

Zusammenfassung BI

Data Warehouse – Warum

Motivation für DWH:

- Aktuelle Situation ergibt neue Problemstellungen
 - Technologischer Fortschritt
 - Internationale Verflechtung von Unternehmen
 - Liberalisierung der Märkte



- steigende Dynamik
- hohe Vernetzung Intransparenz



Problem: Die Beherrschung derartig komplexer Situationen stellt hohe Anforderungen an das Entscheidungsverhalten.

Digitale Transformation:

Wird auch als „Digitaler Wandel“ bezeichnet und beschreibt den fortlaufenden Veränderungsprozess, denn neue digitale Technologien auf die gesamte Gesellschaft und insbesondere Unternehmen ausüben.

Als Basis ist die immer schneller fortschreitende Entwicklung und damit immer leistungstärkere digitale Technologien.

Definition Data Warehouse

- aus einer oder mehreren operativen Datenbanken extrahierte Datenbank
- fasst alle relevanten Daten für den Geschäftsprozess eines Unternehmen zusammen
- aggregiert die Daten und bereitet diese auf
- aggregiert → anhäufen, zusammenballen, zusammentragen
- umfasst Meta-, die Dimensions- und Aggregationsdaten
- ermöglicht Informationsgestützte Entscheidungen
- beinhaltet notwendige Verwaltungsprozesse (CRUD)

Ziele / Anforderungen an Data Warehouse – Systeme

- Aufbau einer zentralen und konsistenten Datenbasis
- für verschiedene Anwendungen
- zur Unterstützung analytischer Aufgaben von Fach- und Führungskräfte
- losgelöst betrieben von den operativen Datenbanken

Anforderungen nach INMON an ein DWH

- Struktur- und Formatvereinheitlichung (Integration):
 - Ablage der Daten einer Datenstruktur mit einheitlichen Format
- Subjektorientierung:
 - Speicherung orientiert an den Subjekten eines Unternehmens
- Zielraumbezug (time variant):
 - Speicherung aller Daten mit Zeitraumbezug (nicht zeitpunktbezogen)
- Nicht-Volatilität
 - keine Änderung einmal gespeicherter Daten

Denormalisierung →

Auf eine Normalisierung nach Codd wird verzichtet, bzw. ist eine Normalisierung vorhanden, wird diese rückgängig gemacht.

Grund: Reduzierung der Zugriffszeiten und damit Gewinn an Performance.

- Keine 3. Normalform
- Entfernen vorhandener Normalformen
- Steigerung der Performance → keine Joins notwendig

Beispiel:

OLTP-System: 3. NF

product_ID	name	category_ID
2102	Blueberry Milk	8
1508	Buttermilk	8
1302	Schokolade Milk	8
2103	Blueberry Yogurt	12
1307	Strawberry Yogurt	12

↓ join ↓

category_ID	name
8	Milk
12	Yogurt

DWH-System: not 3. NF

product_ID	productname	category_ID	categoryname
2102	Blueberry Milk	8	Milk
1508	Buttermilk	8	Milk
1302	Schokolade Milk	8	Milk
2103	Blueberry Yogurt	12	Yogurt
1307	Strawberry Yogurt	12	Yogurt

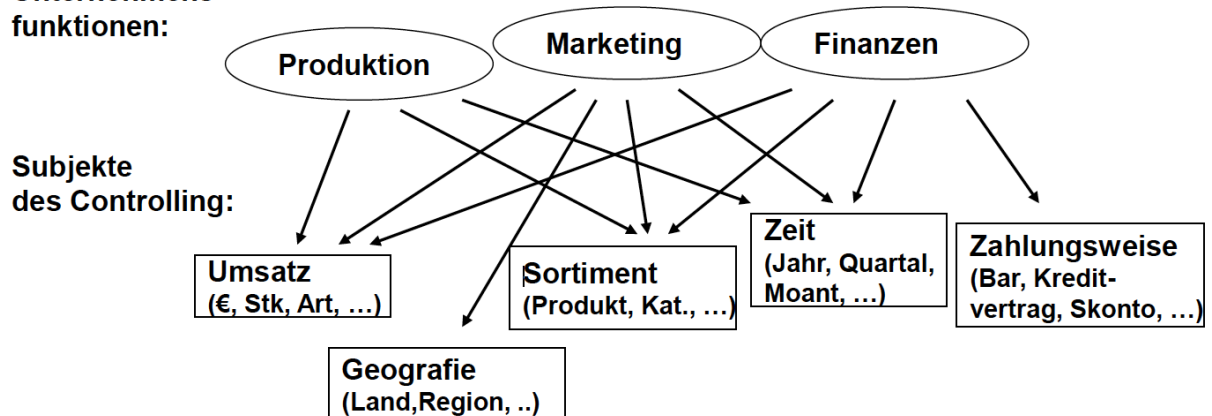
**kein JOIN in Abfrage
notwenig**

Daten-orientiert <--> Subjekt-orientierte Modellierung

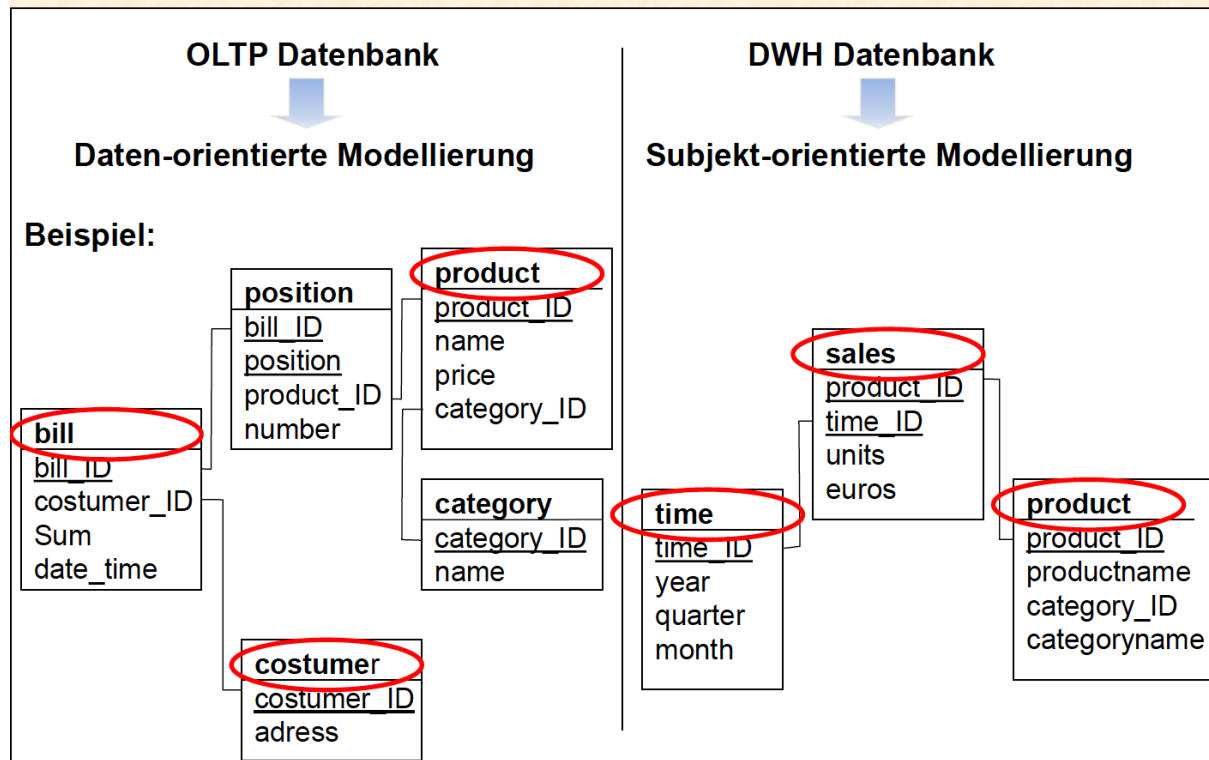
Subjektorientiert:

- Datenbankschema an den Subjekten der Business Analyse und nicht an den Datenbankobjekten (OLTP - **Online-Transaction-Processing** Datenbank) orientiert
- Sinnvoll da unterschiedliche Unternehmensfunktionen gleiche Subjekte des DWH für die Analyse nutzen.
- Beispiel:

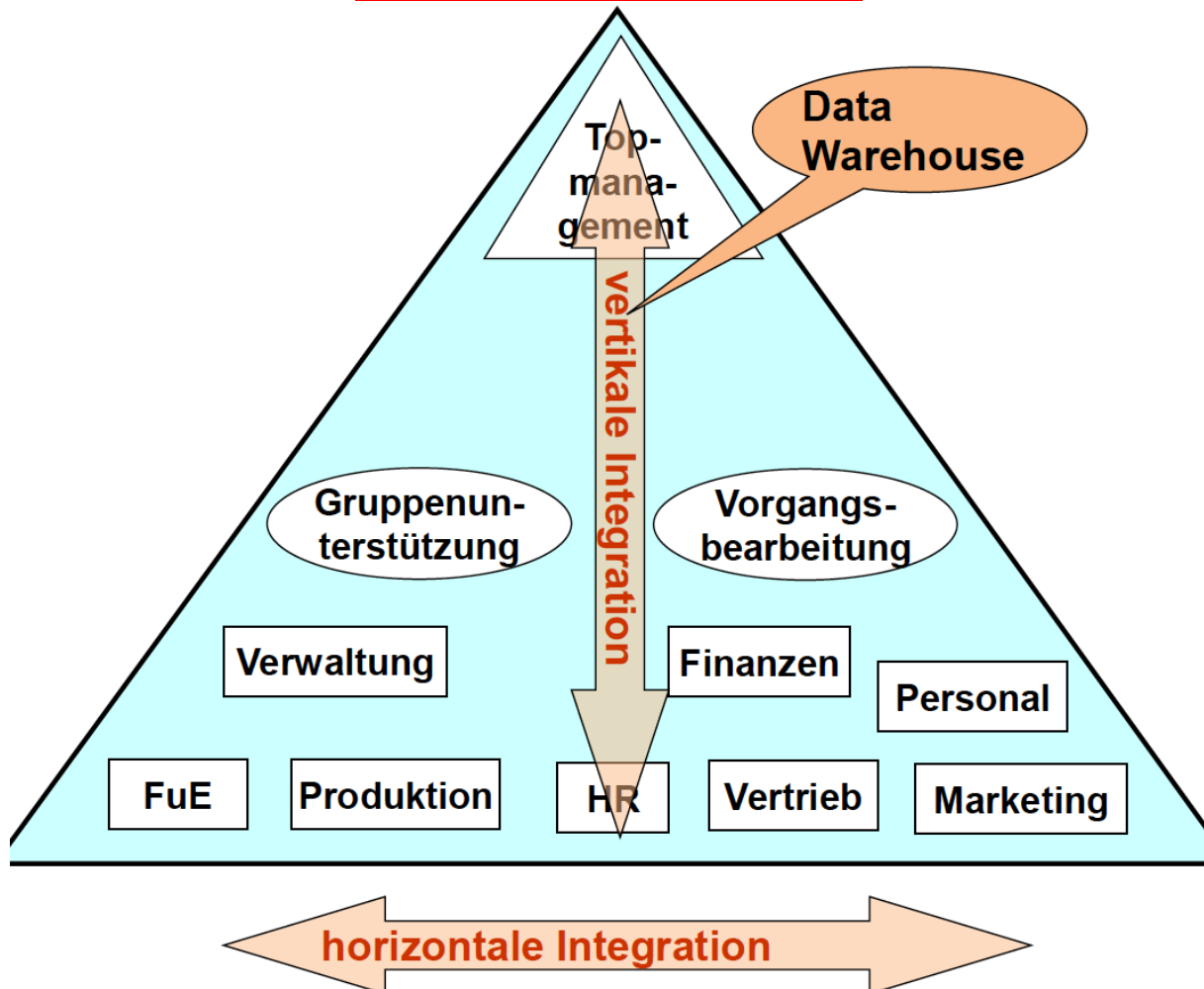
Unternehmensfunktionen:



Daten-orientierte versus Subjekt-orientierte Modellierung



Integration von Informationssystemen



Subjektorientierung

- **Fakten – Kennzahlen**
 - Numerische Messgröße – betriebliche Kennzahlen
 - Bsp. Umsatz, Deckungsbeitrag, Anzahl an Zugriff (Webseite)
- **Dimensionen**
 - Auswertrichtungen – nach den Kennzahlen ausgewertet werden können
 - Beschreiben den Rahmen für die Auswertung der Kennzahlen
 - Sind eine unabhängige Liste an Analyseelemente
 - Orthogonale Struktur des Datenraumes
 - Spannen die Kennzahlen im Raum auf
 - Bsp. Zeit, Geografie, Sortiment, Zahlungsweise
 - Produkt besitzt Attribute → Name, Preis, Subkategorie, Kategorie
- **!!** Beide Subjekte gehören im großen ganzen zusammen
- **Hierarchien** – Sortiment (Hierarchien werden auch Level(Ebene) genant)
 - Kategorie → Subkategorie → Name
- **Attribute** sind Eigenschaften von Dimensionen
 - werden für die Klassifizierung und Filterung der Kennzahlen verwendet
 - können zur Bildung der Hierarchien in der Dimension verwendet werden
 - verwendete Attribute bilden Ebenen der Hierarchie

Dimension: Zeit besitzt Attribute:

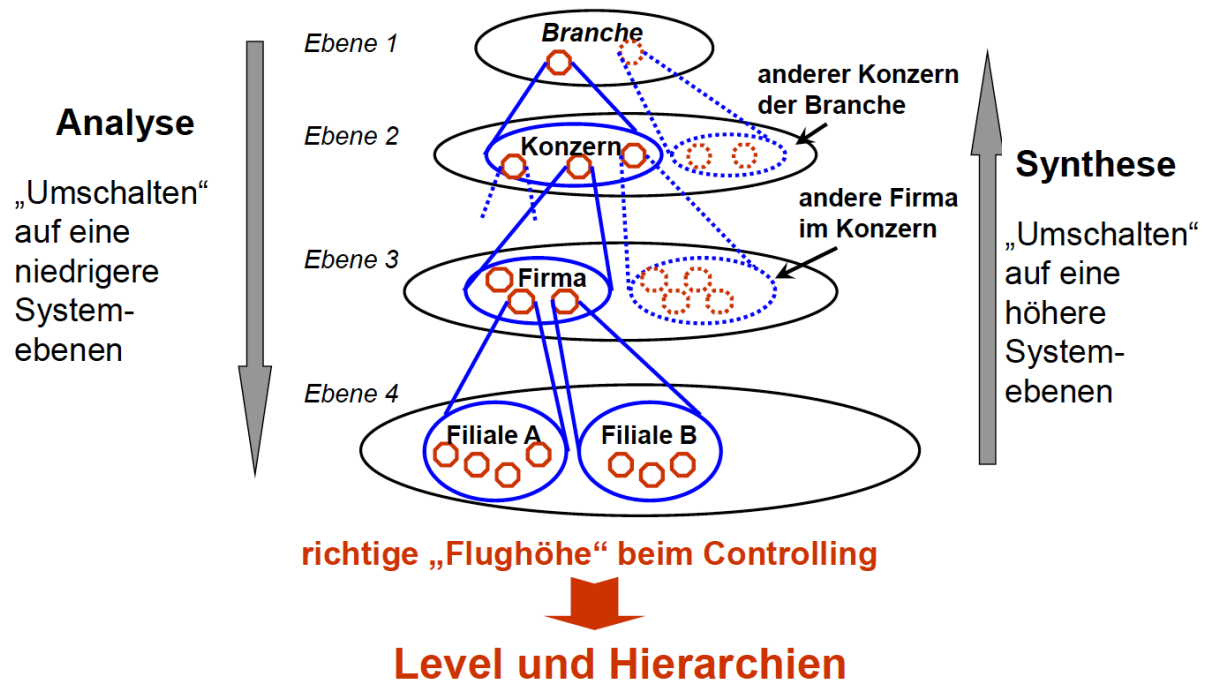
- Jahr, Quartal, Monat, Woche, Tag

Hierarchien:

- Kalender: Jahr → Quartal → Monat
- Gaswirtschaftsjahr: Gasjahr → Quartal → Monat
 - Bsp. GWJ2018
 - Q4/2017 → Okt. Nov. Dez.
 - ... Q3/2018 → Juli, Aug, Sept

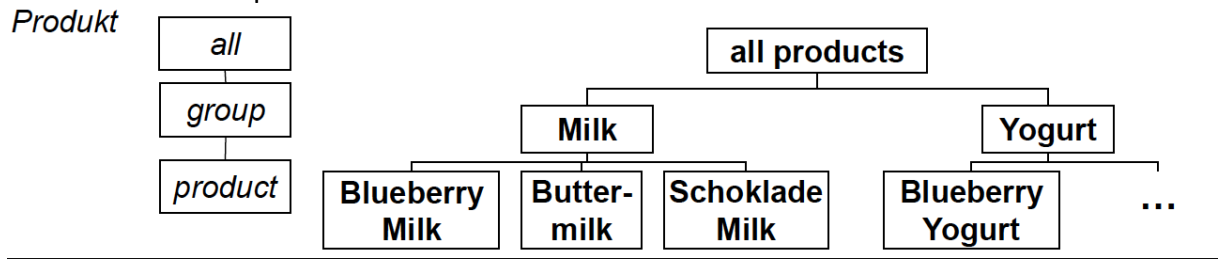
Analyse und Synthese

primäres Ziel: Bewältigung der Komplexität von Systemen

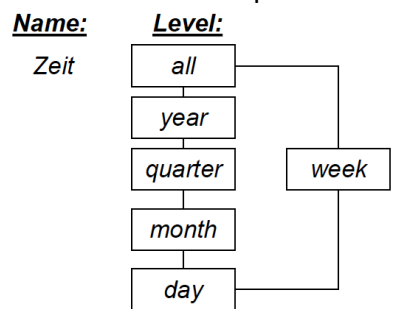


Arten der Dimensionshierarchien

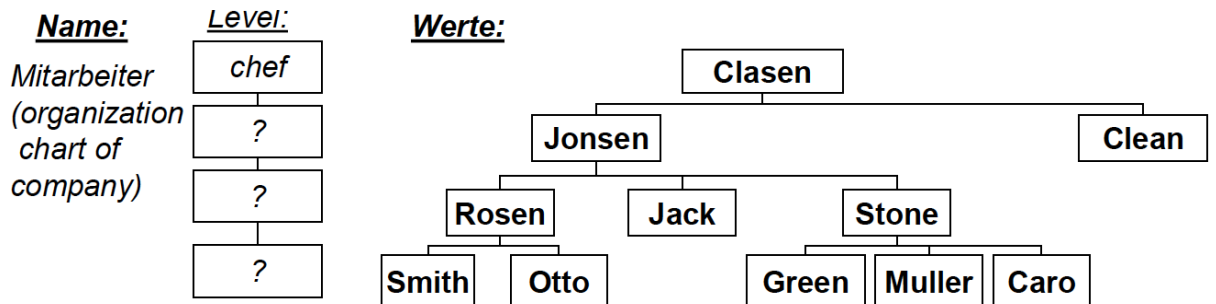
- **Normal = ausgeglichen**
 - Jeder Knoten hat genau einen Vorgänger (Ausnahme: Wurzel)
 - Jeder Weg von der Wurzel zum Blatt hat die gleiche Länge
 - Es existieren keine Lücken in den Werten der Knoten
 - Es existieren keine Lücken in den Werten der Knoten
 - Beispiel:



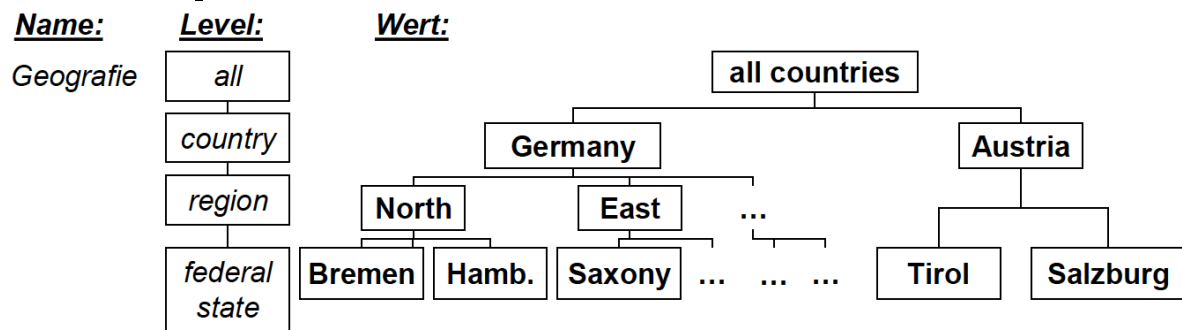
- **Parallel**
 - es gibt Knoten mit mehr als einem übergeordneten Knoten
 - es sind mehr Wege von der Wurzel zu einzelnen Blättern vorhanden
 - es gibt keine Lücken in den Werten der Knoten
 - Beispiel:



- **unausgeglichen**
 - jeder Knoten hat genau einen Vorgänger (Ausnahme: Wurzel)
 - die Wege von der Wurzel zu den Blättern haben unterschiedliche Länge
 - es existieren keine Lücken in den Werten der Knoten
 - Beispiel:



- **unregelmäßig**
 - jeder Knoten hat genau einen Vorgänger
 - in den Zwischenebenen sind zum Teil keine Werte in den Knoten vorhanden
 - _Beispiel:



Kennzahlen

- additive
 - beim Wechsel in ein höheres Level der Hierarchie ist die Summenbildung **immer** erlaubt
- nicht additiv
 - beim Wechsel in ein höheres Level der Hierarchie ist die Summenbildung **nicht** erlaubt
- halb additiv
 - beim Wechsel in ein höheres Level der Hierarchie ist die Summenbildung **teilweise** erlaubt

Relational

- Datenspeicherung im rel. DBMS
- Modellierung der Cubestruktur mittels Relationen (Tabellenform)
- Verwendung des SQL-Standards zur Datenabfrage und -manipulation

Beispiele:

- Star Schema
- Snowflake Schema



ROLAP

- Relational Online Analytical Processing

Nicht Relational

- Datenspeicherung nicht in relationaler Art
- Speicherung des Cube mit klassischen Methoden der Informatik
- Fehlende Standards für die Datenabfrage und -manipulation

Beispiele:

- Arrays
- Hash-Tables
- Bitmap - indices



MOLAP

- Multidimensional Online Analytical Processing

Spezifikation eines Cubes:

Anforderungsdiagramm:

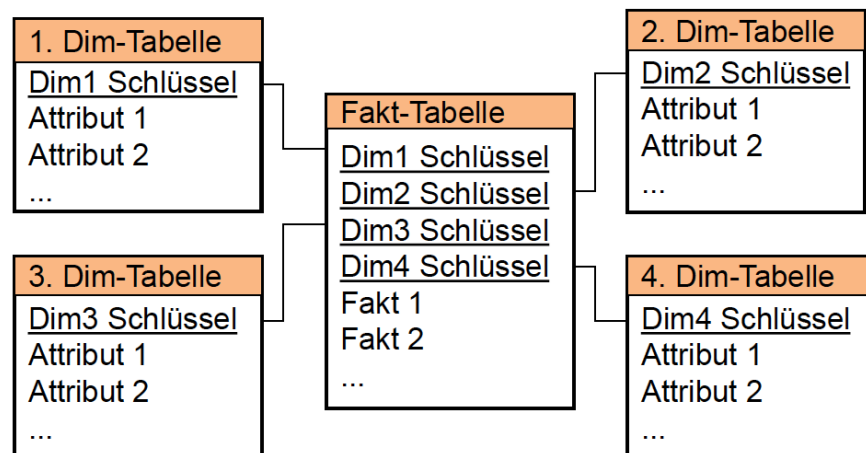
Fakten	Mengenumsatz, Wertumsatz		
Dimensionen	Produkt	Zeit	Geografie
Attribute	<ul style="list-style-type: none">• Kategorie• Name• Farbe	<ul style="list-style-type: none">• Jahr• Monat	<ul style="list-style-type: none">• Land• Region• Filliale
Hierarchien - Level	Sortiment	Kalender	
	Kalender → Name	Jahr → Monat	

DWH-SCHEMA

Star-Schema

- Struktur ermöglicht zur Entscheidungsfindung eine typische Abfrage genutzt werden kann
- Zentrum des Schemas ist die Fakt-Tabelle
- Um die Fakt-Tabelle ordnen sich die Dimensionstabellen
- Verbindungen klassisch über Primär / Fremdschlüssel
- **Verwendet das relationale Datenmodell zur Abbildung multidimensionaler Strukturen**

Aufbau eines Starschemas:



Eigenschaften:

- Bezug mehrere Dimensionstabellen auf eine Fakt-tabelle
- Große Datensatzanzahl in der Fakt-tabelle gegenüber den Dimensionstabelle
- 1:n Beziehung jeder Dim-Tabelle zur Fakt-tabelle
- hohe Abfrageeffizienz
 - Abfrage auf großer Fakt-tabelle mit einfachem JOIN zu kleinen Dim-Tabellen
 - Bildung des JOIN nur zwischen Fakt-tabelle und der jeweiligen Dim-Tabelle
- Einfache Anfrageerstellung durch geringe Tabellenanzahl
- Hoher Aufwand bei Änderung der Dimensionshierarchien

<u>Vorteile</u>	<u>Nachteile</u>
<ul style="list-style-type: none"> • Intuitives Datenmodell • Geringe Anzahl an JOIN-Operationen • Veränderungen und Erweiterungen können leicht umgesetzt werden 	<ul style="list-style-type: none"> • Schlechte Antwortzeiten bei großen Dimensionstabellen • Erhöhter Speicherbedarf in den Dimensionstabellen durch NICHT-Normalisierung • Mehrfaches Speichern identischer Werte → Redundanz in den Dimensionstabellen

Anwendung bei:

- wenn schnelle Abfrageverarbeitungen notwendig sind
- schnell ändernde Datenstrukturen vorliegen
- Dimensionstabellen in ihrer Größe überschaubar bleiben
- viele Nutzer Zugriff benötigen

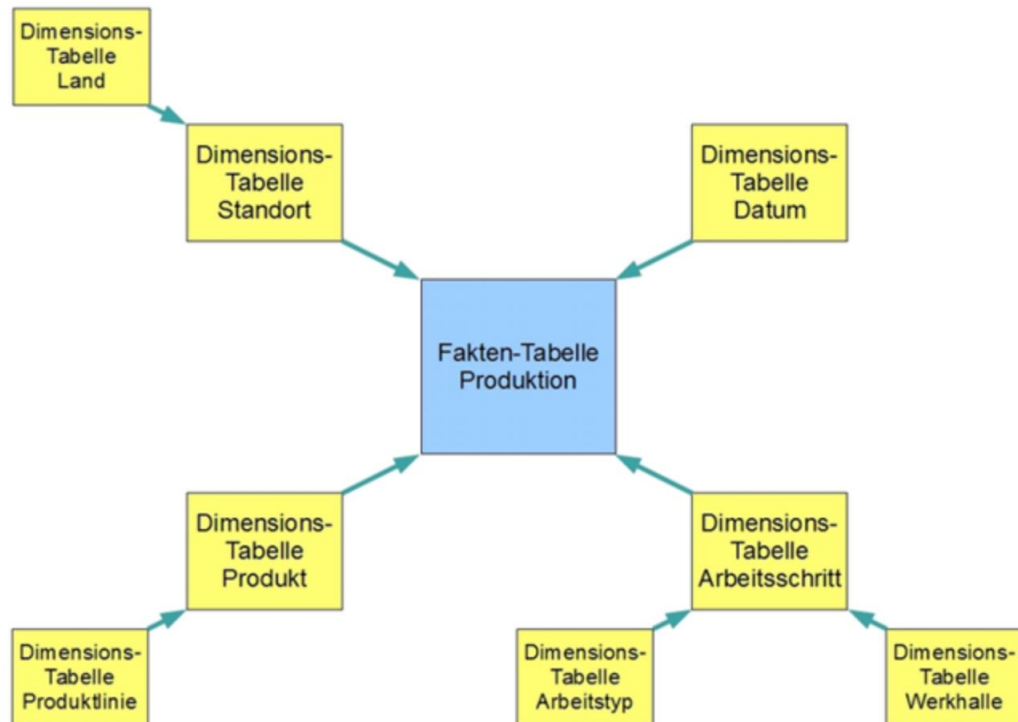
Star-Schema mit Levelattribut	Fact constellation
Aggregate vorberechnen und IN Fakttabelle speichern	Aggregate vorberechnen und in NEUE Fakttabellle speichern
DimTabelle: Level-Attribute einfügen	Keien Änderung der DimTabelle
<u>Nachteil</u> Zugriff auf aggr. Werte = Zugriff auf Detailwerte, da eine Faktabelle	<u>Vorteil</u> - Zugriff auf aggr. Werte schnell - keine LevelAttribute nötig
<u>Vorteil</u> einfaches Schema → einfache Abfrage von aggr. Werten und Detailwerten	<u>Nachteil</u> komplexes Schema → komplexere Abfrage von aggr. Werten und Detailwerten

Snow-Flake-Schema

- abgeleitet aus dem Star-Schema
- normalisiert die Dimensionstabellen
- in jeder Dimension wird für jede Hierarchieebene eine eigene Tabelle eingeführt
- Verbindungen zwischen Dim-Tabellen und Fakttabelle über Fremdschlüssel-Primärschlüssel-Beziehungen realisiert

Dimensionstabelle		Fakttabelle
Enthält: <ul style="list-style-type: none"> • Primärschlüssel für den Hierarchieknoten (z.B. P_Nr) • beschreibendes Attribut (z.B. Name) 		Enthält: <ul style="list-style-type: none"> • Fremdschlüssel der jeweils niedrigsten Hierarchiestufe der Dimensionen (z.B. P_Nr) • Primärschlüssel als zusammengesetzten Schlüssel,

<ul style="list-style-type: none"> Fremdschlüssel der nächst höheren Hierarchieebene (z.B. K_Nr) 		bestehend aus den Fremdschlüsseln der niedrigsten Hierarchiestufen der Dimensionen (z.B. P_Nr, F_Nr, M_Nr)
---	--	--



<u>Vorteile</u>	<u>Nachteile</u>
<ul style="list-style-type: none"> Geringer Speicherplatzverbrauch (Dimensionstabellen enthalten durch Normalisierung keine Redundanzen) N:M Beziehungen zwischen den Aggregationsstufen können über Relationstabellen aufgelöst werden Browsing-Funktionalität: häufige Abfragen über sehr große Dimensionstabellen erbringen Zeitersparnis und Geschwindigkeitsvorteile 	<ul style="list-style-type: none"> Geschwindigkeitsnachteil: durch zusätzliche Verbunde der Dimensionstabellen Große Tabellenanzahl durch komplexe Strukturierung Reorganisationsproblem: Änderungen im semantischen Modell führen zu umfangreicher Reorganisation

Bewegen im Cube

Slice → Filtern im Cube „Scheibe herausschneiden“

Rollup → Aggregieren von Detailwerten

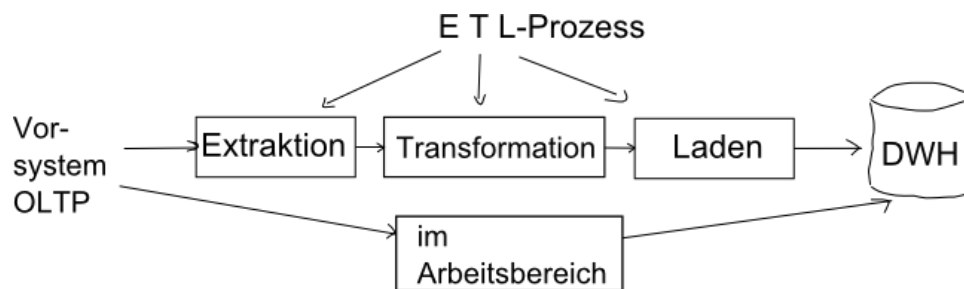
Drill Down → Detaillieren von Detailwerten

Drill Across → Zellinhalte wechseln

Dice → Darstellungsbereich ändern *Ergebnis: kleinerer Cube*

Pivoting → Achsen vertauschen

ETL-Prozess



Extraktion: → holen von Daten aus der Datenquelle

- macht Änderungen in den Quelldaten
- möglich über:
 - Trigger
 - Replikationen
 - Log - basierend
 - Zeitstempel - basierend
 - SnapShot – basierend

Transformation:

<i>Überführung in die DWH - Struktur</i>	
inhaltlich (Instanzintegration) <ul style="list-style-type: none">• bereinigen• harmonisieren• Verdichtung• Anreicherung	strukturtechnisch <ul style="list-style-type: none">• Schemaintegration• → nicht verträgliche Datentypen
<i>Bereinigung</i>	
automatisch	manuell (Falsch- bzw. Fehlereingaben)

Harmonisierung:

- Kodierung: bsp. Geschlecht
- Synonyme: unterschiedliche Attributs Namen → gleiche Bedeutung
- Homonyme: gleiche Attributs Namen → unterschiedliche Bedeutung

Verdichtung

Anreicherung: → betriebswirtschaftliche Kennzahlen bilden – Berechnungen

Laden:

Aufgabe:

- Übertragen der bereinigten und aufbereiteten (z.B. aggregierten) Daten in das Data Warehouse

Besonderheiten:

- i.A. Verwendung spezieller Ladewerkzeuge (z.B. SQL*Loader von Oracle)
- Anwendung von Bulk-Laden
- Historisierung: kein Überschreiben von Daten im DWH bei Änderungen in den Quelldaten, sondern zusätzliches Abspeichern

Ladevorgang

- online: Quelldatenbank und DWH stehen weiterhin zur Verfügung
- offline: Quelldatenbank und DWH stehen nicht zur Verfügung (i.A. Verwendung von Zeitfenstern mit Schwachlast, z.B. nachts oder an Wochenenden)

Data Mining

Ist die Anwendung von Methoden und Algorithmen zur möglichst automatischen Extraktion empirischer Zusammenhänge zwischen Planungsobjekten, deren Daten in einer hierfür aufgebauten Datenbasis bereitgestellt werden.

→ Anwendung effizienter Algorithmen, die in einer Datenbank, einem Data Warehouse enthaltenen Muster liefern.

Anwendungen:

- Warenkorbanalyse – im Handel
- Bewertung Kreditwürdigkeit – Banken
- Analyse von Textinhalten – alle Branchen
- Bewertung Werbewirksamkeit – alle Branchen

Bestandteile:

- Statistik
- Maschinelles Lernen
- Datenbank
- Softcomputing (FUZZY)

Klassen

<u>Klasse</u>	<u>Gegenstand (Aufgabe)</u>	<u>Anwendung-bsp.</u>	<u>Methoden-bsp.</u>
Klassifikation	Individuen bekannten Klassen zuordnen	Bonitätsprüfung	<ul style="list-style-type: none">• Diskriminanzanalyse• Entscheidungsbaum/ Entscheidungsregeln
Clustering	Gruppen auf Basis von Ähnlichkeiten bilden	Ermittlung von Kundengruppen	Clusterverfahren
Vorhersage	Zukünftige Werte berechnen	Aktienkursprognose Strompreisprognose	Regression ARIMAX – Verfahren (ARIMA-Gruppe) KNN (künstliche neuronale Netze)
Assoziation	Abhängigkeiten bestimmen	Warenkorbanalyse	Assoziationsregeln
Text Mining	Textmuster suchen	Information retrieval	KNN Word2Vec(google) ...

Ziel der Klassifikationsregelgenerierung:

- Redundanzarme, vollständige, widerspruchsfreie und effiziente Menge an Klassifikationsregeln erzeugen.

Datenkategorien

Kategorial - kategorisch (auflistend, diskret)		Numerisch - kontinuierlich	
Nominal (auflistend)	Ordinal (sortierend)	Intervall (Abstand)	Ratio (Verhältnis)
<ul style="list-style-type: none"> - vordefinierte endlicher Wertebereich - Wert ist Beschriftung - keine Relationen zwischen den Werten - keine mathem. Operationen - keine Sortierung und Abstand - Prüfung auf Gleichheit <p>Beispiele:</p> <ul style="list-style-type: none"> - Aussicht: sonnig, bewölkt, regnerisch - wenn Ansicht == sonnig, dann ... 	<ul style="list-style-type: none"> - endlicher Wertebereich - Ausprägung sind Namen - Sortierung Sinnvoll - Kein Abstand sinnvoll - Prüfung: = > < <p>Beispiele:</p> <ul style="list-style-type: none"> - Temperatur: heiß, mild, kalt - heiß > mild > kalt 	<ul style="list-style-type: none"> - unendlicher Wertebereich - ausprägungen sind Zahlen - Differenzbildung möglich - Summe nicht sinnvoll <p>Beispiele:</p> <ul style="list-style-type: none"> - Datum: 2010-2007 = 3; 2010+2007=nicht sinnvoll 	<ul style="list-style-type: none"> - Unendlicher Wertebereich - Meßverfahren def. zusätzlich den Nullpunkt - Alle math. Operationen erlaubt <p>Beispiel:</p> <p>Alter einer Person oder Abstand zweier Objekte</p>

Bsp:

ID	Alter	Autotyp	Risikoklasse
1	23	Familie	hoch
2	17	Sport	hoch
3	43	Sport	hoch
4	68	Familie	niedrig
5	32	Lkw	niedrig

**Ratio
Attribut**

**nominales
Attribut**

**ordinales
Attribut**

Prediktorvariablen

= unabhängige Variablen:
d.h. Variablen, die die Werte der
Zielvariablen vorhersagen

Zielvariable

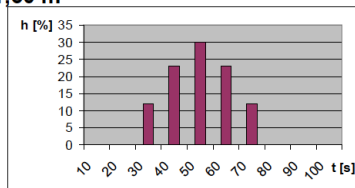
= abhängige Variable:
d.h. Variable, deren Wert mittels
Prediktorvariablen vorhersagt wird

Statistische Unabhängigkeit / Abhängigkeit

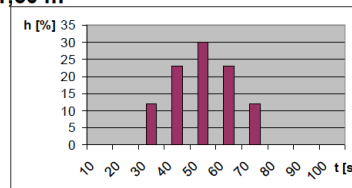
Statistische Unabhängigkeit	Statistische Abhängigkeit
Merkmale variieren in Bezug auf eine Zielvariable	
nicht gemeinsam	gemeinsam
<ul style="list-style-type: none"> gleiches Zentrum der Verteilung und gleicher Verlauf der Verteilung 	<ul style="list-style-type: none"> unterschiedliches Zentrum der Verteilung und/oder unterschiedlicher Verlauf der Verteilung

Beispiel: Abhängigkeit der Reaktionszeit t von der Körpergröße g bei Testpersonen

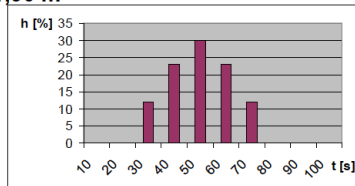
$g = 1,60 \text{ m}$



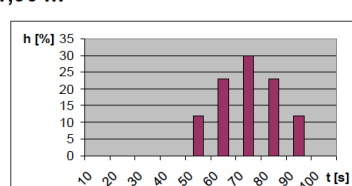
$g = 1,60 \text{ m}$



$g = 1,90 \text{ m}$



$g = 1,90 \text{ m}$



keine gemeinsame Variation



statistische Unabhängigkeit

gemeinsame Variation



statistische Abhängigkeit

Overfitting

- Überanpassung eines Modells (z.B. Entscheidungsbaum) an die zu lernenden Datensätze

Folge:

- Modell bildet gelernte Daten sehr genau ab
- Bei Anwendung des Modells auf unbekannte Datensätzen treten große Fehler
- → fehlerhafte Klassifikationen

Lösungswege:

- aufteilen der Gesamtdatenmenge in
 - Lerndatenmenge → Modellerstellung
 - Testdatenmenge → Modellvalidierung
 - Modellvalidierung mit Kennzahlen
- Pruning

		Realität (reale Ergebnisse in der Testdatenmenge)	
		<i>positiv</i>	<i>negativ</i>
Ergebnis der Modellanwendung auf die Testdatenmenge	<i>positiv</i>	richtig Positive <i>rp</i>	falsch Positive <i>fp</i>
	<i>negativ</i>	falsch Negative <i>fn</i>	richtig Negative <i>rn</i>

Pruning

- Vermeiden bzw. verringern von Overfitting eines Entscheidungsbaums durch kürzen („zurückschneiden“).

Verfahren:

- Begrenzen des Baumaufbaus durch Stoppkriterien(**prepruning**):
 - Vorgabe der maximalen Anzahl an Ebenen im Baum
 - Vorgabe einer Mindestanzahl an Elementen in einem Knoten
- Reduzierung der Komplexität des Baumes
 - Einsatz von Knoten oder Teilbäume durch Blätter (**postpruning**)
- **Bagging**
- **Boosting**
- **Stacking**

Größe der Testdatenmenge

- zu klein: Fehler in der Testdatenmenge nicht signifikant
- zu groß: Lerndatenmenge zu klein (ggf. fehlen wichtige Daten darin)
- verschiedene Wege zur Bestimmung einer möglichst optimalen Aufteilung,

Bagging → Bootstrap Aggregation

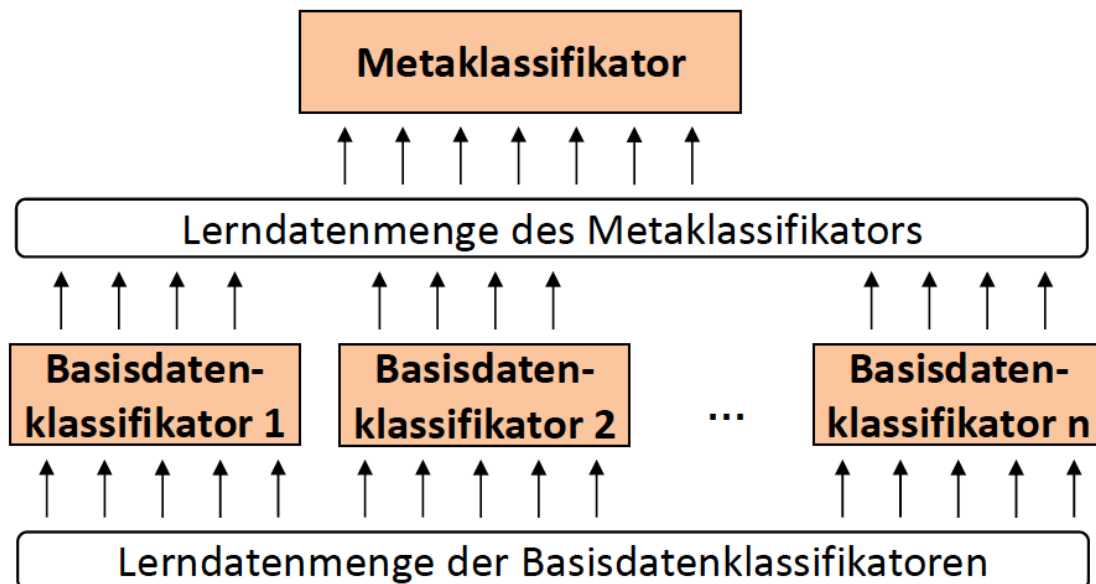
- Kombination von N Modellen zu einem besseren Gesamtmodell(Metalerner):
 - Ziehen von N Bootstrap-Sampeln als Lerndatensätze
 - Erzeugen eines Modells je Bootstrap-Sample mit dessen Lerndatensätzen
- Klassifikation eines neuen (dem Gesamtmodell) unbekannten Datensatz:
 - Klassifikation des Datensatzes durch jedes einzelne Modell
 - Gesamtergebnis = Mehrheitsentscheidung über die Klassifikation der einzelnen als Modell

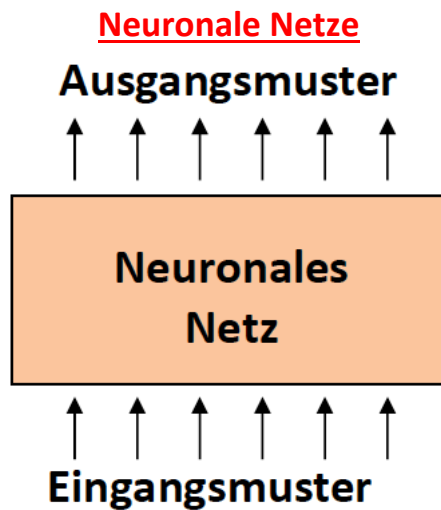
Boosting

- trainiert eine Folge von Modellen auf einer Stichprobe der Lerndatenmenge
- fehlerhaft klassifizierte Datensätze werden im späteren Modell bevorzugt

Stacking

- es werden mehrere unterschiedliche Modelle (Basisdatenklassifikatoren) auf auf den selben Daten trainiert
- Modelle haben verschiedene Stärken und Schwächen
- aus den Ergebnissen wird ein weiteres Modell trainiert (Metaklassifikator)
- Metaklassifikator sucht den besten Basisklassifikator für eine Entscheidung heraus





Funktionsweise

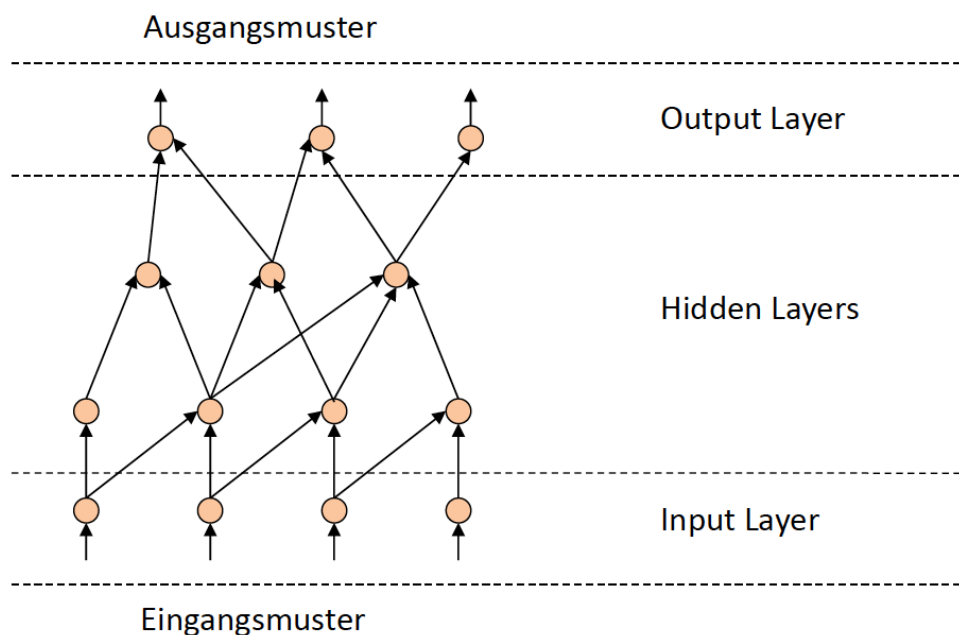
- Eingangsmuster wird mit Hilfe der vernetzten Verarbeitungselemente des neuronalen Netzes verarbeitet und erzeugt daraus ein entsprechendes Ausgangsmuster
- Gewichtete Summierung mit nachfolgender Nichtlinearität

Lernphase → erhält entsprechende Ein-/Ausgangsmusterpaare und entwickelt daraus entsprechende Netzparameter

Gebrauchsphase → neue Eingabemuster werden übergeben und das Netz erzeugt entsprechende Ausgangsmuster

Definition

- System zur Informationsverarbeitung mit Hilfe von einfachen vernetzten Elementen
- hat gerichtete Ein- und Ausgaben



Assoziationsregeln

Eine **Assoziationsregel** ist eine Regel, die eine beliebige Kombination unterschiedlicher Attribut/Werte-Paare enthält.

- beliebige **Kombination** bedeutet → Verwendung von Attributen aus dem Bedingungs- und dem Entscheidungsteil
- Attribut/Werte-Paar wird auch als **Gegenstand (item)** bezeichnet

Abdeckung: Anzahl an Instanzen, die die Assoziationsregel korrekt vorhersagt

Genauigkeit: Verhältnis von der Abdeckung zur Anzahl der Instanzen, auf die die Regel angewendet wird.

Beispiel:

	Aussicht	Temperatur	Luftfeuchte	Wind	Spiel
1.	sonnig	heiß	hoch	nein	nein
2.	sonnig	heiß	hoch	ja	nein
3.	bewölkt	heiß	hoch	nein	ja
4.	regnerisch	mild	hoch	nein	ja
5.	regnerisch	kalt	normal	nein	ja
6.	regnerisch	kalt	normal	ja	nein
7.	bewölkt	kalt	normal	ja	ja
8.	sonnig	mild	hoch	nein	nein
9.	sonnig	kalt	normal	nein	ja
10.	regnerisch	mild	normal	nein	ja
11.	sonnig	mild	normal	ja	ja
12.	bewölkt	mild	hoch	ja	ja
13.	bewölkt	heiß	normal	nein	ja
14.	regnerisch	mild	hoch	ja	nein

Assoziationsregeln:

1. WENN Temperatur = kalt	DANN Luftfeuchte = normal
2. WENN Luftfeuchte = normal UND Wind = nein	DANN Spiel = ja
3. WENN Aussicht = sonnig UND Spiel = nein	DANN Luftfeuchte = hoch
4. WENN Wind = nein UND Spiel = nein	DANN Aussicht = sonnig UND Luftfeuchtigkeit = hoch

Abdeckung: I (Kaffee, Milch)=3/6=50%
II (Kaffee, Milch, Kuchen)=2/6=33%

Genauigkeit Kaffee, Milch →Kuchen=II/I=33%/50%=67%

Abdeckung von allen Werten bilden 1-items set

Kombination

1. Betrachtung von Itemsets, die eine Mindestabdeckung besitzen
Bildung 1-Itemsets→2-Itemsets ... (Abstand dazwischen ist Mindestabdeckung)
2. Umwandlung der Sets in Regeln, die eine Mindestgenauigkeit aufweisen

Berechnen:

item set I1: Luftfeuchte = normal UND Wind = nein

Abdeckung (I1) = 4 (DS: 5, 9, 10, 13)

rule R1: Luftfeuchte = normal UND Wind = nein => Spiel = ja

Abdeckung (R1) = 4 (DS: 5, 9, 10, 13)

Genauigkeit(R1) = Abdeckung (R1) / Abdeckung (I1)
= 4/4 = 100 %







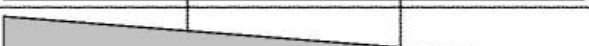
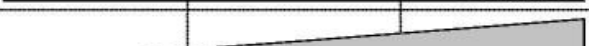
item set I2: Luftfeuchte = normal UND Spiel = ja

Abdeckung (I2) = 6 (DS: 5, 7, 9, 10, 11, 13)

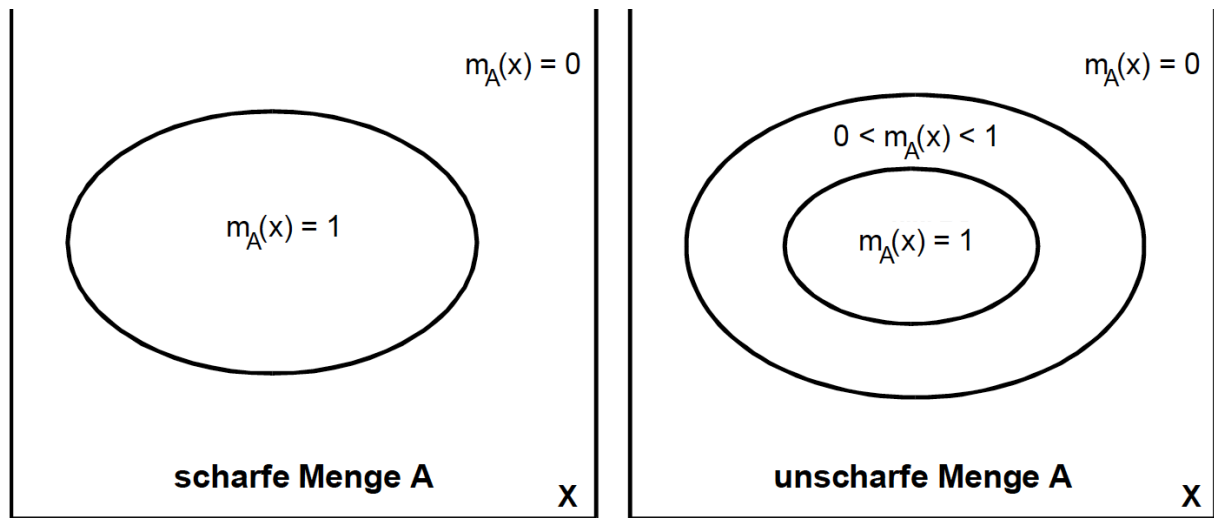
rule R2: Luftfeuchte = normal UND Spiel = ja => Wind = nein

Abdeckung (R2) = 4 (DS: 5, 9, 10, 13)

Genauigkeit(R2) = Abdeckung (R2) / Abdeckung (I2)
= 4/6 = 67 %

	Lower- Management	Middle- Management	Top- Management
Verwendung externer Daten			
Verwendung interner Daten			
Verknüpfung von Variablen			
Detaillierung			
Erforderliche Zugriffszeit			
Häufigkeit des Zugriffs			
Einsatz von analytischen Verfahren			
Einsatz von heuristischen Verfahren (z.B. Simulation)			

Scharfe und unscharfe Mengen



Schritte des Chaid Algorithmus:

Idee:

- Signifikanz eines statistischen Tests nutzen
- Werte der Prädiktorvariable ähnlich (1. Schritt)

Ja: zusammen führen Nein: erhalten

- Auswahl der Prädiktorvariable des nächsten Knotens (2. Schritt)

Merkmal ist intervallskaliert: Korrelation

Merkmal ist ordinalskaliert: Assoziation

Merkmal ist nominalskaliert: Kontingenz

Beispiel:

Merkmale: „Abschalten“ während der Vorlesung nach Geschlecht

Merkmal: Abschalten: stimmt /stimmt nicht

 Geschlecht: männl. / weibl.

1. Schritt: Preparing Predictors (kontinuierliche Werte -> kategoriale Werte)
(Klassenbildung)
2. Schritt: Merging categories (Werte zusammenfassen; Klassifikation)
3. Schritt: Selecting the Splitvariable

Schritt 2 & 3 wiederholen sich je Ast

Arten von IS

- Transaktionssysteme (OLTP) – *Leistungsschicht*
- Büroinformationssysteme (OIS) – *Adminschicht*
- Abfrage und Berichtssystem (QRS) – *Managementschicht*
- Managementunterstützungssysteme (MSS) – *Managementschicht*
- Managementinformationssysteme (MIS) – *Managementschicht*
- Executive Information Systeme (EIS) - *Managementschicht*

ROLAP-Abfragen (SQL)

STAR-JOIN

SELECT	➤ Attribute der Dimensionen ➤ Kennzahlen [aggregiert]	
FROM	➤ Dimensionstabellen ➤ Fakttable	
JOIN ON	➤ JOIN-Bedingungen	
WHERE	➤ Explizite Bedingungen ➤ [Level = Wert]	[] ... Schemavariante ohne level-Attribut
[GROUP BY]	➤ Kenngrößen	[] ... Schemavariante mit level-Attribut

Beispiel: Ermittlung der 2015 im Land „Deutschland“ verkauften Produkte mit Namen „Radeberger“

Dimensionen: Zeit: Jahr, Quartal, Monat
 Produkt: Name, Kategorie
 Geografie: Land, Region, Staat
Kennzahlen: Mengenumsatz, Wertumsatz

```
SELECT Jahr, Land, Sum(Mengenumsatz) FROM Verkauf V
  JOIN Produkt P on V.P_Nr = P.P_Nr
  JOIN Geografie G on V.G_Nr = G.G_Nr
  JOIN Zeit Z on V.Z_Nr = Z.Z_Nr
Where Land = 'Deutschland'
  And Jahr = 2015
  And Name = Radeberger
Group by Land, Jahr
```

Group by – Erweiterungen:

Gruppierung mit Rollup

```
SELECT Z.Jahr, Z.Q_ID, SUM(U.Umsatzbetrag) AS Umsatzbetrag
FROM   Umsatzdaten U
JOIN   Zeit      Z   ON  U.Mon_ID = Z.Mon_ID
GROUP BY Z.Jahr, Z.Q_ID
WITH ROLLUP
```



Jahr	Q_ID	Umsatzbetrag
2013	201301	975509,41
2013	201302	1049897,62
2013	201303	1125466,82
2013	201304	1047118,45
2013	NULL	4197992,30
2014	201401	1156138,95
2014	201402	1314491,31
2014	201403	1218761,16
2014	201404	1238545,65
2014	NULL	4927937,07
NULL	NULL	9125929,37

Gruppierung mit CUBE

```
SELECT Z.Jahr, G.Staat, SUM(U.Umsatzbetrag) AS Umsatzbetrag
FROM   Umsatzdaten U
JOIN   Zeit      Z   ON  U.Mon_ID = Z.Mon_ID
JOIN   Geografie G   ON  U.Land_ID = G.Land_ID
GROUP BY Z.Jahr, G.Staat
WITH CUBE
```



Jahr	Staat	Umsatzbetrag
2013	Deutschland	1823634,11
2014	Deutschland	2134530,02
NULL	Deutschland	3958164,13
2013	Österreich	564187,80
2014	Österreich	662805,93
NULL	Österreich	1226993,73
2013	Schweiz	1810170,39
2014	Schweiz	2130601,12
NULL	Schweiz	3940771,51
NULL	NULL	9125929,37
2013	NULL	4197992,30
2014	NULL	4927937,07

Kreuztabellen (Pivot)

- Über das „Group By“ wird normal gruppiert
- In einer zweiten Ebene eine Gruppierung durchgeführt, unabhängig vom ersten „Group By“ Attribut
- Werte des PIVOT-Attributs werden neue Spalten in der Ereignisrelation

Gruppierungsabfrage			Kreuztabellenabfrage					
ArtikelNr	Verkaufsgebiet	Verkaufte Einheiten	ArtikelNr	Gesamtsumme	Nord	Ost	Süd	West
ALG-001	Ost	150	ALG-001	150		150		
ALG-002	Nord	53	ALG-002	203	53	150		
ALG-002	Ost	150	ALG-003	80	20		30	30
ALG-003	Nord	20	ALG-004	110			30	80
ALG-003	Süd	30	ALG-005	50	40	10		
ALG-003	West	30	ALG-006	243	200	43		
ALG-004	Süd	30	ALG-007	5		5		
ALG-004	West	80	EDV-001	55		25	15	15
ALG-005	Nord	40	EDV-002	78	3	50	25	
ALG-005	Ost	10	EDV-003	55	40	10	5	
ALG-006	Nord	200	EDV-004	52	17		15	20

OLAP-Operationen im Front-End

- Pivoting: Drehen eines Würfels in eine andere Achse
- Roll-UP – eine Hierarchie-Ebene höher
- Drill-Down – eine Hierarchie-Ebene niedriger

	2015	2016	2	Zeit		2015
Nord	12116		Drill down, Roll up		Hamburg	5550
West	11814				Hannover	2890
Mitte	10414				Bremen	3676

- Slice – eine „Scheibe“ eines Würfels, in der Tiefe Filtern, Filter auf nicht-Anzeige-Achse → Ergebnis ist eine zweidimensionale Matrix

3 Produkte			2 Zeit		Schicht Elektrogeräte
Audio/Video	Computer	Elektrogeräte	2015	2016	
1 Region	Hamburg				
	Hannover				
	Bremen				

- Dice – kleiner mehrdimensionaler Ausschnitt des Cubes → neuer mehrdimensionaler Datenraum, der wiederum extrahiert und weiter verarbeitet werden kann

3 Produkte			2 Zeit		Schicht Elektrogeräte
Audio/Video	Computer	Elektro	2015	2016	
1 Region	Nord		12116		
	West		11814	...	
	Mitte		10414		
	Ost		10850		

- Visualize
- Drill-trough: Einzelwerte anzeigen

Elemente des Cubes aus Abfrage Sicht

- Dimension
 - Produkt besitzt Attribute
 - Name, Preis, Subkategorie, Kategorie, Lieferant)
 - Hierarchie Sortiment
 - Level: Kategorie→Subkategorie→Name

MDX

- Multidimensional Expression
- Von MS entwickelt für OLAP-Datenbanken
- Mittlerweile Industriestandard
- Relativ komplex → für IT-Entwickler bzw. Abfragesprache für Applikationen, nicht für Endanwender

Abgrenzung

MDX	SQL
Abfragesprache für Datenbanken	
Microsoft	ANSI und ISO Standard
Abfrageschema basiert auf SELECT, FROM, WHERE	
Basis ist eine multidimensionale OLAP Datenbank (CUBE)	Basis ist eine relationale Datenbank
Versteht Hierarchien, Vorgänger / Nachfolger, Cousin, ... und kann Eigenschaften von Elementen, Zellen auslesen und definieren	
2 – n dimensionales Ergebnis, also Tabelle oder Cube	2 dimensionale Ergebnis, also Tabelle
Ähnliche Basisoperatoren und -funktionen	

Abfrageschema

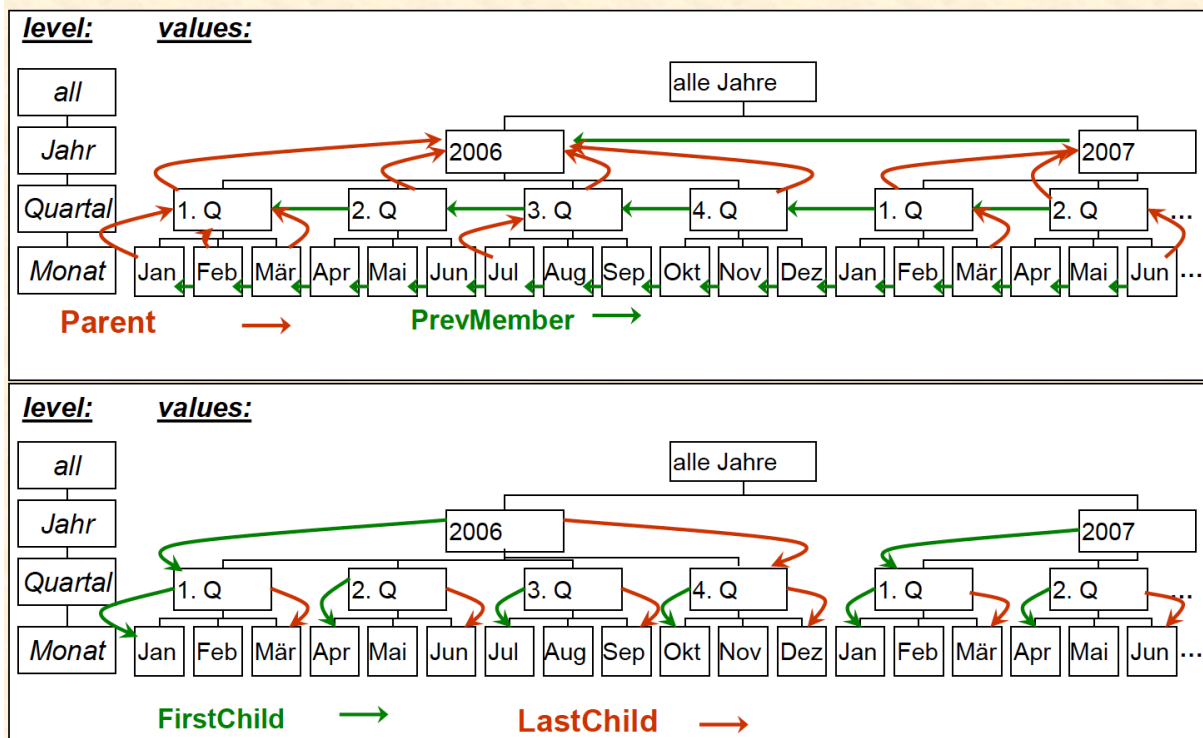
- Aufbau einer MDX – Abfrage

```
SELECT  
    <Abfrageachse> ON COLUMNS,  
    <Abfrageachse> ON ROWS  
FROM <Cube>  
WHERE <Slicerachse>
```

- <Abfrageachse> → Menge aus denen die Daten abgerufen werden
- <Cube> → Cube(s) der (die) abgefragt werden sollen
- <Slicerachse> → Menge oder Tupel auf die die Ergebnismenge eingeschränkt wird

Parent und PrevMember - Beispiel

Parent und PrevMember am Beispiel der Dimension Zeit



Beispiel MDX – Abfragen

(1.) Zeigen Sie Umsatzbetrag und Umsatzmenge der Bundesländer Sachsen und Thüringen an.

*Select {[Measures].[Umsatzbetrag],[Measures].[Umsatzmenge]} on columns,
{{[Geografie].[Bundesland].&[01],[Geografie].[Bundesland].&[15]} on rows
from [Umsatz]*

(2.) Erweitern Sie die Abfrage (1), so dass nur Umsatzbetrag und Umsatzmenge des Jahres 2018 angezeigt werden.

*Select {[Measures].[Umsatzbetrag],[Measures].[Umsatzmenge]} on columns,
{{[Geografie].[Bundesland].&[01],[Geografie].[Bundesland].&[15]} on rows
from [Umsatz]
Where ([Zeit].[Jahr].&[2018])*

(3.) Zeigen Sie die Umsatzbeträge für alle Produktkategorien in den Bundesländern Sachsen und Thüringen für das Jahr 2018 an.

*Select {[Produkt].[Kategorie].AllMembers} on columns,
{{[Geografie].[Bundesland].&[01],[Geografie].[Bundesland].&[15]} on rows
from [Umsatz]
Where ([Zeit].[Jahr].&[2018],[Measures].[Umsatzbetrag])*

(4.) Zeigen Sie die Umsatzbeträge für alle Subkategorien der Produktkategorie Backwaren in den Bundesländern Sachsen und Thüringen für das Jahr 2018 an.

*Select {[Produkt].[Backwaren].Children} on columns,
{{[Geografie].[Bundesland].&[01],[Geografie].[Bundesland].&[15]} on rows
from [Umsatz]
Where ([Zeit].[Jahr].&[2018],[Measures].[Umsatzbetrag])*

(5.) Zeigen Sie die Umsatzbeträge für alle Produktkategorien und alle Staaten für das Jahr 2018 an.

*Select {[Produkt].[Kategorie].Members} on columns,
{{[Geografie].[Staat].Members} on rows
from [Umsatz]
Where ([Zeit].[Jahr].&[2018],[Measures].[Umsatzbetrag])*

(6.) Zeigen Sie die Umsatzbeträge und Umsatzmengen für alle Produktkategorien und alle Staaten für das Jahr 2018 an.

```
Select {[Produkt].[Kategorie].children}*{[Measures].[Umsatzbetrag],[Measures].[Umsatzmenge]} on columns,  
      {[Geografie].[Staat].children} on rows  
From [Umsatz]  
Where ([Zeit].[Jahr].&[2018])
```

(7.) Zeigen Sie den Umsatzbetrag für alle Produktkategorien im Jahr, im Quartal und im Monat an.

```
Select {[Produkt].[Kategorie].Members} on columns,  
      {[Zeit].[Jahr].children}*{[Zeit].[Quartal].children}*{[Zeit].[Monat].children} on rows  
From [Umsatz]  
Where ([Measures].[Umsatzbetrag])
```

(8.) Ändern Sie die Abfrage (7) so ab, dass nun Umsatzbetrag und Umsatzmenge für alle Produktkategorien und die Quartale und Monate des Jahres 2018 angezeigt werden.

```
Select  
{[Measures].[Umsatzbetrag],[Measures].[Umsatzmenge]}*{[Produkt].[Kategorie].children}  
on columns,  
  {[Zeit].[Quartal].children}*{[Zeit].[Monat].children} on rows  
From [Umsatz]  
Where ([Zeit].[Jahr].&[2018])
```

(9.) Wie groß ist die Differenz zwischen Plan- und Ist-Umsatz für die Produktsubkategorien in den Jahren 2017, 2018 und insgesamt? *Hinweis: Verwenden Sie zur Lösung WITH MEMBER und weisen Sie neben der Differenz den Umsatzbeitrag und den Planumsatz aus.*

```
with member [Measures].[Differenz] as  
  [Measures].[Umsatzbetrag]-[Measures].[Umsatzplan]  
Select {[Measures].[Umsatzbetrag],[Measures].[Umsatzplan],[Measures].[Differenz]} on  
columns,  
      {[Produkt].[SubKategorie].children}*{[Zeit].[Jahr].Members} on rows  
from [Umsatz]
```

Unscharfes schließen

Einstellungskriterium:

- (Ausbildung ODER Erfahrung) UND (Selbstständigkeit ODER Teamarbeit) UND Alter
- Kandidaten

Fuzzy Entscheidungsfindung

UND: Minimum

ODER: Maximum

Beispiel Mitarbeiterauswahl

Kriterium (Ausbildung ODER Erfahrung) UND (Selbstständigkeit ODER Teamfähigkeit) UND Alter

Zugehörigkeit	Kandidaten				
	1	2	3	4	5
1. M-Alter	1	0,5	0,7	0,1	0,6
M-Ausbildung	0,2	0,8	0,5	0,8	0,6
M-Erfahrung	0,3	0,2	0,9	1	0,6
M-Selbstständigkeit	0,6	0,4	0,7	1	0,5
M-Teamfähigkeit	0,4	0,5	0,2	1	0,8
1. M-Ausb. ODER M-Erf.	0,3	0,8	0,9	1	0,6
2. M-Selbst. ODER M-Team	0,6	0,5	0,7	1	0,8
M-Krit = M1 und M2 und M3	0,3	0,5	0,7	0,1	0,6

ODER → MAX ; UND = Min(m1,m2,m3) ; optimaler Kandidat → **Kandidat 3** nach den Kriterien

Gegenüberstellung OLTP – DWH

Kriterium	OLTP-System	DWH-System
Anfragearten	Lesen, Schreiben, Ändern, Löschen	Lesen, periodisches Hinzufügen
Transaktionsdauer und typ	kurze Lese- und Schreibtransaktionen	lange Lesetransaktionen
Anfragestruktur	einfach strukturiert	komplex
Datenvolumen je Anfrage	wenige Datensätze	viele Datensätze
Datenmodell	anfragebezogen	analysebezogen
Datenquelle	meist eine	mehrere
Eigenschaften der Daten	nicht abgeleitet, zeitpunkt-bezogen, autonom, dynamisch	abgeleitet, konsolidiert, zeitraum-bezogen, integriert, stabil
Datenvolumen	MByte ... GByte	GByte ... TByte
Zugriffsart	Einzeltuplelzugriff	Tabellenzugriff
Anwendertyp	Ein- und Ausgabe durch Angestellte oder Anwendungssoftware	Manager, Controller, Analyst
Anwenderzahl	sehr viele	wenige (bis einige hundert)
Antwortzeit	ms ... sec	sec ... min

ROLAP - MOLAP

	ROLAP	MOLAP
Bedeutung	Relationales-OLAP	Multidimensionales-OLAP
Datenspeicherung	Daten liegen in relationalen Datenbanken vor.	Daten werden in multidimensionalen Datenbanken als Datenwürfel gespeichert
Daten Form	Relationale Tabellen	Multidimensionale Arrays
Datenvolumen	Hohes Datenvolumen und hohe Nutzerzahl	Mittleres Datenvolumen, da Detaildaten in komprimiertem Format vorliegen
Technologie	Benötigt Komplexe SQL Abfragen, um Daten zu beziehen	Vorberechneter Datenwürfel hält Aggregationen vor
Skalierbarkeit	Beliebig	Eingeschränkt
Antwortgeschwindigkeit	Langsam	Schnell