

# Einführung in SAP Business Information Warehouse

Jorge Marx Gómez  
Claus Rautenstrauch  
Peter Cissek  
Björn Grahlher

# Einführung in SAP Business Information Warehouse

Mit 195 Abbildungen  
und 6 Tabellen

 Springer

Professor Dr.-Ing. Jorge Marx Gómez  
Universität Oldenburg  
Department für Informatik  
Abteilung Wirtschaftsinformatik  
Ammerländer Heerstraße 114-118  
26129 Oldenburg  
E-mail: jorge.marx.gomez@uni-oldenburg.de

Peter Cissek  
Champagne 7  
40822 Mettmann  
E-mail: peter.cissek@gmx.de

Björn Grahlher  
Am Thing 24  
21244 Buchholz  
E-mail: bjoern.grahlher@gmx.de

Professor Dr. Claus Rautenstrauch  
Otto-von-Guericke-Universität  
Fakultät Informatik  
Institut Technische/Betriebliche,  
Informationssysteme  
Universitätsplatz 2  
39106 Magdeburg  
E-mail: rauten@iti.cs.uni-magdeburg.de

Die Deutsche Bibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.ddb.de> abrufbar.

ISBN-10 3-540-31124-6 Springer Berlin Heidelberg New York  
ISBN-13 978-3-540-31124-9 Springer Berlin Heidelberg New York

Dieses Werk ist urheberrechtlich geschützt. Die dadurch begründeten Rechte, insbesondere die der Übersetzung, des Nachdrucks, des Vortrags, der Entnahme von Abbildungen und Tabellen, der Funksendung, der Mikroverfilmung oder der Vervielfältigung auf anderen Wegen und der Speicherung in Datenverarbeitungsanlagen, bleiben, auch bei nur auszugsweiser Verwertung, vorbehalten. Eine Vervielfältigung dieses Werkes oder von Teilen dieses Werkes ist auch im Einzelfall nur in den Grenzen der gesetzlichen Bestimmungen des Urheberrechtsgesetzes der Bundesrepublik Deutschland vom 9. September 1965 in der jeweils geltenden Fassung zulässig. Sie ist grundsätzlich vergütungspflichtig. Zuwiderhandlungen unterliegen den Strafbestimmungen des Urheberrechtsgesetzes.

Springer ist ein Unternehmen von Springer Science+Business Media  
[springer.de](http://springer.de)

© Springer-Verlag Berlin Heidelberg 2006  
Printed in Germany

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Einbandgestaltung: design & production GmbH  
Herstellung: Helmut Petri  
Druck: Strauss Offsetdruck

SPIN 11609759      Gedruckt auf säurefreiem Papier – 42/3153 – 5 4 3 2 1 0

# Vorwort

Data-Warehouse-Systeme sind heute fester Bestandteil der taktischen und strategischen Informationssysteme in Unternehmen. Mit dem SAP® Business Information Warehouse (BW®) hat es die SAP AG in erstaunlich kurzer Zeit geschafft, sowohl die Markt- als auch Technologieführerschaft im Marktsegment der Data-Warehouse-Systeme zu übernehmen. Schon allein dies ist Grund genug, dass Studierende, Lehrende und Praktiker zumindest Grundkenntnisse im Umgang mit derartigen Systemen erwerben. Studierende mit Kenntnissen über das SAP BW haben einen erleichterten Berufseinstieg und Praktiker können bei der Entscheidungsunterstützung in Unternehmen fundiert mitreden, wenn sie BW-Kenntnisse erwerben und umsetzen.

Dieses Buch hat den Anspruch, den Einstieg in die Welt des SAP BW auf einfache und fundierte Weise zu erleichtern. Aufbauend auf den technologischen und konzeptionellen Grundlagen von Data Warehouses werden praxisnahe und erprobte Fallstudien präsentiert, anhand derer die theoretischen Konzepte nachvollzogen und vertieft werden. So hat der geneigte Leser die Möglichkeit, Erkenntnisse sowohl aus der theoretischen Betrachtung sowie aus dem Nachspielen von Fallstudien zu gewinnen. So ist der Lernerfolg hoffentlich vorprogrammiert.

Für die Lehre an Universitäten, Fachhochschulen und Berufsakademien ist die flächendeckende Versorgung mit SAP BW durch die SAP Hochschulkompetenzzentren (SAP HCC) an der Otto-von-Guericke-Universität Magdeburg und der TU München sichergestellt. So sind die Fallstudien in enger Zusammenarbeit mit den SAP HCC Magdeburg entstanden, deren Mitarbeiter trotz stets stressreichem Tagesgeschäft dieses Buchprojekt in dankenswerter Weise mit Rat und Tat unterstützten.

Damit sei dann auch der Reigen der Danksagungen eröffnet. Nicht genug kann man stets den Familien und Partnern danken, die unter der Arbeitswut der Autoren unverschuldet zu leiden haben. Leidenschaft verlangen die Autoren aber auch oft von den Partnern auf der Seite des Verlags. Hier gilt unser Dank Herrn Dr. Werner A. Müller, der dieses Projekt mit stets konstruktiver Kritik, fundierten Ratschlägen und Professionalität begleitet hat. Für die abschließenden Korrektur- und Formatierungsarbeiten danken wir Herrn Dirk Schlehf.

Wir wünschen allen Lesern viel Spaß und Erfolg bei der Lektüre.

Die Autoren

Magdeburg, im Dezember 2005

# Inhalt

<b>Vorwort</b> .....	<b>V</b>
<b>Verzeichnis der Abkürzungen und Akronyme</b> .....	<b>XI</b>
<b>Verzeichnis der eingetragenen Marken</b> .....	<b>XIII</b>
<b>1 Einführung ins SAP Business Information Warehouse</b> .....	<b>1</b>
1.1 Ziele und Aufbau dieses Buches .....	1
1.2 Vorbereitung der Fallstudien .....	2
<b>2 Theoretische Grundlagen der Data Warehouse Technologie</b> .....	<b>3</b>
2.1 Einführung .....	3
2.1.1 Bedeutung der Information für die Entscheidungsunterstützung .....	3
2.1.2 Operative Systeme .....	4
2.1.3 Data Warehouse Systeme .....	6
2.2 Daten und Architektur .....	7
2.2.1 Geschäftsdaten .....	7
2.2.2 Metadaten .....	9
2.2.3 Komponenten .....	9
2.3 Datenmodell .....	11
2.3.1 Fakten, Kennzahlen und Dimensionen .....	12
2.3.2 Modellierung von Zeit .....	15
2.3.3 Multidimensionale Anfragestrukturen .....	15
2.3.4 OLAP Operationen .....	16
2.3.5 ADAPT .....	18
2.3.6 Speicherung multidimensionaler Daten, Star- und Snowflake-Schema .....	21
2.4 Datenquellen und Datenqualität .....	26
2.4.1 Quellen für Geschäftsdaten .....	26
2.4.2 Quellen für Metadaten .....	27
2.4.3 Datenqualität .....	27
2.4.4 Qualitätsmetriken .....	30
2.4.5 Data Cleaning .....	31
2.5 Typisches ETL-Prozessmodell .....	33
2.5.1 Vorbereitungsphase .....	34

---

2.5.2 Extraktionsphase .....	34
2.5.3 Transformationsphase .....	35
2.5.4 Ladephase.....	36
2.6 Bewertung eines Data Warehouse .....	36
2.6.1 Stärken eines Data Warehouse .....	37
2.6.2 Schwächen eines Data Warehouse .....	37
<b>3 Theoretische Grundlagen des SAP Business Information Warehouse.....</b>	<b>41</b>
3.1 Umgang mit SAP Systemen .....	41
3.1.1 SAPlogon .....	41
3.1.2 SAP Fenster.....	42
3.1.3 Elemente der SAP Bildschirmmasken.....	45
3.2 Komponenten des SAP BW .....	47
3.2.1 Datenschicht .....	48
3.2.2 Administrationsschicht.....	48
3.2.3 Analyseschicht .....	49
3.3 Datenmodellierung .....	50
3.3.1 Struktur – InfoArea, InfoProvider und InfoObjectCatalog.....	50
3.3.2 InfoObjects .....	51
3.3.3 InfoCubes und Dimensionen .....	63
3.3.4 ODS-Objekte.....	64
3.3.5 InfoSets .....	66
3.3.6 MultiProvider .....	67
3.4 Datenbeschaffung .....	69
3.4.1 Extraktion .....	71
3.4.2 Transformation .....	77
3.4.3 Laden.....	79
3.5 Berichtswesen .....	83
3.5.1 BEx Query Designer .....	84
3.5.2 BEx Analyzer .....	89
3.5.3 Web-basiertes Berichtswesen.....	90
3.5.4 BEx Web Application Designer .....	102
3.5.5 Formatiertes Reporting .....	108
3.5.6 Mobile Intelligence .....	108
3.5.7 Weitere Funktionen: Offline Reporting und Integration in das SAP Enterprise Portal .....	109
<b>4 Fallstudie I – Ist-Daten-Analyse.....</b>	<b>111</b>
4.1 Teil I – Datenmodellierung.....	113
4.1.1 Anlegen der InfoArea.....	115
4.1.2 Anlegen des InfoObjectCatalogs und der InfoObjects .....	117
4.1.3 Anlegen des InfoCubes für die Ist-Daten .....	130
4.2 Teil II – Datenbeschaffung .....	138
4.2.1 Anlegen der InfoSources .....	140
4.2.2 Einlesen von Stammdaten aus Flat Files .....	143
4.2.3 Manuelle Pflege der Stammdaten.....	154

---

4.2.4	Pflege der Shape Files für die Kartendarstellung .....	159
4.2.5	Einlesen der Bewegungsdaten .....	162
4.3	Teil III – Reporting .....	170
4.3.1	Anlegen der Query .....	172
4.3.2	Reporting mit dem BEx Analyzer unter MS Excel .....	176
4.3.3	Ad hoc – Reporting im Webbrowser .....	182
4.3.4	Reporting mit dem BEx Web Application Designer .....	188
<b>5</b>	<b>Fallstudie II – Soll-Ist Vergleich.....</b>	<b>199</b>
5.1	Teil I – Datenmodellierung.....	200
5.1.1	Anlegen eines ODS-Objekts.....	201
5.1.2	Anlegen des InfoCubes für die Soll-Daten.....	204
5.1.3	Anlegen des Multiproviders .....	205
5.2	Teil II – Datenbeschaffung .....	212
5.2.1	Anlegen der InfoSource für die Soll-Daten.....	213
5.2.2	Schreiben der Daten in das ODS-Objekt .....	218
5.2.3	Fortschreiben der Soll-Daten in den InfoCube sowie der korrigierten Ist-Daten .....	224
5.3	Teil III – Reporting .....	228
5.3.1	Anlegen der Query .....	229
5.3.2	Reporting im BEx Web Application Designer .....	235
<b>6</b>	<b>Fallstudie III – Bestandsdatenanalyse .....</b>	<b>245</b>
6.1	Teil I – Datenmodellierung.....	246
6.1.1	Anlegen der Kennzahlen .....	246
6.1.2	Anlegen des InfoCubes für die Bestandsdaten .....	249
6.2	Teil II – Datenbeschaffung .....	250
6.2.1	Anlegen der InfoSources für Anfangsbestand und Bestandsveränderungen.....	251
6.2.2	Einlesen des Anfangsbestands und der Bestandsveränderungen.....	254
6.3	Teil III – Reporting .....	257
6.3.1	Anlegen der Queries .....	258
6.3.2	Reporting im BEx Web Application Designer .....	261
	<b>Literatur .....</b>	<b>271</b>

# Verzeichnis der Abkürzungen und Akronyme

ABAP	Advanced Business Application Programming
ADAPT	Application Design for Analytical Processing Technologies
ALE	Application Linking and Embedding
API	Application Programming Interface
APO	Advanced Planner and Optimizer
ARIS	Architektur integrierter Informationssysteme
ASCII	American Standard Code for Information Integration
BAPI	Business Application Programming Interface
BEx	Business Explorer
CR	Carriage Return
CRM	Customer Relationship Management
CSS	Cascading Style Sheets
CSV	Character Separated Values
DB	Datenbank (engl. Database)
DBMS	Database Management System
DOM	Document Object Model
DV	Datenverarbeitung
EIS	Executive Information System
ERP	Enterprise Resource Planning
ETL	Extraktion Transformation Laden
GUI	Graphical User Interface
HOLAP	Hybrid Online Analytical Processing
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
IDoc	Intermediate Document
MOLAP	Multidimensional Online Analytical Processing
ODS	Operational Data Store
OLAP	Online Analytical Processing
OLTP	Online Transaction Processing
OMG	Object Management Group
PDA	Personal Digital Assistant
PSA	Persistent Staging Area
SAP	Systeme, Anwendungen und Produkte in der Datenverarbeitung
SEM	Strategic Enterprise Management
SMS	Short Message Service
SOAP	Simple Object Access Protocol
SQL	Structured Query Language



URL	Uniform Resource Locator
WAP	Wireless Application Protocol
WWW	World Wide Web
XML	Extensible Markup Language

# Verzeichnis der eingetragenen Marken

ADAPT® ist eine eingetragene Marke der Symmetry Corporation.

Crystal Reports® ist eine eingetragene Marke der Business Objects SA.

CSS®, DOM®, HTML®, HTTP® und XML® sind eingetragene Marken des W3C®, World Wide Web Consortium, Massachusetts Institute of Technology.

i-Mode® ist eine eingetragene Marke der NTT DoCoMo Inc.

JavaScript® ist eine eingetragene Marke der Sun Microsystems, Inc., verwendet unter der Lizenz der von Netscape entwickelten und implementierten Technologie.

Microsoft®, Excel®, Internet Explorer®, PowerPoint®, Windows® sowie Grafiken und entsprechende Logos sind eingetragene Marken der Microsoft Corporation.

Netscape® und Netscape® Navigator und weitere im Text verwendete Netscape-Produkte sind eingetragene Marken der Netscape Communications Corporation und America Online, Inc.

Nokia® und Nokia Mobiltelefone sind eingetragene Marken der Nokia Corporation.

Palm® und Palm® Pilot sind eingetragene Marken der Palm, Inc.

SAP®, ABAP®, BAPI®, SAP iView®, SAP NetWeaver®, SAP SEM®, R/3® sowie Grafiken, entsprechende Logos und weitere im Text verwendete SAP-Produkte sind eingetragene Marken der SAP AG.

Six Sigma® ist eine eingetragene Marke der Motorola Inc.

Alle anderen Namen von Produkten und Dienstleistungen sind Marken der jeweiligen Firmen.

# 1 Einführung ins SAP Business Information Warehouse

## 1.1 Ziele und Aufbau dieses Buches

Das Ziel dieses Buches ist es, dem Leser anhand von Fallstudien einen praktischen Einstieg in die Software SAP Business Information Warehouse zu ermöglichen. Dem Leser werden dabei keinerlei Vorkenntnisse im Bereich des Data Warehousing oder Erfahrungen mit Produkten der SAP AG abverlangt, sämtliches Basiswissen wird in den Kapiteln 2 Theoretische Grundlagen der Data Warehouse Technologie und 3 Theoretische Grundlagen des SAP Business Information Warehouse vermittelt.

Das Lehrbuch richtet sich primär an Lehrende und Studierende der Fachrichtungen Informatik, Wirtschaftsinformatik und der Wirtschaftswissenschaften. Gleichsam unterstützt es den Anwender im Unternehmen bei seinen ersten Schritten im SAP BW.

An das vorliegende Werk kann auf drei verschiedene Weisen herangegangen werden. Wenn bereits fundierte Kenntnisse der Data Warehouse Technologie vorhanden sind, kann der theoretische Teil übersprungen und direkt im Kapitel 4 Fallstudie I – Ist-Daten-Analyse mit der Fallstudie begonnen werden. Die zweite Herangehensweise besteht darin, streng nach der Kapitelstruktur vorzugehen und somit vor Bearbeitung der Fallstudie zunächst den kompletten Theorieteil zu lesen. Die dritte und von den Autoren empfohlene Möglichkeit ist, grundsätzlich der Kapitelstruktur zu folgen, jedoch parallel zum Kapitel 3 Theoretische Grundlagen des SAP Business Information Warehouse das Kapitel 4 Fallstudie I – Ist-Daten-Analyse zu bearbeiten. Diese Vorgehensweise wird dadurch unterstützt, dass im Verlauf des dritten Kapitels an geeigneter Stelle auf die entsprechenden Abschnitte des vierten Kapitels verwiesen wird. Bevor mit der Ausführung der Fallstudien begonnen wird, ist sicherzustellen, dass die in Kapitel 1.2 Vorbereitung der Fallstudien beschriebenen Maßnahmen zur Einrichtung und Konfiguration des SAP BW Systems vom Systemadministrator durchgeführt worden sind.

Neben zahlreichen Screenshots und Abbildungen wird der Leser bei der Arbeit mit diesem Buch zusätzlich durch das Textlayout unterstützt. In herausgestellten Kästen finden sich bedeutsame Definitionen, Hinweise der Autoren und Zusammenfassungen sowie die Verknüpfung von theoretischem und praktischem Teil.

Innerhalb der Fallstudien sind Bezeichnungen in Bildschirmmasken stets *kursiv* und zu betätigende Schaltflächen oder über die Tastatur einzugebende Zeichenketten immer **fett** gekennzeichnet.

## 1.2 Vorbereitung der Fallstudien

Voraussetzung für eine erfolgreiche Durchführung der Fallstudien ist eine installierte SAP BW Instanz. Den Teilnehmern der Fallstudie sollte durch den Systemadministrator im System eine InfoArea sowie eine Anwendungskomponente zugewiesen werden, außerdem ist ein Quellsystem des Typs Filesystem (*Metadaten manuell, Daten über Dateischnittstelle*) mit **A\_QS\_FSBW** als *Logischer Systemname* und **Externe Daten Fallstudie BW** als *Name des Quellsystems* anzulegen.

Es wird empfohlen, die Zugangsdaten der Teilnehmer derart zu gestalten, dass der Benutzername mit einer dreistelligen Zahlenkombination abschließt. Diese Zahlenkette bezeichnet Gruppe (erste Position) und Benutzer ID (zweite und dritte Position) und wird von den Teilnehmern im Verlauf der Fallstudien zur eindeutigen Benennung von Objekten benötigt.

Neben dem SAP BW System müssen auf den Rechnern der Teilnehmer das Microsoft Office Paket in der Version 2000 oder neuer und ein Texteditor installiert sein. Die Dateien, auf deren Daten die Fallstudie basiert, müssen zum Beispiel über das Internet herunterladbar sein oder über ein lokales Netzwerk bereitgestellt werden.

# 2 Theoretische Grundlagen der Data Warehouse Technologie

## 2.1 Einführung

In diesem Kapitel werden die Data Warehouse Systeme in die Systemlandschaft eines Unternehmens eingeordnet. Es wird darauf eingegangen, zu welchem Zweck ein Data Warehouse eingesetzt wird und welche Charakteristika es beschreiben.

### 2.1.1 Bedeutung der Information für die Entscheidungsunterstützung

Entscheidungsträger haben das bedeutsame Potential der Information zur positiven Beeinflussung der Wettbewerbsposition des Unternehmens erkannt (vgl. Gupta, 1997). Der Grad der Automatisierung in der Geschäftsprozessabwicklung ist so weit angestiegen (vgl. Fuller, 2002b), dass durch weitere Investitionen in diesem Bereich nur geringe Verbesserungen der Produktivität und Profitabilität zu erwarten sind (vgl. Anahory und Murray, 1997).

Information ist handlungsbestimmendes Wissen über historische, gegenwärtige und zukünftige Zustände der Wirklichkeit und Vorgänge in der Wirklichkeit, mit anderen Worten: Information ist Reduktion von Ungewissheit (Heinrich, 1999).

Die Anforderungen an Informationsintegration und -versorgung haben sich in den Bereichen der Flexibilität, Verfügbarkeit und Qualität weiterentwickelt. Eine auf die Bedürfnisse der Anwender zugeschnittene Bereitstellung der Daten und anpassungsfähigere Auswertungen können neue Informationssysteme leisten. Eine Steigerung der Informationsqualität und -quantität hängt dagegen vom Prozess der Informationsgewinnung aus intern und extern beschafften Daten ab. Diese beeinflusst die erfolgreiche Planung und Umsetzung von Maßnahmen und dadurch indirekt den Unternehmenserfolg positiv (vgl. Holthuis, 1999).

Daten sind zum Zwecke der Verarbeitung zusammengefasste Zeichen (Rautenstrauch, 2001).

Typische Fragestellungen, die mit entscheidungsunterstützenden Systemen beantwortet werden können, sind Analysen zur Bewertung der Rentabilität von Standorten und die Überprüfung des Produktportfolios auf Ladenhüter und Verkaufsrenner. Im Fokus steht nicht nur eine auf numerische Werte beschränkte Auswertung, sondern besonders eine graphisch aufbereitete Präsentation der Informationen, in der kritische Aspekte eines Sachverhalts hervorgehoben werden können. Für wiederkehrende oder periodische Analysen werden vorgefertigte Berichte eingesetzt (Reporting), individuell gestaltete und flexible Anfragen werden als Queries bezeichnet (vgl. Sattler und Saake, 2004).

Die in Unternehmen weit verbreiteten operativen Systeme wurden für die Erfassung, Verarbeitung und Verwaltung auf der Datenebene konzipiert. Ein typisches Beispiel hierfür ist das Anzeigen der vollständigen Kontaktdaten eines Geschäftspartners nach Eingabe der Kundennummer in eine Adressverwaltungssoftware. Die Systeme stellen eine Sicht auf die Daten bereit, vollziehen jedoch nicht den Schritt zur Generierung von Information (vgl. Holthuis, 1999).

Gleichzeitig mit dem gewachsenen Informationsbedarf haben sich auch die Organisationsstrukturen dahingehend verändert, dass ein unternehmensweiter, ganzheitlicher Überblick für die Unternehmensführung höchste Priorität hat (vgl. Gupta, 1997). Dieser Anspruch macht eine detaillierte und zugleich flexible Sicht auf die Daten, wie sie zuvor nur auf der Ebene der jeweiligen Abteilung gegeben war, zwingend erforderlich. Die Unternehmensführung hat somit direkten Zugriff auf die Daten, so dass eine hohe Aktualität der Informationen gewährleistet ist sowie eine wirksame und aktive Steuerung und Kontrolle untergeordneter Instanzen. In der heutigen Konkurrenzsituation globaler Märkte ist es unabdingbar, Tendenzen frühzeitig zu erkennen und schnell auf Veränderungen zu reagieren. Außerdem sollen Vergleiche mit externen Marktdaten ermöglicht und die Ergebnisse dieser in die Entscheidungsfindung mit einbezogen werden. Unternehmensweite Informationssysteme haben das Potential, einen großen Beitrag zur Bewältigung dieser Aufgaben zu leisten (vgl. Devlin, 1997).

Im Folgenden werden zunächst die Eigenschaften operativer Systeme erläutert und denen des Data Warehouse gegenübergestellt, um eine klare Abgrenzung der Systeme zu erreichen.

## **2.1.2 Operative Systeme**

Die operativen Systeme unterstützen Anwender, die durch Transaktionen sämtliche in Geschäftsprozessen anfallende Daten erfassen, verarbeiten und verwalten. Wird kein vollständig integriertes System wie SAP R/3 eingesetzt, so obliegt die Verantwortung für Einsatz und Betrieb der operativen Systeme den Abteilungen. Der Umfang der erfassten Daten und die Funktionalität gehen nicht über die Abteilungsgrenzen hinaus, so dass sich die Systeme stark hinsichtlich der Form der gespeicherten Daten und der unterstützten Aufgaben unterscheiden (vgl. Smith, 2002).

In der Regel bezieht ein jedes operatives System seine Daten aus lediglich einer Quelle, das Volumen der Daten eines Systems beträgt dabei bis zu mehreren Gigabyte (vgl. Bauer und Günzel, 2001). Diese Daten lassen sich charakterisieren als nicht abgeleitet, zeitaktuell, dynamisch und unabhängig. Abgeleitete Daten gehen aus anderen Daten durch Anwendung einer Berechnungsvorschrift hervor, zum Beispiel wird der Deckungsbeitrag eines Produktes durch die Differenz aus Erlösen und variablen Kosten bestimmt.

Zeitaktualität und Dynamik bedeuten, dass stets ausschließlich die aktuellste Version eines Datenwertes gehalten wird. Zusammengefasst repräsentieren alle diese Werte die derzeitige Situation des Unternehmens. Die Daten unterliegen häufigen Änderungsvorgängen, die vorherige Version wird jeweils überschrieben. Unter abhängigen Daten versteht man Interdependenzen zwischen zwei oder mehr Elementen, zum Beispiel kann man in Deutschland aus der Kombination von Straße, Hausnummer und Ort eindeutig auf die Postleitzahl schließen. Im Gegensatz zu abgeleiteten Daten handelt es sich lediglich um eine Zuordnung, es liegt keine Funktion oder Regel zugrunde.

Die vom Benutzer ausgeführten Transaktionen umfassen das Lesen, Anlegen, Modifizieren und Löschen von Datensätzen, deshalb sind die Systeme für kurze Lese- und Schreiboperationen und für wiederkehrende Abläufe optimiert (vgl. Gupta, 1997). Typische Anfragen sind vordefiniert, einfach strukturiert und betreffen nur wenige Datensätze, meist werden lediglich einzelne Tupel der Datenbank angefordert. In der Regel greifen viele Anwender, die aus unterschiedlichen Fachbereichen stammen können, auf die Systeme zu (vgl. Sattler und Saake, 2004).

Ansätze, um die zur Entscheidungsunterstützung gewünschten Informationen direkt aus den operativen Systemen zu gewinnen, liefern kein befriedigendes Ergebnis (vgl. Inmon, 2001). Es bestehen Schwierigkeiten für die Anwender, in der heterogenen, daher komplexen und unübersichtlichen Systemlandschaft alle benötigten Daten aufzufinden. Darüber hinaus stehen dem einzelnen Anwender unter Umständen nicht alle erforderlichen Berechtigungen für den Zugriff zur Verfügung. Es fehlt oft eine hinreichende Dokumentation der Daten, so dass Anfragen nicht exakt formuliert werden können und so nicht das erwartete Ergebnis liefern. Um Anfragergebnisse schließlich in seiner Arbeitsumgebung verwenden zu können, muss der Benutzer die Daten zunächst in diese transferieren. Aufgrund des Zeitpunktbezugs der Daten lassen sich keine Trends ableiten, die Antwortzeiten bei Analysen über die Datenbestände mehrerer operativer Systeme übersteigen schnell ein von den Nutzern akzeptiertes Maß und verzögern zudem die Verarbeitung geschäftsprozessrelevanter Transaktionen anderer Anwender (vgl. Devlin, 1997).

Die operativen Systeme können die Anforderungen der Anwender wie Manager, Abteilungsleiter und Fachkräfte an die Entscheidungsunterstützung also nicht hinreichend erfüllen. Es kommt somit zu einer Unterscheidung von den Informationssystemen, die nicht die Verwaltung, sondern die Analyse der Daten als zentrale Aufgabe haben. Entscheidungsträger entwickeln häufig erst beim Betrachten einer Auswertung, die verschiedene Perspektiven auf die Daten zulässt, Ideen und Strategien zur Lösung des anstehenden Problems. Eine mög-

lichst große Flexibilität hinsichtlich der Auswahl der Perspektive unmittelbar während der Betrachtung ist wünschenswert (vgl. Holthuis, 1999).

Um eine flexible und leistungsfähige Entscheidungsunterstützung zu gewährleisten, ist es notwendig, alle zur Analyse benötigten Daten in einer einzigen Quelle konsistent und nicht redundant zu speichern. Dazu müssen die Daten verschiedener Quellen in ein einheitliches Format überführt werden, Datenqualität und Datenschutz sind von besonderer Bedeutung. Die IT-Abteilung wird durch den Betrieb eines Informationssystems, das auf einen eigenen Datenbestand zugreift, weniger belastet als durch eine direkt auf den operativen Systemen basierende Lösung (vgl. Inmon, 2001). Diese bereitet Probleme, wenn operative Systeme Änderungen unterzogen werden, da eine Anpassung der Informationssysteme unmittelbar erforderlich ist. Eine Konsolidierung der Daten lassen die Eigenschaften der operativen Systeme nicht zu, zum Beispiel kann eine redundante Datenhaltung oft nicht vermieden werden (vgl. Devlin, 1997).

### 2.1.3 Data Warehouse Systeme

Der Begriff Data Warehouse wird häufig synonym mit dem Begriff Data Warehouse System gebraucht. In dieser Arbeit wird daher ebenfalls der verkürzte Ausdruck Data Warehouse verwendet, lediglich in Kapitel 2.2.3 Komponenten, in dem detailliert die Komponenten des Data Warehouse Systems vorgestellt werden, wird eine strenge begriffliche Unterscheidung vorgenommen. Inmon definierte Data Warehouse wie folgt:

A Data Warehouse is a subject-oriented, integrated, non-volatile and time-variant collection of data in support of management decisions (vgl. Inmon, 1999a).

In dieser Definition sind Fachorientierung sowie eine integrierte und persistente Haltung historischer Daten zentrale Kriterien für die Unterstützung von Entscheidungsprozessen der Unternehmen. Eine Umsetzung der Definition findet sich in den Versprechen wieder, die mit einem Data Warehouse verbunden werden (vgl. Inmon 2001):

- Möglichkeit des Zugriffs auf Daten, die inkonsistent in Altsystemen eines Unternehmens vorliegen
- Erreichen eines hohen Grades an Integration bei der Informationsverarbeitung
- Nutzen aus neuen Plattformen und neuer Softwaretechnologie ziehen
- Schaffung, Verwaltung und Möglichkeit zur wettbewerbsfähigen Nutzung von Informationen, nicht Daten
- Bereitstellen von Strukturen, um historische wie auch aktuelle Daten handhaben zu können



Der Data Warehouse Prozess, der alle Schritte der Beschaffung, der Modellierung und der Analyse umfasst, wird als Data Warehousing bezeichnet (vgl. Bauer und Günzel, 2001).

Ein Data Warehouse überwindet die Heterogenität auf System-, Schema- und Datenebene (vgl. Sattler und Saake, 2004). Es bezieht seine Daten sowohl aus unternehmensexternen Quellen als auch aus einer Vielzahl verschiedener -interner operativer Systeme (Überwindung der Heterogenität auf Systemebene). In den Quellsystemen sind die Daten nach unterschiedlichen Schemata abgelegt, dies betrifft zum Beispiel Abhängigkeiten zwischen Datensätzen. Daher müssen die Daten vor dem Einfügen in das Data Warehouse in ein einheitliches Format überführt werden, zusätzlich müssen etwa Duplikate entfernt werden (Homogenität auf Datenebene). Sind die Daten einmal in den Datenbestand eingefügt, so werden sie in der Regel nicht mehr verändert. Dies geschieht nur in Ausnahmefällen, unter anderem wenn sich Daten im Nachhinein als fehlerhaft erwiesen. Änderungen im Datenbestand der operativen Systeme wirken sich erst nach der nächsten Extraktion aus, der Grad der Aktualität der Daten des Data Warehouse hängt somit von der Häufigkeit des Ladevorgangs ab (vgl. Holthuis, 1999).

Die Anfragen an ein Data Warehouse sind komplex und betreffen eine Vielzahl Datensätze, dies führt zu längeren Antwortzeiten als sie aus dem Einsatz operativer Systeme bekannt sind (vgl. Gupta, 1997). Sie erfolgen ausschließlich lesend und die Ergebnisse sind aufgrund der Nicht-Volatilität des Datenbestandes jederzeit reproduzierbar. Um die genannten Anforderungen zu erfüllen, werden in einem Data Warehouse multidimensionale Strukturen zur Datenhaltung verwendet.

## 2.2 Daten und Architektur

Es existieren im Unternehmen verschiedene Typen von Daten, die sich nicht alle für eine Verwendung im Data Warehouse eignen. Lediglich Geschäfts- und Metadaten können in einem solchen System erfasst werden. Nach der Unterscheidung der beiden Datenarten wird die typische logische Architektur eines Data Warehouse erläutert.

### 2.2.1 Geschäftsdaten

Geschäftsdaten werden bei der Abarbeitung von Geschäftsprozessen verwendet und sind das Betrachtungsobjekt von Auswertungen. Man unterscheidet sie in drei Schichten: Echtzeitdaten sind die Datenbasis für die operativen Systeme. Sie sind hochaktuell, sehr detailliert und werden im laufenden Betrieb durch Schreib- und Leseoperationen manipuliert. In dieser nicht konsolidierten Form werden die Echtzeitdaten nicht zum Entscheidungsprozeß herangezogen und daher nicht im Data Warehouse dauerhaft gespeichert, sie bilden jedoch die Grundlage für alle weiteren Schichten (vgl. Devlin, 1997). Als Beispiel können die Umsätze einer Periode

in den Vereinigten Staaten von Amerika (in US-Dollar) und in Europa (in Euro) dienen.

Konsolidierte Daten haben historischen Charakter und wurden mit Betonung auf höchste systemweite Konsistenz aus den Echtzeitdaten erzeugt. Diese Konsistenz betrifft sowohl den Datentyp als auch die Semantik und den Zeitbezug. Das Datenmodell muss gegenwärtige und zukünftig mögliche Beziehungen zwischen den Daten abbilden können. Im Konsolidierungsprozess werden keine neuen Daten hinzugefügt. Für das obige Beispiel bedeutet dies, dass die Umsätze der verschiedenen Kontinente in die Einheitswährung Euro überführt werden.

Abgeleitete Daten werden traditionell zur Entscheidungsunterstützung genutzt. Sie werden durch einen wohldefinierten Prozess aus konsolidierten Daten gewonnen, der Aggregationsgrad hängt dabei von den Anforderungen der jeweiligen Aufgabe ab (vgl. Lambert, 1996). So kann eine abgeleitete Größe ohne Umformung aus der Menge konsolidierter Daten hervorgehen oder durch die Kombination vorhandener, logisch zusammenhängender Daten angereichert werden. Die Verwendung abgeleiteter Daten für Anfragen verbessert das Antwortzeitverhalten, da aggregierende Operationen bereits bei der Ableitung aus den konsolidierten Daten erfolgt sind und nicht erneut ausgeführt werden müssen. Im Beispiel werden die konsolidierten Umsätze zu einem Gesamtumsatz der Periode aufsummiert und bilden einen einzigen Datensatz.

Extern bezogene Daten müssen zur Sicherstellung von Konsistenz und Qualität zunächst in Geschäftsdaten überführt und somit mit dem lokalen Unternehmensdatenmodell vereint werden, bevor sie im Data Warehouse verwendet werden können (vgl. Devlin, 1997).

Die dauerhafte Speicherung konsolidierter Daten erscheint zunächst überflüssig und eine temporäre Datenhaltung ausreichend. Gegen diese Verfahrensweise spricht, dass neue, nicht vorhergesehene Bedarfe an Daten schneller und mit weniger Aufwand befriedigt werden können weil eine erneute aufwändige Konsolidierung entfällt. Unter Umständen wird das Unternehmensdatenmodell allein in dieser Schicht verwirklicht, dadurch wird die Verständlichkeit des Modells erhöht. Dies kann nützlich sein, wenn sich bei der Entwicklung einer neuen Applikation auf das Modell bezogen wird.

Im Bereich der Konsolidierung konzentriert sich die Verantwortung für die Sicherung der Datenqualität. Um dafür eine langfristig stabile Grundlage zu schaffen, müssen die konsolidierten Daten persistent gespeichert werden. Zudem wird redundante Datenhaltung im Entscheidungsunterstützungssystem vermieden, da eine einzige Quelle für die abgeleiteten Daten geschaffen wird und nicht mehr auf die Echtzeitdaten verschiedener operativer Systeme zugegriffen werden muss. Besonders das Reengineering operativer Systeme wird durch die Trennung der Systemlandschaft in OLTP-System und Entscheidungsunterstützungssystem erleichtert, da historische Daten hoher Granularität bereits im Data Warehouse abgelegt und für Analysen verfügbar sind (vgl. Devlin, 1997).

### 2.2.2 Metadaten

Nur in Verbindung mit Metadaten können gespeicherte Daten aussagekräftig interpretiert werden. Die Ziffer 5 wird erst durch den Zusatz der Einheit Euro als Geldbetrag verstanden.

Unter Metadaten versteht man die Abstraktion betrieblicher Datenobjekte. Sie beschreiben unternehmensweit Daten, Funktionen sowie Komponenten, dokumentieren die Verknüpfungen zwischen Daten und Geschäftsprozessen und dienen der Organisation der Benutzerberechtigungen.

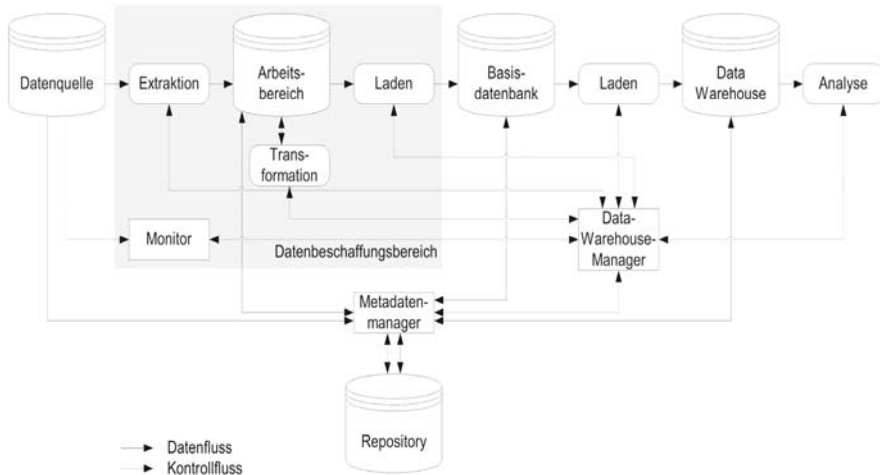
Metadaten werden bei der Entwicklung und Anwendung der operativen Systeme erstellt. Sie befinden sich zum Beispiel in Datenbankkatalogen oder werden von Werkzeugen für die Softwareentwicklung erstellt (vgl. Inmon 1999e). Sie werden im Rahmen der Datenmodellierung und Datenübernahme ins Data Warehouse übertragen und konsolidiert (vgl. Inmon, 1999c). Daten externer Quellen werden um entsprechende Metadaten ergänzt, bevor sie in das System aufgenommen werden. Erst die Metadaten ermöglichen ein schnelles und sicheres Auffinden der benötigten Daten. Aus Anwendersicht unterteilen sich die Metadaten in betriebswirtschaftliche und technische (vgl. Fuller, 2002c, und Inmon, 1999g). Betriebswirtschaftliche Metadaten sind für geschäftliche Anwender relevant (vgl. McDonald et al., 2002), es handelt sich um Kontextinformationen wie etwa die Terminologie betrieblicher Kommunikation oder das Unternehmensmodell. Technische Metadaten dagegen richten sich an das IT-Personal, sie liefern Informationen, die zur Administration der Systeme benötigt werden wie etwa den Datentyp oder Zugriffstatistiken (vgl. Devlin, 1997).

Die Metadaten werden im Metadaten Repository konsistent abgelegt und können dort bei Bedarf eingesehen werden. Es wird mit dem Ziel verwendet, einen optimalen Informationsgewinn für alle Anwender zu gewährleisten und so den Aufwand für Aufbau und Betrieb des Data Warehouse zu minimieren (vgl. Inmon, 1999f). Metadaten sind stabil, d.h. sie unterliegen nur geringen zeitlichen Änderungen. Findet eine Änderung einzelner Metadaten statt, müssen vorherige Versionen im System verbleiben, um das Verständnis zeitlich zurückliegender Datensätze zu erhalten (vgl. Sattler und Saake, 2004, und Inmon, 1999h). Mit dem Common Warehouse Metamodel (OMG Standard 1999) und dem Open Information Model (Standard der Metadata Coalition) wurden Standards zur Definition und zum Austausch von Metadaten formuliert (vgl. Inmon, 1999e).

### 2.2.3 Komponenten

Die Komponenten eines Data Warehouse Systems können in drei Funktionsgruppen gegliedert werden. Arbeitsbereich, Basisdatenbank, Data Warehouse und Repository dienen der Datenhaltung. Extraktion, Transformation, Laden und Analysewerkzeuge unterstützen den Anwender im Data Warehouse Prozess. Die Steuerungs- und Kontrollfunktionen werden durch Data Warehouse Manager,

Metadaten Manager und Monitor bereitgestellt. Nachstehende **Abb. 2.1**. Komponenten eines Data Warehouse Systems (vgl. Bauer und Günzel, 2001) zeigt mit der Referenzarchitektur die Anordnung der Komponenten im Data Warehousing und ihre Beziehungen:



**Abb. 2.1.** Komponenten eines Data Warehouse Systems (vgl. Bauer und Günzel, 2001)

Der Arbeitsbereich ist eine der zentralen Datenhaltungskomponenten und dient der temporären Speicherung der Echtzeitdaten. Die zur Konsolidierung notwendigen Operationen werden direkt im Arbeitsbereich ausgeführt, nach Abschluss der Transformation werden die Daten in die Basisdatenbank geladen. Die Quelldaten der operativen Systeme bleiben somit unverändert und die Übernahme fehlerhafter Daten in das Data Warehouse wird vermieden.

Aufgabe der Basisdatenbank ist die Bereitstellung detaillierter, historischer, konsistenter, modellierter und normalisierter Daten für die Weiterverwendung im Data Warehouse. Das Data Warehouse enthält die abgeleiteten Daten und stellt sie für Analysezwecke bereit. Die Metadaten über Geschäftsdaten, Funktionen und Abläufe sind im Repository abgelegt.

Ausschnitte der Datenbasis des Data Warehouse Systems aus Basisdatenbank und Data Warehouse können aus Gründen der Performance und Übersichtlichkeit in so genannten Data Marts redundant gespeichert werden (vgl. Inmon, 1999d). Da sich die zu durchsuchende Datenmenge verkleinert, wird die Verarbeitung von Anfragen beschleunigt. Außerdem kann durch die physische Partitionierung eine effiziente Zugriffssteuerung vorgenommen werden (vgl. Holthuis, 1999).

Der Einsatz von Data Marts muss sorgfältig überdacht werden, da den Vorteilen einige Nachteile gegenüberstehen. Die Wahrung der Konsistenz wird erschwert, da unter Umständen einzelne Abteilungen individuelle Veränderungen nur lokal in Data Marts vornehmen, um diese möglichst gut ihren Bedürfnissen anzupassen. Der Betrieb von Data Marts verursacht hohe Kosten für zusätzliche Hard- und Software, einmal implementierte Strategien können zudem nicht ohne

erheblichen Aufwand geändert werden. Es ist also empfehlenswert, die Data Marts ausschließlich über das zentrale Data Warehouse System zu befüllen (vgl. Anahory und Murray, 1997).

Die Übertragung der Daten aus den internen und externen Datenquellen in den Arbeitsbereich erfolgt über die Extraktionskomponente. Die Funktion kann periodisch, auf Anfrage, ereignisgesteuert oder sofort gestartet werden und überträgt die Daten über Schnittstellen in das Data Warehouse System. Die Transformationskomponente bereitet die Daten für den Konsolidierungsprozess auf, sie werden in ein einheitliches Format überführt und von Fehlern bereinigt. Zum Erkennen von Auffälligkeiten können Data Mining Verfahren ausgeführt werden. Nach der Transformation überträgt die Ladekomponente die Daten zur persistenten Speicherung in die Basisdatenbank und von dort in das Data Warehouse. Die Ladevorgänge können online (die Datenbasis steht den Nutzern weiterhin zur Verfügung) oder offline (die Datenbasis wird für den Ladevorgang gesperrt) erfolgen.

Die Steuerung der Metadatenverwaltung erfolgt über den Metadaten Manager. Er stellt Werkzeuge für Zugriff, Anfrage, Navigation sowie Versions- und Konfigurationsverwaltung bereit. Monitoring Software überwacht Datenquellen auf Manipulationen des Datenbestandes. Es können verschiedene Strategien zur Überwachung eingesetzt werden, so können zum Beispiel aktive Datenbankmechanismen (Trigger) bei einer Änderung einer Tabelle ausgelöst werden. Eine weitere Möglichkeit ist die Verwendung von Log – Dateien, welche die einzelnen Änderungen protokollieren, oder Delta-Snapshots, die den kompletten Datenbestand periodisch erfassen und paarweise auf Unterschiede untersucht werden. Auch die den Datensätzen zugeordneten Zeitstempel können für eine Identifikation der seit der letzten Datenübernahme veränderten Tupel verwendet werden (vgl. Bauer und Günzel, 2001).

## 2.3 Datenmodell

Modelle lassen Gesetzmäßigkeiten der Systemstruktur erkennen und helfen dem Betrachter beim Verständnis der Zusammenhänge, um Konsequenzen seiner möglichen Aktionen mit der Umwelt abschätzen zu können.

Ein Modell ist ein abstraktes, immaterielles Abbild realer Strukturen bzw. des realen Verhaltens für Zwecke des Subjekts (vgl. Rautenstrauch, 2003).

Das Datenmodell enthält korrekte und aussagekräftige Definitionen der Unternehmensdaten. Es identifiziert valide konstante Strukturen der Daten, aus denen Informationen abgeleitet werden können, und es zeigt Unterschiede und Gemeinsamkeiten zwischen Daten verschiedener Quellen auf. Daneben enthält es eine Reihe von Regeln zur Datenverwendung, damit die Konsistenz, Korrektheit und Vollständigkeit gewahrt bleiben. Die Abbildung erfolgt auf konzeptueller Ebene, die vom Zielsystem weitgehend unabhängig ist (vgl. Holthuis, 1999).

Das Datenmodell bildet die Grundlage zur Kommunikation zwischen IT-Abteilung und Entscheidungsträgern. Für das Verständnis bei Entscheidungsträgern ist es erforderlich, dass das Modell möglichst einfach und leicht überschaubar und somit ohne DV-Fachkenntnisse schnell nachvollziehbar ist. Mitarbeiter der IT-Abteilung hingegen verwenden es, um die Anwendungen eng an den Erfordernissen des Unternehmens orientiert zu entwickeln. Da das Datenmodell etwa zur Erstellung eines Datenbankschemas herangezogen wird, ist eine hinreichende Detaillierung notwendig. Die Anforderungen von Entscheidungsträgern und IT-Abteilung an das Datenmodell divergieren also. Eine Lösung kann zum Beispiel eine Unterteilung in drei Schichten sein, nämlich Fach-, DV- und Implementierungskonzept. Im Fachkonzept wird die betriebswirtschaftliche Problemstellung semiformal und implementierungsunabhängig beschrieben. Das Fachkonzept wird im DV-Konzept um Anforderungen und Restriktionen der DV-technischen Umsetzung angereichert, die Formulierung der DV-technischen Umsetzung erfolgt schließlich im Implementierungskonzept (vgl. Bauer und Günzel, 2001).

Eine Implementierung dieses Konzepts findet sich in der Modellierungssoftware ARIS Toolset wieder, das mit Organisations-, Daten-, Funktions- und Steuerungssicht das gesamte Unternehmen abbildet. Das Datenmodell ist Teil dieses umfassenden Unternehmensmodells. Die Organisationssicht beinhaltet alle Organisationseinheiten eines Unternehmens und die Beziehungen zwischen ihnen, die Datensicht beschreibt eine Abstraktion der Datenobjekte und ihrer Zusammenhänge. Vorgänge und Funktionen, die durch das Informationssystem unterstützt werden sollen, werden durch die Funktionssicht wiedergegeben. Die zentrale Steuerungssicht stellt die Verbindungen zwischen den Objekten der Organisations-, Daten- und Funktionssicht dar, um Geschäftsprozesse abzubilden (vgl. Karcher, 2005).

Die Verbindungen zwischen den einzelnen Sichten werden durch das ARIS Haus repräsentiert. Alle diese Sichten lassen sich jeweils in Fach-, DV- und Implementierungskonzept unterteilen. Das ARIS Toolset ermöglicht also eine Erstellung umfassender Modelle mit variablem Detaillierungsgrad und bietet daher für Fach- und Führungskräfte gleichermaßen ein großes Nutzenpotential (vgl. Karcher, 2005).

### **2.3.1 Fakten, Kennzahlen und Dimensionen**

Die Geschäftsdaten eines Data Warehouse werden in Fakten, Kennzahlen und Merkmale unterteilt.

Fakten sind numerische Messgrößen, die betriebliche Sachverhalte beschreiben (vgl. Sattler und Saake, 2004).

Fakten werden auch als Basiskennzahlen bezeichnet, da aus ihnen durch arithmetische Operationen weitere Kennzahlen abgeleitet werden. Beispiele sind Umsatz und Kosten als Fakten, aus denen die Kennzahl Gewinn berechnet wird.

Kennzahlen sind Fakten oder aus Fakten abgeleitete Größen. Sie haben informativen Charakter und dienen dazu, durch systematisches Vergleichen Ursachen und Trends ableiten zu können (vgl. Rautenstrauch, 2004).

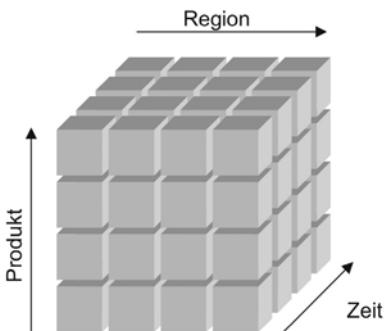
Kennzahlen gewinnen erst an Aussagekraft, wenn ihnen weitere beschreibende Attribute wie ein Verkaufsland oder Quartal zugeordnet werden.

Merkmale sind betriebswirtschaftliche Bezugsgrößen, nach denen eine sinnvolle Gruppierung von Kennzahlen möglich ist (vgl. Mehrwald, 2004).

Das Datenmodell eines Data Warehouse kann anhand eines Würfels visualisiert und als Grundlage einer Anfrage gesehen werden. Dessen Achsen werden als Dimensionen bezeichnet, auf denen die Kriterien zur Einordnung der Daten abgetragen werden, während im Inneren des Würfels an den Schnittpunkten der Dimensionsausprägungen die Daten enthalten sind.

Dimensionen beschreiben die technische Struktur der Daten und so eine mögliche Sicht auf die assoziierte Kennzahl (vgl. Sattler und Saake, 2004).

Der gesamte Würfel ist also wiederum aus mehreren kleinen Würfeln zusammengesetzt, die auf elementarer Ebene einzelne Datensätze repräsentieren. In der Praxis geht die Anzahl der Dimensionen über drei hinaus, dies prägt den Begriff der Multidimensionalität, kann jedoch nicht mehr sinnvoll grafisch veranschaulicht werden.



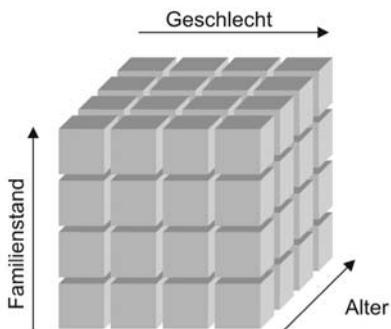
**Abb. 2.2.** Datenwürfel

Die Dimensionen werden hinsichtlich ihres Typs unterschieden, ihre Ausprägungen können diskrete Werte oder einen Wertebereich repräsentieren. Nicht-hierarchische Dimensionen verfügen über eine einfache interne Struktur, die durch keine vertikalen Beziehungen gekennzeichnet ist. Daher ist es nicht möglich, Aggregationen über mehrere Ausprägungen zu bilden (vgl. Holthuis, 1999). Ein Beispiel ist eine Dimension mit den Elementen Ist, Soll und Abweichung. Diese Elemente

können nicht in einem einzigen Hierarchiebaum so angeordnet werden, dass sich jeweils aus der Summe der Kindknoten der Wert des Vaterknotens ergibt.

Hierarchische Dimensionen dagegen ermöglichen unterschiedliche Verdichtungsstufen und Niveaus, die anhand fester Regeln und Berechnungsvorschriften gebildet werden. Eine Produkthierarchie kann aus den Verdichtungsstufen Produkthauptgruppe und Produktgruppe bestehen, die Werte der Produkthauptgruppen sind mit der Summe der jeweils untergeordneten Produktgruppen identisch.

Dimensionen des kategorischen Typs verwendet man, wenn einzelne Ausprägungen für die Problemstellung von herausragender Bedeutung sind. Zum Beispiel können beim Marketing detaillierte Kundeninformationen derart bedeutsam sein, dass die Ausprägungen der Kundendimension eigenen Dimensionen zugeordnet werden.



**Abb. 2.3.** Kategorischer Dimensionstyp

Innerhalb von Dimensionen kann es zu einer Reihe von Strukturanomalien kommen (vgl. Holthuis, 1999). So ist es etwa für die Durchführung von Aggregationsoperationen von Bedeutung, ob einer Dimension ein ausgeglichener Hierarchiebaum zugrunde liegt. Zum Beispiel ist eine weltweite Auswertung nach Bundesländern nicht möglich. Außerdem kann in einem Hierarchiebaum nicht gewährleistet werden, dass jeder Knoten lediglich einen Vaterknoten besitzt (so genannte multiple Hierarchie). So kann eine Rechnungslegungsabteilung für zwei oder mehr übergeordnete Abteilungen tätig sein. Sollen nun auf der übergeordneten Ebene die Personalkosten verglichen werden, so ist eine eindeutige Zurechnung der Kosten dieser Rechnungslegungsabteilung nicht ohne weiteres möglich.

In einer Dimension kann es der Fall sein, dass mehrere Ausprägungen jeweils alle erhobenen Werte enthalten. Die Anzahl der Ausprägungen auf der darunter liegenden Ebene sowie die Aufteilung der Elemente zwischen diesen unterscheiden sich. Als Beispiel kann eine gleichzeitige Unterteilung in der Kundendimension nach Familienstand und Alter dienen. In beiden Gruppen sind sämtliche Kunden erfasst, die Anzahl der Untergruppen kann variieren. Es kann weiterhin erforderlich sein, Dimensionshierarchien zu verändern, wenn beispielsweise eine Produktgruppe so umsatzstark wird, dass sie zur Hauptgruppe aufsteigen soll.



### 2.3.2 Modellierung von Zeit

Da im Data Warehouse historische Daten gespeichert werden, kommt der zeitlichen Dimension eine besondere Bedeutung zu (vgl. Kimball, 1997).

Grundsätzlich ist zwischen Zeitpunkt- und Zeitraumbezug zu unterscheiden (vgl. Holthuis, 1999). Mit zeitpunktbezogenen Daten werden Bestände im Unternehmen erfasst, die Zeit zwischen den Bestandsaufnahmen sollte zur Verstärkung der Aussagekraft konstant gehalten werden. Bei zeitraumbezogenen Daten sind die konsolidierten Daten in der feinsten für die Auswertungen benötigten Granularität in das Data Warehouse einzuspielen. Eine Vergleichbarkeit der erfassten Daten ist nur gegeben, wenn die Erfassungszeiträume identisch sind. Kennzahlen werden in einem Unternehmen unter Umständen mit unterschiedlichem Zeitbezug abgebildet, so können etwa tägliche Umsätze in der Kostenrechnung zu Tageskursen und in der Rechnungslegung mit einem Durchschnittswert des jeweiligen Monats bewertet werden.

Probleme bei der Modellierung von Zeit entstehen auch dadurch, dass bei zeitraumbezogener Erfassung die Daten nicht unbedingt korrekt in die nächst höhere Ebene überführt werden können. So können aus wochenweise erhobenen Daten durch mathematische Operationen keine Monatsdaten erzeugt werden. Außerdem können sich einzelne Zeiträume auf ein- und derselben Stufe überschneiden, wenn etwa Daten nach Jahreszeiten und ebenfalls nach Quartalen erfasst werden. Weitere Schwierigkeiten entstehen, da vielfältige abzubildende Zeiträume existieren (z.B. Kalenderjahr und Fiskaljahr, weitere Abweichungen zwischen unterschiedlichen Staaten möglich).

### 2.3.3 Multidimensionale Anfragestrukturen

Anders als die Datenbanken operativer Systeme, ist das Data Warehouse komplexen Anfragen ausgesetzt, deren Verarbeitung mit einem hohen zeitlichen Aufwand verbunden ist. Die Anfragen sollen sowohl flexibel als auch intuitiv zu erstellen sein, dem Anwender sollen dazu keine tieferen Kenntnisse der internen Strukturen der Datenhaltung abverlangt werden.

Die multidimensionale Struktur gewährleistet kürzere Antwortzeiten, da sie auf zeilenübergreifende Summenoperationen optimiert ist. Ein relationales Datenbanksystem müsste zunächst die korrespondierenden Attribute durch ihre Werte identifizieren. Die Notwendigkeit dieser zeitaufwändigen Join-Operation entfällt bei multidimensionalen Strukturen (vgl. Sattler und Saake, 2004).

Es besteht der Bedarf einer speziellen Anfragesprache für multidimensionale Systeme sowie der Bereitstellung einer leicht zu erlernenden Navigation im Datenbestand über eine intuitive Benutzeroberfläche. Mit der Komplexität einer Anfrage sollte der Anwender möglichst nicht konfrontiert werden. Als letzte Anforderung ist eine entsprechende Ergebnisaufbereitung zu nennen, damit die Resultate vom Anwender in Entscheidungen umgesetzt werden können (vgl. Holthuis, 1999).

**Tabelle 2.1.** OLAP Charakteristika

Typische Operation	Read-only
Ansichten der Daten	Benutzerdefiniert
Datenmenge je Transaktion	Groß
Niveau der Daten	Verdichtet / aufbereitet
Zeitbezug der Daten	Historisch, gegenwärtig oder zukünftig
Verarbeitungseinheit	Matrizen, sachbezogen, übergreifend

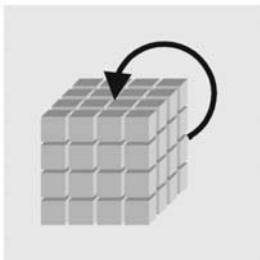
OLAP stellt also unternehmensinterne Informationen in aufbereiteter und verdichteter Form bereit. Von Interesse sind dabei nicht mehr einzelne Transaktionen, sondern die Analyse managementrelevanter Kenngrößen. OLAP eignet sich aufgrund der intuitiven Datenbearbeitung und Navigation innerhalb der verschiedenen Dimensionen und Verdichtungsstufen zur Unterstützung von Situationsanalyse und Planung, es stellt jedoch keine Methoden für die Zielbildung oder Prognose bereit. Es kann lediglich versucht werden, für bestimmte Kenngrößen anhand historischer Daten eine Entwicklung abzuleiten und zu visualisieren. So können mögliche Problemlösungen vorgeschlagen und analysiert werden, die endgültige Entscheidung ist aber stets vom Manager selbst zu treffen.

Das eigentliche Haupteinsatzgebiet des OLAP liegt in der Unterstützung von Kontrollaufgaben wie der Ad hoc-Berichterstattung und dem Exception Reporting. Auch hier bleibt dem Anwender die Interpretation der vom System zurück gelieferten Werte vorbehalten (vgl. Holthuis, 1999).

### 2.3.4 OLAP Operationen

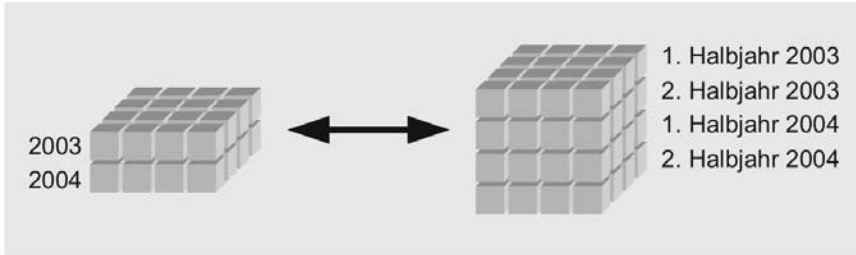
Die typischen Operationen auf multidimensionalen Datenstrukturen während der Datenanalyse werden als Pivotierung, Roll-Up, Drill-Down, Drill-Across, Slicing und Dicing bezeichnet. Sie lassen sich anhand des bereits beschriebenen Datenwürfels erklären (vgl. Sattler und Saake, 2004, und Kimball, 1996a).

**Pivotierung:** Diese Operation dient der Analyse der Daten aus verschiedenen Perspektiven. Hierzu wird der Datenwürfel durch Vertauschen der Dimensionen gedreht, daher wird die Pivotierung ebenfalls als Rotation bezeichnet.

**Abb. 2.4.** Pivotierung

**Roll-Up:** Das Roll-Up erzeugt neue Informationen durch Aggregation entlang des Verdichtungspfades. Die Anzahl der Dimensionen bleibt somit erhalten, es wird nur auf einzelnen Dimensionen ein höheres Verdichtungslevel erreicht.

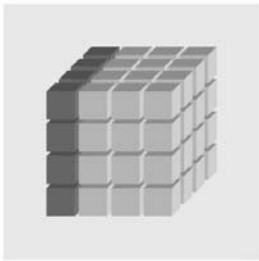
**Drill-Down:** Das Drill-Down verhält sich komplementär zum Roll-Up, es wird anhand einer Dimensionshierarchie von aggregierten zu Detaildaten navigiert.



**Abb. 2.5.** Drill-Down und Roll-Up

**Drill-Across:** Wechsel von einem Datenwürfel zu einem anderen, dies kann etwa erfolgen, wenn Ist- und Soll-Daten in separaten Würfeln gehalten werden.

**Slice:** Durch das Herausschneiden von Schichten aus dem Würfel werden individuelle Sichten erzeugt und die Dimensionalität verringert (zum Beispiel Betrachtung der Werte des aktuellen Jahres).



**Abb. 2.6.** Slice

**Dice:** Die Dice Operation extrahiert einzelne Teilwürfel und schafft so neue Sichten auf die Daten. Die Dimensionalität bleibt erhalten, jedoch verändert sich die Hierarchie einzelner Dimensionen, wenn nur über bestimmte Produkte oder Regionen analysiert wird.

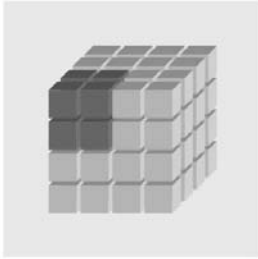


Abb. 2.7. Dice

### 2.3.5 ADAPT

Ein erster Ansatz um multidimensionale Datenstrukturen zu modellieren, ist ADAPT (Application Design for Analytical Processing Technologies). Es wurde mit dem Ziel entwickelt, einen von der Implementation unabhängigen Modellierungsstandard für die Kommunikation zwischen Entwicklern und Anwendern zu schaffen (vgl. Bulos und Forsman, 2001).

ADAPT besteht aus den neun Elementen Hypercube, Dimension, Dimensionselement und Elementgruppe, Hierarchie und Hierarchieebene, Attribut, Berechnungsvorschrift und Contextcube.

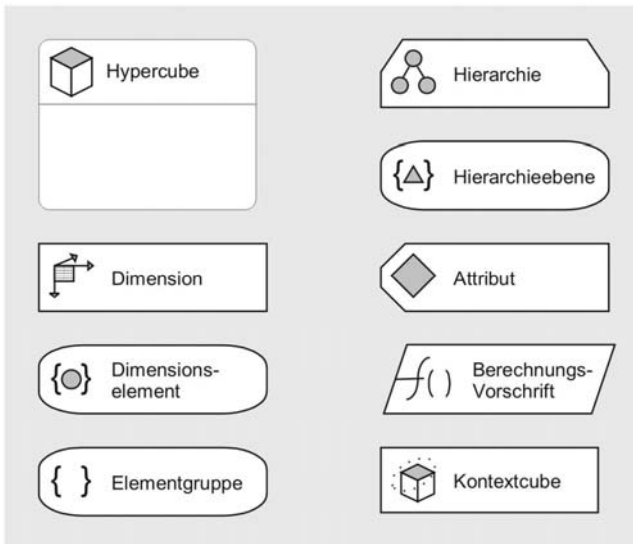


Abb. 2.8. ADAPT Notation