

Zusammenfassung W.WIINM32.H10

# Data Warehousing



19.12.2010, Michael Baumli

## Einleitung

Diese Zusammenfassung ist anhand des Zeitplanes (Herbstsemester 2010) strukturiert. Das Modul gliedert sich in zwei Kurse. Kurs 1 behandelt die Data-Warehouse-Technologien, im Kurs 2 beschäftigt sich mit dem Thema Data-Warehouse-Projekt. Auf die Gruppenarbeiten – welche auch prüfungsrelevant sind – wurde hier nicht eingegangen. Die Ziele des Modules Data Warehousing:

- Grundbegriffe, Mechanismen und Komponenten eines Data-Warehouse Systems.
- Aktualisierung des Data Warehouses
- Strukturen und Operationen des multidimensionalen Datenmodells
- Rolle der Metadaten
- Massnahmen zur Sicherung der Datenqualität
- Organisation und Management eines Daten Warehousing Projektes
- Strategien zur Evaluation von Hardware und Software
- Vorgehen beim Aufbau eines Data-Warehouse-Systems in einem konkreten Projekt

## Bibliographie

- A. Bauer, H. Günzel (eds): **Data-Warehouse-Systeme. Architektur, Entwicklung, Anwendung.** dpunkt.verlag, Heidelberg, Dezember 2000
- R. Gabriel, P. Gluchowski, A. Pastwa: **Data Warehouse and Data Mining.** W3L-Verlag, 2009
- U. Leser, F. Naumann: **Informationsintegration.** dpunkt.verlag, Heidelberg, 2007
- P. Westerman: **Data Warehousing. Using the Wal-Mart Model.** Morgan Kaufmann Publishers, San Francisco, CA, 2001
- R. Kimball, L. Reeves, M. Ross, W. Thornthwaite: **The Data Warehouse Lifecycle Toolkit.** John Wiley & Sons, New York, NY, 1998
- W. Lehner: **Datenbanktechnologie für Data-Warehouse-Systeme.** dpunkt.verlag, Heidelberg, 2002
- W.H. Inmon: **Building the Data Warehouse.** 4. Auflage, Wiley Publishing, 2005

## Inhaltsverzeichnis

Kurs 1: Data-Warehouse-Technologien.....	5
1 Kapitel 1: Grundlagen von Data Warehousing .....	5
1.1 Motivation.....	5
1.1.1 Data, Data everywhere.....	5
1.1.2 Von Daten zu Informationen .....	5
1.1.3 Datenbanksysteme und ihre Rolle.....	5
1.1.4 Neue Anforderungen an Data Management.....	5
1.1.5 Evolution der Informationssysteme .....	6
1.2 Definitionen.....	6
1.2.1 Data Warehouse .....	6
1.2.2 OLAP (Online Analytical Processing).....	6
1.2.3 Data Mining .....	6
1.2.4 Datenintegration .....	7
1.2.5 Historisierung.....	7
1.2.6 Data-Warehouse-System.....	7
1.2.7 Data Warehousing .....	7
1.2.8 Datenübernahme aus operativen Systemen.....	8
1.3 Abgrenzung von Data Warehousing .....	8
1.3.1 Managementinformationssysteme (MIS) (anno 1970, 1980) .....	8

1.3.2	Die Pyramide der Informationssysteme .....	8
1.3.3	Business Intelligence (BI) .....	8
1.3.4	Datenintegration (Informationsintegration) .....	9
1.3.5	Architektur monolithischer Datenbanksysteme.....	10
1.3.6	Zwei Arten von Informationsintegration .....	10
1.3.7	Virtuelle integrierte Informationssysteme .....	11
1.4	Anwendungsbereiche von Data Warehousing.....	12
1.4.1	Verkauf .....	12
1.4.2	Marketing .....	12
2	Referenzarchitektur eines Data-Warehouse-Systems.....	12
2.1	Der Begriff der Referenzarchitektur.....	12
2.2	Komponenten der Referenzarchitektur .....	13
2.2.1	Datenquellen .....	13
2.2.2	Datenquellen - Qualitätsanforderungen .....	14
2.2.3	Datenquellen - Klassifikation .....	14
2.2.4	Extraktionskomponente .....	14
2.2.5	Extraktionskomponente – Zeitpunkt der Extraktion .....	14
2.2.6	Extraktionskomponente - Realisierung .....	14
2.2.7	Monitoring Komponente .....	15
2.2.8	Monitoring Komponente – Techniken für die Realisierung .....	15
2.2.9	Arbeitsbereich (Staging Area).....	16
2.2.10	Transformationskomponente.....	16
2.2.11	Basisdatenbank.....	16
2.2.12	Ladekomponente .....	17
2.2.13	Data Warehouse Manager .....	17
2.2.14	Data Warehouse .....	17
2.2.15	Metadaten Manager & Metadaten Repository.....	17
2.2.16	Analyse.....	18
2.3	Prozesse der Referenzarchitektur.....	18
2.4	Varianten der Referenzarchitektur .....	18
2.4.1	„Ad-hoc“ Data Warehouse (quick'n'dirty).....	19
2.4.2	Zentrales DWH mit virtuellen Data Marts .....	19
2.4.3	Zentrales DWH mit persistenten Data Marts .....	19
2.4.4	Föderierte DWH Architektur (virtuelle Integration) .....	19
2.4.5	Kein zentrales DWH .....	19
3	Aufbau eines Data-Warehouse-Systems .....	20
3.1	Data-Warehouse-Strategie .....	20
3.1.1	Strategie.....	20
3.1.2	Grundstrategien in der IT .....	21
3.1.3	Data-Warehouse-Strategie .....	22
3.2	Reifegradmodell.....	22
3.2.1	Anforderungen an BI/DWH- Reifegradmodelle.....	22
3.2.2	Business Intelligence Maturity Model (biMM®) .....	22
3.2.3	biMM Vorgehen.....	23
3.2.4	biMM Maturitätsstufen .....	23
3.2.5	Entwicklungen einer DWH-Lösung .....	24
3.3	Ableitung der Data-Warehouse-Architektur.....	24
3.3.1	Definition IT Architektur .....	24

3.3.2 Architektur Framework .....	24
3.3.3 Data Warehouse Framework.....	25
<b>4 Multidimensionale Modellierung .....</b>	<b>26</b>
<b>4.1 Datenstrukturen und Operationen im multidimensionalen Datenmodell (MDDM) .....</b>	<b>26</b>
4.1.1 Datenmodell .....	26
4.1.2 Multidimensionales Datenmodell .....	26
4.1.3 Kennzahlen .....	27
4.1.4 Dimensionen und Klassifikationshierarchien .....	27
4.1.5 Klassifikationshierarchie .....	27
4.1.6 Würfel .....	28
4.1.7 Operationen.....	28
<b>4.2 Speicherung und Verwaltung von multidimensionalen Daten .....</b>	<b>29</b>
4.2.1 Datenbankentwurf in Datenbanksystemen .....	29
4.2.2 Datenbankentwurf in Data-Warehouse-Systemen .....	30
4.2.3 Beispiel ME/R Modell .....	30
4.2.4 ROLAP vs. MOLAP .....	30
4.2.5 Grundlagen von ROLAP.....	31
4.2.6 ROLAP – Snowflake Schema .....	31
4.2.7 ROLAP – Sternschema .....	32
4.2.8 Entwicklung eines Sternschema .....	33
4.2.9 Evaluation Sternschema .....	34
4.2.10 Evaluation Sternschema .....	34
4.2.11 SQL 2003 und OLAP-Erweiterung .....	34
<b>Kurs 2: Das Data-Warehouse-Projekt.....</b>	<b>36</b>
<b>5 Kapitel 1: Das Data-Warehouse-Projekt.....</b>	<b>36</b>
<b>5.1 DWH Projekt- und Changemanagement.....</b>	<b>36</b>
5.1.1 Projekt, Programm, Projektportfolio.....	36
5.1.2 Projektvorgehensmodelle .....	36
5.1.3 Projektorganisation und Rollen .....	38
5.1.4 Projektrollen und Themenbereiche.....	39
5.1.5 Organizational Change Management (OCM) .....	39
5.1.6 Konfliktmanagement .....	39
<b>5.2 DWH Business Case.....</b>	<b>40</b>
<b>5.3 Softwareauswahl.....</b>	<b>40</b>
5.3.1 Klassifikation der Produkte.....	40
5.3.2 Projektvorgehen zur Softwareauswahl .....	41
<b>5.4 Herausforderungen und Erfolgsfaktoren.....</b>	<b>41</b>
5.4.1 Herausforderungen im Projektmanagement .....	41
5.4.2 Herausforderungen bei DWH-Projekten .....	41
5.4.3 Erfolgsfaktoren in DWH-Projekten .....	43

# Kurs 1: Data-Warehouse-Technologien

## 1 Kapitel 1: Grundlagen von Data Warehousing

### 1.1 Motivation

#### 1.1.1 Data, Data everywhere...

Die Datenmenge wächst immens. Sie verdoppelt sich innerhalb von 12 bis 18 Monaten! Ebay zum Beispiel hat heute mehrere hundert Datenbanken mit 4-stelligen Terabytes von Daten, mit ca. 750'000 Anfragen pro Tag. Wal-Mart sammelt Daten über jeden verkauften Artikel, jeder Filiale, jeden Tag sodass jeden Tag ca. 1 Billion Tupel geändert werden.

#### 1.1.2 Von Daten zu Informationen

Informationen werden aus Daten gewonnen. Auf Basis solcher Daten können Unternehmen und Organisationen Entscheidungen treffen, Prognosen entwickeln etc. Unternehmen stellen sich z.B. folgende Fragen:

- Wie viele Flaschen Cola wurden letzten Monat verkauft?
- Wie hat sich der Verkauf von NZZ im letzten Jahr entwickelt?
- Wer sind unsere Top Kunden?

Die Herausforderung zur Beantwortung dieser Fragen liegt darin die Daten welche aus mehreren externen Quellen kommen so aufzubereiten, dass die gewünschten Informationen daraus gewonnen werden können. Die Lösung ist ein zentrales Datawarehouse.



**Information:** Wie viele abgeschlossene Bestellungen haben wir jeweils im Monat vor Weihnachten, aufgeschlüsselt nach Produktgruppen und Promotion?

**Daten:** Daten über Kunden, Produkte, Bestellungen in verschiedenen Datenbanken verteilt in mehreren Regionen.

#### 1.1.3 Datenbanksysteme und ihre Rolle

Datenbanksysteme bringen Ordnung in die Informationsflut. Es soll alles Wissen dieser Welt elektronisch **speichern** und **jederzeit an jedem Ort jedem autorisierten Benutzer** zur Verfügung stehen. Wichtige Begriffe zu Datenbanksystemen:

- **Datenmodell:** Zeigt einen Ausschnitt aus der realen Welt. Während dem Datenbankentwurf wird die Struktur der Daten anhand der Konstrukte des Datenmodells erstellt.
- **Persistenz:** Die Datenbank ist dauerhaft verfügbar.
- **Integritätsbedingung:** Die Datenbank weisst eine hohe „Datenqualität“ auf.
- **Anfragesprache:** Ermöglicht effiziente Handhabung der Daten
- **Skalierbarkeit:** Antwortzeiten bleiben gut, unabhängig von der Anzahl User

#### 1.1.4 Neue Anforderungen an Data Management

Heutzutage sind Daten auf unterschiedlichsten Quellen verteilt. Diese Daten müssen entsprechend aufbereitet werden. Es ist schwierig anhand „transaktionsorientierten“ Datenbankanwendungen Auswertungen und Analysen zu erstellen. Dafür ist die „klassische“ Datenbanktechnolo-

gie nicht ausreichend. → Es wird eine Technologie benötigt, welche auf analytische Anwendungen zugeschnitten ist.

### 1.1.5 Evolution der Informationssysteme

<b>1960 Flat Files</b>	Flaches File mit Nummerierung (Die Struktur muss man kennen um das File zu „verstehen“)
<b>1980 Relationale Datenbanksysteme</b>	Ermöglicht die Korrektheit der Daten auch bei Mehrbenutzerbetrieb.
<b>1990 Informationsintegration</b>	Anfrage von Daten aus unterschiedlichen und autonomen und heterogenen Datenquellen. Daten als Basis zur Entscheidungsunterstützung.
<b>1995 Data Warehousing und Business Intelligence</b>	Trennung von analytischen und operativen Systemen.

## 1.2 Definitionen

### 1.2.1 Data Warehouse

Erste Data Warehouse Definition von „Inmon, 1996“ (Mr. Datawarehouse):

A Data Warehouse is a subject oriented, integrated, non-volatile and time variant collection of data in support of management's decisions.

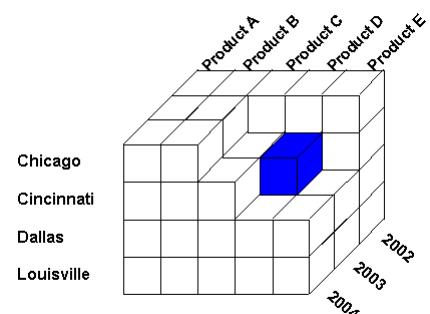
- **Subject Oriented** (Fachorientierung): Zweck der Datenbasis ist nicht die Erfüllung einer Aufgabe sondern die Modellierung eines spezifischen Analyseziels. (Die Datenanalyse kann mit OLAP oder Data Mining Methoden erfolgen).
- **Integrated** (integrierte Datenbasis): Die Datenverarbeitung findet auf integrierte Daten aus mehreren Datenbanken statt.
- **Non-volatile** (Nicht-volatile Datenbasis): Das Datawarehouse ist stabil, kein Update oder Delete der Daten.
- **Time variant** (Historische): Daten werden über einen längeren Zeitraum gehalten damit Vergleiche über die Zeit mögliche sind.

### 1.2.2 OLAP (Online Analytical Processing)

Als OLAP wird die **interaktive Datenanalyse** und die **Navigation** in Daten bezeichnet, basierend auf einer **multidimensionale Modellierung** der Daten.

Berichte die aus OLAP hervorgehen, enthalten verdichtete Daten in Form von Kennzahlen. Navigationsoperationen wie „drill down“ und „roll up“ erlauben Daten detaillierter bzw. auf einem höheren Level zu untersuchen. Gruppierungen und Berechnungen sind ebenfalls möglich.

Der OLAP Würfel rechts zeigt beispielsweise: Den Umsatz in Cincinnati für Produkt D im Jahr 2003.



### 1.2.3 Data Mining

Data Mining Funktionen erweitern Analysemethoden durch den Einsatz von **Statistiktechniken** und maschinellen **Lerntechniken**, sodass ohne eine exakte Anfrageformulierung (wie bei OLAP nötig) bisher unentdeckte Zusammenhänge aus den Daten aufgedeckt werden. (Hypothesen).

Bei Data Mining geht es um Mustererkennung, Zusammenhänge, Trends, wobei eine Methode das Durchsuchen grosser Datenbestände ist. Data Mining will **Assoziationsregeln** des Datenbestandes erkennen. Ähnliche Objekte des Datenbestandes **segmentieren** (Gruppen, Clustering). Und Data Mining will neue Elemente automatisch bestimmen und **klassifizieren**.

#### 1.2.4 Datenintegration

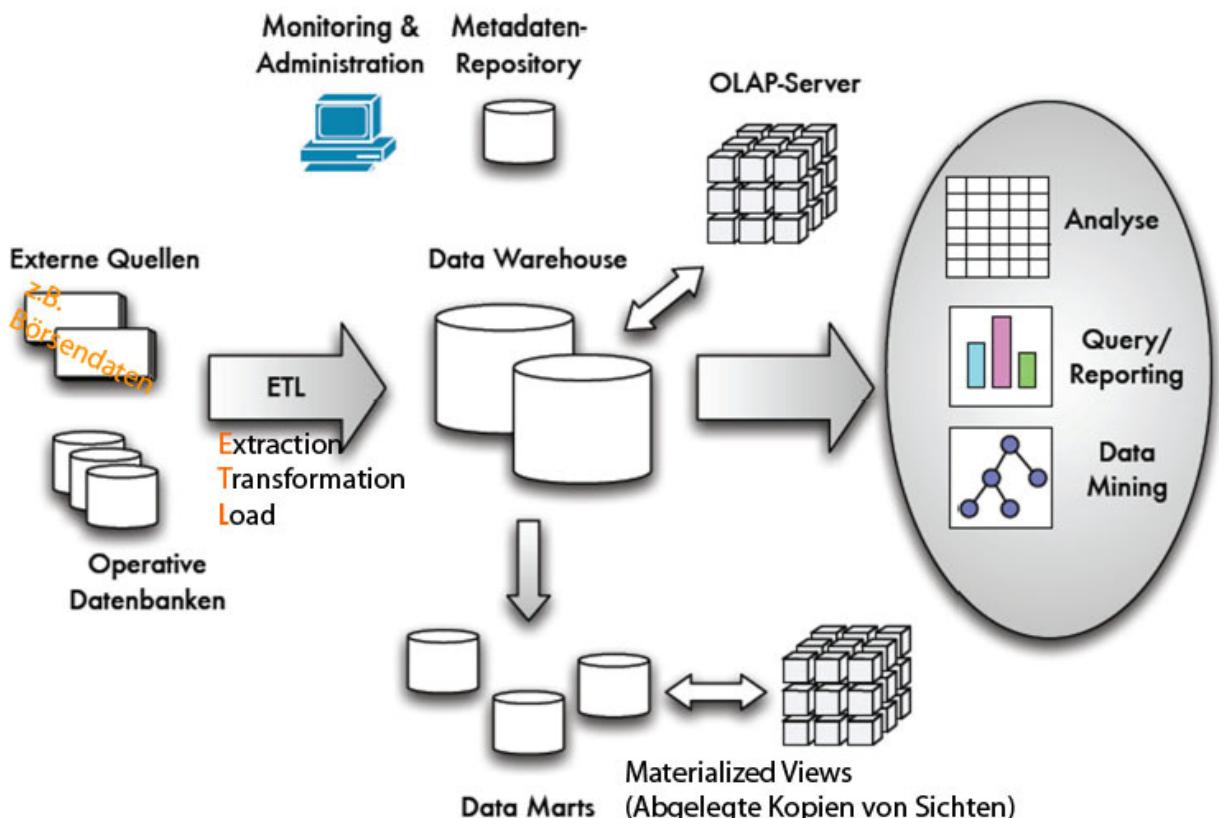
Bei der Datenintegration in einem Data Warehouse müssen alle **relevanten Daten** berücksichtigt werden. **Quelldaten** sind auf viele Datenquellen **verteilt** (CRM, ERP, Applikationen etc..). Es gibt **interne** und **externe Datenquellen**. Meist sind diese **heterogen** (Uneinheitlich).

#### 1.2.5 Historisierung

Ein Datawarehouse wächst mit seinem Alter, da **alte Daten nicht gelöscht** werden. Dies aufgrund dessen, da Auswertungen und Analysen oft die zeitliche Entwicklung aufzeigen soll. Alle Daten im DWH (Datawarehouse) werden daher mit Zeitinformationen versehen.

#### 1.2.6 Data-Warehouse-System

Ein **Data-Warehouse-System** umfasst alle notwendigen Komponenten für die Datenbeschaffung, die Integration der Daten und deren Speicherung im Data Warehouse.



#### 1.2.7 Data Warehousing

**Data Warehousing** umfasst alle Schritte angefangen beim Datenbeschaffungsprozess bis zur Speicherung der Daten im Datawarehouse.

Data Warehousing ist also der Prozess wie man Daten beschafft, integriert und speichert. Das Data Warehouse hingegen ist die Datenbank die alle Daten enthält.

## 1.2.8 Datenübernahme aus operativen Systemen

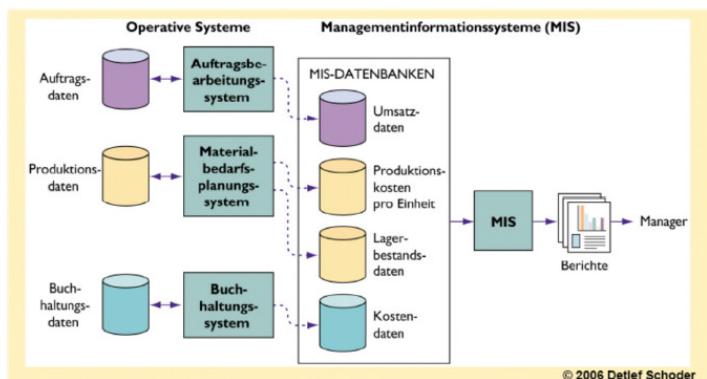


## 1.3 Abgrenzung von Data Warehousing

### 1.3.1 Managementinformationssysteme (MIS) (anno 1970, 1980)

Synonyme für MIS sind: Executive Information System, Führungsinformationssystem, Entscheidungsunterstützungsinformationssystem, Decision Support System. Sie alle haben das gleiche Ziel: **Entscheidungsunterstützung**.

Der Durchbruch von MIS scheiterte. Es fehlte an schnellen Kommunikationstechnologien, GUI's, Leistungsfähiger Datenspeicher etc. → Data Warehouse Systeme haben in den 90ern den Durchbruch geschafft.



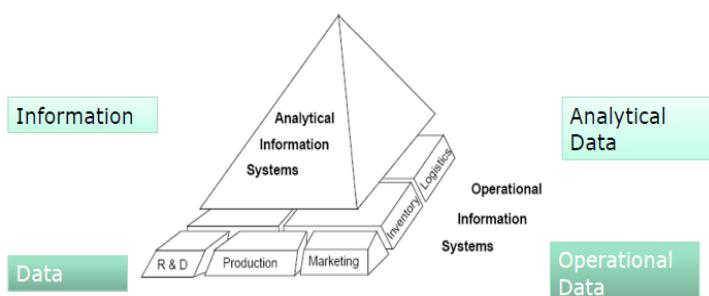
### 1.3.2 Die Pyramide der Informationssysteme

#### Operationale Daten

- Real und aktuell
- Im tägl. Geschäft gewonnen/bearbeitet
- Werden öfters geändert
- Datenkonsistenz/Effizienz ist ein Muss.

#### Analytische Daten

- Historisch, werden nicht überschrieben
- Oftmals verdichtet
- Anfragen sind komplex
- Qualitätsanforderungen sind hoch.



### 1.3.3 Business Intelligence (BI)

**Business Intelligence** ist ein integrierter unternehmensspezifischer IT-basierter Gesamtansatz zur betrieblichen Entscheidungsunterstützung.

Heute 15 Jahre nach dem Entstehen von Data Warehousing verwendet man die zwei Begriffe „Data Warehousing“ und „Business Intelligence“ gleich. Man kann sagen, dass Data Warehousing ein Teil von Business Intelligence ist. Data Warehousing ist sozusagen das **Backend** um Business Intelligence überhaupt zu ermöglichen. OLAP kann man klar an Data Warehousing zuordnen. Data Mining hingegen gehört ganz klar zu Business Intelligence.

#### 1.3.4 Datenintegration (Informationsintegration)

In der Informationsintegration führt man Daten und Inhalte verschiedener Quellen zusammen zu einer einheitlichen Informationsmenge. Die Informationsintegration soll eine **korrekte, vollständige** und **effiziente** Zusammenführung von Daten sein, welche aus **heterogenen** Quellen gewonnen werden. Es entsteht eine einheitliche **strukturierte** Informationsmenge zur **effektiven Interpretation** durch Nutzer und Anwendungen. Informationsintegration bringt 3 Herausforderungen:

- Physische Verteilung der Daten (verschiedene Orte)
- Autonomie der Datenquellen (Rechte, wer was sehen kann)
- Heterogenität zwischen den Datenquellen und dem Integrationssystem (HTML, SQL...)

Diese 3 Herausforderungen werden hier genauer erläutert:

##### ▪ **Verteilung der Daten**

**Physische Verteilung:** Die Daten liegen auf unterschiedlichen Systemen, welche miteinander vernetzt sind. D.h. die Orte müssen z.B. via URL lokalisiert werden. Zusätzlich sind die Daten in verschiedenen Schemata, also Heterogen bezüglich Anfragen etc.

**Logische Verteilung:** Es ist möglich, dass gleiche Daten an verschiedenen Orten liegen, z.B. kann der Name einer Person in verschiedenen DB's vorhanden sein. Es muss also mit unkontrollierter Redundanz gerechnet werden. Um die Konsistenz zu gewährleisten muss man Duplikate und Widersprüche erkennen.

##### ▪ **Autonomie der Datenquellen**

Die Autonomie der Datenquellen ist die Freiheit der Datenquellen unabhängig über ihre Daten und Zugriffsmöglichkeiten zu entscheiden. Es gibt 3 Arten von Autonomie:

- Designerautonomie: Aspekte des Datenbankentwurfs (Objektorientiert, relational...)
- Schnittstellenautonomie: Zugriffsmöglichkeiten auf Daten (Anfragesprache, Schnittstellen)
- Zugriffsautonomie: Wer auf welche Daten zugreifen kann. (Autorsierung, Datensicherheit)

##### ▪ **Heterogenität**

Systeme gelten als Heterogen, wenn Sie nicht exakt die gleichen Methoden, Modelle und Strukturen zum Zugriff auf die Daten anbieten. Sie entsteht durch unterschiedliche Anforderungen und durch die zeitliche Entwicklung. Außerdem kann man sagen, dass der Grad der Heterogenität nimmt mit dem Grad der Autonomie zu nimmt.

→ **Die Heterogenität zu überbrücken ist die Kernaufgabe der Informationsintegration!**

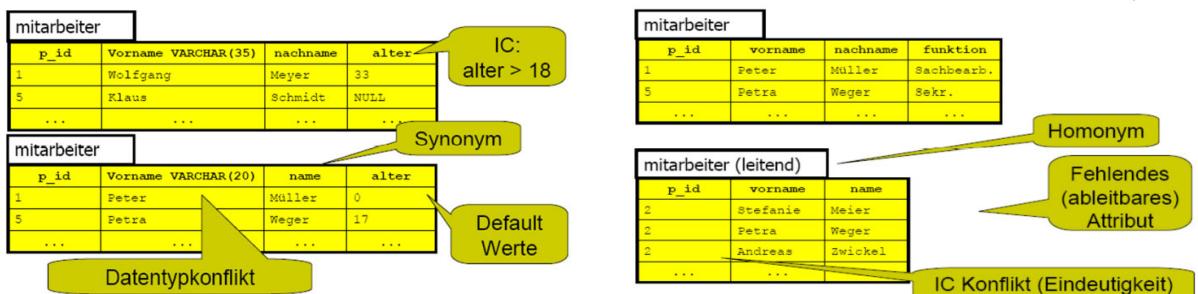
Es gibt 5 verschiedenen Arten von Heterogenität:

- **Technische Heterogenität:** Umfasst alle Möglichkeiten des Zugriffs auf die Daten. Dieser Aspekt hat einen hohen Stellenwert in der Informationsintegration.
  1. Anfragemöglichkeit: Anfragesprache (z.B. SQL), Funktionen, Formulare
  2. Austauschformat: Binärdaten, XML, HTML, Tabellen
  3. Kommunikationsprotokoll: http, JDBC, SOAP

- **Syntaktische Heterogenität:** Beschreibt die Unterschiede in der Darstellung gleicher Sachverhalte. Dies können unterschiedliche binäre Zahlenformate, Zeichencodierung (Unicode, ASCII) oder Trennzeichen in Textformaten sein. Diese Probleme können anhand von Transformationen leicht überwunden werden.
- **Heterogenität auf Datenmodellebene:** Gemeint ist hiermit, dass jede Datenquelle ein anderes Datenmodell verwenden (z.B. relational, objektorientiert, flat file, XML...). Zur Überwindung verwendet man „reichere“ Modelle als Metamodelle.
- **Strukturelle Heterogenität:** Auch wenn ein gleiches Datenmodell verwendet wird, können verschiedene Schemata auftreten. Dies kommt vor, weil der Datenbankentwurf recht vielfältig abläuft und Optimierungsschritte durchläuft. Das untenstehende Beispiel zeigt, dass unterschiedliche Elemente eines Datenmodells verwendet werden, um denselben Sachverhalt zu modellieren:

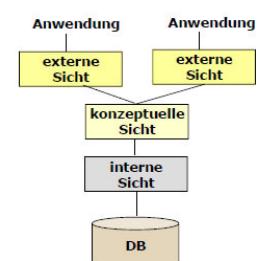
spielfilm (id, titel, laenge) und dokumentarfilm (id, titel , laenge)  
 film (id, titel, laenge, spielfilm, doku)  
 film (id, titel, laenge, typ)

- **Semantische Heterogenität:** Werte alleine haben keine eindeutige Bedeutung, z.B. nur die Zahl „1979“ ist keine Information. Erst durch Attribute kann die Information interpretiert werden. Nun können aufgrund von Synonymen (unterschiedlicher Name aber gleiche Bedeutung) und Homonymen (Bank) gewisse Konflikte entstehen. Dies ist schwierig herauszufinden und meist nur durch den Kontext möglich.



### 1.3.5 Architektur monolithischer Datenbanksysteme

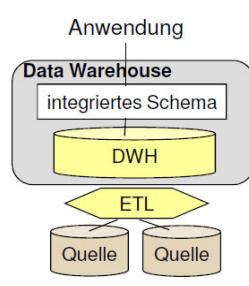
Die 3-Schichten Architektur ist die Basisarchitektur für Datenbanksysteme. Die Trennung zwischen interner und konzeptioneller Sicht gewährleistet die physische Datenunabhängigkeit. (Wo befindet sich das Tupel?). Die Trennung zwischen externer und konzeptioneller Sicht gewährleistet die logische Datenunabhängigkeit. (Datenmodell)



### 1.3.6 Zwei Arten von Informationsintegration

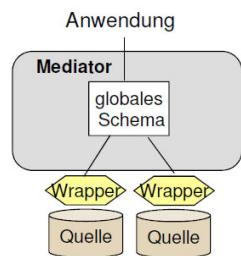
- **Materialisierte Integration**

Integrierte Daten werden zentral abgespeichert (Data Warehouse). Somit wird die Datenqualität während des ETL (Extraction Transformation Loading) Prozesses erhöht.



- **Virtuelle Integration**

Daten werden in kleinen Ausschnitten während der Anfragebearbeitung aus den Quellsystemen herausgezogen und nach der Anfrage wieder verworfen. Bei jeder Anfrage findet somit eine Integration statt.



	<b>Materialisiert</b>	<b>Virtuell</b>
<b>Aktualität</b>	Je nach Aktualisierungsstrategie	Immer aktuell
<b>Antwortzeit</b>	Niedrig, direkt auf dem DHW, Optimierung möglich	i.d.R. ein Nachteil
<b>Komplexität</b>	Hoch bei der Datenaufbereitung (ETL)	Hoch bei der Aufteilung der Anfragen an die Datenquellen und das spätere Zusammenführen
<b>Speicherbedarf</b>	Hoch	Niedrig: Nur für Metadaten
<b>Datenqualität</b>	Wird unterstützt	Je nachdem falls die Quellen die Qualität bieten.

### 1.3.7 Virtuelle integrierte Informationssysteme

- **Multidatenbanken**

Multidatenbanken integrieren mehrere heterogene Datenquellen auf der Anfrageebene. Die autonomen Datenbanken sind lose miteinander gekoppelt und der Zugriff erfolgt via einer Multidatenbanksprache. Jede Datenquelle unterhält ihr eigenes Exportschema.

- **Föderierte Datenbanken**

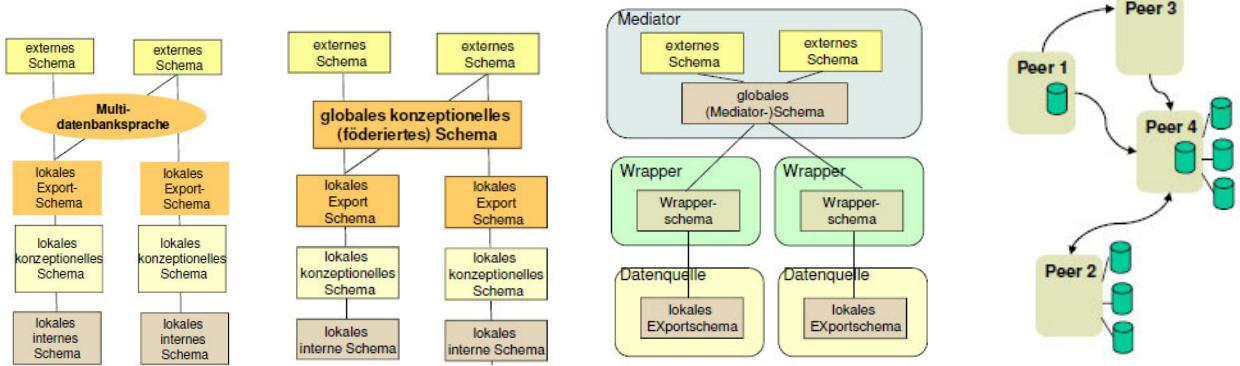
Föderierte Datenbanken integrieren mehrere heterogene Datenquellen auf Schemaebene. Es existiert ein konzeptionelles Schema d.h. Anfragen werden deklarativ an ein globales Schema gestellt.

- **Mediatorenbasierte Systeme**

Es ist eine Verallgemeinerung der anderen Ansätze. Sogenannte Wrappers sind zuständig für den Zugriff auf die einzelnen Datenquellen. Diese überwinden die Datenmodell Heterogenität. Die Mediatoren sind zuständig für die strukturelle und semantische Integration der Daten (Diese enthalten Wissen über die Datenquellen, Transformationen, Integration).

- **Peer-Daten-Management**

Peer steht für „auf der gleichen Ebene“. Das Peer-Daten-Management löst die Unterscheidung zwischen Datenquellen und integriertem System auf. Jedes System darf Anfragen an ein anderes stellen. Peer ist somit Datenquelle (gibt Antwort aus eigenen lokalen Datenbeständen) und Mediator (benutzt andere Peer's um eine Antwort zu finden). **Vorteil** ist das einfache Einfügen einer neuen Datenquelle. **Nachteilig** ist die Komplexität der Anfragebearbeitung.



## 1.4 Anwendungsbereiche von Data Warehousing

Man setzt Data Warehouse Systeme ein zur **Aufbereitung** einer homogenen, integrierten und historisierten Datenbasis welche eine effiziente **Analyse** von Daten ermöglicht. Die Anwendungsbereiche sind vielfältig:

- Betriebswirtschaftliche Anwendungsbereiche
- Wissenschaftliche Anwendungsbereiche
- Technische Anwendungsbereiche

### 1.4.1 Verkauf

Im Verkauf dient das DWH dazu dem Management wichtige Veränderungen aufzuzeigen. Das DWH fungiert somit als Kernstück. Es werden Interne und Externe Daten benötigt. Die Aufbereitung kann grafisch erfolgen (Diagramme), zudem wird oft ein Dashboard oder Cockpit verwendet.

**Daten:** Verkaufsdaten von Produkten, in verschiedenen Filialen, an Kunden

**Ziel:** Gewinnmaximierung

**Verwendung der Analysen für:** Optimierung der Verkaufszahlen, Identifikation von Kassenschlagnern, Erkennen von Produkttrends etc.

### 1.4.2 Marketing

**Daten:** Daten über aktuelle, potenzielle, ehemalige Kunden, Kampagnen, Rückläufe

**Ziel:** Datenbereitstellung für das Customer Relationship Management (CRM)

**Verwendung der Analysen für:** Bewerbung potenzieller Kunden, Kundensegmentierung Kampagnenplanung etc.

*Hinweis Autor: Im Foliensatz werden noch weitere Anwendungsbereiche vorgestellt: Lagerhaltung, Finanzdienstleister, Telekommunikation, Klimatologie und Meteorologie. Auf diese wird hier nicht genauer eingegangen.*

## 2 Referenzarchitektur eines Data-Warehouse-Systems

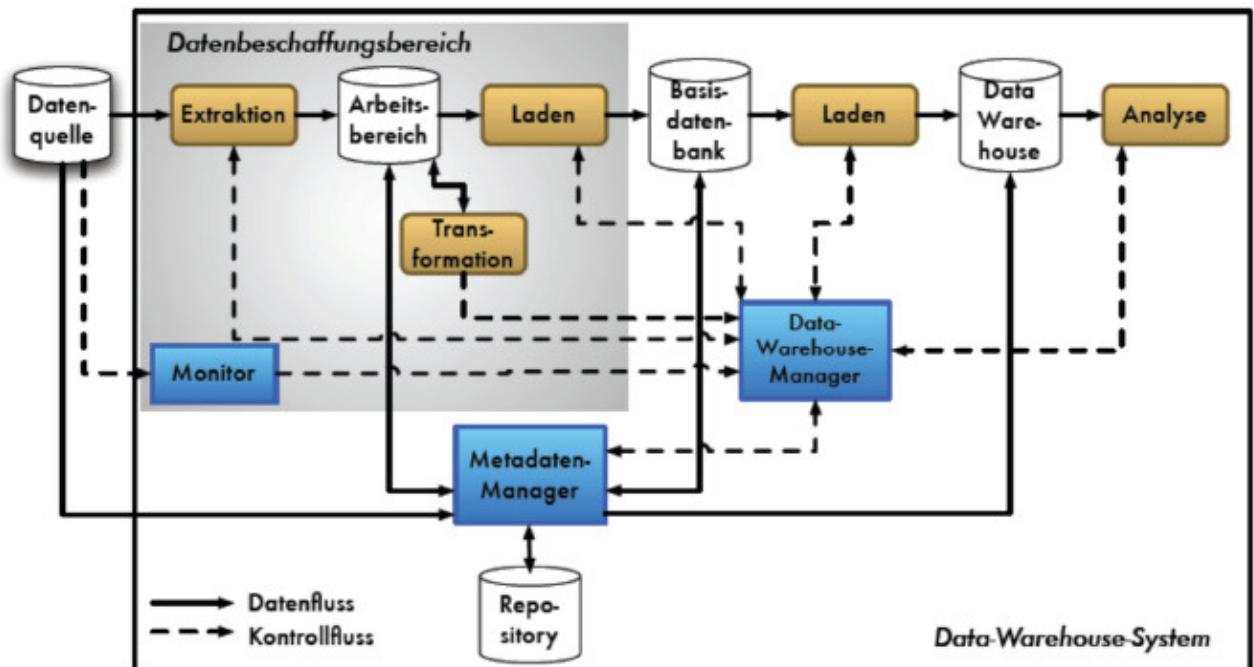
### 2.1 Der Begriff der Referenzarchitektur

Eine Referenzarchitektur ist eine modular aufgebaute Architektur. Ein Modularer Aufbau empfiehlt sich, da allfällige Ausfälle auf Module beschränkt sind, eine gute Skalierbarkeit aufweist und es möglich ist weitere Datenquellen aufzunehmen. Eine Referenzarchitektur ist auch noch in einigen Jahren gültig. Sie bildet die Basis für die Definition eines konkreten Data-Warehouse-Systems. Außerdem verhilft Sie zu einem gemeinsamen Verständnis, sofern andere diese Referenzarchitektur kennen.

## Anforderungen an die Referenzarchitektur eines DWH-Systems:

- Dauerhafte Bereitstellung aktueller sowie historisierter Daten
- Automatisierung der DWH Prozesse
- Unabhängigkeit zwischen Datenquelle und Analysesystemen.
- Durchführung beliebiger Auswertungen
- Erweiterbarkeit (Integration von neuen Quellen)

## 2.2 Komponenten der Referenzarchitektur

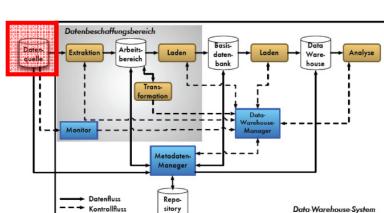


### Die wichtigsten Komponenten in der Übersicht:

- **ETL** (Extraktion Transformation Laden) entspricht der Datenintegration.
- **Arbeitsbereich** (Staging Hub) ist ein temporärer Zwischenspeicher aller extrahierten Daten aus einzelnen Quellen
- **Basisdatenbank** wird in der Umgangssprache DWH genannt. Enthält Daten, welche aber nicht verdichtet sind.
- **Datawarehouse** enthält die verdichteten Daten, persistent gespeichert, für Analysezwecke.

In den nächsten Kapiteln folgt nun eine ausführliche Beschreibung aller Komponenten der Referenzarchitektur.

### 2.2.1 Datenquellen



Die Datenquellen sind die Lieferanten der Daten für das DWH. Die Quellen können intern oder extern sein. Meist sind diese heterogen bezüglich der Struktur, Inhalt und Schnittstellen. Die Auswahl der Quellen und deren Qualität ist äußerst wichtig. Man sollte unbedingt beachten, dass umso mehr Quellen man hat, umso mehr Schnittstellen entstehen. Dies wiederum führt zu negativen Auswirkungen auf die Performance. Wichtige Kriterien zur Auswahl der

#### Datenquellen:

- Brauchen wir die Datenquelle?
- Qualität der Quellen
- Verfügbarkeit
- Preis für den Erwerb der Daten (externe Quellen)

## 2.2.2 Datenquellen - Qualitätsanforderungen

Das Bestimmen der Qualitätsanforderungen an das Datawarehouse bzw. die Datenquellen soll in der Vorstudie passieren, bevor das DWH aufgebaut wird. Folgende Qualitätsanforderungen gibt es:

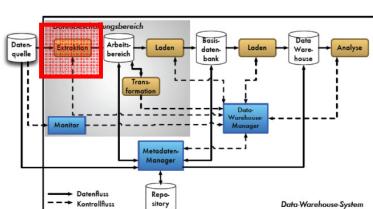
- Konsistenz (z.B. Unterschiedliche Daten in verschiedenen DB's)
- Korrektheit (z.B. negatives Geburtsdatum)
- Vollständigkeit (z.B. Fehlen von Werten oder Attributen)
- Genauigkeit (z.B. Anzahl Nachkommastellen.)
- Verwendbarkeit und Relevanz (DWH nicht mit unnötigen Daten belasten.)

**Garbage in, Garbage out.** Füttert man das DWH mit qualitativ schlechten Daten, kommt auch am anderen Ende wieder garbage raus. Mangelhafte Quelldaten verursachen erhebliche Kosten. Diese verursachen Aufwand beim nachträglichen Beseitigen im Transformationsprozess, zudem können Fehleinschätzungen durch das Management gemacht werden (fehlerhafte strategische Entscheidungen).

## 2.2.3 Datenquellen - Klassifikation

- Herkunft: intern, extern
- Zeit: aktuell (DB), historisch (Data-Mart)
- Nutzungsebene: Primärdaten, Metadaten (Geben an, wie man transformiert hat, wie man zu den Daten gekommen ist.)
- Inhalt: Zahl, Zeichenkette, Grafik, Referenz, Dokument
- Darstellung: numerisch, alphanumerisch
- Vertraulichkeitsgrad: intern, vertraulich, geheim

## 2.2.4 Extraktionskomponente



Die Extraktionskomponente überträgt die Daten aus den Quellen in den Arbeitsbereich. Diese überwindet die Systemgrenzen durch den Zugriff auf: HOST Systeme, Standard Software (ERP) oder Datenbanksysteme via Anfragesprachen (z.B. SQL) bzw. via einer Schnittstelle,

## 2.2.5 Extraktionskomponente – Zeitpunkt der Extraktion

- Die Extraktion findet **beim ersten Laden des DWH** statt. Das heisst eine grosse Menge von relevanten Daten wird aus den Datenquellen extrahiert und zur Aufbereitung weitergegeben.
- Die Extraktion findet **bei der Aktualisierung** während dem Betrieb des DWH statt. Die Extraktion hängt von der Monitoring-Strategie ab, welche aussagt, wie das DWH die Änderungen der Quelldaten erkennen kann.

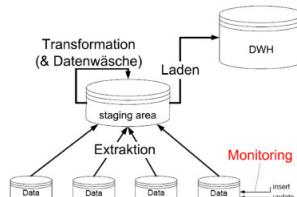
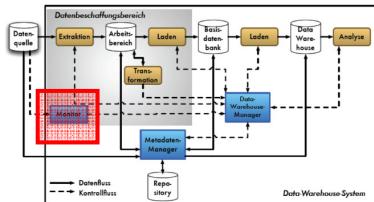
Es gibt zwei mögliche Monitoring Strategien:

- Synchron: Quelle propagiert unmittelbar jede Änderung
- Asynchron: Information über die Änderung erreicht das DWH mit Verspätung, z.b. via Log-File.

## 2.2.6 Extraktionskomponente - Realisierung

- **Make:** Eigene Entwicklung einer Extraktionskomponente. Dies kann je nach DWH Projekt sehr aufwendig sein.
- **Buy:** Kommerzielle Tools verwenden. Ermöglichen den Zugriff auf Standard Software und unterstützen den gesamten ETL Prozess.

## 2.2.7 Monitoring Komponente

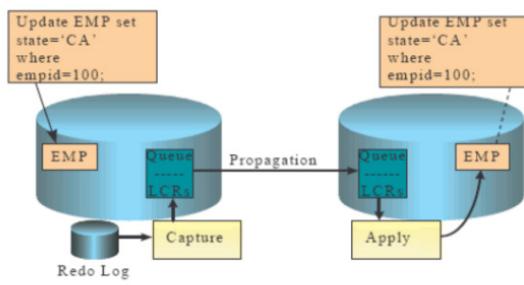


Die Aufgabe von Monitoren ist die Entdeckung von Update- und Insert Operationen auf relevante Daten der Datenquellen. Monitore sind bei der Aktualisierung des DWH involviert.

## 2.2.8 Monitoring Komponente – Techniken für die Realisierung

Die Technik für die Realisierung von Monitoren hängen von der Art der Datenquelle ab. Sie beeinflussen die Implementierung der Extraktionskomponente.

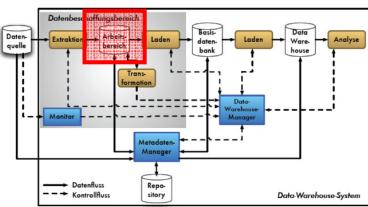
- Aktive Datenquellen:** Bieten Mechanismen welche Update- und/oder Insert Operationen ihrer Umgebung bekanntgeben. Monitoren sind:
- Trigger-basiert:** 1. Auslösen von Triggern bei Datenänderung. 2. Kopieren der geänderten Tupel in anderen Bereich. Beim Eintreten eines Ereignisses (insert, update) wird der Trigger ausgelöst und die Bedingung evaluiert, wenn erfüllt, wird Aktion ausgeführt.
- Replikations-basiert:** Nutzung von Replikationsmechanismen zur Übertragung geänderter Daten. Die Datenquellen replizieren alle Datenbankänderungen in einer Datenbank. Sobald wieder eine Verbindung besteht wird diese abgeglichen (quasi eine Synchronisation). Ein Beispiel eines Replikationsbasierten bzw. Logbasierten Monitors, Oracle Streams: Die Function Capture durchläuft das Log-File (Redo-Log) und trägt jede Änderung im File „Logical Change Record“ LCR ein. Das LCR-File wird repliziert und von der Extraktionskomponente abgefangen.



- Passive Datenquellen:** Keine „aktive Kommunikation“. Monitore sind:
  - Log-basiert:** Analyse von Transaktions-Log-Dateien der DMBS (read, update, delete, insert)
  - Zeitstempel-basiert:** Zuordnung von Zeitstempel zu Tupeln und Aktualisierung bei Änderung seit der letzten Extraktion durch Zeitvergleich.
  - Snapshot-basiert:** Periodisches Kopieren des Datenbestandes in Datei und Vergleich zur Identifizierung von Änderungen. Dieses System ist nur für kleinere Datenbanken geeignet, da ein Aufwand für die Speicherung und Erstellung des Snapshots entsteht. Zudem können je nach Häufigkeit der Snapshot-Erstellung Zwischenänderungen verloren gehen.

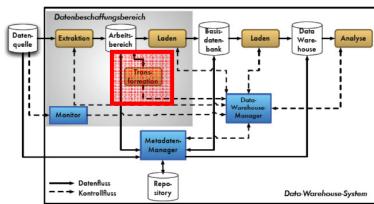
Quelle	Technik	Aktualität des DWH	Belastung der Extraktionskomponente	Belastung der Quellen
<b>Aktive Quellen (synchrone)</b>	- Triggers - Replikation	Maximal	Niedrig: Alle Änderungen werden automatisch in einem File eingetragen	Hoch
<b>Passive Quellen (asynchrone)</b>	Periodische Aktualisierung: - Logs - Snapshots - Zeitstempel	Je nach Frequenz des Pollings der Log-Dateien	Hoch: Die Extraktionskomponente muss die Änderungen bestimmen	Niedrig
	Aktualisierung nur vor der Benutzung der DWH-Daten	Maximal	Hoch	Mittel

## 2.2.9 Arbeitsbereich (Staging Area)



Der Arbeitsbereich ist ein temporärer Zwischenspeicher aller extrahierten Daten aus den einzelnen Quellen. Hier werden die Transformationen ausgeführt (Bereinigung, Integration etc.). Nach dem transformieren gelangen die Daten ins DWH bzw. in die Basisdatenbank. Die Vorteile des Arbeitsbereiches sind: Die Quellen bzw. das DWH wird nicht beeinflusst und fehlerbehaftete Daten werden nicht übernommen.

## 2.2.10 Transformationskomponente

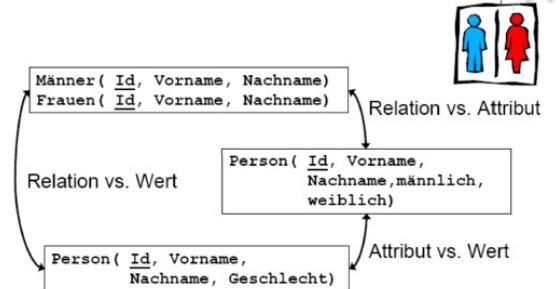


Kritisch bei der Transformationskomponente sind die unterschiedlichen bzw. heterogenen Datenquellen bzw. DWH's. Das Ziel der Transformationskomponente ist die Vorbereitung und Anpassung der Daten für das Laden. Sie hat zwei Aufgaben:

- Datentransformation und Datenintegration
- Datenbereinigung: Data Cleaning / Data Cleansing

### Datentransformation und Datenintegration

Bei der **Datentransformation** werden alle Daten in ein einheitliches Format/Schema überführt. Transformiert werden z.B. Datentypen, Datumsangaben, Masseneinheiten etc. Bei der **Datenintegration** geht es primär um die Problematik der unterschiedlichen Modellierungen bzw. was ist ein Wert, was ist ein Attribut? Die Datenintegration ist anspruchsvoll und nicht komplett automatisierbar. (s.h. Bild rechts.) Weitere Bsp. Finden sich im Foliensatz S. 8 ff.



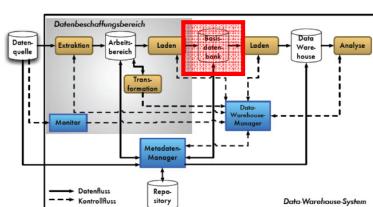
### Datenbereinigung

Die Datenbereinigung kann automatisiert erfolgen. Bereinigt werden müssen die Schlüssel (Primär und Fremdschlüssel) denn diese sind nur für eine Quelle eindeutig, d.h. für das DWH müssen neue erzeugt werden. Fehlende Daten, Attribute, Tupel müssen ergänzt werden. Falsche Daten müssen bereinigt werden, z.B. negative Preise, Falsches Datum (32.01.2010). Die Semantik der NULL Werte muss geprüft werden. → Besser NULL als ein Wert der falsch ist!

Die Datenbereinigung ist notwendig, da die Datenqualität wichtig ist: Bei geringer Qualität können falsche Prognosen, oder Folgekosten durch Fehlentscheidungen durch das Management entstehen. Probleme in der Qualität entstehen auf folgenden Ebenen:

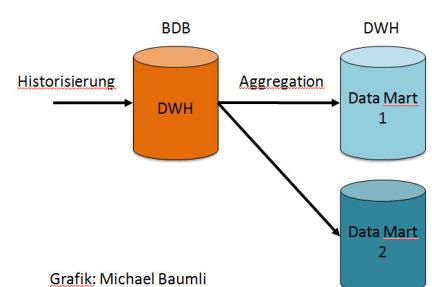
- Ursprungsdaten (Eingabe, Fremdfirmen)
- Quellsysteme (Konsistenz, Fehler)
- ETL-Prozess (d.h. DWH produziert den Fehler selber)

## 2.2.11 Basisdatenbank



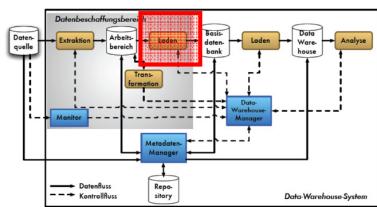
Die Basisdatenbank wird in der Umgangssprache auch als Data Warehouse bezeichnet! In der Basisdatenbank sind die Daten **nicht** verdichtet. Sie fungiert also als Verteilerzentrale für Daten

welche in verschiedene DWH's fließen. In der Praxis wird dies jedoch oft weggelassen.



Grafik: Michael Baumli

## 2.2.12 Ladekomponente

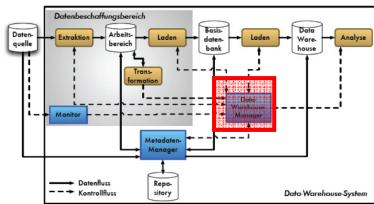


Die Ladekomponente überträgt die bereinigten Daten in die Basisdatenbank bzw. das DWH. Bei der Bildung von Historien dürfen Änderungen in Quellen die DWH Daten nicht überschreiben, sondern zusätzlich abspeichern. Die Load-Vorgänge können unter Umständen die komplette Datenbank blockieren. (Schreibsperrre). Es gibt somit zwei unterschiedliche Ladevorgänge:

- **Online:** Die BDB steht während des Ladevorgangs weiterhin zur Verfügung
- **Offline:** Die BDB steht während des Ladevorgangs nicht zur Verfügung. Deshalb geschieht das Laden während der Nacht bzw. am Wochenende.
- **Transformation beim Laden der Daten**

Für sehr einfache Anwendungen ist es auch möglich die Daten während des Ladevorgangs zu transformieren und ohne Zwischenspeicher direkt ins DWH zu stellen.

## 2.2.13 Data Warehouse Manager

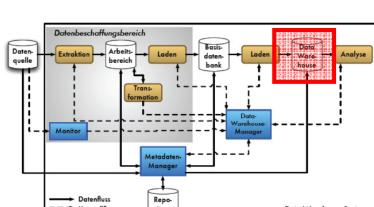


Der Data Warehouse Manager ist die zentrale Komponente eines DWH Systems. Er übernimmt die Steuerung und Überwachung der einzelnen Prozesse. Abhängig von der Monitoring-Strategie wird von ihm der Datenbeschaffungsprozess ausgelöst:

- **Asynchron:** In regelmässigen Zeitabständen (jede Nacht, am Wochenende). D.h. die Extraktion der Daten aus den Quellen in den Arbeitsbereich wird gestartet
- **Synchron:** DWH-Manager wird von der Quelle benachrichtigt und startet dann die Extraktion.

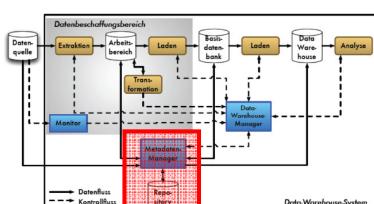
Nach dem Auslösen des Extraktionsprozesses initiiert der DWH-Manager den Datenaufbereitungsprozess. Er kümmert sich um die Überwachung der Schritte (Transformation, Bereinigung, Integration etc...). Außerdem werden von ihm Fehlermeldungen an den Administrator gesandt. Und schlussendlich erfolgt durch ihn der Zugriff zu den Metadaten.

## 2.2.14 Data Warehouse



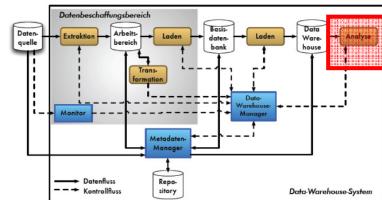
Im DWH befinden sich nun die verdichteten Daten. Das DWH ist eine Datenbank, welche die aufbereiteten Daten persistent (dauerhaft) speichert. Diese Daten werden dann für Analysezwecke verwendet. Die Herausforderungen bei einem DWH ist die Auswahl des Datenmodells, der Entwurf des Data Warehouses und die Schwierigkeit effiziente Anfragen zu ermöglichen.

## 2.2.15 Metadaten Manager & Metadaten Repository

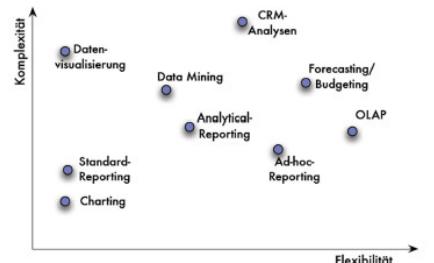


Ein Repository ist nichts anderes als eine Datenbank für die Speicherung von Metadaten. Metadaten vereinfachen den Aufbau, die Wartung und die Administration des DWH Systems. Der Metadaten Manager steuert die Verwaltung, Zugriffe, Anfragen an Metadaten.

## 2.2.16 Analyse



Analysewerkzeuge werden auch Business Intelligence Tools genannt. Diese sind verantwortlich für die Analyse der aufbereiteten Informationen.

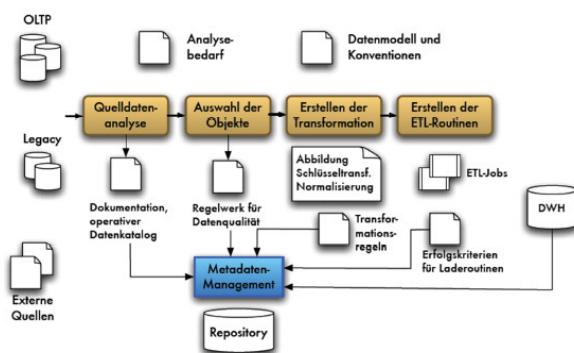


Ein Dashboard beispielsweise ist ein sogenanntes Reporting Werkzeug, welches einfach zu bedienen ist und dem Management eine Übersicht über aktuelle Informationen bereitstellt.

In diese Kategorie fällt ebenfalls OLAP, welche eine interaktive Datenanalyse – durch drill down, roll up etc. - ermöglicht.

## 2.3 Prozesse der Referenzarchitektur

- **Prozesse bei der Entwicklung eines DWH**
  - ETL-Prozess
  - Analyseprozess (bei grossen Datenvolumen eine Herausforderung)
- **Prozesse bei der Einführung von Anwendungen und Plattformkomponenten**
  - Testen
  - Change Management
- **Prozesse zum Betrieb eines DWH (Produktiv)**
  - Starten / Stoppen von Komponenten
  - Überwachung
  - Performance und Capacity Monitoring
  - Backup und Recovery
- **Der ETL Prozess**
  - Extraktion: Selektion von Daten aus den Quellen u. Bereitstellung zur Transformation.
  - Transformation: Anpassung der Daten an das vorgegebene Schema.
  - Laden: Einbringung der Daten aus dem Arbeitsbereich in das DWH.



## 2.4 Varianten der Referenzarchitektur

- **Strategisches DWH**  
Ist ein Instrument zur Analyse und Optimierung von Geschäftsprozessen. Es wird mit einem langfristigen Horizont aufgebaut und soll lange bestehen.

## ▪ Taktisches DWH

Wird für einen bestimmten und klar begrenzten Zweck gebaut. Der Aufbau erfolgt möglichst schnell, damit Analyseergebnisse schnell verfügbar sind.

Es gibt folgende Architekturvarianten für ein Data Warehouse:

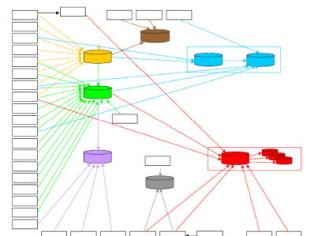
- „Ad-hoc“ DHW (à la « Quick’n’dirty »)
- Zentrales DWH mit virtuellen Data Marts
- Zentrales DWH mit persistenten Data Marts
- Föderierte DHW Architektur (virtuelle Integration)
- Kein zentrales DWH

### 2.4.1 „Ad-hoc“ Data Warehouse (quick’n’dirty)

Es handelt sich hierbei um ein DWH ohne Bezug zur Referenzarchitektur.

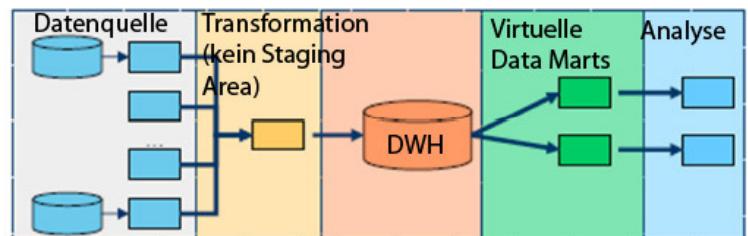
Es ist angemessen für ein taktisches DWH-System auf Abteilungsebene.

Der An-, Aus- und Umbau erfolgt je nach Bedarf. Solch ein DWH ist flexibel, isoliert, teuer und schwer wartbar → Weil nicht geplant, wie es gewartet werden muss!



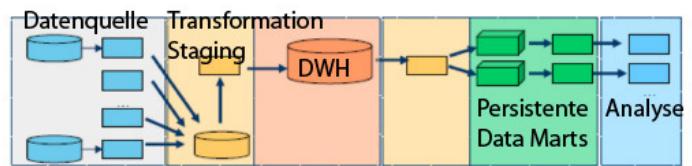
### 2.4.2 Zentrales DWH mit virtuellen Data Marts

In dieser Architektur erfolgt die Trennung zwischen DWH und Data Mart. Die Data Marts sind jedoch als virtuelle Sichten über dem DWH definiert. D.h. die Ergebnisse müssen immer neu berechnet werden. Es eignet sich als strategisches DWH, und für mittlere Unternehmen. Nachteil ist, dass die Quellen stärker belastet werden, da kein Staging Area (Arbeitsbereich) vorhanden ist.



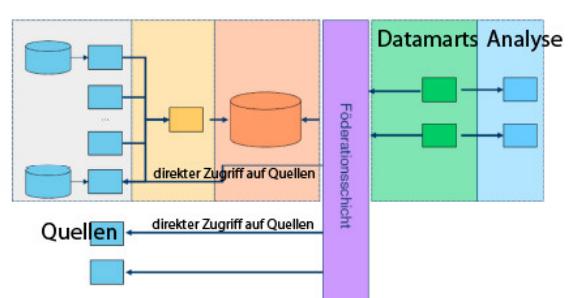
### 2.4.3 Zentrales DWH mit persistenten Data Marts

Ebenfalls erfolgt hier eine Trennung des DWH und Data Marts. Hier sind die Data Marts jedoch persistent. Diese Architektur eignet sich für strategisches DWH. Die Performance ist aufgrund der persistenten Data Marts gut.



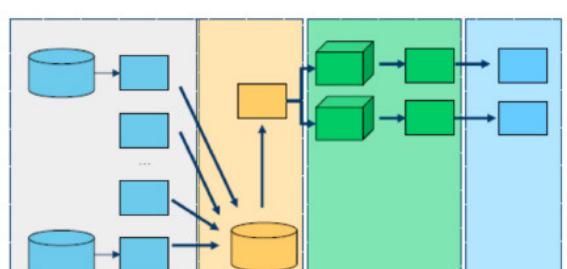
### 2.4.4 Föderierte DHW Architektur (virtuelle Integration)

Die föderierte Architektur hat eine kurze Time-to-Market, ist also schnell erstellt. Es erfolgt keine Historisierung. Die Historisierung ist quasi direkt bei den Quellen. Durch den direkten Zugriff auf die Quellen und operativen Systemen erfolgt deren Beeinträchtigung.



### 2.4.5 Kein zentrales DWH

Es existiert kein zentrales physisches DWH. Die aufbereiteten Daten werden in die Data Marts aufgeteilt. Es eignet sich als taktisches DWH.



### 3 Aufbau eines Data-Warehouse-Systems

#### 3.1 Data-Warehouse-Strategie

In diesem Kapitel wird genauer auf den Begriff Strategie im Unternehmen eingegangen. Dies ist nötig, denn die Business-Strategie hängt mit der IT-Strategie zusammen, und diese wieder mit der DWH Strategie.

##### 3.1.1 Strategie

Unter einer **Strategie** versteht man **Massnahmen und Handlungen** für die nächsten **3-5 Jahre**. Es handelt sich somit um mittel- bis langfristige Ziele. Diese Strategie dient als Orientierungshilfe für Entscheidungen. Eine **Taktik** sind Massnahmen und Handlungen um mit den gegebenen Mitteln **kurz oder Mittelfristige Ziele** zu erreichen. Die groben Bestandteile einer Strategie sind die folgenden:

- Aktuelle Ausgangslage (wo stehen wir heute)
- Strategische Positionierung und Vorgaben (Ziel und Soll-Zustand)
- Der Weg zum Ziel (Wie kommen wir dorthin?)

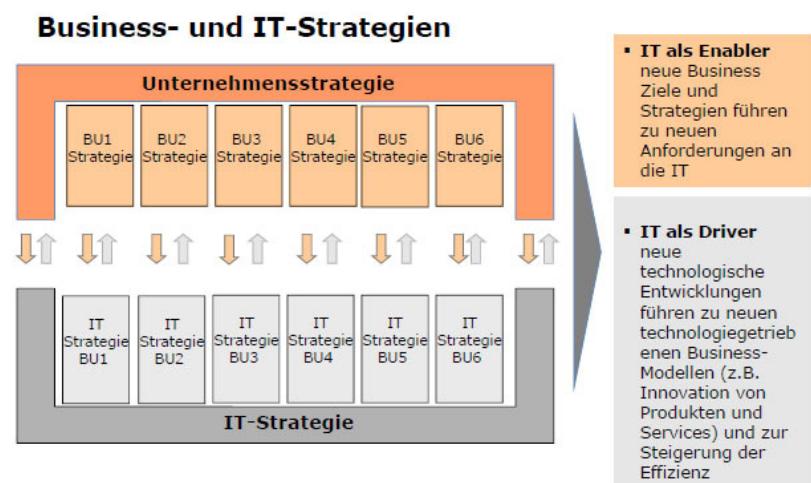
Die Geschäftsbereiche eines Unternehmens müssen ihre Business-Strategien aufgrund der Unternehmensziele festlegen. Je nach Komplexität des Unternehmens kann der Bereich IT eigene IT-Bereichsstrategien festlegen:

- Business Intelligence bzw. DWH-Strategie
- Lieferantenstrategie
- Datenstrategie
- Netzwerk/Hardware Strategie

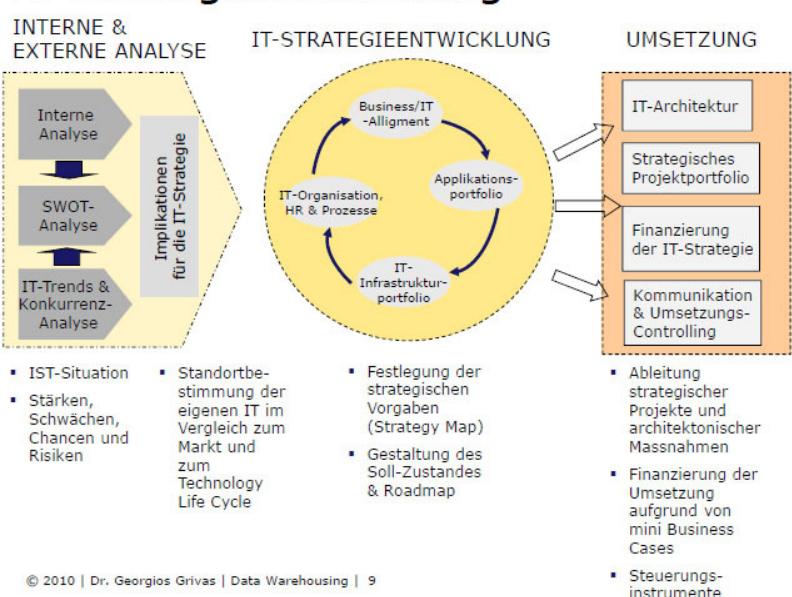
Die IT selber kann als „**Enabler**“ auftreten. Sprich die IT ermöglicht die Ziele und Strategien mit neuen IT Lösungen zu decken.

Die IT selber kann aber auch als „**Driver**“ auftreten. Sprich Sie tritt als **treibende Kraft** auf und neue technologien führen zu neuen Business Modellen.

Folgende Grafik zeigt die **IT-Strategieentwicklung**. Zuerst werden Analysen durchgeführt und die IST-Situation abgebildet. Anschließend erfolgt die Strategieentwicklung, dort miteinbezogen wir das Business, die Applikationen, Infrastruktur und IT-Organisation. Nach der Entwicklung folgt die Umsetzung durch Projekte.



#### IT-Strategieentwicklung

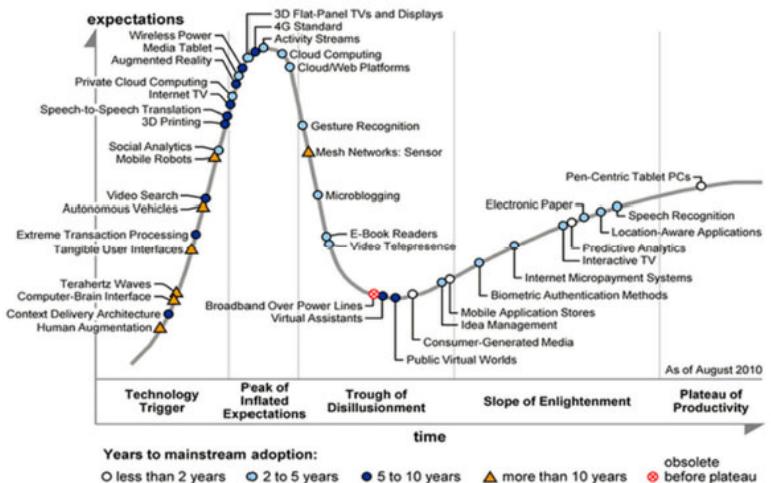


Der Hype Cycle zeigt den Lebenszyklus eines Trends.

## Gartner Hype Cycle 2010

Wirtschaft

- **Technology Trigger:** Durchbruch einer neuen Technologie.
- **Peak of inflated expectations:** Technologie ist in aller Munde.
- **Through of disillusionment:** Abklingen der Euphorie.
- **Slope of Enlightenment:** Langzeiterfahrung, stabile Tools.
- **Plateau of productivity:** bewährte Technologien.



### 3.1.2 Grundstrategien in der IT

Eine Grundstrategie ist beständig und wird nicht von Technologie- oder Produktwechsel beeinflusst. Es gibt folgende Grundstrategien:

- **Technologische Innovation**
  - **First mover (hohes Risiko)**  
Darunter zählen technologische Innovationen, neue Produkte, Erschließung neuer Märkte und somit höhere Gewinnspanne. Gefahren können eine unstabile Technologie und hohe Investitionskosten sein.
  - **Early Follower (mittleres Risiko)**  
Frühes erkennen des Nutzen von First Mover. Es herrscht eine gewisse Sicherheit, da First Mover den Markt schon erprobt hat.
  - **Late Follower (geringes Risiko)**  
Längst bewährte Strategien, Leistung zu geringeren Kosten, keine Wettbewerbsvorteile
- **Applikations- und IT-Infrastrukturstrategien**
  - **Housekeeping**  
Diese Strategie wird angewandt, wenn die Firma Liquiditätsprobleme hat. Das Unternehmen versucht mit der bestehenden Applikationslandschaft zu überleben. Mögliche Sofortmassnahmen: Bereinigung der Stammdaten, Zentralisierung des IT-Einkaufs, Überprüfung der Serviceverträge.
  - **Big Bang**  
Die Big Bang Strategie kann mit „Greenfield“ verglichen werden. Man eliminiert die alte Systemlandschaft und führt eine völlig neue ein, quasi auf einer grünen Wiese also. Das System wird auf einen Schlag erneuert. Die Kosten und Risiken sind hoch.
  - **Managed Evolution**  
Im Gegensatz zur Big Bang Strategie wird hier der neue Zustand schrittweise erreicht. Schrittweise Änderungen und kontrollierte Weiterentwicklung.
  - **Big Core**  
Ein Umfassendes Kernsystem wird eingekauft und die IT zentralisiert. Eine Lösung für alle Ländergesellschaften.

- **Tuning**  
Basis bildet das aktuelle System. Plattformen werden womöglich ausgebaut. D.h. die bestehende Applikationslandschaft wird immer zuerst optimiert, bevor in neue Lösungen investiert wird.
- **Best-of-Breed**  
Für jeden Bereich wird eine individuelle Lösung evaluiert. z.B. Länderspezifische Einzellösungen.
- **Make or buy**  
Einsatz von Standardsoftware oder Eigenentwicklung oder eine Kombination.

### 3.1.3 Data-Warehouse-Strategie

Die DWH Strategie ist ein Teil der Business-Intelligence bzw. der IT-Strategie. Die DWH Strategie fokussiert sich vor allem auf die Datenintegration und Analyse. Sie muss mit der IT-Strategie abgestimmt sein. Für ein DWH müssen die **strategischen Einsatzfelder** inhaltlich definiert werden:

- Planung und Reporting
- Strategisches Controlling (Erlös-, Marketing-, Vertriebs-, Controlling, Kennzahlensysteme etc.)
- Analyseorientierte Anwendungen.

Zudem muss der **Umfang** der **strategischen Einsatzfelder** definiert werden:

- Organisatorisch: isolierte Anwendung, oder Unterstützung mehrerer Bereiche
- Physisch: Ein zentraler Standort, mehrere regionale Standorte etc...
- Zeitlich: Implementierungs-Roadmap der DWH Anwendungen.

## 3.2 Reifegradmodell

Ein Reifegradmodell misst die Qualität und Reife der Softwareentwicklung, Business- und IT-Prozesse in einer Unternehmung. Reifegradmodelle für BI und DWH Lösungen sind neu und haben folgende Ziele:

- Standortbestimmung des Unternehmens (SWOT Analyse, Benchmarking)
- Analyse und Bewertung der eigenen BI/DWH Lösung (Maturität und Reifegrad)
- Ableitung von Verbesserungsmassnahmen.
- Zertifizierungen

### 3.2.1 Anforderungen an BI/DWH- Reifegradmodelle

Ein umfassendes Modell berücksichtigt die folgenden drei Perspektiven:

- Fachlichkeit
- Technologie
- Organisation

Ein gutes Reifegradmodell sollte die grosse individuelle Komplexität der bestehenden DWH Lösung berücksichtigen. Zudem muss das Modell abteilungsübergreifend sein, denn Entscheidungen betreffend meist mehrere Abteilungen.

### 3.2.2 Business Intelligence Maturity Model (biMM®)

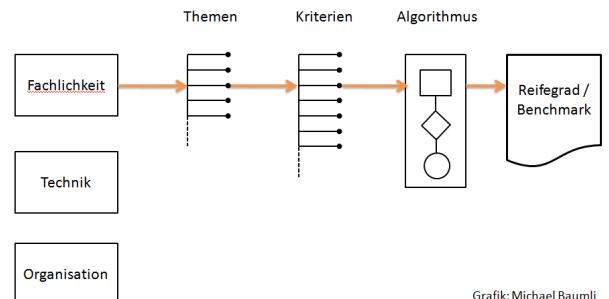
Es existieren zahlreiche BI-Reifegradmodelle in der Praxis. Im folgenden (so wie auch in unserem Unterricht) gehen wir nur auf das „Business Intelligence Maturity Model (biMM®) der Firma „Steria Mummert Consulting“ ein. Das biMM basiert auf einer Lebenszyklusbetrachtung, die BI/DWH-Prozesse werden anhand der drei Perspektiven: **Fachlichkeit** (Informationsarchitektur, Wirkungsbereich), **Technik** (Datenmgmt., Infrastruktur, Techn. Architektur) und **Organisation** (BI-Prozesse, BI-Aufbauorganisation etc.) analysiert. Das eigentliche Ergebnis des Reifegradmodells ist die Bestimmung des Entwicklungsstandes (Maturität bzw. Reifegrad) und die einfache Vergleichbarkeit mit der Konkurrenz.

### 3.2.3 biMM Vorgehen

Je nach Problemstellung erfolgt eine gezielte Auswahl von Untersuchungsfeldern (Perspektiven):

- fachliche Themen wie Customer Relationship
- technologische Themen wie SAP
- spezielle Themen wie unternehmensweite Konsolidierung von BI-Systemen

Zu den jeweiligen Untersuchungsfeldern stehen 15 Themen mit jeweils bis zu 150 Kriterien zur Verfügung. Das vorgehen kann somit wie folgt grafisch dargestellt werden. (s.h. rechts)



Grafik: Michael Baumli

### 3.2.4 biMM Maturitätsstufen

Je nachdem wie eine Firma organisiert ist, kommt Sie in eine bestimmte Stufe. Die Maturitätsstufe ist quasi abgeleitet von den Kriterien.

biMM®		Stufe 1: Einzel- Information	Stufe 2: Informations- inseln	Stufe 3: Informations- integration	Stufe 4: Information Intelligence	Stufe 5: Enterprise Information Management
Fachlichkeit	Einzelberichtssicht	Bereichsbezogenes Geschäfts- verständnis	Fokussierung	Strategisches Alignment	Operative Integration	
Technik	Datenanarchie	Data Mart	Data Warehousing	Zukunfts- orientierung	Zeitnahe Informations- bereitstellung	
Organisation	Initial	Projekt	Eigenständige BI-Organisation	Prozessorientierte IT	Unternehmensweite BI-Organisation	

#### ▪ Maturitätsstufe 1 – Einzelinformation

In der ersten Maturitätsstufe kommen **starre** fachbereichsbezogene **Reports** vor. Es existieren **Datenredundanzen** durch parallele Berichte über den gleichen Inhalt in unterschiedlichen Hierarchiestufen, was zu **Inkonsistenzen** führt. Es existiert **keine richtige Analysefunktion**. Zudem herrscht eine Überflutung von ungefilterten, **irrelevanten** und **unverdichteten Daten**.

#### ▪ Maturitätsstufe 2 – Informationsinseln

Die 2. Maturitätsstufe hat zwar **schon viel vom DWH Prinzip erreicht, besteht jedoch nur aus Data Marts**. Hierbei handelt es sich um ein Fachbereich bezogenes DWH, welches aus **redundanzfreien** eindeutigen Daten besteht. Es existiert eine **Ad-hoc-Analysefunktionalität**. Eine Historisierung ist möglich.

#### ▪ Maturitätsstufe 3 – Informationsintegration

Bei der Stufe 3 handelt es sich um ein **unternehmensweites DWH** mit vereinheitlichter Nutzung von Daten aus weiten Teilen des Unternehmens. Es ist **redundanzfrei**, und **ermöglicht bereichsübergreifendes Reporting**. Architektur: **Zentrales DWH und Data Marts**. Da es **hohe Kosten** verursacht, muss es auch auf hoher Führungsebene Unterstützung finden.

#### ▪ Maturitätsstufe 4 – Information Intelligence

**Erweiterte Entscheidungsunterstützung** auf Basis der DWH Lösung, mit **anspruchsvollen Analysemethoden und Werkzeugen**. Es existieren intelligente Methoden zur Datenaufbereitung wie: **Data Mining** für die Analyse, Trendextrapolation für die Planung.

- **Maturitätsstufe 5 – Enterprise Information Management**

Es besteht eine **vollständige Sicht auf die relevanten Geschäftsobjekte**. Durch Enterprise Application Integration (**EAI**) Plattformen können Unternehmen ihre **DWH-Lösung in Echtzeit** mit Daten versorgen. Dies ermöglicht eine zeitnahe Informationsbereitstellung.

### 3.2.5 Entwicklungen einer DWH-Lösung

Eine höhere Maturitätsstufe von Reifegradmodelle bedeutet einen fortgeschrittenen Status der DWH-Lösung. Es müssen nicht alle Stufen durchlaufen werden, und nicht alle Unternehmen streben unbedingt höhere Maturitätsstufen an.

- **Perspektive Fachlichkeit**

- Stufe 2: Dezentrale Lösungen
- Stufe 3: Zentralisierung steht im Vordergrund.
- Der Übergang auf Stufe 3 erfolgt ohne fachliche Erweiterung, sondern lediglich durch die Migration der bestehenden Applikationen bzw. Auswertungen.
- Stufe 4: Fokus verschiebt sich auf Flexibilität und das DWH wird wieder dezentraler.

- **Perspektive Organisation**

- Stufe 2: Dezentrale Spezialisten entwickeln in den Fachbereichen
- Stufe 3: Zentrale DWH-Organisation in der IT.
- Stufe 4/5: Die Forderung nach Flexibilität führt zu einer stärkeren Verantwortung in den Fachbereichen.

- **Perspektive Technologie**

- Wandel zwischen Standardisierung und lokaler Kontrolle
- Stufe 3: Best-of-Breed-Ansätze und Standardisierung steht im Vordergrund
- Stufe 4/5: Teilweise Abbau der Standardlösungen.

## 3.3 Ableitung der Data-Warehouse-Architektur

### 3.3.1 Definition IT Architektur

Der **Begriff IT-Architektur** umfasst alle **statischen** und **dynamischen Aspekte der IT** in einer Organisation. Dies kann unter anderem die **Infrastruktur** (Hardware, Standorte, Netzwerk, Software, Daten etc.) und das **Management** (Release, Change, Konfigurationsmgmt., Datensicherung, Verfügbarkeit etc.) sein. Darüber hinaus sind **funktionale Aspekte** wie **Schnittstellen, IT-Unterstützung der Prozesse** in der Organisation ebenfalls ein Teil der Architektur.

Die Architektur legt die Grundstrukturen fest und definiert Regeln, die das dynamische Zusammenspiel aller Komponenten koordinieren. Sie ist quasi eine Leitlinie für alle Personen und Gruppen, welche an der Planung, Bau und Betrieb der wesentlichen IT-Systemen und Infrastruktur beteiligt sind.

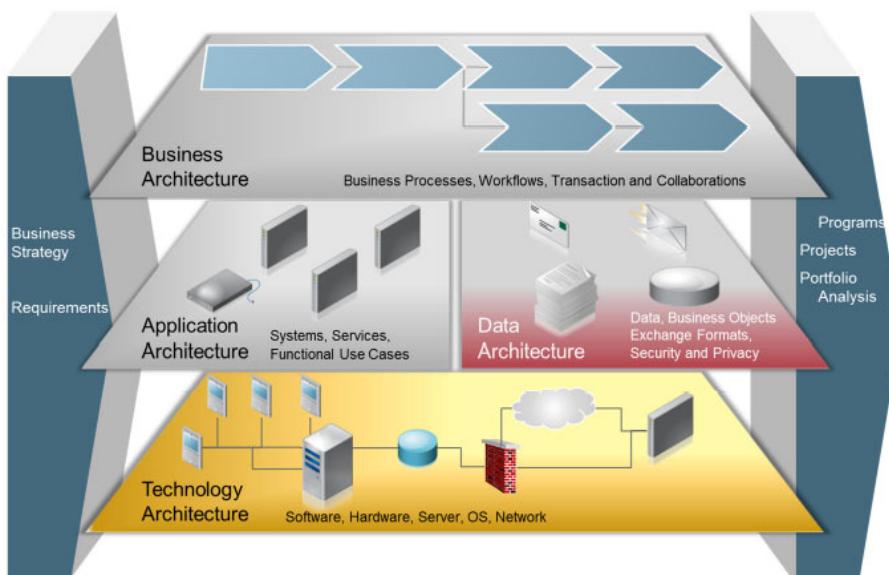
### 3.3.2 Architektur Framework

Ein Framework ist ein Hilfsmittel, um bei der Systemplanung und Entwicklung alle relevanten Aspekte aus allen Perspektiven zu berücksichtigen. Bekannte IT Architecture Frameworks sind:

- **Zachmann Framework (1987)** als rollenbasiertes Modell für die Entwicklung von Informationssystemen; kein Focus auf die Prozesse und die Schnittstellen; Grundstein für viele Ansätze und Ideen.
- **The Open Group Architecture Framework (TOGAF)** basiert auf dem „Technical Architecture Framework for Information Management“ (TAFIM) des Department of Defense (DoD)
- **US Federal Enterprise Architecture Framework (FEAF)** Struktur für die Unternehmensarchitektur von US-Behörden ermöglicht die Entwicklung einheitlicher Prozesse
- **Department of Defense Architecture Framework (DoDAF)**; Einsatz für Unternehmensarchitekturen im militärischen Bereich der USA; besonders geeignet für große Systeme mit komplexen Integrations- und Kommunikationsaufgaben.
- **Extended Enterprise Architecture Framework (E2AF)** basiert auf bestehenden Frameworks wie FEAF und TOGAF

*Hinweis Autor: Auf die obenstehenden Frameworks wird hier nicht genauer eingegangen. Lediglich die Grafik zum TOGAF Framework wird hier abgebildet → because Grivas likes it.*

### The Open Group Architecture Framework (TOGAF)



### 3.3.3 Data Warehouse Framework

Im Folgenden werden die verschiedenen Architekturen erklärt. Diese werden zum Aufbau eines Data Warehouse Frameworks benötigt. Der Aufbau eines Frameworks geschieht Initial (am Anfang) in einem Data-Warehouse-Projekt. Das Framework basiert auf internen Analysen und der IT-Strategie. Es folgt eine Vorgabe vom Management bzgl. Der Entwicklung des DWH's. Als erstet wird dann die SOLL-Business Architektur erstellt, und daraus abgeleitet werden die SOLL Applikationsarchitektur und die SOLL-Datenarchitektur definiert.

- **Business-Architektur**
  - Beschreibt wie das Business Model eines Unternehmens implementiert wird.
  - Enthält eine Übersicht für alle Business-Prozesse, Aktivitäten und Zusammenhänge.
  - Ist Teil des Architektur Frameworks.
  - Weist auf die aktuellen strategischen Ziele.

- **Applikations-Architektur**
  - Besteht aus einer Menge von Applikationen und deren Zusammenspiel
  - Zeigt auf welche Anwendungen für die Ausführung der Geschäftsprozesse erforderlich sind.
  - Beziehungen zwischen den Schnittstellen werden betrachtet.
- **Daten-Architektur**
  - Beschreibt die Daten mit ihren Beziehungen welche für die Durchführung der Geschäftsprozesse notwendig sind.
  - Eine Beziehung zur Applikationsarchitektur muss vorhanden sein. Welche Applikationen greifen auf welche Daten zu?
  - Beschreibt die Metadaten.
- **Technologie-Architektur**
  - Beschreibt die Infrastruktur (Software und Hardware)
  - Bezieht sich auf die Datenarchitektur und Applikationsarchitektur.
  - Zeigt Möglichkeiten für die gemeinsame Nutzung von Komponenten.

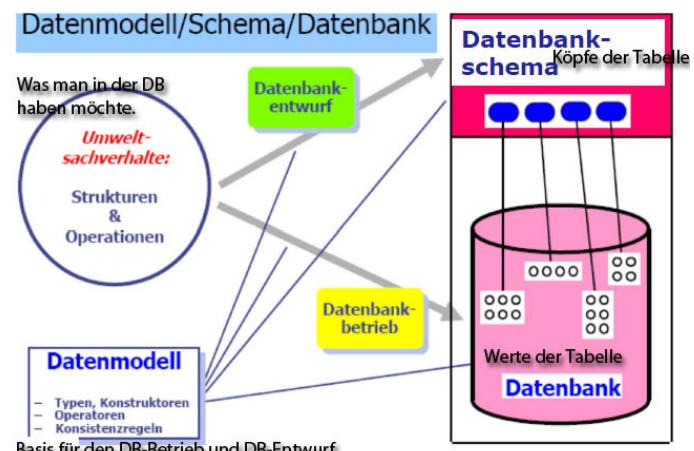
## 4 Multidimensionale Modellierung

### 4.1 Datenstrukturen und Operationen im multidimensionalen Datenmodell (MDDM)

#### 4.1.1 Datenmodell

Ein Datenmodell kann z.B. relational oder objektorientiert sein. Ein Datenmodell ist eine Sammlung von **Konzepten** welche benutzt werden um die **Struktur** einer Datenbank zu beschreiben. Es umfasst ebenfalls die **Basisoperationen** (update, insert, delete) welche für die Spezifikation von Anfragen an eine DB nötig sind. Datenmodelle können in Klassifikationen unterteilt werden:

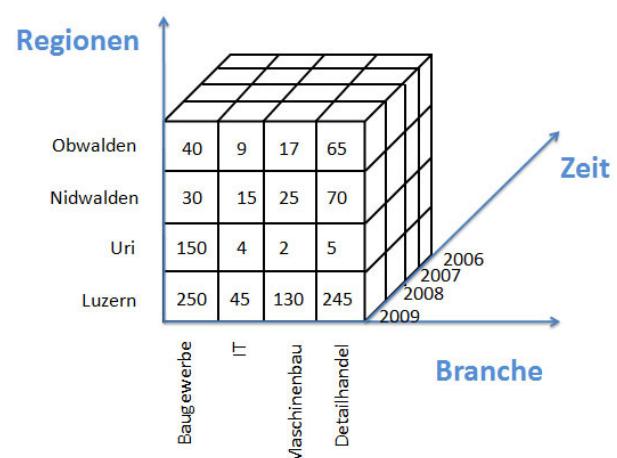
- |  |  |
|--|--|
| <ul style="list-style-type: none"> <li>▪ Logische Datenmodelle</li> <li>▪ Konzeptuelle Datenmodelle</li> <li>▪ Semistrukturierte Datenmodelle</li> </ul> | relational, objektorientiert, <b>multidimensional</b> , etc.<br>ERM, UML, OEM<br>HTML, XML |
|--|--|



#### 4.1.2 Multidimensionales Datenmodell

Durch neue Datenstrukturen und Operationen, werden die Anfrage-Techniken von OLAP optimal unterstützt. Es gibt die folgenden Anfrage-Techniken in OLAP:

- Ausgehend von Detailwerten, Berechnung von verdichteten Daten in Form von Kennzahlen. Dies erlaubt eine multidimensionale Sichtweise.
- Einfache Navigation zwischen verdichteten Daten und Detailwerten.



### 4.1.3 Kennzahlen

Kennzahlen liefern **verdichtete Informationen**. In einem Unternehmen sind dies aussagekräftige Größen. Man nutzt Kennzahlen zur **Beurteilung** von Unternehmen. Berechnet werden Sie durch OLAP. Die Kennzahl dient:

- Als Basis für Entscheidungen (Probleme, Stark- Schwachstellen etc.)
- Zur Kontrolle (Soll-Ist)
- Zur Koordination wichtiger Sachverhaltung im Unternehmen

Kennzahlen kommen auch im Benchmarking vor. Die Kennzahl des besten Unternehmens stellt somit ein Wert dar, an dem sich andere Unternehmen orientieren können. Kennzahlen können in Absolute und Relative Kennzahlen unterteilt werden:

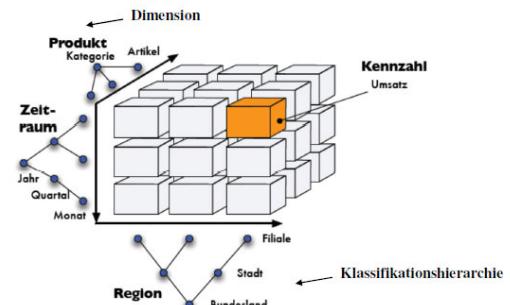
- **Absolute Kennzahlen**
  - Unabhängig von anderen Zahlengrößen (z.B. Umsatz)
  - Bedeutung durch ein Verhältnis (Soll/Ist)
- **Relative Kennzahlen**
  - Zwei oder mehr Kennzahlen miteinander in Beziehung gesetzt.
  - Gliederungszahlen (Anteil Produktumsatz am Gesamtumsatz)
  - Messzahlen (Index)

### 4.1.4 Dimensionen und Klassifikationshierarchien

Eine **Dimension** des multidimensionalen Datenmodells ist eine ausgewählte Entität, welche die Auswertungssicht definiert, z.B. Region, Zeitraum, Produkt.

Jede Dimension hat:

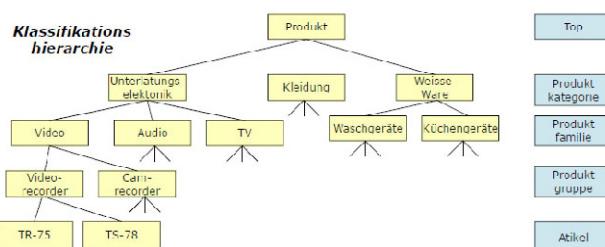
- **Klassifikationshierarchie:**
  - Produkt: Kategorie, Artikel
  - Zeit: Tag, Woche, Jahr
- **Dimensionselemente (Werte)**
  - Tag: 01.01.10, 02.01.10 etc.
  - Woche: 1/2010, 2/2010... 52/2010
  - Jahr: 2008, 2009, 2010...



### 4.1.5 Klassifikationshierarchie

Jede Dimension besitzt eine Klassifikationshierarchie. Jede Klassifikationshierarchie hat mehrere Klassifikationsstufen, d.h. verschiedene Verdichtungsstufen. (Tag → Monat → Quartal → Jahr) Auf den untersten Stufen werden atomare Werte für die Kennzahlen hinterlegt. Eine höhere Klassifikationshierarchie enthält die aggregierten Werte. Es stellt sich somit die Frage was die richtige Klassifikationshierarchie ist? Das kann man eigentlich nicht beantworten, es gibt lediglich bessere und weniger gute.

**Beispiel einer Klassifikationshierarchie für die Dimension «Produkt»**



Lehrstellenmarkt (Anz. Stellen)

Dimensionen und Klassifikationshierarchien:

- Top ←      Bottom →
- **Regionen:** Schweiz, Kanton, Gemeinde
  - **Zeit:** Jahr, Quartal, Monat
  - **Branche:** Sektor, Berufsgattung

#### 4.1.6 Würfel

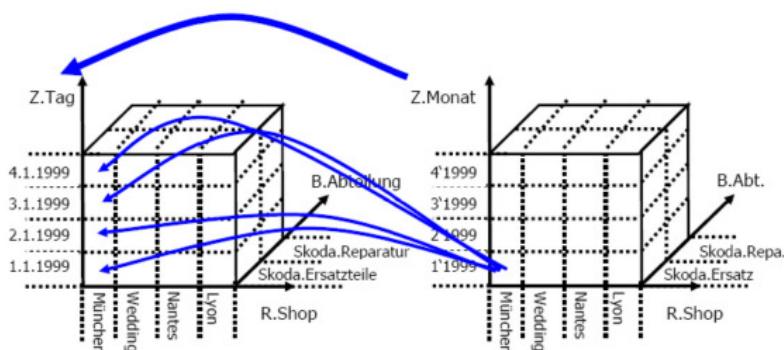
Der Würfel ist die Grundlage der multidimensionalen Analyse (drei Dimensionen). Die Kanten entsprechen den Dimensionen. Die Kantenlänge ist die Anzahl der Dimensionselemente. Die Würfellebenen enthalten eine Kennzahl basierend auf Rohdaten. (Bsp. Würfel s.h. Kap. 4.1.2)

#### 4.1.7 Operationen

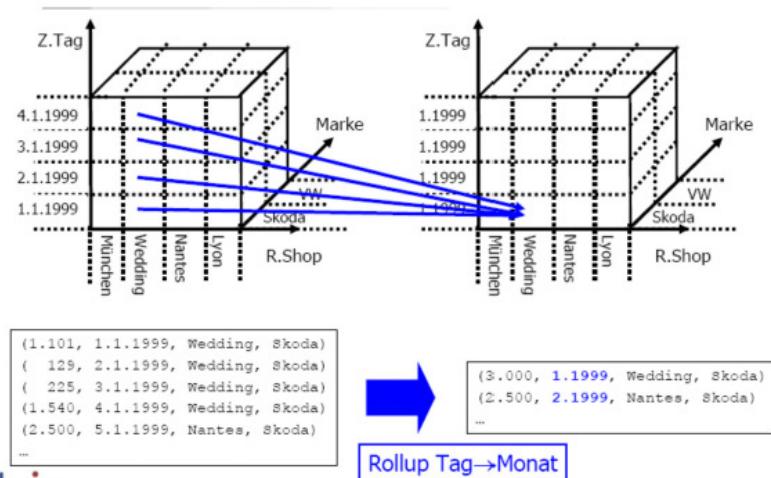
Die Operationen im multidimensionalen Datenmodell unterstützen die Analysevorgänge:

- **Drill-Down:** Aggregation, auf detaillierte Werte herunterbrechen
- **Roll-Up:** Gegenoperation zu Drill-Down, verdichten auf höhere Hierarchiestufe (z.B. von Monats auf Jahressicht.)
- **Pivoting/Rotation:** Drehen des Datenwürfels durch Vertauschen der Dimensionen um seine Achsen. Ermöglicht die Analyse aus einer anderen Perspektive.
- **Slicing:** Ausschneiden von Scheiben aus dem Datenwürfel.
- **Dicing:** Gleichzeitige Slicing-Vorgänge, in unterschiedlichen dimensionen.

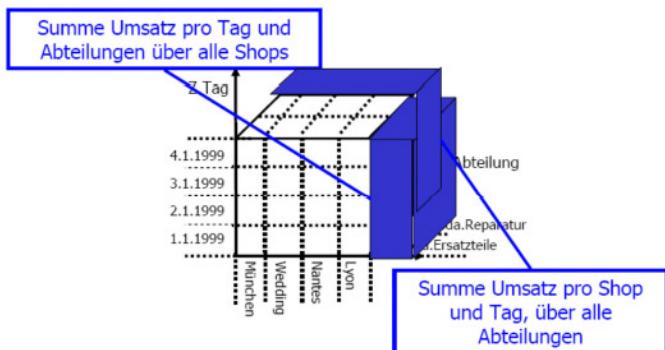
#### Bsp. Drill-Down (Verfeinerung von Monat auf Tag)



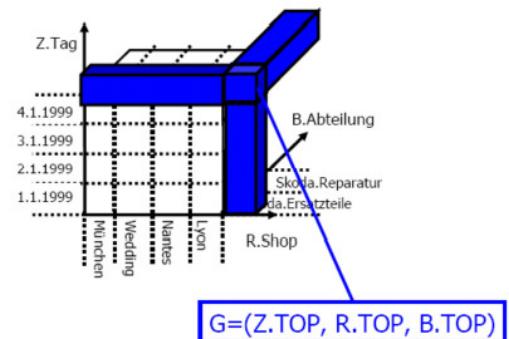
#### Bsp. Roll-Up (Verdichtung von Tag auf Monat)



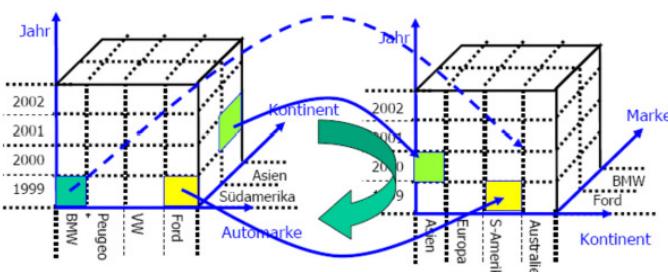
### Bsp. Roll-Up (Verdichtung bis zum TOP)



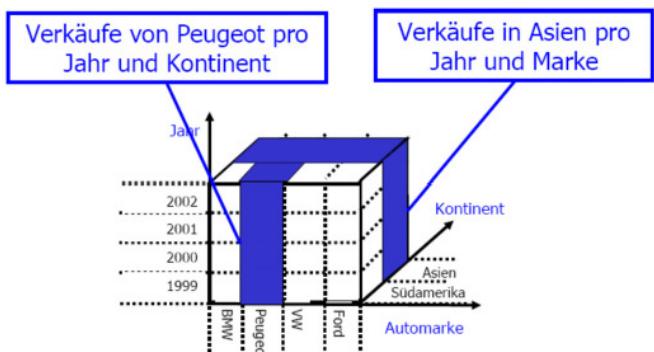
### Bsp. Roll-Up in mehreren Dimensionen



### Bsp. Pivoting (Rotation)

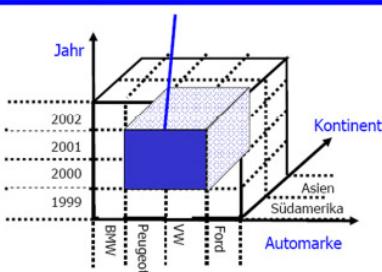


### Bsp. Slicing (Selektion einer Scheibe)



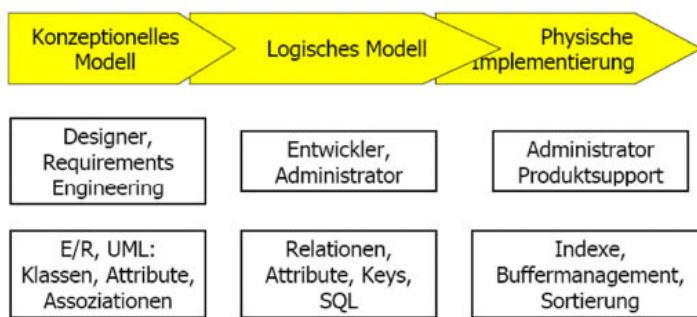
### Bsp. Dicing (Auswahl von Unterwürfeln)

Verkäufe von (Peugeot, VW) in (2000, 2001) pro Kontinent

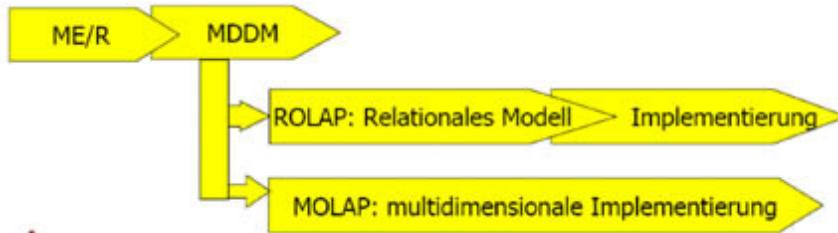


## 4.2 Speicherung und Verwaltung von multidimensionalen Daten

### 4.2.1 Datenbankentwurf in Datenbanksystemen



## 4.2.2 Datenbankentwurf in Data-Warehouse-Systemen

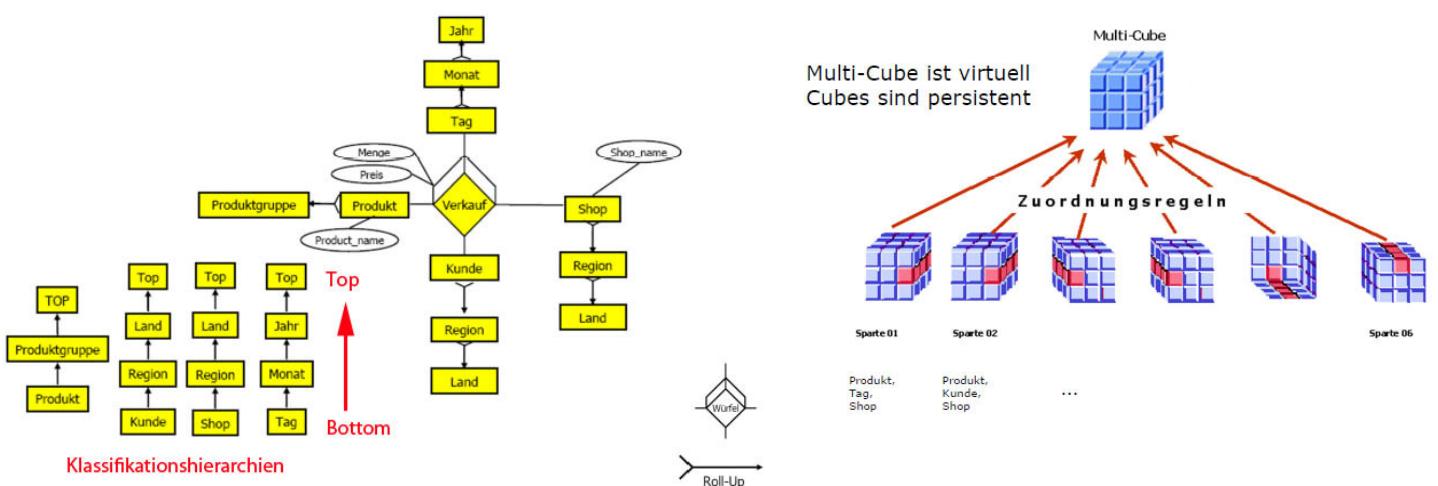


ME/R Multi Entity Relationship

MDDM Multi Dimensionales Daten Modell

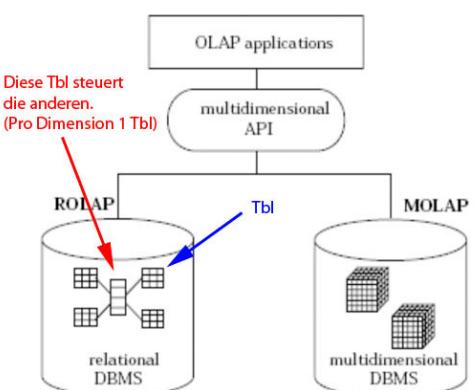
ROLAP und MOLAP sind die Umsetzungsmöglichkeiten eines Multi Dimensionalen Daten Modells. Je nach Datenmodell des DBMS wird entweder MOLAP oder ROLAP angewendet. In der Praxis wird zu 80% ROLAP eingesetzt, deshalb wird hier auch genauer darauf eingegangen.

## 4.2.3 Beispiel ME/R Modell



## 4.2.4 ROLAP vs. MOLAP

- **ROLAP: relationales OLAP**
  - ROLAP bezeichnet multidimensionale Analysen von Daten, die sich physikalisch in einer relationalen Datenbank befinden. (z.B. Oracle, SQL-Server)
  - Wegen der technischen Reife (gute Performance und Skalierbarkeit) werden relationale Datenbanksysteme zur Speicherung und Verwaltung der DWH-Daten verwendet.
  - Ein Transformationsschritt ist notwendig, denn die multidimensionale Strukturen (Würfel) müssen ja irgendwie als Relationen abgebildet werden. Mehr dazu später.
  
- **MOLAP: multidimensionales OLAP**
  - Hier erfolgt eine direkte Speicherung von mehrdimensionalen Würfeln in einem multidimensionalen DBMS.
  - Es ist also kein Mapping notwendig (Bei ROLAP schon.)
  - MOLAP ist kein standardisiertes Modell wie das SQL-Standard.



#### 4.2.5 Grundlagen von ROLAP

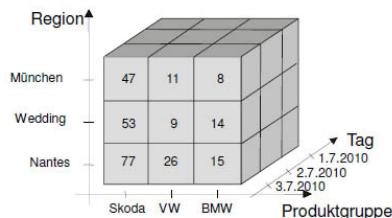
Bei ROLAP (relationales OLAP) werden die Daten in einer relationalen Datenbank gespeichert. Nun, wie werden nun die multidimensionalen Strukturen (Würfen) als Relationen abgebildet? Beziehungsweise, wie würde das Schema aussehen? Folgende Anforderungen stellt man an das Schema:

- Möglichst wenig Semantik vom multidimensionalen Modell darf verloren gehen (Klassifikationshierarchien).
- Übersetzungen der multidimensionalen Anfragen (z.B. SELECT) müssen möglichst effizient laufen.
- Das Laden des DWH (INSERT) muss effizient laufen.

Nun gibt es verschiedene Schemata, die sich je nach Abbildungsart der Klassifikationshierarchien z.B. Snowflake oder Stern-Schema unterscheiden. Man kann aber grundsätzlich sagen:

- Spalten der Relation werden als Dimensionen des Würfels betrachtet.
- Tupel einer Tabelle entspricht einer Zelle im multidimensionalen Würfel.

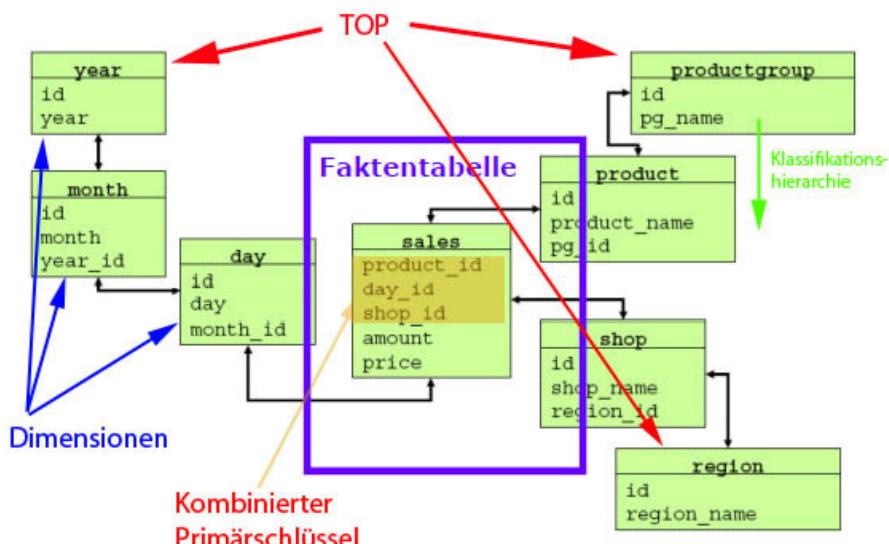
Schaut man sich nun bei ROLAP den Würfel und die entsprechende Tabelle an, sieht das so aus:



Produktgruppe	Region	Tag	Sales
Skoda	Mantes	3.7.2010	77
Skoda	München	3.7.2010	47
BMW	Nantes	3.7.2010	15

#### 4.2.6 ROLAP – Snowflake Schema

- Für jede Klassifikationsstufe wird eine eigene Tabelle angelegt.
- Die Tabelle enthält neben dem eigenen Primärschlüssel, die ID des nächsten Klassifikationsknoten als Fremdschlüssel, sowie die Attribute.
- Zwischen zwei „benachbarten“ Klassifikationsstufen herrscht eine 1:n Beziehung
- Die Faktentabelle enthält die Kennzahlen des Datenwürfels.

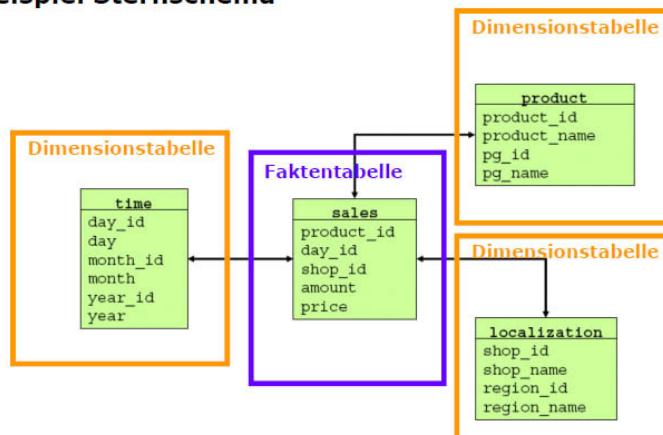


Die Abfrage: „Alle Verkäufe der Produktgruppe BMW für jeden Shop für ein Jahr“ benötigt bereits **6 Joins!** Die Anzahl der Joins in der Anfrage steigt in jeder Dimension linear mit der Länge der Verdichtung. Man kann sagen: Je höher die Verdichtung desto mehr Joins, desto mehr Performance wird gebraucht.

#### 4.2.7 ROLAP – Sternschema

Das Sternschema hat sich in der Praxis durchgesetzt. Das Sternschema ordnet die Dimensionstabellen um eine Faktentabelle. Es ermöglicht die Abbildung mehrdimensionaler Strukturen (Würfel) auf zweidimensionale Tabellen. Für jede Dimension gibt es genau eine Dimensionstabelle. Somit entstehen weniger Tabellen, da die Klassifikationsstufen zusammengefasst sind:

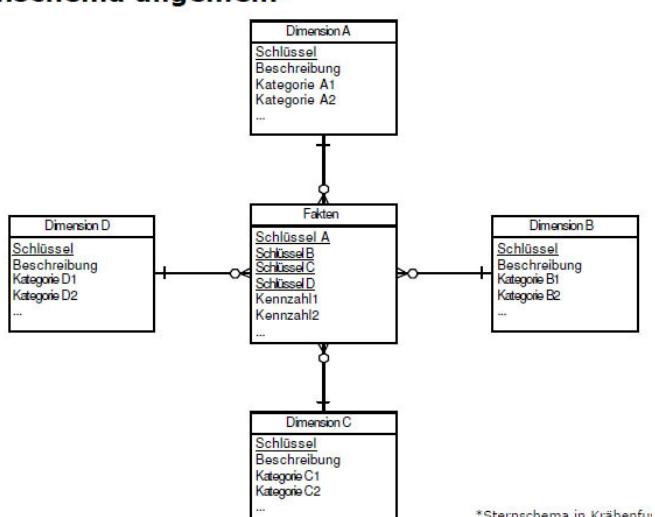
##### Beispiel Sternschema



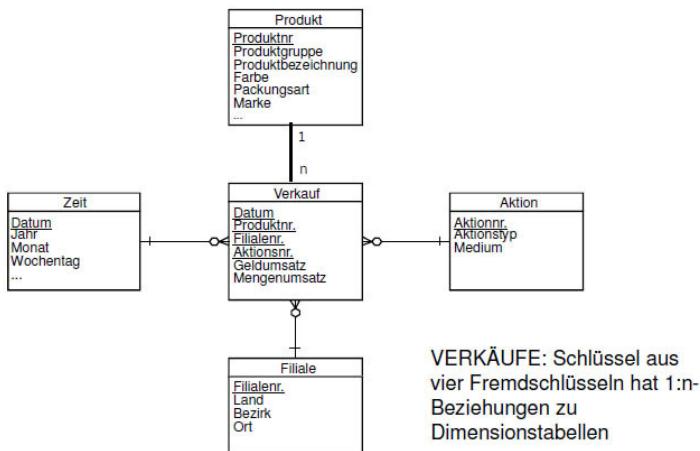
Die Abfrage: „Alle Verkäufe der Produktgruppe BMW für jeden Shop für ein Jahr“ benötigt nur noch **3 Joins!** Weniger Joins bedeutet auch, dass das Sternschema schneller als das Snowflake Schema ist. Im Sternschema ist die Anzahl Joins unabhängig von der Länge der Verdichtung in der Anfrage. Sie steigt linear mit der Anzahl Dimensionen in der Anfrage. Das Sternschema hat folgende Eigenschaften:

- Mehrere Dimensionstabellen beziehen sich auf eine Faktentabelle
- Faktentabelle enthält Attribute, welche die Kennzahlen messen
- Jede Dimensionstabelle steht in einer 1:n Beziehung zu einer Faktentabelle.
- Die Faktentabelle enthält einen Primärschlüssel und n Fremdschlüsselelemente je einen für jede Dimensionstabelle.
- Die Faktentabelle integriert m:n Beziehungen in einer einzigen Tabelle und hat deshalb viel Redundanz (Im Gegensatz zu Snowflake).

##### Sternschema allgemein\*



## Beispiel Sternschema Tabellen für den Handel



Jeder Zeile einer Dimensionstabelle sind in der Regel mehrere Zeilen der Faktentabelle zugeordnet. Denn das Mountainike 231 kann ja mehrmals verkauft werden:

Dimensionstabelle: PRODUKT

Produkt#	Produktkategorie	Produktname	Farbe	...
231	Mountain Bike	„Colorado“	grün	...
...	...	...	...	...

Faktentabelle: VERKAUF

Datum	Produkt#	Filiale#	Aktion#	Mengenumsatz	...
2.3.98	231	14	3	1863	...
3.3.98	231	14	4	533	...
...	...	...	...	...	...

### 4.2.8 Entwicklung eines Sternschemas

#### ▪ 1. Betriebliche Anforderungen sammeln

Der Aufbau des Sternschemas muss definiert werden. Ziel bestimmen: Welche **Fakten** interessieren nach welchen **Dimensionen**? Welches sind die verfügbaren Daten und welche sind die Zielauswertungen?

Beispiele	Fakt pro Dimension
Zeitvergleiche	VERKÄUFE pro PERIODE
Produktvergleich	VERKÄUFE pro PRODUKT
Lieferantenvergleich	VERKÄUFE pro LIEFERANT
Warenkorbanalyse	VERKÄUFE pro KUNDE

Beispiel Handel: Verfügbare Daten aus den operativen Systemen könnten Mengenumsatz, Geldumsatz, Kunden etc. sein. Mögliche Zielauswertungen könnten die folgenden sein:

#### ▪ 2. Anforderungsdiagramm erstellen

Die Spezifikation eines Sternschemas lässt sich in einem Anforderungsdiagramm zusammenfassen. Folgende Punkte müssen spezifiziert werden: **Erforderliche Kennzahlen** (Attribute der Faktentabelle, die das Ergebnis einer Unternehmenseinheit bewerten.), **Dimensionen** (Attribute entlang welcher Kennzahlen gemessen werden), **Klassifikationshierarchien** (Wertebereich der Dimension)

#### ▪ 3. Resultat

1 Faktentabelle und wenige Dimensionstabellen (ca. 5 – 15)

Für grössere Unternehmen entstehen ca. 10 – 25 Sternschemata

- **Beispiel eines möglichen Anforderungsdiagramms**

**Kennzahlen:** Mengenumsatz, Geldumsatz, Kundenzahl ...

(Anmerkung Autor: Eine mögliche Prüfungsfrage wäre das Erstellen eines solchen Diagramms)

Dimensionen	Klassifikationshierarchien
Produkt	Produktgruppe, Farbe, Marke, ...
Periode	Jahr, Quartal, Monat, ...
Ort	Land, Region, Filiale, ...
...	

#### 4.2.9 Evaluation Sternschema

- **Vorteile**

- Die Bildung von Verdichtungen wird optimal unterstützt.
- Browsing-Funktionalität wird unterstützt.
- Keine redundanten Einträge in den Dimensionstabellen.

- **Nachteile**

- Erhöhte Anzahl an physikalischen RDBMS-Joins
- Erhöhte Anzahl physikalischer DWH-Tabellen
- Höherer Aufwand bei der Wartung (komplexer Ladeprozess)
- Komplexere SQL-Befehlsgenerierung

#### 4.2.10 Evaluation Sternschema

- **Einfachheit**

- Wenige Tabellen und einfache Anfragen (wenige Joins)
- Konzentration auf wesentliche Aussagen
- Eine Faktentabelle liefert nur eine Grundaussage

- **Verständlichkeit**

- Betriebsnähe mit Fakten und Dimensionen (Modernisierung des Berichtswesens)
- Datensammlung zur Trenderkennung
- Datensammlung für Data Mining

- **Nachteile**

- unvollständig: bilden zum Beispiel nicht alle Beziehungen ab
- wartungsaufwändig: insbesondere änderungsfeindlich
- Redundante Einträge in den Dimensionstabellen

#### 4.2.11 SQL 2003 und OLAP-Erweiterung

Der SQL Standard 2003 bringt vor allem im OLAP Bereich wichtige Erweiterungen für das Standard SQL:

- GROUP BY ROLLUP
- GROUP BY CUBE
- GROUP BY GROUPING SETS

Die folgenden Grafiken bauen auf dieser Beispieldatenebene auf →

Tore		
Spieler	Saison	Anzahl
Elber	1998	11
Elber	1999	13
Elber	2000	14
Filber	2001	15
Scholl	1996	10
Scholl	1997	5
Scholl	1998	9
Scholl	1999	4
Scholl	2000	6
Scholl	2001	9

## ▪ SQL 2003 und OLAP – Beispiel mit GROUP BY ROLLUP

„Anzahl Tore pro Spieler für alle Saisons.“

```
SELECT Spieler, Saison,
SUM (Anzahl) AS Tore
FROM Tore
GROUP BY ROLLUP (Spieler, Saison);
Anzahl Tore pro Spieler für alle Saisons
```

entspricht

```
SELECT Spieler, Saison, Anzahl AS Tore
FROM Tore
UNION ALL
SELECT Spieler, NULL AS Saison,
SUM(Anzahl) AS Tore
FROM Tore
GROUP BY Spieler
UNION ALL
SELECT NULL AS Spieler, NULL AS Saison,
SUM(Anzahl) AS Tore
FROM Tore
ORDER BY Spieler, Saison;
```

Spieler	Saison	Tore
Elber	1998	11
Elber	1999	13
Elber	2000	14
Elber	2001	15
Elber	NULL	53
Scholl	1996	10
Scholl	1997	5
Scholl	1998	9
Scholl	1999	4
Scholl	2000	6
Scholl	2001	9
Scholl	NULL	43
NULL	NULL	96

## ▪ SQL 2003 und OLAP – Beispiel mit GROUP BY CUBE

„Anzahl Tore pro Saison für alle Spieler.“

```
SELECT Spieler, Saison,
SUM (Anzahl) AS Tore
FROM Tore
GROUP BY CUBE (Spieler, Saison);
Anzahl Tore pro Saison für alle Spieler
```

entspricht

```
SELECT Spieler, Saison, Anzahl AS Tore
FROM Tore
UNION ALL
SELECT Spieler, NULL AS Saison,
SUM(Anzahl) AS Tore
FROM Tore GROUP BY Spieler
UNION ALL
SELECT NULL AS Spieler, Saison,
SUM(Anzahl) AS Tore
FROM Tore
UNION ALL
SELECT NULL AS Spieler, NULL AS
Saison, SUM(Anzahl) AS Tore
FROM Tore
ORDER BY Spieler, Saison;
```

© 2010 | Dr. Georgios Grivas | Data Warehousing | 28

Spieler	Saison	Tore
Elber	1998	11
Elber	1999	13
Elber	2000	14
Elber	2001	15
Elber	NULL	53
Scholl	1996	10
Scholl	1997	5
Scholl	1998	9
Scholl	1999	4
Scholl	2000	6
Scholl	2001	9
Scholl	NULL	43
NULL	1996	10
NULL	1997	5
NULL	1998	20
NULL	1999	17
NULL	2000	20
NULL	2001	24
NULL	NULL	96

## ▪ SQL 2003 und OLAP – Beispiel mit GROUP BY GROUPING SETS

```
SELECT Spieler, Saison,
SUM (Anzahl) AS Tore
FROM Tore
GROUP BY GROUPING SETS ((Spieler), (Saison));
entspricht
```

```
SELECT Spieler, NULL AS Saison,
SUM(Anzahl) AS Tore
FROM Tore GROUP BY Spieler
UNION ALL
SELECT NULL AS Spieler, Saison, SUM(Anzahl) AS Tore
FROM Tore
GROUP BY Saison;
```

Spieler	Saison	Tore
Elber	NULL	53
Scholl	NULL	43
NULL	1996	10
NULL	1997	5
NULL	1998	20
NULL	1999	17
NULL	2000	20
NULL	2001	24

# Kurs 2: Das Data-Warehouse-Projekt

## 5 Kapitel 1: Das Data-Warehouse-Projekt

### 5.1 DWH Projekt- und Changemanagement

#### 5.1.1 Projekt, Programm, Projektportfolio

- **Projekt:** Ein Projekt ist ein komplexes Vorhaben welches **einmaligen** Charakter, und immer ein **Beginn** und **Ende** hat. Es hat eine begrenzte personelle und finanzielle Ressource.
- **Programm:** Programme bestehen aus Projekten , welche das **gleiche strategische Ziel** verfolgen, Abhängigkeiten von einander aufweisen und deshalb unter **gemeinsamer Programmleitung** stehen. Projekte kommen nur in einem Programm vor.
- **Projektportfolio:** Das Projektportfolio beinhaltet **alle in einem Unternehmen** laufenden und anstehenden **Projekte**. Das Portfolio beinhaltet auch die Abhängigkeiten zwischen den verschiedenen Projekten.
- **Projektportfoliomangement:** Es verwaltet das Projektportfolio einer Unternehmung.
- **DWH-Programm:** Durch die Definition eines DWH-Programms kann man ein grosses Vorhaben in DWH-Projekte unterteilen, somit bleibt das strategische Ziel des Programmes über alle DWH-Projekte dasselbe. Mögliche Beispiele von DWH-Projekten: Unabhängige Data Marts, Entwicklung von konkreten Reports welcher einer Business-Einheit helfen können.

#### 5.1.2 Projektvorgehensmodelle

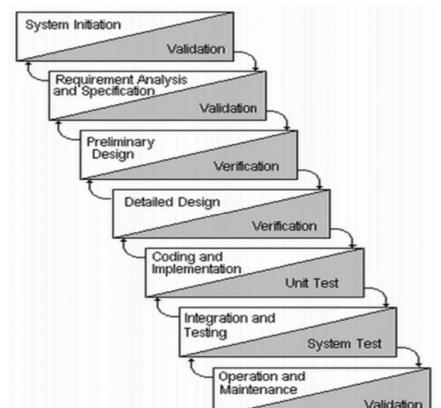
Da Data-Warehouse-Projekte Softwareentwicklungsprojekte sind, kommen die typischen klassischen Vorgehensmodelle, sowie ich die Reifegradmodelle zum Einsatz:

- **Klassische Vorgehensmodelle:** Wasserfallmodell, Prototypenmodell, Evolutionäres/Iteratives Modell, Spiralmodell, Mischformen.
- **Reifegradmodelle:** CMMI (Capability Maturity Model Integration), SPICE (Software Process Improvement and Capability Determination).

#### ▪ **Wasserfallmodell mit Iterationen**

Das Wasserfallmodell erlaubt **keine Rücksprünge** in eine vorherige Phase. Werden Anforderungen geändert erhält man am Schluss ein Produkt, welches den Anforderungen nicht gerecht wird, oder man beginnt komplett neu.

Das **erweiterte Modell** ermöglicht, dass lediglich die Arbeiten der zurückgesprungenen Phase verworfen werden müssen.



## ▪ Spiralmodell

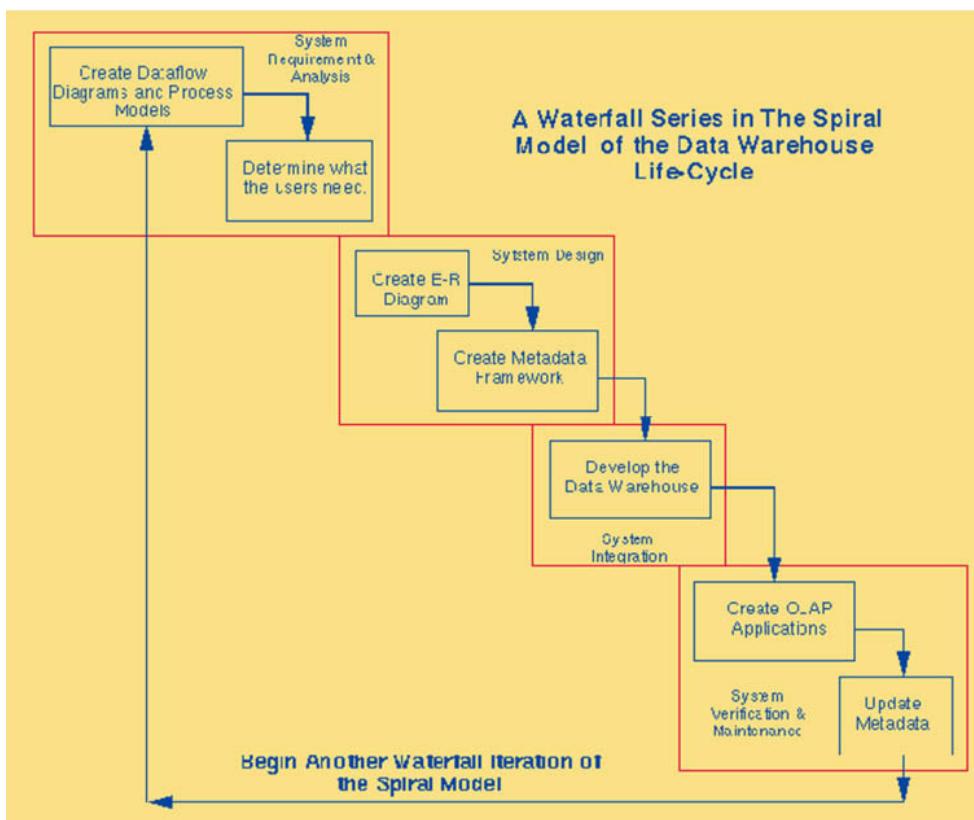
Der Entwicklungsprozess ist ein iterativer Prozess, wobei jeder Zyklus in den einzelnen Quadranten folgende Aktivitäten enthält:

1. Festlegung von Zielen, Identifikation von Alternativen und Beschreibung von Rahmenbedingungen
2. Evaluierung der Alternativen und das Erkennen, Abschätzen und Reduzieren von Risiken, z.B. durch Analysen, Simulationen oder Prototyping
3. Realisierung und Überprüfung des Zwischenprodukts
4. Planung des nächsten Zyklus der Projektfortsetzung.



## ▪ DWH-Projektvorgehen: Wasserfall-Kaskade im Spiralmodell

Einer der wichtigsten Punkte beim Entwicklungsprozess ist, dass man das System aus der Sicht des Lebenszyklus entwickelt. Da das Entwerfen und erstellen eines DWH ein iterativer Prozess ist, eignet sich die Spiral Methode am besten. Das folgende Diagramm zeigt eine Wasserfall Serie in einem vorgeschlagenen Spiral Modell eines DWH-Lebenszyklus.

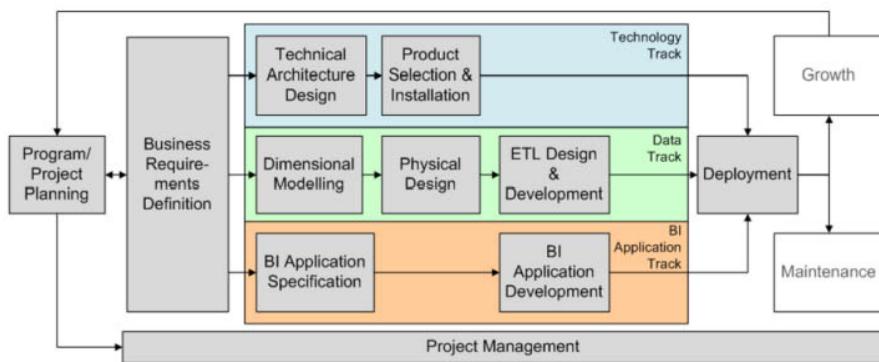


## ▪ DWH-Projektvorgehen nach Kimball, 1998

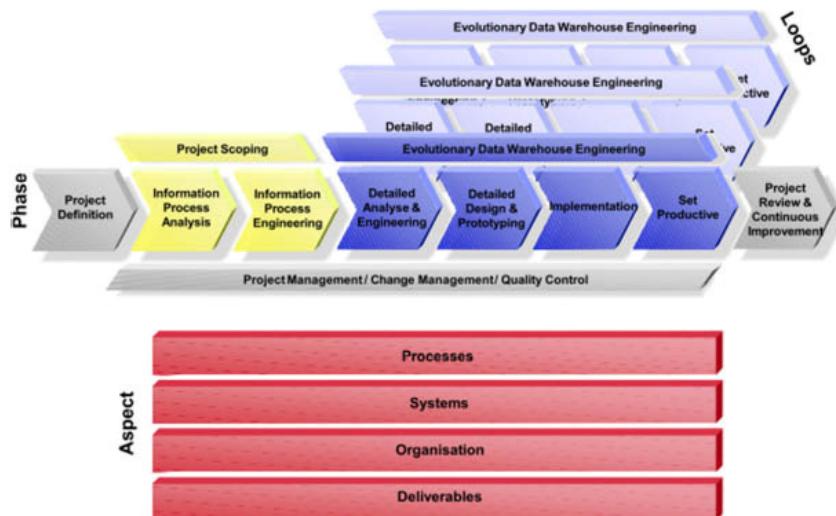
Das Projektvorgehens Modell ist in 3 verschiedene Tracks aufgeteilt:

- **Technology Track:** Hier soll das Framework für das DWH System festgelegt werden, zudem sollen anhand des Technischen Architektur Plans die Komponenten definiert werden, welche für das DWH-Projekt nötig sind.

- **Data Track:** Hier werden die Dimensionen modelliert. Im Physical Design wird die Datenbank entwickelt. In der Phase ETL Design & Development werden die schwierigen ETL-Prozeduren entwickelt.
- **BI Application Track:** Hier designt und wählt man nützliche Applikationen aus, welche das Business unterstützen sollen.

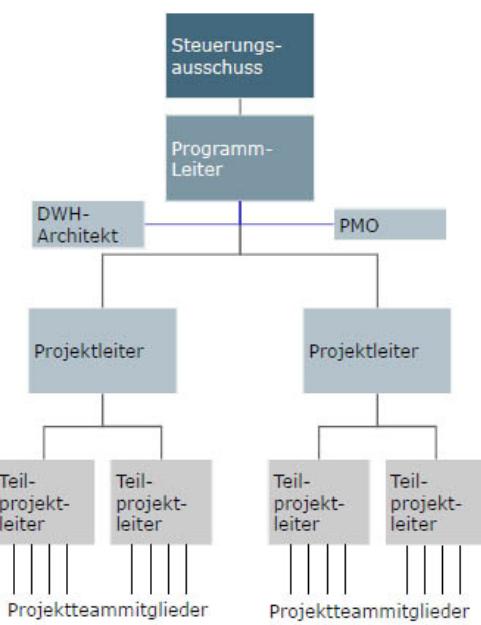


#### ▪ DWH-Projektvorgehen nach PLAUT Consulting



#### 5.1.3 Projektorganisation und Rollen

- **Steuerungsausschuss**
  - Oberstes Entscheidungsgremium
  - Setzt Prioritäten
  - Löst Probleme und Konflikte
- **Auftraggeber**
  - Gesamtverantwortung
  - Leiter des Steuerungsausschusses
  - Budget Owner
- **Auftragnehmer**
  - Bestimmt Programm- und Projektleitung
  - Mitglied des Steuerungsausschusses
  - verantwortlich gegenüber Auftraggeber



- **Programm-/Projektleiter**
  - Programm/Projektsteuerung
  - Verantwortlich für
  - Programm/Projektziele
- **Projektteammitglieder**
  - Führung die einzelnen
  - Projektaufgaben
  - Verantwortlich für die Ziele der
  - Arbeitspakete
  - Steuerungsausschuss

#### 5.1.4 Projektrollen und Themenbereiche

Rolle	Themenbereich
DWH-Programmleiter	Programmmanagement
DWH-Projektleiter	Projektmanagement
DWH-Anwender	Analyse (Standard)
DWH-Architekt	Methoden, Konzepte, Modellierung
DWH-Miner	Konzepte, Analyse (Non-Standard)
Spezialist des Fachbereichs	Betriebliches Fachwissen
DWH-Systementwickler	System- und Metadatenmanagement
	ETL (Extraktion, Transformation, Laden)
	Data Warehouse, Data Marts
PMO	Programm Management Office, Support des Programmleiters und der Projektleitern in der Planung und Steuerung des Programmes

#### 5.1.5 Organizational Change Management (OCM)

Es hat sich gezeigt, dass 75% der gescheiterten Projekte **nicht** aus technischen Gründen gescheitert sind. Vielmehr waren die Probleme: Keine Akzeptanz, Kommunikationsprobleme, Probleme mit den Projektressourcen, kein unternehmensweites Daten Modell möglich etc. Meist sind die organisatorischen Herausforderungen der Veränderung/Change viel grösser als die technischen. Anhand von OCM-Instrumenten lässt sich die Veränderungswirkung auf die Organisation und der Akzeptanz eines Projektes messen.

#### 5.1.6 Konfliktmanagement

Das Konfliktmanagement kann anhand des Spannungsfeldes, des sogenannten Bermuda-Dreiecks aufgezeigt werden:

- Mehr Funktionen und Qualität erhöht die Kosten für Personal und Sachmittel und hat Auswirkung auf die Termine
- Führt zu Motivationsverluste des Projektteams
- Kommunikation zwischen den verschiedenen Parteien unabdingbar
- Konfliktvermeidung steht vor Konfliktmanagement durch systematisches OCM Spannungsfeld des sog. magischen oder Bermuda-Dreiecks



## 5.2 DWH Business Case

Zuerst sollte man sich grundlegende Fragen zum Business Case überlegen. Lohnt sich das Vorhaben? Falls ja, durchführen, ansonsten nicht. Es stellen sich im übrigen folgende Fragen:

1. Was ist das Ziel des DWH? Passt es mit der IT und DWH-Strategie?
2. Welchen Mehrwert erzielen wir
3. Wie hoch werden die Kosten konservativ geschätzt?
4. „Must“ oder „Nice to have“?

<b>Strategiekonformität</b>
Beurteilung der Strategiekonformität aufgrund der IT- und DWH-Strategie
<b>Wirtschaftlichkeit</b>
Beurteilung der DWH-Projekt-Wirtschaftlichkeit (Pay Back, NPV, ROI)

### Projektrisiko

Standardisierte Risiko-Bewertung des DWH-Projektes



### ▪ Wirtschaftlichkeit - Effekte

Der Nutzen in der Wirtschaftlichkeitsbetrachtung ist immer quantifizierbarer Nutzen. Es handelt sich dabei um Effekte des Projektabschlusses auf die Kosten- und Ertragsseite der Unternehmung.

### ▪ Wirtschaftlichkeit - Projektinvestitionen

Die Projektinvestitionen beinhalten alle Projektkosten, sowohl interne Personalkosten als auch externe Beratungskosten. Darüber hinaus fallen Ausbildungskosten (Seminare, Schulungen), Werbekosten, Büromaterial, Einmallizenzen, Spesen (Reise, Verpflegung), Hardware, Mobiliar an. Alle diese Angaben beziehen sich auf die Zukunft (z.B. für die kommenden 10 Jahre).

### Mini Business Case

- Fact Sheet
- Geschäftsidee
  - Kunde / Markt
  - Produkt / Dienstleistung
  - Prozesse / Organisation
  - IT
- Projekt Scope
  - Ziele
  - Rahmenbedingungen
- Projekt Management
  - Projekt Organisation
  - Risiken
- Wirtschaftlichkeit
  - Nutzen
  - Max. Investitionskosten
  - Max. Betriebskosten
- Finanzierung 1. Phase
- Antrag
- Entscheid

### Business Case nach der Vorstudie

- Fact Sheet
- Ausgangslage
- Projekt Scope
  - Ziele
  - Rahmenbedingungen
  - Systemgrenzen
- Lösung
  - Kunde / Markt
  - Produkte / Dienstleistungen
  - Prozesse / Organisation
  - Interne & externe Kommunikation
  - IT
- Projekt Management
  - Projekt Organisation
  - Projekt Aktivitäten and Planung
- Risiken
- Wirtschaftlichkeit
  - Nutzen
  - Investitionskosten
  - Betriebskosten
  - Net Present Value (NPV)
- Finanzierung
- Antrag
- Entscheid

## 5.3 Softwareauswahl

Kommerzielle DWH-Produkte erlauben eine schnelle Realisierung des DWH-Projektes. Jedoch gibt es keine optimale DWH-Produkte. Eine Evaluation ist äusserst wichtig, denn sollte das falsche Produkt gewählt werden, steht das ganze Projekt in Frage. Das Werkzeug für die Endanwender (Analysen) muss benutzerfreundlich sein, was wiederum mit grosser Akzeptanz verbunden ist.

### 5.3.1 Klassifikation der Produkte

#### ▪ Produkte der Datenbeschaffung (ETL-Tools)

ETL ermöglicht die Extraktion, Transformation und das Laden der Daten. Data-Cleansing (Datenbereinigung) Produkte erlauben erweiterte Funktionen, wie das Entfernen und Korrigieren von Datenfehlern in Datenbanken.

#### ▪ Datenbanksystem für das DWH

Oftmals genügen die vorhanden DBMS. Datenbank Management Systeme sind notwendig für:

- Plattform-Portabilität: Unabhängigkeit von Betriebssystem und Hardware
- Skalierbarkeit: Erweiterbarkeit bei steigender Benutzerzahl
- Flexibilität: Support von Analysen und OLTP
- Real-Time-DWH: Veränderte operative Daten gelangen ohne Verzögerung ins DWH
- Operational BI: BI in operative Prozesse integrierbar

### 5.3.2 Projektvorgehen zur Softwareauswahl

Um bei der Software bzw. Produktauswahl erfolgreich zu sein gilt es folgende Punkte zu beachten. Man sollte das Produkt immer objektiv bewerte und wenn möglich alle relevanten Personengruppen bei der Anforderungsaufnahme mit einbeziehen. Man sollte frühzeitig mit der Kostenkalkulation beginnen (Lizenz- Wartungskosten etc.).



## 5.4 Herausforderungen und Erfolgsfaktoren

Möchte man die grösste, komplexeste und funktionell all-inclusive DWH-Lösung, führt dies automatisch zu der letzten, neusten und teuersten Technologie welche von einem Hersteller angeboten wird. Der Aufbau eines DWH entspricht nicht dem Aufbau der grössten Datenbank oder intelligentesten Technologie, sondern ist oftmals das Zusammenführen von verschiedenen, z.T. gut etablierten Systemen. Dies in einer Art und Weise, dass Sie die Analyse-Anforderungen einer Organisation erfüllen.

### 5.4.1 Herausforderungen im Projektmanagement

- Hohe Komplexität
- Heterogenität der Quelldaten
- Geografisch verteilte Datenquellen
- Unklare Anforderungen
- Unterschiedliche Systeme der Quelldaten
- Integration in die Basisdatenbank

### 5.4.2 Herausforderungen bei DWH-Projekten

Die meisten DWH-Projekte scheitern nicht an der Technik. Für ein erfolgreiches Projekt sollten folgende Punkte geklärt werden:

- Besteht „Strong Management Support“
- Erzeugt das Projekt einen Mehrwert? ROI muss deutlich sein.
- Das DWH ist abhängig von Daten, also auch von den Datenherren
- Man muss Fragen stellen wollen und für Antworten bereit sein
- Feasibility (Machbarkeit): Existieren Daten, können diese verwendet werden?

Budget-überschreitung	<ul style="list-style-type: none"> <li><b>DWH Projekte sind teuer</b></li> <li>Viele Abteilungen / Systeme betroffen, Hardwareanforderungen, semantische Integration</li> </ul>
Zeit-überschreitung	<ul style="list-style-type: none"> <li><b>Viele Stakeholder</b> müssen befragt werden</li> <li>Folge: Prototypen werden weggelassen</li> </ul>
Fehlende Funktionen	<ul style="list-style-type: none"> <li><b>Falsche Einschätzung der / fehlender Konsens über wichtigen Funktionen</b> (man macht die einfachen zuerst, nicht die wichtigen)</li> <li>Typisch: Zeitüberschreitung mit Funktionsreduzierung begegnen, schließlich Funktionen ganz kippen</li> </ul>
Keine Akzeptanz bei Benutzern	<ul style="list-style-type: none"> <li>Zu hohe Erwartungen</li> <li>Zu viele Versprechungen</li> <li><b>DWH weckt Ideen</b> – „Mensch, da könnte man doch auch noch...“</li> </ul>

Performanzprobleme	<ul style="list-style-type: none"> <li>Zwei Komponenten: Anfragen und ETL</li> <li><b>ETL-Zeiten häufig unterschätzt</b></li> <li>Unrealistisch kleine Prototypen</li> <li>Werkzeuge f. ad-hoc Queries; point&amp;click mit unvorhersehbaren Auswirkungen</li> <li>Fehlendes Performance Bewusstsein bei Anwendern</li> <li>Fehlendes Performance-Auditing</li> <li>Fehlende Vereinbarungen – welche Queries werden wann laufen, bei welchen Datenmengen, wie lange darf es dauern, ...</li> <li><b>Große Datenmengen verlangen eigene Methoden</b> (Modell, Implementierung, Indexe, Prozesse, ...)</li> <li><b>Design for the Large</b></li> <li><b>Ohne Vorgaben ist eine Query nie ausreichend schnell!</b></li> </ul>
--------------------	--

Verfügbarkeitsprobleme	<ul style="list-style-type: none"> <li>Geplante versus ungeplante Downtimes</li> <li><b>DWH oft bei ETL Zyklen gesperrt</b></li> <li>Komplexe ad-hoc Queries erwecken Eindruck fehlender Verfügbarkeit</li> </ul>
Fehlende Erweiterbarkeit	<ul style="list-style-type: none"> <li>Mehr Daten, mehr Benutzer, mehr Analysen, mehr Quellen, mehr Dimensionen, größere Zeitraum, ...</li> <li><b>Gerade ein hochoptimiertes Design ist schwierig zu erweitern</b></li> </ul>
Schlechte Datenqualität	<ul style="list-style-type: none"> <li>Falsche Reports sind schlechter als keine Reports!</li> <li>Datenqualitätsprobleme werden unterschätzt und lange ignoriert</li> <li><b>Die Fehler der alten Reports kennt man seit Jahren, die der neuen will man nicht verzeihen</b></li> <li>Fehlender Mut zum Daten-weglassen</li> <li>Oftmals <b>Business Process Reengineering</b> notwendig</li> </ul>

Zu kompliziert	<ul style="list-style-type: none"> <li><b>Was-wäre-wenn Analysen, mehrdimensional aggregierte Reports, Data Mining, etc. sind hochkomplexe Themen</b></li> <li>Verlass auf DWH-GUIs kann User in die Irre führen</li> <li>Unterschiedliche Benutzer – unterschiedliche Anforderungen</li> </ul>
Fehlende Kosteneffizienz	<ul style="list-style-type: none"> <li>Preis / Nutzen Verhältnis</li> <li>Nutzen schwierig quantifizierbar; liegt eher beim Management</li> <li>ROI von DWH Projekten schwierig zu berechnen</li> </ul>

### 5.4.3 Erfolgsfaktoren in DWH-Projekten

- Das Projektmanagement ist der entscheidender Erfolgsfaktor in einem DWH-Projekt
- Der Projektleiter kommt idealerweise aus dem Business, da fachspezifische Problemstellungen zu lösen sind
- Der Sponsor kommt aus dem Topmanagement, muss respektiert werden und hat eine gesunde Skepsis gegenüber der Technologie
- Involvieren der End-Users von Anfang an klare Formulierung ihrer Informationsbedarf und Funktionsanforderungen
- Bei einer Wahl eines externen Beraters, sind das fundierte Wissen (nicht nur über die Tools), eine ausreichende relevante Projekterfahrung und das betriebswirtschaftliche Know-how massgebend
- Abgestimmtes Projektvorgehensmodell
- Übertriebene Anforderungen sollten auf das Machbare reduziert werden
- Aufbau eines unternehmensweiten gemeinsamen Begriffsverständnis
- Think big – start small: stufenweise Aufbau des DWH mithilfe kleiner Teilbereiche
- Konsequentes OCM (Organizational Change Management) führt zu:
  - Internes Marketing unter Berücksichtigung der organisatorischen Auswirkungen
  - Berücksichtigung der Interessen aller Beteiligten
  - Identifikation der Projektgegner
  - Massnahmen für die Sicherstellung des Projekterfolgs
- Sorgfältig erarbeitete Business Case inkl. Projektbeschreibung und – identifikation
- Nutzen des DWH-Projektes klar beschrieben, so dass nach Abschluss des Projektes nachweisbar ist
- Anbindung von neuen Datenquellen sollte gut geplant sein