Database Management Systems
Chapter 9: Data Warehousing

University of Maryland
Robert H. Smith
School of Business

Dr. Adam Lee

Objectives

- Define terms.
- Explore reasons for information gap between information needs and availability.
- Understand reasons for need of data warehousing.
- Describe three levels of data warehouse architectures.
- Describe two components of star schema.
- Estimate fact table size.
- Design a data mart.
- Develop requirements for a data mart.

Definitions

- **Data Warehouse** – A subject-oriented, integrated, time-variant, non-updatable collection of data used in support of management decision-making processes.
  - **Subject-oriented**: e.g. customers, patients, students, products.
  - **Integrated**: consistent naming conventions, formats, encoding structures; from multiple data sources.
  - **Time-variant**: can study trends and changes.
  - **Non-updatable**: read-only, periodically refreshed.

- **Data Mart** – A data warehouse that is limited in scope.

History Leading to Data Warehousing

- Improvement in database technologies, especially relational DBMSs.
- Advances in computer hardware, including mass storage and parallel architectures.
- Emergence of end-user computing with powerful interfaces and tools.
- Advances in middleware, enabling heterogeneous database connectivity.
- Recognition of difference between operational and informational systems.

Need for Data Warehousing

- Integrated, company-wide view of high-quality information (from disparate databases).
- Separation of operational and informational systems and data (for improved performance).

Issues with Company-Wide View

- ✗ Inconsistent key structures.
- ✗ Synonyms.
- ✗ Free-form vs. structured fields.
- ✗ Inconsistent data values.
- ✗ Missing data.

See Figure 9-1 for example.

Figure 9-1: Examples of Heterogeneous Data

STUDENT DATA

StudentNo	LastName	MI	FirstName	Telephone	Status	• • •
123-45-6789	Enright	T	Mark	483-1967	Soph	
389-21-4062	Smith	R	Elaine	283-4195	Jr	

STUDENT EMPLOYEE

StudentID	Address	Dept	Hours	• • •
123-45-6789	1218 Elk Drive, Phoenix, AZ 91304	Soc	8	
389-21-4062	134 Mesa Road, Tempe, AZ 90142	Math	10	

STUDENT HEALTH

StudentName	Telephone	Insurance	ID	• • •
Mark T. Enright	483-1967	Blue Cross	123-45-6789	
Elaine R. Smith	555-7828	?	389-21-4062	

Separating Operational and Informational Systems

- **Operational system** – A system that is used to run a business in real time, based on current data; also called a system of record.

- **Informational system** – A system designed to support decision making based on historical point-in-time and prediction data for complex queries or data-mining applications.

Table 9-1: Comparison of Operational and Informational Systems

TABLE 9-1 Comparison of Operational and Informational Systems

Characteristic	Operational Systems	Informational Systems
Primary purpose	Run the business on a current basis	Support managerial decision making
Type of data	Current representation of state of the business	Historical point-in-time (snapshots) and predictions
Primary users	Clerks, salespersons, administrators	Managers, business analysts, customers
Scope of usage	Narrow, planned, and simple updates and queries	Broad, ad hoc, complex queries and analysis
Design goal	Performance: throughput, availability	Ease of flexible access and use
Volume	Many constant updates and queries on one or a few table rows	Periodic batch updates and queries requiring many or all rows

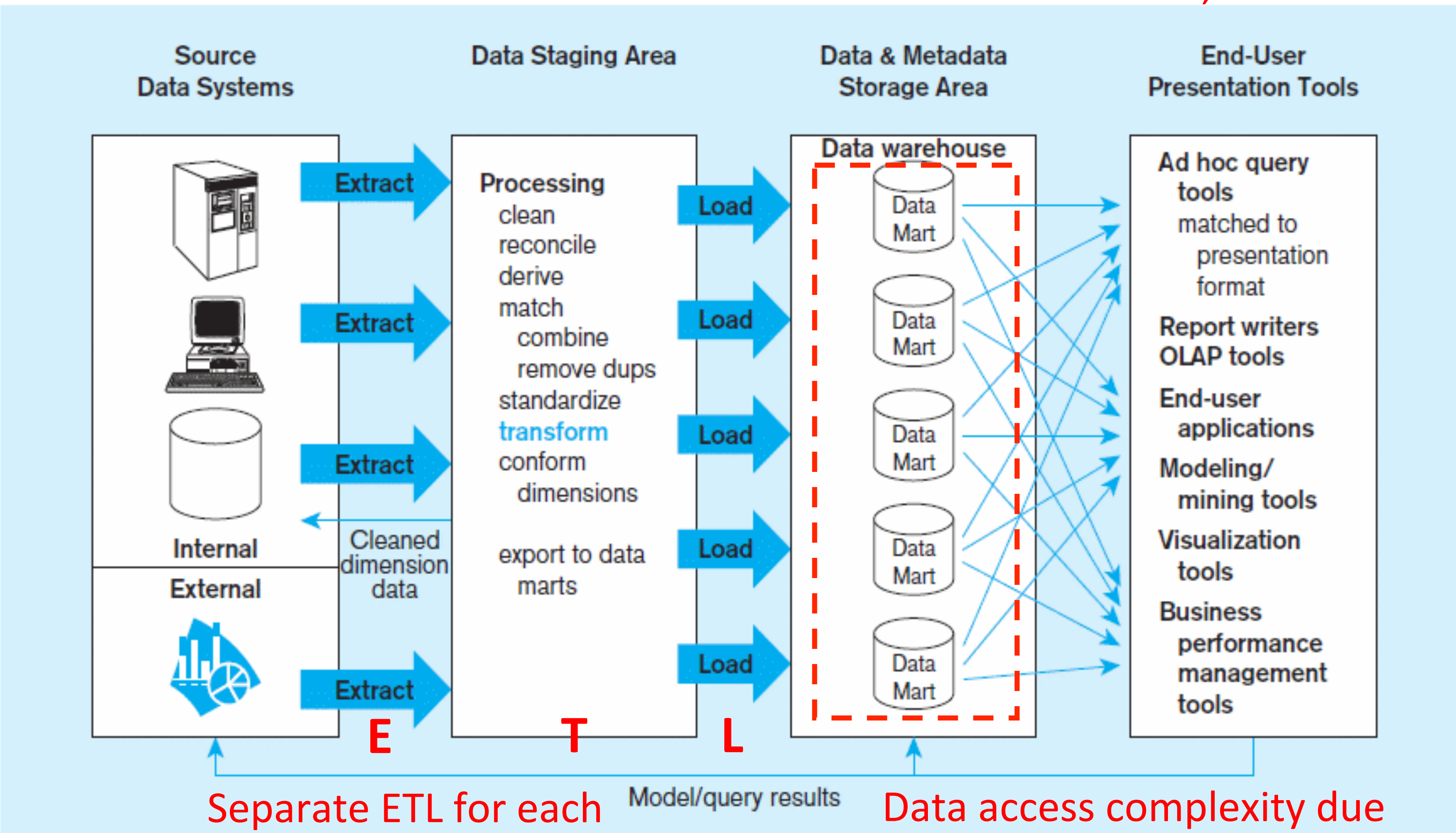
Data Warehouse Architectures

- Independent data mart.
- Dependent data mart and operational data store.
- Logical data mart and real-time data warehouse.
- Three-layer architecture.

All involve some form of ***extract***, ***transform*** and ***load*** (**ETL**).

Figure 9-2: Independent Data Mart Data Warehousing Architecture

Data marts:
Mini-warehouses, limited in scope.

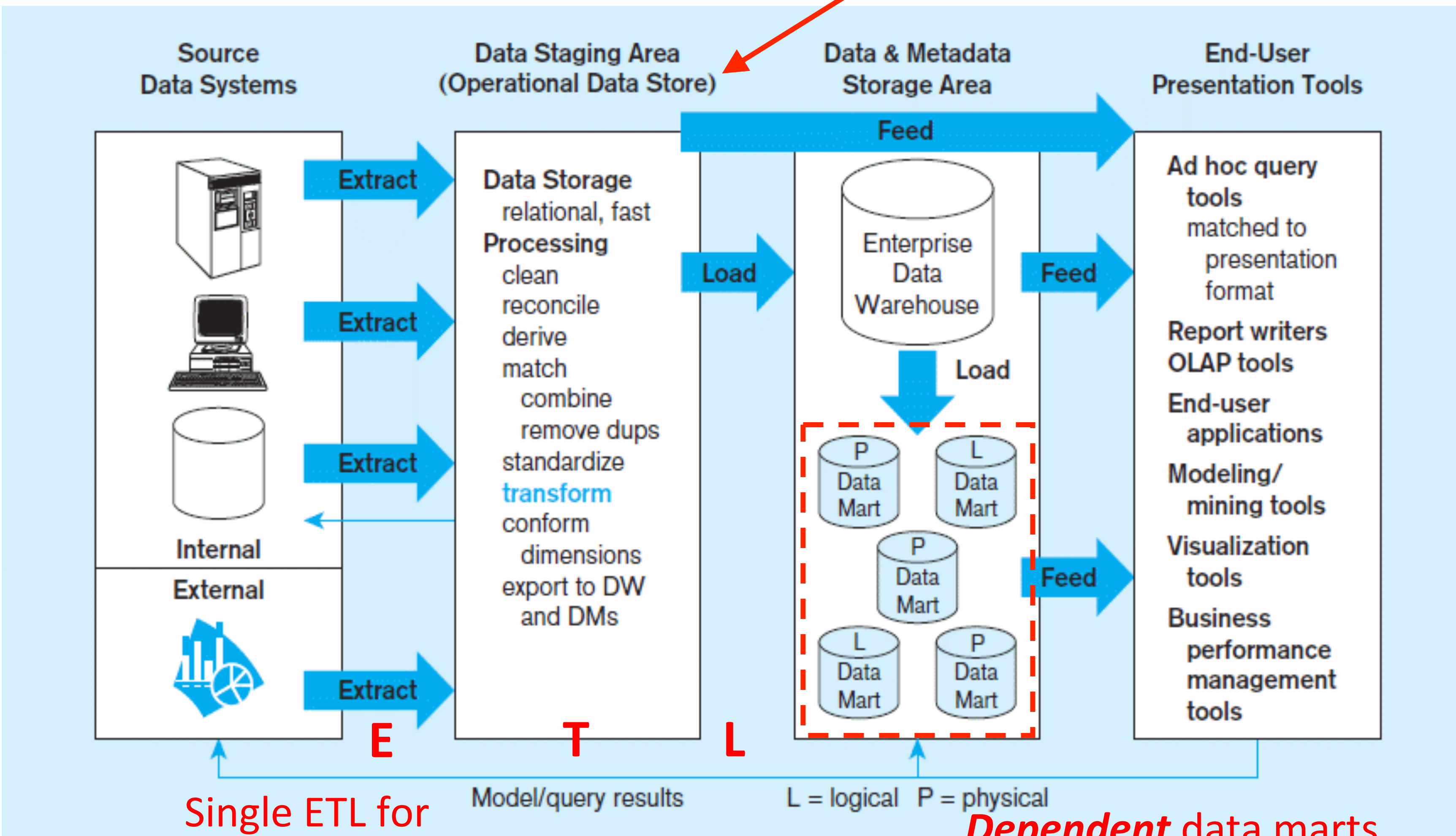


Separate ETL for each ***independent*** data mart.

Data access complexity due to ***multiple*** data marts.

Figure 9-3: Dependent Data Mart with Operational Data Store

ODS provides option for obtaining ***current*** data.

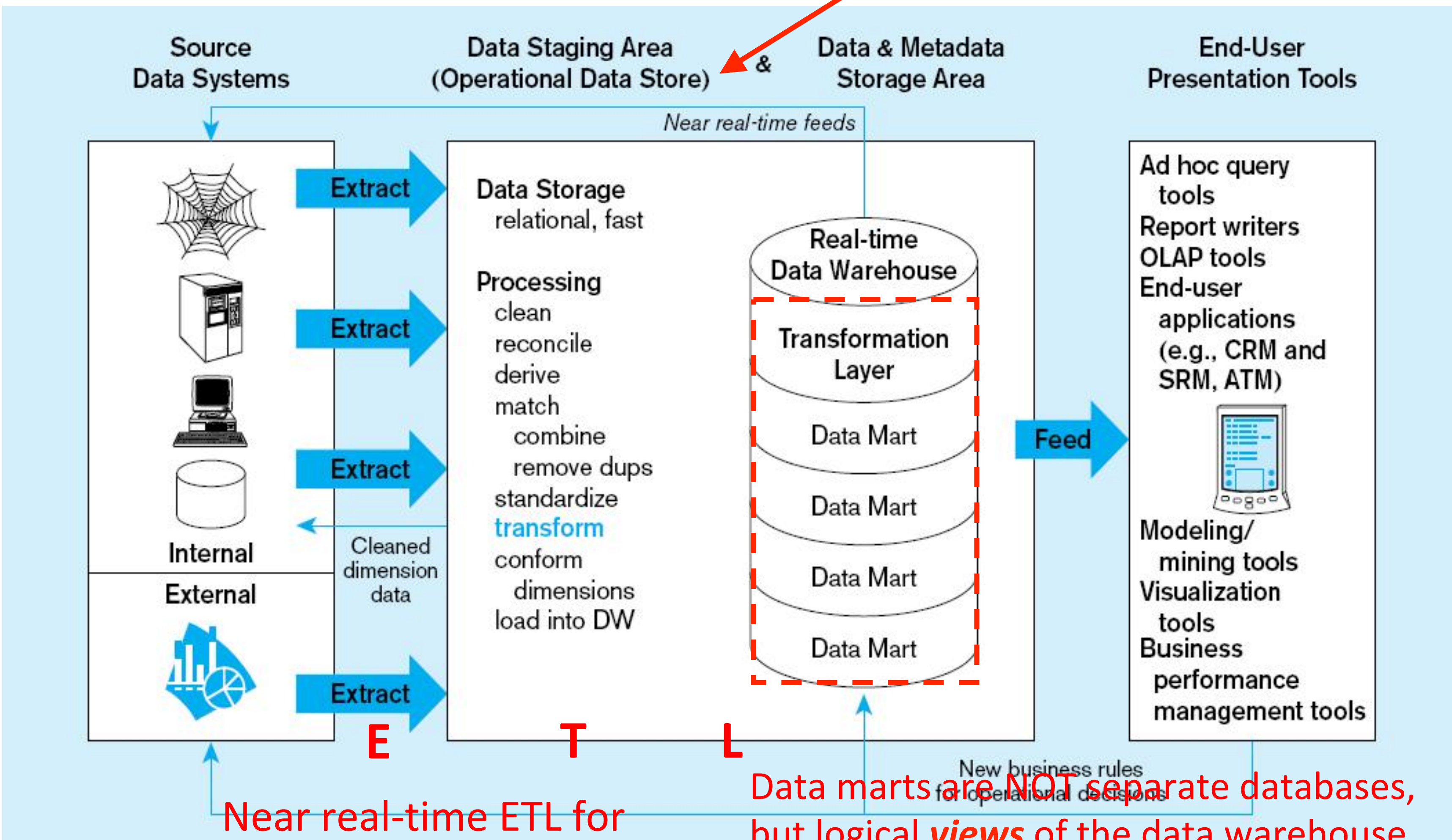


Single ETL for ***enterprise data warehouse (EDW)***.

Dependent data marts loaded from EDW.

Figure 9-4: Logical Data Mart and Real Time Warehouse Architecture

ODS and **data warehouse** are one and the same.



Near real-time ETL for ***Data Warehouse***.

Data marts are NOT separate databases, but logical ***views*** of the data warehouse.
➔ Easier to create new data marts.

Table 9-2: Data Warehouse Versus Data Mart

TABLE 9-2 Data Warehouse Versus Data Mart

Data Warehouse	Data Mart
Scope	**Scope**
• Application independent • Centralized, possibly enterprise-wide • Planned	• Specific DSS application • Decentralized by user area • Organic, possibly not planned
Data	**Data**
• Historical, detailed, and summarized • Lightly denormalized	• Some history, detailed, and summarized • Highly denormalized
Subjects	**Subjects**
• Multiple subjects	• One central subject of concern to users
Sources	**Sources**
• Many internal and external sources	• Few internal and external sources
Other Characteristics	**Other Characteristics**
• Flexible • Data oriented • Long life • Large • Single complex structure	• Restrictive • Project oriented • Short life • Start small, becomes large • Multi, semi-complex structures, together complex

Figure 9-5: Three-Layer Data Architecture for a Data Warehouse

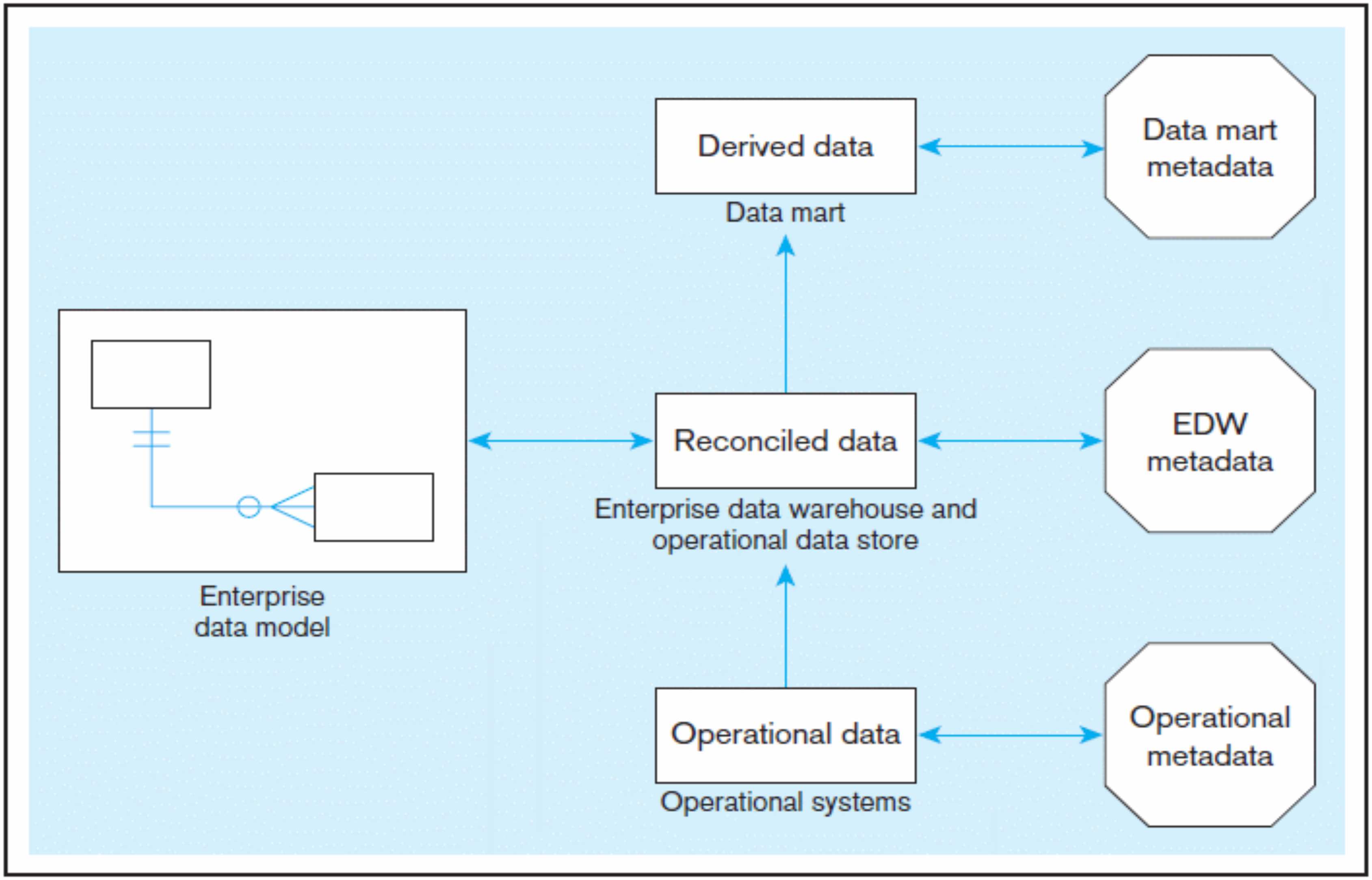
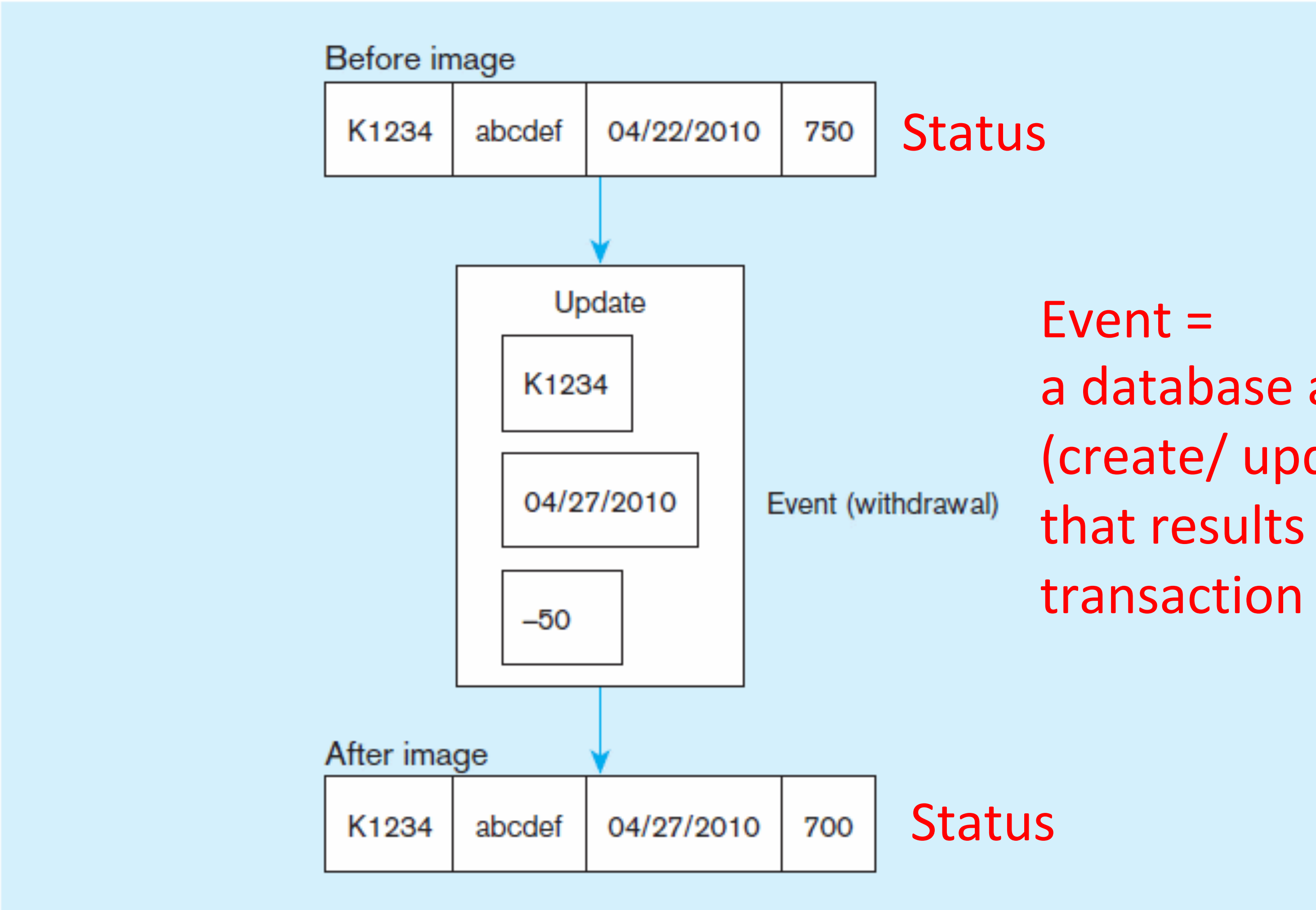


Figure 9-6: Example of DBMS Log Entry (Data Characteristics Status versus Event Data)



Event =
a database action
(create/ update/ delete)
that results from a
transaction

Figure 9-7: Transient Operational Data (Data Characteristics Status versus Event Data)

Table X (10/09)

Key	A	B
001	a	b
002	c	d
003	e	f
004	g	h

Table X (10/10)

Key	A	B
001	a	b
002	r	d
003	e	f
004	y	h
005	m	n

Table X (10/11)

Key	A	B
001	a	b
002	r	d
003	e	t
005	m	n

With transient data, changes to existing records are written over previous records, thus destroying the previous data content.

Figure 9-8: Periodic Warehouse Data (Data Characteristics Status versus Event Data)

Table X (10/09)

Key	Date	A	B	Action
001	10/09	a	b	C
002	10/09	c	d	C
003	10/09	e	f	C
004	10/09	g	h	C

Table X (10/10)

Key	Date	A	B	Action
001	10/09	a	b	C
002	10/09	c	d	C
002	10/10	r	d	U
003	10/09	e	f	C
004	10/09	g	h	C
004	10/10	y	h	U
005	10/10	m	n	C

Table X (10/11)

Key	Date	A	B	Action
001	10/09	a	b	C
002	10/09	c	d	C
002	10/10	r	d	U
003	10/09	e	f	C
003	10/11	e	t	U
004	10/09	g	h	C
004	10/10	y	h	U
004	10/11	y	h	D
005	10/10	m	n	C

Periodic data are never physically altered or deleted once they have been added to the store.

Other Data Warehouse Changes

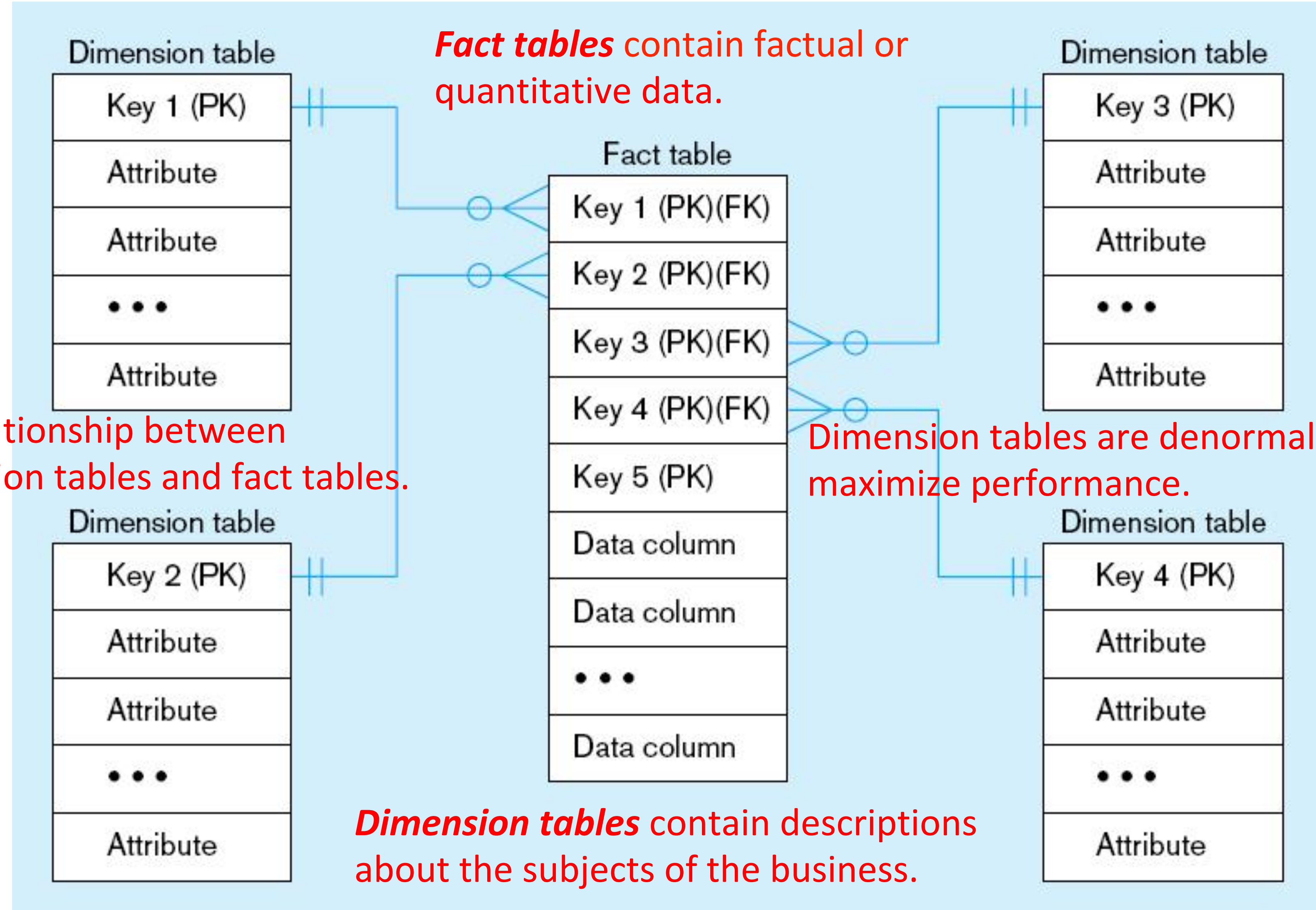
- New descriptive attributes.
- New business activity attributes.
- New classes of descriptive attributes.
- Descriptive attributes become more refined.
- Descriptive data are related to one another.
- New source of data.

Derived Data

- Objectives:
  - Ease of use for decision support applications.
  - Fast response to predefined user queries.
  - Customized data for particular target audiences.
  - Ad-hoc query support.
  - Data mining capabilities.
- Characteristics:
  - Detailed (mostly periodic) data.
  - Aggregate (for summary).
  - Distributed (to departmental servers).

Most common data model = **dimensional model** (usually implemented as a **star schema**)

Figure 9-9: Components of a Star Schema



Fact tables contain factual or quantitative data.

1:N relationship between dimension tables and fact tables.

Dimension tables are denormalized to maximize performance.

Dimension tables contain descriptions about the subjects of the business.

Excellent for ad-hoc queries, but bad for online transaction processing

Figure 9-10: Star Schema Example

Fact table provides statistics for sales broken down by product, period and store dimensions.

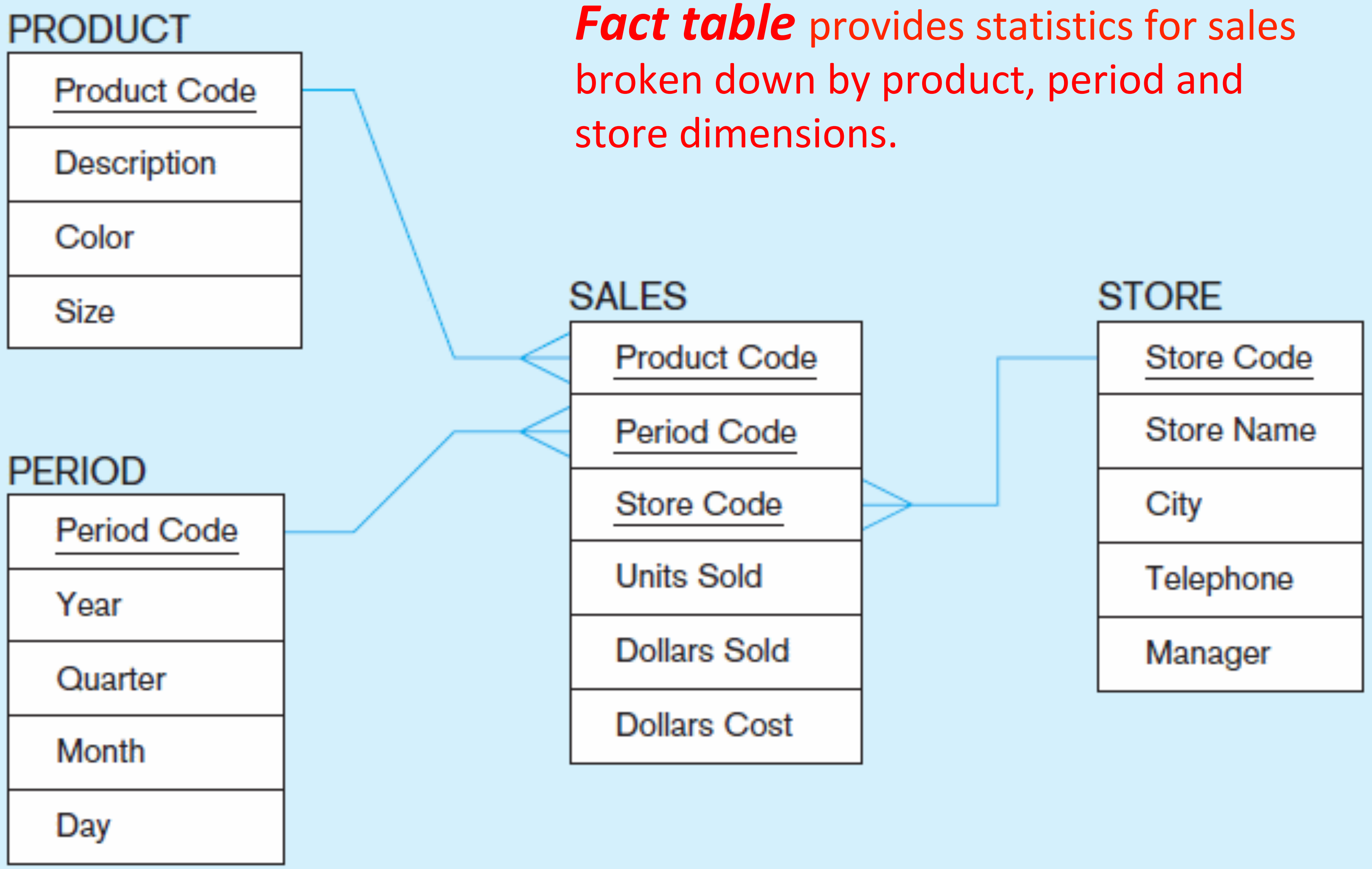


Figure 9-11: Star Schema with Sample Data

Product

Product Code	Description	Color	Size
100	Sweater	Blue	40
110	Shoes	Brown	10 1/2
125	Gloves	Tan	M
• • •			

Period

Period Code	Year	Quarter	Month
001	2010	1	4
002	2010	1	5
003	2010	1	6
• • •			

Sales

Product Code	Period Code	Store Code	Units Sold	Dollars Sold	Dollars Cost
110	002	S1	30	1500	1200
125	003	S2	50	1000	600
100	001	S1	40	1600	1000
110	002	S3	40	2000	1200
100	003	S2	30	1200	750
• • •					

Store

Store Code	Store Name	City	Telephone	Manager
S1	Jan's	San Antonio	683-192-1400	Burgess
S2	Bill's	Portland	943-681-2135	Thomas
S3	Ed's	Boulder	417-196-8037	Perry
• • •				

Surrogate Keys

- Dimension table keys should be surrogate (non-intelligent and non-business related), because:
  - Business keys may change over time.
  - Helps keep track of non-key attribute values for a given production key.
  - Surrogate keys are simpler and shorter.
  - Surrogate keys can be same length and format for all key.

Grain of the Fact Table

- Granularity of Fact Table – what level of detail do you want?
  - Transactional grain – finest level.
  - Aggregated grain – more summarized.
  - Finer grains → better market basket analysis capability.
  - Finer grain → more dimension tables, more rows in fact table.
  - In Web-based commerce, finest granularity is a click.

Duration of the Database

- Natural duration – 13 months or 5 quarters.
- Financial institutions may need longer duration.
- Older data is more difficult to source and cleanse.

Size of Fact Table

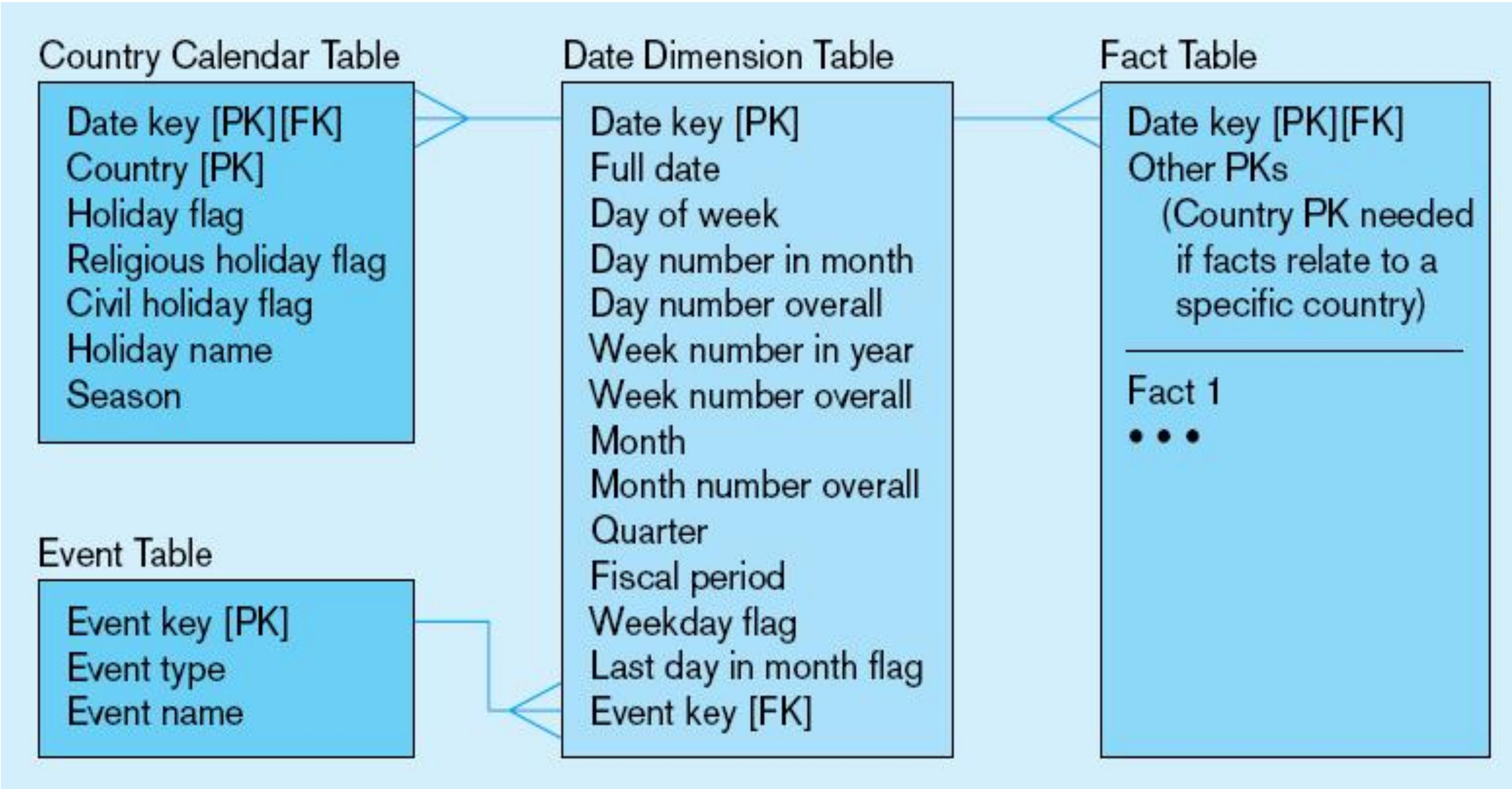
- Depends on the number of dimensions and the grain of the fact table
- Number of rows = product of number of possible values for each dimension associated with the fact table
- Example: Assume the following for Figure 9-11:

Total number of stores = 1,000
Total number of products = 10,000
Total number of periods = 24 (2 years' worth of monthly data)

- Total rows calculated as follows (assuming only half the products record sales for a given month):

Total rows = 1,000 stores × 5,000 active products × 24 months
= 120,000,000 rows (!)

Figure 9-12 Modeling Dates



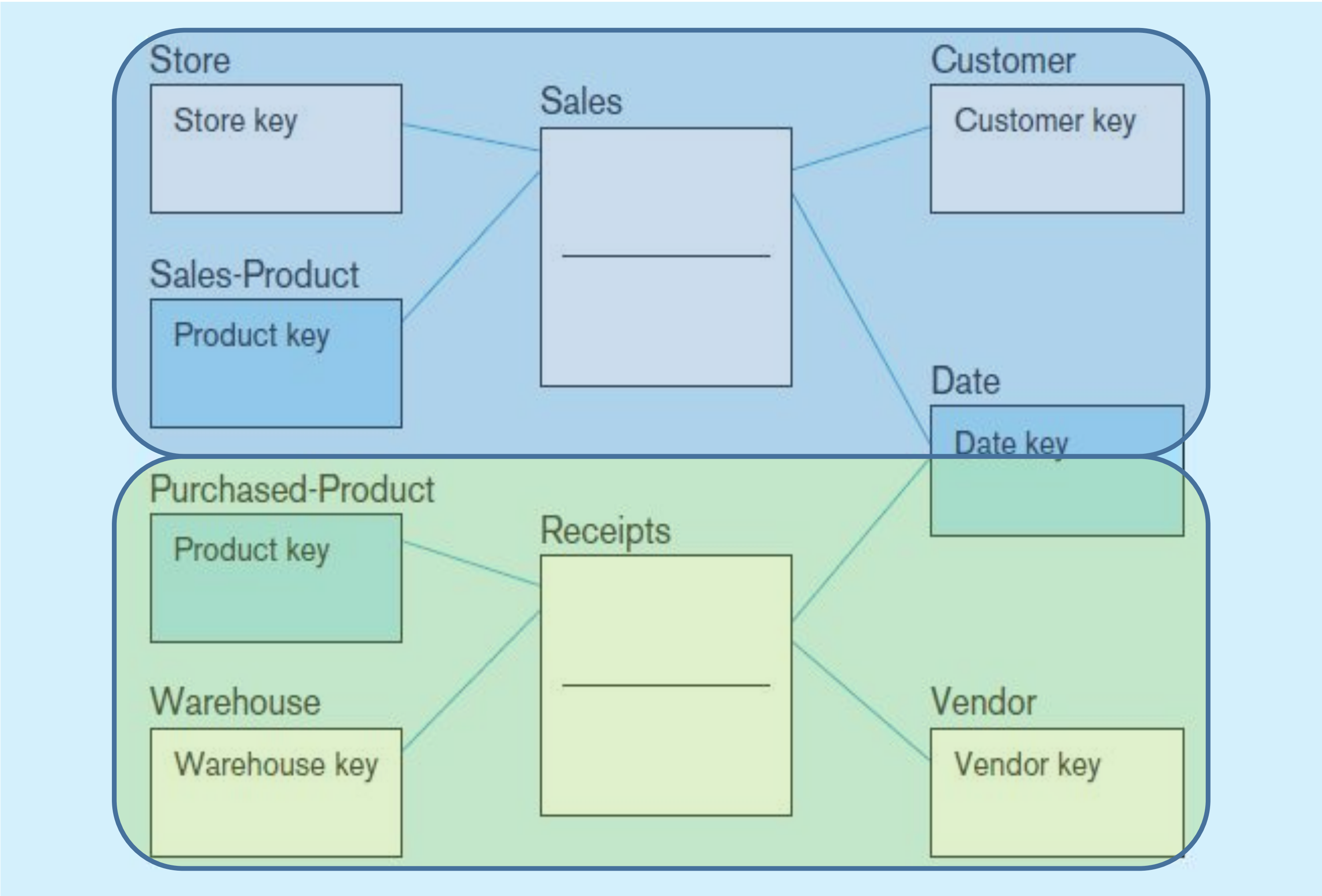
Fact tables contain time-period data
➔ Date dimensions are important

Variations of the Star Schema

- Multiple Facts Tables:
  - Can improve performance.
  - Often used to store facts for different combinations of dimensions.
  - Conformed dimensions.
- Factless Facts Tables:
  - No non-key data, but foreign keys for associated dimensions.
  - Used for: tracking events, inventory coverage, …

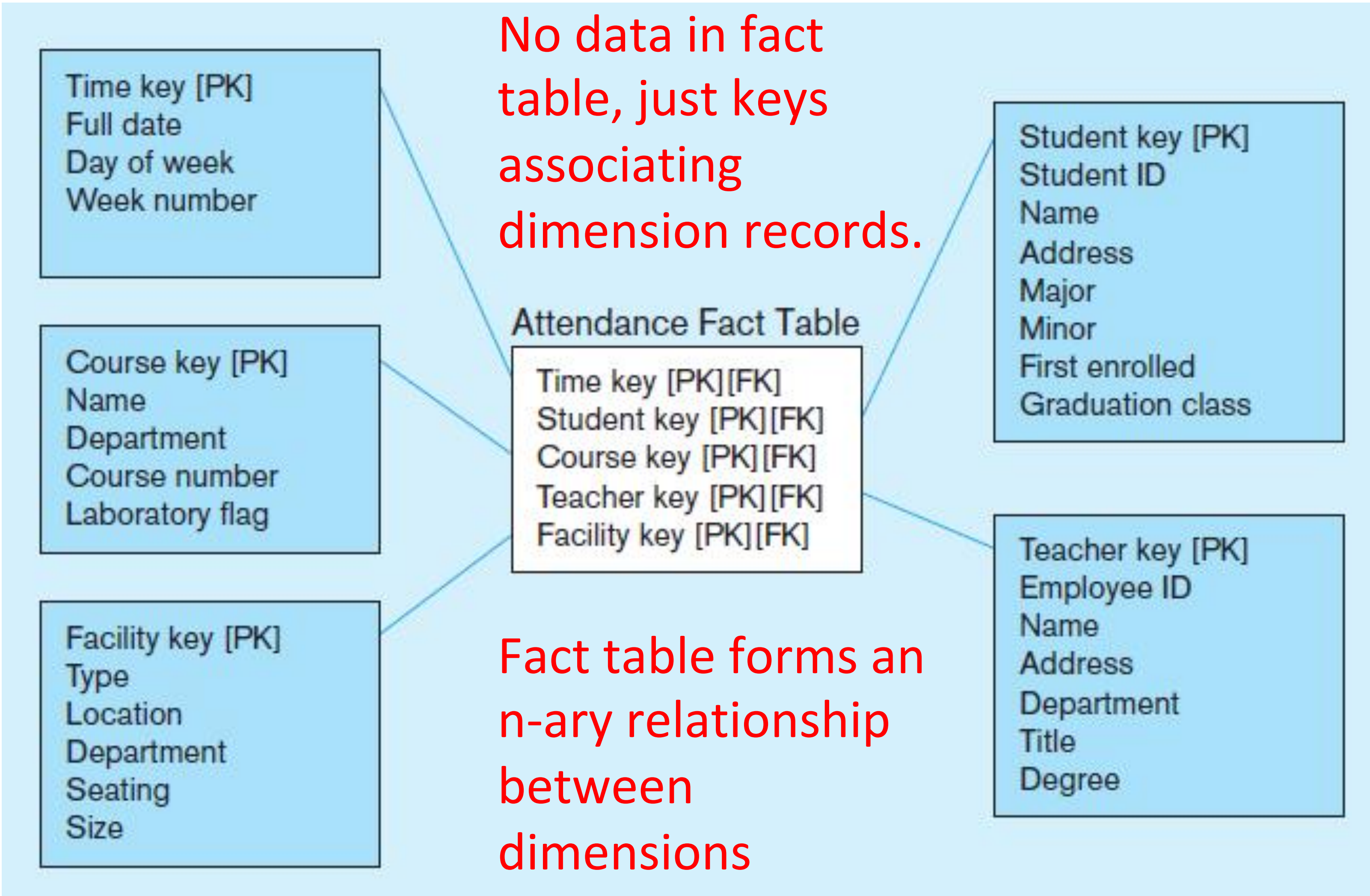
Figure 9-13: Conformed Dimensions

Two fact tables → two (connected) star schemas.



Conformed dimension
Associated with multiple fact tables.

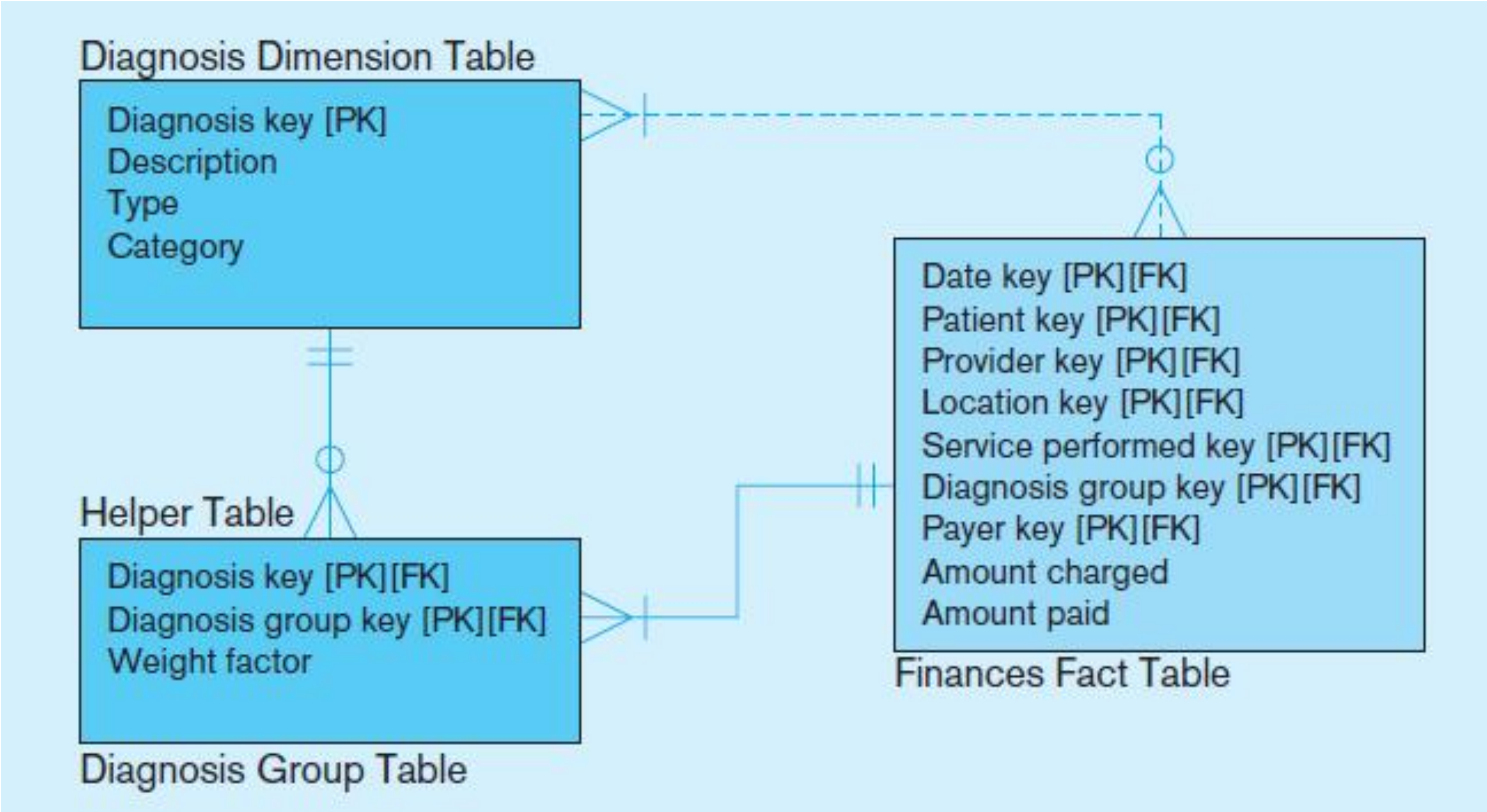
Figure 9-14: Factless Fact Table Showing Occurrence of an Event



Normalizing Dimension Tables

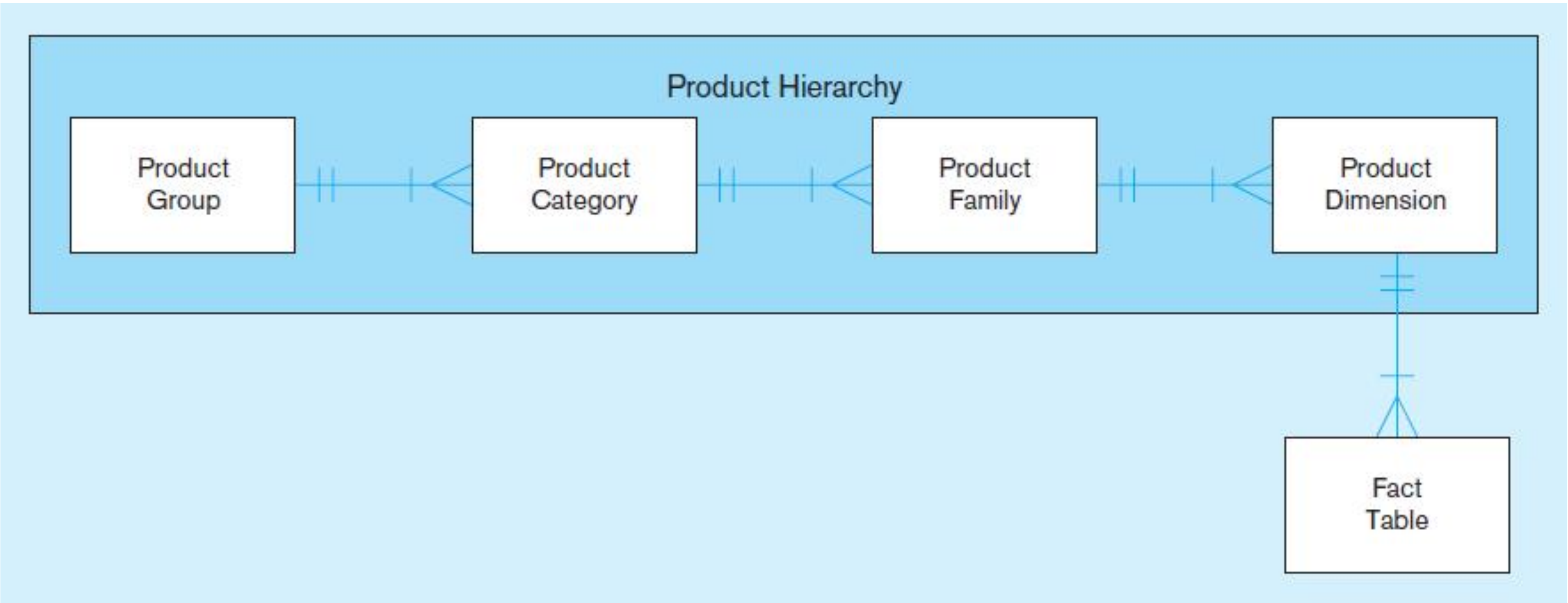
- Multivalued Dimensions:
  - Facts qualified by a set of values for the same business subject.
  - Normalization involves creating a table for an associative entity between dimensions.
- Hierarchies:
  - Sometimes a dimension forms a natural, fixed depth hierarchy.
- Design options:
  - Include all information for each level in a single denormalized table.
  - Normalize the dimension into a nested set of 1:M table relationships.

Figure 9-15: Multivalued Dimension



Helper table is an associative entity that implements a M:N relationship between dimension and fact.

Figure 9-16: Fixed Product Hierarchy

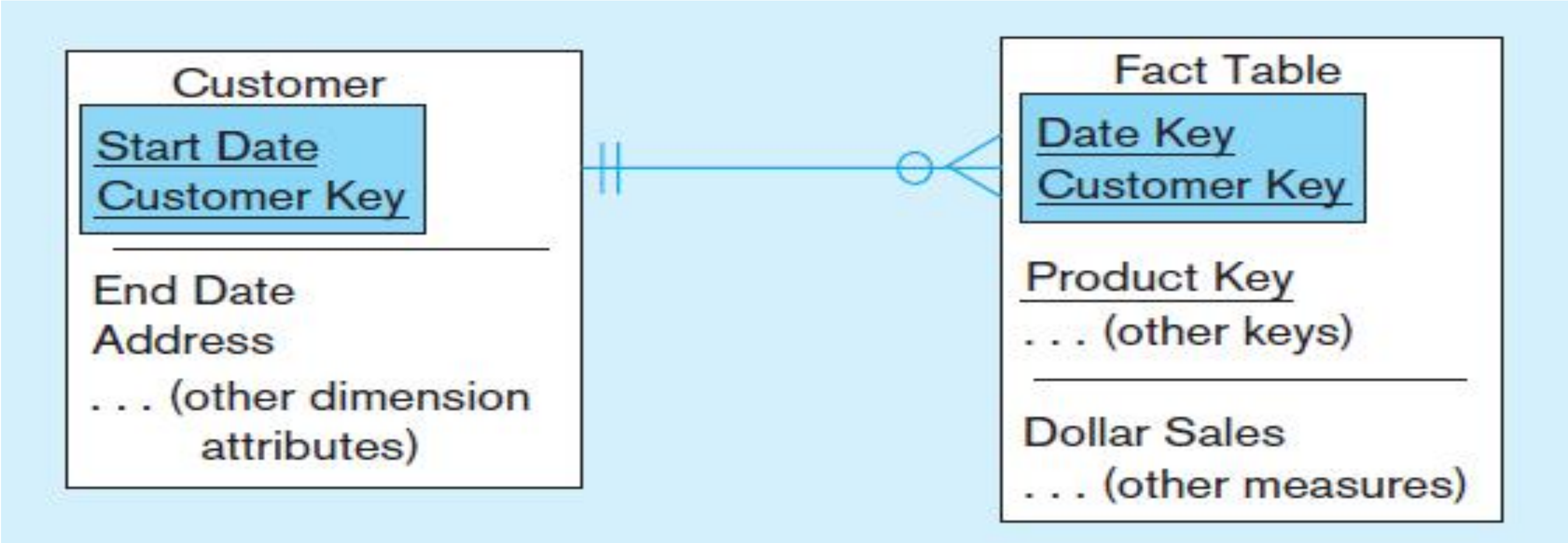


Dimension hierarchies help to provide levels of aggregation for users wanting summary information in a data warehouse.

Slowly Changing Dimensions (SCD)

- How to maintain knowledge of the past?
- Kimble's approaches:
  - Type 1: just replace old data with new. (lose historical data)
  - Type 2: create a new dimension table row each time the dimension object changes, with all dimension characteristics at the time of change. (most common approach)
  - Type 3: for each changing attribute, create a current value field and several old-valued fields. (multivalued)

Figure 9-18: Example of Type 2 SCD Customer Dimension Table

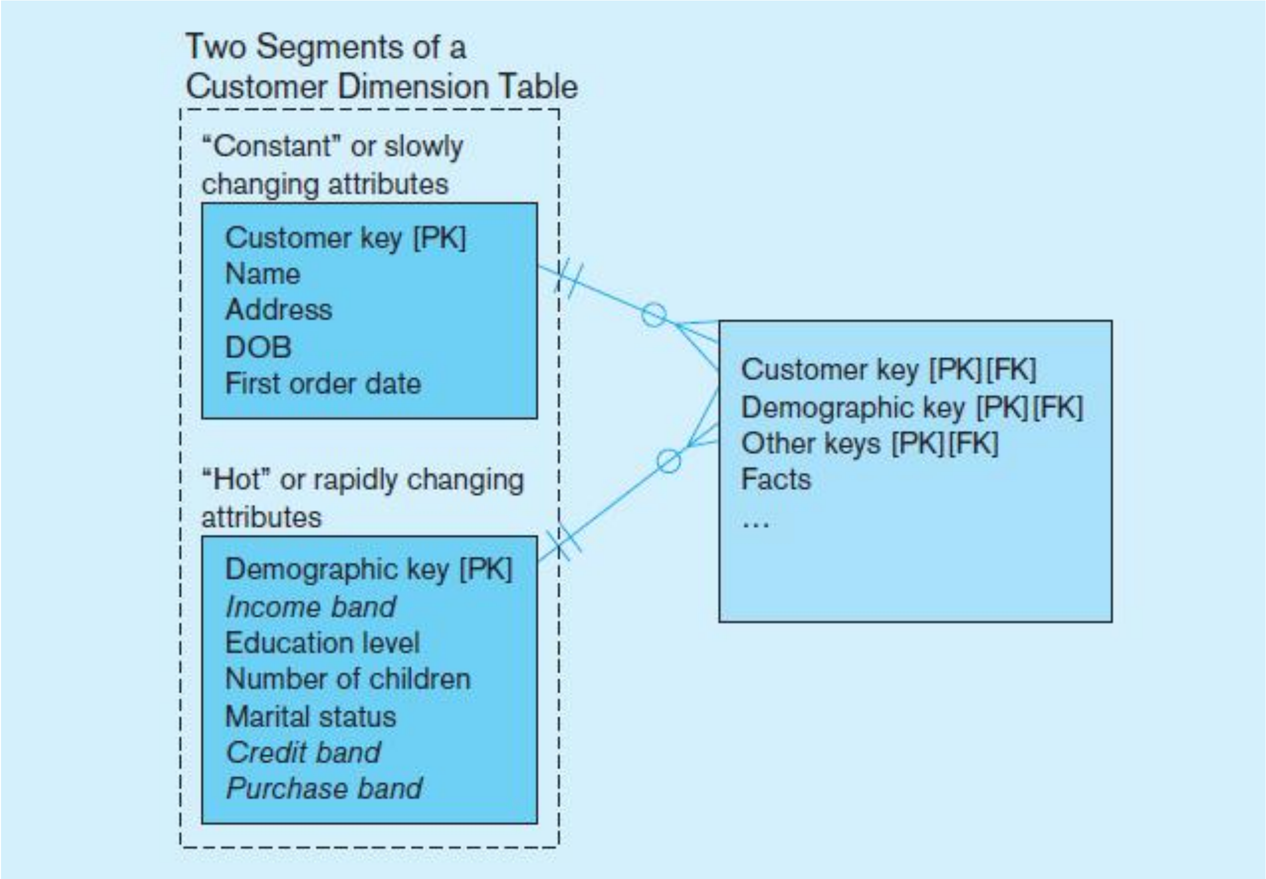


The dimension table contains several records for the same customer. The specific customer record to use depends on the key and the date of the fact, which should be between start and end dates of the SCD customer record.

```
WHERE Fact.CustomerKey = Customer.CustomerKey
AND Fact.DateKey BETWEEN Customer.StartDate and Customer.EndDate
```

Figure 9-19: Dimension Segmentation

For rapidly changing attributes (hot attributes), Type 2 SCD approach creates too many rows and too much redundant data. Use segmentation instead.



UNIVERSITY OF MARYLAND
ROBERT H. SMITH SCHOOL OF BUSINESS

10 Essential Rules For Dimensional Modeling

1. Use atomic facts.
2. Create single-process fact tables.
3. Include a date dimension for each fact table.
4. Enforce consistent grain.
5. Disallow null keys in fact tables.
6. Honor hierarchies.
7. Decode dimension tables.
8. Use surrogate keys.
9. Conform dimensions.
10. Balance requirements with actual data.

Other Data Warehouse Advances

- Move data warehouse into the cloud to enjoy the benefits of lower total cost of ownership (TCO).
  - IBM, Oracle, Microsoft, Teradata, SAP (HANA), Amazon (Redshift).
- Columnar databases:
  - Issue of Big Data (huge volume, often unstructured).
  - Optimize storage for summary data of few columns.
  - Sybase, Vertica, Infobright.
- NoSQL:
  - "Not only SQL".
  - Deals with unstructured data.
  - MongoDB, CouchDB, Apache Cassandra.