

A Conceptual Modelling Perspective for Data Warehouses

Jaroslav Pokorný
Charles University
Praha,
Czech Republic

Peter Sokolowsky
Charles University
Praha, Czech Republic
& IKS
Saarbrücken, Germany



Outline

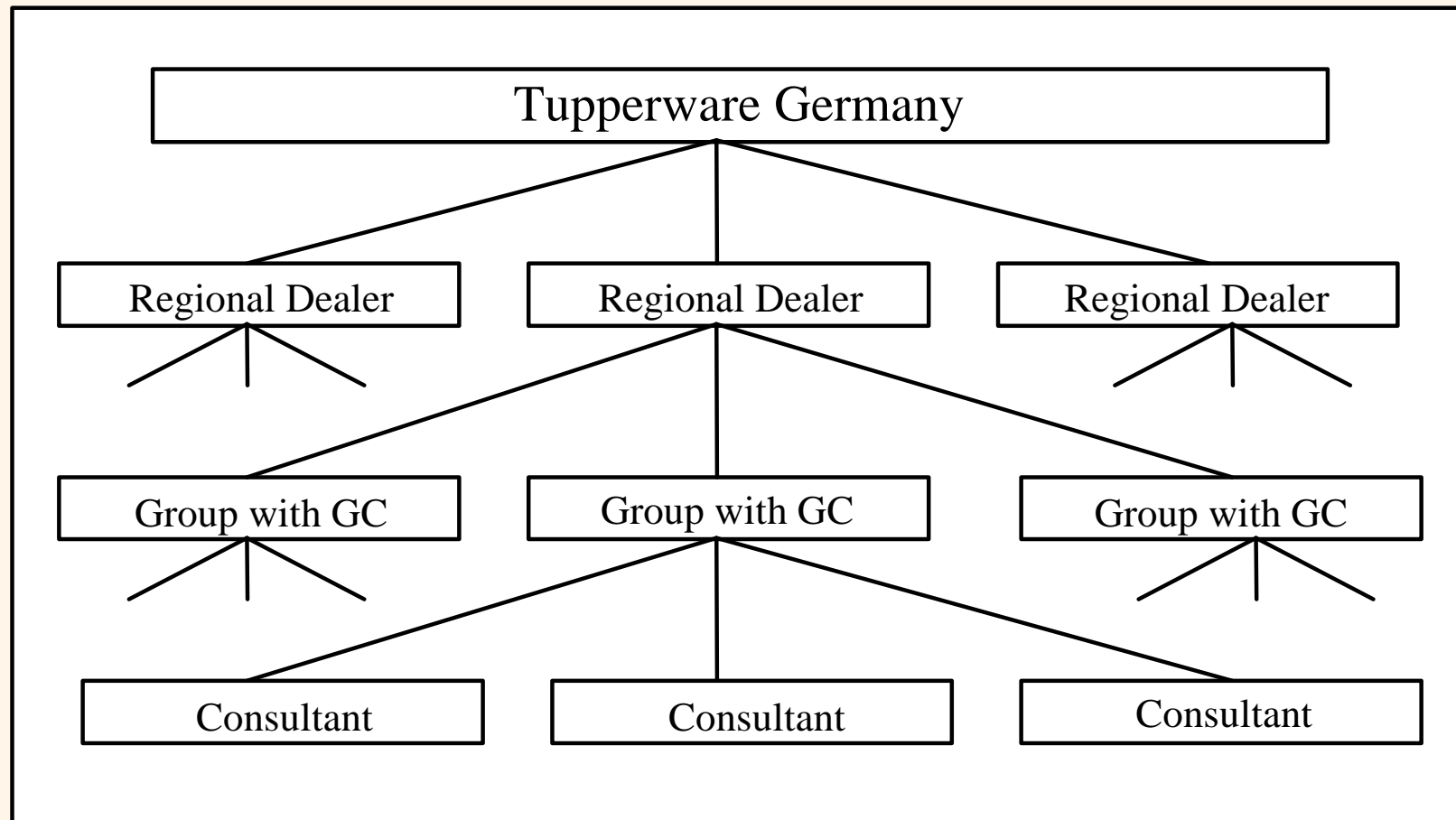
1. Introduction - DW, OLAP
2. An Example
3. Functionalities of OLAP
4. OLAP Database Design
5. Multidimensional modelling
6. Conclusions



1. Introduction

- DW (Data Warehouse) denotes a database architecture used for a maintenance of historic data which are obtained for one or more operational databases. Typically, these data are cleaned and restructured to support queries, summaries, and analyses (Creative Data, Inc. 1997).
appropriate for: direct querying and analysis
- OLAP (On-Line Analytical Processing) - a more user-oriented variant how to approach the data in DW.

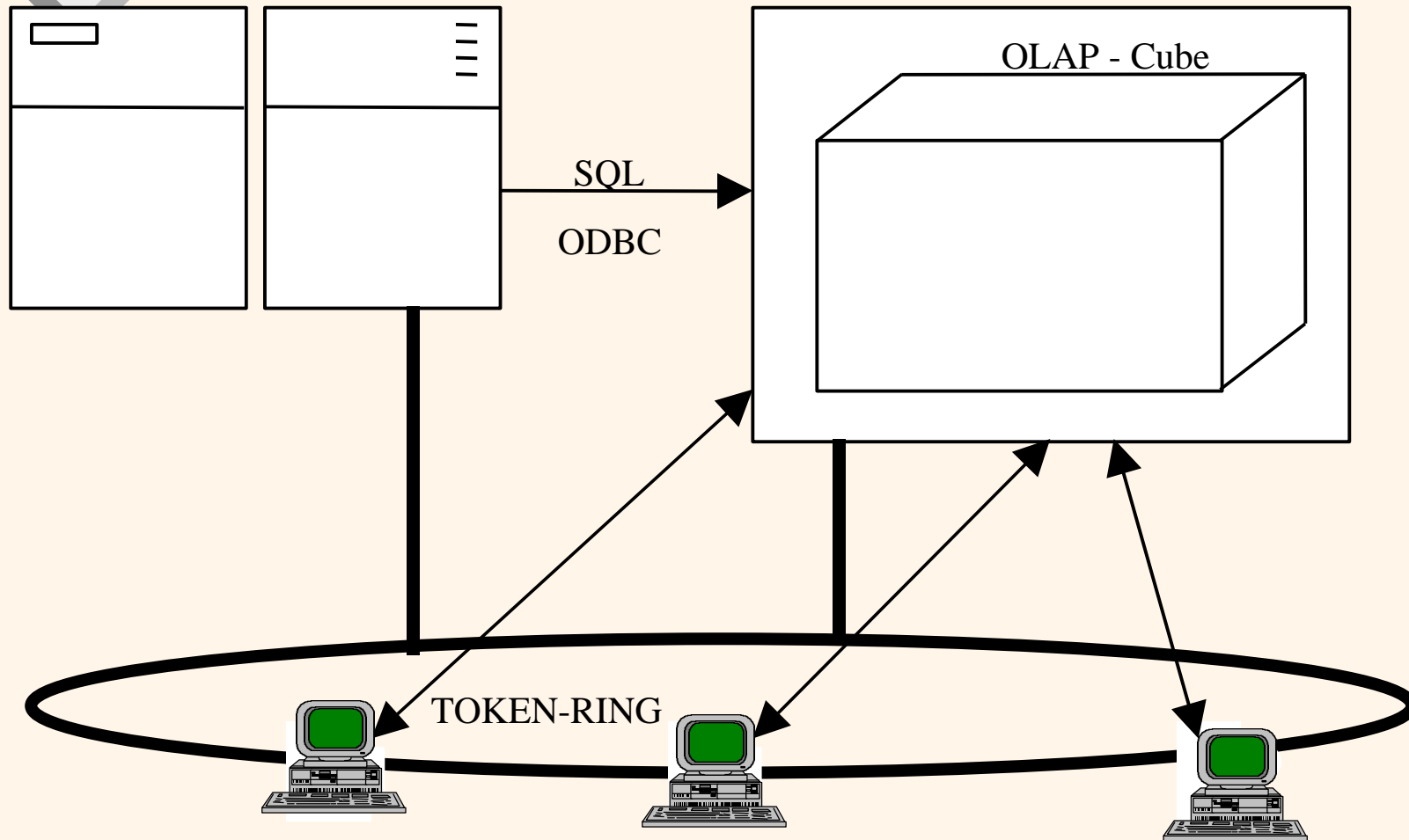
2. An Example - Tupperware Company



2. An Example - Tupperware Company

AS/400 Model 510

OLAP - Server



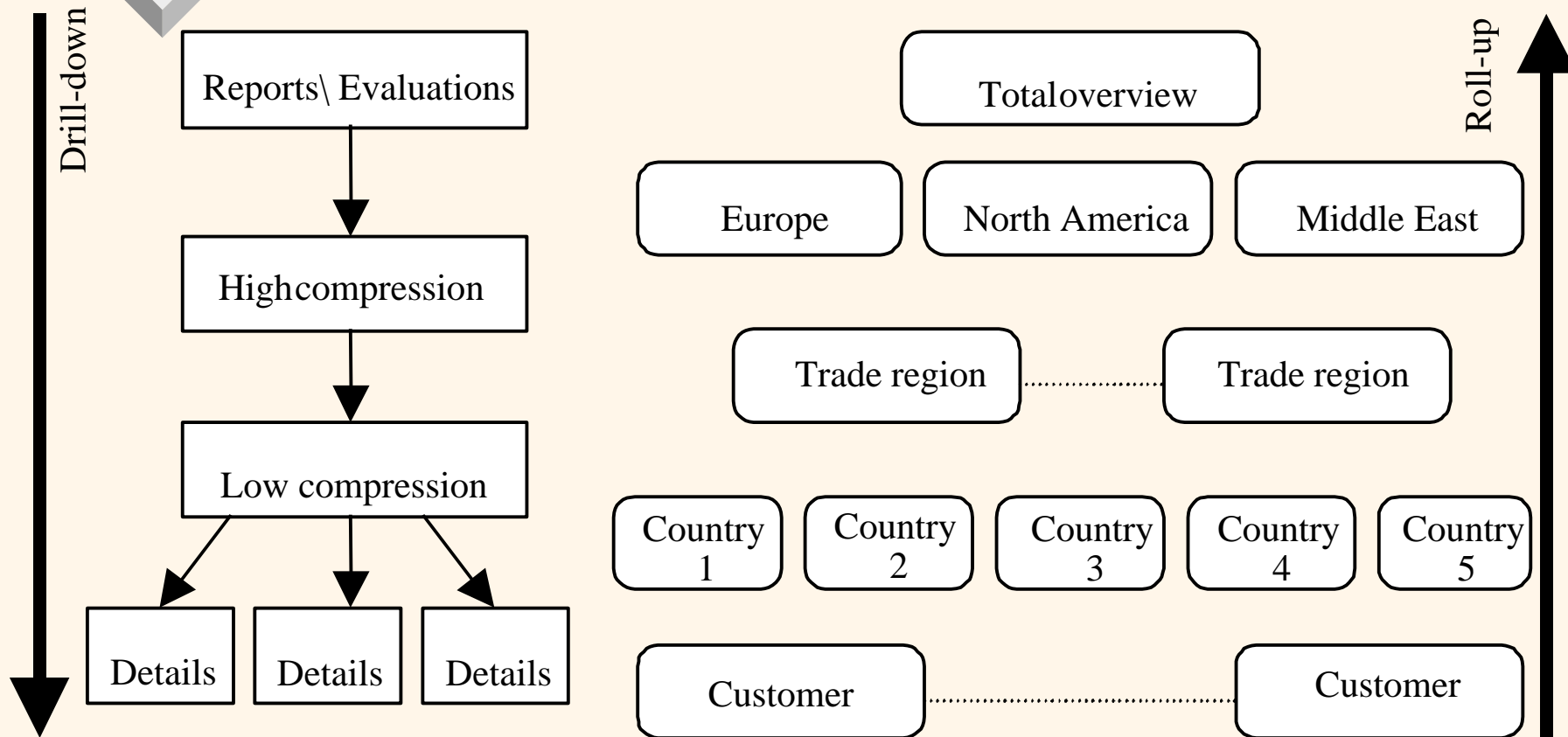
System structure of DW



3. *Functionalities of OLAP*

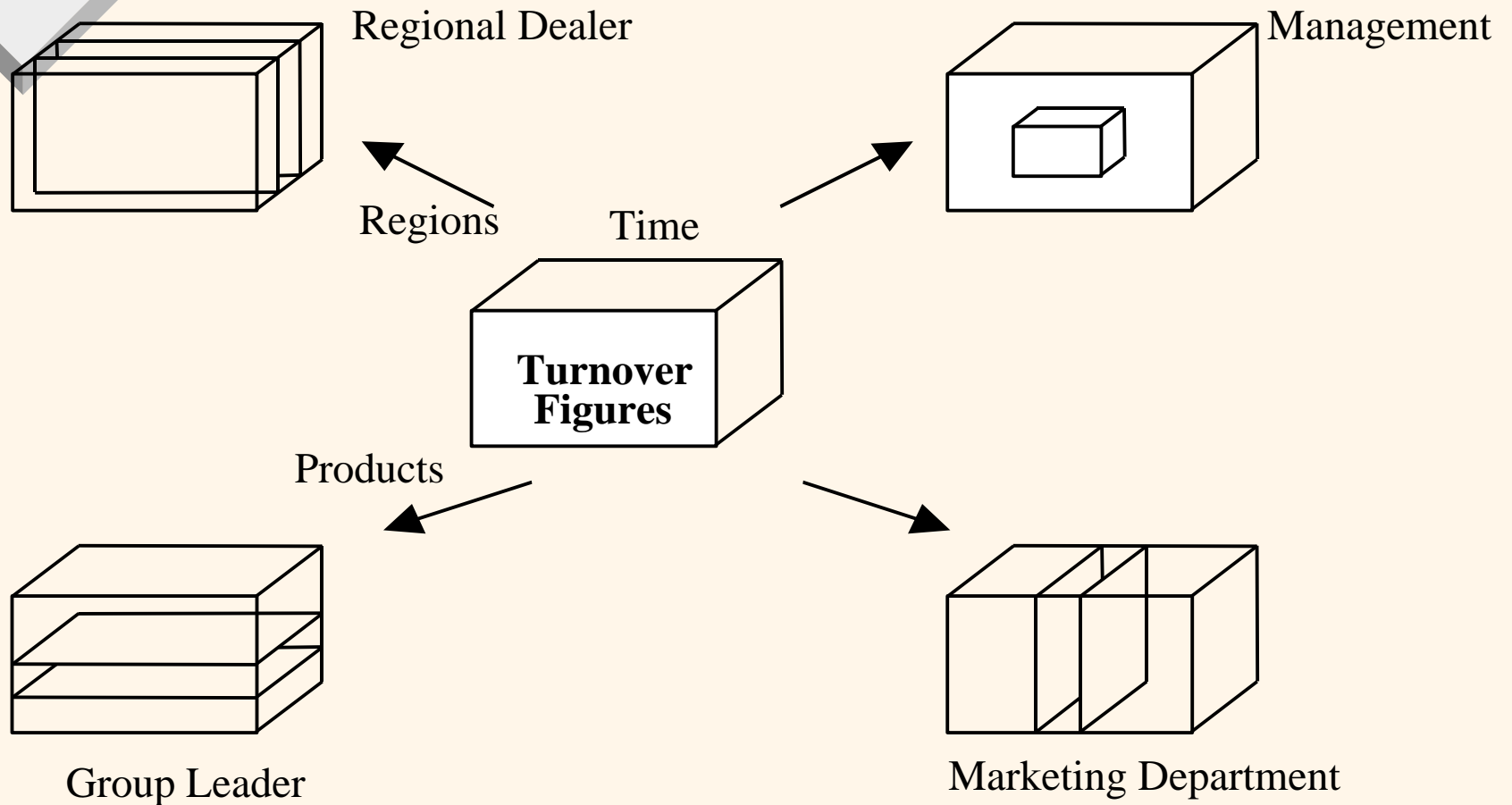
- creation of formulas
- multidimensional consolidation
- drill down and roll up
- slice and dice
- data retention in the OLAP server
- analysis technology
- heterogeneous environment

3. Functionalities of OLAP



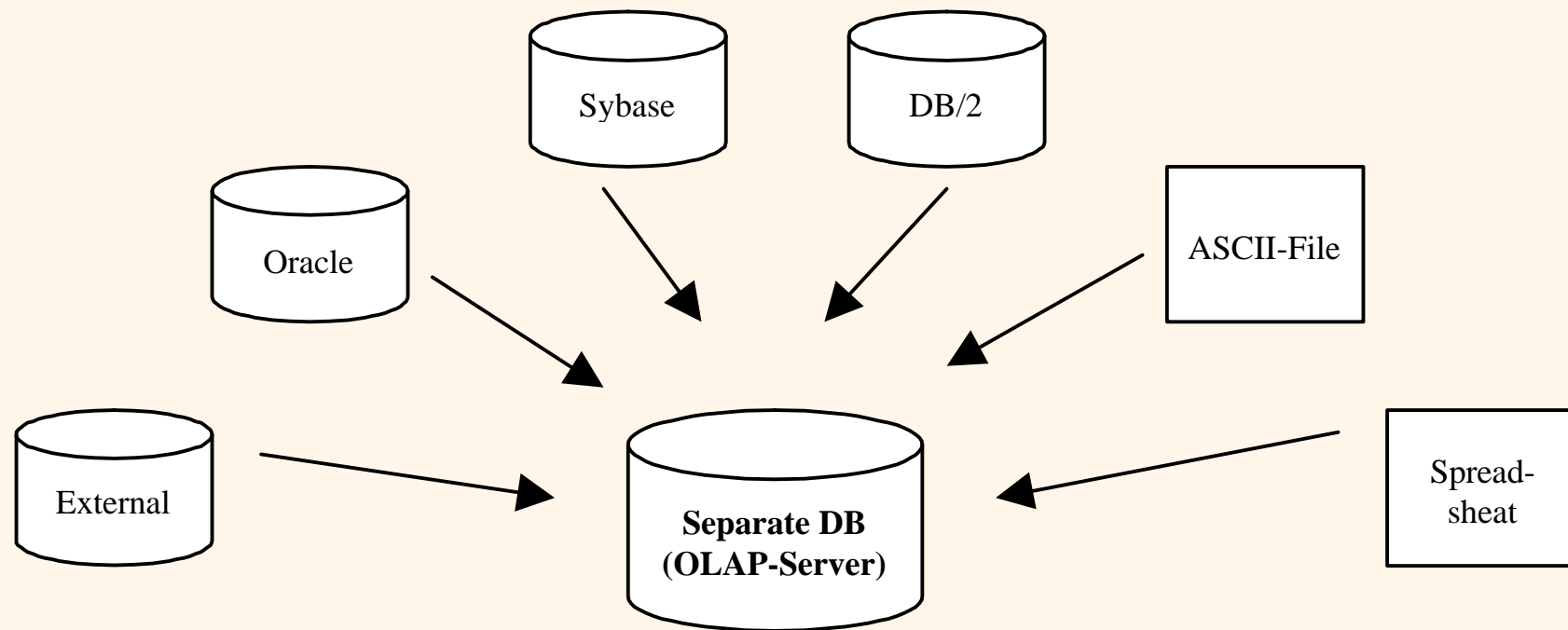
drill down and roll up

3. Functionalities of OLAP



slicing and dicing

3. Functionalities of OLAP



Heterogeneous data sources for OLAP server

4. OLAP and DW design

<i>Criteria</i>	<i>OLAP</i>	<i>OLTP</i>
Queries	In part, not predictable, (answer time: seconds to minutes)	Predictable (answer time: 0-5 seconds)
Data contents	Several years, Deduced and aggregated data	Current periods, Possibly, short histories
Data organization	The investigation can extend to cover the whole of the enterprise	Application oriented
Dimensionality	Frequently multi-dimensional	Two dimensional
Use of data	Mostly unstructured, the investigation is at the core	High degree of structuring (transaction oriented and enables location of individual data records)
Information types	Formatted or, resp., unformatted and internal/external information	Formatted and internal information
Redundancy	Monitored redundancy (star and snowflake)	Minor
Access	Mainly reading	Reading and writing



4. OLAP and DW design

Great debate: E-R vs. multidimensional approaches

Kimball, 96: E-R is inappropriate in DW, OLAP;
more recent approaches: E-R schema is *necessary* as
the first stage of modelling

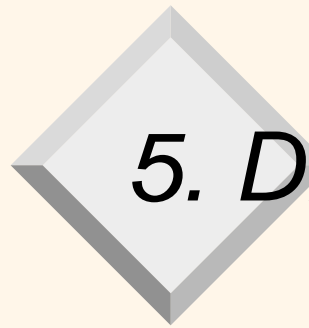
McGuff, 96: conceptual modelling
multidimensional modelling
representation modelling
physical modelling



5. *(Multi)dimensional Modelling*

Two basic approaches to multidimensional modelling (DM):

- conceptual structures are based on *tables* (*dimension* and *fact tables*) arranged into so called *star schemes*,
- conceptual structures are based on *hypercubes* (*cubes*, *multidimensional arrays*) that represent the data as a multidimensional structure.



5. DM - dimension and fact tables

MODEL

Type
Description
Number of seats
Class

COLOUR

Colour_ID
Name

SALES

Type
SalesOrg_ID
Colour_ID
Quantity
Cost
Revenue
Profit

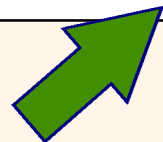
SALES ORGANIZATION

SalesOrg_ID
Representative
Office

5. DM - dimension and fact tables

MODEL

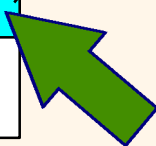
Type
Description
Number of seats
Class



attributes

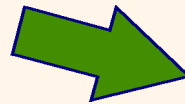
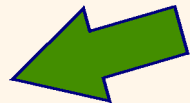
COLOUR

Colour_ID
Name



key

dimensional
tables



SALES

Type
SalesOrg_ID
Colour_ID
Quantity
Cost
Revenue
Profit



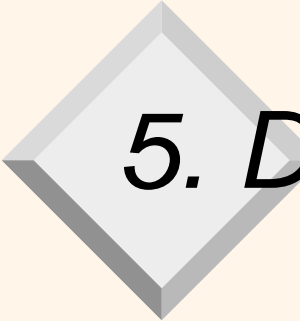
facts

Star schema

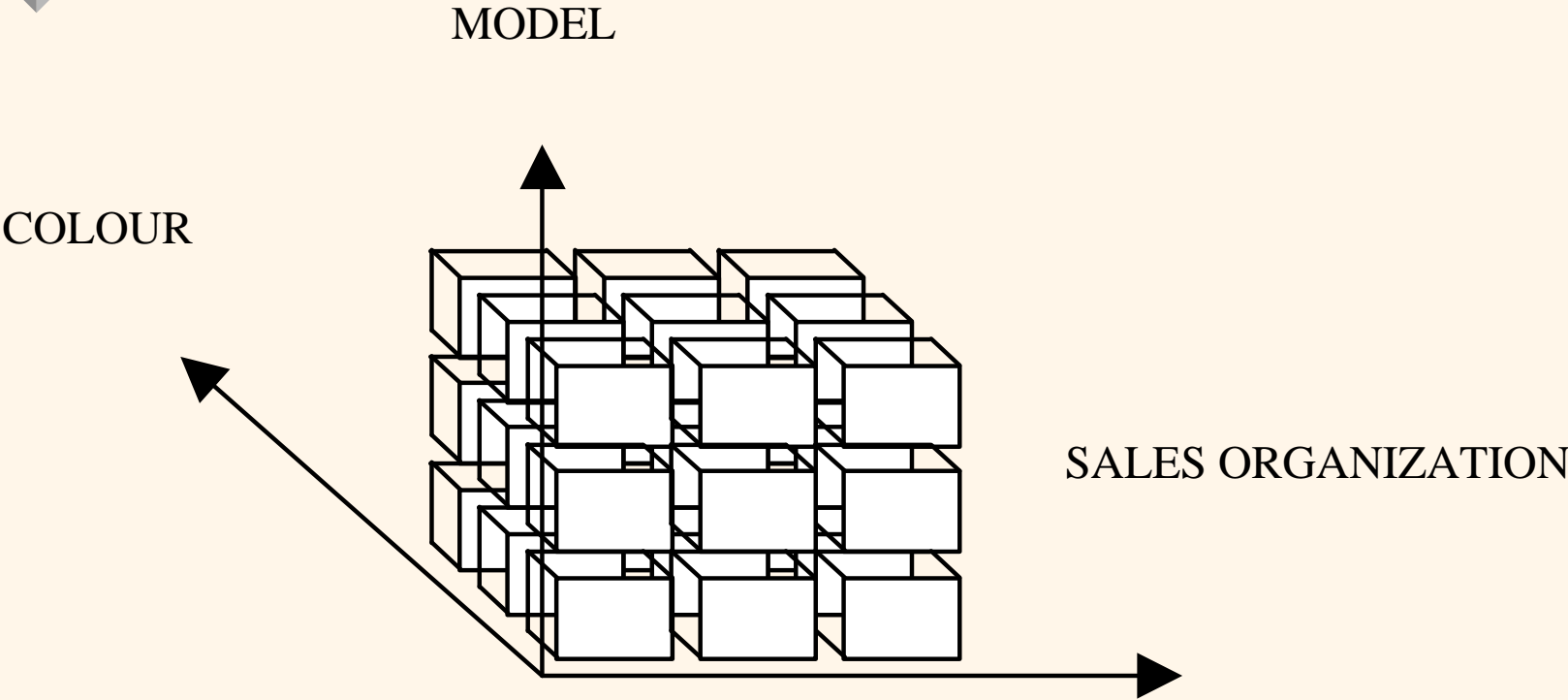
SALES ORGANIZATION

SalesOrg_ID
Representative
Office

fact
table



5. *DM - hypercube*



5. *DM - Dimensions*

dimensions are classes of descriptors of facts

dimension hierarchies:

item → class (H1)

date → month → quarter → year (H2)

office → district → region (H3)

multiple hierarchies:

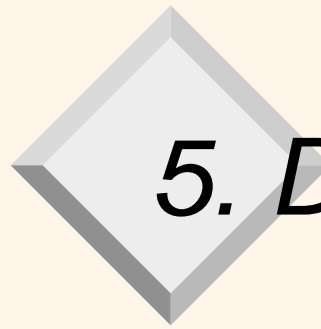
TIME: date → month → quarter → year

date → week

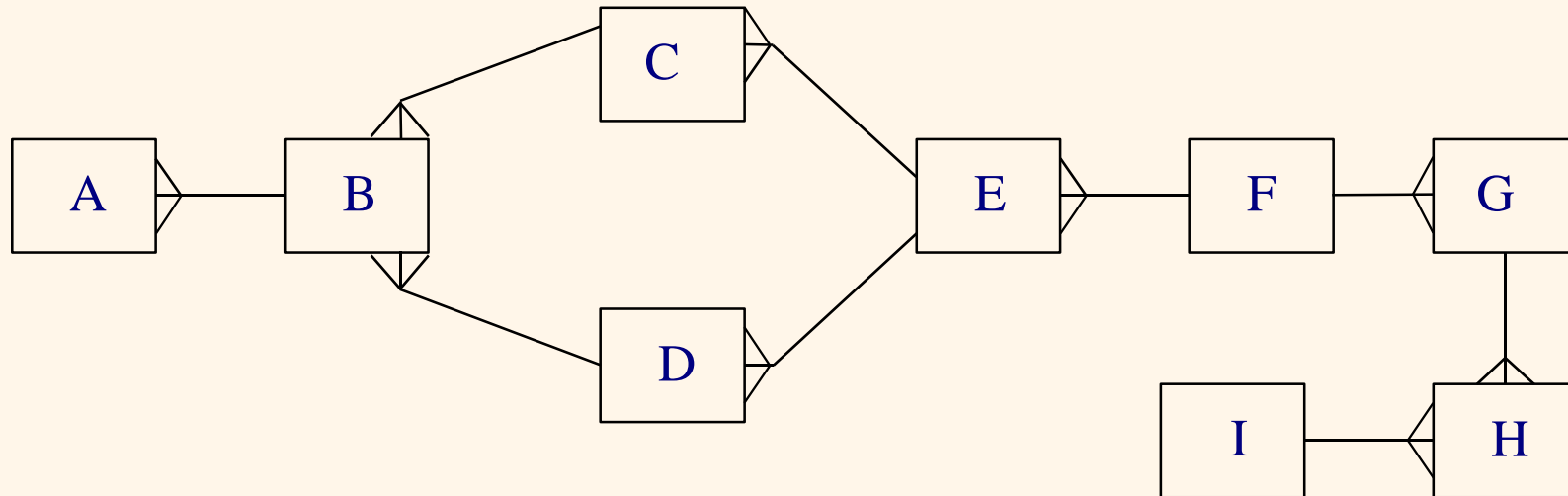
extension of a hierarchy



members of hierarchy

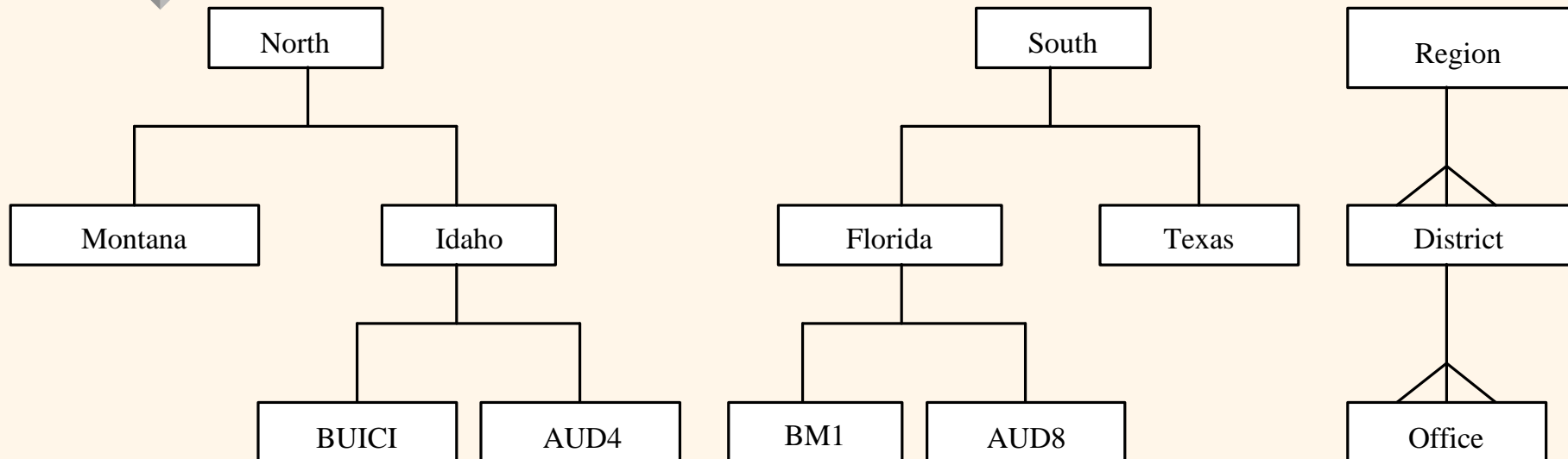


5. DM - Dimensions



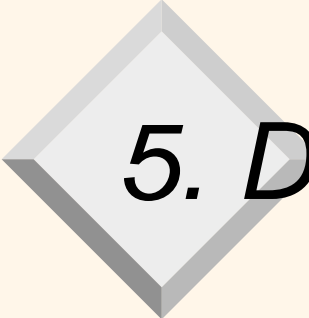
multiple hierarchies

5. DM - Dimensions



dimension extension

dimension scheme



5. *DM - facts*

facts are usually numeric quantities that describe relationships among elements of dimensions.

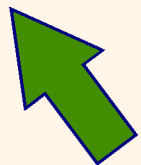
Tendency: storing a hierarchy in one dimension table D and to keep aggregations in one fact table F

⇒ how to construct the key of D.

Solution: *generated keys*

5. DM - dimension hierarchy in one table

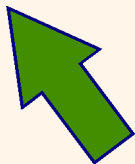
SO_Key	SalesOrg_ID	Office	Representative	District	Region	Region manager	Level
234	STO3276	BUICI	Jones	Idaho	North	Smith	Office
235	STO3189	BMI	Hover	Florida	South	Navara	Office
236	STY5478	AUD4	Archwood	Idaho	North	Smith	Office
237	STQ6781	AUD8	Seaman	Florida	South	Navara	Office
238	NULL	NULL	NULL	Florida	South	Navara	District
390	NULL	NULL	NULL	Idaho	North	Smith	District
240	NULL	NULL	NULL	NULL	North	Smith	Region
241	NULL	NULL	NULL	NULL	North	Smith	Region



generated keys

5. DM - dimension hierarchy in one table

Dimension_Key	District	Region	Level
STO327	Idaho	North	0
STO318	Florida	South	0
STY547	Idaho	North	0
STQ678	Florida	South	0
Idaho	NULL	North	1
Florida	NULL	South	1
North	NULL	NULL	2
South	NULL	NULL	2



self-identified keys

5. DM - Fact constellation schema

MODEL

Type
Description
Number of seats
Class

COLOUR

Colour_ID
Name

SALES

Type
SalesOrg_ID
Colour_ID
Quantity
Cost
Revenue
Profit

SALES ORGANIZATION

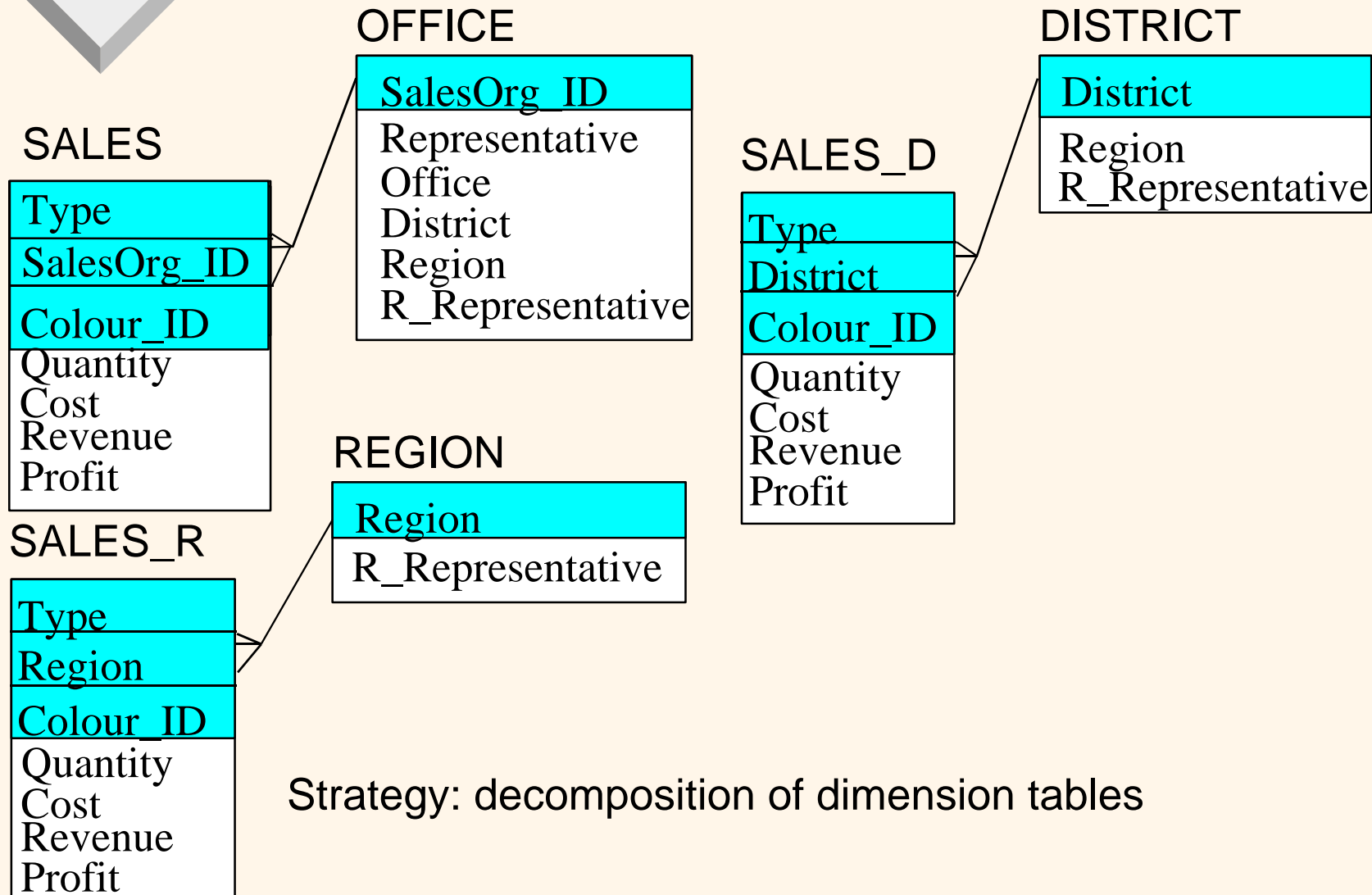
SalesOrg_ID
Representative
Office
District
Region
R_Representative

SALES_D

Type
District
Colour_ID
Quantity
Cost
Revenue
Profit

Strategy: roll-up along H3 (here only: office → district)

5. DM - Snowflakes



5. DM - with explicit dimension hierarchies

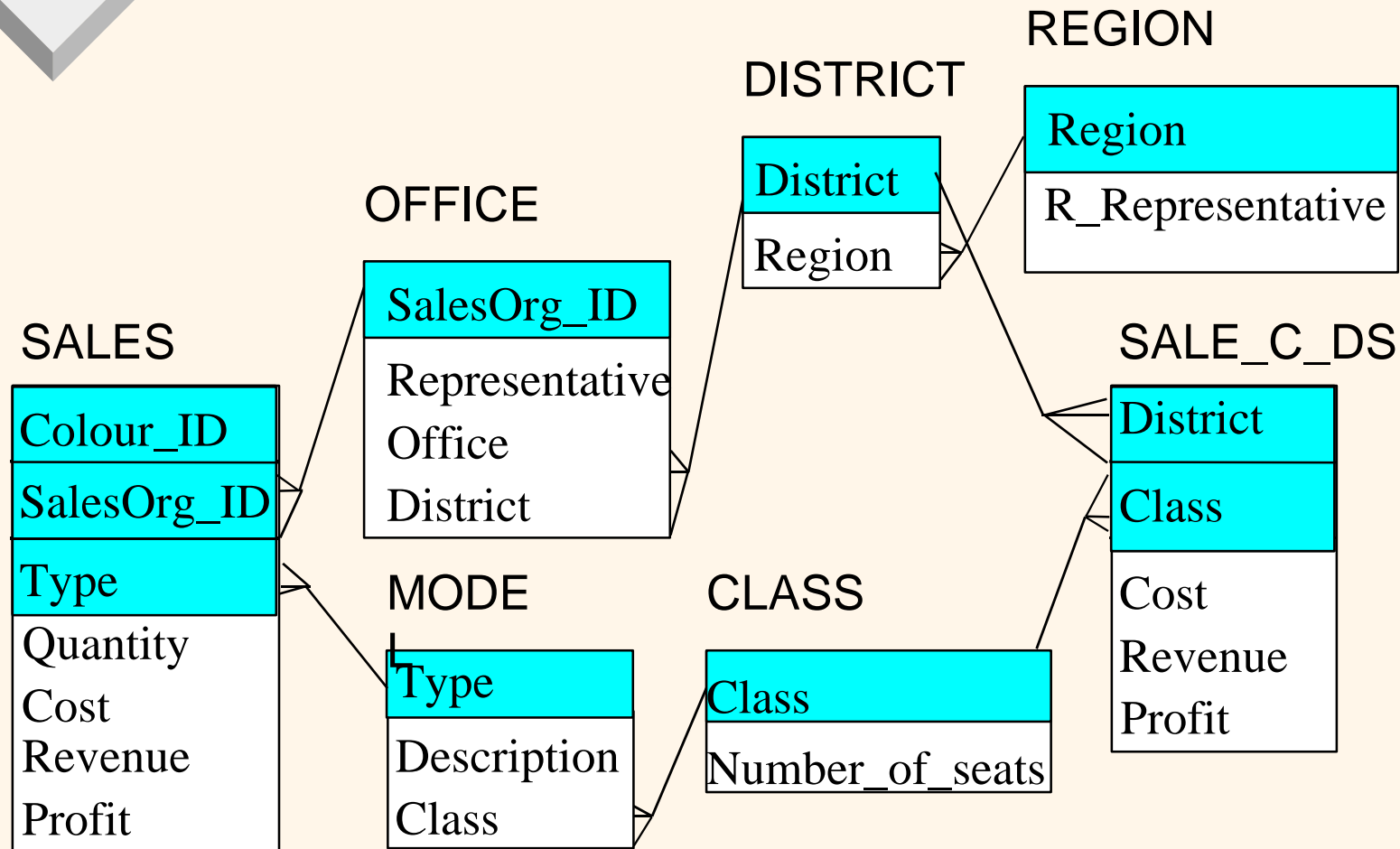


Fig. 10: Strategy: decomposition and normalization of dimension tables



5. *DM* - with explicit dimension hierarchies

constellation schema with explicit dimension hierarchies

Con is a quadruple $\langle \mathbf{D}, \mathbf{F}, \mathbf{H}, \mathbf{CC} \rangle$, where

D is a set of *dimension table schemes*

F is a set of *fact table schemes*,

H (*dimension hierarchies*) is a subset $\mathbf{D} \times \mathbf{D}$, and

CC is a set of cardinality constraints.

dimension hierarchy is a sequence $\{D_{i_1}, \dots, D_{i_k}\}$, $k > 1$,
where $(D_{i_j}, D_{i_{j+1}}) \in \mathbf{H}$, $j=1, \dots, k-1$, or $\{D\}$, $D \in \mathbf{D}$, such
that four conditions hold:

- all dimension table schemes in the sequence are
different, */acyclicity/*

5. DM - with explicit dimension hierarchies

- there are no two dimension tables schemes D' and D'' , such that (D', D_{i1}) and (D_{ik}, D'') are in \mathbf{H} ,
/paths maximality/
- if $(D_j, D_k) \in \mathbf{H}$, KD_k is the key of D_k , then KD_k is also an attribute of D_j ,
/referential integrity/
- each element of \mathbf{D} and \mathbf{H} participates at least in one dimension hierarchy,
/completeness/
- if $\{D\}$ is a dimension hierarchy, then D is not a member of any couple from \mathbf{H} .
/singleton/

each fact table is a part of a star schema

$$\mathbf{CC} = \mathbf{IC}_D \cup \mathbf{IC}_F$$

\Rightarrow *multidimensional database*

5. *DM - with explicit dimension hierarchies*

A methodology for designing DW:

- (1) defining E-R schema for basic business entities and their relationships,
- (2) revealing relationships for fact tables and dimensional hierarchies
- (3) transforming the E-R schema into a DM schema,
- (4) extending the DM schema with additional fact attributes (e.g. explicit aggregations),
- (5) defining query constraints



6. *Conclusions*

Future research:

- how to integrate multidimensional schemes,
- what query languages are possible to design (not SQL!)
- how to prove an information capacity of multidimensional schemes with aggregate data
- methodologies associated with approaches to multidimensional modelling.