

---

## 14 Data Warehouse

Olaf Herden

Ein **Data Warehouse** ist eine *physische* Datenbank, die eine *integrierte* Sicht auf beliebige Daten ermöglicht.

Data-Warehouse-Systeme (DWS) haben sich in den letzten Jahren als Standardansatz für entscheidungsunterstützende Systeme etabliert /14.4/, /14.13/, /14.16/, /14.18/, /14.20/. Zunächst wird eine Referenzarchitektur (→ 14.1) beschrieben, bevor das multidimensionale Datenmodell (→ 14.2) folgt. Nach der Behandlung verschiedener Speicherungsformen multidimensionaler Schemata (→ 14.3) werden Erweiterungen relationaler Datenbanken (→ 14.4) vorgestellt.

### 14.1 Architektur

Ein **Data-Warehouse-System (DWS)** enthält neben der Datenbank (dem eigentlichen *Data Warehouse, DWH*) „die Menge aller Komponenten (und deren Beziehungen untereinander), die für das Beschaffen, Aufbereiten, Speichern und Analysieren von Daten“ /14.4/ notwendig sind.

Bild 14.1 zeigt die Architektur eines DWS, deren einzelne Komponenten im Folgenden vorgestellt werden.

#### 14.1.1 Datenquellen

**Datenquellen** enthalten alle für das DWH relevanten Daten und sind somit Ausgangspunkt einer datenflussorientierten Betrachtung der Architektur. Die typischerweise stark heterogenen Datenquellen können internen Ursprungs (operative Systeme der eigenen Organisation) oder aber extern sein, wobei insbesondere das Internet als Quelle in Frage kommt. Der Auswahl geeigneter Quellen zum Aufbau eines DWS kommt erhebliche Bedeutung zu. Bei dieser Auswahl spielen insbesondere die Faktoren Qualität der Daten, technische wie organisatorische Verfügbarkeit und der Preis für den Erwerb der Quelldaten eine Rolle.

#### 14.1.2 Back-End-Bereich

Der **Back-End-Bereich** /14.14/ wird von Werkzeugen gebildet, die zwischen den Datenquellen und dem DWH angesiedelt sind. Neben Monitoren

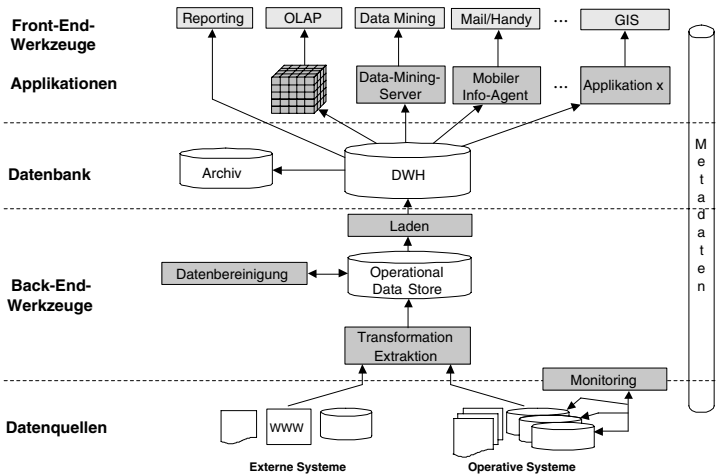


Bild 14.1 Referenzarchitektur Data-Warehouse-System

sind dies vor allem Werkzeuge zur **Extraktion**, **Transformation** und zum **Laden** von Daten. Daraus ergibt sich die Bezeichnung **ETL-Bereich** für den Back-End-Bereich bzw. **ETL-Prozess** für den durchgeführten Prozess.

Eine weitere wichtige Aufgabe des Back-End-Bereichs ist die **Sicherstellung der Datenqualität**, die durch ein durchgängiges und umfassendes Datenqualitätsmanagementsystem gewährleistet werden sollte /14.11/, welches eine hohe Güte der zu analysierenden Daten garantiert.

Als physischer Zwischenspeicher dient im Back-End-Bereich eine als **Operational Data Store (ODS)** bezeichnete Datenbank.

### 14.1.2.1 Monitore

**Monitore** sind für die Entdeckung von Datenänderungen (neue, geänderte oder gelöschte Datensätze) in Quellsystemen zuständig. Um den ODS – und nachfolgend auch das DWH – aktuell zu halten, müssen **Veränderungen** in den Quellsystemen **inkrementell in das DWS propagiert** werden. Die konkrete Funktionsweise eines Monitors hängt unmittelbar von den Charakteristika des angeschlossenen Quellsystems sowie von den Anforderungen der Analysekomponenten ab. Folgende **Monitoring-Strategien** können unterschieden werden /14.22/:

- **Trigger-basiert.** Handelt es sich bei der Datenquelle um ein Datenbanksystem, welches aktive Mechanismen in Form von Triggern (→ 5.4.1.3)

unterstützt, kann das Monitoring erfolgen, indem jede Datenmanipulation einen Trigger auslöst (z. B. After Update), der das geänderte Tupel in eine Datei oder eine andere Datenstruktur schreibt.

- **Replikationsbasiert.** Moderne DBMS bieten Replikationsdienste an. Diese Dienste können so spezifiziert werden, dass sie geänderte Tupel in spezielle Tabellen schreiben.
- **Zeitstempelbasiert.** Jedem Datensatz ist ein Zeitstempel zugeordnet, der im Falle einer Änderung des Datensatzes auf den Zeitpunkt der Änderung gesetzt wird. Anhand der Zeitstempel kann später entschieden werden, welche Datensätze sich seit dem Zeitpunkt der letzten Extraktion geändert haben.
- **Log-basiert.** In diesem Fall nutzt man die Fähigkeit von DBMS aus, vorgenommene Transaktionen in einer Log-Datei zu protokollieren. Durch Analyse einer solchen Log-Datei kann ermittelt werden, welche Daten sich geändert haben.
- **Snapshot-basiert.** Bei dieser Variante wird der Datenbestand einer Quelle in periodischen Zeitabständen in eine Datei, den sog. Snapshot, geschrieben. Durch einen Vergleich aufeinander folgender Snapshots (Delta-Berechnung) können Änderungen identifiziert werden.
- *Hinweis:* Von den vorgestellten Monitoring-Strategien erfordert die Snapshot-basierte Variante den größten (Implementierungs-)Aufwand, da sie keine vom Quellsystem bereitgestellten Dienste nutzt. Jedoch ist gerade bei Altsystemen, die solche Dienste meist nicht anbieten, Snapshot-Monitoring häufig die einzige anwendbare Technik zur Entdeckung von Änderungen im Datenbestand.

### 14.1.2.2 Extraktionskomponenten

Die **Extraktionskomponente** einer an ein DWS angebundenen Datenquelle ist für die Übertragung von Daten in den Transformationsbereich verantwortlich. Je nach verwendeter Monitoring-Strategie (→ 14.1.2.1) gestaltet sich die Extraktion sehr unterschiedlich: Bei der Trigger-basierten Variante sind die geänderten Tupel aus den entsprechenden Dateien auszulesen, bei Verwendung der Replikationsdienste können sie per SQL-Anfrage aus den Replikationstabellen selektiert werden. Die zeitstempelbasierte Variante erfordert lediglich die Selektion von Tupeln anhand ihres Zeitstempels. Bei der Log- bzw. Snapshot-Variante hängt das Vorgehen von der gewählten Umsetzung der Log-Analyse bzw. des Snapshot-Vergleichs ab. Werden die als geändert identifizierten Tupel beispielsweise in eine Datei geschrieben, so ist diese Datei zu importieren.

### Selektion der Datenquellen

Eine grundlegende Entscheidung besteht in der Festlegung, welche Datenquellen bzw. Ausschnitte daraus in ein DWS zu integrieren sind. Diese

Selektion hängt stark von der inhaltlichen Relevanz der Datenquellen für die geplanten Auswertungen sowie von der Qualität der Quelldaten ab. Bei der Extraktion werden die selektierten bzw. durch das Monitoring als geändert identifizierten Daten aus den quellenspezifischen Datenstrukturen ausgelesen und in die Datenstrukturen des Transformationsbereichs überführt. Aufgrund der u. U. großen Datenvolumina kann eine Komprimierung der zu transferierenden Daten sinnvoll sein.

Die Zeitpunkte, zu denen eine Extraktion durchgeführt wird, sollten je nach Bedarf unterschiedlich gewählt werden. Es gibt folgende prinzipielle Vorgehensweisen /14.15/:

- **Periodisch.** Die Extraktion wird in periodischen Zeitabständen durchgeführt, wobei die Periodendauer von der Dynamik der Daten bzw. von den gestellten Anforderungen an deren Aktualität abhängt. So sind z. B. Börsenkurse oder Wetterdaten (mehrmals) täglich zu aktualisieren, während Angaben über technische Spezifikationen von Produkten typischerweise beständiger sind und daher mit einer längeren Periodendauer auskommen.
- **Anfragegesteuert.** In diesem Fall wird die Extraktion durch eine explizite Anfrage angestoßen. Wenn beispielsweise eine Produktgruppe um einen neuen Artikel erweitert wird, so kann die Extraktionskomponente angewiesen werden, die in den operativen Quellen zu diesem neuen Artikel gespeicherten Informationen in das DWS zu übertragen.
- **Ereignisgesteuert.** Häufig ist es sinnvoll, einen Extraktionsvorgang durch ein Zeit-, Datenbank- oder externes Ereignis auszulösen. Ein typisches Datenbankereignis wäre beispielsweise das Erreichen einer a priori festgelegten Anzahl von Änderungen. Ein externes Ereignis würde z. B. vorliegen, wenn ein Börsenindex eine bestimmte Marke über- oder unterschreitet.
- **Sofort.** Bei besonders hohen Anforderungen an die Aktualität von Daten, z. B. bei Börsenkursen, kann es erforderlich sein, Änderungen in den operativen Quellen unmittelbar (real time) in den ODS eines DWS zu propagieren. Die Daten im ODS sind damit praktisch immer genauso aktuell wie die Daten in den operativen Systemen.
- **Hinweis:** Strenggenommen sind auch periodische und anfragegesteuerte Extraktionen ereignisgesteuert, da sie einem speziellen Zeitereignis bzw. einem durch den Anwender ausgelösten Ereignis entsprechen.

### 14.1.2.3 Transformationskomponente

Ein DWS wird i. Allg. von mehreren heterogenen Datenquellen versorgt, die Daten mit disjunkten oder sich überschneidenden Inhalten in unterschiedlichen Repräsentationsformen liefern. Bevor die aus den Quellen

extrahierten Daten in ein DWH geladen werden können, müssen sie in einen Zustand einheitlicher Repräsentation gebracht werden. Transformationen, die dem Zweck der Vereinheitlichung dienen, sind unter dem Begriff **Data Migration** zusammengefasst.

Die transformierten Daten werden schließlich im ODS abgelegt. Dabei ist eventuell eine **Historisierung inkrementeller Aktualisierungen** vorzunehmen. Ein im DWS gespeicherter Datensatz, zu dem es eine Änderung in einer Datenquelle gegeben hat, darf daher nicht grundsätzlich mit den geänderten Werten überschrieben werden. Stattdessen ist der geänderte Datensatz ggf. mit einem Zeitstempel zu versehen und zusätzlich zu dem bereits „veralteten“ Datensatz im DWS abzulegen.

In diesem Schritt wird ausschließlich die Transformation von Daten behandelt, nicht hingegen die Transformation von Datenstrukturen bzw. Schemata. **Schemaintegration** ist natürlich eine entscheidende Voraussetzung für die **Datenintegration** (→ 13.7).

#### 14.1.2.4 Ladekomponente

Die **Ladekomponente** ist für die Übertragung konsolidierter Daten aus dem ODS in die analyseorientierten Strukturen des DWH zuständig. Zur technischen Durchführung dient dabei oft das Ladewerkzeug des zugrunde liegenden DBMS. Es kann zwischen *Online*- und *Offline*-Ladevorgängen unterschieden werden /14.3/: Bei **Online-Ladevorgängen** steht das DWH auch während des Ladens für Anfragen zur Verfügung, bei **Offline-Ladevorgängen** ist dies nicht der Fall. Üblicherweise findet nur das initiale Laden offline statt. **Inkrementelle Updates** sollten hingegen nur online durchgeführt werden, um den laufenden Betrieb des DWS nicht unterbrechen zu müssen. Das Zeitfenster für den Ladevorgang ist allerdings so zu wählen, dass die Beeinträchtigung der Benutzung minimiert wird, z. B. nachts oder an Wochenenden.

#### 14.1.3 Datenbank

In dieser Architekturschicht sind mit dem DWH und dem **Archiv** die zwei wesentlichen datenspeichernden Komponenten angesiedelt. Als Variante bzw. denkbare Gestaltungsmöglichkeit des DWH sind sog. *Data Marts* populär, die ebenfalls kurz vorgestellt werden (→ 14.1.3.2).

##### 14.1.3.1 Data Warehouse

**Zentrale Komponente** eines DWS ist das **Data Warehouse**, eine durch folgende Charakteristika gekennzeichnete Datenbank:

- Die Datenbank ist *physisch realisiert*,
- sowohl *Schema* als auch *Daten* sind *integriert*,
- das Schema ist *analyseorientiert*,
- i. Allg. werden *an den Daten keine Modifikationen* vorgenommen,
- in der Regel liegen die *Daten historisiert* vor.

Die wesentlichen Unterschiede zwischen herkömmlichen operativen Datenbanken und DWHs fasst Tabelle 14.1 zusammen /14.4/, /14.5/.

Tabelle 14.1 Vergleich Data Warehouse – operative Datenbanken

Kriterium	Operative Datenbank	Data Warehouse
Zweck und Anwendercharakteristik		
Verwendung	transaktional, Unterstützung von Geschäftsprozessen	analytisch, Informationen für Entscheidungsträger
Anwenderzahl(-typ)	sehr viele (Sachbearbeiter)	wenige (Analysten, Entscheider)
Zugriffscharakteristika		
Zugriffsart von Applikationen	Lesen, Schreiben, Modifizieren, Löschen	Lesen, periodisches Hinzufügen
Transaktionsdauer und -typ	kurze Lese- und Schreibtransaktionen	lange Lesetransaktionen
Anfragecharakteristik (Struktur, Typ)	einfach strukturiert, vorhersagbar	komplex, rechenintensiv, häufig Ad-hoc-Anfragen
Zugriffsart	einzelne Datensätze	Bereichsanfragen
Volumen der (Zwischen-)Ergebnisse	wenige Datensätze	viele Datensätze
Erwartete Antwortzeiten	(Milli-)Sekunden	Sekunden bis Minuten
Update	laufend, konkurrierend	ergänzend, in periodischen Abständen
Daten- und Schemacharakteristika		
Eigenschaften der Daten	nicht abgeleitet, zeitaktuell, autonom, dynamisch	abgeleitet, konsolidiert, historisiert, integriert, stabil
Datenquellen	(meistens) eine	(fast immer) mehrere
Granularität der Daten	Detailldaten	Detailldaten, Aggregate
Aktualität der Daten	Online, Realtime	(meistens) zeitversetzt
Datenschema	normalisiert	zweckgebunden, analyseorientiert
Datenvolumen	Megabyte bis Gigabyte	Gigabyte bis Petabyte

### 14.1.3.2 Data Marts

#### Nachteile von DWH

Die Realisierung eines zentralen (d. h. unternehmensweiten) DWH kann sich in einigen Fällen als organisatorisch oder technisch schwer durchsetzbar erweisen. Möglicherweise ist eine solche Lösung zu teuer, organisatorisch nicht realisierbar oder zu ressourcenintensiv. Technische Probleme kann eine zentrale Lösung hinsichtlich der Skalierbarkeit bei steigender Benutzerzahl und/oder anwachsenden Datenbeständen aufwerfen.

Aus diesem Grunde haben sich **Data Marts** („kleine DWHs“) als Lösung etabliert, was aus Datenbanksicht einer Verteilung des DWH-Datenbestandes entspricht.

Prinzipiell lassen sich die beiden in Bild 14.2 dargestellten Ausprägungen unterscheiden. **Abhängige Data Marts** sind Extrakte aus einem zentralen, integrierten DWH, während **unabhängige Data Marts** als isolierte Sichten auf die Quellsysteme ohne Verwendung einer „großen, gemeinsamen Datenbasis“ realisiert werden.

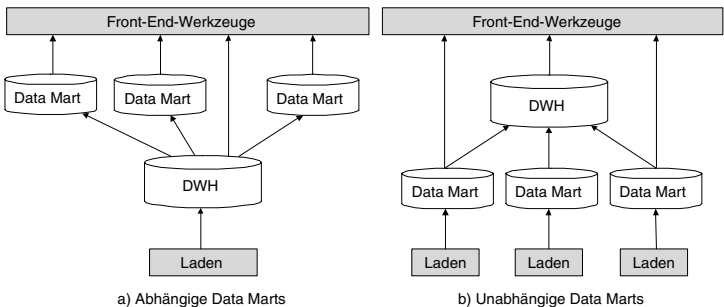


Bild 14.2 Abhängige und unabhängige Data Marts

Bei der Konzeption von abhängigen Data Marts ist die wichtigste Frage, welche Daten der Extrakt enthalten soll. Prinzipiell gibt es drei verschiedene Arten, die auch in Kombination angewendet werden:

- **Struktureller Extrakt.** Nur ein Teil des DWH-Schemas wird in den Data Mart repliziert und somit nur eine bestimmte Gruppe von Analysen ermöglicht.
- **Inhaltlicher Extrakt.** Es wird zwar das gesamte Schema, aber nur ein Teil der Daten im Data Mart vorgehalten, z. B. nur die Daten einer bestimmten Periode oder einer bestimmten Organisationseinheit.

- **Aggregierter Extrakt.** Im Data Mart werden die Daten mit einem **geringeren Detaillierungsgrad**, d. h. auf einer höheren Abstraktionsebene, gespeichert, so dass sich das **Datenvolumen verringert**.

#### 14.1.3.3 Archiv-Datenbank

In einem DWS muss eine definierte „**Entsorgung**“ der Daten aus dem DWH **vorgesehen werden**. Gründe hierfür können **mangelndes Interesse an veralteten Daten** oder aber **Platz- und Performance-Probleme** sein.

Bei dieser Entsorgung stellt sich die Frage nach endgültigem Löschen oder der Übernahme in eine **Archiv-Datenbank**. In dieser werden die Daten an einem separaten Ort langfristig gehalten, um sie im Bedarfsfall wieder im DWH zur Verfügung stellen zu können.

**Häufig kombiniert man den Archivierungsvorgang auch mit einem erneuten Schreibvorgang in das DWH:** Die veralteten Detaildaten werden in die Archiv-Datenbank übertragen, eine verdichtete Version (aggregierter Extrakt) dieser Daten im DWH abgelegt.

#### 14.1.4 Front-End-Werkzeuge

**Front-End-Werkzeuge** sind **auf dem DWH basierende Applikationen**. Je nach Funktionsumfang lassen sich verschiedene Klassen von Front-End-Werkzeugen identifizieren.

##### 14.1.4.1 Berichts- und Anfragewerkzeuge

**Berichtswerkzeuge** erzeugen **mittels vordefinierter, oft parametrisierter Anfragen Auswertungen der Daten**, reichern diese eventuell um einfache arithmetische Operationen an und repräsentieren sie in Form von Berichten. Diese können tabellarisch oder als Diagramme dargestellt sein. Berichtswerkzeuge lassen sich **entweder für die Generierung periodisch wiederkehrender Berichte im Batchbetrieb einsetzen oder stehen als Endbenutzeranwendungen für unregelmäßige Auswertungen** (sog. Ad-hoc-Anfragen) zur Verfügung.

Ebenfalls in diese Kategorie eingeordnet werden können **Anfragewerkzeuge**, bei denen **zwischen DB und Benutzer eine Zwischenschicht verwendet wird**, die es **durch „Point and Click“-Bedienung ermöglicht, Anfragen zu formulieren** und so dem Endbenutzer das Darstellen komplexer SQL-Anfragen abnimmt.



### 14.1.4.2 OLAP-Werkzeuge

**OLAP-Werkzeuge** (Online Analytical Processing) bieten die Möglichkeit der interaktiven Datenanalyse. Die Sicht auf die Daten erfolgt meistens multidimensional (→ 14.2). Dem Anwender wird die Möglichkeit geboten, in Abhängigkeit von der aktuellen Anfrage eine Zusammenstellung der Daten hinsichtlich Verdichtungsgrad und Präsentation vorzunehmen. Damit bieten sie gegenüber Berichtswerkzeugen die Vorteile einer interaktiven, auf die individuellen Bedürfnisse zugeschnittenen Datenanalyse.

### 14.1.4.3 Data-Mining-Werkzeuge

Im Gegensatz zum OLAP ist das **Data Mining** (→ 15) ein induktiver Prozess. Es werden keine Vermutungen menschlicher Benutzer durch interaktive Analysen erhärtet. Stattdessen wird versucht, bisher unbekannte Zusammenhänge und Trends im Datenbestand anhand von Algorithmen des maschinellen Lernens und statistischer Verfahren zu entdecken.

Obwohl Data Mining auch ohne DWH möglich ist, bietet das Aufsetzen auf einem DWH erhebliche Vorteile. So können viele Zusammenhänge erst durch die Integration verschiedener Datenbestände oder das Vorliegen eines versionierten Datenbestandes entdeckt werden.

Nachteile des Data Mining in einer DWH-Umgebung sind z. B. folgende: Die für OLAP optimierte Struktur des Schemas ist nicht zwangsläufig für Data Mining geeignet und durch die Bereinigung und Integration von Daten können manche Analysen nicht mehr sinnvoll durchgeführt werden.

Tabelle 14.2 fasst die drei Front-End-Werkzeugtypen zusammen /14.12/.

Tabelle 14.2 Vergleich Front-End-Werkzeugklassen

Kriterium	Berichte	OLAP	Data Mining
Systemausrichtung	berichtsorientiert	berichtsorientiert	algorithmisch
Nutzungsinitiative	systeminduziert	benutzerinduziert	benutzerinduziert
Nutzungsfrequenz	(a-)periodisch	aperiodisch	aperiodisch
DV-Kompetenz der Benutzer	wenig	mittel	(sehr) groß

### 14.1.4.4 Sonstige Front-End-Werkzeuge

Neben den drei „klassischen“ auf einem DWH aufsetzenden Front-End-Komponenten sind beliebige Applikationen denkbar, die sich mit Daten aus dem DWH versorgen (→ Bild 14.1).

- ❑ *Beispiel 1:* Mobile Informationsagenten können das DWH als Datenbasis nutzen und durch zusätzliches Wissen kritische Fälle erkennen, in denen sie Benutzer benachrichtigen.
- ❑ *Beispiel 2:* Unter Berücksichtigung der Tatsache, dass (fast) alle entscheidungsrelevanten Daten einen Raumbezug haben, ist auch die Kombination mit einem GIS (geografisches Informationssystem) eine interessante Möglichkeit, raumbezogene Auswertungen attraktiv aufbereitet zu präsentieren (→ 17).

### 14.1.5 Sonstige Werkzeuge

Ergänzt werden die bisher vorgestellten Komponenten in einem DWS typischerweise um einen *DWS-Manager* und ein *Metadaten-Repository*.

#### 14.1.5.1 DWS-Manager

Der **DWS-Manager** ist ein Administrationswerkzeug, das für die Steuerung und Überwachung der einzelnen im DWS stattfindenden Prozesse zuständig ist. Dazu koordiniert er insbesondere das Zusammenspiel der beteiligten Komponenten, inklusive einer angemessenen *Ausnahme- und Fehlerbehandlung*. Weitere Aufgaben des DWS-Managers sind die *Unterstützung des DWS-Administrators* bei der Prozessplanung und die *Kommunikation mit dem DWS-Administrator* während der Prozessausführung.

#### 14.1.5.2 Metadaten-Repository

Grundvoraussetzung für den effektiven Umgang mit Daten ist das Vorhandensein von Wissen über deren Syntax und Semantik. Sind diese **Metadaten** formal repräsentiert und zentral verfügbar, können Datenverarbeitungsprozesse einerseits *automatisiert*, andererseits *sehr flexibel* gestaltet werden. In der DWS-Architektur ist das **Metadaten-Repository** für die zentrale Verwaltung und Bereitstellung von Metadaten zuständig.

Üblicherweise werden *administrative*, *domänenspezifische* und *operative* Metadaten unterschieden /14.6/.

Zu den **administrativen Metadaten** zählen u. a.:

- *Schemainformationen*, d. h. Metadaten im klassischen Sinne als „Daten über Daten“,
- Informationen über *Quell- und Zielsysteme*, z. B. technische Charakteristika über den Zugriff wie Rechner- oder DB-Namen,
- *Datenabhängigkeiten* in Form von Transformationsregeln zwischen den Quellsystemen und dem DWH,
- *Datenabhängigkeiten* in Form von Anfragen o. Ä. zwischen dem DWH und den Front-End-Werkzeugen.

**Operative Metadaten** sind u. a.:

- *Systemstatistiken* für die Ressourcenplanung und Optimierung, d. h. Anfragemuster oder nutzer- bzw. gruppenspezifisches Nutzungsverhalten,
- Informationen über *Scheduling, Logging* und *Jobausführung* des DWH,
- Regeln und Funktionen für *Nachladen* und *Archivierung*.

Den **domänenspezifischen Metadaten** werden u. a. zugerechnet:

- *Informationsmodelle* und *konzeptuelle Schemata* (→ 2), die der implementierungsunabhängigen Dokumentation dienen,
- *Organisations- bzw. branchenspezifische Begriffswerke* (*Vokabulare, Terminologien* und *Taxonomien*),
- *Abbildungen* zwischen diesen drei Begriffswerken und den korrespondierenden Elementen im DWH,
- Informationen über Organisationsstrukturen und Geschäftsprozesse,
- *Konzeptionelle Beschreibungen* von Berichten, Anfragen, Kennzahlen,
- Angaben über die *Datenqualität*.

Das Metadaten-Repository kommuniziert mit den anderen DWS-Komponenten, die entweder Metadaten anfordern (z. B. Schemabeschreibungen) oder aber ihrerseits erzeugte Metadaten im Repository ablegen (z. B. Zugriffsstatistiken).

Typischerweise existieren in der Praxis neben einem zentralen Repository bei den einzelnen Werkzeugen lokale Datenhaltungskomponenten, in denen Metainformationen abgelegt werden.

## 14.2 Multidimensionale Datenmodelle

Für die Einführung statischer und dynamischer Konzepte multidimensionaler Datenmodelle soll das folgende Beispiel-Szenario dienen: In einem Unternehmen werden die Verkaufszahlen von Produkten pro Tag und Filiale analysiert; relevante Zeiteinheiten neben dem Tag sind Woche, Monat, Quartal und Jahr. Die Produkte sollen einerseits zu Produktgruppen, andererseits zu Marken und Herstellern zusammengefasst werden; Filialen lassen sich immer einer Stadt zuordnen, diese einer Region und diese wiederum einem Land.

### 14.2.1 Statische Aspekte

#### Grundbegriffe

Hauptcharakteristikum multidimensionaler Datenmodelle ( /14.4/, /14.10/) ist die Klassifikation in *quantifizierende* und *qualifizierende* Daten. **Fakten**