durch Einsatz von Codegeneratoren.

Die generierten Datenmodelle des Frameworks bringen für den weiteren Ausbau und strategische Betrachtungen des DWH keinerlei Einschränkungen.

Funktionsumfang

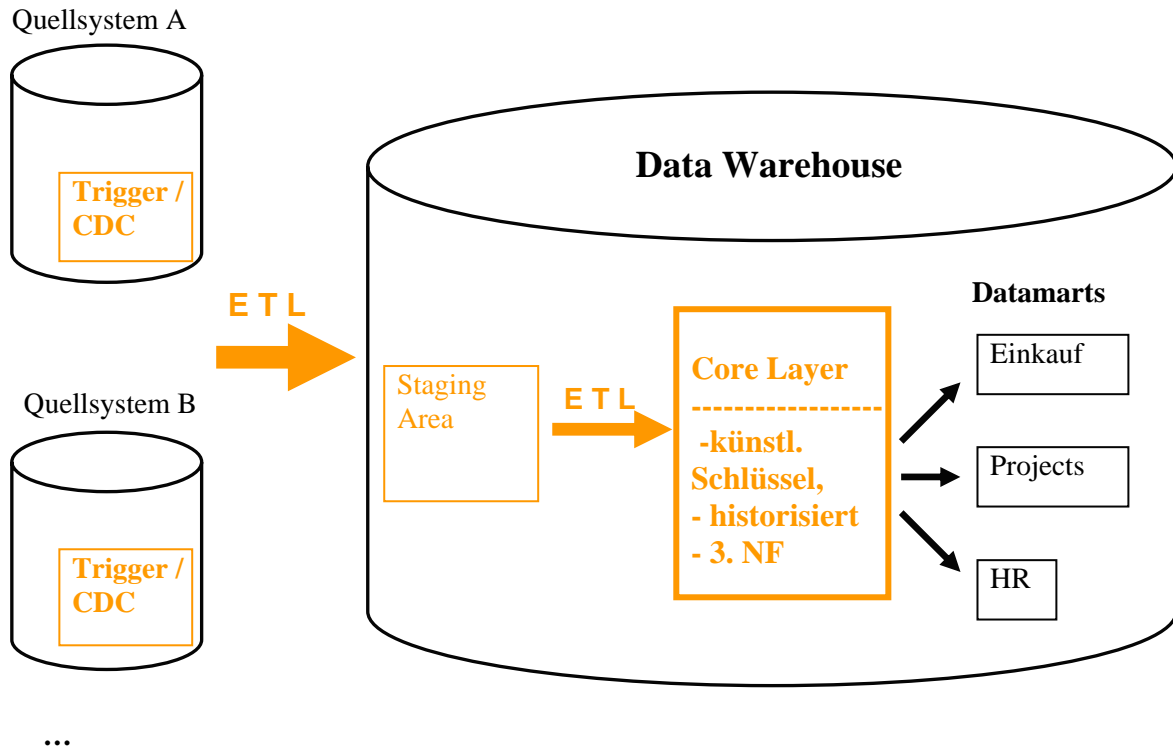


Abb. 1: Orange markiert: Vom Framework generierte Bereiche

Die Generatoren erzeugen die Staging Area, die technische Schicht (Core Layer) eines Data Warehouse und alle ETL – Routinen zum Befüllen.

Die Datenmodelle (Spaltennamen, Datentypen) der Staging Area und Core Layer werden dabei mittels 1 zentralen SQL direkt aus den Quell – Tabellen abgeleitet/generiert.

Die technische Schicht (Core Layer) erhält dabei die gängigen DWH - Features:

- 3. NF
- künstl. Schlüssel
- Historisierungen (gemäß Slowly Changing Dimension I - III)
- Löschflags

Folgender Code wird generiert:

- komplettes DDL (Datenmodelle: Stage und Core Schicht)
- kompletter ETL Code (als OMBPLUS/Tcl -Code, welcher die Mappings im Oracle Warehouse Builder anlegt) :
 1. Quell – Tabellen → Staging Area
 2. Staging Area → Core Layer

- Change Data Capture (CDC) bzw. Trigger - Code auf Quelltabellen zur Ermittlung der ins Data Warehouse nachzuladenen Delta's
- Synchronisationscode (DDL, DML) bei Datenmodelländerungen an den Quelltabellen, die im Data Warehouse nachgezogen werden müssen

Fachliche Datenmodelle (Star/Snowflake) werden nicht generiert.

Für Tabellen, die aufgrund Ihrer Größe und Ihres Wachstums partitioniert werden müssen muss die Partitionierung manuell hinterlegt werden (wird nicht generiert).

Die derzeitigen Codegeneratoren erzeugen Code für Oracle RDBMS Version 8.x – 11.2.

Generierungsprozess

Bevor mit der Generierung des Codes begonnen werden kann muss hinterlegt werden welche Tabellen aus den Quellsystemen (Quell – Tabellen) ans DWH angebunden werden sollen

Zu jeder an das DWH anzubindenden Quell - Tabelle müssen ein paar Mindesteinstellungen (Metadaten) hinterlegt werden:

- eindeutiger Bezeichner: <BaseName> (max. 26 Zeichen), aus dem alle generierten Objektnamen (DWH Tabellen, OWB-Mappings, Trigger etc.) abgeleitet werden
- Welche Historisierung
- Welcher Extraktionsmechanismus aus den Quell DB's (periodischer Abzug der geänderten Datensätze / Jedes Mal Komplettabzug)
- Primary oder Unique Key (Normalerweise nicht manuell einzupflegen, da in Quell-DB angelegt)

Diese Metadatenverwaltung geschieht von Beginn der Analysephase an (1. Treffen mit Fachbereich) direkt auf einer Datenbank in 2 zentralen Steuertabellen (Erstaufnahme in Excel, dann Übertragung via ODBC / External Table in die Steuertabelle in der Datenbank).

Es entsteht kein Wust von Exceltabellen und unstrukturierten Worddokumenten in denen diese entscheidenden Informationen verstreut herumliegen.

In diesem Anforderungsbeispiel:

- alle Spalten der Quell - Tabellen werden übernommen
- keine der Quelltabellen hat einen Namen mit mehr als 26 Zeichen und kein Tabellename kommt mehrfach vor → BaseName = Quelltabellename
- alle Tabellen eines Schemas oder eines Patterns im Namen, nach SCD II zu historisieren
- via Change Data Capture (CDC) Delta-zu-Laden
- alle Quell - Tabellen haben PK/UK

dauert der Aufbau dieser Metadaten gerade einmal 10 Minuten durch Ausführung dieser beiden SQL's, die je nachdem in welcher DB die Metadaten abgelegt werden sollen, via DB-Link die Steuertabellen mit allen an das DWH anzubindenden Tabellen (**SRC_TBLS4DWH**) und den zugehörigen eindeutigen Schlüssel (**DCLG_NATKEYS**) füllen:

```
insert into SRC_TBLS4DWH
  ( basename, db, owner, table_name, LOAD_MECHAN, DELTA_MECHAN, ... )
select table_name,'DB_1', owner, table_name, 'SCD2', 'CDC', ...
  from all_tables@SOURCE_DATABASE_1
  where owner in ( ... )
;
```

```

insert into DCLG_NATKEYS
  ( basename,      col,      dwh_system )
select c1.table_name, column_name , 'DWH_PROTOTYPE_1'
from all_cons_columns c1, all_constraints c2
  where constraint_type = 'P'
     and c1.OWNER in ( ... )
     and c1.constraint_name = c2.constraint_name
;

```

Ab diesem Moment ist die Analysephase für diese erste Menge an Tabellen (1. Ausbaustufe des DWH) abgeschlossen und es werden nur noch Codegeneratoren customized, gestartet und der entstandene Code deployed bzw. Oracle Warehouse Builder (OWB) Metadaten synchronisiert.

Im Vortrag wird im Rahmen einer Demo schrittweise dieser Generierungsprozess gezeigt ausgehend von ein paar simulierten Kundenanforderungen.

Darin kommt sowohl die Erst – Installation inkl. Initial – Befüllung des DWH, als auch die Synchronisation des DWH mit Änderungen auf den Quellsystemen vor.

Verwaltung der Codefragmente / Metadaten

Nach Abschluss des Generierungsprozess der aus diesen zyklischen Einzelschritten besteht:

- Feststellung Release X auf den Quelldatenmodellen
- Generierung und Deployment DWH Datenmodelle (Staging Area, Core Layer)
- Import/Synchronisation des Oracle Warehouse Builder (OWB) - Repository
- Generierung und Deployment OWB Mappings (OMB_PLUS)

sind **ALLE Metadaten und Codefragmente zentral** in einer Steuertabelle in der Datenbank verfügbar.

Hier ein Ausschnitt (SELECT einiger Spalten dieser Tabelle) :

BASENAME	LOAD_MECHAN	DELTA_MECHAN	STA_CREATE	DWH_CREATE	SEQ_SID	SEQ_OID	TRIGG_OR_CDC	OMB_PLUS
AX_MARKUPAUTOLINE	SCD1D	CDC	CREATE TABLE STAGE_SAX_MARKUPAUTOLINE (TABLETABLEID NUMBER(10), TABLERECID NUMBER(20), LINENUM NUMBER(32,16), MARKUPCODE	CREATE TABLE DWH.CAX_MARKUPAUTOLINE (DCLG_SID NUMBER(20) PRIMARY KEY, NOT NULL, DCLG_LOADID NUMBER(10) NOT NULL, DCLG_CREATED DATE NOT NULL,	CREATE SEQUENCE DWH.SCAX_MARKUPAUTOLINE_S START WITH 1 MAXVALUE 9999999999	CREATE SEQUENCE DWH.SCAX_MARKUPAUTOLINE_O START WITH 1 MAXVALUE 9999999999	BEGIN DBMS_CDC_PUBLISH.CREATE_CHANGE_TABLE(owner => 'AX_CHANGES', change_table_name => 'AX_MARKUPAUTOLINE_CT', change_set_name	set OMBCONTINUE_ON_ERROR TRUE OMBDROP MAPPING 'MAX_MARKUPAUTOLINE' set OMBCONTINUE_ON_ERROR FALSE OMBCREATE MAPPING 'MAX_MARKUPAUTOLINE' ADD TABLE OPERATOR 'SAX_MARKUPAUTOLINE' BOUND TO TABLE '.../STAGING/SAX_MARKUPAUTOLINE'
AX_MARKUPTBL	SCD3	CDC	CREATE TABLE STAGE_SAX_MARKUPTBL (VARCHAR2(120), CUSTPOSTING	CREATE TABLE DWH.CAX_MARKUPTBL (DCLG_SID NUMBER(20) PRIMARY KEY, DCLG_OID	CREATE SEQUENCE DWH.SCAX_MARKUPTBL_S START WITH 1 MAXVALUE 9999999999	CREATE SEQUENCE DWH.SCAX_MARKUPTBL_O START WITH 1 MAXVALUE 9999999999	BEGIN DBMS_CDC_PUBLISH.CREATE_CHANGE_TABLE(owner => 'AX_CHANGES',	set OMBCONTINUE_ON_ERROR TRUE OMBDROP MAPPING 'MAX_MARKUPTBL' set OMBCONTINUE_ON_ERROR FALSE OMBCREATE MAPPING
LI_EWB_KONTO	SCD3	FULL	CREATE TABLE STAGE_STA_LI_EWB_KONTO	CREATE TABLE DWH.CORE_LI_EWB_KONTO	CREATE SEQUENCE DWH.SQ_LI_EWB_KONTO	CREATE SEQUENCE DWH.SQ_LI_EWB_KONTO	BEGIN DBMS_CDC_PUBLISH.CREATE_C	set OMBCONTINUE_ON_ERROR TRUE OMBDROP MAPPING
AX_PRICEDISCGROUP	SCD2	CDC	CREATE TABLE STAGE_SAX_PRICEDISCGROUP	CREATE TABLE DWH.CAX_PRICEDISCGROUP	CREATE SEQUENCE DWH.SCAX_PRICEDIS	CREATE SEQUENCE DWH.SCAX_PRICEDIS	BEGIN DBMS_CDC_PUBLISH.CREATE_C	set OMBCONTINUE_ON_ERROR TRUE OMBDROP MAPPING
LO_LAX_ARTK	SCD1	TS	CREATE TABLE STAGE_SLO_LAX_ARTK (ARTKID NUMBER(12), ARTKBZNGID	CREATE TABLE DWH.CLO_LAX_ARTK (_ARTK (DCLG_SID NUMBER(20) PRIMARY KEY	CREATE SEQUENCE DWH.SCLO_LAX_ARTK_S START WITH 1 MAXVALUE	CREATE SEQUENCE DWH.SCLO_LAX_ARTK_O START WITH 1 MAXVALUE	CREATE OR REPLACE TRIGGER "OLTP_XY"."LO_LAX_ARTK_TTS" BEFORE INSERT OR UPDATE ON	set OMBCONTINUE_ON_ERROR TRUE OMBDROP MAPPING 'MLO_LAX_ARTK' set OMBCONTINUE_ON_ERROR FALSE

Abb. 2: Vom Framework generierter und in DB Tabelle abgelegter Code

Damit kann der Code jedes generierten DWH – System flexibel gesichert und in andere Ziel Datenbanken deployed werden.

Legende:

Spalte	Bedeutung	Erläuterungen
BASENAME	DWH – weit eindeutiger Bezeichner für die angebundene Quell – Tabelle	Basis aller generierten Objekt-namen im DWH
LOAD_MECHAN	Historisierung bzw. Löschflag: Ja/Nein , Falls Ja verschiedene Verfahren (Historisierungen nach Kimball ...)	Bsp.: SCD3 ist eine Kimball SCD II Historisierung mit zusätzlichem Löschflag für Datensätze, die im Quellsystem gelöscht wurden
DELTA_MECHAN	CDC: Change Data Capture (Deletes / Inserts / Updates auf der Quell – Tabelle werden protokolliert und übertragen TS: Es soll ein Trigger, der einen Zeitstempel auf der Quelltablettelle setzt für den Delta-Abzug verwendet werden FULL: Es werden jedes Mal alle Datensätze übertragen und abgeglichen	
STA_CREATE	DDL für die zugehörige Staging Tabelle	
DWH_CREATE	DDL für die zugehörige Core – Layer Tabelle	
SEQ_*	DDL für die Künstlichen Schlüssel	
TRIGG_OR_CDC	Je nach Wahl in Spalte DELTA_MECHAN steht hier der Code für den Trigger oder CDC – Installation auf der Quell – Tabelle	
OMB_PLUS	Tcl - Code, der die OWB – Mappings anlegt	

Kontaktadresse:

Marc Werner
Freiberufler
Lychener Str. 10
D-10437 Berlin

Telefon: +49 (0) 1721068289
E-Mail: marc-werner@gmx.net
Internet: <http://www.gulp.de/profil/mw73.html>