POLITECNICO
MILANO 1863

Data Warehouse design

Cinzia Cappiello
A.A. 2023-2024

The problem

Relational DBs have the following problems:

Complexity o f the applications

High response time for answering to complex queries

Consequences

Raw data are used at the operations level

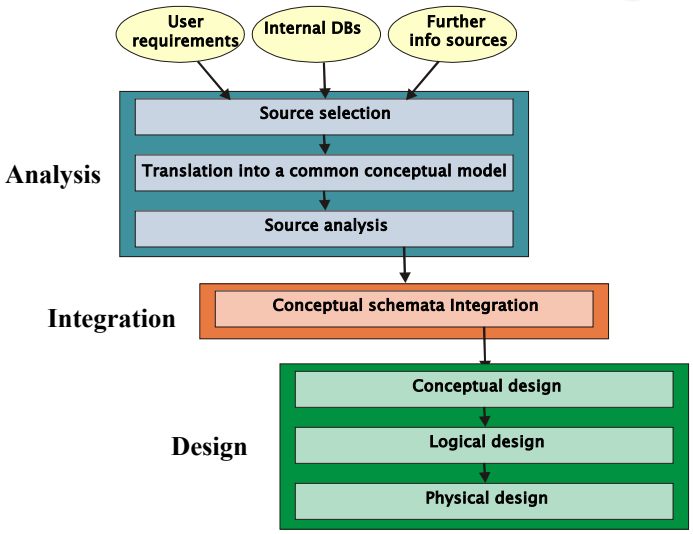
Raw data are scarcely used at the strategic level

Data Warehouse design

- The design of a data warehouse is different from the design of a traditional db
  - Data have different characteristics
  - Design is based on the available data sources
  - Design is driven by different criteria
- The design of a data warehouse aims to maintain a low number of entities but high coverage

3

Data Warehouse Design

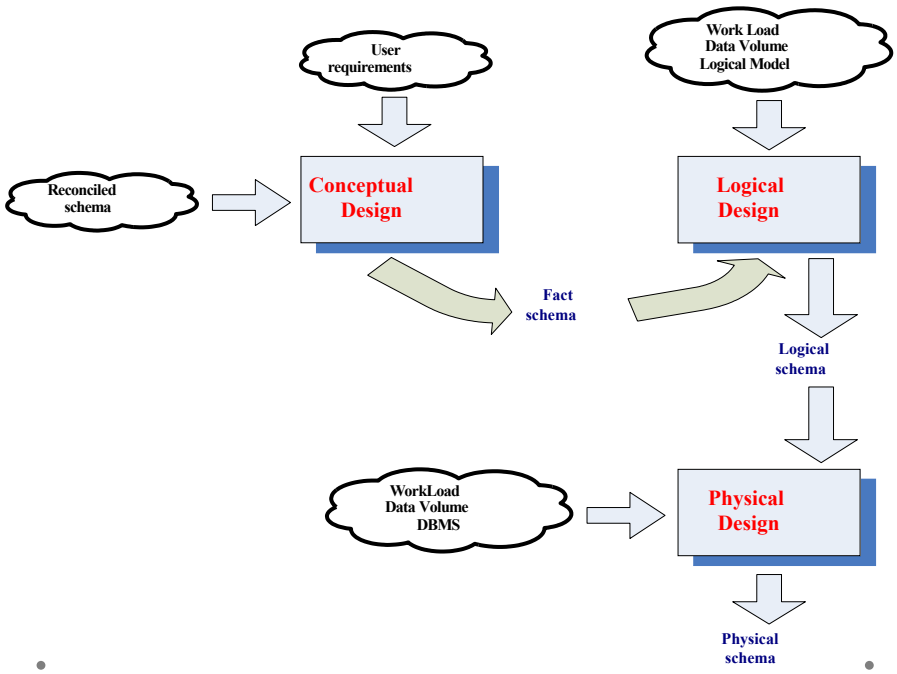


4

Data Warehouse Design

- Data Warehouses are based on the **multidimensional model**
- A standard conceptual model for DW does not exist
- The Entity/Relationship model cannot be used in the DW conceptual design

5



6

Requirements elicitation

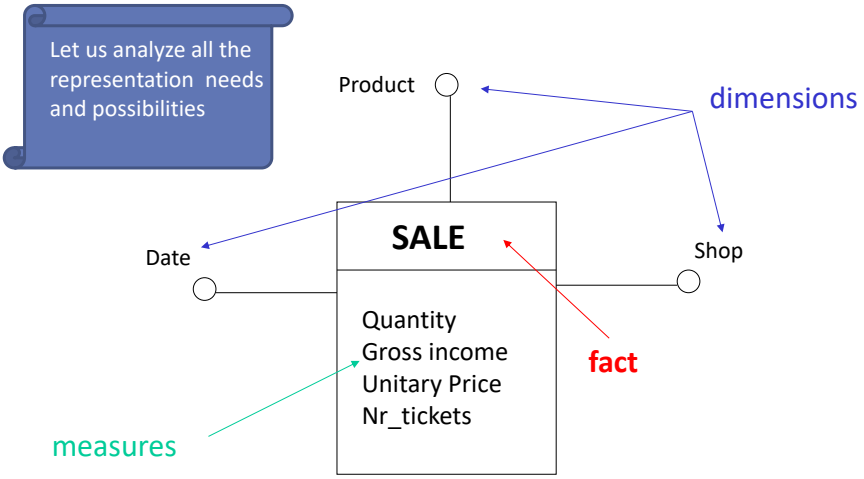
- In order to select facts it is important to understand which are the users requirements
- Requirements elicitation is conducted by interviewing the people that have to perform the analysis

7

Conceptual Model

8

Fact Schema



9

From E/R to Dimensional Fact Model (DFM)

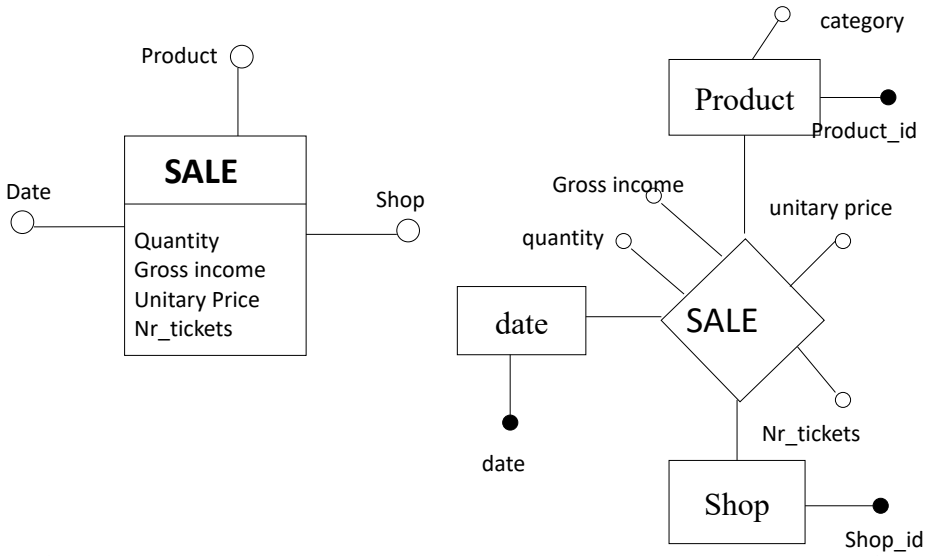
- A fact describes an entity or an N to M relationship among its dimensions. Entities that are often updated (e.g., sales) are good candidate for being transformed in facts.
- The fact value must uniquely determine the value of each dimension, e.g. a sale uniquely determines the day in which it has been done. This is represented as

  sale → day, month, year

- Naming convention: the dimensions of a same fact schema must have distinct names

10

DFM and E/R

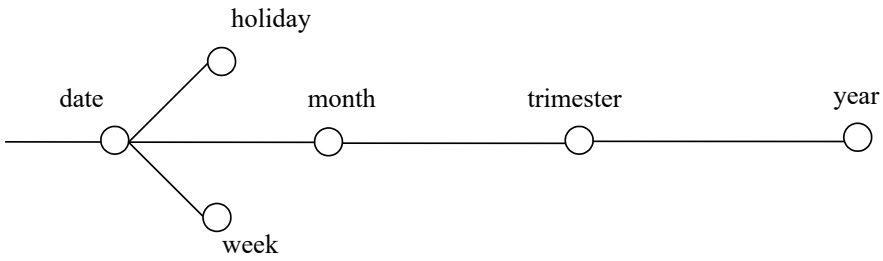


Dimensional attribute

- A dimensional attribute must assume discrete values, so that it can contribute to represent a dimension
- Dimensional attributes can be organized into hierarchies

Hierarchy

- A dimensional hierarchy is a directional tree where
  - Nodes are dimensional attributes
  - Edges describe n:1 associations between pairs of dimensional attributes
  - Root is the considered dimension



13

Events and aggregations

- A primary event is an occurrence of a fact; it is represented by means of a tuple of values
  - ✓ On 10/10/2001, ten 'Brillo' detergent packets were sold at the BigShop for a total amount of 25 euros

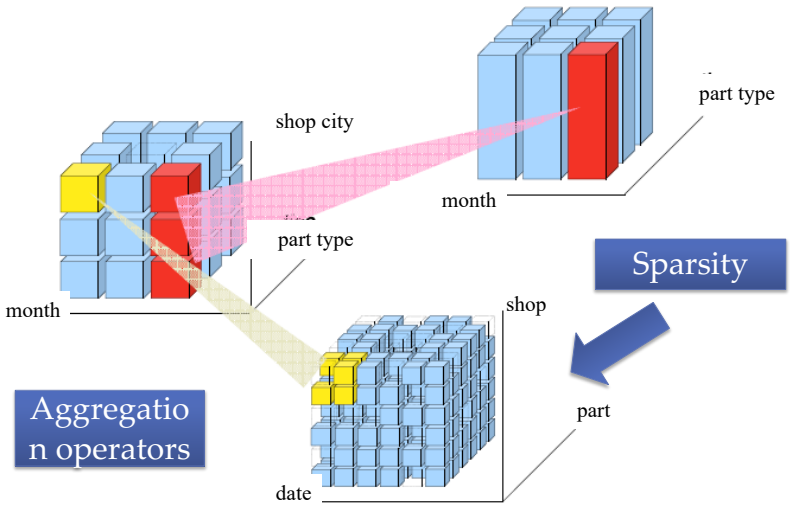
14

Events and aggregations (2)

- A hierarchy describes how it is possible to group and select primary events
- The root of a hierarchy represents the finest aggregation granularity present in the warehouse (e.g.sales one by one, or by day, or by week, depending on what the designer deems appropriate)

15

Events and aggregations



16

Events and aggregations (3)

- Given a set of dimensional attributes (pattern), each tuple of their values identifies a secondary event that aggregates (all) the corresponding primary events
- For each dimensional attribute, a value is associated with the secondary event; this value summarizes the values taken by the corresponding measure in the primary events
- For example the sales can be grouped by Product and Month:
  - ✓ in October 2001, 230 'Brillo' detergent packets were sold at the BigShop for a total amount of 575 euros

17

Secondary event

- The sales can be further grouped by Product, Month, and City
- If we consider city, product and month as dimensional attributes, the tuple

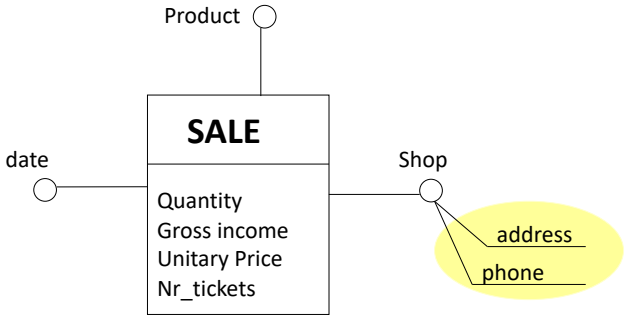
  (city: 'Rome', product: 'Brillo', month: 10/2001)

  identifies another secondary event
- It aggregates all the sales related to the product 'Brillo' in shops of 'Rome' during the month October 2001

18

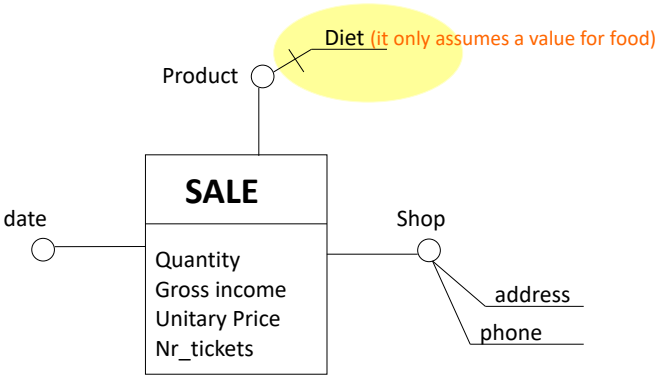
Descriptive attributes

- A descriptive attribute contains additional information about a dimensional attribute
- They are uniquely determined by the corresponding dimensional attribute
- They are relevant for analytical purposes only as selection predicates

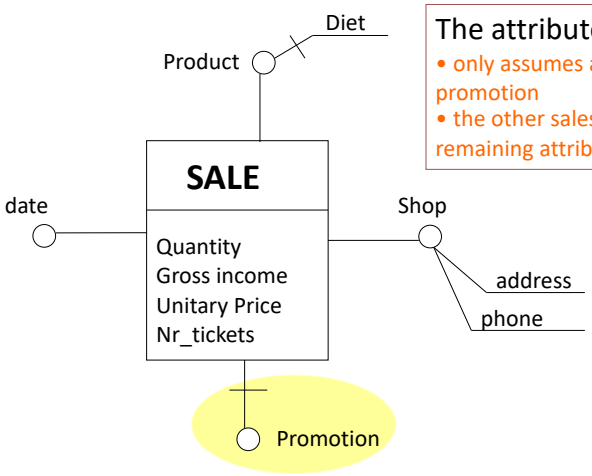


Optional edges

- Some edges of a fact schema could be optional



Optional dimensions



The attribute Promotion:
- only assumes a value for products in promotion
- the other sales are characterized by the remaining attributes

Cross-dimensional attributes

- A cross-dimensional attribute is a dimensional or a descriptive attribute whose value is obtained by combining values of some dimensional attributes
  - ✓ For example, IVA (VAT) is computed based on the product category and the state

Convergence

- It is related to the structure of a hierarchy
  - ✓ Two dimensional attributes can be connected by more than two distinct directed edges
  - ✓ For example:

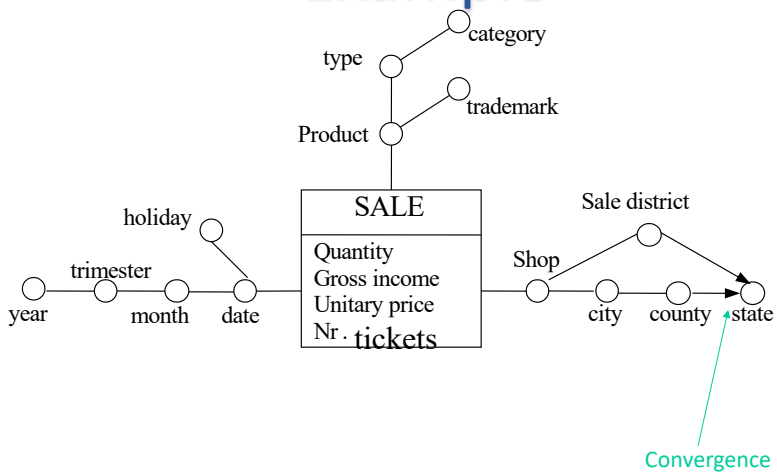
  Shop → city → county→ state

  or

  Shop→ sale district → state

23

Example

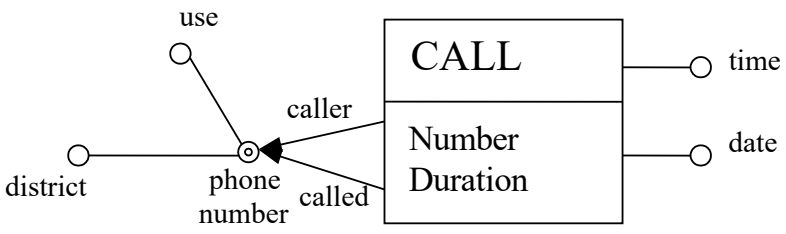


24

Hierarchy Sharing

- In a fact schema, some portions of a hierarchy might be duplicated
- As a shorthand we allow hierarchy sharing
- If the sharing starts with a dimension attribute, it is necessary to indicate the roles on the incoming edges
- Necessary condition: the unicity of the value must hold on both branches

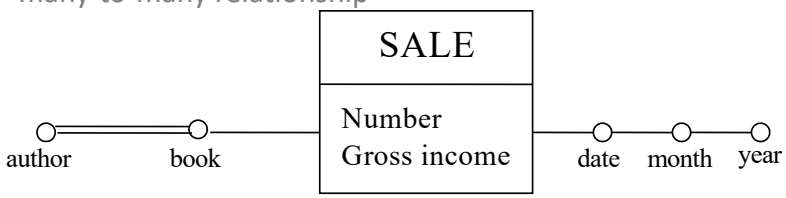
Hierarchy Sharing



It is in fact a shorthand to represent
the duplication of the whole hierarchy

Multiple edges

- Recall: the dimension values must be uniquely determined by the fact
- Some attributes, or some dimensions, may be related by a many-to-many relationship



- we denote them by multiple edges
- they are dealt with in a special way at logical design time

27

Measure Aggregation

- Aggregation requires to specify an operator to combine values related to primary events into a unique value related to a secondary event (e.g. sum of sold quantity aggregated by month)
- A measure is additive w.r.t. a given dimension iff the SUM operator is applicable to that measure along that dimension

28

Measure Classification: Additivity

- Additive measures (flow or rate measures): Can be meaningfully summarized using addition along all dimensions
  - E.g., sales amount can be summarized when the hierarchies in Store, Time, and Product dimensions are traversed
- Semiadditive measures (stock or level measures): Can be meaningfully summarized using addition along some (not all) dimensions
  - E.g., inventory quantities, can be aggregated in the Store dimension, but cannot be aggregated in the Time dimension
- Nonadditive measures (value-per-unit measures): Cannot be meaningfully summarized using addition along any dimension
  - E.g., item price, cost per unit, exchange rate

Elzbieta Malinowski & Esteban Zimányi 2008

29

29

The n.of tickets is non-additive (and in general non-aggregable) w.r.t. the product

- By n. of tickets we mean the n. of "buyings" i.e. the ticket count
- The association between product and ticket is many-to-many
- E.g. by summing up the ticket count on the product type we count the same type twice if it is the type of products that are in the same ticket

Ticket	Product	Type
S1	P1	T1
S1	P2	T1
S2	P1	T1
S2	P3	T2

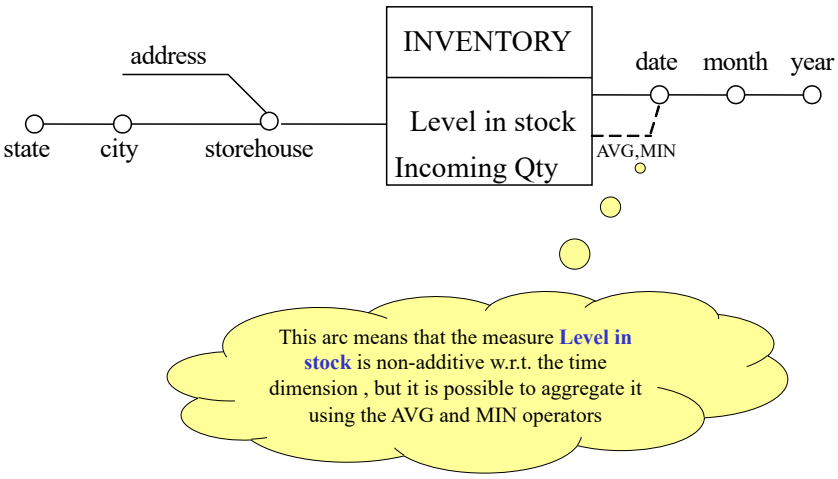
how many tickets containing p1 ? → 2
how many tickets containing p2 ? → 1
how many tickets containing p3 ? → 1
how many tickets with products of type t1 ? → 2

BUT

Sum(tickets with type(product) =t1) = 3 !!!

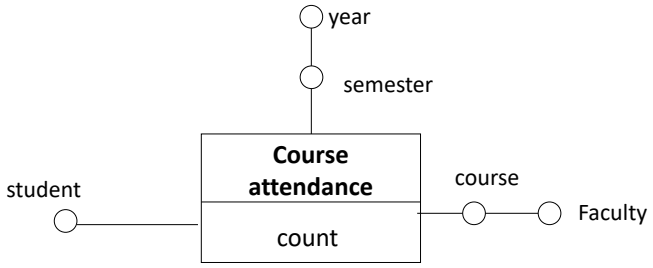
30

Aggregability



35

Empty fact schemata



A fact schema is empty if there are no measures.
In fact, the default measure is the count

37

Conceptual design

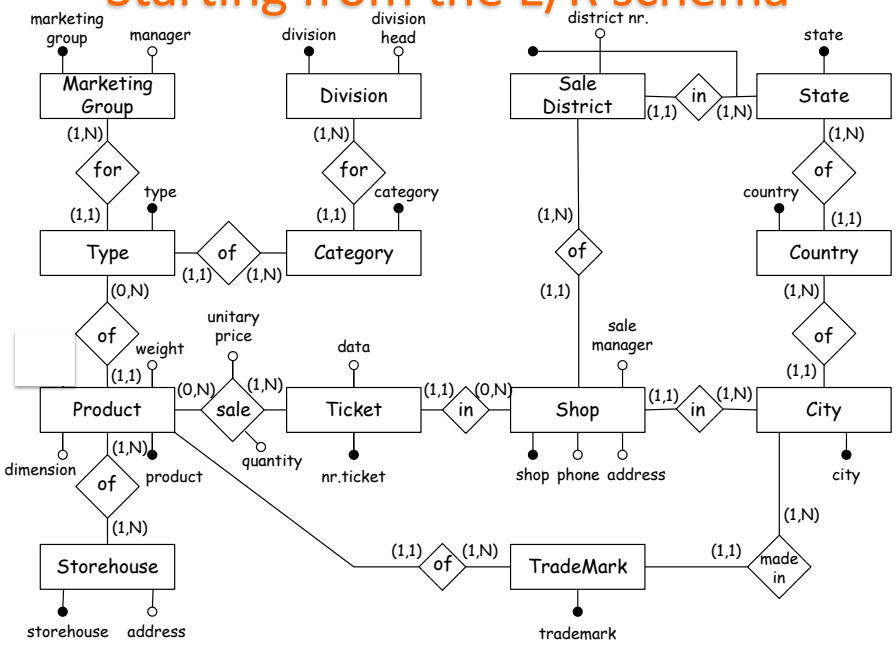
Conceptual design

- Conceptual design takes into account the documentation related to the integrated, reconciled input database
  - Conceptual schema (e.g. Entity/Relationship)
  - Logical schema (e.g. relational, XML…)

Top-down methodology

1. Fact definition (a subject oriented collection of data !!)
2. For each fact:
   1. Attribute tree definition
   2. Attribute tree editing
   3. Dimension definition
   4. Measure definition
   5. Fact schema creation

Starting from the E/R schema



Starting from the Relational Schema

Product(product,weight,dimension,trademark:TradeMark,type:Type)
Shop(shop,address,phone,salemanager,(ditrictnr,state):District,city:City)
Ticket(nrticket,date,shop:Shop)
Sale(product:Product,nrticket:Ticket,quantity,unitaryprice)
Storehouse(storehouse,address)
City(city,country:Country)
Country(country,state:State)
State(state)
District(district,state:State)
Prod_Storehouse(product:Product,storehouse:Storehouse)
TradeMark(trademark,madein:City)
Type(type,marketinggroup:MarketingGroup,category:Category)
MarketingGroup(marketinggroup,manager)
Category(category,division:Division)
Division(division,divisionhead)

Fact definition

- Facts correspond to events that dynamically happen in the organization
  - In an E/R schema, it can correspond to an entity F or to an association among *n* entities $E_1, E_2, \ldots, E_n$
  - In a relational schema, a fact corresponds to a relation (table) R

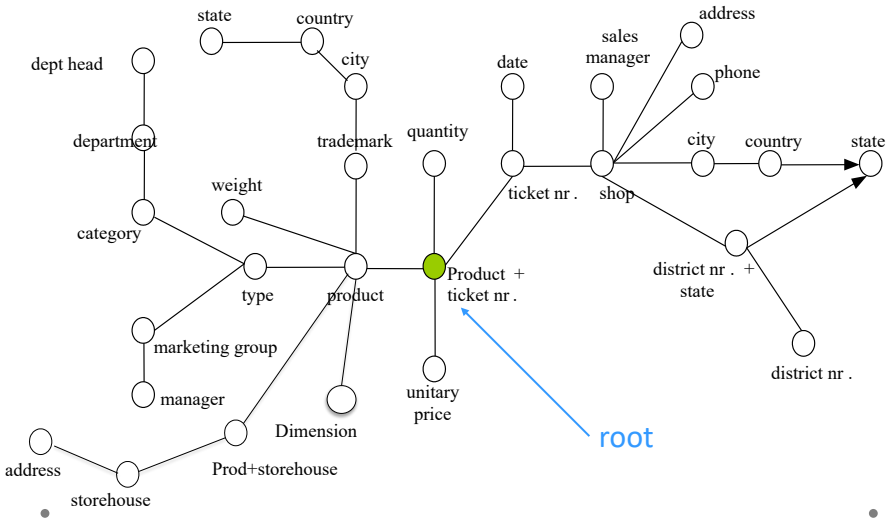
Fact definition

- Good fact candidates: entities or relationships representing **frequently updated data**
- Static archives: **NO!**
- **Remark**: when a fact is identified, it becomes the root of a new fact schema

Attribute tree definition

- The attribute tree is composed by:
  - Nodes, corresponding to attributes (simple or complex) of the source schema
  - Root, corresponding to the primary key of the fact F
  - For each node, the corresponding attribute uniquely determines its descendant attributes

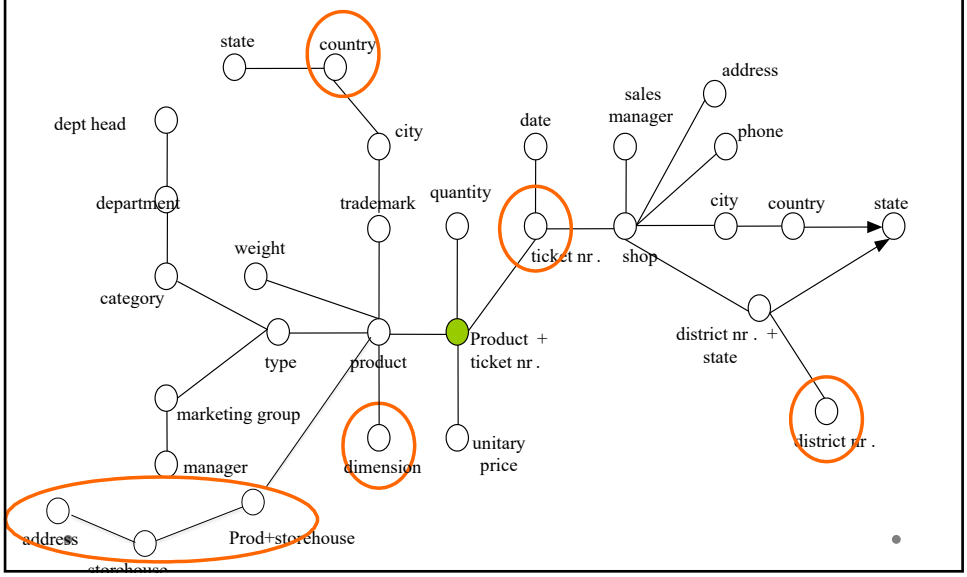
Attribute tree: example



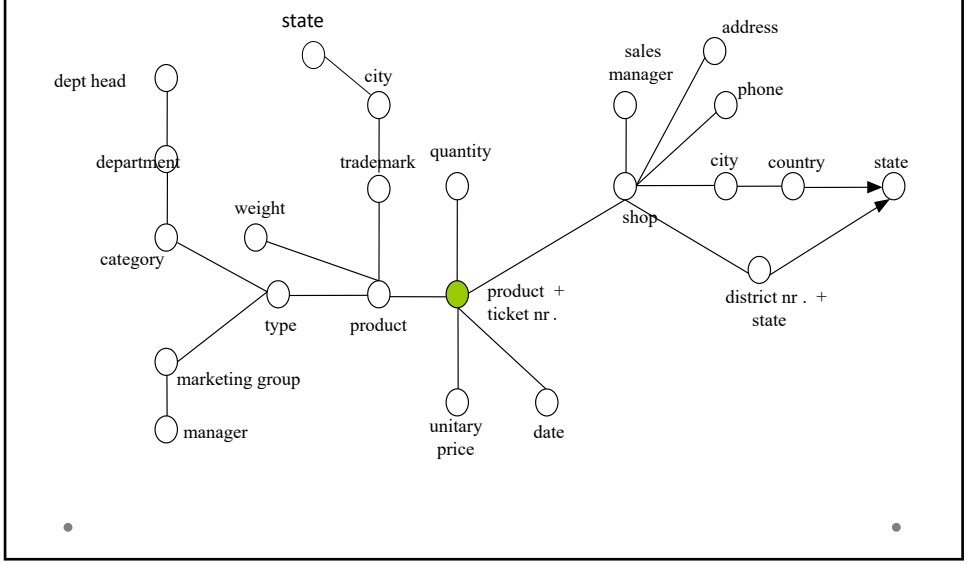
Attribute tree editing

- The editing phase allows to remove some attributes which are irrelevant for the data mart
  - Pruning of a node v: the subtree rooted in v is deleted
  - Grafting of a node v: the children of v are directly connected to the father of v

Attribute tree editing: example



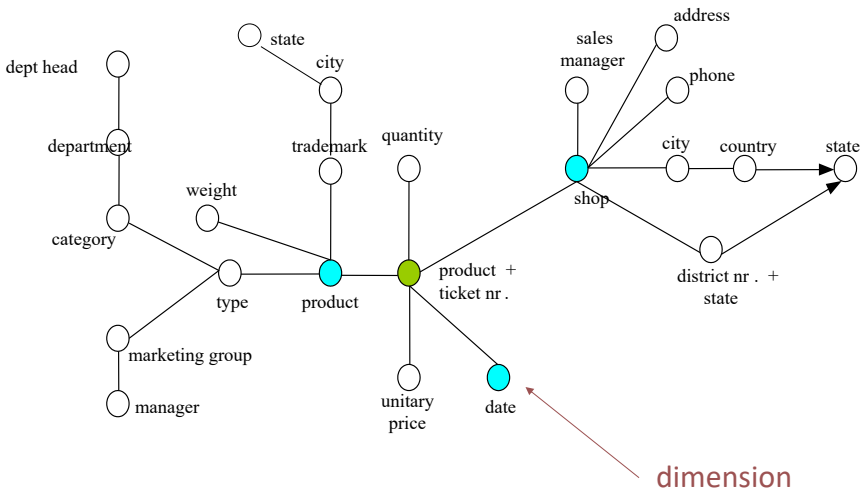
Attribute tree editing: example



Dimension definition

- Dimensions can be chosen among the children of the root
- Time should always be a dimension
  - Historical source: time is an attribute
  - Snapshot source: not always time is directly represented. In this case it is necessary to add time.

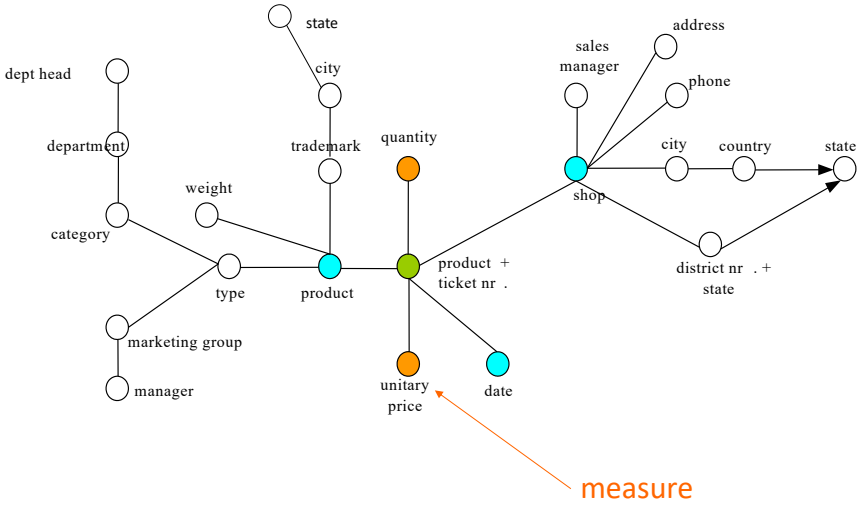
Dimensions definition: example



Measure definition

- If the fact identifier (set of attributes) is included in the set of dimensions, then numerical attributes that are children of the root (fact) are measures
- Further measures are defined by applying aggregate functions to numerical attributes of the tree
  - Generally: sum, average, min, max, count
- It is possible that a fact has no measures (empty)

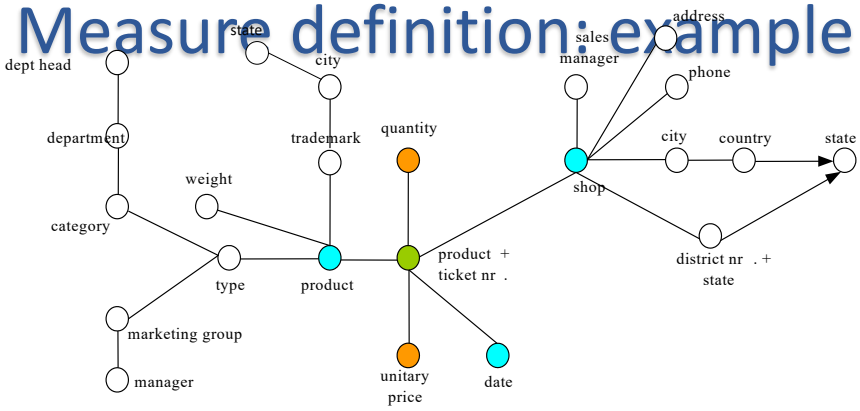
Measure definition: example



Glossary

- In the glossary, an expression is associated with each measure
  - The expression describes how we obtain the measure at the different levels of aggregation starting from the attributes of the source schema

Measure definition: example

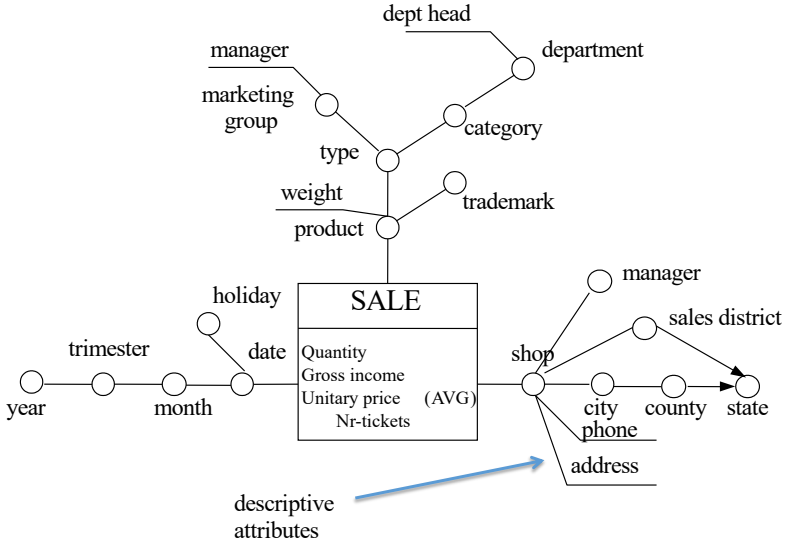


Quantity = SUM(Sale.quantity)
Gross income=SUM(Sale.quantity*Sale.unitaryprice)
Unitary price=AVG(Sale.unitaryprice)
Nr-tickets=COUNT(*)

Fact schema creation

- The attribute tree is translated into a fact schema including dimensions and measures
  - Dimension hierarchies correspond to subtrees having as roots the different dimensions (with the least granularity)
  - The fact name corresponds to the name of the selected entity

Fact schema creation: example



Exercise

- The ER schema is a portion of a database related to a video content streaming service. Starting from this DB, we want to build a DW to make decisions regarding the catalog of contents for the following season and advertising to customers.
- In particular, we want to analyze:
  - Which are the TV series that have been preferred in the last year (highest number of views); it is requested also the possibility to have details about the individual seasons or single episodes;
  - Which are the most successful series (highest number of views) for a type of customer or a geographical area