

Data Warehousing and Data Mining

Outline

1. Introduction and Terminology

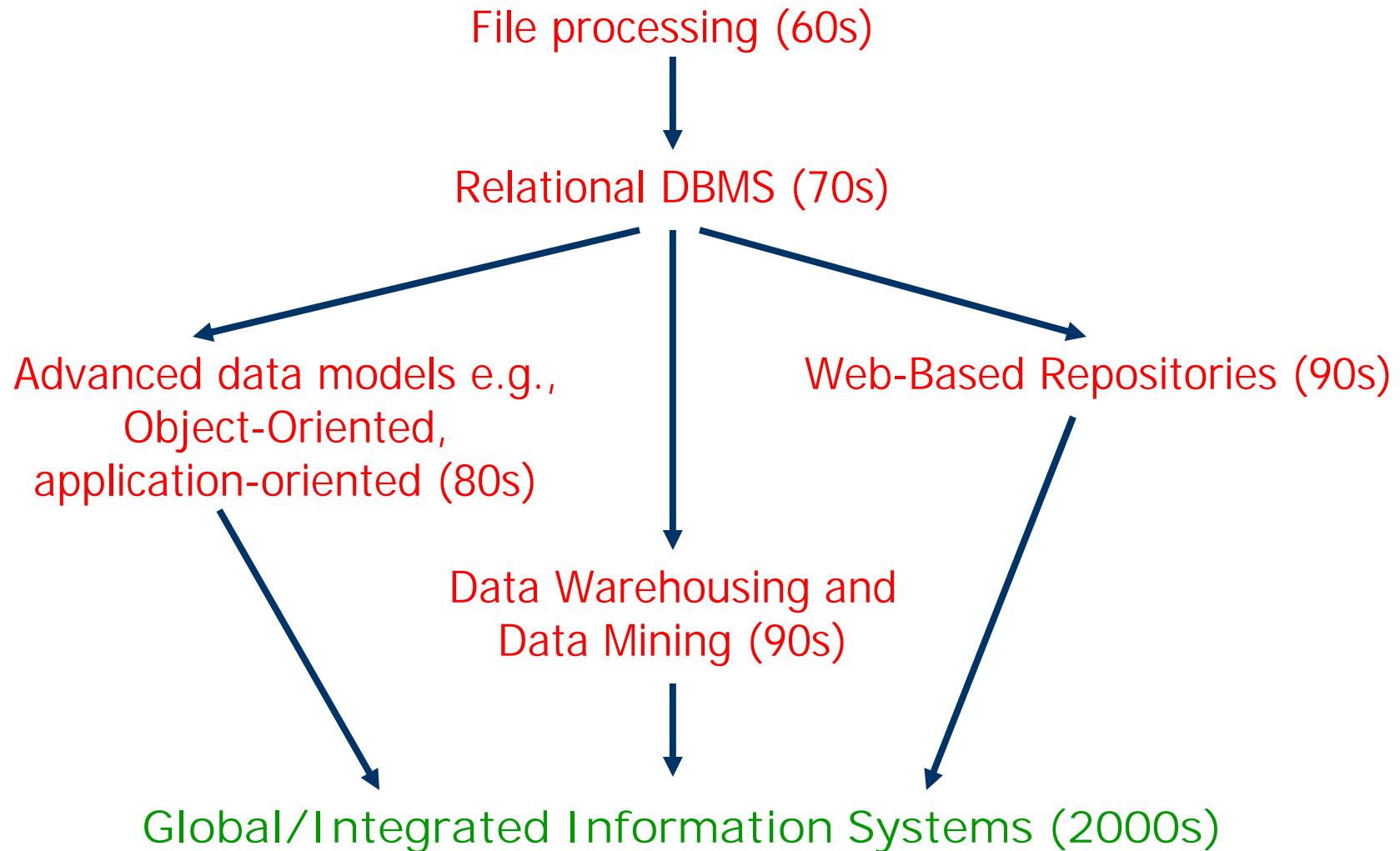
2. Data Warehousing

3. Data Mining

- Association rules
- Sequential patterns
- Classification
- Clustering

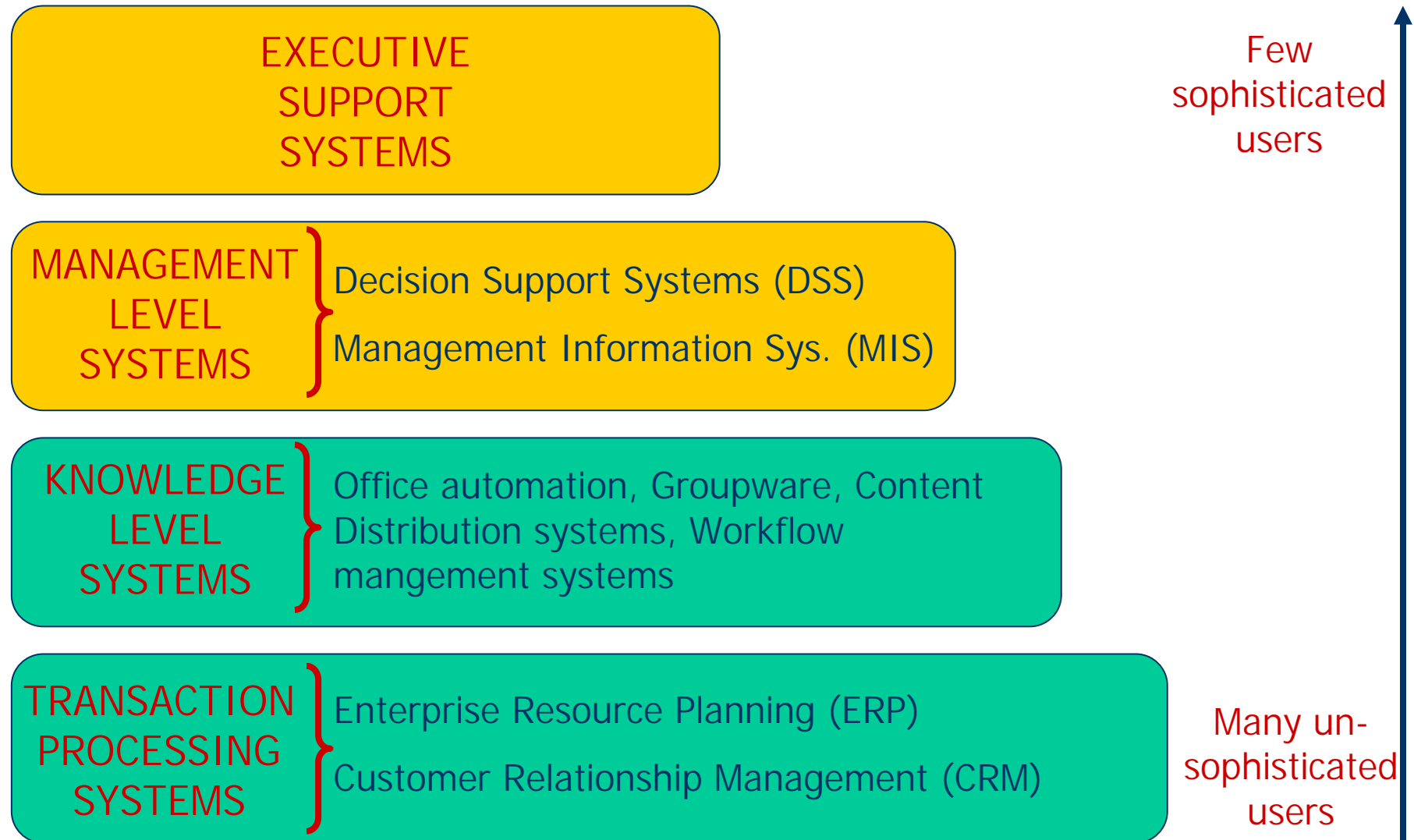
Introduction and Terminology

Evolution of database technology



Introduction and Terminology

Major types of information systems within an organization



Introduction and Terminology

Transaction processing systems:

- Support the operational level of the organization, possibly integrating needs of different functional areas (ERP);
- Perform and record the daily transactions necessary to the conduct of the business
- Execute simple read/update operations on traditional databases, aiming at maximizing transaction throughput
- Their activity is described as:

OLTP (On-Line Transaction Processing)

Introduction and Terminology

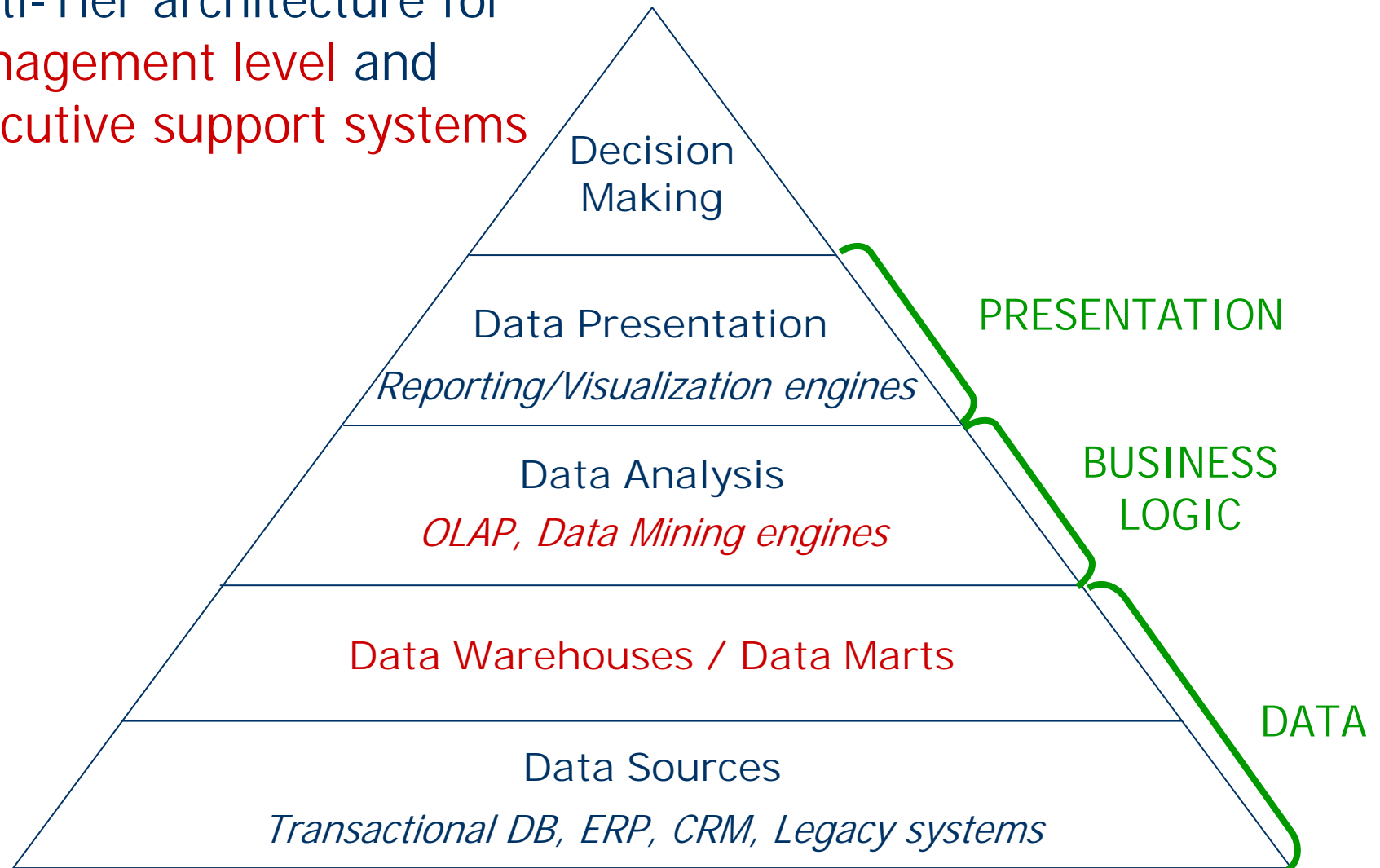
Knowledge level systems: provide digital support for managing documents (office automation), user cooperation and communication (groupware), storing and retrieving information (content distribution), automation of business procedures (workflow management)

Management level systems: support planning, controlling and semi-structured decision making at management level by providing *reports* and analyses of *current and historical data*

Executive support systems: support unstructured decision making at the strategic level of the organization

Introduction and Terminology

Multi-Tier architecture for
Management level and
Executive support systems



Introduction and Terminology

OLAP (On-Line Analytical Processing):

- Reporting based on (multidimensional) data analysis
- Read-only access on repositories of moderate-large size (typically, data warehouses), aiming at maximizing response time

Data Mining:

- Discovery of novel, implicit patterns from, possibly heterogeneous, data sources
- Use a mix of sophisticated statistical and high-performance computing techniques

Outline

1. Introduction and Terminology

2. Data Warehousing

3. Data Mining

- Association rules
- Sequential patterns
- Classification
- Clustering

DATA WAREHOUSE

Database with the following distinctive characteristics:

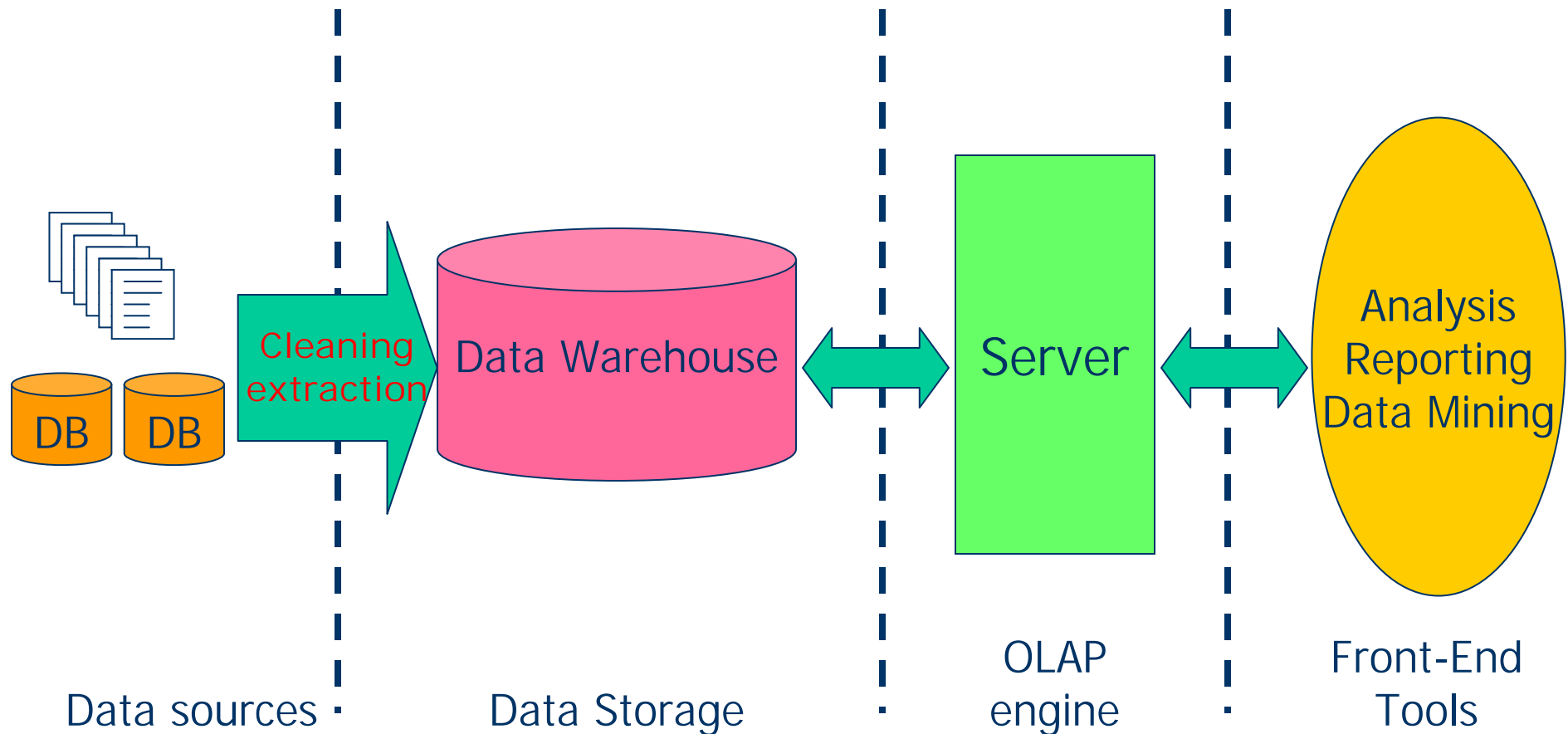
- Separate from operational databases
- Subject oriented: provides a simple, concise view on one or more selected areas, in support of the decision process
- Constructed by integrating multiple, heterogeneous data sources
- Contains historical data: spans a much longer time horizon than operational databases
- (Mostly) Read-Only access: periodic, infrequent updates

Types of Data Warehouses

- **Enterprise Warehouse:** covers all areas of interest for an organization
- **Data Mart:** covers a subset of corporate-wide data that is of interest for a specific user group (e.g., marketing).
- **Virtual Warehouse:** offers a set of views constructed on demand on operational databases. Some of the views could be materialized (precomputed)

Data Warehousing

Multi-Tier Architecture



Multidimensional (logical) Model

Data are organized around one or more **FACT TABLEs**. Each Fact Table collects a set of omogeneous events (**facts**) characterized by **dimensions** and **dependent attributes**

Example: Sales at a chain of stores

<u>Product</u>	<u>Supplier</u>	<u>Store</u>	<u>Period</u>	Units	Sales
P1	S1	St1	1qtr	30	1500€
P2	S1	St3	2qtr	100	9000€

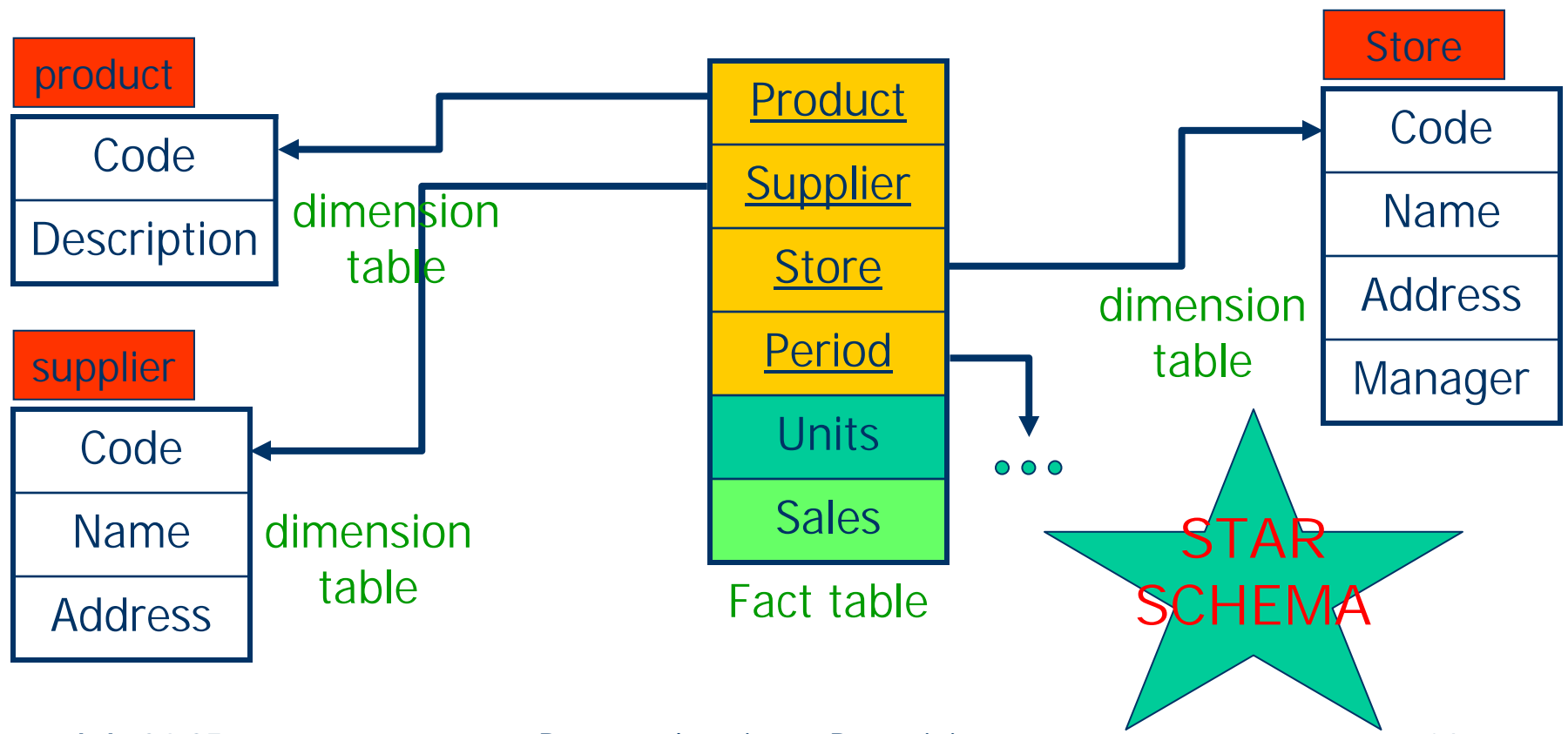
dimensions

dependent attributes

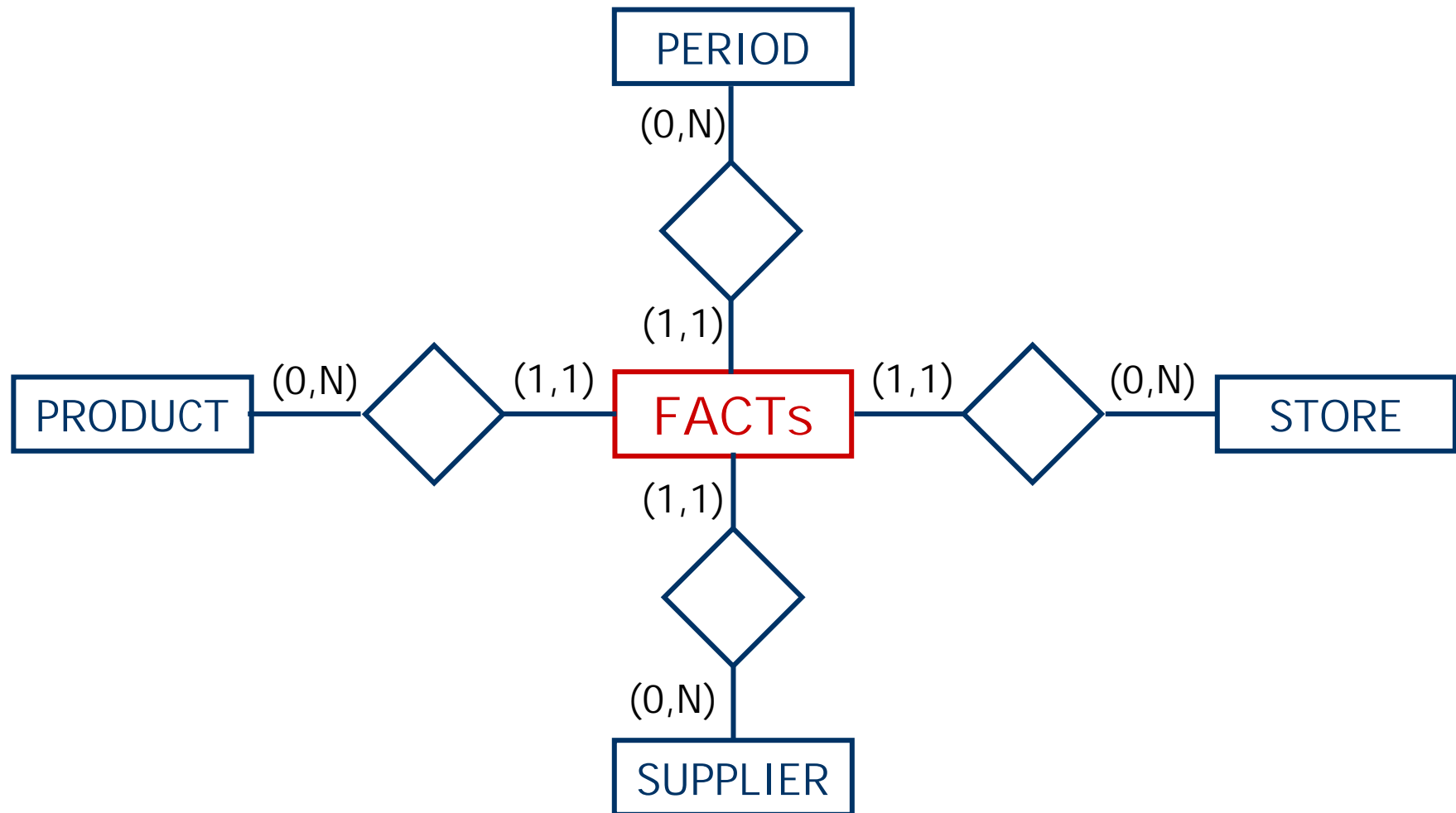
Data Warehousing

Multidimensional (logical) Model (cont'd)

Each dimension can in turn consist of a number of attributes.
In this case the value in the fact table is a foreign key referring to an appropriate **dimension table**



Conceptual Star Schema (E-R Model)



OLAP Server Architectures

They are classified based on the underlying storage layouts

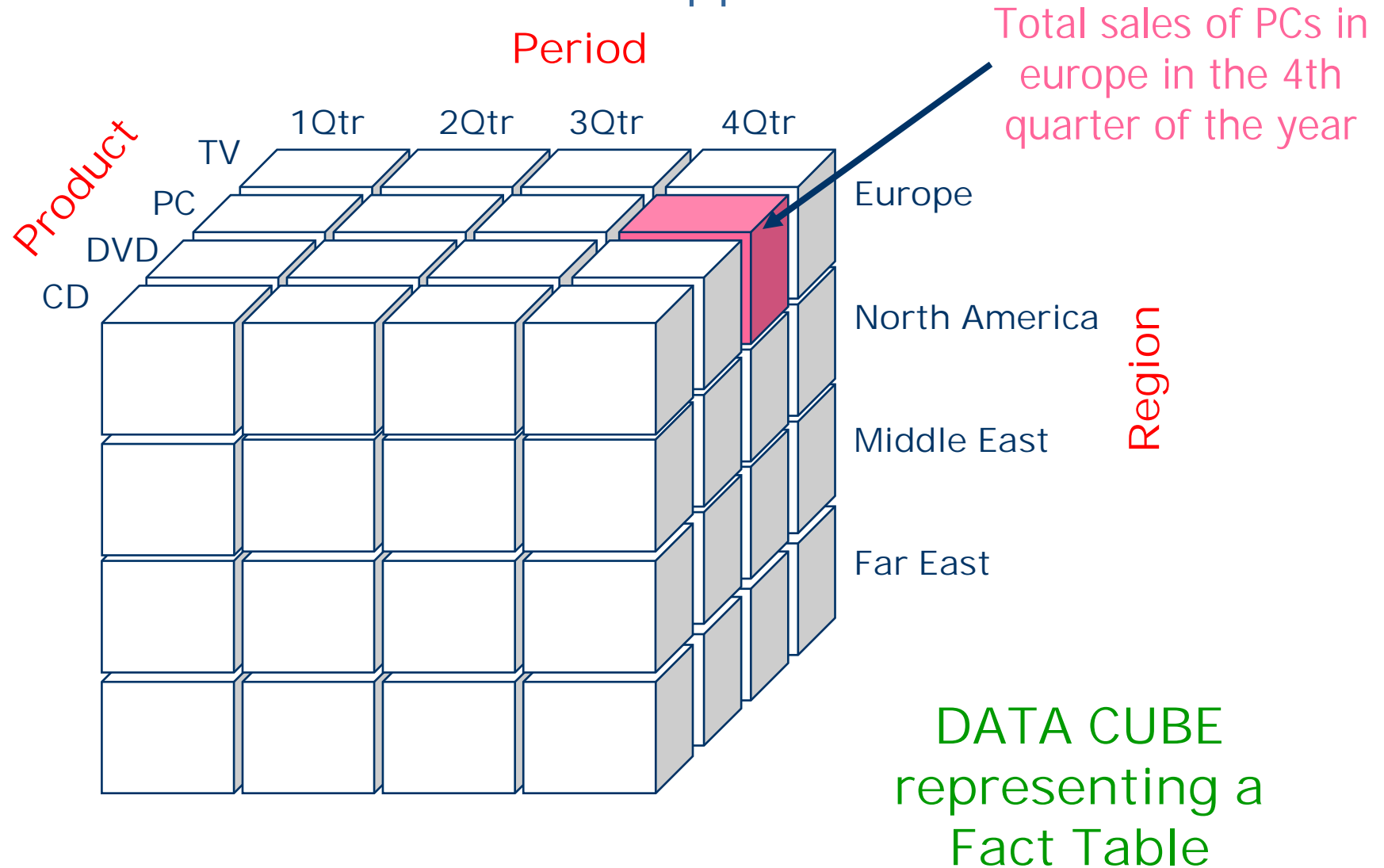
ROLAP (Relational OLAP): uses relational DBMS to store and manage warehouse data (i.e., table-oriented organization), and specific middleware to support OLAP queries.

MOLAP (Multidimensional OLAP): uses array-based data structures and pre-computed aggregated data. It shows higher performance than OLAP but may not scale well if not properly implemented

HOLAP (Hybird OLAP): ROLAP approach for low-level raw data, MOLAP approach for higher-level data (aggregations).

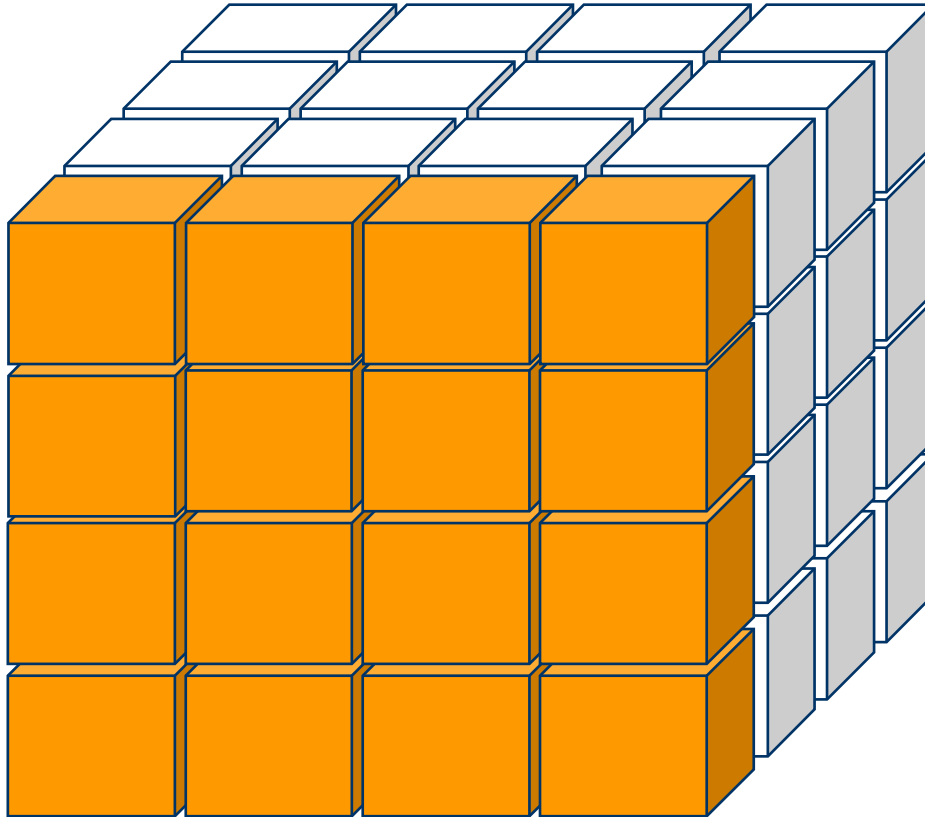
Data Warehousing

MOLAP Approach



Data Warehousing

OLAP Operations: SLICE

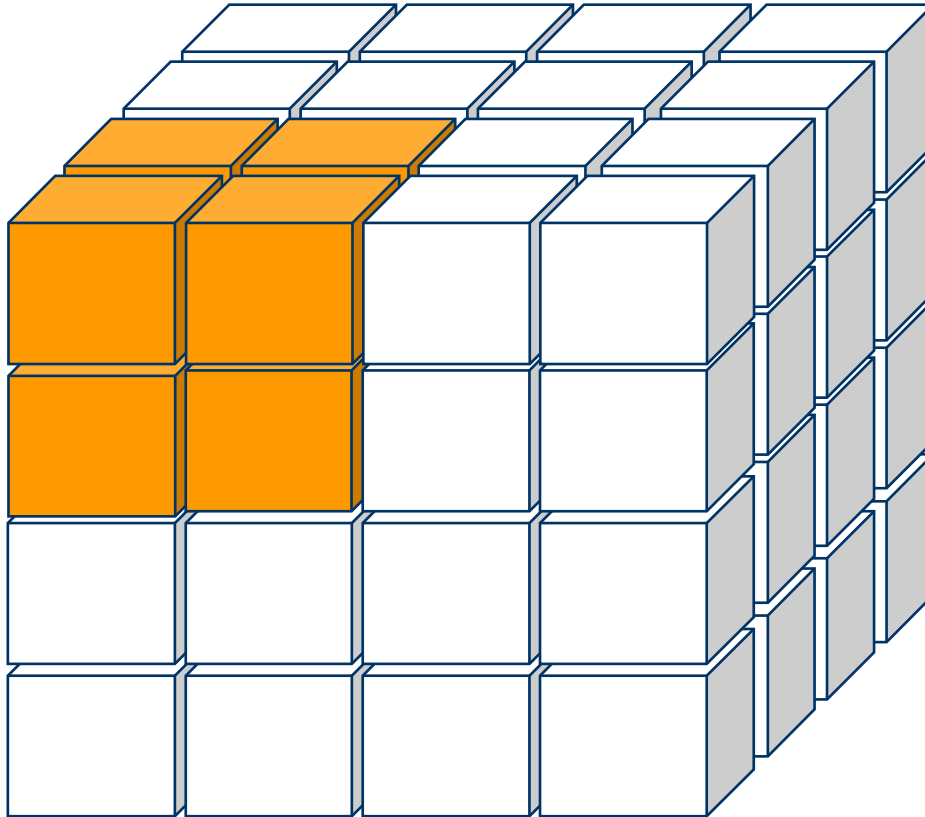


Fix values for one or more dimensions

E.g. **Product** = **CD**

Data Warehousing

OLAP Operations: DICE



Fix ranges for one or more dimensions

E.g.

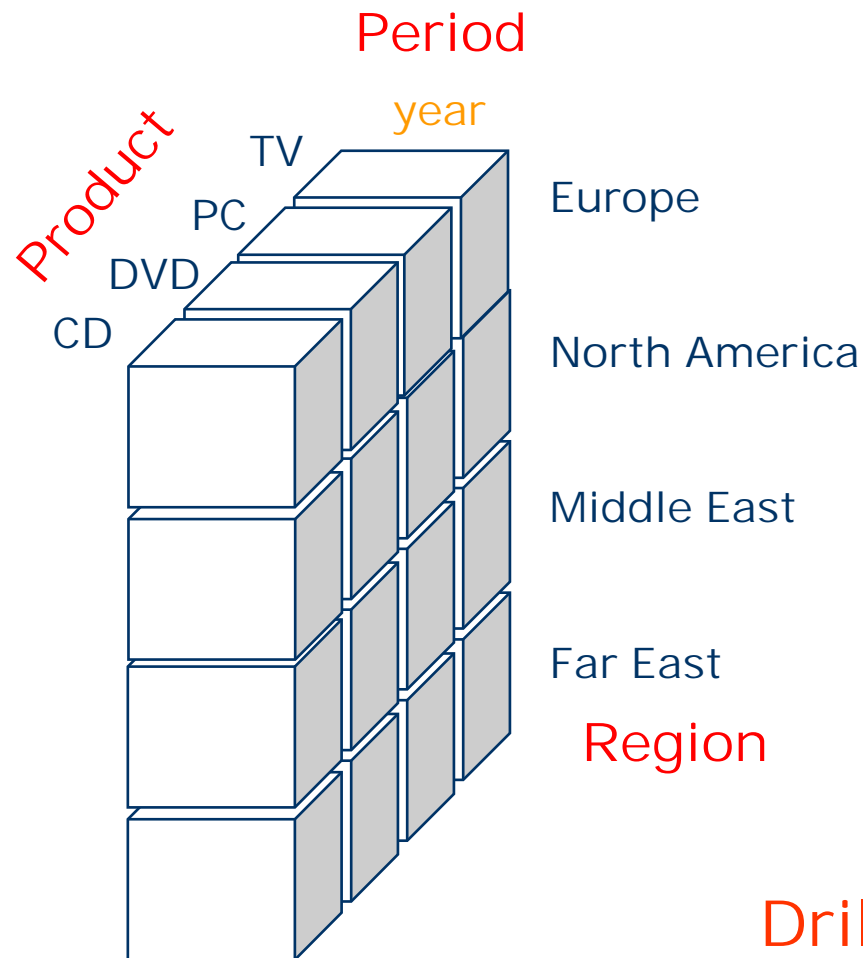
Product = CD or DVD

Period = 1Qrt or 2Qrt

Region = Europe or North America

Data Warehousing

OLAP Operations: Roll-Up



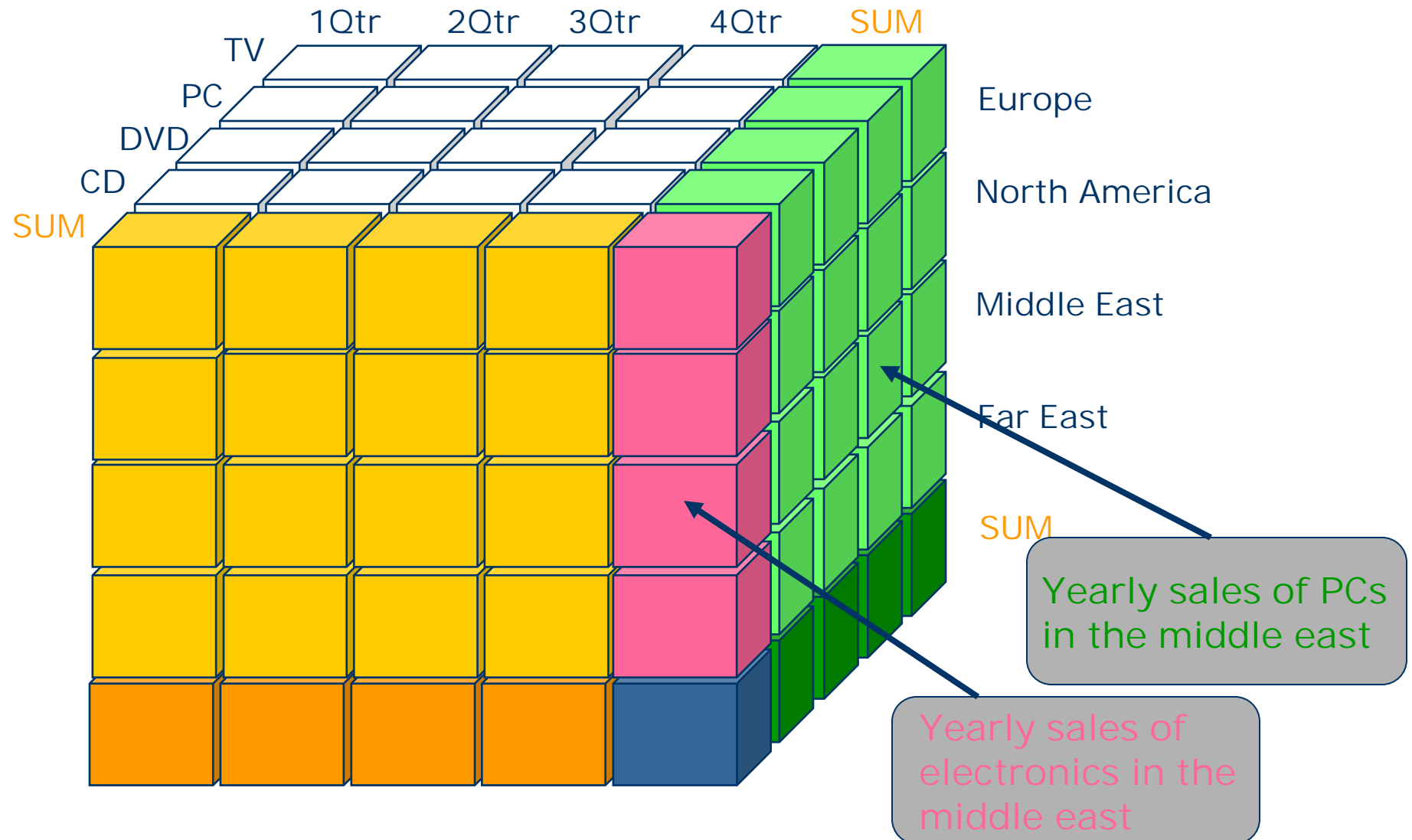
Aggregate data by grouping along one (or more) dimensions

E.g.: group quarters

Drill-Down = (Roll-Up)⁻¹

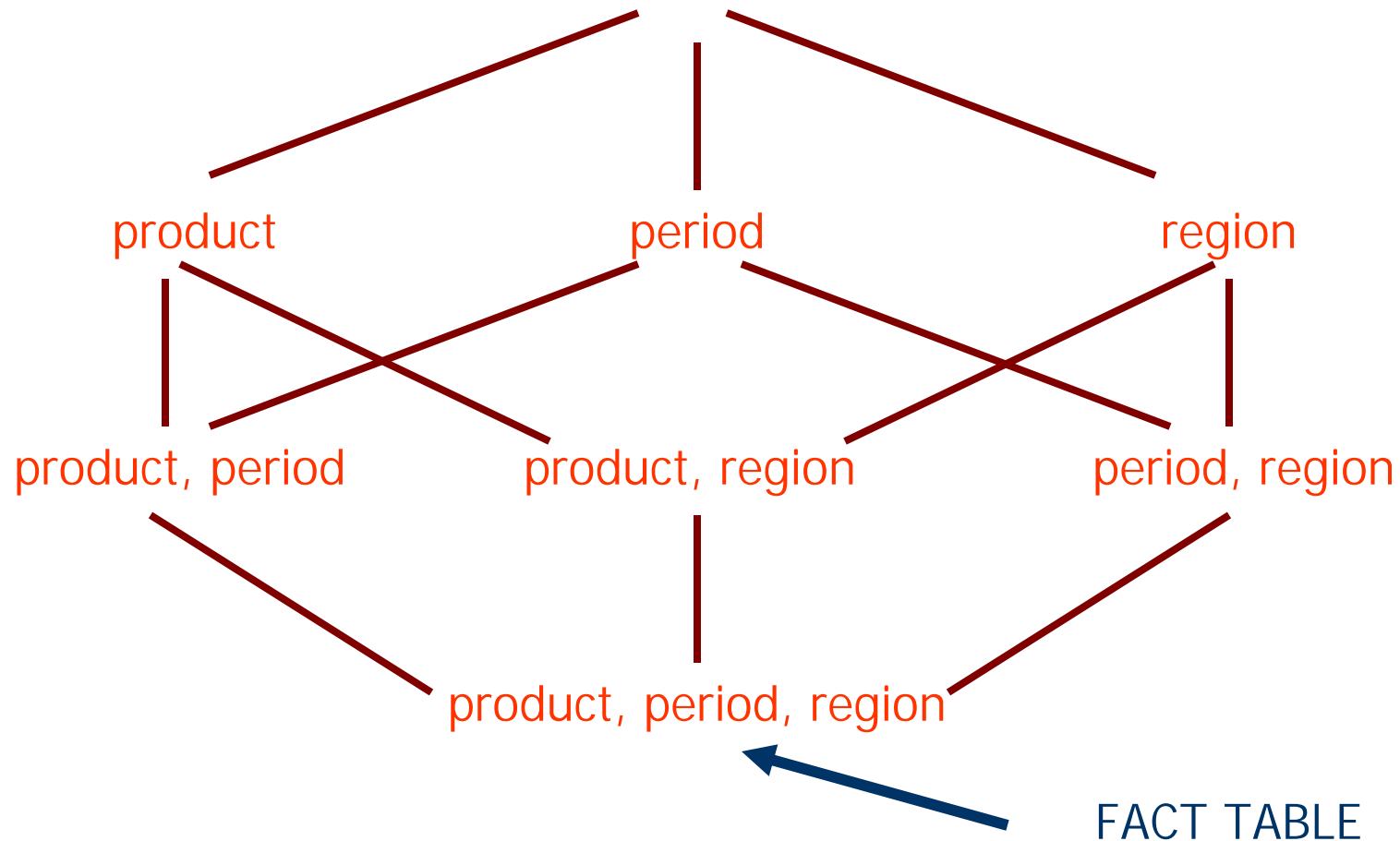
Data Warehousing

Cube Operator: summaries for each subset of dimensions



Data Warehousing

Cube Operator: it is equivalent to computing the following lattice of cuboids



Cube Operator In SQL:

```
SELECT Product, Period, Region, SUM(Total Sales)
FROM FACT-TABLE
GROUP BY Product, Period, Region
WITH CUBE
```

Data Warehousing

Product	Period	Region	Tot. Sales	
CD	1qtr	Europe	*	All combinations of Product, Period and Region
...	*	
CD	1qtr	ALL	*	All combinations of Product and Period
...	...	ALL	*	
ALL	1qtr	Europe	*	
ALL	*	
CD	ALL	Europe	*	
...	ALL	...	*	
CD	ALL	ALL	*	All combinations of Product
...	*	
ALL	1qtr	ALL	*	
...	*	
ALL	ALL	Europe	*	
ALL	ALL	...	*	
ALL	ALL	ALL	*	

Roll-up (= partial cube) Operator In SQL:

```
SELECT Product, Period, Region, SUM(Total Sales)
FROM FACT-TABLE
GROUP BY Product, Period, Region
WITH ROLL-UP
```

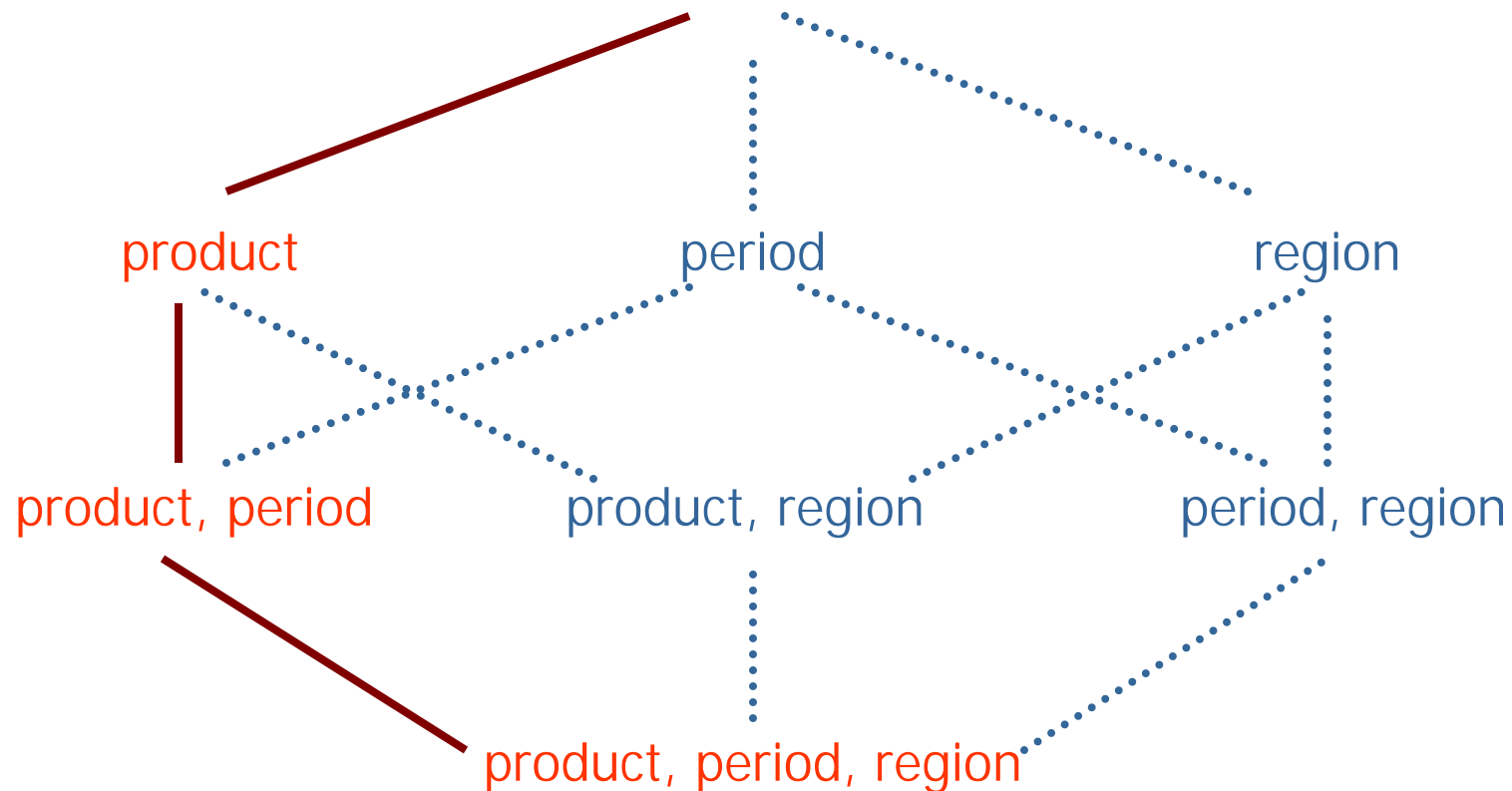
Data Warehousing

Product	Period	Region	Tot. Sales	
CD	1qtr	Europe	*	All combinations of Product, Period and Region
...	*	
CD	1qtr	ALL	*	All combinations of Product and Period
...	...	ALL	*	
CD	ALL	ALL	*	All combinations of Product
...	ALL	ALL	*	
ALL	ALL	ALL	*	

Reduces the complexity from exponential to linear in the number of dimensions

Data Warehousing

it is equivalent to computing the following subset of the lattice of cuboids



Outline

1. Introduction and Terminology

2. Data Warehousing

3. Data Mining

- Association rules
- Sequential patterns
- Classification
- Clustering

Data Mining

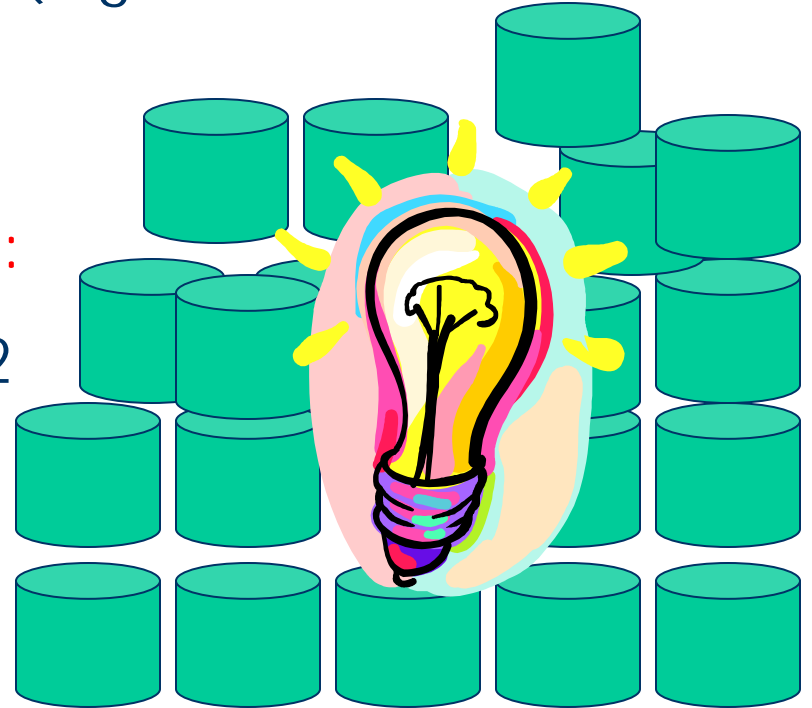
Data Explosion: tremendous amount of data accumulated in *digital repositories* around the world (e.g., databases, data warehouses, web, etc.)

Production of digital data /Year:

- 3-5 Exabytes (10^{18} bytes) in 2002
- 30% increase per year (99-02)

See:

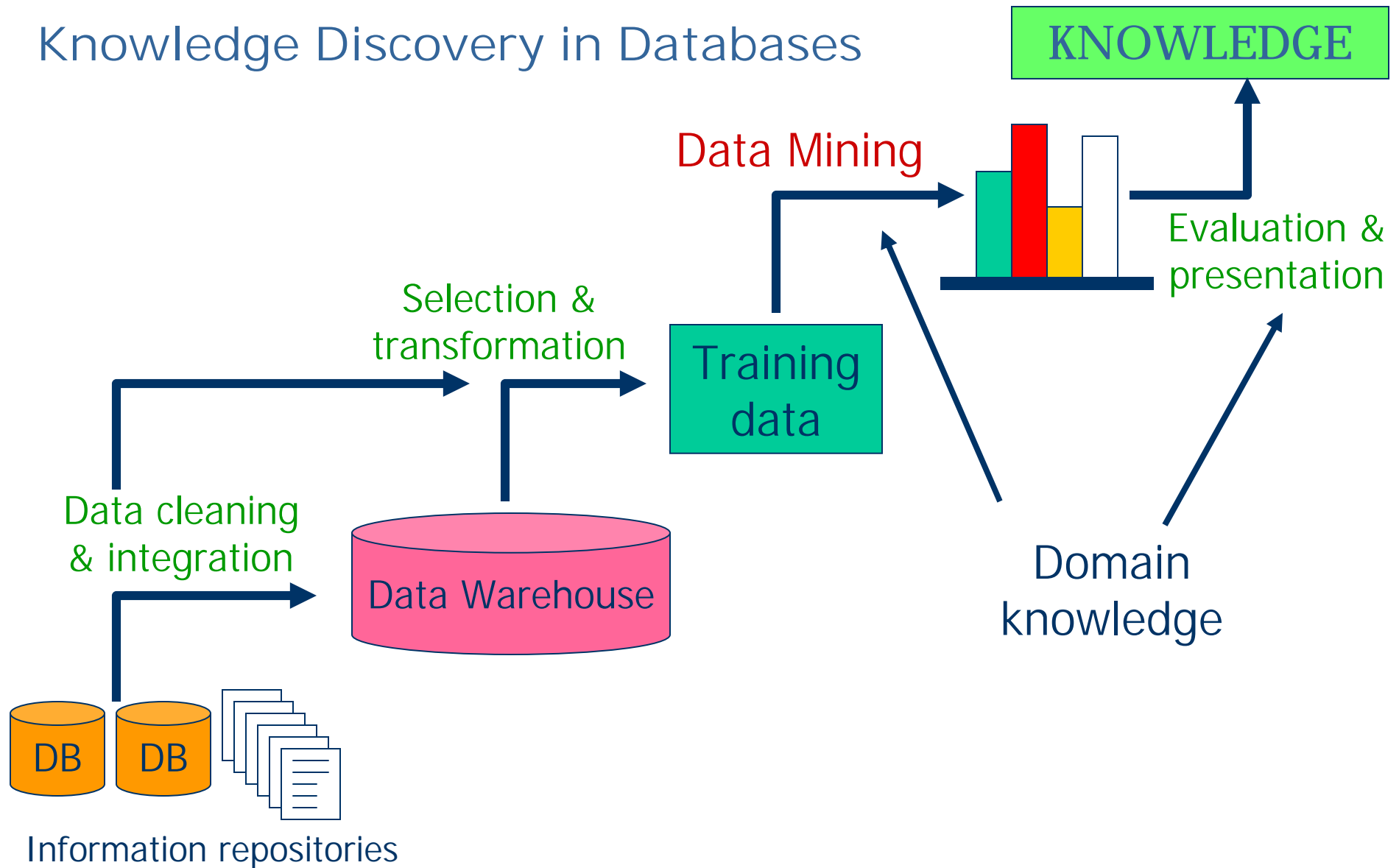
www.sims.berkeley.edu/how-much-info



We are drowning in data, but starving for knowledge

Data Mining

Knowledge Discovery in Databases



Typologies of input data:

- Unaggregated data (e.g., records, transactions)
- Aggregated data (e.g., summaries)
- Spatial, geographic data
- Data from time-series databases
- Text, video, audio, web data

DATA MINING

Process of discovering interesting patterns or knowledge from a (typically) large amount of data stored either in databases, data warehouses, or other information repositories

Interesting: non-trivial, implicit, previously unknown, potentially useful

Alternative names: knowledge discovery/extraction, information harvesting, business intelligence

In fact, data mining is a step of the more general process of knowledge discovery in databases (KDD)

Interestingness measures

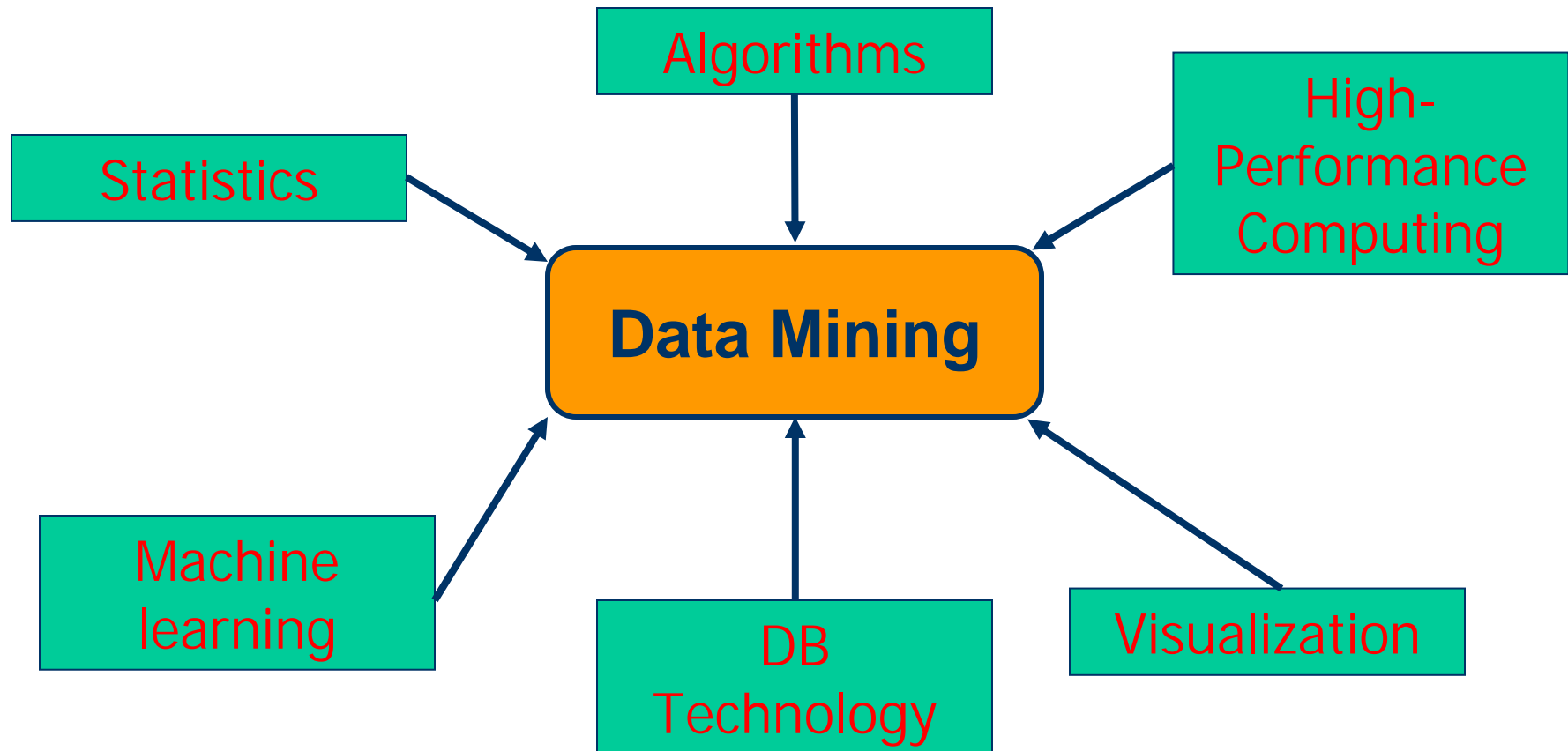
Purpose: filter irrelevant patterns to convey concise and useful knowledge. Certain data mining tasks can produce thousands or millions of patterns most of which are redundant, trivial, irrelevant.

Objective measures: based on statistics and structure of patterns (e.g., frequency counts)

Subjective measures: based on user's belief about the data. Patterns may become interesting if they confirm or contradict a user's hypothesis, depending on the context.

Interestingness measures can be employed both after and during the pattern discovery. In the latter case, they improve the search efficiency

Multidisciplinary of Data Mining



Data Mining Problems

Association Rules: discovery of rules $X \rightarrow Y$ ("objects that satisfy condition X are also *likely* to satisfy condition Y "). The problem first found application in market basket or transaction data analysis, where "objects" are transactions and "conditions" are containment of certain itemsets

Association Rules

Statement of the Problem

I = Set of items

D = Set of transactions: $t \in D$ then $t \subseteq I$

For an itemset $X \subseteq I$,

$\text{support}(X)$ = fraction or number of transactions containing X

ASSOCIATION RULE: $X \rightarrow Y$, with $Y \cap X = \emptyset$

support = $\text{support}(X)$

confidence = $\text{support}(X)/\text{support}(X \cup Y)$

PROBLEM: find all association rules with support $\geq \text{min sup}$ and confidence $\geq \text{min confidence}$

Association Rules

Market Basket Analysis:

Items = products sold in a store or chain of stores

Transactions = customers' shopping baskets

Rule X-Y **Y** = customers who buy items in X-Y are likely to buy items in Y

Of diapers and beer

Analysis of customers behaviour in a supermarket chain has revealed that **males** who on **thursdays** and **saturdays** buy **diapers** are likely to buy also **beer**

.... That's why these two items are found close to each other in most stores

Association Rules

Applications:

- **Cross/Up-selling** (especially in e-comm., e.g., Amazon)
 - Cross-selling: push complementary products
 - Up-selling: push similar products
- **Catalog design**
- **Store layout** (e.g., diapers and beer close to each other)
- **Financial forecast**
- **Medical diagnosis**

Association Rules

Def.: Frequent itemset = itemset with support $\geq \text{min sup}$

General Strategy to discover all association rules:

1. Find all frequent itemsets
2. For each frequent itemset X , output all rules $X - Y \Rightarrow Y$, with $Y \subset X$, which satisfy the min confidence constraint

Observation:

Min sup and min confidence are objective measures of interestingness. Their proper setting, however, requires user's domain knowledge. Low values may yield exponentially (in $|I|$) many rules, high values may cut off interesting rules

Association Rules

Example

Transaction-id	Items bought
10	A, B, C
20	A, C
30	A, D
40	B, E, F



Frequent Itemsets	Support
{A}	75%
{B}	50%
{C}	50%
{A, C}	50%

Min. sup 50%

Min. confidence 50%

For rule {A} → {C} :

support = support({A,C}) = 50%

confidence = support({A,C})/support({A}) = 66.6%

For rule {C} → {A} (same support as {A} → {C}):

confidence = support({A,C})/support({C}) = 100%

Dealing with Large Outputs

Observation: depending on the values of **min sup** and on the **dataset**, the number frequent itemsets can be *exponentially large* but may contain a lot of **redundancy**

Goal: determine a subset of frequent itemsets of considerably smaller size, which provides the same information content, i.e., from which the complete set of frequent itemsets can be derived without further information.

Notion of Closedness

I (items), D (transactions), min sup (support threshold)

Closed Frequent Itemsets =

$$\{X \subseteq I : \text{supp}(X) \geq \text{min sup} \ \& \ \nexists Y \supset X \text{ such that } \text{supp}(Y) = \text{supp}(X)\}$$

- It's a subset of all frequent itemsets
- For every frequent itemset X there exists a closed frequent itemset Y \supseteq X such that $\text{supp}(Y) = \text{supp}(X)$, i.e., Y and X occur in exactly the same transactions
- All frequent itemsets and their frequencies can be derived from the closed frequent itemsets without further information

Notion of Maximality

Maximal Frequent Itemsets =

$$\{X \mid \text{supp}(X) \geq \text{min sup} \ \& \ \nexists Y \supset X \text{ such that } \text{supp}(Y) \geq \text{min sup}\}$$

- It's a subset of the closed frequent itemsets, hence of all frequent itemsets
- For every (closed) frequent itemset X there exists a maximal frequent itemset $Y \supset X$
- All frequent itemsets can be derived from the maximal frequent itemsets without further information, **however their frequencies must be determined from D**

information loss

Association Rules

Example of closed and maximal frequent itemsets

Tid	Items
10	B, A, D, F, G, H
20	B, L
30	B, A, D, F, L, H
40	B, L, G, H
50	B, A, D, F
60	B, A, D, F, L, G

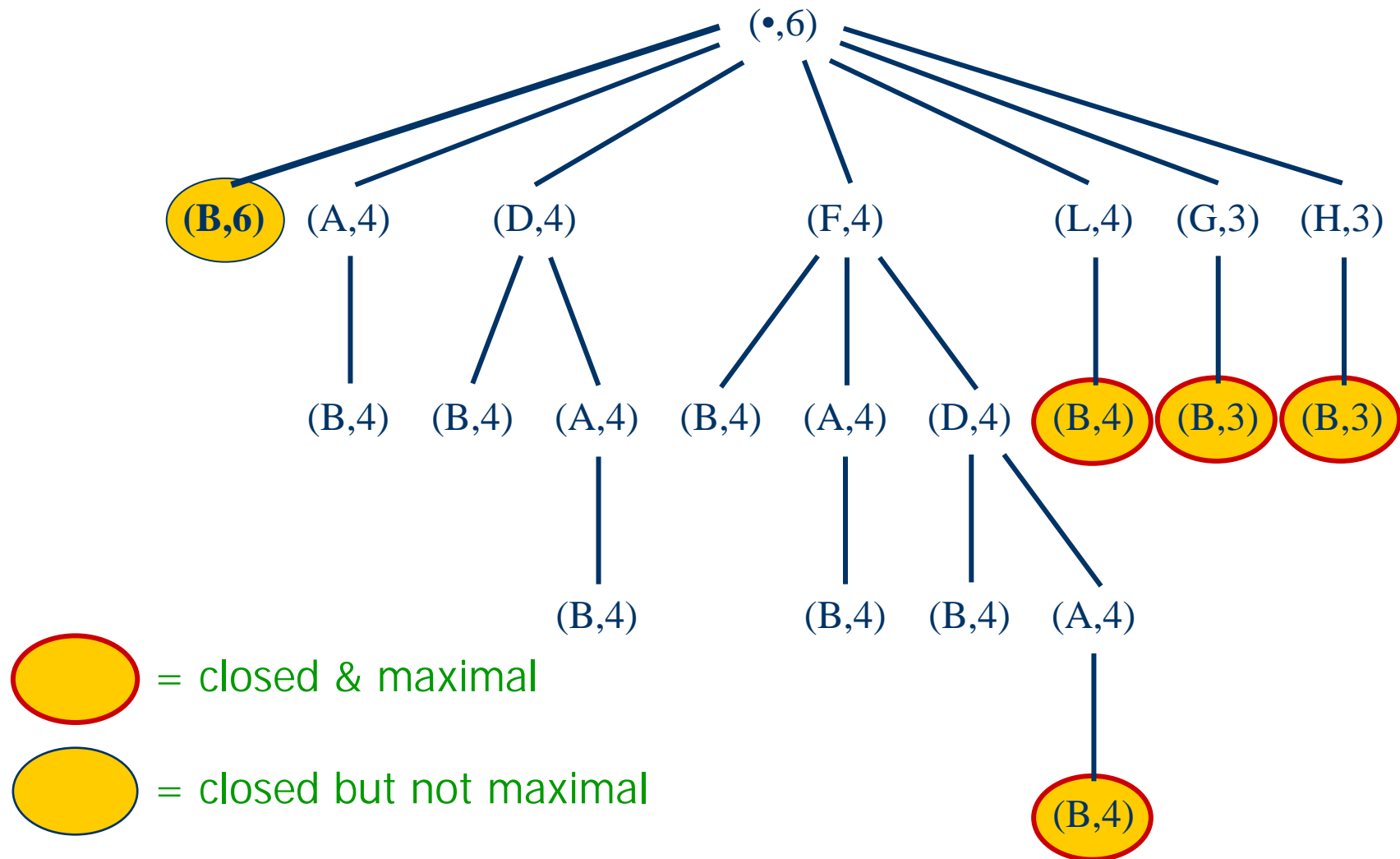
DATASET

Min sup = 3

Closed Frequent Itemsets	Maximal	Support	Supporting Transactions
B	NO	6	all
B, A, D, F	YES	4	10, 30, 50, 60
B, L	YES	4	20, 30, 40, 60
B, G	YES	3	10, 40, 60
B, H	YES	3	10, 30, 40

Association Rules

(All frequent) VS (closed frequent) VS (maximal frequent)



Data Mining Problems

Sequential Patterns: discovery of frequent subsequences in a collection of sequences (sequence database), each representing a set of events occurring at subsequent times. The ordering of the events in the subsequences is relevant.

Sequential Patterns

Sequential Patterns

PROBLEM: Given a set of sequences, find the complete set of *frequent subsequences*

sequence database

SID	sequence
10	<(a)(<u>abc</u>)(a <u>c</u>)(d)(cf)>
20	<(ad)(c)(bc)(ae)>
30	<(ef)(<u>ab</u>)(df)(<u>c</u>)(b)>
40	<(e)(g)(af)(c)(b)(c)>

A **sequence**: <(ef)(ab)(df)(c)(b)>

An element may contain a set of items. Items within an element are unordered and are listed alphabetically.

<(a)(bc)(d)(c)>
is a **subsequence** of
<(a)(abc)(ac)(d)(cf)>

For min sup = 2, <(ab)(c)> is a **frequent subsequence**

Sequential Patterns

Applications:

- Marketing
- Natural disaster forecast
- Analysis of web log data
- DNA analysis

Data Mining Problems

Classification/Regression: discovery of a model or function that maps objects into predefined classes (classification) or into suitable values (regression). The model/function is computed on a training set (supervised learning)

Statement of the Problem

Training Set: $T = \{t_1, \dots, t_n\}$ set of n examples

Each example t_i

- characterized by m features ($t_i(A_1), \dots, t_i(A_m)$)
- belongs to one of k classes ($C_i : 1 \leq i \leq k$)

GOAL

From the training data find a **model** to describe the classes accurately and synthetically using the data's features. The model will then be used to assign class labels to unknown (previously unseen) records

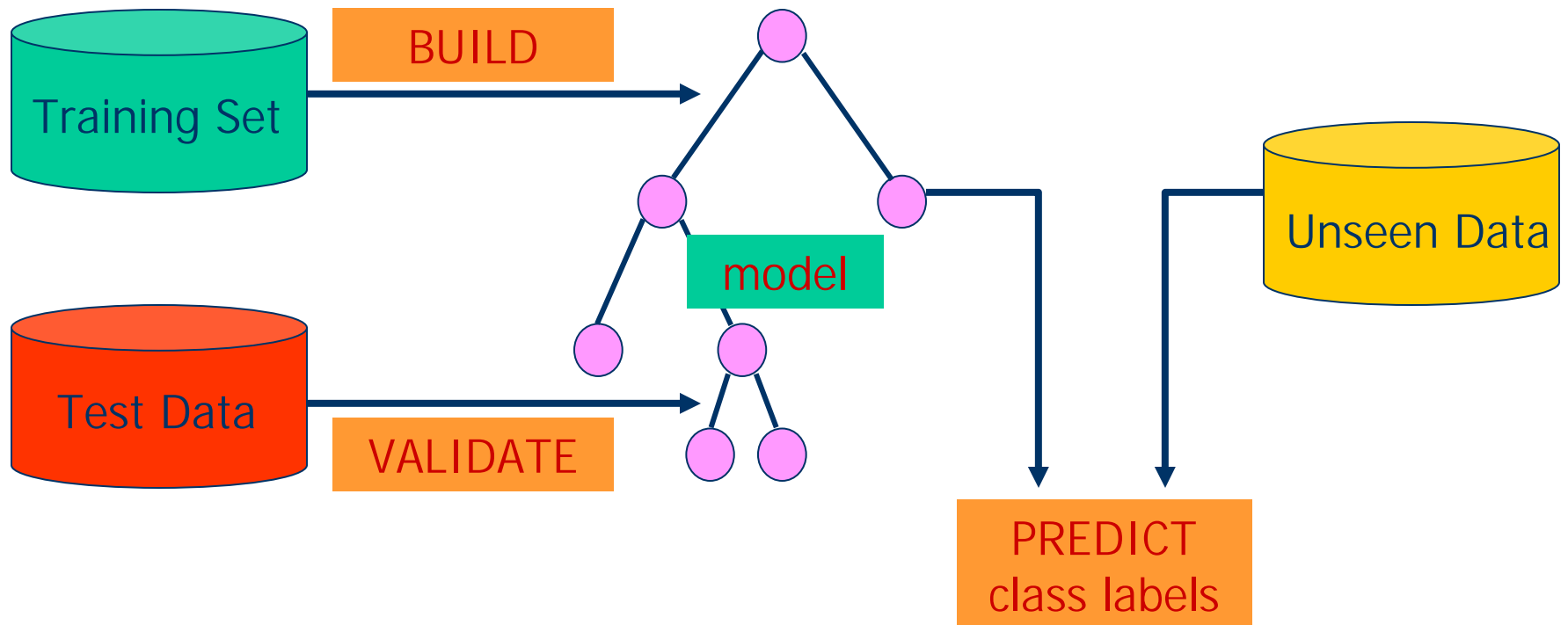
Classification

Applications:

- Classification of (potential) customers for: Credit approval, risk prediction, selective marketing
- Performance prediction based on selected indicators
- Medical diagnosis based on symptoms or reactions to therapy

Classification

Classification process



Classification

Observations:

- **Features** can be either **categorical** if belonging to unordered domains (e.g., Car Type), or **continuous**, if belonging to ordered domains (e.g., Age)
- The class could be regarded as an additional attribute of the examples, which we want to predict
- **Classification vs Regression:**

Classification: builds models for categorical classes

Regression: builds models for continuous classes

- Several **types of models** exist: decision trees, neural networks, bayesian (statistical) classifiers.

Classification using decision trees

Definition: Decision Tree for a training set T

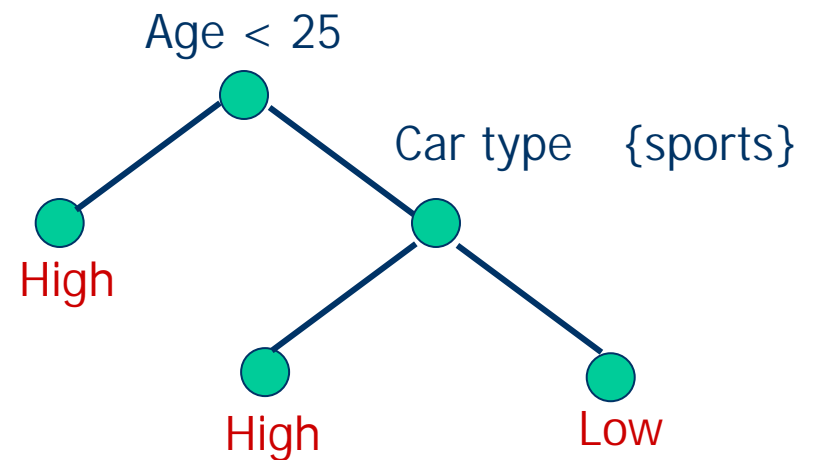
- Labeled tree
- Each **internal node v** represents a test on a feature. The edges from v to its children are labeled with mutually exclusive results of the test
- Each **leaf w** represents the subset of examples of T whose features values are consistent with the test results found along the path from the root to w . **The leaf is labeled with the majority class of the examples it contains**

Classification

Example:

examples = car insurance applicants, class = insurance risk

features		class
Age	Car Type	Risk
23	family	high
18	sports	high
43	sports	high
68	family	low
32	truck	low
20	family	high



Model
(decision tree)

Data Mining Problems

Clustering: grouping objects into classes with the objective of maximizing intra-class similarity and minimizing inter-class similarity (**unsupervised learning**)

Statement of the Problem

GIVEN: N objects, each characterized by p attributes (a.k.a. variables)

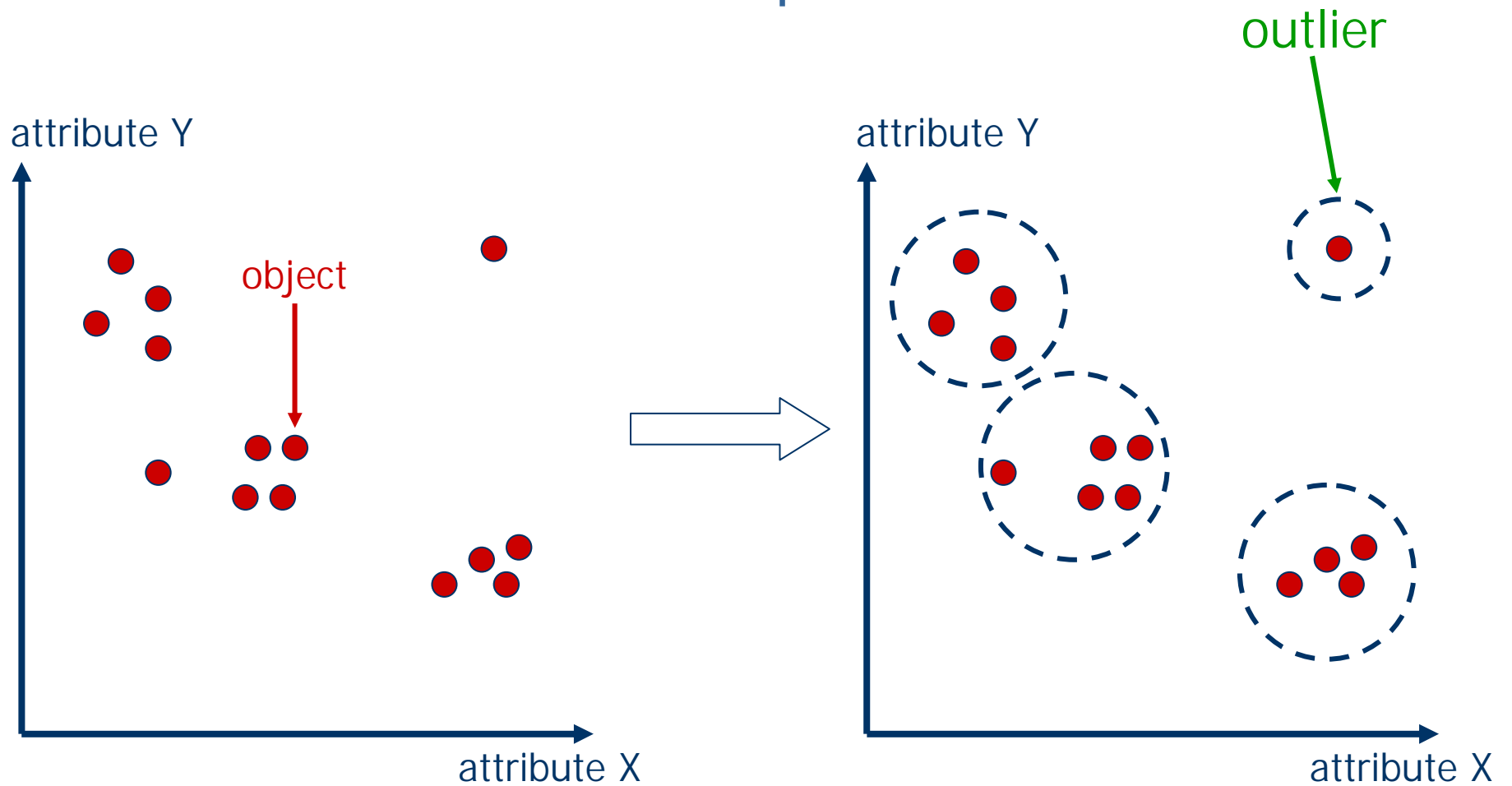
GROUP: the objects into K clusters featuring

- High intra-cluster *similarity*
- Low inter-cluster *similarity*

Remark: Clustering is an instance of unsupervised learning or learning by observations, as opposed to supervised learning or learning by examples (classification)

Clustering

Example



Clustering

Several types of clustering problems exist depending on the specific **input/output requirements**, and on the notion of **similarity**:

- The number of clusters K may be provided in input or not
- As output, for each cluster one may want a representative object, or a set of aggregate measurements, or the complete set of objects belonging to the cluster
- **Distance-based clustering**: similarity of objects is related to some kind of geometric distance
- **Conceptual clustering**: a group of objects forms a cluster if they define a certain concept

Applications:

- **Marketing**: identify groups of customers based on their purchasing patterns
- **Biology**: categorize genes with similar functionalities
- **Image processing**: clustering of pixels
- **Web**: clustering of documents in meta-search engines
- **GIS**: identification of areas of similar land use

Challenges:

- **Scalability**: strategies to deal with very large datasets
- **Variety of attribute types**: defining a good notion of similarity is hard in the presence of different types of attributes and/or different scales of values
- **Variety of cluster shapes**: common distance measures provide only spherical clusters
- **Noisy data**: outliers may affect the quality of the clustering
- **Sensitivity to input ordering**: the ordering of the input data should not affect the output (or its quality)

Main Distance-Based Clustering Methods

Partitioning Methods: create an initial **partition** of the objects into K clusters, and refine the clustering using iterative relocation techniques. A cluster is represented either by the **mean** value of its component objects or by a centrally located component object (**medoid**)

Hierarchical Methods: start with all objects belonging to distinct clusters and then successively merge the pair of closest clusters/objects into one single cluster (**agglomerative approach**); or start with all objects belonging to one cluster and successively split up a cluster into smaller ones (**divisive approach**)