Fraunhofer Institut Autonome Intelligente Systeme

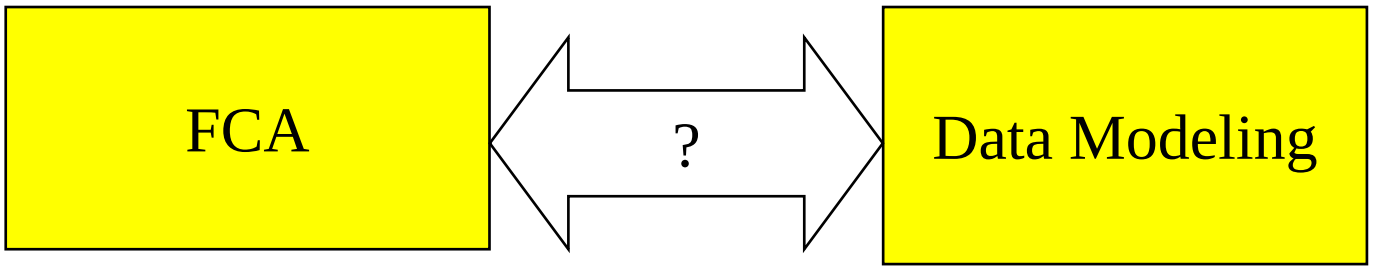
Hierarchical Multidimensional Modelling in the Concept-Oriented Data Model

Alexandr Savinov
Fraunhofer Institute for Autonomous Intelligent Systems
Knowledge Discovery Team
Germany
alexandr.savinov@ais.fraunhofer.de

Contents

Fraunhofer Institut Autonome Intelligente Systeme

- Introduction
- Logical and Physical Structure
- Dimensions and Inverse Dimensions
- Projection and De-projection
- Multidimensional Grouping and Aggregation
- Conclusion



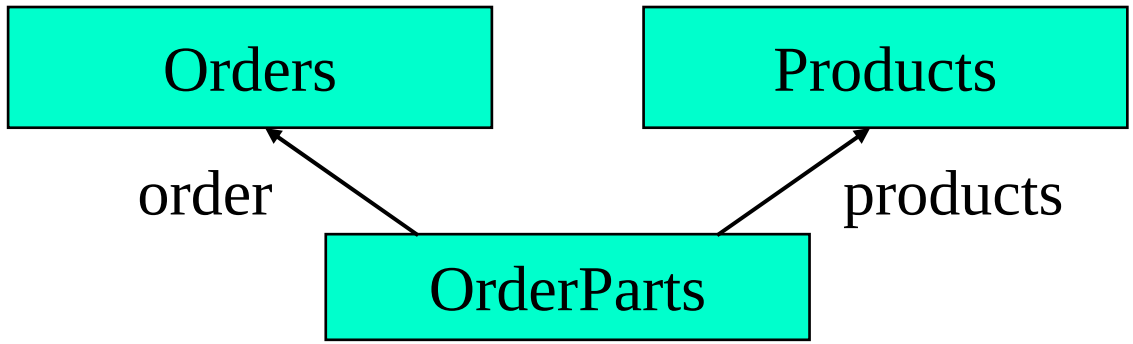
Introduction

Data models and dimensionality modelling

Fraunhofer Institut Autonome Intelligente Systeme

- Entities and relationships (ERM)
- Logic and predicates (deductive databases)
- Relations (RM)
- Facts (ORM)
- Objects (OODM)
- Dimensions (OLAP, multidimensional databases)

- Dimension is a named link between subconcept and superconcept

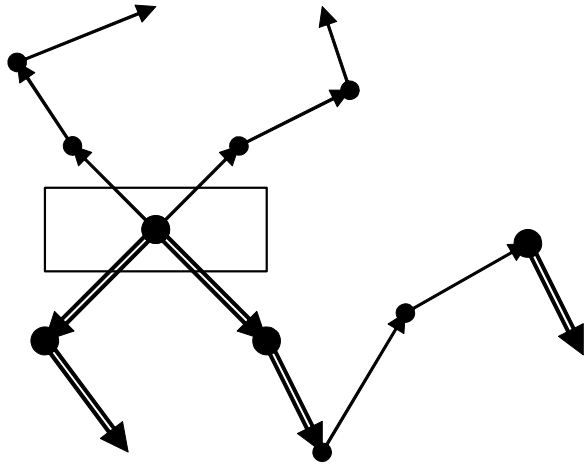


Introduction

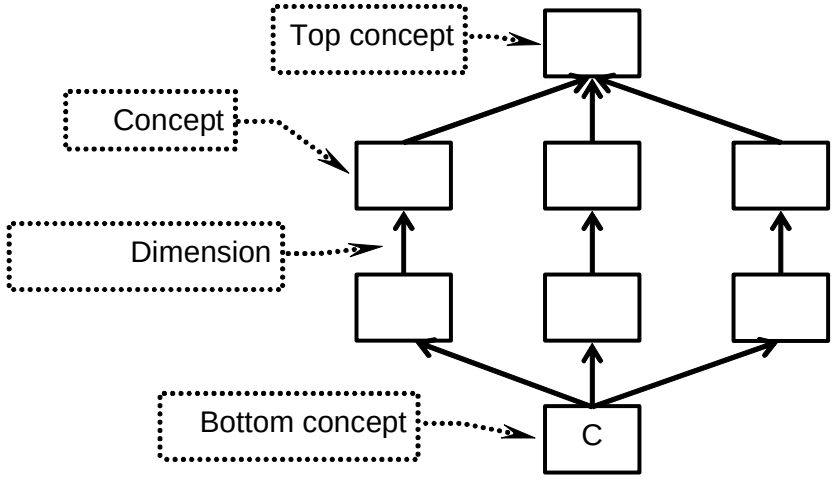
Assumptions and related work

Fraunhofer Institut Autonome Intelligente Systeme

- Global semantics (URM)



- Using the structure for navigation (FDM)
- Hierarchical structure (FCA)



- Level of details (OLAP)

Introduction

FCA

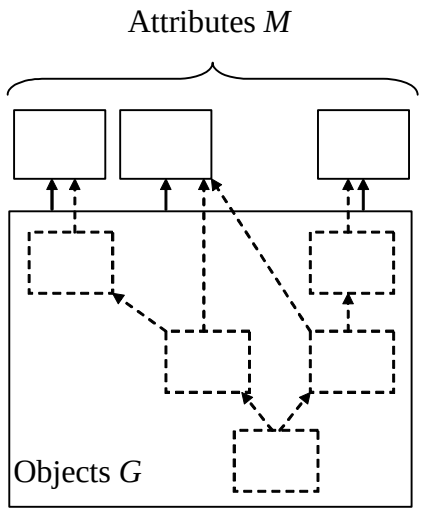
Fraunhofer Institut Autonome Intelligente Systeme

- Concept -> Concept
- Object -> Data item
- Attribute -> Primitive concept

- In FCA concepts depend on data while in COM data depends on concepts, that is concepts define a structure for data (in FCA the structure is derived from data semantics)

- Items belong to one concept while in FCA object may belong to many concepts

- COM concept is a (non-primitive) attribute for subconcepts

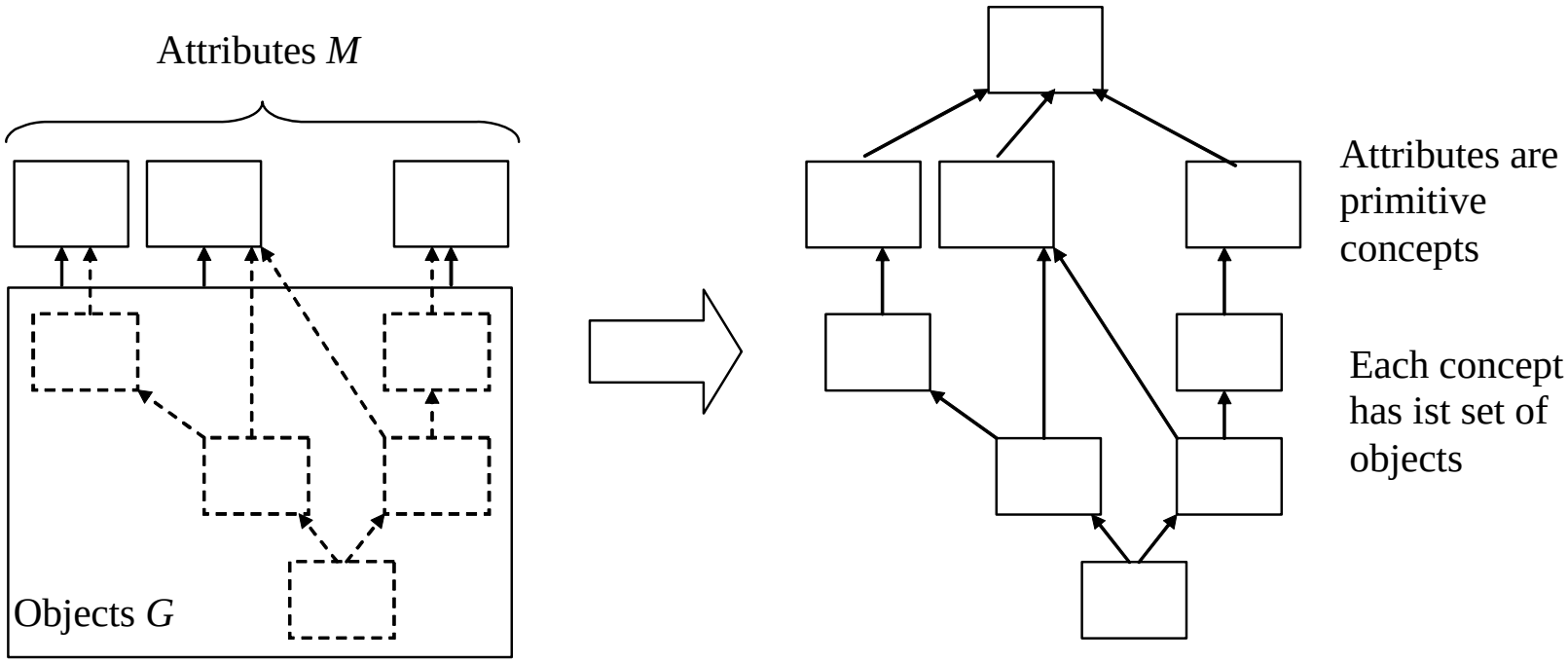


Introduction

Questions

Fraunhofer Institut Autonome Intelligente Systeme

- Why we have (primitive) attributes defined at structural level while concepts are derived from data semantics?
  Why not to have a possibility to define a (non-primitive) attribute as a concept?



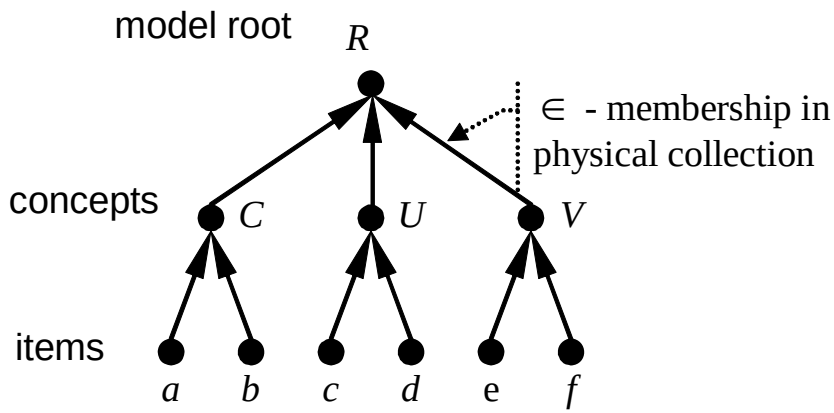
Physical and Logical Structure

Physical structure

- At physical level an element of the model is a collection of other elements
- Physical structure is used for representation and access
- Physical structure is used to implement reference
- Physical structure is hierarchical where each element has only one parent

$$C = \{a, b, \ldots\}$$

$$a \in A, b \in A, \ldots$$



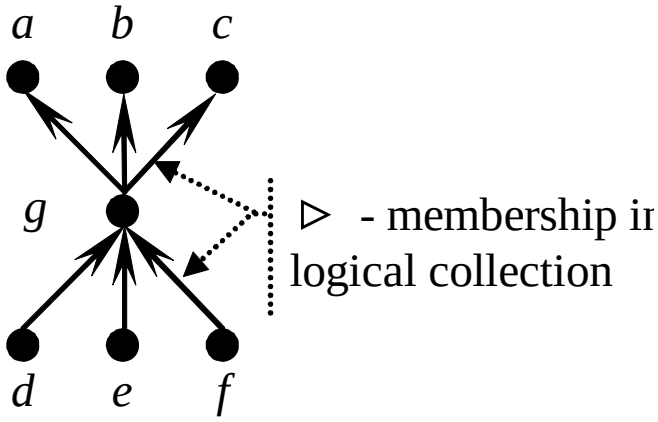
Physical and Logical Structure

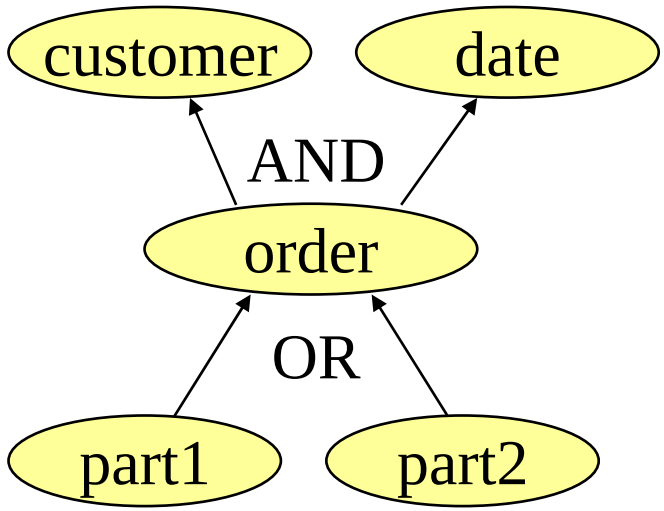
Logical structure

Fraunhofer Institut Autonome Intelligente Systeme

- Each element is a combination of other elements (by reference)
- Logical structure is used to represent data semantics (properties)
- Logical collection is a dual combination
- Each element has many parents and many children

$$g = \langle a, b, \ldots, c \rangle,\ a \triangleleft g, b \triangleleft g, \ldots, c \triangleleft g$$





$$g = \{d, e, \ldots, f\},\ d \triangleright g, e \triangleright g, \ldots, f \triangleright g$$

Physical and Logical Structure

Two level model

Fraunhofer Institut Autonome Intelligente Systeme

- **[Root]** One root element *R* is a physical collection of concepts, $R = \{C_1, C_2, \ldots, C_N\}$
- **[Syntax]** Each concept is
  - (i) a combination of other concepts called *superconcepts* (while this concept is a *subconcept*), $C = \langle C_1, C_2, \ldots, C_n \rangle \in R$
  - (ii) a physical collection of *data items* (or concept instances), $C = \{i_1, i_2, \ldots\} \in R$
- **[Semantics]** Each data item is
  - (i) a combination of other data items called *superitems* (while this item is a *subitem*), $i = \langle i_1, i_2, \ldots, i_n \rangle \in C$
  - (ii) empty physical collection, $i = \{\}$

Physical and Logical Structure

Two level model

Fraunhofer Institut Autonome Intelligente Systeme

- **[Special elements]** If a concept does not have a superconcept then it is referred to as *primitive* and its superconcept is one common *top concept*; and if a concept does not have a subconcept then it is assumed to be one common *bottom concept*, and an absence of superitem is denoted by one special *null item*.

- **[Cycles]** Cycles in subconcept-superconcept relation and subitem-superitem relation are not allowed,

- **[Syntactic constraints]** Each data item from a concept may combine only items from its superconcepts.

Syntax and Semantics

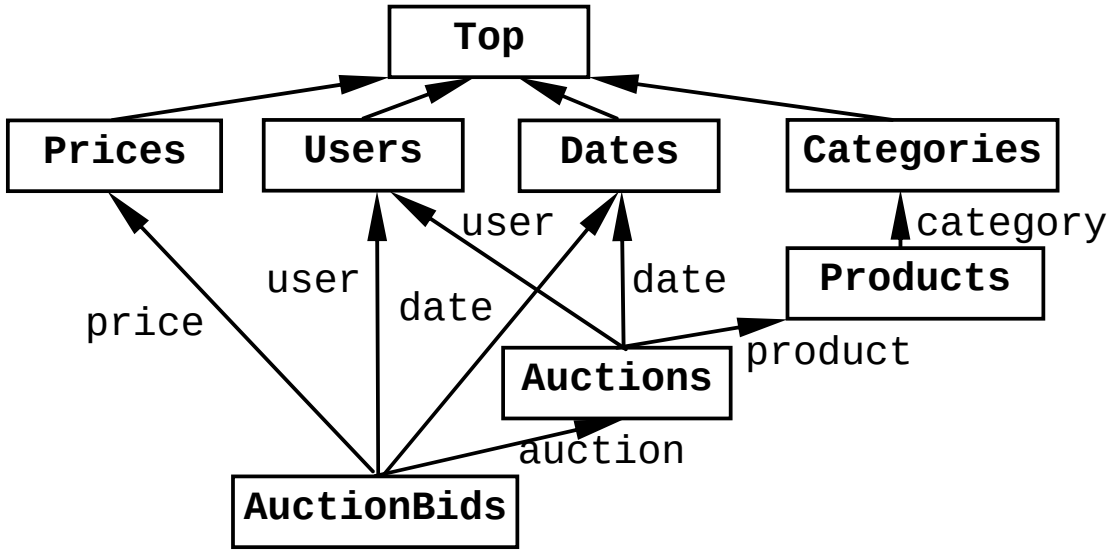
Model syntax

Fraunhofer Institut Autonome Intelligente Systeme

- At syntactic level a concept is a combination of ist superconcepts

$$C = \langle x_1 : C_1, x_2 : C_2, \ldots, x_n : C_n \rangle$$

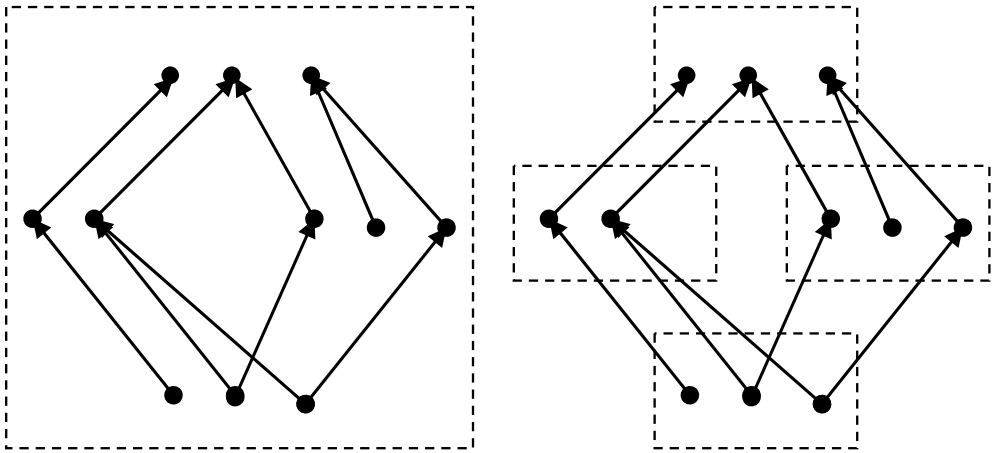
- Each superconcept is identified by dimension name, that is, dimension is a relative position of superconcept



Syntax and Semantics

Model semantics

- Each concept is a set of items: $C = \{i_1, i_2, \ldots\}$
- An item is a combination of its superitems: $i = \langle i_1, i_2, \ldots, i_n \rangle$
- There is no difference between objects and attribute values: an object has values in other objects, and it is a value for other objects



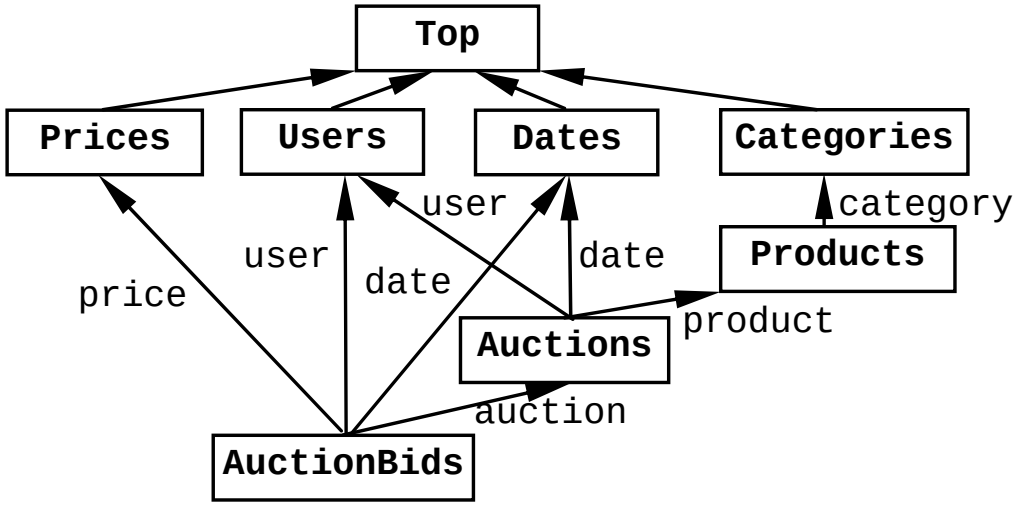
a) b)

Model Dimensionality

Dimensions

Fraunhofer Institut Autonome Intelligente Systeme

- Dimension is a named position of superconcept
- Superconcept is referred to as the domain
- Dimensions of higher rank consists of many (local) dimensions
- Dimension with the domain in a primitive concept is a *primitive dimension*
- The number of primitive dimensions is the model *primitive dimensionality*

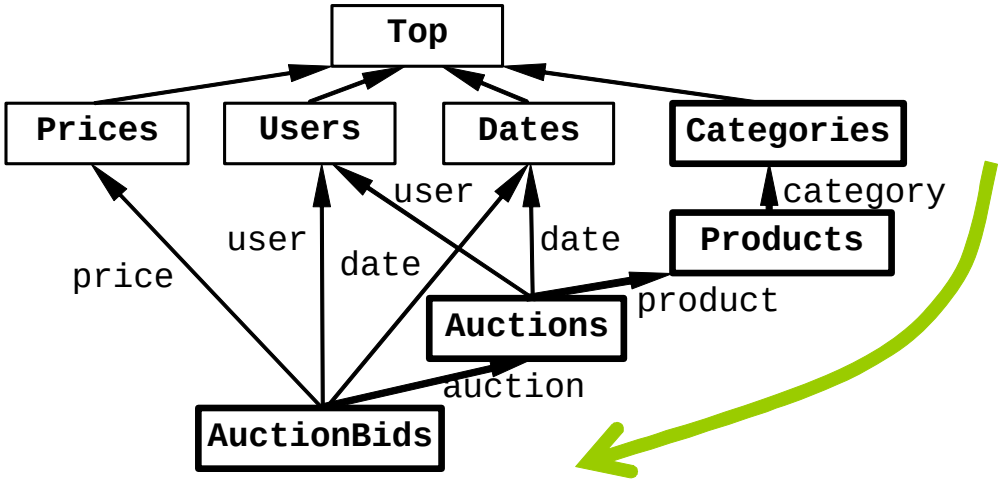


Model Dimensionality

Inverse dimensions

Fraunhofer Institut Autonome Intelligente Systeme

- Inverse dimension has an opposite direction
- Inverse dimension identifies a subconcept
- Inverse dimensions are multi-valued (while dimensions are one-valued)
- The number of primtive dimensions is equal to the number of primtive inverse dimensions
- {AuctionBids.auction.product.category}

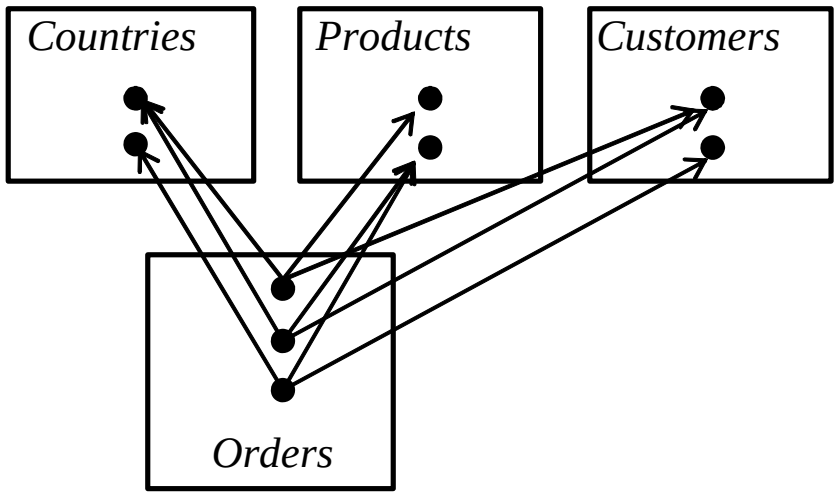


Model Dimensionality

Logical collections

Fraunhofer Institut Autonome Intelligente Systeme

- A concept is a logical collection of its subconcepts
- An item is logical collection of its subitems
- An item is group for its subitems

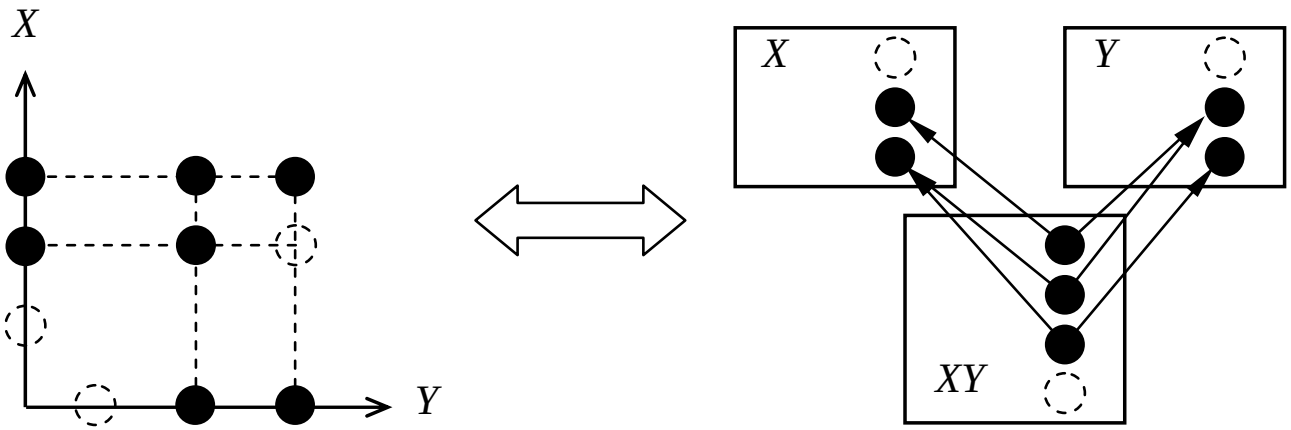


Model Dimensionality

Hierarchical coordinate system

Fraunhofer Institut Autonome Intelligente Systeme

- A concept can be interpreted as an axis with items as coordinates
- A coordinate has its own coordinates and points can be used as coordinates for other points



Projection and De-projection

Projection

Fraunhofer Institut Autonome Intelligente Systeme

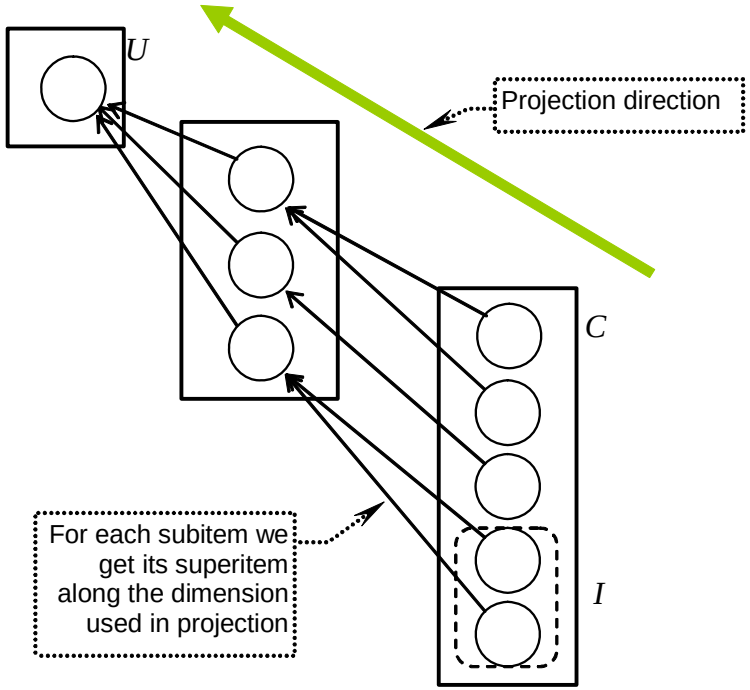
- Projection of a subset of subitems along some dimension path:

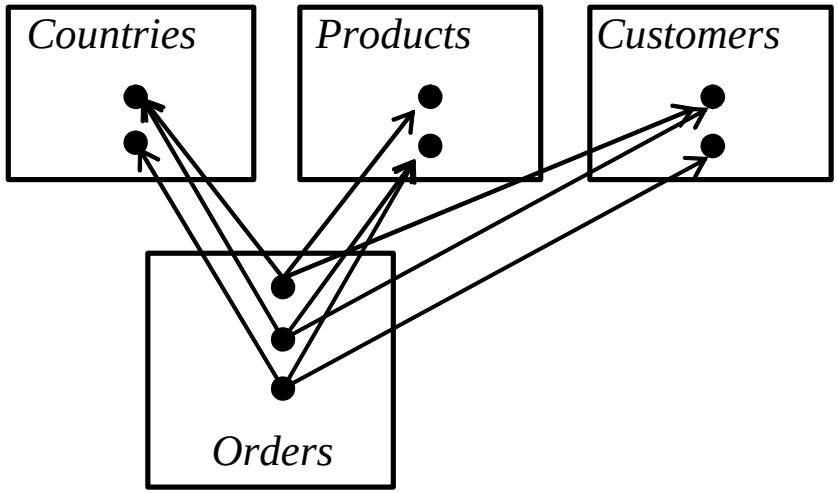
$$I \to d = \{u \in U \mid i.d = u, i \in I \subseteq C\}$$

$$I \subseteq C$$

$$d = d^1.d^2.\cdots.d^k$$

$$U = \mathrm{Dom}(d)$$





Projection and De-projection

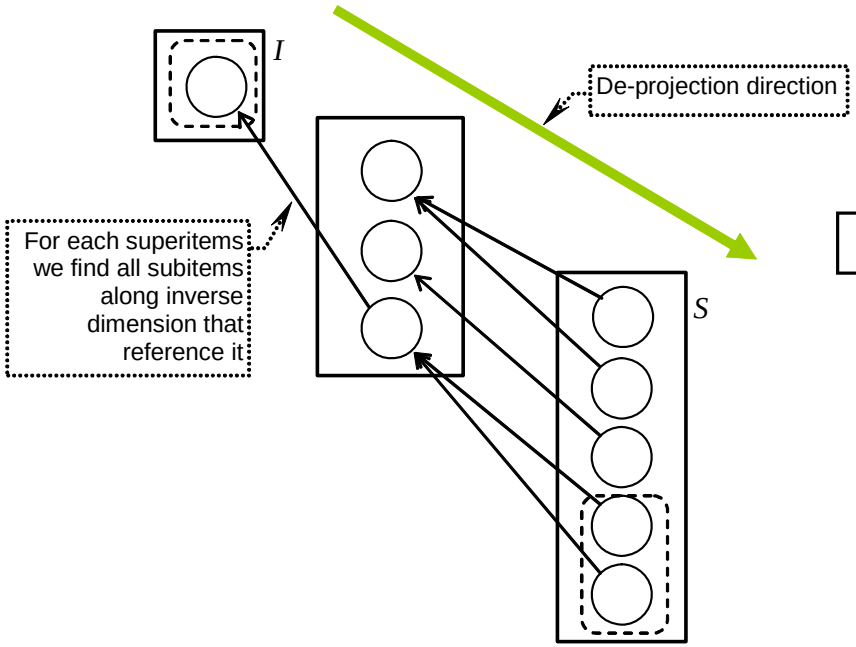
De-projection

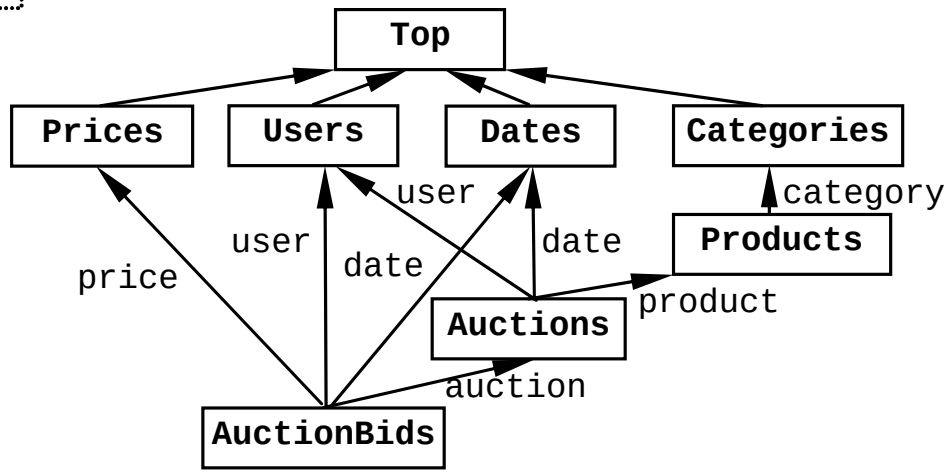
Fraunhofer Institut Autonome Intelligente Systeme

- De-projection of a subset of superitems along some inverse dimension:

$$I \to \{d\} = \{s \in S \mid s.d = i, i \in I \subseteq C\} \qquad \{d\} = \{d^1.d^2.\cdots.d^k\}$$

$$S = \mathrm{Dom}(\{d\})$$



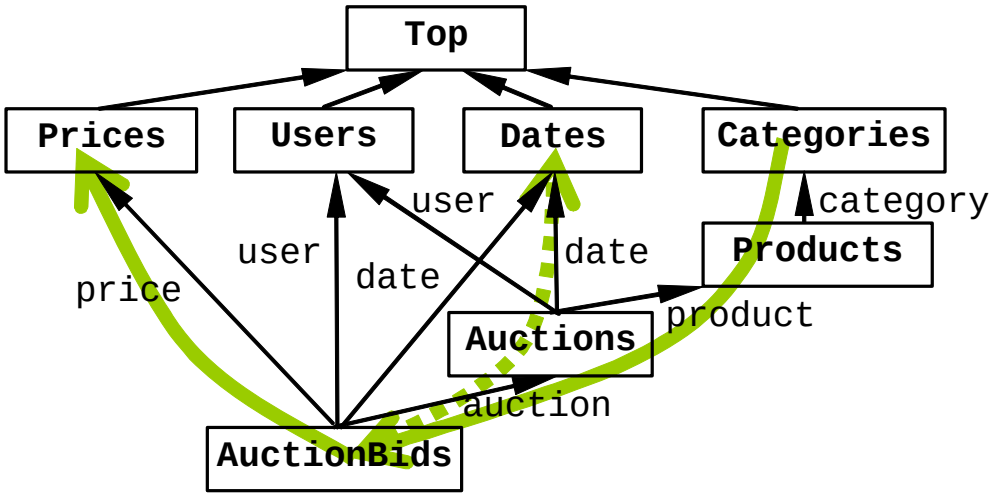


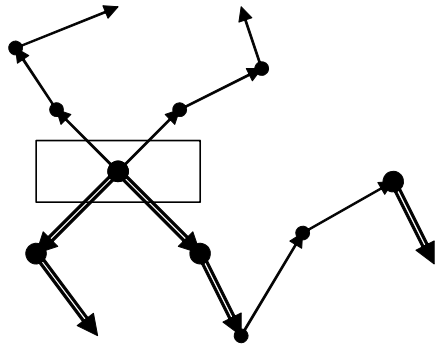
Projection and De-projection

Access path

Fraunhofer Institut Autonome Intelligente Systeme

- Access path is a sequence of projections and de-projections possibly with constraints
- Derived property is a named definition of an access path or a query
- ```
  Category.meanPriceForTenDays = avg(
    {ab in AuctionBids.auction.product.category |
    ab.auction.date > today-10 }.price
    );
  ```





- Navigational approach with no hierarchical structure:
  - OODB
  - FDM
  - Network model
  ```

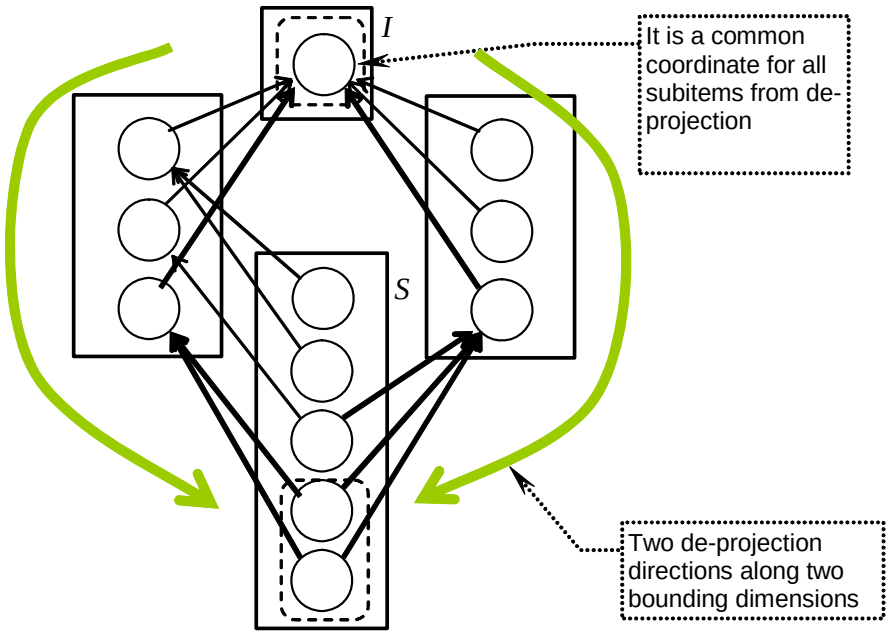
Grouping and Aggregation

Multidimensional de-projection

Fraunhofer Institut Autonome Intelligente Systeme

- More than one bounding dimension
- Multidimensional de-projection returns a set of subitems referencing source items along all bounding dimensions:

$$I \rightarrow \{d_1, d_2, \ldots, d_n\} = \{s \in S \mid s.d_1 = i \wedge s.d_2 = i \wedge \ldots \wedge s.d_n = i, i \in I \subseteq C\}$$



Grouping and Aggregation

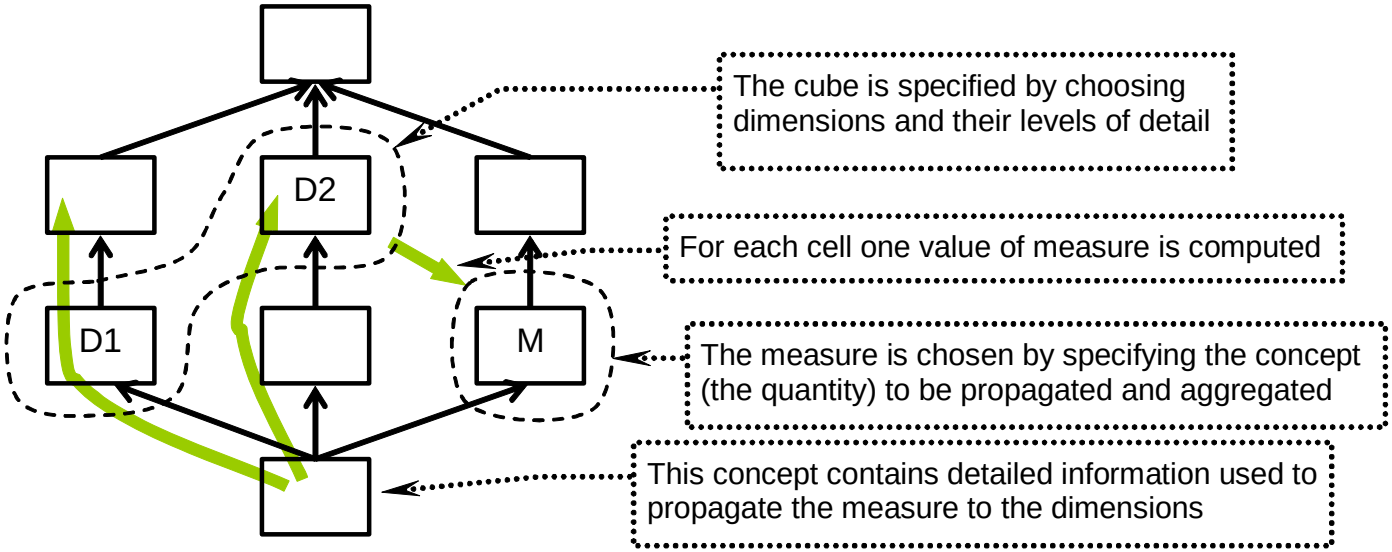
Aggregation

Fraunhofer Institut Autonome Intelligente Systeme

- A dimension hierarchy is one dimension path
- Along each hierarchy we choose a concept called a level
- Universe of discourse is the Cartesian product of the chosen levels

$$\Omega_L = D_1 \times D_2 \times \ldots \times D_n = \{\omega = \langle \omega_1, \omega_2, \ldots, \omega_n \rangle \mid \omega_j \in D_j\}$$

- For each point from UoD we find de-projection
- De-projection is aggregated

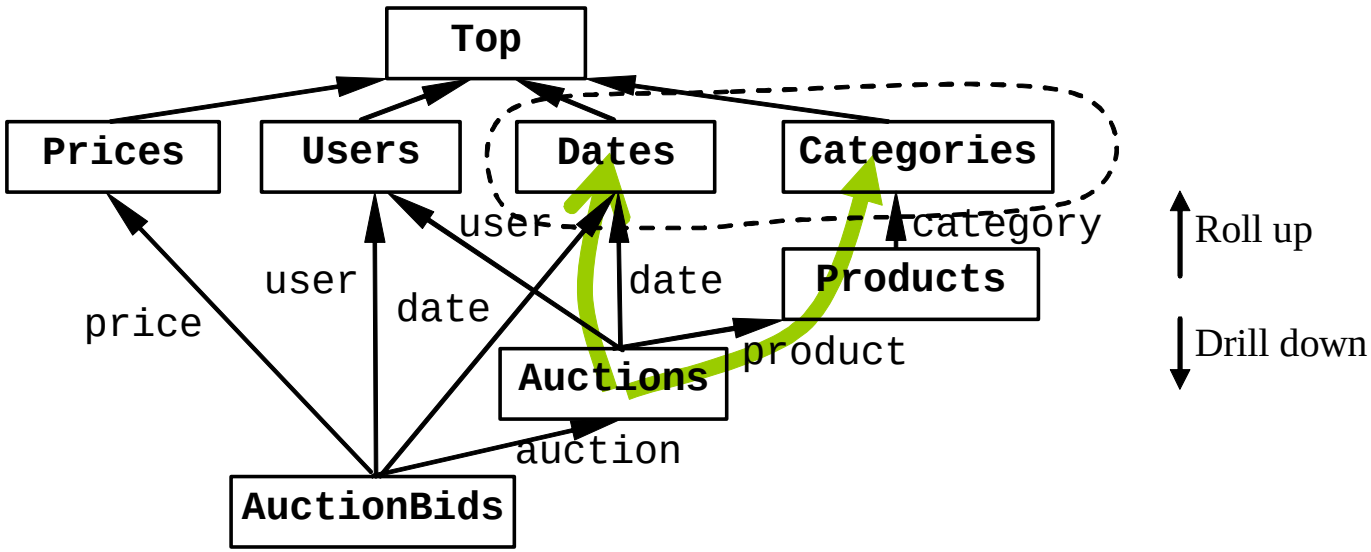


Grouping and Aggregation

Example

Fraunhofer Institut Autonome Intelligente Systeme

- {d : Dates, c : Categories | isLastWeek(d) }< avg(
  this->{Auctions.date, Auctions.product.category}.maxBid
) as averagePrice >



Conclusions

Fraunhofer Institut Autonome Intelligente Systeme

- Features:
  - Global semantics
  - Hierarchical multidimensional logical structure
  - Navigation via access paths, dimensions and inverse dimensions
  - Multidimensional aggregation and analysis
  - Concept transformations (not described in this presentation)
  - Constraint propagation and inference (not described in this presentation)
- Advantages:
  - Clarity of operations
  - Easiness of use
  - Formal syntax and semantics
  - Simple query language (no joins)