Database and Information Systems

Data Warehousing and OLAP

- Decision support systems
- Data Warehouses
- Dimensional Modelling
- Online Analytical Processing
- Data Warehouse 2.0

Data Warehousing and OLAP

Readings:

Heuer, Andreas; Saake, Gunter: Datenbanken - Konzepte und Sprachen, 2nd edition, Thomson Int., 2000, Section 4.6, 10.2.3.

Conolly, Thomas; Begg, Carolyn: Database Systems - A Practical Approach to Design, Implementation, and Management, 3rd edition, Addison Wesley, 2002, Chapter 30-32.

Kifer, Michael; Bernstein, Arthur; Lewis Philip M.: Database Systems - An Application-Oriented Approach. 2nd edition. Pearson Education 2005, Chapter 15.

Dunham, Margaret H.: Data Mining - Introductory and Advanced Topics. Pearson Education, 2003, Chapter 2.

Data Warehousing and OLAP

Readings:

William H. Inmon; Derek Strauss; Genia Neushloss: DW 2.0 - The Architecture for the Next Generation of Data Warehousing. Morgang Kaufmann, Amsterdam etc., 2008.

Kris Krishnan: Data Warehousing in the Age of Big Data. Morgan Kaufmann, Amsterdam, 2013.

William H. Inmon; Anthony Nesavich: Tapping into Unstructured Data - Integrating Unstructured Data and Textual Analytics into Business Intelligence. Pearson, 2008.

Data Warehousing and OLAP

- Decision support systems
- Data Warehouses
- Dimensional Modelling
- Online Analytical Processing
- Data Warehouse 2.0

Decision Support Systems

- also: executive information systems, executive support systems
- purpose:
  assisting managers in making decisions and solving problems
- traditional databases vs. decision support systems?

Decision Support Systems

- traditional databases:
  - task specific collections of operational data
    - billing
    - inventory control
    - payroll
    - procurement
    - manufacturing support
  - typical services
    - online transaction processing
    - batch reporting

Decision Support Systems

- decision support systems:
  - informational data for
    - strategic analysis
    - planning
    - forecasting
  - typical services
    - ad hoc queries
    - customized information
  - data are usually organized along dimensions
  - data warehouse technology is useful but not necessary

Data Warehouses and OLAP

- Decision support systems
- Data Warehouses
- Dimensional Modelling
- Online Analytical Processing
- Data Warehouse 2.0

Data Warehousing

- Why data warehousing? - an alternative view (Sahuguet 1997)
  - DB vendors have to create new needs
  - Consultants have to create new concepts
  - PhD students have to find fancy thesis subjects
  - faculty members have to find new funding
- but the idea of data warehousing is still alive

Data Warehouse

- set of data that supports decision support systems and is subject-oriented, integrated, time-variant, and non-volatile
- single repository for corporate-wide data
  - including historical ones
- notion traced back to William H. Inmom (1995)
  first used in the early 1980ies

Characteristics of a Data Warehouse

- traditional databases: operational data for the day-to-day needs
  - inventory control, payroll, manufacturing support
  - online transaction processing and batch reporting
- data warehouse: informational data supporting decision taking
  - strategic analysis, planning, forecasting

- operational data needs to be transformed into informational ones
- relevant information is precomputed in advance of queries

Characteristics of a Data Warehouse

- data warehousing is a (pro-)active approach

active	passive
anticipation of queries	waiting for queries
"eager"	"lazy"
in advance	on demand

- data is completely static: only loaded, never updated

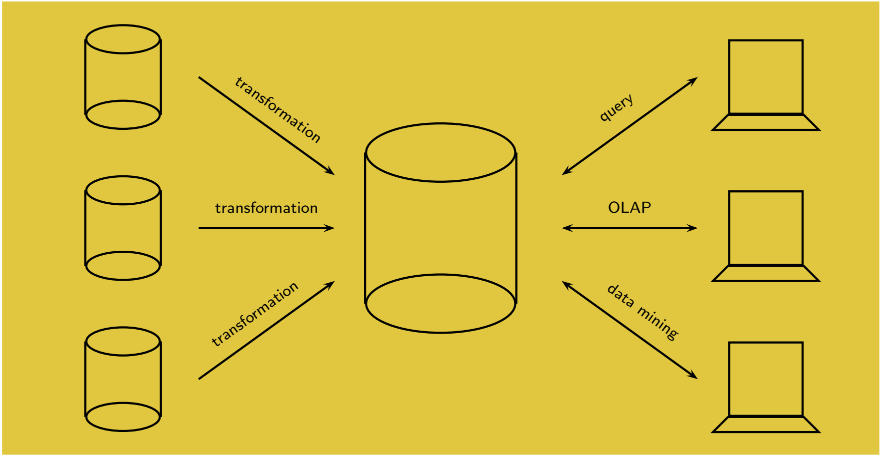
Characteristics of a Data Warehouse

	Operational Data	Data Warehouse
Application	OLTP	OLAP
Usage	Standard Workflow	ad hoc Queries
Temporal charact.	Snapshot	Historical
Modification	Dynamic	Static
Orientation	Application	Business Enterprise
Data	Operational Values	Aggregated Values
Level	Detailed	Summarized
Access	Frequently	Less Frequently
Response	Few Seconds	Minutes

Components of a Data Warehouse

- data migration tools
- the data warehouse
- access tools

Environment of a Data Warehouse



[Berson, Smith 1997]

Data Migration

- ETL: extract, transform, load
  - minimizing latency
- extraction
  - selection of relevant data
  - data profiling, estimation of data quality
  - periodicity: periodic, event-driven, query-driven
- transformation:
  - converting heterogeneous sources into one common schema
- loading
  - combination of snapshots into a historical data base

Data Migration

- syntactic transformations
  - reformatting (date, time, ...)
  - different data types

- semantic transformations
  - encoding conventions
    - code mapping (countries, gender, ...)
    - time zone mapping
    - units of measurement
  - schema mapping
  - harmonization of terminology

Data Migration

- semantic transformations (cont.)
  - inserting derived data
    - relative instead of absolute time information
      e.g. age $\rightarrow$ day of birth
  - cleansing
    - handling of missing and erroneous data
    - elimination of duplicates
  - summarization
    - aggregation of data

Problems with Data Warehousing

- problems in setting up a data warehouse (Greenfield 1996)
  - underestimation of resources for data loading
  - hidden problems with the source systems (e.g. missing data)
  - required data not captured
  - increased end-user demands
  - data homogenization (differences between different source systems are lost)
  - demanding resource requirements
  - conflicts between owners of data
  - high maintenance requirements
  - long-duration project
  - complexity of integration (different requirements, different tools, ...)

Performance Gains through Data Warehousing

- preaggregation
  - summarization during data transformation
  - 20 ... 100% increase in storage space $\rightarrow$ 2 .. 10 times speedup [Singh 1998]
- denormalization:
  - reduction of joins
  - redundancy can be tolerated
  - update anomalies are not a problem

Meta Data for Data Warehousing

- additional meta data requirements
  - origin of the data
  - changes made to the data during upload
  - aggregation procedures
  - table partitions and partion keys
  - profiling: typical queries for different users and user groups
  - user-group specific meanings of attributes and changes in meaning
  - synchronizing meta data between different systems and tools

Derived Concepts for Data Warehousing

- active data warehouse: allows online processing and data updates
- problems:
  - expensive to maintain data and transaction integrity
    both not really needed for data warehousing
  - peak period processing: used only infrequently, but needs substantial (mostly idle) backup capacity

Derived Concepts for Data Warehousing

- virtual data warehouse: implemented as a view on the operational data
- problems
  - poor performance: sharing resources with transaction processing
  - little support for historical data
  - non-replicable queries: data base mostly contains snapshot information
  - no possibility to combine data from different operational data bases

Derived Concepts for Data Warehousing

- federated data warehouse: cooperating operational data bases
- problems
  - poor performance: sharing resources with transaction processing
  - little support for historical data
  - non-replicable queries: data base mostly contains snapshot information
  - poor performance: reduced availability of components, no centralized optimization of queries to avoid duplicated effort
  - lack of data integration: data has to be used as it is available
  - complex technical infrastructure
  - fixed data granularity, inherited from the individual application data bases

Derived Concepts for Data Warehousing

- data mart: subset of a data warehouse at a departmental, regional, or functional level
- problems:
  - no reconciliation of data: different data marts will produce different answers to the same query
  - extract proliferation: every data mart requires specific data extraction components and procedures
  - change propagation: changes in the business model are likely to affect every data mart
  - difficult to extend: almost impossible to reuse existing components from one data mart for setting up another one

Data Warehousing and OLAP

- Decision support systems
- Data Warehouses
- Dimensional Modelling
- Online Analytical Processing
- Data Warehouse 2.0

Dimensional Modelling

- analysis-oriented way to represent and query data in a database
  - to be used in decision support systems
- special emphasis: efficient access to dimension-based data
- dimension: collection of logically related attributes
  - regions
  - time intervals
  - product classes
  - organisational hierarchies

  each viewed as an axis for modelling

Example

ProductID	LocationID	Date	Quantity	UnitPrice
176	London	2004-01-05	5	2900
352	Madrid	2004-01-07	9	5400
176	Prague	2004-01-12	3	2500
210	Manchester	2004-01-19	4	1500
176	Munich	2004-01-28	1	2800
176	Munich	2004-01-28	9	2700
317	Dresden	2004-02-04	3	4600
289	Milan	2004-02-06	100	990

Dimensional Modelling

- granularity:
  - unit of measurement, can vary depending on purpose
    - year, quarter, month, decade, week, day, hour, minute, second

  → granularity levels of a dimension
- changing the level of granularity:
  - roll up, drill down
- granularity problem:
  - selection of keys depends on the level of granularity
    c.f. `176/Munich/2004-01-28`

Dimensional Modelling

- target data:
  - usually numeric values for statistical purposes
    - organized along dimensions
    - need to be stored and queried on all levels
    - can be aggregated

  $\rightarrow$ facts

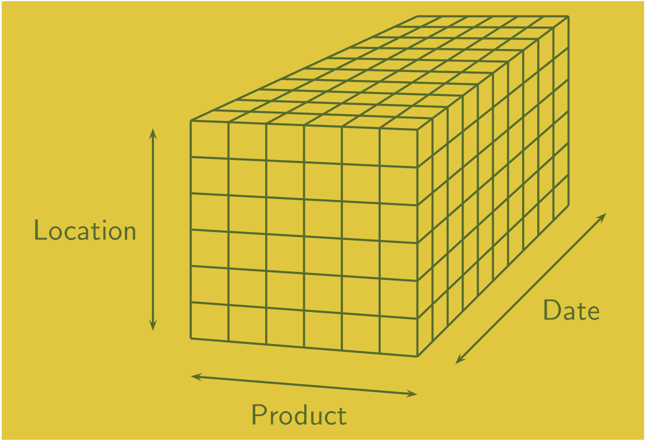
  $\rightarrow$ fact table

Example

ProductID	LocationID	Date	Quantity	UnitPrice
176	London	2004-01-05	5	2900
352	Madrid	2004-01-07	9	5400
176	Prague	2004-01-12	3	2500
210	Manchester	2004-01-19	4	1500
176	Munich	2004-01-28	1	2800
176	Munich	2004-01-28	9	2700
317	Dresden	2004-02-04	3	4600
289	Milan	2004-02-06	100	990

Dimensional Modelling

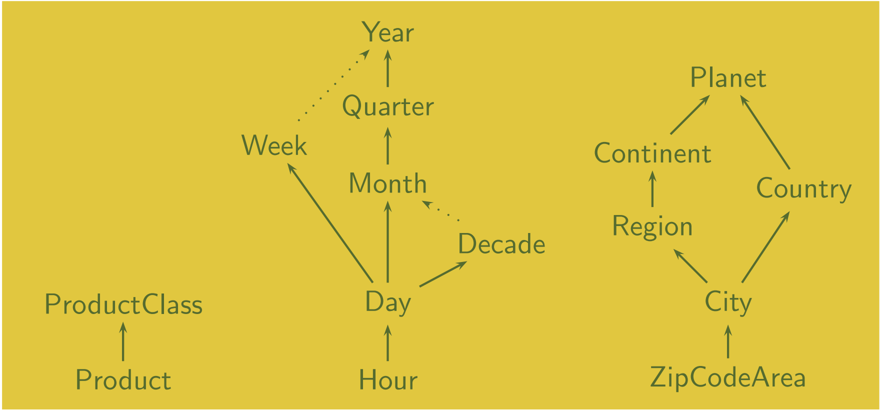
- the data cube



- fast access required
- but possibly extremely sparse

Dimensional Modelling

- dimensional hierarchy:
  - partial ordering of granularity levels according to an inclusion relationship ($<$)



Dimensional Modelling

- aggregation: If $X < Y$ then there is an aggregate type of relationship among the facts, e.g. additive

$$\text{quantity}(\text{product\_class}) = \sum_{\text{product}_i \in \text{product\_type}} \text{quantity}(\text{product}_i)$$

$$\text{quantity}(\text{month}) = \sum_{\text{day}_i \in \text{month}} \text{quantity}(\text{day}_i)$$

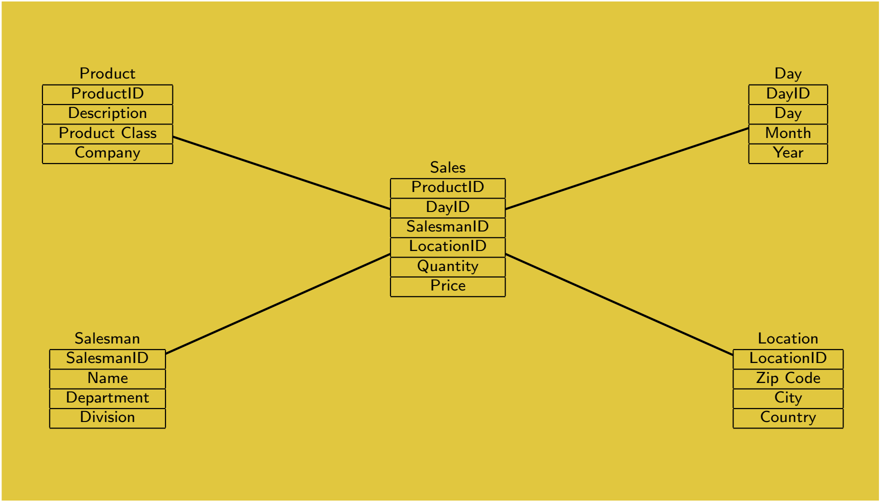
- other aggregate operations: average, maximum, minimum
- non-additive dimensions require a more complicated roll up/drill down

Dimensional Modelling

- DB schemas for multidimensional data
  - star schema
  - snowflake schema
  - fact constellation schema

- center: fact tables (major tables)
- periphery: dimension tables (minor tables)

Star Schema


Product
ProductID
Description
Product Class
Company
Day
DayID
Day
Month
Year
Sales
ProductID
DayID
SalesmanID
LocationID
Quantity
Price
Salesman
SalesmanID
Name
Department
Division
Location
LocationID
Zip Code
City
Country

Star Schema

- several fact tables are possible, dimension tables might point to other dimension tables
- fact table can be indexed, but amount of data is usually huge
- aggregation requirements must be supported efficiently

Star schema

- four storage models for dimension tables [Purdy/Brobst 1999]
  - flattened
  - normalized
  - expanded
  - levelized

Flattened Star Schema

- stores facts only at the lowest level of granularity
- key: all level attributes for the dimensions

```
sales(ProductID,DayID,SalesmanID,LocationID,
      Quantity,UnitPrice)
product(ProductID,Description,ProductClass,
      Company)
day(DayID,Day,Month,Year)
salesman(SalesmanID,Name,Department,Division)
location(LocationID,ZipCode,City,Country)
```

- roll up: sum aggregation
- problems:
  - time requirements
  - redundancies in the dimension tables

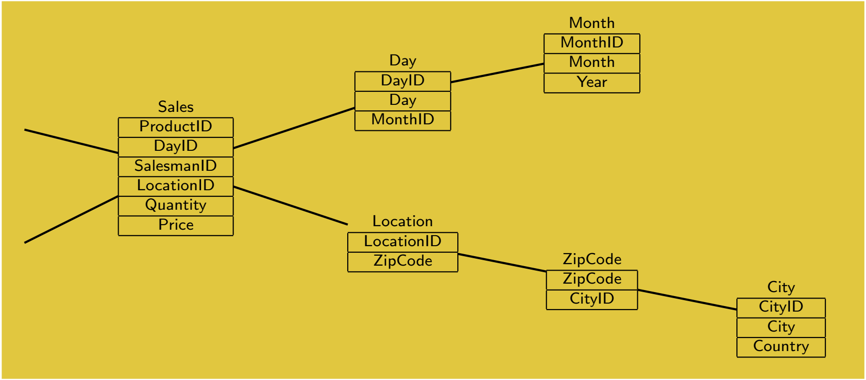
Normalized Star Schema

- dependencies resolved

```
sales(ProductID,DayID,SalesmanID,LocationID,...)
product(ProductID,Description,ProductClass,...)
day(DayID,Day,MonthID)
month(MonthID,Month,Year)
salesman(SalesmanID,Name,Department)
department(Department,Division)
location(LocationID,ZipCode)
zipcode(ZipCode,CityID)
city(CityID,City,Country)
```

  - duplication/redundancy is removed

Normalized Star Schema



- expensive access due to joins in the dimension tables
  $\rightarrow$ denormalization

Expanded Star Schema

- denormalization of the *dimension tables*
- stores dimensional data for all levels of granularity

```
sales(ProductID,DayID,SalesmanID,LocationID,...)
product(ProductID,Description,ProductClass,...)
day(DayID,Day,Month,Quarter,Year,MonthID)
month(MonthId,Month,Quarter,Year,QuarterID)
quarter(QuarterID,Quarter,Year)
salesman(SalesmanID,Department,Division)
department(Department,Division)
location(LocationID,ZipCode,City,Country)
zipcode(ZipCode,City,Country)
city(CityID,City,Country)
```

Expanded Star Schema

- even more space expensive than the flattened schema
- substantial amount of redundancy
  - $\rightarrow$ transformation from operational data!
- fast access
  - no join operations for the dimension tables

Levelized Star Schema

- denormalization of the *fact table*
- aggregation is precomputed for all granularity levels
- dimensional data also include a level indicator

```
sales(ProductID,TimeID,AgentID,LocationID,...)
product(ProductID,Description,ProductClass,...)
day(TimeID,Day,Month,Quarter,Year,LevelID)
salesman(AgentID,Agent,Dpmt,Division,LevelID)
location(LocationID,ZipCode,City,Country,LevelID)
```

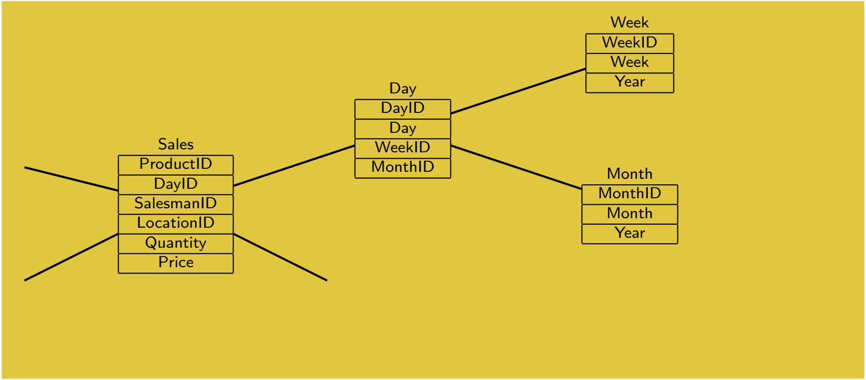
  - one tuple for each instance of each level in the dimension
  - massive redundancy
    - $\rightarrow$ transformation from operational data!
- fast access
  - no join operations for dimension access, no aggregation for roll up

Problems with the Star Schema

- brittleness in case of changing requirements
- limited extensibility
- geared towards specific user requirements
  $\rightarrow$ proliferation of specific schemata for different user groups

Snowflake Schema

- generalization of the normalized star schema
- aggregation hierarchy is directly represented in the DB schema



normalized star schema is a special case

Dimensional Modelling

- Indexing:
  - bitmap indices: each value in each domain is represented by a bit
    - $\rightarrow$ one bit vector per tuple
    - size of the vector: $\sum_i |Dom(A_i)|$
    - supports efficient join and aggregation through arithmetic operations
    - efficiency gains for attributes with few values
  - join indices: precomputation of tuples that join together
    - e.g. fact and dimension table
  - B-trees
    - more efficient if number of values is high

Data Warehousing and OLAP

- Decision support systems
- Dimensional Modelling
- Data Warehouses
- Online Analytical Processing
- Data Warehouse 2.0

OLAP

- OnLine Analytical Processing
- Codd 1993
- no clear definition
- mixture of goals and implementation issues

OLAP

- Codd's rules
  - multi-dimensional conceptual view
  - transparency
  - accessibility
  - consistent reporting performance
  - dynamic sparse matrix handling
  - multi-user support
  - unrestricted cross-dimensional operations
  - intuitive data manipulation
  - flexible reporting
  - unlimited dimensions and aggregation levels

OLAP

- OLAP council white paper
  - multidimensional view of data
  - calculation-intensive capabilities (related to aggregation functions)
  - time intelligence
- FASMI: Fast Analysis of Shared Multidimensional Information
- OLAP is an application view, not a data structure or a schema

OLAP Tools

- MOLAP: multidimensional OLAP
  - modelled, viewed and physically stored in a multidimensional database (MDD)
  - n-dimensional array
  - cube view is stored directly
    - + ad-hoc products (no SQL limitations)
    - + good mapping with data
    - + good performance for small cubes
    - – no standard (API may change over time)
    - – no common query language
    - – storage limitations

OLAP Tools

- ROLAP: relational OLAP
  - data stored in a relational database
  - ROLAP server creates the multidimensional view
    + support of RDBMS
    – relation has no inherent order, array has
    – virtual cube + meta data
    – time requirements (joins)
    – higher storage requirements (for fact table)

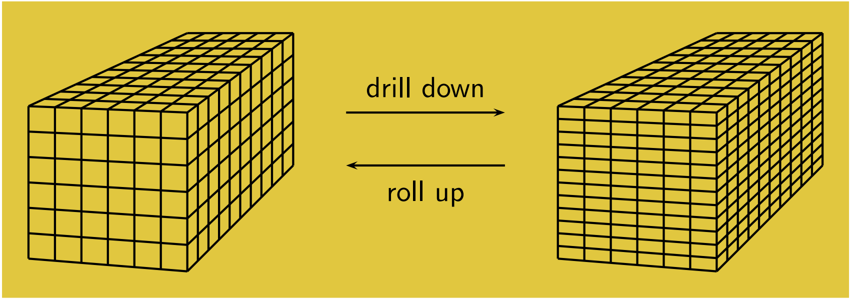
$$
\begin{aligned}
|\text{fact(MOLAP)}| &= |d_1| \cdot \ldots \cdot |d_n| \cdot |value| \\
|\text{fact(ROLAP)}| &= |d_1| \cdot \ldots \cdot |d_n| \cdot |[k_1, \ldots, k_n, value]| \\
&= (n+1) \cdot |\text{fact(MOLAP)}|
\end{aligned}
$$

OLAP

- HOLAP: hybrid OLAP
  - combination of MOLAP and ROLAP
  - full data repository as a ROLAP database
  - partitioning: data subsets are downloaded to a MOLAP workplace
    - data cube tailored to specific analysis needs
    - easier access to less complex data
    - efficiency advantages of MOLAP are optimally used

OLAP Operations

- drill down: zooming into a finer granularity level
- roll up: zooming out to a more coarse granularity level (aggregation)



OLAP Operations

- cube: precomputation of a full data cube
  - generalized roll up
  - n attributes
    $\rightarrow$ aggregated values for 2^n attribute combinations
    `group by` –;
    `group by` a_1;
    `group by` a_2;
    `group by` a_3;
    `group by` a_1, a_2;
    `group by` a_2, a_3;
    `group by` a_1, a_3;
    `group by` a_1, a_2, a_3;

OLAP Operations

- cube corresponds to a (n-dimensional) cross tabulation

	small	medium	large	total
budget	24	31	12	67
premium	11	15	17	43
total	35	46	29	100

OLAP Operations

- relational representation of the cube

quality	size	amount
budget	small	24
budget	medium	31
budget	large	12
budget	all	67
premium	small	11
premium	medium	15
premium	large	17
premium	all	43
all	small	35
all	medium	46
all	large	29
all	all	100

OLAP Operations

- slice: dimension reduction by value selection
- dice: slice on two or more dimensions
- pivot / rotate: reorient the cube
  - only for navigation purposes (visualization, dimensionality reduction)
- window: range query
- ranking: sorting fact values along a dimension
- visualisation (playing around with data)

OLAP Extentions to SQL (RISQL)

- Decode: replace internal codes by readable versions
- Cume: computes a running (or cumulative) total of an attribute
- MovingAvg(n): computes the moving average of an attribute with a window size of n
- MovingSum(n): computes the moving sum of an attribute with a window size of n
- Rank ... When: compute the ranking of the top n or bottom n tuples according to the values of an attribute
- RatioToReport: percentage of an attribute value with respect to the total for that attribute
- Tertile: three valued binning (high, medium, low) with respect to the values of an attribute
- CreateMacro: define a parameterized macro for repeated use

Data Warehousing and OLAP

- Decision support systems
- Dimensional Modelling
- Data Warehouses
- Online Analytical Processing
- Data Warehouse 2.0

Data Warehouse 2.0

novel requirements

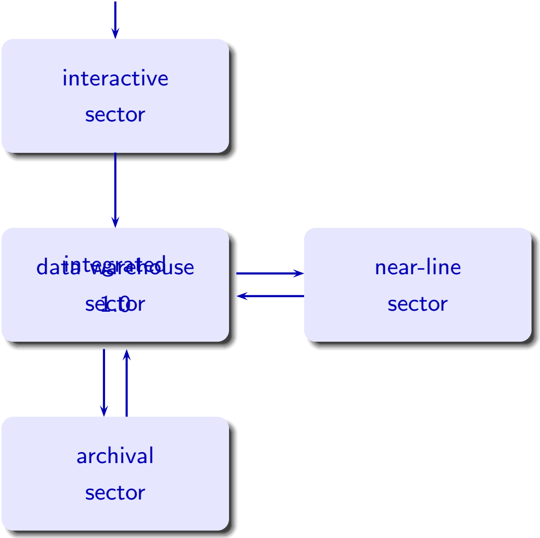
- dealing with all kinds of data
    - in particular unstructured ones
- rapid integration of data from online transactions
- malleability
    - adaptation to changing requirements
- maintaining meta data

Data Warehouse 2.0

recognition of the life cycle of data

- data volume is huge
- but probability of access declines over time
  - recent data is used more frequently
- cost awareness
  - assigning the same kind of storage media across the data warehouse wastes resources
  - goal: scalability with sub-linear resource requirements

Architecture



The Interactive Sector

- receives the data from online applications
- rapid response transactions
- updates are possible
- mixed granularity data
- responses can be inconsistent (data has not yet been integrated)
- sends the data to the integrated sector

The Integrated Sector

- receives the data from the interactive sector through an ETL procedure
- integration: transition from application to corporate data
- no updates of data, but keeping record of any changes in the interactive sector
- data is maintained on the lowest level of granularity
    - arbitrary decision procedures with different granularity requirements have to be supported

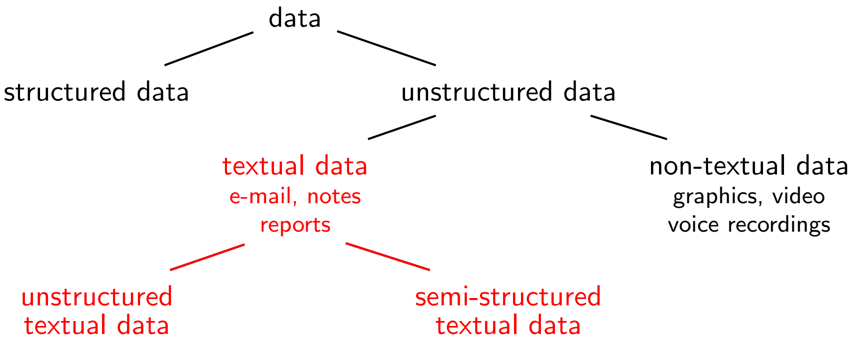
The Near-Line Sector

- caches data from the integrated sector wit low probability of access
- low cost mass storage devices (cartridge robots)

Challenges

- referential integrity across sectors
- availablity of meta data
  - focus on enterprise-wide meta data in addition to the local ones
  - focus on business-oriented meta data in addition to the technology-oriented ones
- maintenance of meta data
  - in the archival sector meta data is kept together with the data
  - in the other sectors an active repository is set up

Unstructured Data



- available textual data usually one order of magnitude larger than the structured ones
- useful applications combining textual with structured data
- high share of irrelevant data

Semistructured Data

two different notions

- text formatted in a structured manner

Service report	
date:	2014-05-11
time (from – to):	10 – 12 am
service person:	John Myers
maintenance procedure:	checking the cooling system
repairs done:	O-ring replaced (pump 17)
additional observations:	oil leakage at compressor unit

- unstructured text with partial markup

  The <maintenance_procedure> inspection of the cooling system </maintenance_procedure> was carried out on <date> Monday, May 11th, 2014 </date> between <time_interval> <start> 10 </start> and <end> 12 am </end> </time_interval> by <service_person> John Myers </service_person>. An <repair> O-ring was replaced at pump 17 </repair> and an <observation> oil leakage at the compressor unit </observation> was noticed.

Unstructured (Textual) Data

- initially no schema information available
  $\rightarrow$ schema information derived from the data
- no updates required, only load and access
- little amount of unstructured data in the interactive sector
- normally no near-line sector for unstructured data
- standards for archiving often defined by legal requirements

ETL for Textual Data

- transforming unstructured data into a structured representation

- removal of formatting information
- removal of non-textual information
  - graphics, numbers, ...
- removal of stop words (function words)
  - high frequency words
  - assumption: neglectable contribution to the content of a text
  - English: *the, a, an, and, or, of, for, ...*
  - German: *der, die, das, den, dem, und, oder, ...*
- removal of interpunction characters
- removal of infrequent words (too many)

ETL for Textual Data

- lemmatization: determining the (canonical) citation form
- often approximated by stemming: removal of inflectional endings
- but many exceptions
  - English:
    cars $\rightarrow$ *car*, *glasses* $\rightarrow$ *glass*,
    children $\rightarrow$ *child*, *men* $\rightarrow$ *man*
  - German:
    Bilder $\rightarrow$ *Bild*, *Maler* $\rightarrow$ *Maler*,
    Bildern $\rightarrow$ *Bild*, *Malern* $\rightarrow$ *Maler*,
    Zeiten $\rightarrow$ *Zeit*, *Fallen* $\rightarrow$ *Falle*,
    Äste $\rightarrow$ *Ast*, *Ähren* $\rightarrow$ *Ähre*
  - splitting of compounds:
    Weltuntergang $\rightarrow$ *Welt* + *Untergang*
    Innovationsschwäche $\rightarrow$ *Innovation* + *Schwäche*

ETL for Textual Data

- normalization, harmonization
  - inconsistent terminology
  - alternative spellings
  - ambiguous terminology
- approaches
  - superimposing general text to the specific one
  - synonym replacement: original wording is lost
  - synonym concatenation: adding a standardized term

    head/director vs. head/auris
- required resources
  - terminological databases (thesaurus, taxonomy, glossary, ontology)
  - human involvement
  - no reliable automatic procedures available
- terminology is constantly changing

ETL for Textual Data

- ambiguity
  head (of a human or animal)
  head (of a department/government/school/...)
  head (of a glass of beer)
  head (of liquids)
  head (core of a boil; medicine)
  head (of a cask)
  head (of a pin)
  head (of an article/a section)
  head (of a flock of animals; head count)
  head (of cereal)
  head (of a water power station)
  head (of a cask/a bin/a container/a tank)
  head (of a yacht)
  ...
- 1873 entries in Leo (dict.leo.org)

ETL for Textual Data

- building document indexes
- establishing links to structured data
- clustering of texts
    - terminology extraction
- pattern recognition to detect/create semi-structured data

Searching Textual Data

- direct keyword-based search
- query expansion based on taxonomic information
  - synonyms
  - semantically similar words
  - more general / more specific words
- query expansion can and must be tailored depending on the number of hits

Analysis of Textual Data

- preferably using existing tools for structured data
- many analysis tasks require specifically tailored tools
  - co-reference detection
  - sentiment detection
  - opinion mining
  - trend detection
  - ...
- borderline between ETL and analysis procedures is shifting

Metadata for Textual Data

- documenting the data sources and the ETL process

- thesauri, taxonomies, glossaries, ontologies
- stop word lists
- synonym definitions
- homograph mappings
- alternative spellings

Change Management

- data warehouse needs to support changing requirements
- adaptation as a continuous maintenance activity
- project-based change mode is not appropriate
  - next change comes faster than the last update has been accomplished

Change Management

- change in content
- most important: changes in the semantics (schema adaptation, new attributes)
- separating semantically static from semantically temporal data

semantically static:	ID, name, address, data of birth, ...
semantically temporal:	cell-phone number, mail address, fax number, ...

- creating snapshots of schema information
- maintaining a history of schema information