

# Neuere Konzepte von Informationssystemen - Teil II

Kapitel 1: Data-Warehousing-Architektur

# Analyse von Geschäftsprozessen

---

## ■ Mögliche Fragestellungen

- Wie entwickelt sich unser Umsatz im Vergleich zum letzten Jahr?
- In welchen Regionen liegt der erzielte Gewinn unterhalb der Erwartungen?
- Mit welchen Produktgruppen erzielen wir den größten Gewinn?
- Ich möchte wöchentlich einen Bericht über die wichtigsten Kennzahlen und deren Veränderung in den letzten Wochen!
- Welchen durchschnittlichen Umsatz erwirtschaften wir pro Filiale?

# Herausforderungen

- Die benötigten Daten
  - sind eventuell auf die verschiedenen Filialen **verteilt**,
  - werden teilweise **unterschiedlich benannt**,
  - müssen aus **verschiedenen Anwendungen** extrahiert werden,
  - liegen in **unterschiedlicher Granularität** (Lebensmittel – Milchprodukte, Obst & Gemüse) oder in **unterschiedlicher Struktur** vor,
  - sind
    - **fehlerbehaftet** oder **unvollständig**,
    - bereits **gelöscht** oder auf Band **archiviert**.
- Analyse-Anfragen
  - **stören** möglicherweise **den laufenden Betrieb**

# Früher ...

---

- Telefonaktionen
- Taschenrechner
- Listen
- ...
- hoher zeitlicher Aufwand
- hoher Personalaufwand
- hohe Kosten
- hohe Fehleranfälligkeit
- schwierige Nachvollziehbarkeit

# Ziel

---

- Daten sollen
  - **einheitlich benannt** sein,
  - **einheitliche Bedeutung** besitzen.
- Zugriff soll
  - **jederzeit**, möglichst **aktuell** und **schnell** möglich sein,
  - auch **komplexe Fragestellungen** erlauben,
  - **laufenden Betrieb** nicht beeinträchtigen,
  - auch auf **historische Daten** möglich sein,
  - **automatisiert** ablaufen,
  - **nachvollziehbar** sein,
  - **zuverlässig** erfolgen.

# Definition

---

“A **Data Warehouse** is a

- **subject oriented,**
- **integrated,**
- **non-volatile,** and
- **time variant**

collection of data in support of management's **decision-making** process.“

(Inmon, 1996)

# Definition

---

- **subject oriented**  
themenbezogen, nicht auf einzelne Anwendungen bezogen
- **integrated**  
integriert Daten aus verschiedenen Systemen und Standorten
- **non-volatile**  
dauerhafte Speicherung der Daten, kein Ändern, kein Löschen von Daten
- **time variant**  
zeitpunktbezogene Speicherung, häufig Vergleiche von Werten zu verschiedenen Zeitpunkten

# transkational vs. analytisch

- Transaktionale/Operative Systeme
  - Benutzer: Sachbearbeiter, Verkäufer
  - dienen der täglichen Arbeit
  - **OLTP** = OnLine Transactional Processing
- Analytische/Entscheidungsunterstützende Systeme
  - Benutzer: Analysten, Management
  - helfen, strategische Entscheidungen zu fällen
  - Basis für alle entscheidungsunterstützenden Anwendungen, wie z. B.
    - Reporting
    - **OLAP** = (OnLine Analytical Processing)
    - Data Mining
  - DSS = Decision Support System
  - EUS = Entscheidungsunterstützendes System



# Analytische Systeme

---

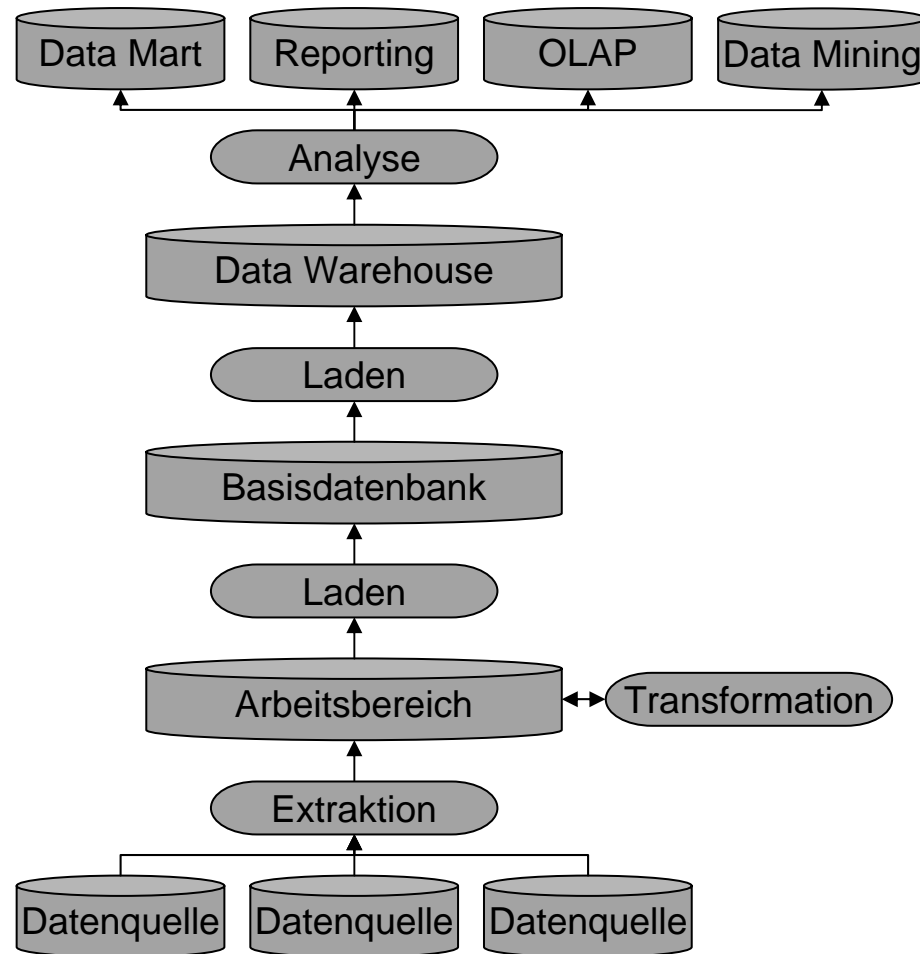
## ■ Eigenschaften:

- enthalten sehr große Datenmengen (möglichst alle relevanten Unternehmensdaten) über mehrere Jahre
- wenige Benutzer und Zugriffe, aber mit hohem Datenvolumen
- keine Änderungen oder Löschungen von Einträgen
- überwiegend historische, zusammengefasste Daten
- "Schnappschüsse" der operativen Daten
- relativ hohe Redundanz
- Daten strukturiert, integriert und konsolidiert

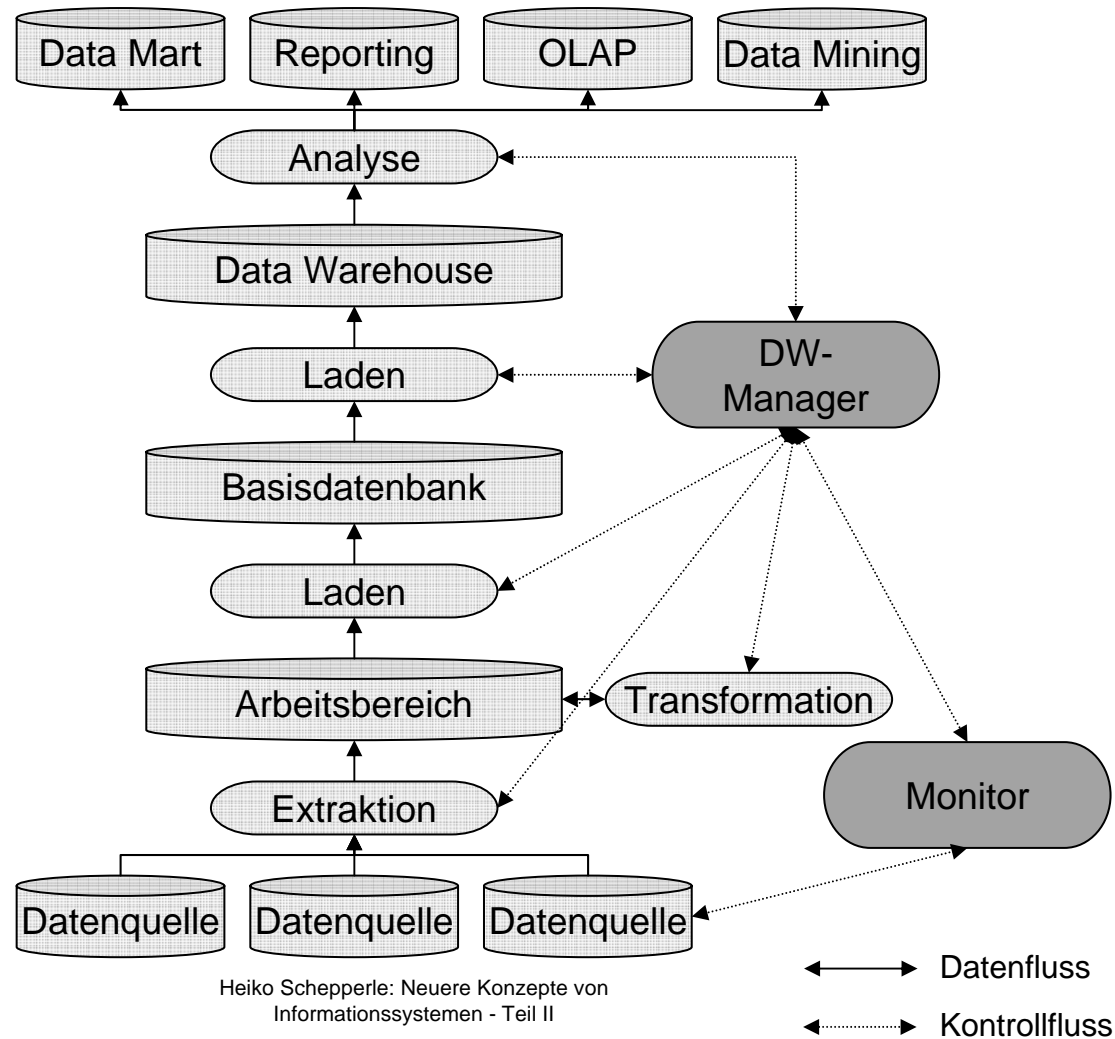
# OLTP vs. OLAP

	<b>OLTP (transaktional)</b>	<b>OLAP (analytisch)</b>
Fokus	Lesen, Schreiben, Modifizieren, Löschen	Lesen, periodisches Hinzufügen
Transaktionsdauer und -typ	kurze Lese-/Schreibtransaktionen	lange Lesetransaktionen
Anfragestruktur	einfach strukturiert	komplex
Datenvolumen einer Anfrage	wenige Datensätze	viele Datensätze
Datenvolumen eines Systems	Gigabyte (GB) – Terabyte (TB)	Terabyte (TB) – Petabyte (PB)
Datenalter	aktuell - mehrere Monate	aktuell - viele Jahre
Datenmodell	anfrageflexibles Datenmodell	analysebezogenes Datenmodell
Verfügbarkeit	sehr hoch	hoch
Anzahl Benutzer	hoch	gering

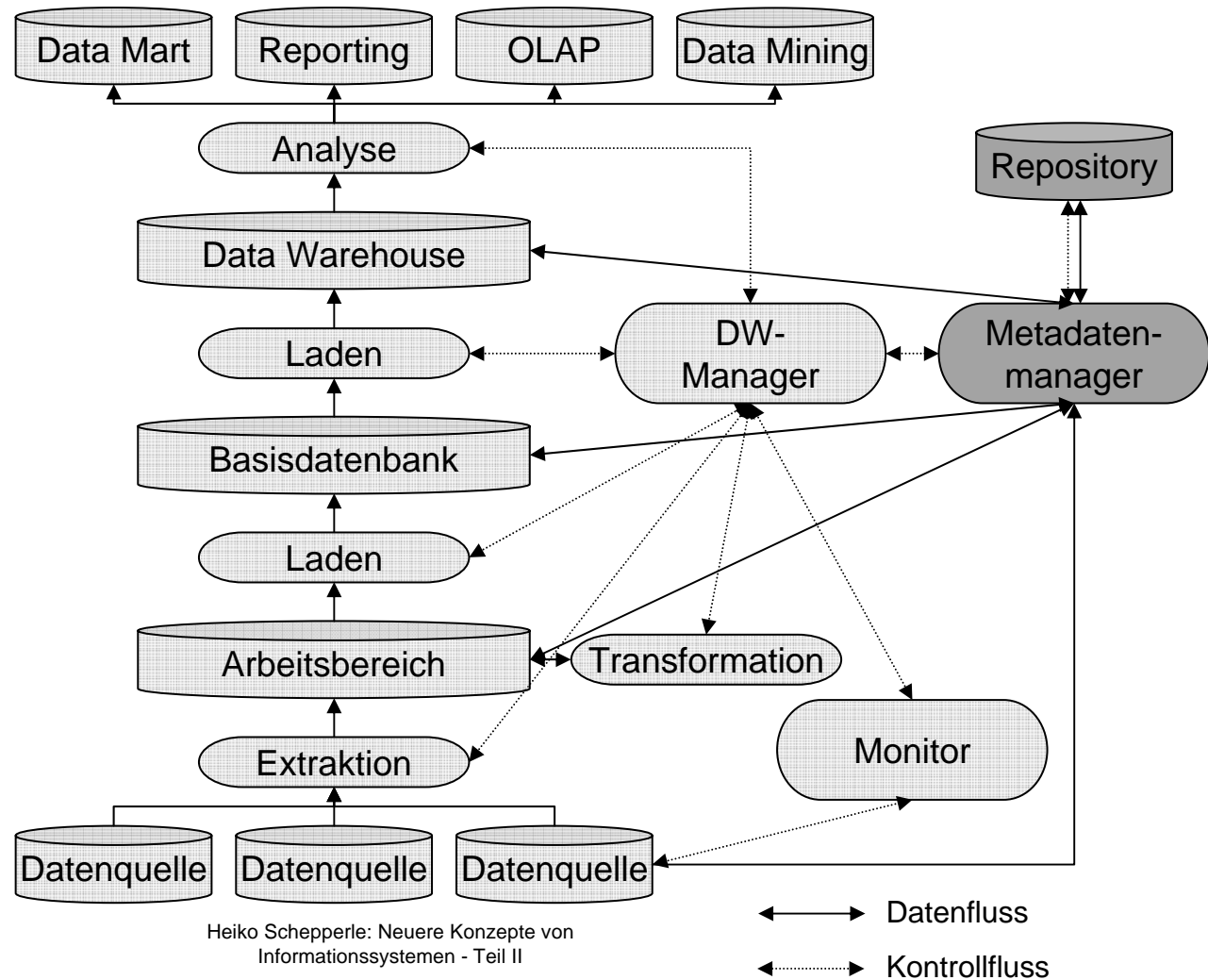
# Referenzarchitektur eines DWS



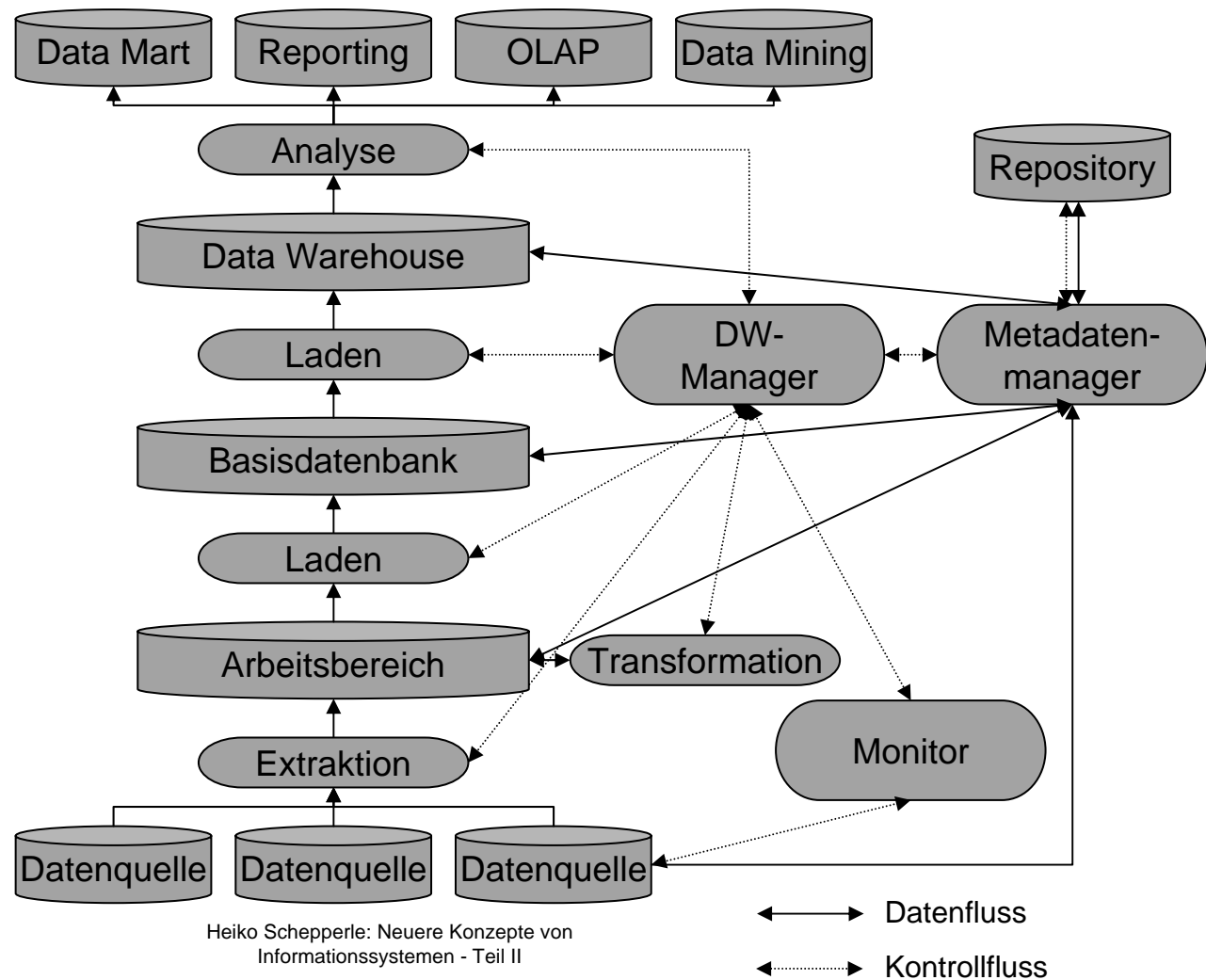
# Data-Warehouse-Manager



# Metadatenmanager



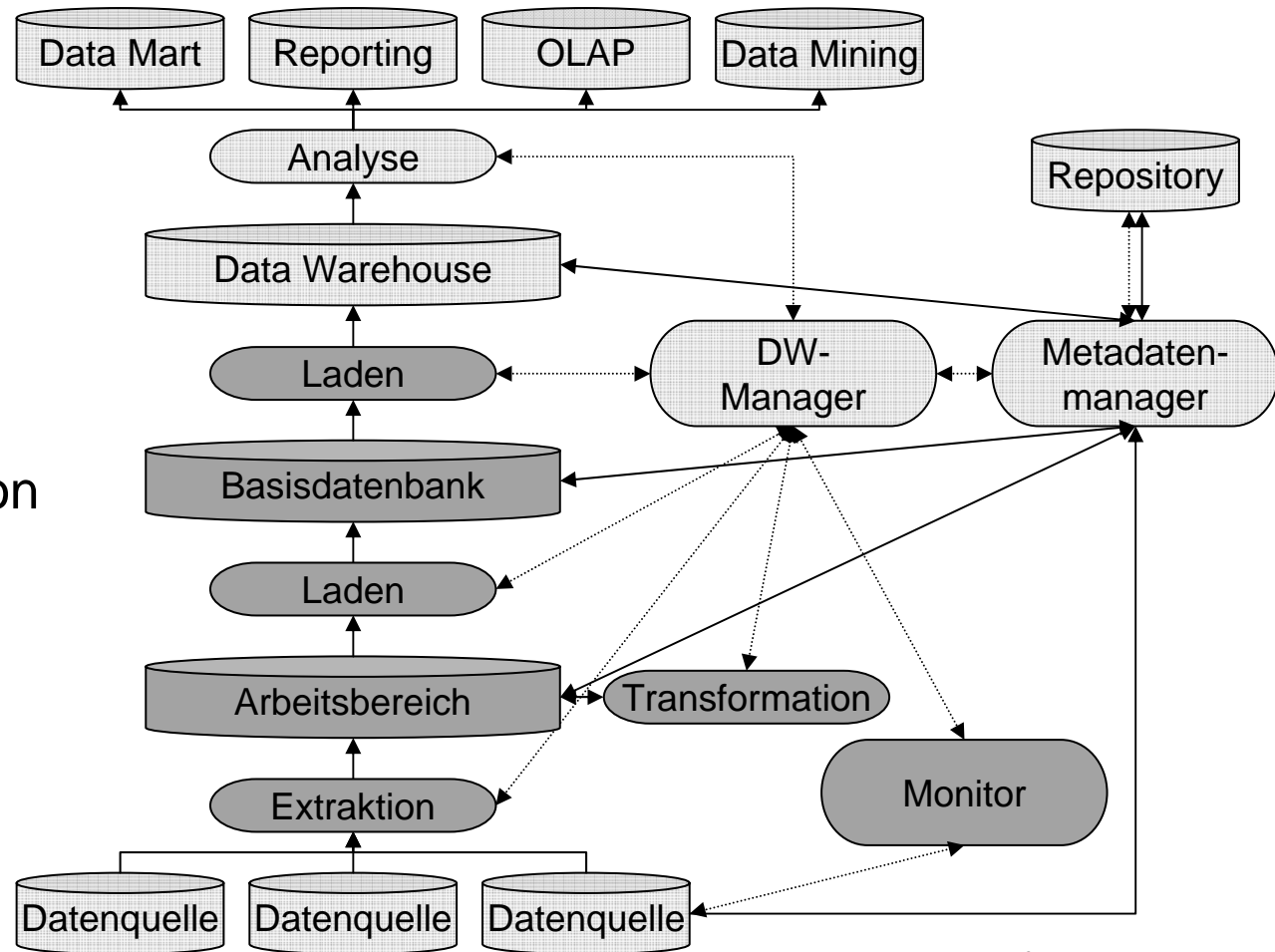
# Referenzarchitektur eines DWS



# ETL-Prozess

ETL =

- Extraktion
- Transformation
- Laden



↔ Datenfluss  
 - - - - - Kontrollfluss

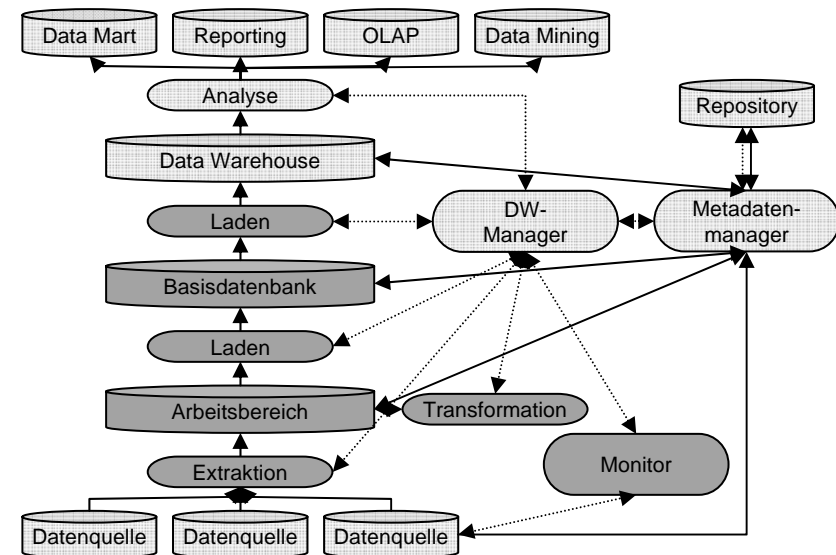
# ETL-Prozess: Extraktion

## ■ Extrahieren der Daten aus den Quellsystemen

- Wann?
  - periodisch
  - auf Anfrage
  - ereignisgesteuert (z. B. nach x Änderungen)
  - sofort
  
- Wie?
  - vollständig
  - inkrementell
  
- Wie bekommt man Änderungen mit?
  - Trigger
  - SQL-Anfrage
  - Zeitstempel-basiert
  - Snapshot-Vergleich

## ■ Unterschiedliche Quellen

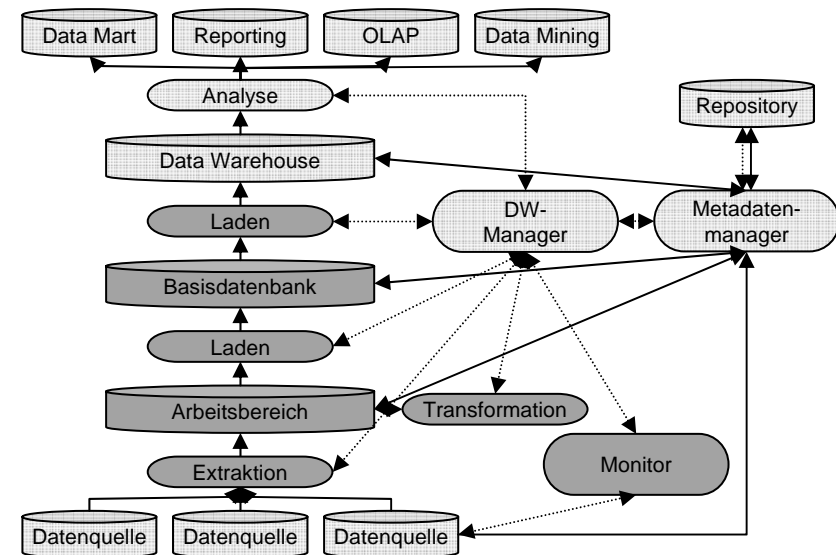
- Datenbanken
- Webseiten
- Dateien
- ...





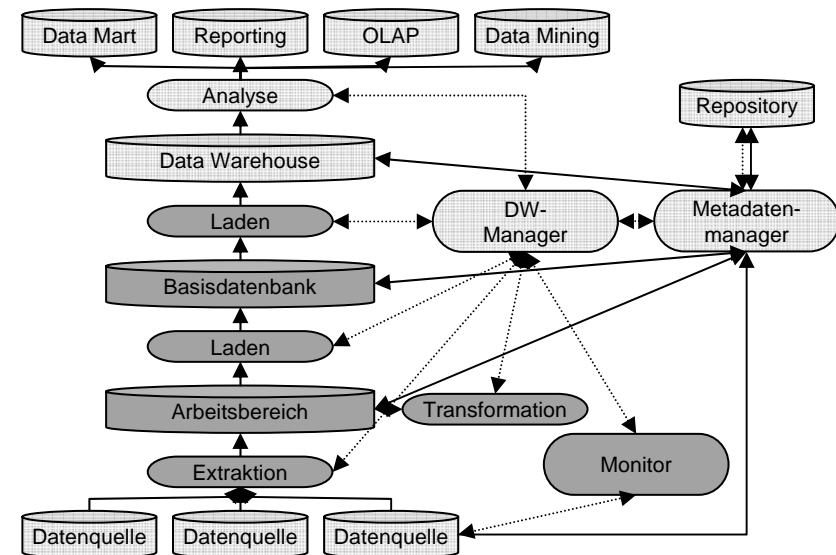
# ETL-Prozess: Transformation #1

- Entfernen irrelevanter Daten
  - Säubern der Daten
    - Ergänzung fehlender Werte
    - Belegung mit default-Werten
    - Prüfung auf falsche Werte (Plausibilitätsprüfung)
  - Überbrückung struktureller Differenzen
- Beispiel: Speditionsfilialen:
- 1 Tabelle pro Filiale: Gut, Menge
    - 1 Zeile pro Gut
  - 1 Tabelle pro Gut: Auftraggeber, Menge
    - 1 Zeile pro Auftraggeber
  - 1 Tabelle pro Filiale: Kohle, Öl, ...
    - 1 Zeile pro Verkaufstag



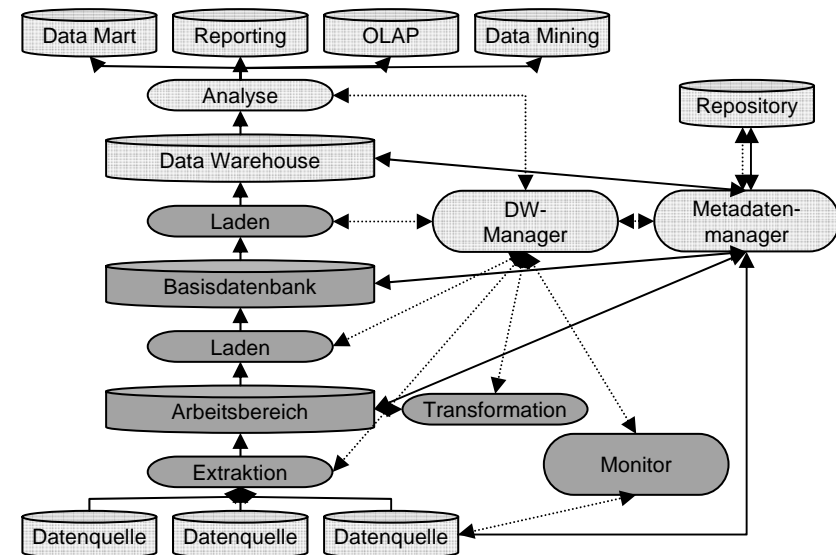
# ETL-Prozess: Transformation #2

- Anpassen der Werte
  - Übersetzungen zwischen Bezugssystemen
    - Maßeinheiten (inch/m, EUR/\$)
    - Kodierungen (ASCII/Unicode)
    - Datumsangaben (D/US)
  - Anpassungen bei semantischen Differenzen
    - Umsatz brutto oder netto
    - Gewinn vor Steuern, nach Steuern
    - mit/ohne Sonderbelastungen
    - ...



# ETL-Prozess: Laden

- Laden von analyseunabhängigen Detaildaten in die Basisdatenbank
- Laden von analyseabhängigen Daten in das Data Warehouse
- ➔ Definition und Befüllen der Basistabellen
- ➔ Aufbau und Wartung von Indexstrukturen
- ➔ Anpassen abgeleiteter Sichten



# Basisdatenbank

## ■ Eigenschaften

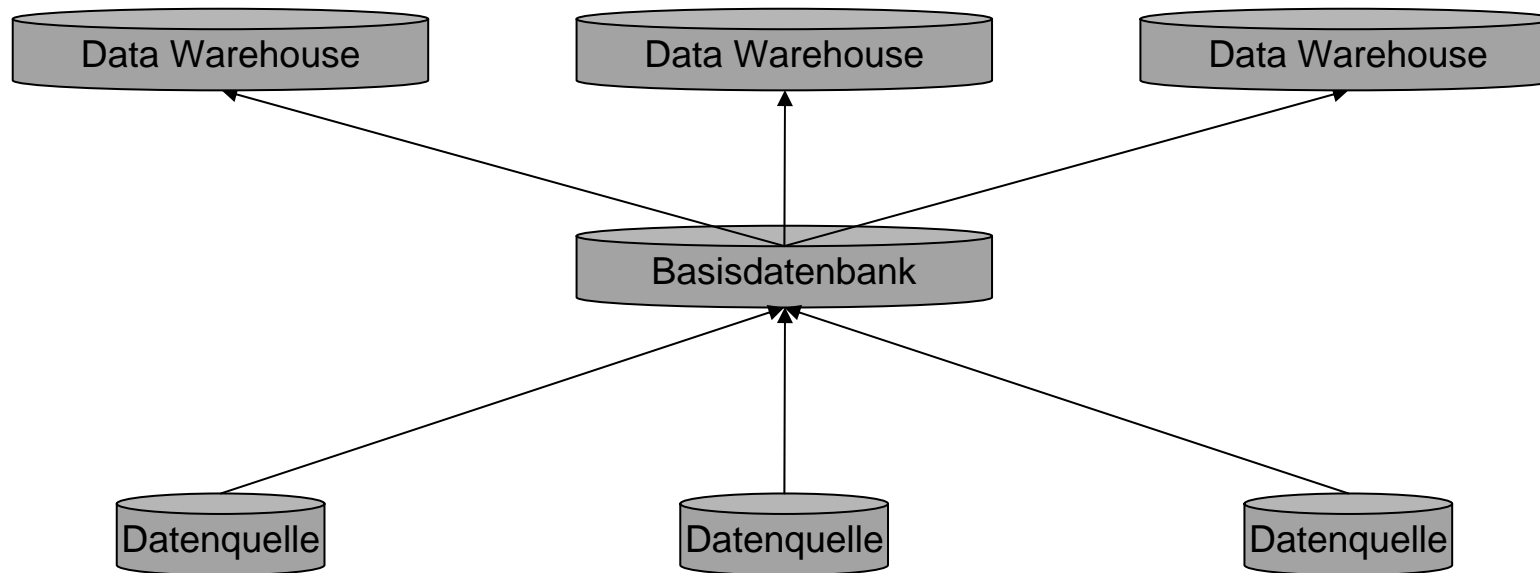
- integrierte Sicht
- umfassend bezüglich Zeit und Granularität
- Modellierung und Optimierung anwendungsneutral
- Daten werden nach definierter Zeit in ein Data Warehouse übertragen
- Aktualisierung zu beliebigem Zeitpunkt möglich
- Daten liegen bereinigt vor

## ■ Funktion

- Sammel- und Integrationsfunktion  
(logisch) zentrales Datenlager
- Distributionsfunktion  
Versorgung aller Data Warehouses
- Auswertungsfunktion  
Nur wenn direkt auf Basisdatenbank zugegriffen wird!

# Basisdatenbank

## ■ Nabe-Speiche-Architektur



# Data Warehouse

---

## ■ Eigenschaften

- Datenbank für Analysezwecke
- Enthält alle für Analysezweck benötigten Daten
- Strukturierung orientiert sich an Analysebedürfnissen

## ■ Funktion

- Unterstützung des Ladeprozesses
- Unterstützung des Analyseprozesses

# Data Warehouse und Data Mart

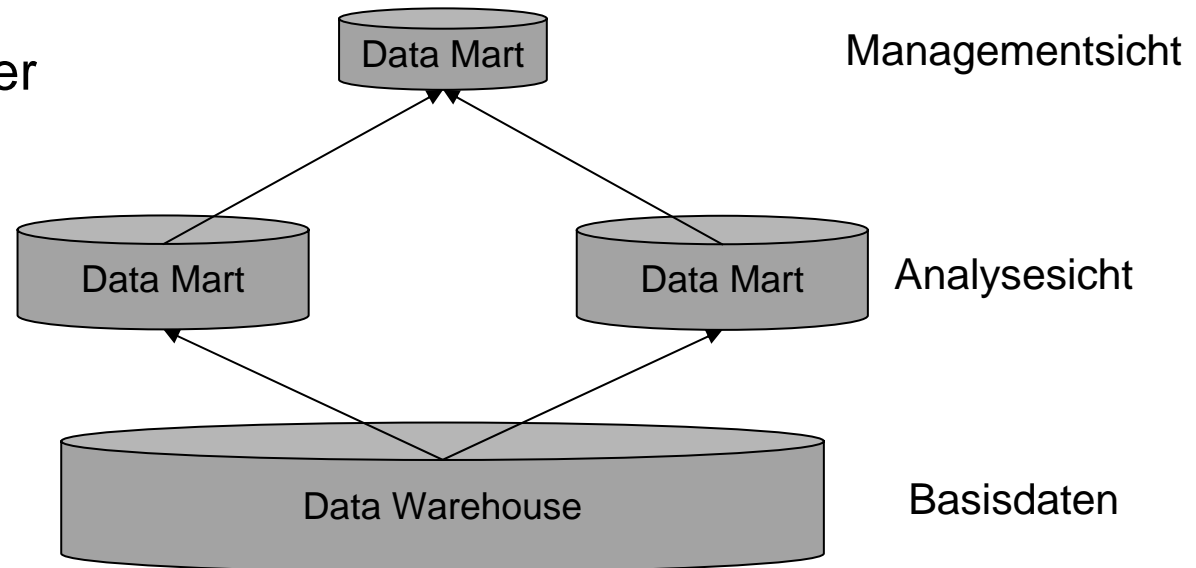
---

- Erfahrung:
  - Data-Warehouse-Projekte scheitern oft an ihrer Größe
  - Anwender brauchen oftmals keine Sicht auf das komplette Data Warehouse
- Daher oftmals auch sogenannte Data Marts:
  - „kleine Data Warehouses“ für überschaubaren Bereich
  - Entwicklung entweder
    - als Teilbestand eines bereits existierenden großen Data Warehouse oder
    - als erster Schritt hin zu einem integrierten unternehmensweiten Data Warehouse.

# Aggregation und Gruppierung

Durch Aggregation und Gruppierung werden die einzelnen Sichten anwendergerecht verdichtet.

- Aufbereitung der Daten für verschiedene Nutzersichten
- Ableitung von Sichten verschiedener Granularität aus den Basisdaten





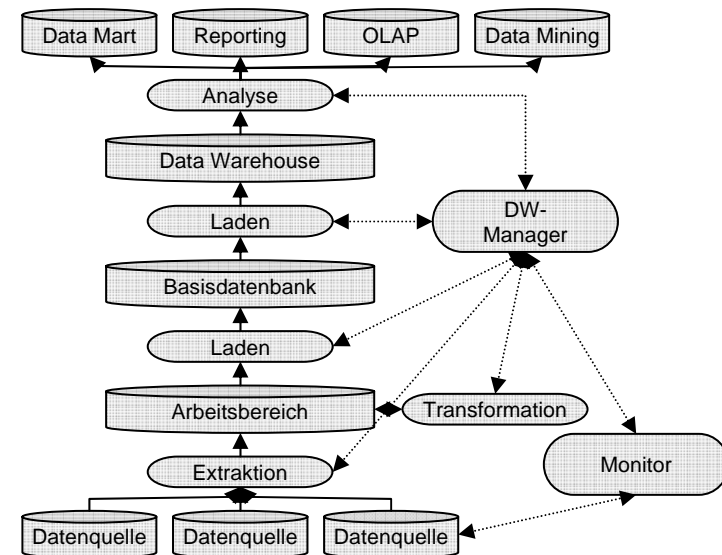
# Data-Warehouse-Manager

---

- Zentrale Steuerung aller Data-Warehouse-Komponenten
  - Monitore
  - Extraktoren
  - Transformatoren
  - Ladekomponenten
  - Analysekomponenten
  
- Initiierung des Datenbeschaffungsprozesses
  - regelmäßige Zeitintervalle
  - von Datenänderungen abhängig
  - auf explizites Verlangen

# Metadaten

- Metadaten sind Daten über Daten
- Wozu werden Metadaten benötigt?
- Welche Metadaten werden in einem Data-Warehouse-System benötigt?



# Metadaten

---

## ■ Partnerarbeit:

- **Variante A**

Sie sind ab sofort für ein laufendes Data-Warehouse-System verantwortlich: Welche Informationen benötigen Sie? Woher bekommen Sie diese Informationen?

- **Variante B**

Sie waren für ein laufendes Data-Warehouse-System verantwortlich: Welche Informationen geben Sie bei der Übergabe Ihrem Nachfolger?

- **Variante C**

Sie sind der Vorgesetzte eines für ein Data-Warehouse-System verantwortlichen Mitarbeiters und haben schlechte Erfahrungen mit der Dokumentation gemacht: Welche Vorgaben geben Sie einem neuen Verantwortlichen bezüglich der Dokumentation des Data-Warehouse-Systems?

# Metadaten sind wichtig!

---

## ■ Historie

- Warum wurde eine Entscheidung getroffen?
  - beim Aufbau,
  - bei der Erweiterung
  - beim Betrieb

## ■ Unternehmen

- Ansprechpartner
- Datenqualität der Datenquellen
- ...

# Metadaten sind wichtig!

- Konzeptuelle Sicht
  - Anforderungen
    - Welche Anfragen möchten Analyseanwender stellen?
  - Terminologie
    - Einheitliche Begriffsdefinitionen: Welche Terminologie verwenden Analyseanwender?
  - ETL-Prozess
    - Welche Quelldaten werden benötigt?
    - Welche Transformationen werden benötigt?
  - Aggregation
    - Wann werden welche Daten wie aggregiert?
    - Welche Aggregationen sind erlaubt?

# Metadaten sind wichtig!

---

- Logische Sicht
  - Schemata
    - Datenquellen, Basisdatenbank, Data Warehouse, Analyseanwendungen, ...
  - Regeln
    - Transformationsregeln
    - Aggregierungsregeln
  
- Physische Sicht
  - Tabellen
  - Indexstrukturen
  - Sichten
  - Materialisierung
  - Partitionierung
  - ...

# Metadaten sind wichtig!

---

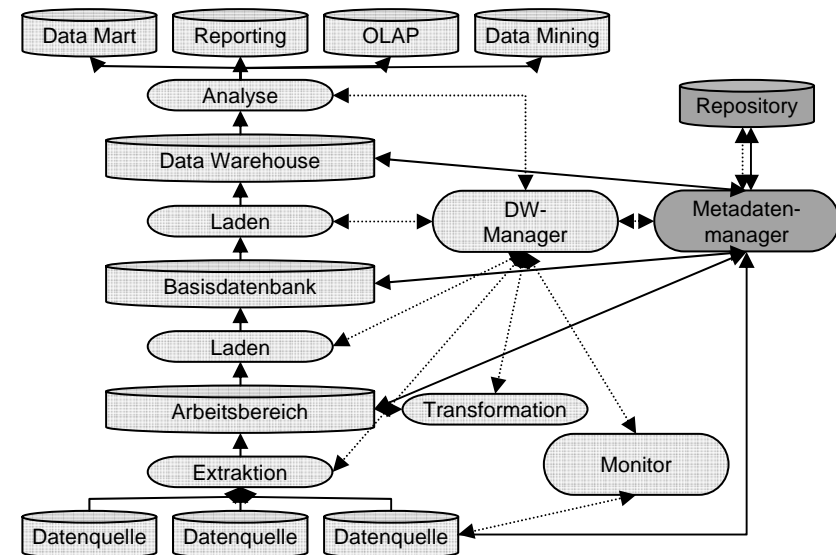
## ■ Organisation

- Wann werden Daten ins Data Warehouse geladen?
- Welche Datenmengen sind zu verwalten bzw. zu laden?
- Benutzergruppen, Zugriffsrechte
- Backup-Strategie
- Datensicherheit
- Datenschutz
- ...

# Metadatenmanager

## ■ Der Metadatenmanager

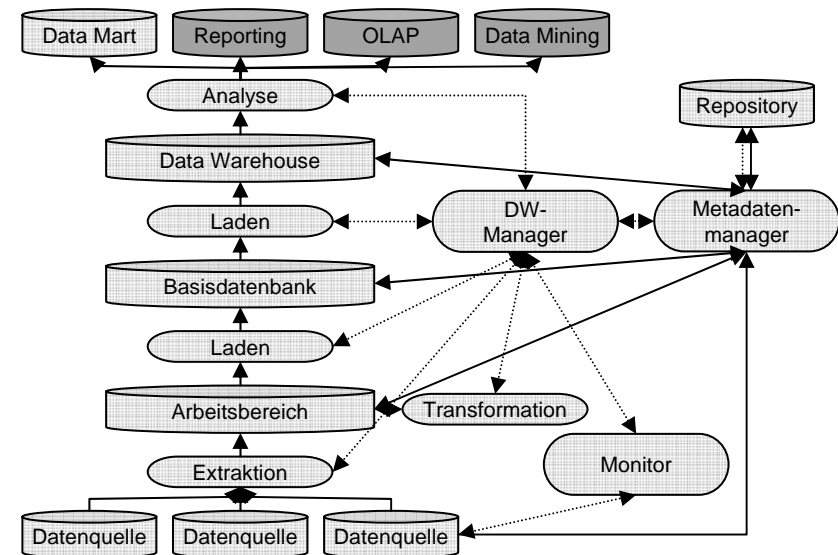
- sammelt alle Metadaten,
- speichert Metadaten zentral im Repository,
- arbeitet mit dem Data-Warehouse-Manager zusammen.





# Analyseanwendungen

- Reporting
  - einfache, automatisch erzeugbare Berichte
- OLAP
  - aufwändige, automatisch erzeugbare Berichte mit Navigationsmöglichkeit
- später
- Data Mining
  - Suche von Mustern in Daten
- später



# DW-Entwicklungszyklus

## ■ Iterativer Prozess

- Anfangs ist den Beteiligten das Potential eines DWS unklar.
  - Neue Anforderungen kommen erst bei laufendem Betrieb oder nach erstem Prototyp.
  - Qualität steigt durch **Anwender-Feedback!**
- Aufbau eines DWS in einem einzigen Schritt besitzt hohe **Komplexität** und erfordert hohen **zeitlichen** und **personellen Aufwand**
- Management oft erst nach ersten Ergebnissen zu überzeugen
  - besser erst vielversprechenstes Anwendungsgebiet aufbauen (positives Feedback)
  - sukzessive weitere Datenquellen integrieren und weitere Data Marts aufbauen
  - „**Think big, start small, grow step by step!**“

# DW-Entwicklungszyklus

---

- Laufender Betrieb: Alles erledigt?
  - Nein!
  - Monitoring der DW-Benutzung
    - Welche Daten werden regelmäßig genutzt?
    - Wie schnell wächst der Datenbestand?
    - Wer benutzt das DW?
    - Sind die Antwortzeiten noch akzeptabel?
  - ... und natürlich sämtliche Veränderungen im Unternehmen im Auge behalten und gegebenenfalls reagieren ...

# Literatur

---

- A. Bauer, H. Günzel: ***Data Warehouse Systeme***. Architektur, Entwicklung, Anwendung. dpunkt, 2001.
- W. Lehner: ***Datenbanktechnologie für Data-Warehouse-Systeme***. dpunkt, 2003.