The data warehouse architecture

**The back room**
- External sources
- Operational source systems
- Extract Transform Load
- Data warehouse
- Data marts

**The front room**
- Analysis/OLAP
- Query/Reporting
- Data mining
- Data access tools (RK)
- End user tools
- Presentation (OLAP) servers

Operational source systems (RK)
- Legacy systems
- OLTP/TP systems

Data staging area (RK)
- Back end tools

Data presentation area (RK)
- "The data warehouse"

Operational source systems
- Front end
- Back end
- Business Intelligence tools

Product 1
- Time 1
- Value 1
- Value 11

Product 2
- Time 2
- Value 2
- Value 21

Product 3
- Time 3
- Value 3
- Value 31

Product 4
- Time 4
- Value 4
- Value 41
Operational Source Systems

Operational source systems characteristics:

- the source data often in OLTP (Online Transaction Processing) systems, also called TPS (Transaction Processing Systems)
- high level of performance and availability
- often one-record-at-a-time queries
- already occupied by the normal operations of the organisation

OLTP vs. DSS (Decision Support Systems)
OLTP vs. OLAP (Online analytical processing)

Operational Source Systems

More operational source systems characteristics:

- a OLTP system may be reliable and consistent, but there are often inconsistencies between different OLTP systems

- different types of data format and data structures in different OLTP systems AND DIFFERENT SEMANTICS
Operational Source Systems

Kimball et al’s assumptions (p 7):

- Source systems are not queried in the broad and unexpected ways
- Maintain little historical data
- Each source systems is often a natural stovepipe application

Operational source systems

Data staging area

Operational source systems

Data staging area

Data presentation area

Data access tools

External sources

Operational source systems

Extract Transform Load

Data warehouse

Serve

Data marts

Data warehouse

Analysis/OLAP

Query/Reporting

Data mining

Product 1

Product 2

Product 3

Product 4

Value 1

Value 2

Value 3

Value 4

Time 1

Time 2

Time 3

Time 4

Data warehouse

Operational source systems

Data presentation area

Data access tools
The Data Staging Area

Often the most complex part in the architecture, and involves...

- Extraction (E)
- Transformation (T)
- Load (L)
- Indexing

ETL-tools can be used
Scripts for extraction, transformation and load are implemented

Data staging area

Extraction
means reading and understanding the source data and copying the data needed for the data warehouse into staging area for further manipulation, i.e. transformation
Transformation involves...

- data conversion/transformation
  (specify transformation rules to convert to a common data format and common terms/semantics)
- data cleaning/cleansing
  - data scrubbing (use domain-specific knowledge (e.g., postal addresses) to check the data)
  - data auditing (discover suspicious pattern, discover violation of stated rules)
- combining data from multiple sources
- assigning warehouse (surrogate) keys
- data aggregation

A debate question:
Should the data in the data staging area be stored in a 3NF relational database and loaded into the presentation area for querying and reporting?

Kimball (p 8-9): a 3NF relational database in data staging area requires more time and resources for development, periodic loading and updating and more capacity of storing the multiple copies of the data
Flat file

db2 table(s)

Staging area for checking, analysing, cleaning, complementing etc transaction data

SQL, C++ ??

db2 preliminary target DW

+aggregation (new program)

db2 final target DW

Fees (manually adjusted to individual agreements)

Customer data

Start balance

Fees

Various source files

Customer data

Some cleansing and scrubbing may be needed here

Three star/join schemas comprising altogether 8 tables

Fact tables:
- transactions (10 attributes)
- fees (7 attributes)
- start balance (4 attributes)

Dimensional tables:
- time (7 attr)
- customer (> 40 attr)
- company (> 90 attr)
- product (13 attr)
- “Service charged” (2 attr)

E complemented with some aggregated tables

---

A Real World Example

---

DW architecture: Data presentation area

External sources

Operational source systems

Operational source systems

Data staging area

Data presentation area

Data access tools

Data warehouse

Data marts

Analysis/OLAP

Query/Reporting

Data mining

Operational source systems

Data presentation area
What is OLAP?

- Acronym for "On-line analytical processing"
- A decision support system (DSS) that support ad-hoc querying, i.e. enables managers and analysts to interactively manipulate data. The idea is to allow the users to easy and quickly manipulate and visualise the data through multidimensional views, i.e. different perspectives.

Kimball: Dimensional modelling
3 NF modelling vs. Dimensional modelling

Key difference between 3NF and Dimensional modelling:
- the degree of normalisation

3 NF modelling
- a logical design technique to eliminate data redundancy to keep consistency and storage efficiency, and makes transaction simple and deterministic
- ER models for enterprise are usually complex, e.g. they often have hundreds, or even thousands, of entities/tables

Dimensional modelling
- a logical design technique that present data in a intuitive, i.e. easier to navigate for the user
- allow high performance access/queries (the complexity of 3NF models overwhelms the database systems optimizer, which means bad performance)
- aims at model decision support data

Data presentation area - Data marts

Kimball et al (p.10-12 and 396)

“we refer to the presentation area as a series of integrated data marts”

“a data mart is a flexible set of data, ideally based on the most atomic (granular) data possible to extract from operational source, and presented in a symmetric (dimensional) model that is resilient when faced with unexpected user queries”

“in its most simplistic form a data mart represent data from a single business process” (business process=purchase order, store inventory and so on)
Data marts

The data warehouse bus architecture

[Kimball et al, p 78-79]
A dimensional model for a large data warehouse consists of between 10 and 25 similar-looking data marts. Each data marts will have 5 to 15 dimensional tables.

Kimball et al’s strong opinions (p.10-12)

- all data in the presentation area should be presented, stored and accessed in dimensional models
- the data marts must contain detailed, atomic data (it is unacceptable that the detailed data should be locked up in 3 NF models for drill-down)
- the data marts dimensions should be conformed for drill-across techniques, which tie the data marts together in the data warehouse bus architecture
More about data marts:

- far smaller data volumes, fewer data sources
- easier data cleaning process, faster roll-out
- allows a "piecemeal" approach to some of the enormous integration problems involved in creating an enterprise wide data model, but complex integration in the long term
The presentation/OLAP servers

**Extended Relational DBMS (ROLAP servers)**
- data stored in RDB
- star-join schemas
- support SQL extensions
- index structures

**Multidimensional DBMS (MOLAP servers)**
- data stored in arrays (n-dimensional array)
- direct access to array data structure
- excellent indexing properties
- poor storage utilisation, especially when the data is sparse.

More about presentation servers

What is characteristics regarding data warehouse, according to Chaudhiri&Dayal:

- Index structures (bit map indexes, join indexes)
- SQL extensions (operators like Cube, Crossjoin)
- Materialised views (pre-aggregations)
What is metadata?

“Data about data”/“Information about data”

Main functions are to give...

• data definitions
• the origin of data
• the structure of data
• rules for the selection and transfer of data
• qualitative and quantitative data about data

Contained in metadata repository →
The metadata repository

An integrated complete source of metadata

- is at the heart of the data warehouse architecture
- supports the information needs of...
  - system developers
  - data administrators
  - system administrators
  - users
  - applications on the data warehouse
- very complex data structure
- must contain full version history
- must always be up to date

Metadata life cycle activities

- **Collection**
  - identify and capture metadata in a central repository
- **Maintenance**
  - establish processes to synchronise metadata with the changing data structure
- **Deployment**
  - provide metadata to users in the right form and with the right tools
Different types of metadata

- **Administrative metadata**
  (includes all information necessary for setting up and using a DW, e.g. Information about source databases, dw schemas, dimensions, hierarchies, predefined queries, physical organisation, rules and script for extraction, transformation and load, back-end and front end tools)

- **Business metadata**
  (business terms and definitions, ownership of data)

- **Operational metadata**
  (information collected during the operations of the DW, e.g. usage statistics, error reports)

## DW architecture: End user applications

![Diagram of DW architecture]

- **External sources**
- **Operational DBs**
- **Operational source systems**
- **Data staging area**
- **Data marts**
- **Data presentation area**
- **Data access tools**
- **Metadata repository**
- **Extract Transform Load Refresh**
- **Data warehouse**
- **OLAP servers**
- **Serve**
- **Analysis**
- **Query/Reporting**
- **Data mining**
End user applications

- OLAP tools, BI apps, DSS
- Query/Reporting tools
- Data mining

Spreadsheet output of OLAP tool

<table>
<thead>
<tr>
<th>Product Group</th>
<th>Region</th>
<th>First Quarter - 1997</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group A</td>
<td>ABC</td>
<td>1245</td>
</tr>
<tr>
<td>Group A</td>
<td>XYZ</td>
<td>34534</td>
</tr>
<tr>
<td>Group B</td>
<td>ABC</td>
<td>45543</td>
</tr>
<tr>
<td>Group B</td>
<td>XYZ</td>
<td>34533</td>
</tr>
</tbody>
</table>

Column headers (join constraints):
- Product
- Product group
- Month
- Quarter
- Office
- Region

Column header (application constraint):
- Answer set representing fiscal event
Graphical output of OLAP tool

Functionalities of OLAP tools

- **Drill-down** - decreasing the level of aggregation
- **Drill-up/Roll-up/Consolidation** - increasing the level of aggregation
- **Drill-across** - move between different star-join schemas using conformed dimensions and joins
- **Slicing and dicing** - ability to look at the database from different views, e.g. one slice shows all sales of product type within regions, another slice shows all sales by sales channel within each product type
- **Pivoting** - e.g. change columns to rows, rows to columns
- **Ranking** - sorting

“Think of an OLAP data structure as a Rubik’s Cube of data that users can twist and twirl in different ways to work through what-if an what-happend scenarios”  

[Lee Thé]
Business Intelligence (BI) apps

**Strategic**
- **Who:** strategic leaders
- **What:** formulate strategy and monitor corporate performance
- **Examples:** Balance scorecard, Strategic Planning

**Operational**
- **Who:** operational managers
- **What:** execution of strategy against objectives
- **Examples:** Budgeting, Sales forecasting

**Analytical**
- **Who:** analysts, knowledge worker, controller
- **What:** ad-hoc analysis
- **Examples:** Financial and Sales Analysis, Customer Segmentation, Clickstream analysis

Problems of Data Warehousing

- Complexity of integration
  - Hidden problems with source systems
  - Data homogenisation
  - Underestimation of resources for data loading
- Required data not captured
- High maintenance
- Long duration projects
- Why not integrating the legacy applications (OLTP systems) instead?
Operational Data Store (ODS)

No singel universal definiton...

**ODS definition 1:** Implemented to deliver operational reporting, especially when neither the legacy nor the modern OLTP systems provide adequate operational reports - fixed queries and for tactical decision making

**ODS definition 2:** Built to support real-time interactions, especially in Customer Relationship Management applications - the traditional data warehouse typically is not in a position to support the demand for near-real-time data

OMG’s standards

- Meta Object Facility (MOF)
  - M3 layer
  - Meta metamodel
- UML Metamodel
  - M2 layer
  - Metamodel
- CWM Metamodel
  - M1 layer
  - Model
- Instances
  - M0 layer
  - Helen Nagy
  - Invoice no 34
The collection of metamodels by CWM can be used to model the whole data warehousing environment i.e from data sources to end use analysis, and data warehouse management.

Common Warehouse Metamodel (CWM)

• Common Warehouse Metamodel (CWM) is a language specifically design to model data warehousing and data mining applications, i.e. integrating data warehousing and business analysis (business intelligence) tools

• CWM has a lot in common with the UML metamodel but has a number of special metamodels (metaclasses), e.g modelling relational databases, multidimensional databases, OLAP, schema transformations, XML

[Kleppe et al, p.139-140 (2003)]
Why meta-modelling?

Meta model level or Reference model

Model level

[Rosemann, Green, 2002]

CWM packages

Management

<table>
<thead>
<tr>
<th>Analysis</th>
<th>Warehouse Process</th>
<th>Warehouse Operation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource</td>
<td>Transformation</td>
<td>OLAP</td>
</tr>
<tr>
<td>Foundation</td>
<td>Relational</td>
<td>Record</td>
</tr>
<tr>
<td>Object</td>
<td>Business Information</td>
<td>Data Types</td>
</tr>
<tr>
<td></td>
<td>Core</td>
<td>Behavioral</td>
</tr>
</tbody>
</table>

Packages/Metamodels

[Rosemann, Green, 2002]
CWM packages layers

- Object layer - base metamodels/packages, which are (re)used by the other metamodels/packages

- Foundation layer - extends the object layer with services required which are (re)used by the other metamodels/packages, e.g. “unique key” in the Key Indexes metamodel/package is used by relational databases, OO-databases and record-oriented

- Resource layer - defines metamodels/packages for various types of data resources

- Analysis layer - analysis-oriented metadata

- Management layer - describing the data warehousing process as a whole

CWM packages relations
CWM classifier equality

Object  Package  Classifier (Klass)  Feature (Attribut)
Relational  Schema  Table  Column
Record  Record file  RecordDef  Field
Multi Dimensional  Schema  Dimension  Dimensioned Object
XML  Schema  Element Type  Attribute

More about CWM

Common Representation
Tool Y Metamodel
Tool X Metamodel
Tool Z Metamodel
<<metamodels>>
CWM Packages
Business Dimensional Lifecycle

The Data Warehouse Architecture Framework