Conceptual Modeling

COMP323 Chapter 2

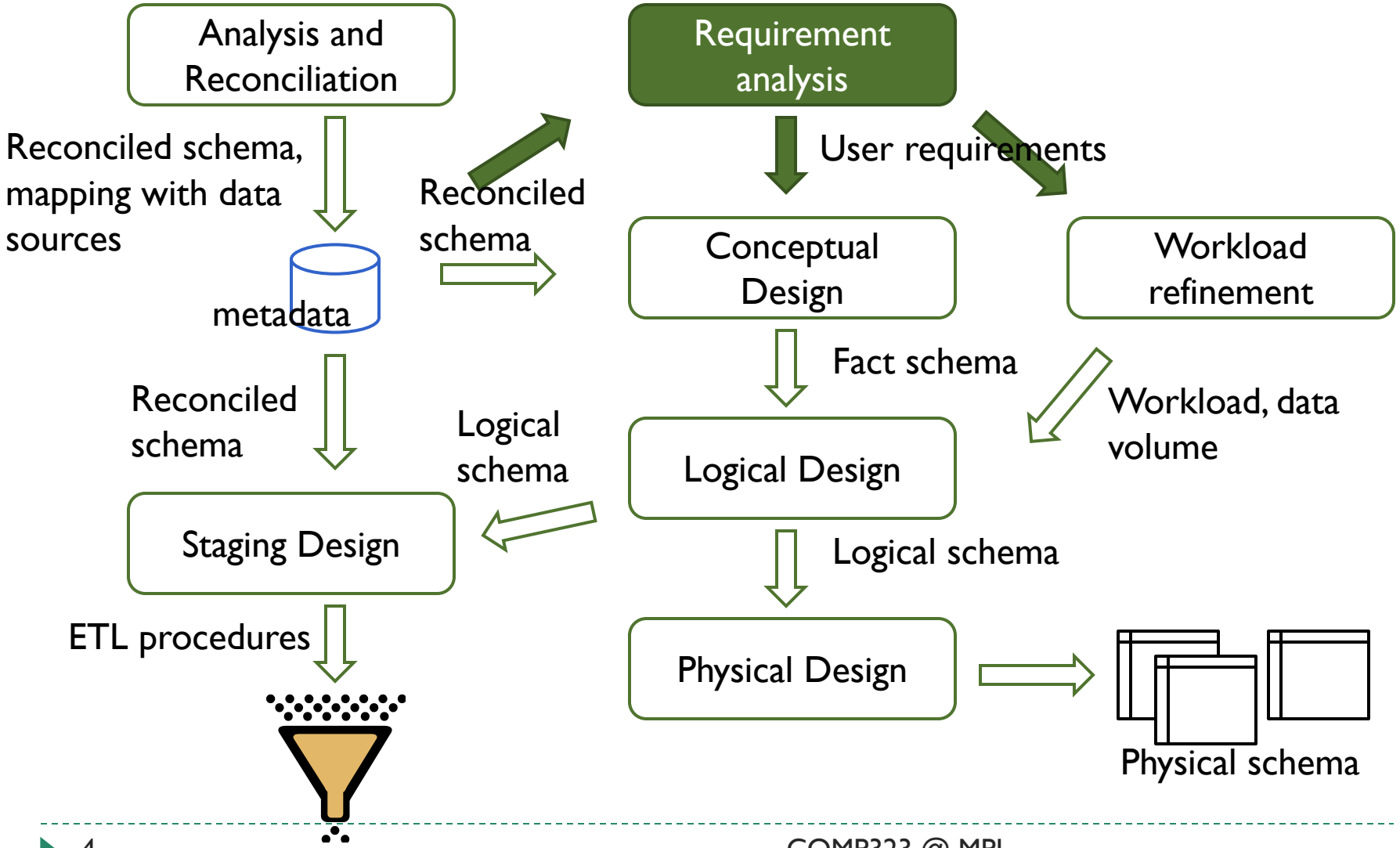
Outline

- A. Requirement Analysis
  - Identifying fact (with measures) and dimensions (with levels) from typical analysis workload and reports
- B. Basics of Conceptual modeling
  - Notation
  - Granularity, functional dependency between dim attr, others…
  - Case study: data-driven
- C. Temporal Nature and Aggregation
  - Transactional vs. snapshot facts
  - Event, aggregation, additivity of measures
  - Calculated measures

Part A. Requirement Analysis

- Difficulties in user requirement analysis
- Multidimensional model
  - Dimensional nature of business data
  - Fact, dimension, measure, hierarchies
- Identifying Facts and Workload
  - Studying existing reports
  - User requirements as typical analysis query workload
    - Keywords: 'by', 'for'
  - Identifying facts with conformed dimensions
  - User Requirement Glossary

Design Methodology



Requirement Analysis

- Collect end user needs for data mart applications and usage
  - "What information is needed by users in decision making?"
  - "Are the information available in operational databases?"
- Model the information needs of end users using the multidimensional data model:
  - Identify Facts, Measures, Dimensions
- Significant because it influences many things:
  - Conceptual design
  - Data-staging design
  - Requirement for data analysis applications

Difficulties

- Data warehousing is a long term project. Difficult to get and arrange every requirement from the beginning
- Requirements may be difficult to explain because decision-making processes ..
  - have very flexible structures,
  - are poorly shared across large organizations,
  - are guarded with care by managers, and
  - may greatly vary as time goes by to keep up with new business process evolution.

Typical decision-making queries

How much did my new product generate
month by month, in the southern division, by customer demographic, by sales office, relative to the previous version, and compared to plan?

The marketing vice president wants the revenue numbers broken down by month, division, customer demographics, sales office, product version, and plan.

Give me sales statistics
by products, summarized by product categories, daily, weekly, and monthly, by sale districts, by distribution channels.

The marketing manager wants sales statistics (e.g. sold quantity, receipts) of products, broken down by product categories, time (day, week, month), sale districts and distribution channels.

Modeling Business Measurements

"What are the gross margins by *product category* for *January*?"

"Show me the trend of monthly sales receipts for *'soft drinks' last year*."

- To answer these queries, we need to make measurements of 'Sales' business process.
  - Two measures 'gross margin' and 'receipts' in the fact 'Sales'
- Measurement is only meaningful given some context. E.g. the sales receipts of 'text books' in 'Jan 2013' is $23000
  - Dimensions are date (month, year), product (product category)
  - Also used to filter, sort, and group measures

Exercise

- What are the facts, measures and dimensions in the following queries?
  - "What are the gross margins by *product category* for *January*?"
  - "What is the average account balance by *education level*?"
  - "How many sick days were taken by *marketing employees last year*?"
  - "What is the return rate by *supplier*?"

Characteristics of Analysis Queries

- Decision-making queries usually involve
  - Analysis along one or more dimension, e.g. "What are the daily receipts per store?"
  - Aggregation / summary of numerical data from a large amount of data, e.g. "What is the total amount of receipts recorded last year per city and per product category?"
- It is not intuitive to represent multidimensional data in a 2D data model like the relational model

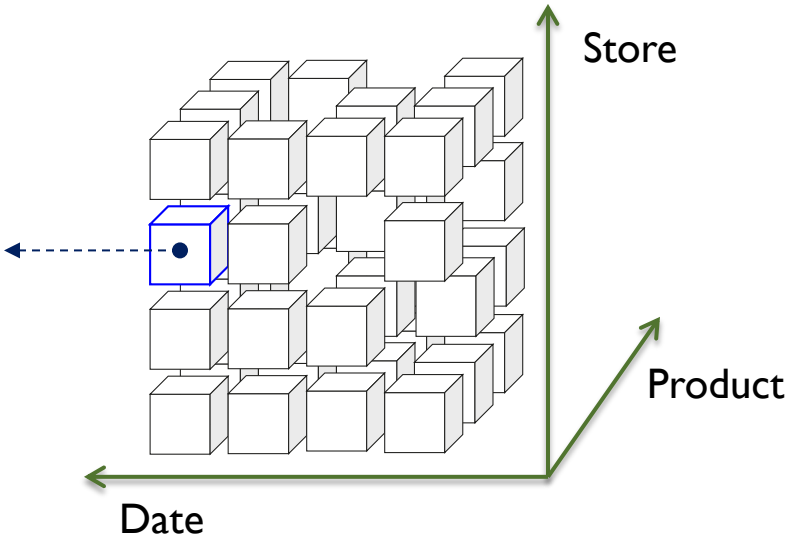
Give me sales statistics
by products, summarized by product categories, daily, weekly, and monthly, and by store location.

Multidimensional model

- The multidimensional model represents business data in multidimensional cubes
- Example: A cube for the fact *Sales*, with dimensions *Date*, *Product*, *Store*, and measures *quantity* and *receipts*
  - Notice that some cells in the cube are empty. This cube is sparse.
  - In general, the 'cube' may have >3 dimensions

Dimensions	**Date**	1/13/2012
	Store	EverMore
	Product	Coca cola
Measures	**quantity**	2
	receipts	$8.00

An event in the Sales fact



Concepts

- Facts are concepts on which end users base their decision-making process. Facts refer to a category of events taking place in the business.
  - E.g. sales, shipments, hospital admissions, surgeries
- Instances of a fact correspond to events that occurred.
  - E.g. every single sale or shipment carried out is an event
- Each fact is described by the values of a set of relevant measures that provide a quantitative description of events.
  - E.g. sales receipts, amounts shipped, hospital admission costs, and surgery time
- The large number of events may be selected and sorted out by different dimensions.
  - E.g. Sales in a store chain can be represented in a 3D space whose dimensions are products, stores (geography), and dates (time).

Examples: Facts and Dimensions

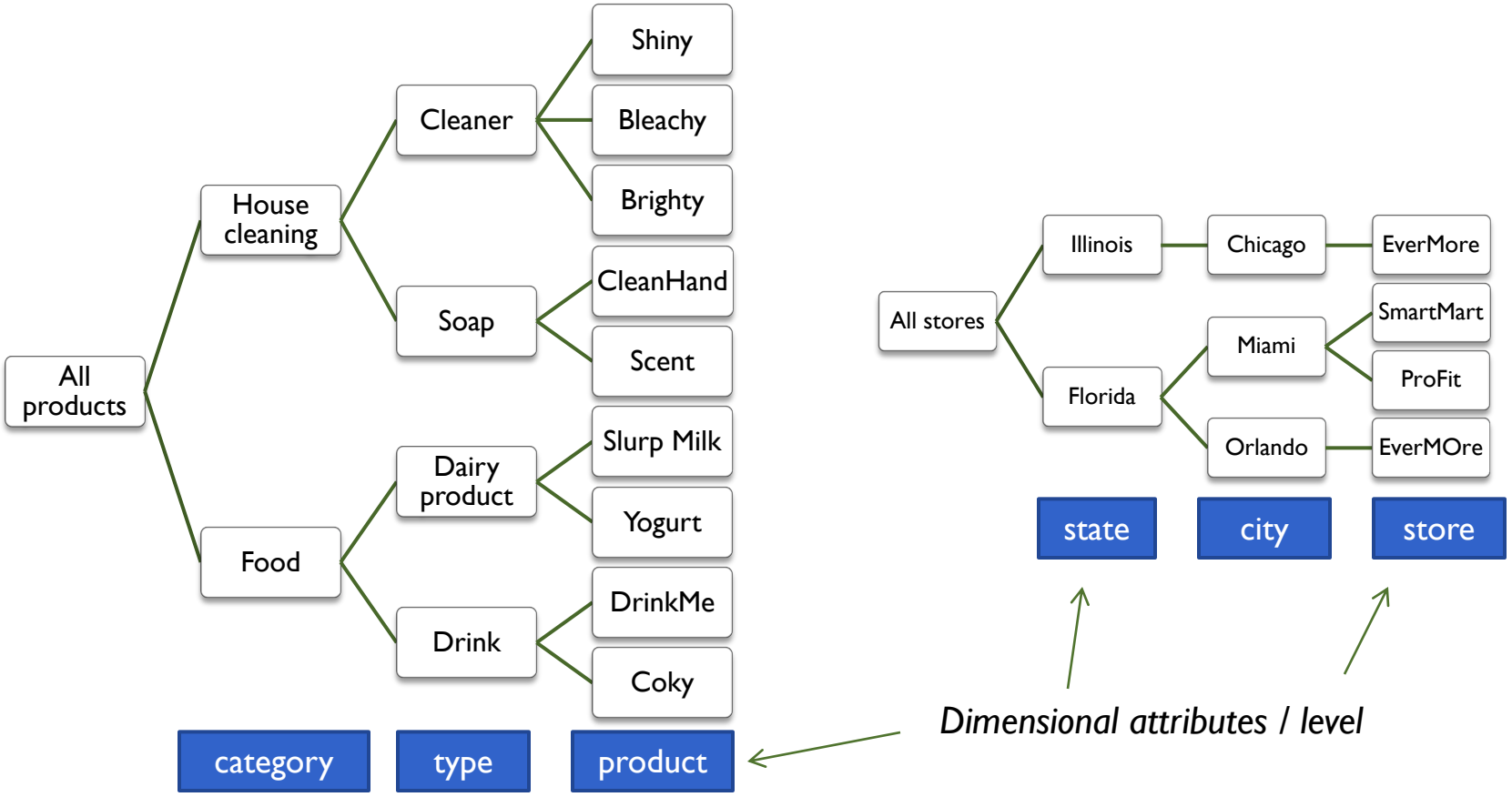
Business	Facts	Dimensions
Supermarket Chain	Sales	Time, Product, Store, Promotion
Insurance Business	Claims	Time, Claim, Insured Party, Policy, Status, Agent
Manufacturing Company	Shipments	Time, Product, Cust ship-to, Ship from, Ship mode, Deal
Airlines Company	Frequent Flyer Flights	Time, Customer, Flight, Fare class, Airport, Status

- Example: one would like to analyze claims data by agent, individual claim, time, insured party, individual policy, and status of the claim
- Time is a common dimension
- Business dimensions are different and relevant to the industry and to the subject for analysis.

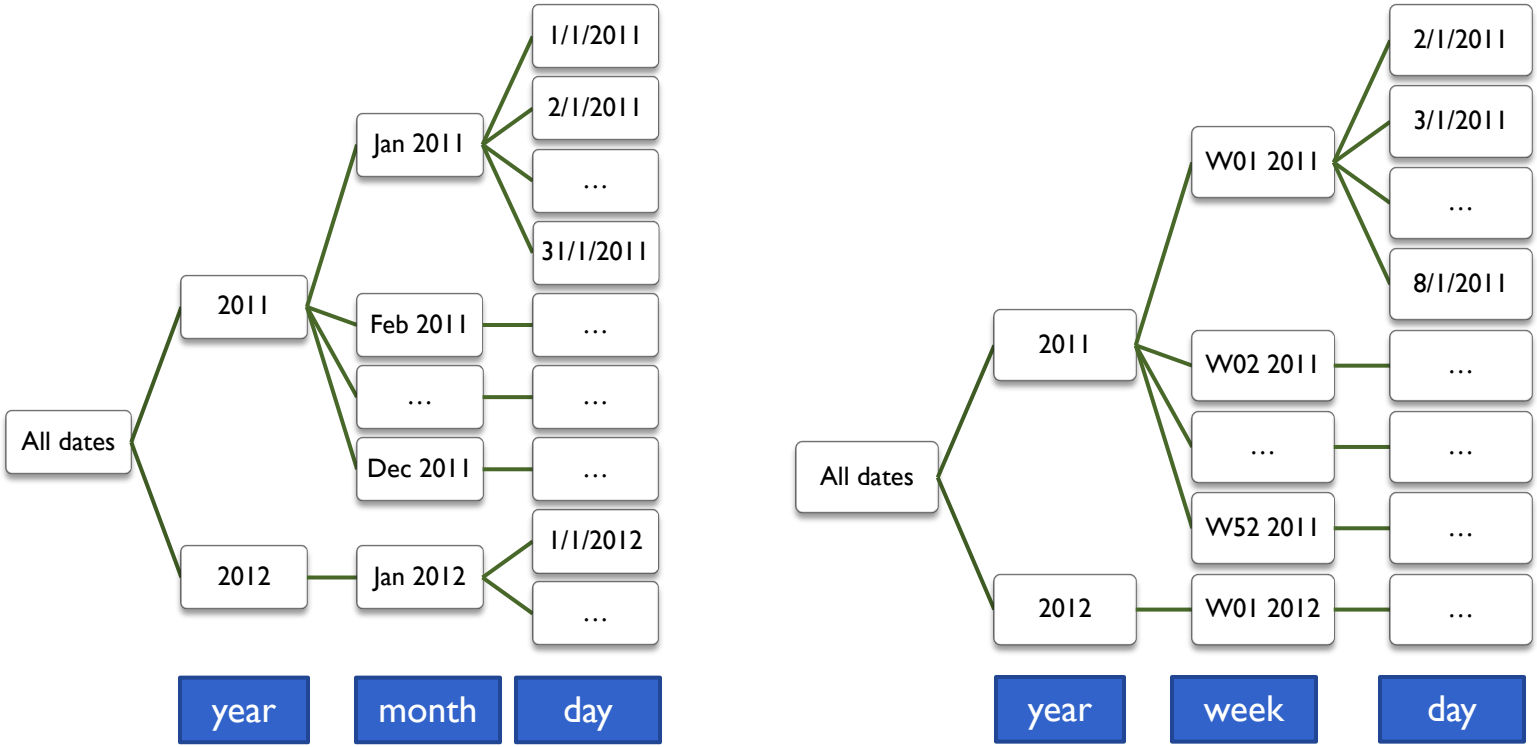
Dimensions: levels and hierarchies

- Each dimension is further described by many dimensional attributes
  - Also commonly known as levels
- Examples:
  - Levels for dimension 'Date': date, month, year
  - Levels for dimension 'Product': product, type, category
  - Levels for dimension 'Store': store, city, state
- Dimensional attributes organize instances of dimension in a hierarchy

Example: Hierarchies for Product and Store



Example: Hierarchies for Date



A dimension may have more than 1 hierarchy that organizes the instances in the dimension in different ways

Restriction and Aggregation using Levels

- Users usually perform data analysis with a subset of summarized data. A fact cube presents too many details.
- Two ways to reduce the amount of data
  - Restriction: e.g. show only sales data for last month in stores in a city
  - Aggregation: e.g. show total sales receipts by product types and by city (of the store)
- Both restriction and aggregation are achieved with levels

Example: Restriction

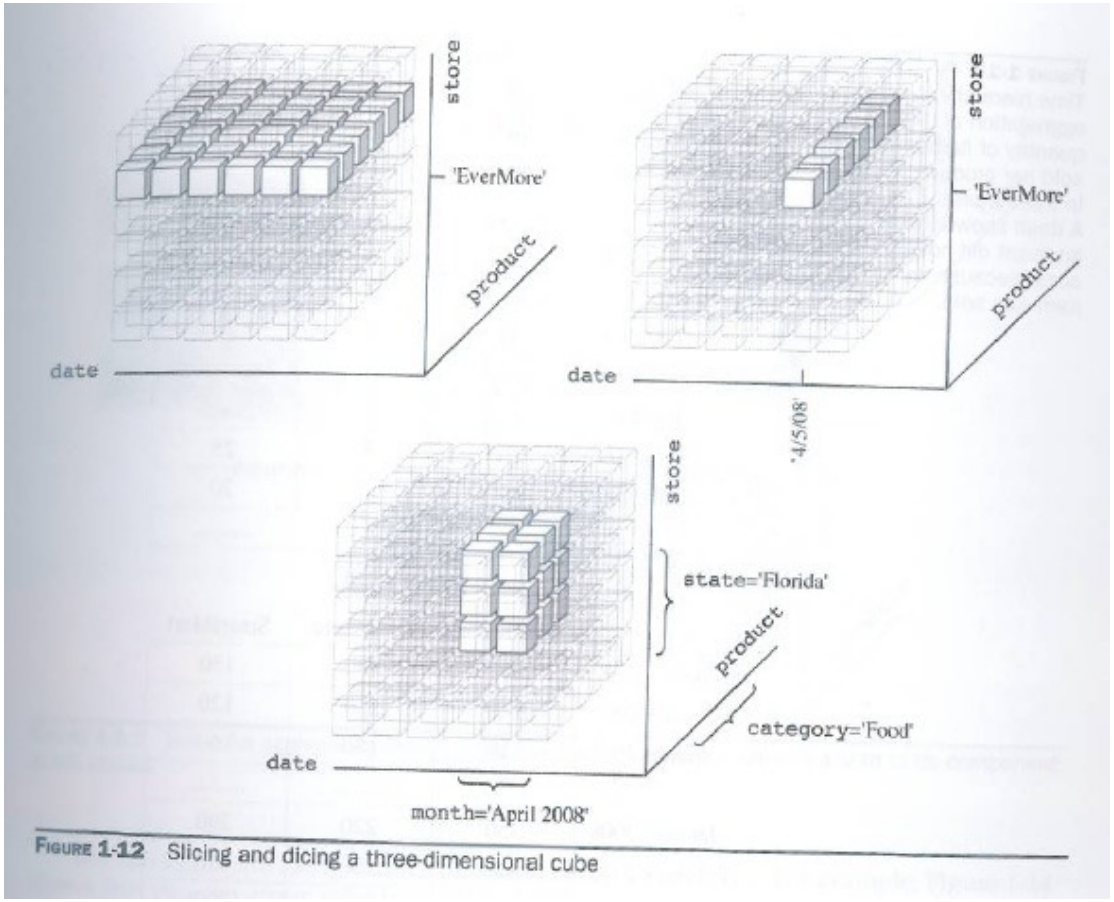


FIGURE 1-12 Slicing and dicing a three-dimensional cube

Example: Aggregation

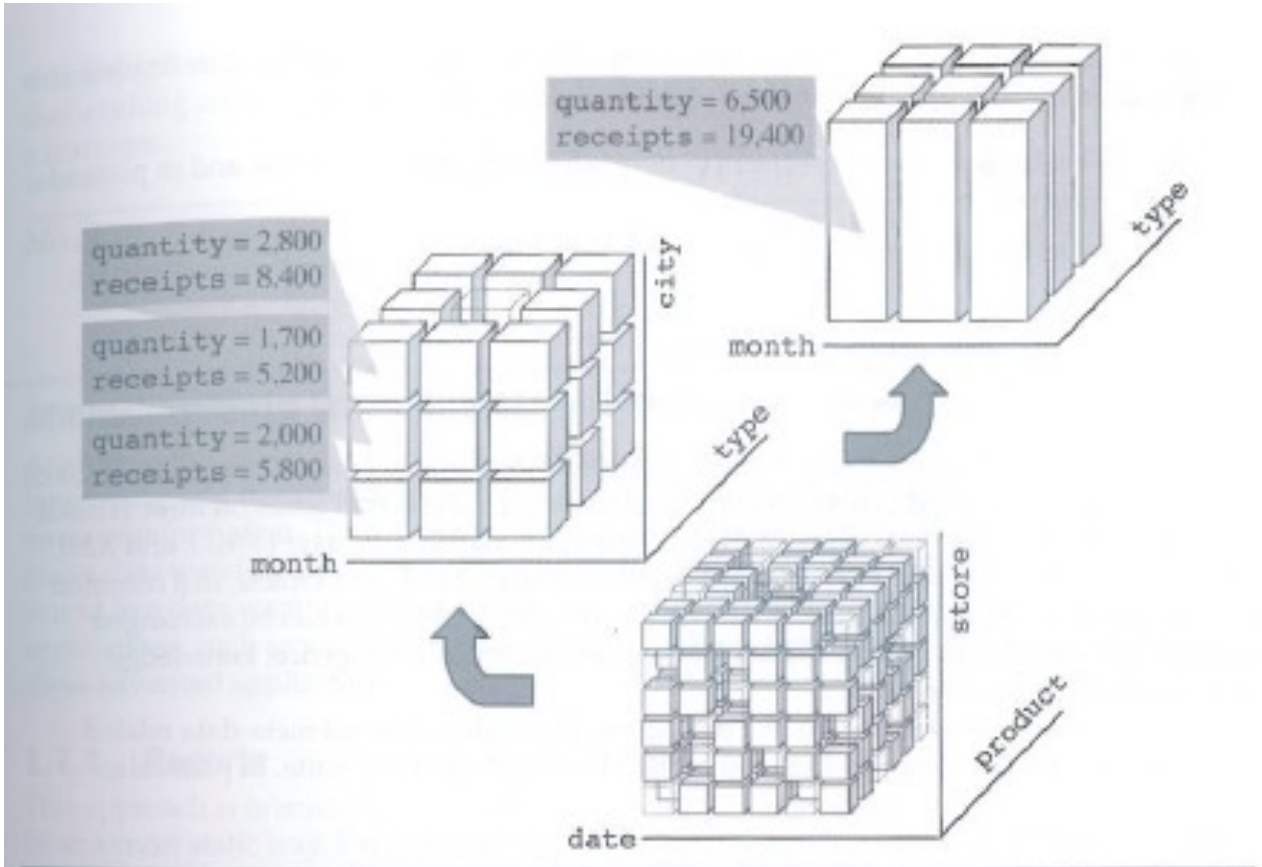
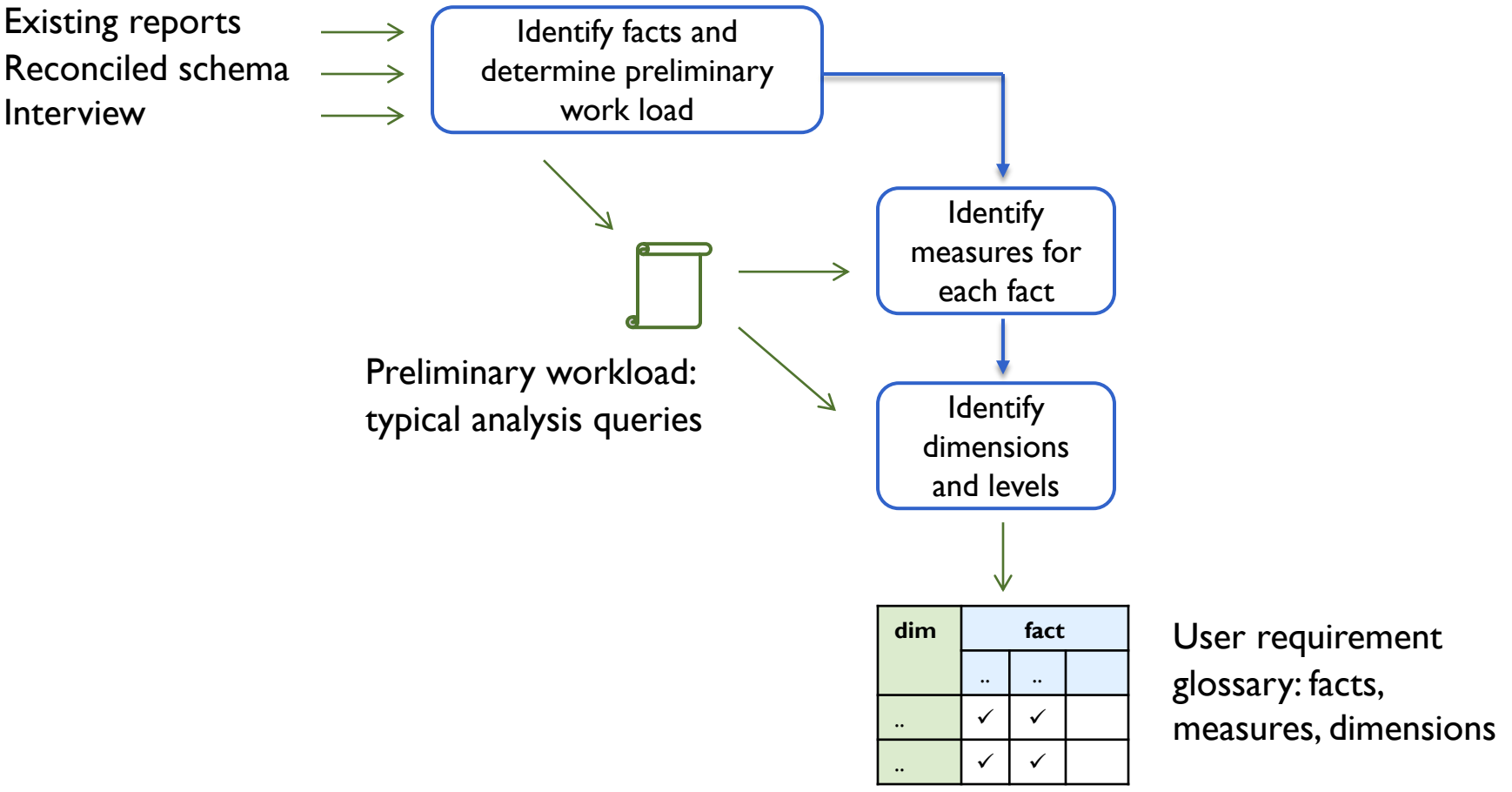


FIGURE 1-14 Two cube aggregation levels. Every macro-event measure value is a sum of its component event values.

Steps in Requirement Analysis



Identifying Facts

- Facts usually refer to business processes that the users want to measure
- Events described by facts should make reference to a time frame (usually there is a time-related dimension)

Application Field	Data Mart	Facts
Manufacturing	Supplies	Purchases, stock inventory, distribution
	Production	Packaging, inventory, delivery, manufacturing
Finance	Banks	Checking accounts, bank transfers, mortgage loans, loans
	Investments	Securities, stock exchange transaction
	Services	Credit cards, bill payment
Health service	Division	Admissions, discharges, transfers, surgical operations, diagnosis, prescriptions
	Accident & emergency	Admissions, tests, discharges
Management	Human resources	Hiring, resignation, firing, transfers

Reports

- We can decide facts, dimensions and measures by examining existing reports
  - Numerical data in the report are measures
  - Axes / headers are dimensions to sort out data

Ship location is a dimension to sort out data. Its levels include ShipCountry, ShipRegion and ShipCity.

Measures			Total Sales 2009
ShipCountry	ShipRegion	ShipCity	
Germany			$244,640.63
USA	NM	Albuquerque	$52,245.90
		NM	$52,245.90
	OR	Portland	$7,619.60
		Eugene	$19,711.13
		Elgin	$3,063.20
		OR	$30,393.93
	ID		$115,673.39
	MT		$1,947.24
	WA		$31,001.65
	WY		$12,489.70
	CA		$3,490.02
	AK		$16,325.15
		USA	$263,566.98
Mexico			$24,073.45
Switzerland			$32,919.50

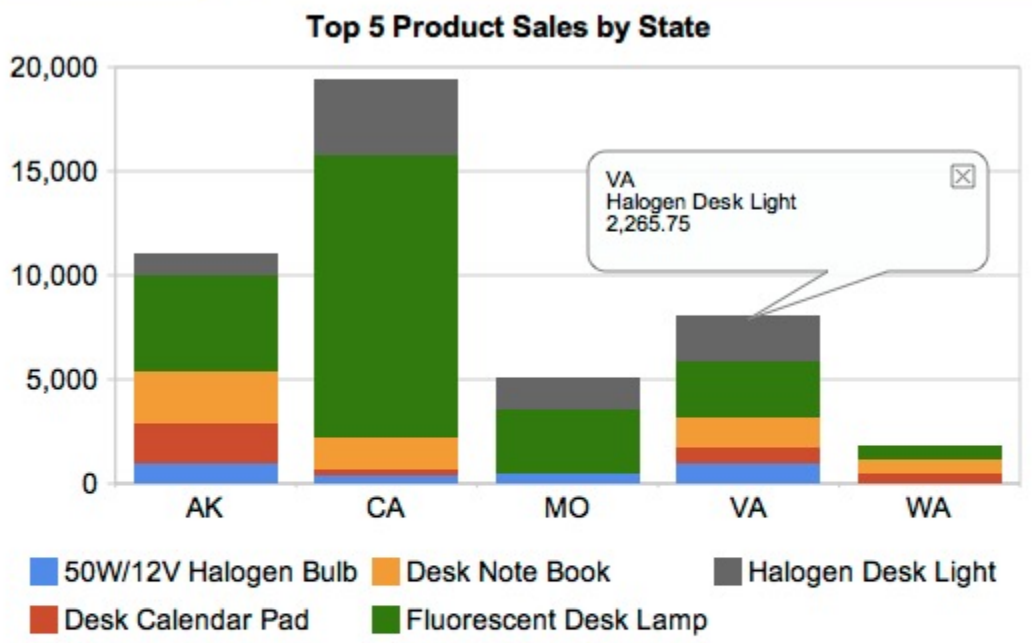
Date (level year) is a dimension for selecting data.

Fact is Sales. Measure is sales amount in dollars

Exercise: Reports with Charts

- This report shows the top 5 product sales in a large grocery chain. The y-axis shows the sales amount in dollar, and the x-axis is the location (state) of the stores.

- Fact is Sales
- Measure is sales amount in dollars
- Dimensions are … ?
- Their levels are … ?



Preliminary Workload

- Collect a set of users' specifications on analysis queries that is going to be issued to the data mart frequently.
  - through interview and examination of existing reports

Facts	Query
Stock inventory	What is the average quantity of each product made available monthly in every warehouse? Which product stocks ran out at least once last week at the same time in every warehouse? What's the daily trend of all the stocks grouped by product type?
Sales	What's the total amount per product sold last month? What are the daily receipts per store? What are the receipts per product category of a specific store on a specific day? What is the annual report of receipts per city per product?
Order lines	What's the total amount of goods ordered from a specific supplier every year? What's the daily total amount ordered last month for a specific product type? What's the best discount given by each supplier last year and grouped by product category?

Case: A national supermarket chain.

Identifying Measures

- Refer to preliminary workload for data required for analysis
- Refer to the reconciled layer for available data on the facts
- May need to summarize some data in case of different granularity
  - E.g. the operational data may keep data about each transaction from Point-of-sales. If the data mart only keeps daily receipts, then the receipts measure has to be aggregated from the operational database.
  - Note: 'receipts' refers to the sales amount in dollar

Identifying the Measures

Facts	Query	Measures
Stock inventory	What is the average quantity of each product made available monthly in every warehouse? Which product stocks ran out at least once last week at the same time in every warehouse? What's the daily trend of all the stocks grouped by product type?	Stocked quantity
Sales	What's the total amount per product sold last month? What are the daily receipts per store? What are the receipts per product category of a specific store on a specific day? What is the annual report of receipts per city per product?	Sold quantity, receipts
Order lines	What's the total amount of goods ordered from a specific supplier every year? What's the daily total amount ordered last month for a specific product type? What's the best discount given by each supplier last year and grouped by product category?	Ordered quantity, discount

Identifying Dimensions

- To identify dimensions for a fact, notice what are used to
  - select data for analysis
  - group data for aggregation
- In the data mart bus architecture, facts should share the same set of dimensions
  - Known as conformed dimensions
  - This allows joining two cubes in OLAP queries
- The levels of a dimension must satisfy queries about facts that share the dimension

Identifying the Dimensions (1)

Facts	Query
Stock inventory	What is the average quantity made available **monthly** in every warehouse? Which product stocks ran out at least once **last week** at the same time in every warehouse? What's the **daily** trend of all the stocks grouped by product type?
Sales	What's the total amount per product sold **last month**? What are the **daily** receipts per store? What are the receipts per product category of a specific store on a **specific day**? What is the **annual** report of receipts per city per product?
Order lines	What's the total amount of goods ordered from a specific supplier **every year**? What's the **daily** total amount ordered last month for a specific product type? What's the best discount given by each supplier **last year** and grouped by product category?

- Each fact typically has a time dimension (Date)
- Check the granularity of Date in selecting and sorting out the facts. This determines the hierarchy levels
  - Date levels: day, week, month, year

Finest granularity: one day

Identifying the Dimensions (2)

Facts	Query
Stock inventory	What is the average quantity made available monthly in every warehouse? **Which product** stocks ran out at least once last week at the same time in every warehouse? What's the daily trend of all the stocks grouped by **product type**?
Sales	What's the total amount **per product** sold last month? What are the daily receipts per store? What are the receipts per **product category** of a specific store on a specific day? What is the annual report of receipts per city **per product**?
Order lines	What's the total amount of **goods** ordered from a specific supplier every year? What's the daily total amount ordered last month for a **specific product type**? What's the best discount given by each supplier last year and grouped by **product category**?

- The three facts also share the Product dimension
- Check the granularity to determine the hierarchy levels
  - Product levels: product, type, category

Finest granularity: individual product

Identifying the Dimensions (3)

Facts	Query
Stock inventory	What is the average quantity made available monthly in **every warehouse**? Which product stocks ran out at least once last week at the same time in **every warehouse**? What's the daily trend of all the stocks grouped by product type?
Sales	What's the total amount per product sold last month? What are the daily receipts **per store**? What are the receipts per product category of a **specific store** on a specific day? What is the annual report of receipts **per city** per product?
Order lines	What's the total amount of goods ordered from a **specific supplier** every year? What's the daily total amount ordered last month for a specific product type? What's the best discount given by **each supplier** last year and grouped by product category?

- One more dimension for each fact:
  - Dimension warehouse, levels: warehouse
  - Dimension store, levels: store, city
  - Dimension supplier, levels: supplier
- No hints for levels for some dimensions
  - Assume finest granularity (e.g. individual warehouse and supplier)
  - May add some levels by using attributes in source data about the facts (e.g. location of a warehouse, and supplier)

Dimensions and Levels

Facts	Query
Stock inventory	What is the average quantity made available monthly in every warehouse? Which product stocks ran out at least once last week at the same time in every warehouse? What's the daily trend of all the stocks grouped by product type?
Sales	What's the total amount per product sold last month? What are the daily receipts per store? What are the receipts per product category of a specific store on a specific day? What is the annual report of receipts per city per product?
Order lines	What's the total amount of goods ordered from a specific supplier every year? What's the daily total amount ordered last month for a specific product type? What's the best discount given by each supplier last year and grouped by product category?

Dimensions	Levels in hierarchy
Date	day, week, month, year, …
Product	product, type, category, …
Warehouse	warehouse, …
Store	store, city, …
Supplier	supplier,

User Requirement Glossary

Fact	Measures	Dimensions
Stock inventory	stocked quantity	product, date, warehouse
Sales	sold quantity, receipts, discount	product, date, store
Order lines	ordered quantity, discount	product, date, supplier

Dimension	Levels in hierarchy
Date	day, week, month, year
Product	product, type, category
Warehouse	warehouse
Store	store, city
Supplier	supplier

Matrix Forms

Dimensions	Facts		
	Stock Inventory	Sales	Order lines
Date	✓	✓	✓
Product	✓	✓	✓
Warehouse	✓		
Store		✓	
Supplier			✓

Dimension	Levels
Date	day, week, month, year
Product	product, type, category
Warehouse	warehouse
Store	store, city
Supplier	supplier

Fact	Measures
Stock inventory	stocked quantity
Sales	sold quantity, receipts, discount
Order lines	ordered quantity, discount

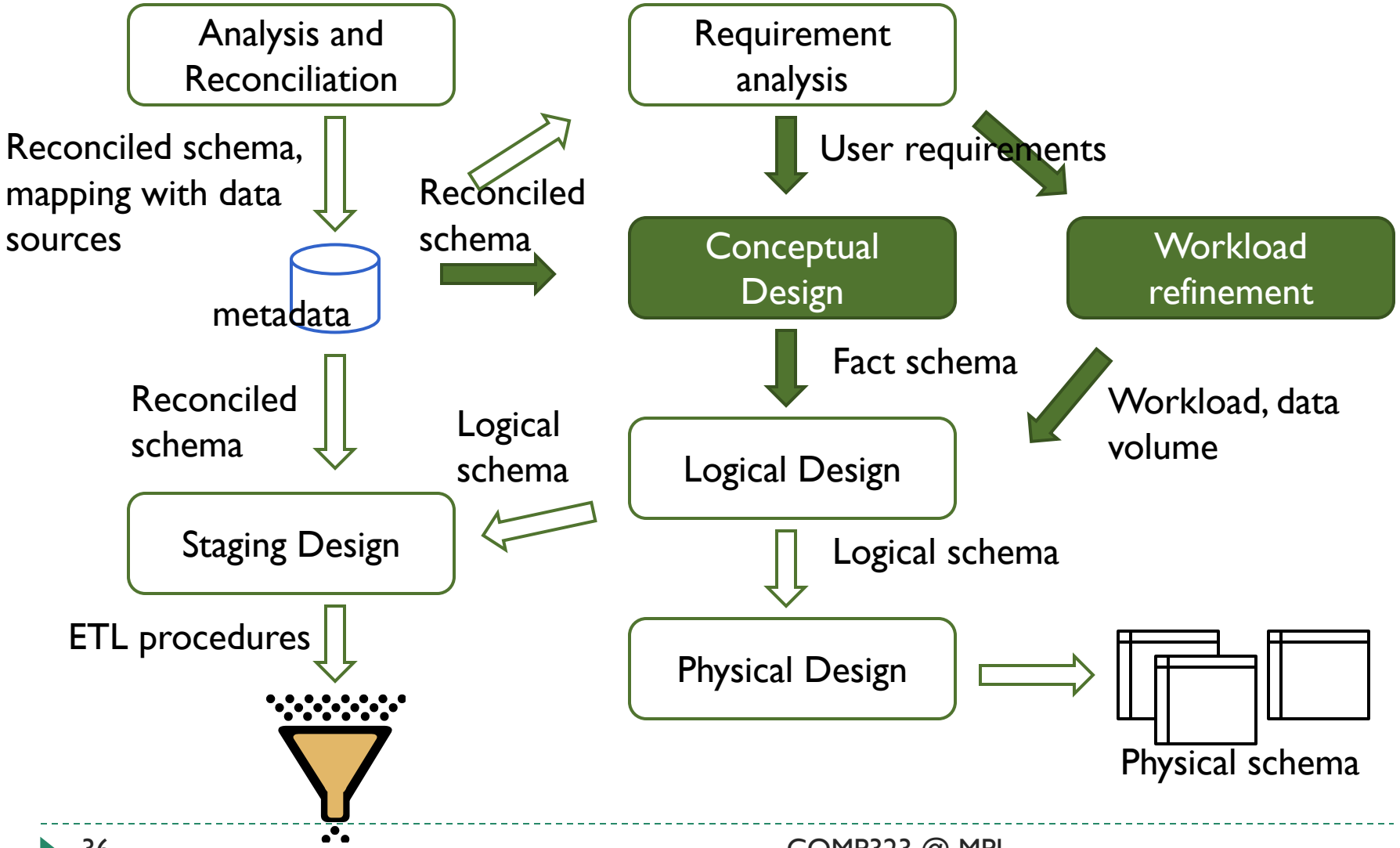
Review questions

- What are the essential differences between defining requirements for operational systems and for data warehouses?
- What are some difficulties in capturing user requirements in data mart design?
- Use an example to explain the multidimensional model.
- List some dimensions for fact(s) related to academic performance of university students.

Part B. Conceptual models, basics

- Conceptual models: What and why?
- Fact vs. dimensions
  - Should a value be a measure or a dimensional attributes?
  - Numerical value as dimensional attribute
  - Primary events and granularity
- Dimensions
  - Functional dependency among dimensional attributes.
  - Hierarchy of instances in a dimension
- More than one facts
  - Conformed dimension

Design Methodology



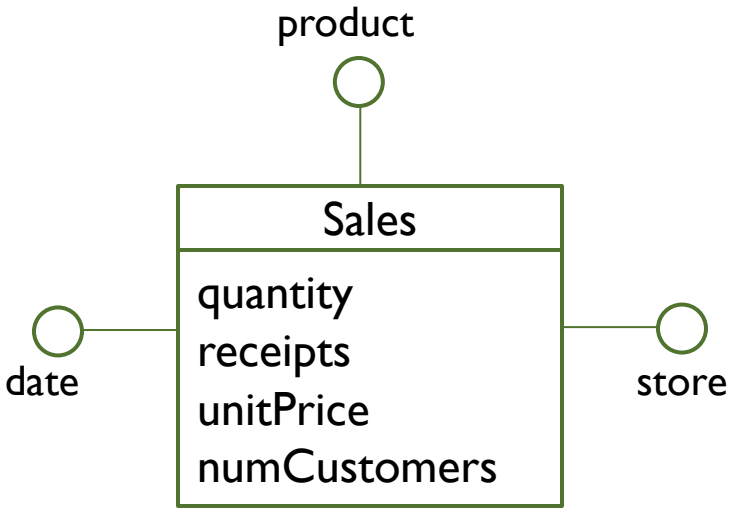
Conceptual Modeling

- A conceptual model captures concepts and relationship between them
- Although ER diagram is expressive enough to represent most concepts in multidimensional data, it cannot accurately highlight the distinctive features of the data.
- Some designers skip conceptual modeling and base their design on a logical model like the star schema. However,
  - It is less responsive to requirements, harder to maintain and reuse
  - Denormalizaton of dimension hides some functional dependencies
- We will use the Dimensional Fact Model (DFM) to do conceptual design

Steps in Conceptual Design

- Examine the user requirement glossary and schema of the reconciled data.
- Identify dimensional attributes (i.e. levels) and find all functional dependency between them. Diagram the relationship as a tree of attributes in the fact schema.
  - (Conformed dimension) Multiple fact schemas should share the same dimension for one concept (e.g. date, product.)
- Study the temporal nature of facts and dimensions.
- Define the measures of the facts and examine the additivity of each measure against each dimension.

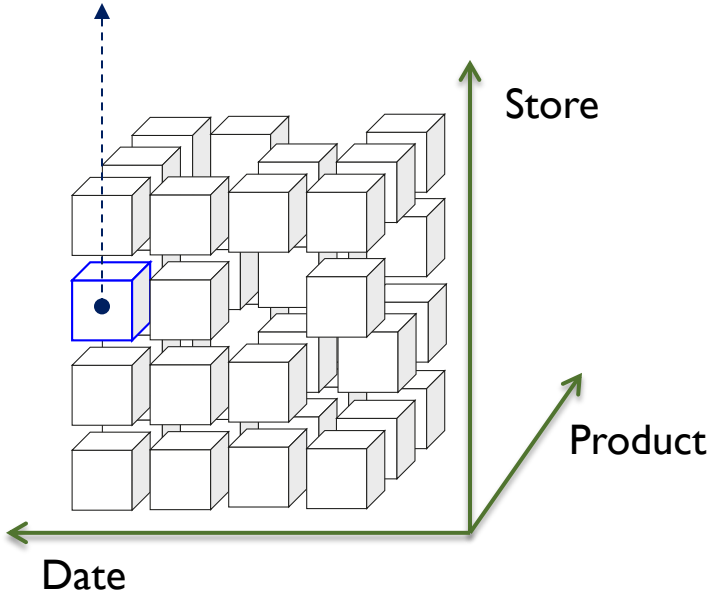
The Dimensional Fact Model



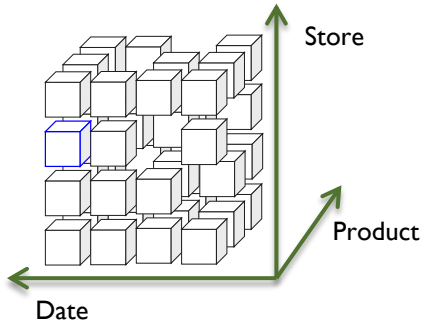
A fact schema in the DFM notation

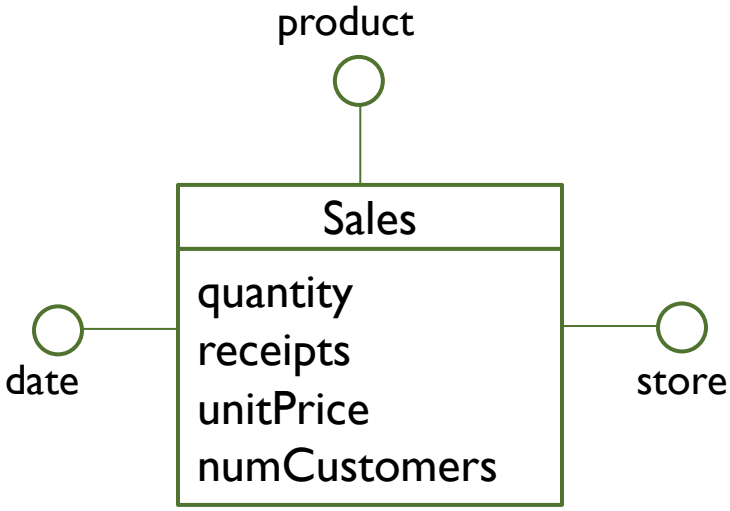
An event in the Sales fact

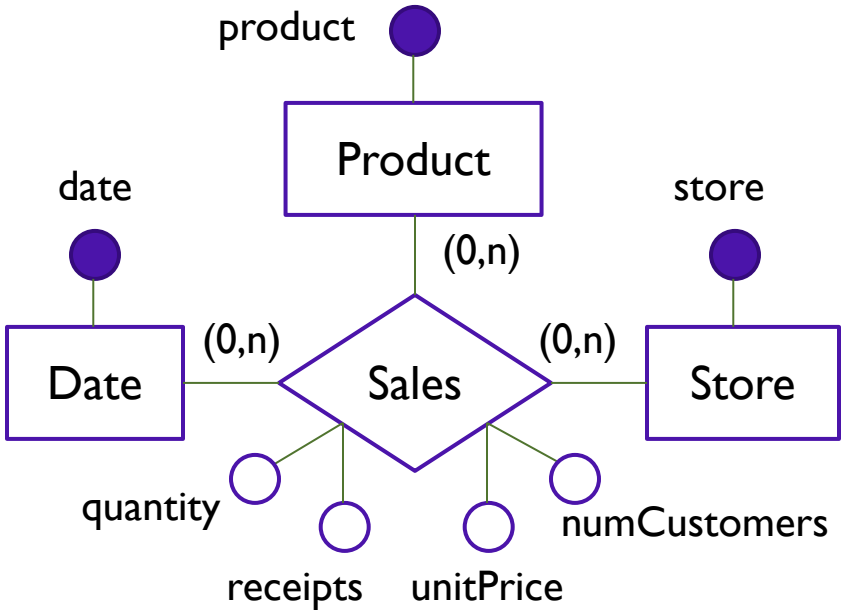
Dimensions	**Date**	1/13/2012
	Store	EverMore
	Product	Coca cola
Measures	**quantity**	4
	receipts	$16.00
	unitPrice	$4.00
	numCustomers	4



Comparison between DFM and ERM

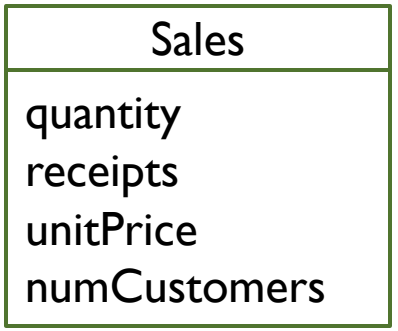






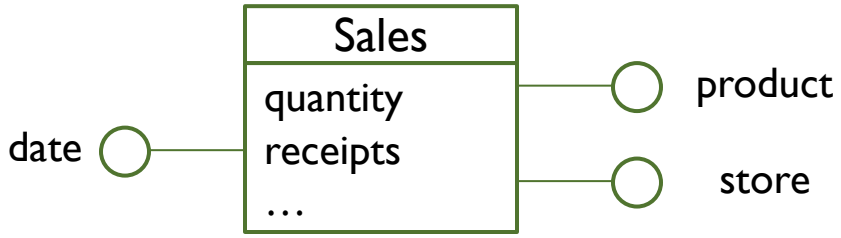
Concepts: Fact and Measure

- A fact is a concept relevant to decision-making processes. It typically models a set of events taking place within a company
  - A fact have dynamic properties or evolve in some way over time
- A measure is a numerical property of a fact and describes a quantitative fact aspect that is relevant to analysis
  - Measures are preferably numeric because it is easy to make calculation
  - Need to consider how to obtain measure values from operational data
  - An empty fact schema has no measures and records only occurrence of an event.



Concepts: Dimension

- A dimension describes an analysis coordinate of the fact.
- The dimensions of a fact define its minimum representation granularity.
  - E.g. The fact Sales, which has the dimensions product, store and date, can represent product sales in one store in one day. It cannot distinguish sales made by different customers or at different times of day.
- The grain statement describes the meaning of an event in the fact
  - E.g. On a certain date, some customers bought a certain product at a certain store. The number of items sold is the measure 'quantity' and the total sales account in dollar is the measure 'receipts'.
- Because facts are generally dynamic, a fact schema will almost certainly have at least one temporal dimension.



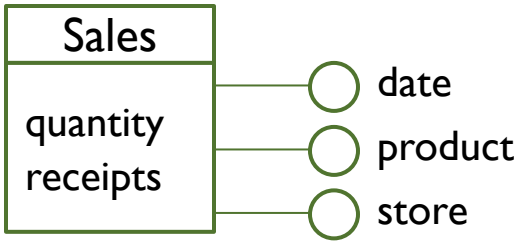
Concepts: Primary event

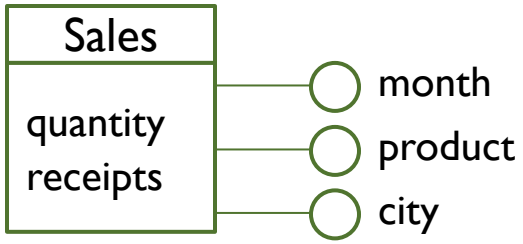
- A primary event is a particular occurrence of a fact, identified by one tuple made up of a value for each dimension. A value for each measure is associated with each primary event.
  - Depending on granularity of the fact schema, a primary event may not have one-to-one relationship with discrete events in the business.

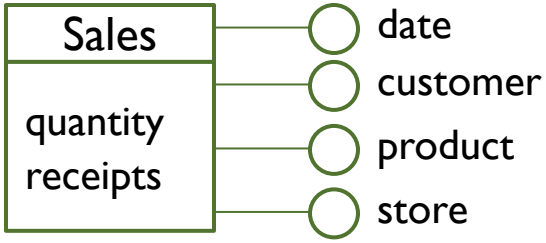
An event in the Sales fact

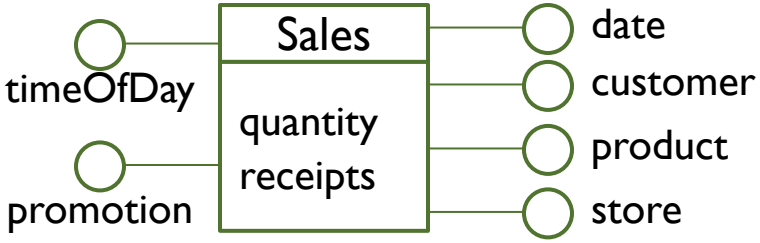
Dimensions	**Date**	1/13/2012
	Store	EverMore
	Product	Coca cola
Measures	**quantity**	4
	receipts	$16.00
	unitPrice	$4.00
	numCustomers	4

Example: Granularity of Facts









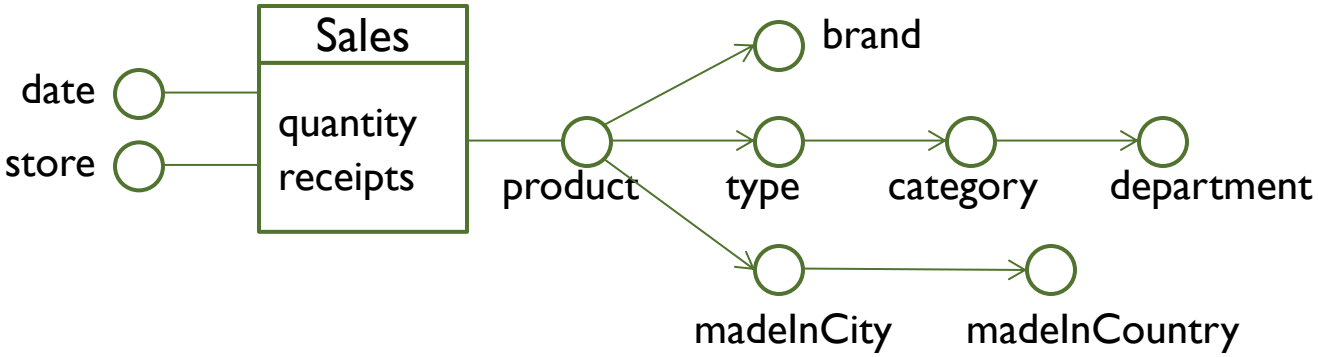
- Compare the granularity of four fact schema for analysis sales statistics in a supermarket chain. Write a grain statement to describe each case.
- Consider the case that a customer visits a store two times in a day and buy the same product. Can any of these fact schema distinguish the two transactions?

Concepts: Dimensional attributes

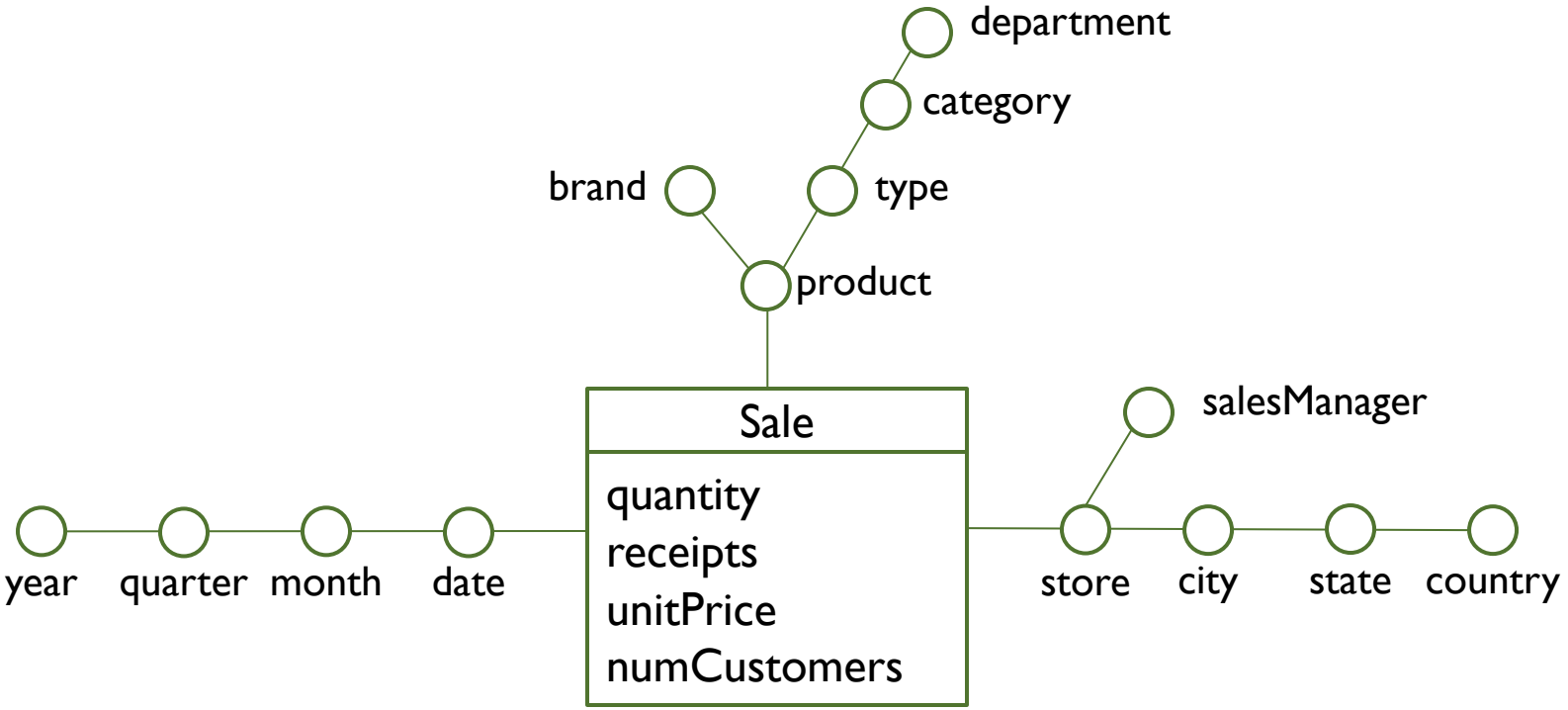
- Dimensional attributes describe instances of a dimension
  - E.g. the *product* 'coca-cola zero' belongs to the *type* 'soft drink', which is under the *category* 'drink'. The product has a *brand* 'coca-cola', and is sold in the *department* 'soft drink' of the supermarket.
- We use a unique name for the instances to represent the dimension in the schema. This name is also regarded as a dimensional attribute
  - E.g. the attribute *product* (sample value 'coca-cola zero') in the product dimension, or the attribute *date* (sample value '2015-01-22') in the date dimension
- Some dimensional attributes have many-to-one association. This is modeled as functional dependency in the relational model.
  - E.g. a specific product belongs to one brand, while one brand is associated with multiple products. The FD is *product* → *brand*

Dimension as a tree of attributes

- Dimensional attributes of a dimension can be modeled as a directed tree whose nodes are dimensional attributes and whose arcs model many-to-one associations (functional dependency) between dimensional attribute pairs.
- If there are no ambiguity, we may omit the arrow in the lines between dimensional attributes.

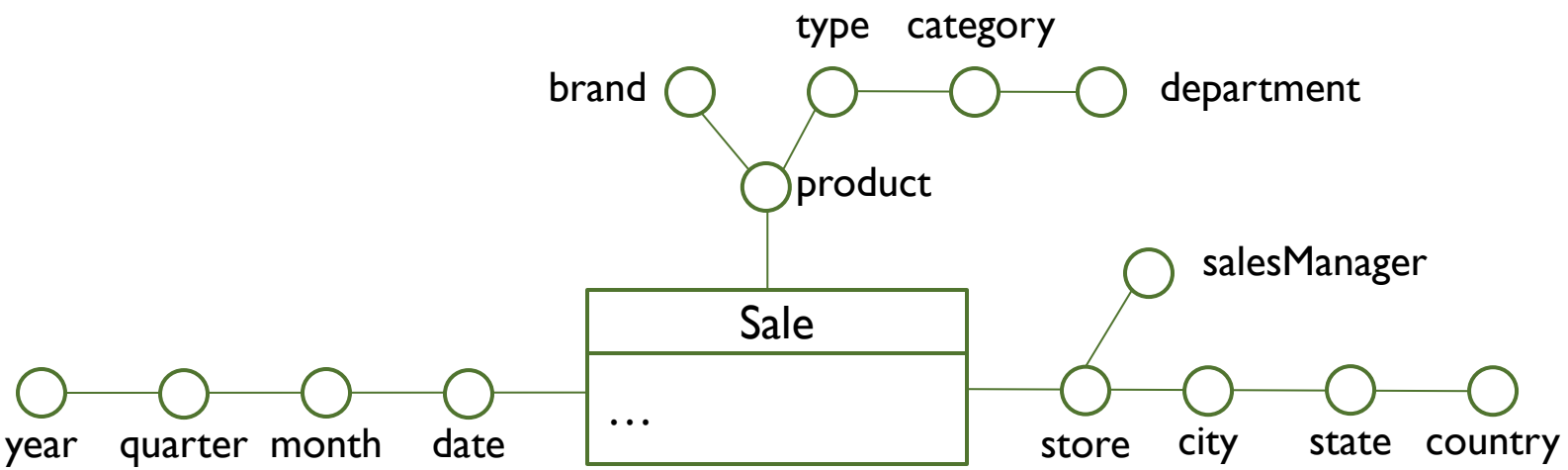


Example: a fact schema with trees of dimensional attributes



Dimensional attributes are structured as trees with their roots in dimensions. It is often not necessary to explicitly show arc directions as each arc is implicitly oriented in a direction moving away from the root.

Example: a primary event



Dimensions	**Date**	1/13/2012	**Month**	Jan 2012	**Quarter**	Q1 2012	**Year**	2012		
	Store	EverMore	**City**	Los Angeles	**State**	CA	**Country**	USA	**salesManager**	Peter Chan
	Product	Coca cola	**Type**	Soda	**Category**	Canned drink	**Department**	Beverage	**Brand**	Coca-cola
Measures	**quantity**	4								
	receipts	$8.00								
	...									

Notes on functional dependency, 1

Date	Month	Quarter	Year	DOW
1/1/2012	Jan 2012	Q1 2012	2012	Sunday
1/2/2012	Jan 2012	Q1 2012	2012	Monday
…				
4/10/2012	Apr 2012	Q2 2012	2012	Tuesday
…				
1/2/2013	Jan 2013	Q1 2013	2013	Wednesday

Notice the functional dependency date → month → quarter → year . We CANNOT write the month as 'January' alone. A month must functionally decides a year.

Notes on functional dependency, 2

- Functional dependency ensures many-to-one relationship between dimensional attributes. This is important to the roll-up and drill-down operations in OLAP.

		Year	2011	2011	2011	2011
		Quarter	Q1 '11	Q2 '11	Q3 '11	Q4 '11
Type	Product					
Dairy	Milk		20	25	30	30
Dairy	Yogurt		30	35	35	40
Drink	Coke		100	80	80	75
Drink	Juice		80	70	75	80

rollup

drilldown

		Year	2011
Type	Product		
Dairy	Milk		105
Dairy	Yogurt		140
Drink	Coke		335
Drink	Juice		305

rollup

drilldown

	Year	2011	2011	2011	2011
	Quarter	Q1 '11	Q2 '11	Q3 '11	Q4 '11
Type					
Dairy		50	60	65	70
Drink		180	150	155	155

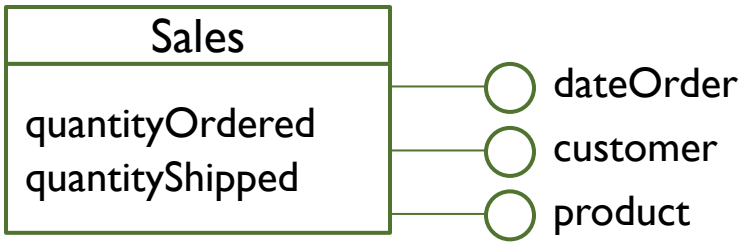
Numerical values as dimensional attributes

- Dimensional attributes usually have descriptive textual values
  - E.g. 'Jan 2013', '$10k-$20k', 'Beijing, China'
  - Simple to use as labels in reports and in SQL queries
  - Flags (boolean attributes) should be coded as readable strings, e.g. 'credit order' and 'not credit order'
- Sometimes, numeric data may be used to select and sort analysis data. These may be modeled as dimensional attributes
  - (Ex.1) to answer the query 'Compare the sales quantity of soft drinks by unit price in last quarter', sales quantity is a measure and unit price is a dimensional attribute.
  - (Ex. 2) to answer the query 'What are the changes in average unit price of soft drinks in the last 5 years?', unit price is a measure!

One or More Facts?

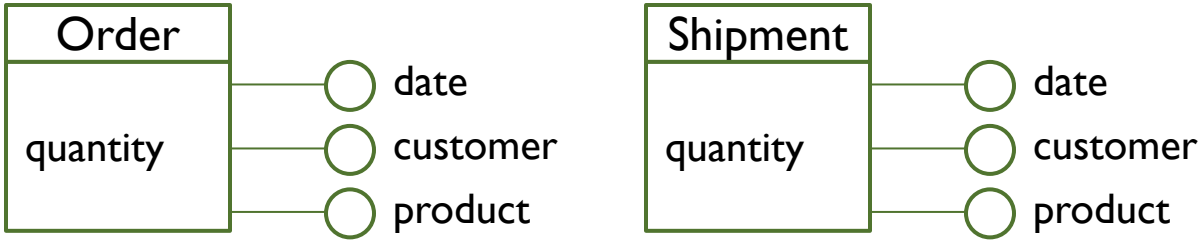
- To determine whether two measures belong to the same fact, consider two questions:
  - Do these measures occur simultaneously?
  - Are these measures available at the same level of detail?
- If either is 'false', the two measures belong to separate facts.

Example



Exercise: What's wrong with a single fact design? Split this into TWO facts: Order and Shipment.

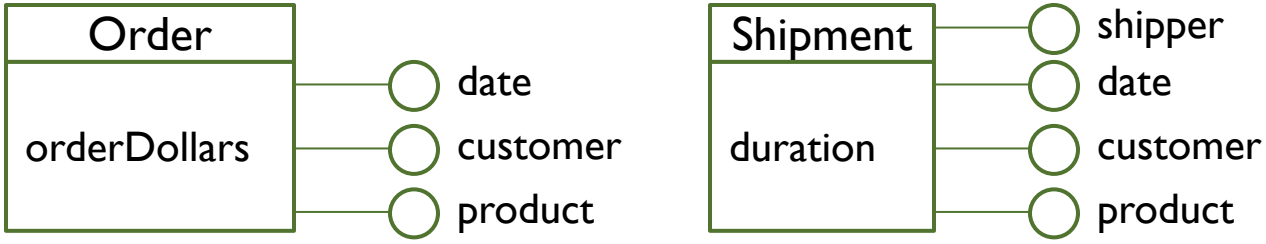
Consider a data mart that analyzes the quantity ordered and quantity shipped for an online shopping mall. We would like to analyze both values by Date, Customer and Product. But shipping usually happens *AFTER* order.



Date of order vs. date of shipment …

Example

- Assume that each purchase order only contains 1 product. It is shipped on the same day as the order. We want to measure how many days ('duration') it takes to ship the orders.
  - Analyze order dollar amount by Date, Customer, and Product
  - Analyze shipment duration by Date, Customer, Product, and Shipper
  - Why we need two facts?



Conformed Dimension and Drill-across

- To perform analysis across multiple facts, the facts must use conformed dimension.
  - The shared dimensions must have same structure and same value.
- A simple case is to use the same dimension.

Product	Quantity ordered
P111	100
P222	200
P333	50

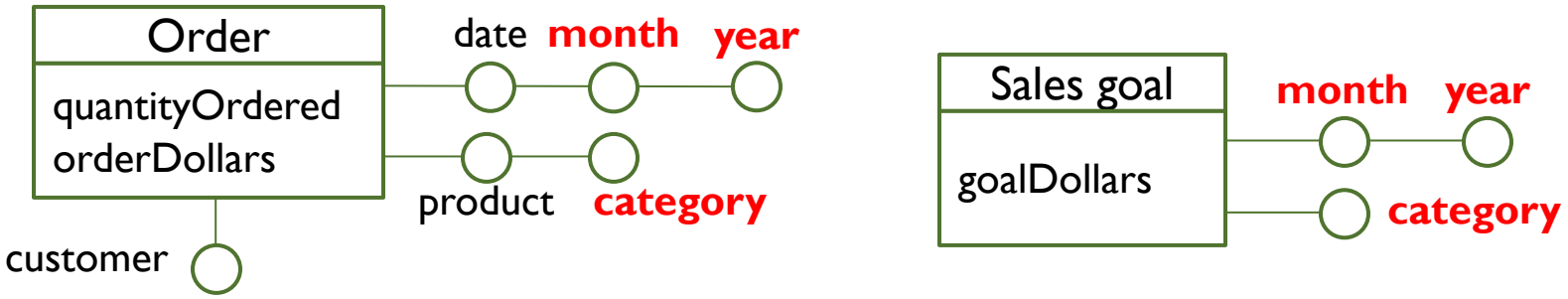
Product	Quantity shipped
P111	100
P222	150

The order and shipment in a certain month are merged on common dimensional attribute (product).

Product	Quantity ordered	Quantity shipped	Ratio
P111	100	100	100%
P222	200	150	75%
P333	50		0%

Advanced case for conformed dimension

- Identical dimensions are not required. Dimensions are conformed when
  - The dimensional attributes of one dimension are a subset of the dimensional attributes of the other.
  - The common dimension attributes share the same structure and content



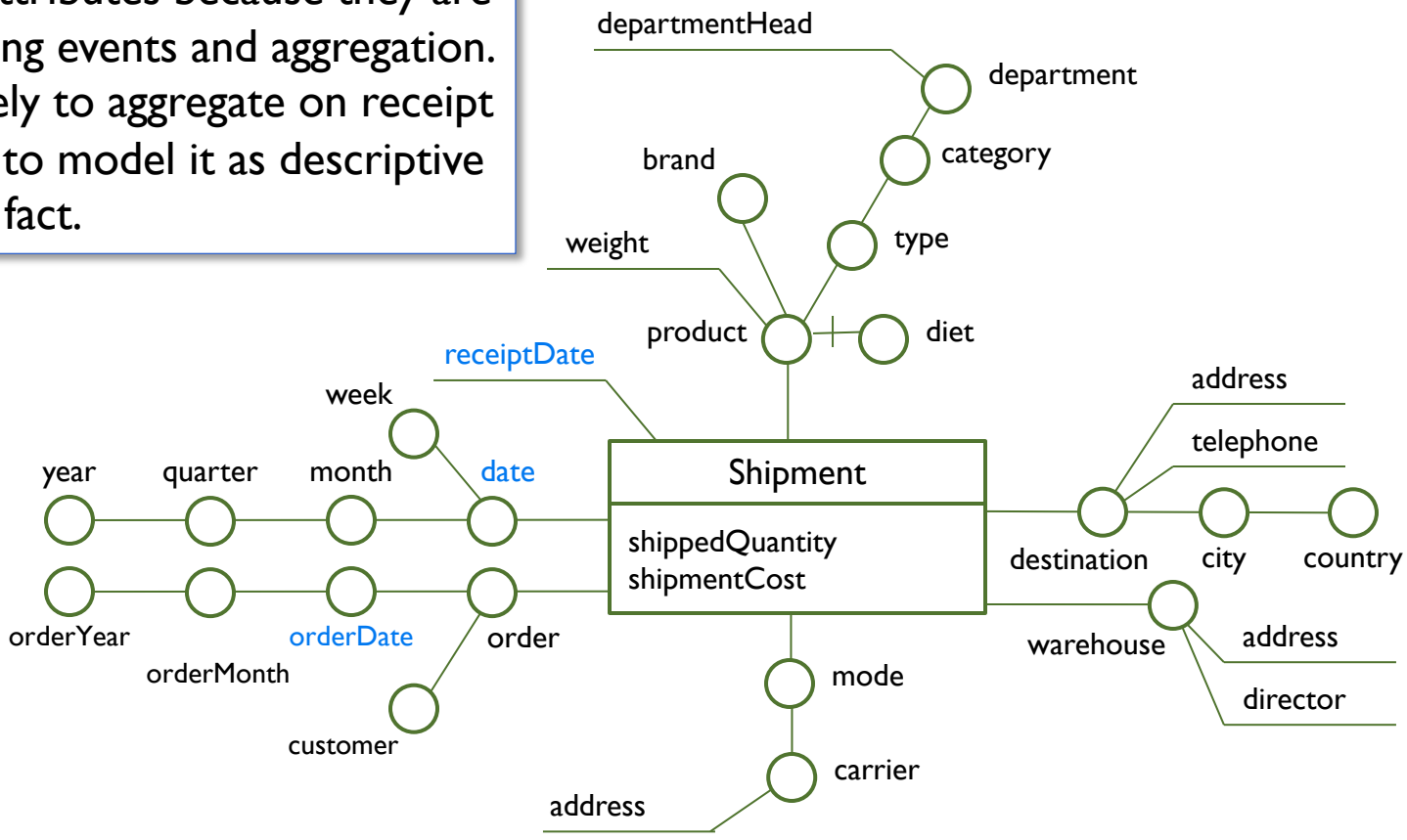
It is easy to generate a report to compare the order dollars and the sales goals by product category and month, because the two facts have conformed dimensions.

Descriptive Attributes

- Descriptive attributes are additional information that are not likely to be used as aggregation criteria
  - may appear on a dimensional attribute in a hierarchy or a fact
  - may be string values (e.g. store address)
  - may take continuous values (e.g. weight = 1.23kg) and won't be used for aggregation
- A descriptive attribute on a dimensional attribute is functionally determined by the dimensional attribute
  - E.g. weight of a product is fixed for a specific product
- A descriptive attribute on a fact describes a primary event

Example: Dimensional attribute or Descriptive attribute?

The order and shipping dates are modeled as dimensional attributes because they are useful for selecting events and aggregation. If we are not likely to aggregate on receipt date, it is better to model it as descriptive attribute on the fact.



Workload refinement

- Verify the conceptual schema against the preliminary workload.
- Some common problems in the verification:
  - Cannot aggregate data at the required level because of lacking dimensional attributes (e.g. in the Date dimension)
  - A false functional dependency between the attributes prevents a valid group-by set that can satisfy an analysis query (e.g. a FD between course and teacher prevents comparison of student performance taking the same course taught by different teachers)
  - User cannot select events the way they want to, because of lacking dimensional attributes (e.g. total sales during the CNY holiday in last 5 years, sales by credits vs. cash)
  - Missing measure. Can we derive it from existing measures or data in the operational data? Should we store it in primary events?

Case study: Data-driven Conceptual Design

- Define the conceptual schema for a data mart in relation to the structure of an operational data source (reconciled database)
  - The fact corresponds to one table or a join of several tables
  - A measure may be calculated from attributes in these tables
  - Dimension and levels (dimensional attributes) are associated with these tables by foreign keys (functional dependency)
- Refer to the preliminary workload for hints on measures, dimensions and levels.
- Verify the conceptual schema against the preliminary workload.

Problem

- Management in a large grocery chain needs to analyze sales statistics in each store. After examining existing reports and interviews, you compiled the following preliminary workload.

Preliminary workload for the fact Sales
What's the total amount per product sold last month?
What are the daily receipts per store?
What are the receipts per product category of a specific store on a specific day?
What is the annual report of receipts per city per product?
What are the total receipts in stores by each sales manager?
Which are the 5 best-selling products on Saturday's and Sunday's?
Which brands of soft drink generate the largest receipts during the Christmas season?

- The schema of the reconciled data from POS is given on the next page.

Schema of Reconciled Data

PRODUCTS (product, weight, size, diet, brand: BRANDS, type: TYPES)
STORES (store, address, telephone, salesManager,
    (districtNum, country): SALES_DISTRICTS, inCity: CITIES)
SALES_RECEIPTS (saleReceiptNum, date, store: STORES)
SALES (product: PRODUCTS, saleReceiptNum: SALE_RECEIPTS, quantity, unitPrice)
CITIES (city, state: STATES)
STATES (state, country: COUNTRIES)
COUNTRIES (country)
SALES_DISTRICTS (districtNum, country: COUNTRIES)
BRANDS (codBrand, producedIn: CITIES)
TYPES (type, marketingGroup: MARK_GROUPS, category: CATEGORIES)
MARK_GROUPS (marketingGroup, director)
CATEGORIES (category, department: DEPARTMENTS)
DEPARTMENTS (department, departmentHead)

Questions

- Perform conceptual design of the Sales data mart that satisfies the user requirements with the available data from the reconciled data.
  - Draw a fact schema
  - Write a grain statement to describe a primary event in the schema
  - Is this a transactional or snapshot fact schema? Are the measures additive?
  - How do you decide whether an attribute is a dimensional attribute or a descriptive attribute? Give rationale for one dimensional / descriptive attribute in your schema.

Review questions

- What are the concepts in conceptual modeling of data marts? What are some possible relationship between them?
- Explain the implication of functional dependency in the drilldown and rollup operations of OLAP.
- How do you decide whether an attribute is dimensional attributes or descriptive attributes?

Part C. Temporal Nature, Additivity and Aggregation

- Transactional Facts
  - Primary events as summary of business transactions
  - Properties: grain, sparse, additive measures
  - Benefits of additive measures
  - Handling non-additive measures
  - Special case: Fact with no measures
- Snapshot Facts
  - Periodic measurement
  - Semi-additive measures
  - Additive measures in snapshot facts

Two Kinds of Fact Schemas

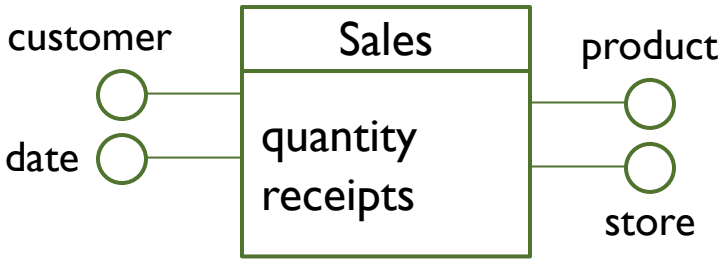
- Transactional fact schemas
  - Each primary event summarizes 1 or more activities in a business process
  - No primary event created if there are no corresponding business transaction
  - Use SUM to summarize the measures
- Snapshot fact schemas
  - Each primary event records the status measurement done periodically
  - Primary events have no direct correspondence with business transaction
  - Cannot use SUM in some situations

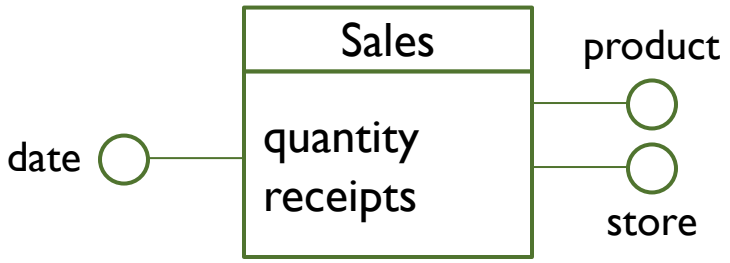
Transactional Fact Schema

- A transactional fact schema tracks the individual activities that define a business process and supports several measures that describe these activities.
- Properties of transactional fact schema
  - Grain: Lossy and lossless, Sparse
- Additive measures
  - Easy to summarize in rollup
  - Pre-calculated aggregates (materialized view)
- Handling non-additive measures
  - Ratio and Average: use stored additive components to calculate in query time
  - Use MIN,MAX,AVG to summarize non-additive measures

Lossy-grained transactional fact

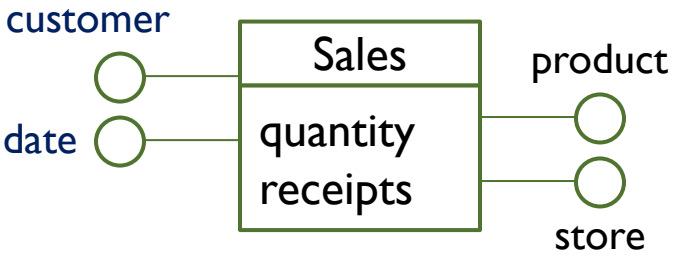
- In a lossy-grained fact, a primary event summarizes 1 or more business transactions
  - E.g. the Sales fact on the left records the total sale quantity and receipts of a certain product on a certain date in a certain store sold to a certain customer
  - Exercise: How many primary events are recorded in the two facts? On a certain day in store X, customer A buys 3 apples at 8:00. Customer B buys 3 oranges and 5 bananas at 9:00. Later, customer A visits again and buys 2 apples and 1 orange at 10:00.





Exercise

- How many primary events are recorded in the two facts? On a certain day in store X, customer A buys 3 apples at 8:00. Customer B buys 3 oranges and 5 bananas at 9:00. Later, customer A visits again and buys 2 apples and 1 orange at 10:00.

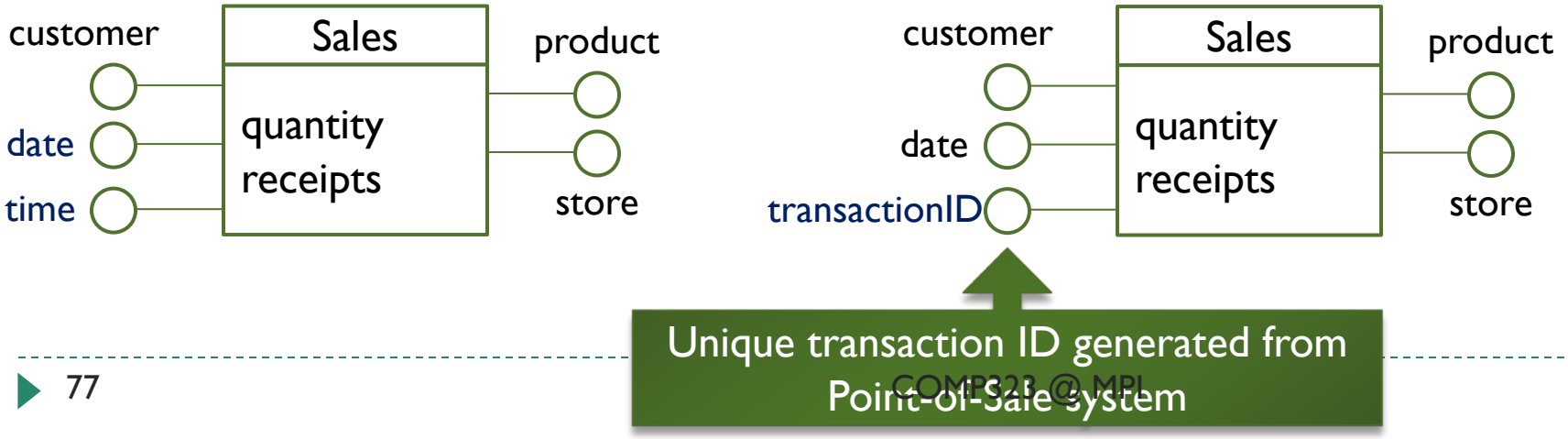


Primary events

Date	Store	Customer	Product	quantity	receipts
2015-01-01	X	A	apple		

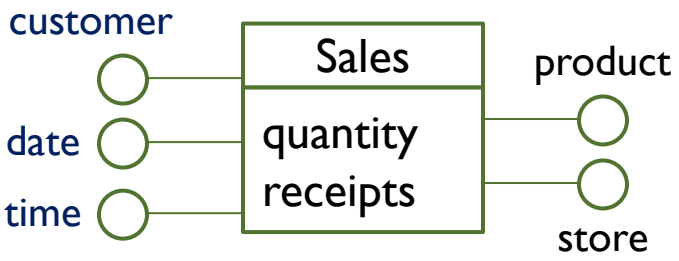
Lossless-grained transactional fact

- In a lossless-grained fact, a primary event records detail of each business transaction
  - E.g. the Sales fact on the left records the sale of a certain product at a specific time on a certain date in a certain store sold to a certain customer
- Fine granularity: usually requires a timestamp and/or transaction ID
- Exercise: repeat the exercise on previous page



Exercise

- How many primary events are recorded in the two facts? On a certain day in store X, customer A buys 3 apples at 8:00. Customer B buys 3 oranges and 5 bananas at 9:00. Later, customer A visits again and buys 2 apples and 1 orange at 10:00.

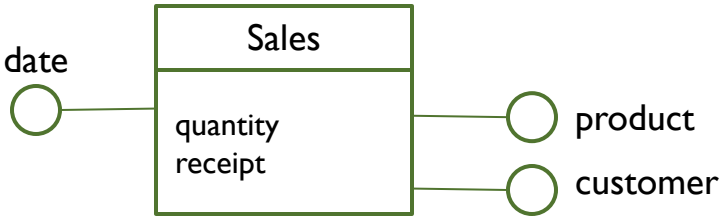


Primary events

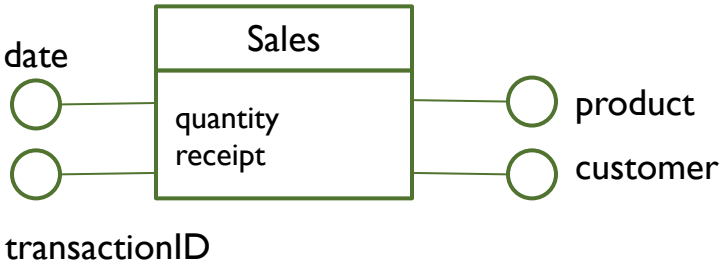
Date	Time	Store	Customer	Product	quantity	receipts
2015-01-01	8:00	X	A	apple		..

Example: transaction ID

- This fact cannot distinguish sale transactions of a product within 1 day to the same customer.

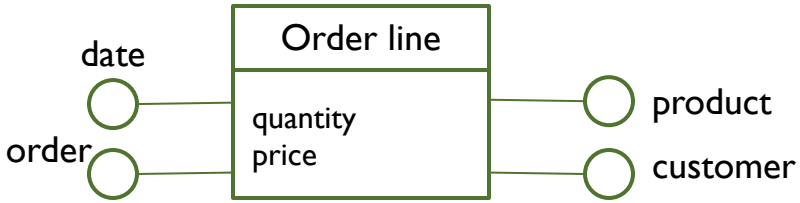


- This fact adds an transaction ID from POS system. Product sales in 1 shopping cart share the transaction ID.
- Sample queries that use the transaction ID:
  - What are the average receipts per shopping cart last month?
  - Break down the receipt per shopping cart by the product category.
  - How many transactions involve the product type 'soft drink', broken down by day-of-week?

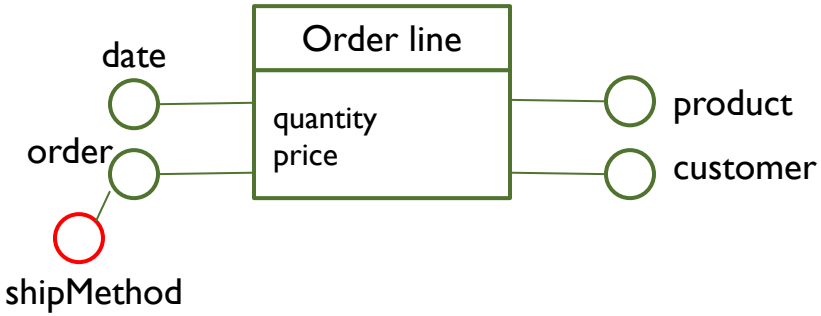


Example: Grouping orders

- An event in this fact represent 1 line in a purchase order. The order dimension is the order ID of the P.O.

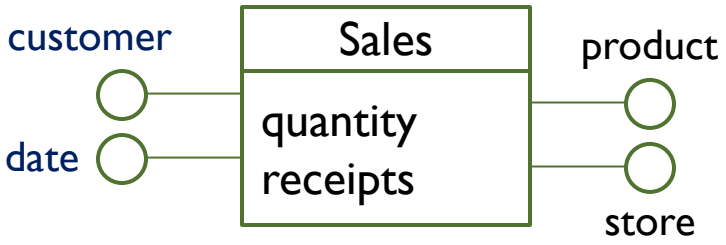


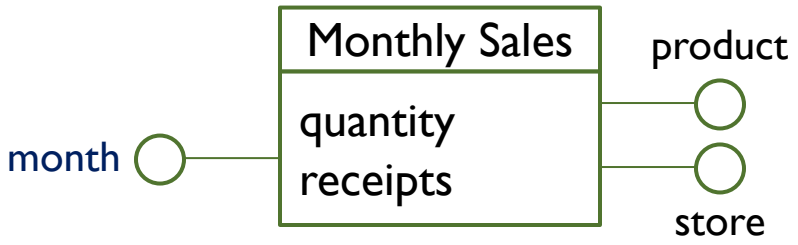
- We can add additional attributes to the order dimension to filter and aggregate P.O.s.



Sparse

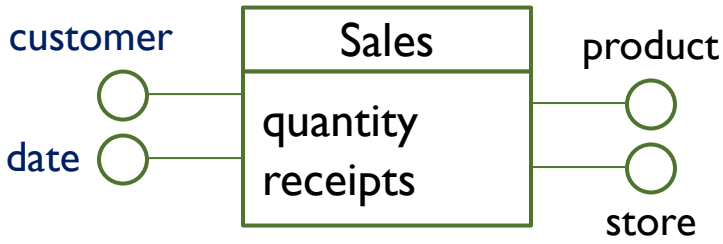
- Transactional facts are generally sparse
  - No primary events for many combinations of possible values of the dimensions.
  - E.g. only a few products are sold every day in all stores
- Absent primary events infer zero value for the measures
  - E.g. there are no primary events recorded if a product is not sold (quantity=0).
- Generally, a fact that records lower level of detail has a higher sparsity (i.e. lower density).





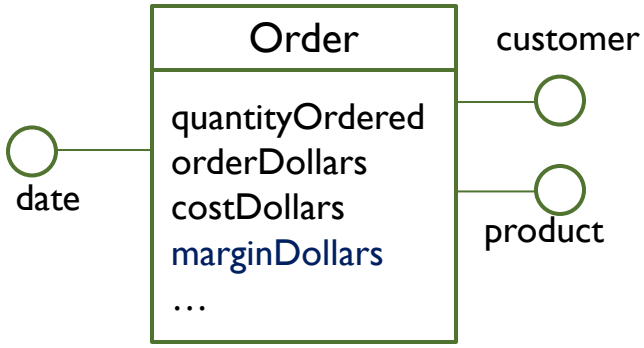
Additive Measures

- Measures in transactional facts are usually additive
- A measure is called additive when you can use the SUM operator to aggregate along any dimension.
  - i.e. the sum of measures of a group of events are meaningful
- Generally, we store additive measures in primary events.
- Measures that sum up count or dollar amount in a business transaction are usually additive
  - E.g. total quantity of apples sold in a store yesterday,
    total sales receipts from each customer in a store yesterday,
    total sales receipts from selling fruits in each store last month



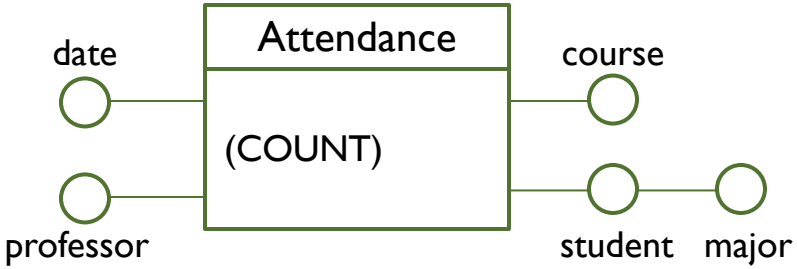
Differences are also Additive

- Differences between additive measures are also additive
  - E.g. margin $ = order $ - cost $
- Although such measures can be calculated at query time efficiently, storing them in primary events is often recommended because
  - Consistent definition of the measures, correct value available to all applications



Empty Fact Schema

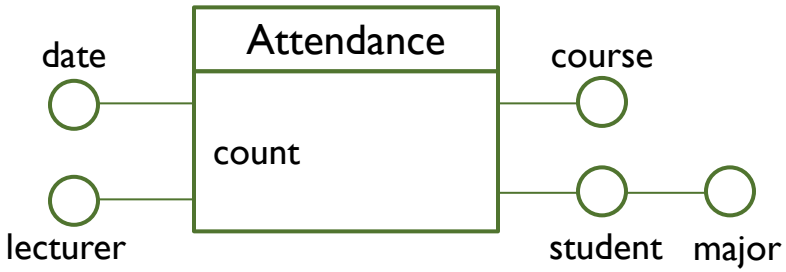
- An empty fact schema does not have any measures
- Each primary event records the occurrence of an activity, e.g. customer service request, click-thru count of online advertisement, web page view
  - Use COUNT to calculate the number of primary events in a secondary event



This fact can answer questions like "How many lectures does a certain student attend in each course?", "Compare the average attendance of courses.", "Which majors have a higher attendance rate?"

Alternative to empty fact schema

- Another way to model occurrence of events is to add a count measure and initialize it to 1 for all primary events
- Count is additive
- When aggregate the data, just calculate the total count.
  - E.g. "how many lectures does a certain student attend in each courses" becomes "the total count of attendance of a certain student in each course".
- Better support in OLAP engines…



Non-additive measures

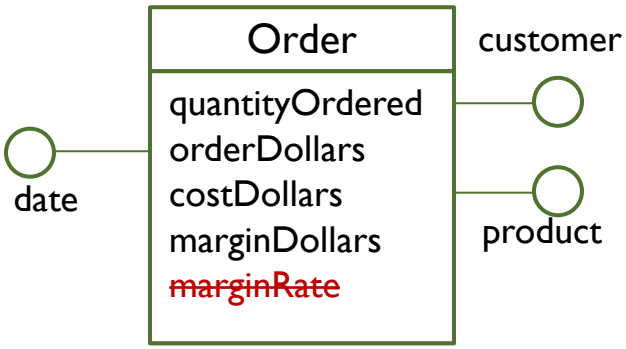
- Business users also request non-additive measures in analysis queries.
- Common examples: ratio and average
  - Ex.1 margin rate = ('total order dollars' – 'total cost dollars') / 'total order dollars'
  - Storing the margin rate in primary events is not useful adding the margin rate of several transactions does not give the margin rate of the transactions.
  - Ex.2 average order dollars = 'total order dollars' / 'total quantity ordered'
  - Storing the average order dollars of each transaction is not useful either.
- Basic strategy: store additive components in primary events, and calculate non-additive measures at query time. These are called calculated measures.

What measures to store in base fact?

- Stored measures are stored in primary events for a fact
  - Usually additive measures
  - (special attention when aggregating non-additive measures)
  - Stored in both primary events and aggregates
  - Are used in calculation of some calculated measures
- Calculated measures are *not* stored in primary events, but are calculated at query time
  - Usually not included in fact schema
  - Some common non-additive measures can be calculated from additive components that are stored in the fact …

Ratio and Percentage Measures

- Ratio and percentage are not additive themselves, but their components are.
  - Store the additive components as measures in fact schema
  - Do not store the ratio / percentage directly!



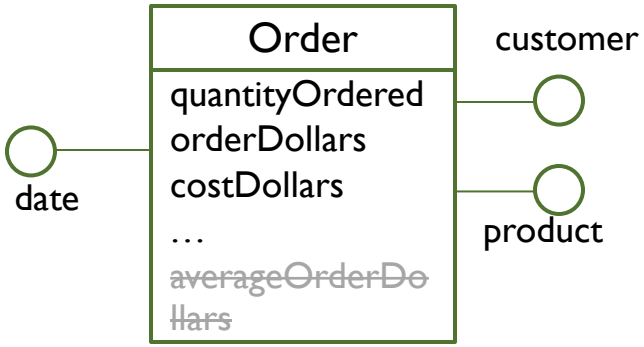
Product	Order $	Cost $	Margin $	Margin Rate
P111	100	80	20	20% = 20/100
P222	200	150	50	25% = 50 / 200
P333	50	45	5	10% = 5 / 50
All products	350	275	75	75/350 = 21%

'Margin rate' = (order$-cost$) / order$
margin$ = order$-cost$

Margin rate is not additive, but margin$ and order$ are additive.
Therefore we can store the measure margin$ in primary events.

Average Measures

- average = 'total' divided by 'count'
- Averages are not additive, but ‘total’ and ‘count’ are usually additive
- We should store 'total' and 'count' as measures to speed up calculation of averages.



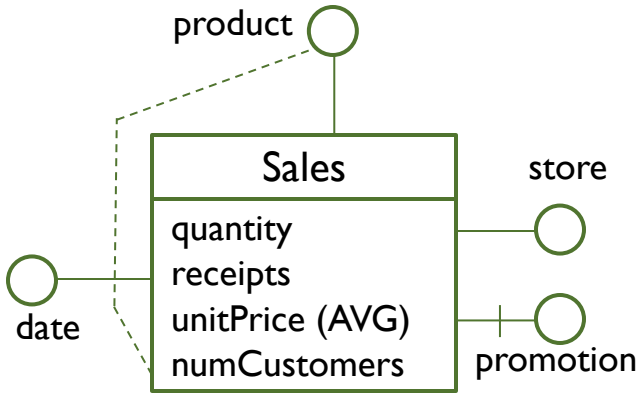
‘average order dollars’ = ‘order dollars’ / ‘quantity ordered’. It is not additive, but can be calculated in query time from two additive stored measures.

Advanced: storing non-additive measures

- Sometimes, non-additive measures are stored in primary events.
- A non-additive measure may be aggregated using other operators:
  - MIN, MAX, COUNT
  - AVG
  - However, some reporting tools may only support SUM
- Careful in aggregating the measures. Need to document the additivity of the measures.

Example

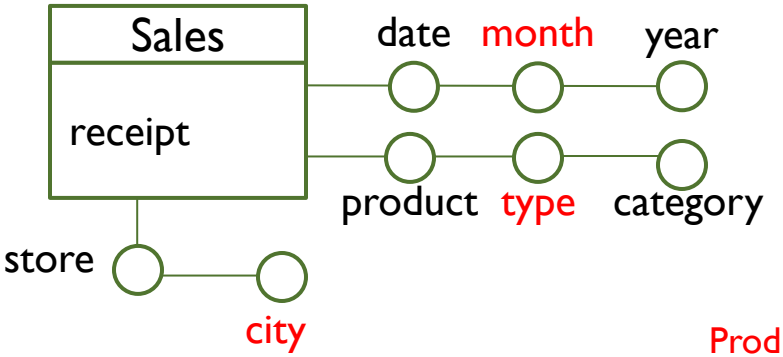
- The number of customers is estimated from the number of sale receipts issued on a day in a store for a certain product.
- numCustomers is not additive along Product, but additive along other dimensions. It is semi-additive.
- unitPrice is not additive along any dimension. But it's still ok to apply AVG, MIN, or MAX.



Example query: what's the average price of soft drinks sold in each city? What are the price range (min and max price) of soft drinks being sold in each store?

Aggregation and Secondary Events

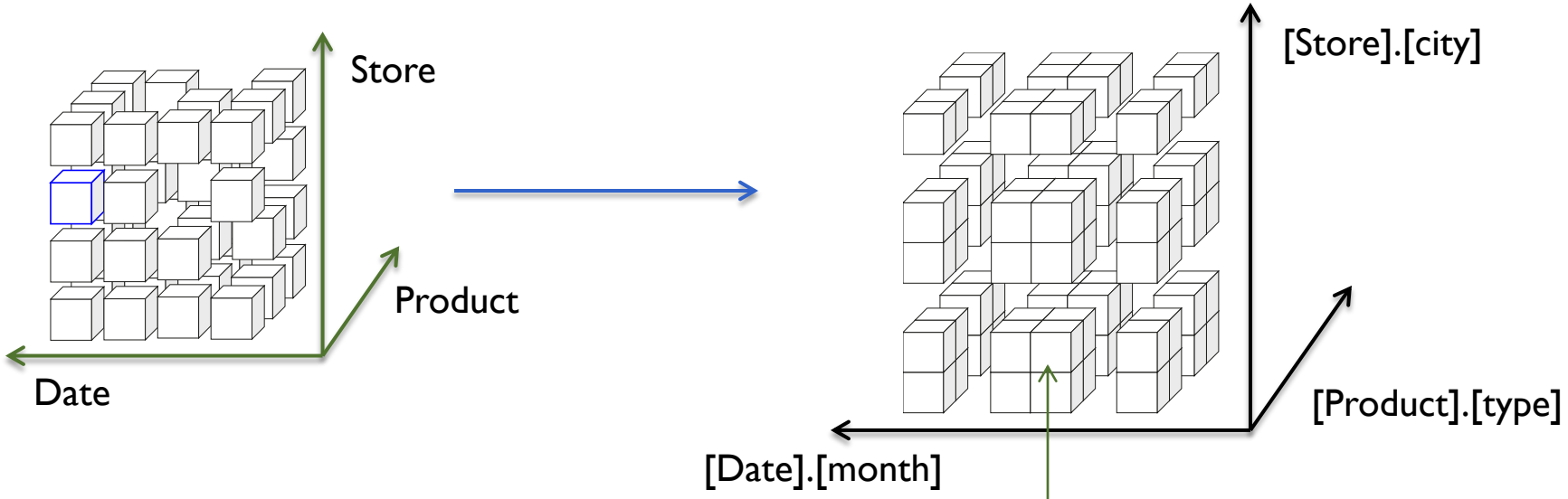
- Analysis often requires grouping primary events into sets of coarser granularity. These groups are known as secondary events.
  - E.g. to analyze the sales receipt of each product type at stores in each city in the last two months, we can group the primary events by product type, by store city and by months.



City	Macau		Hong Kong	
Month	Dec 2011	Jan 2012	Dec 2011	Jan 2012
Product Type				
Dairy	320	350	830	810
Drink	210	280	967	896

Total sales receipt by month, city and product type

Example: group by product type, store city and month



Primary events inside a secondary event

1/1, milk, hk store A, $10
1/1, cream, hk store B, $25
1/1, yogurt, hk store A, $25
1/2, milk, hk store B, $15
1/2, yogurt, hk store C, $30
1/3, milk, hk store D, $20
…
1/31, cream, hk store E, $42

Summarizing measures in secondary events

- A secondary event consists of a large number of primary events, each with a value for a measure.
- When the measure is additive, we can use the SUM operator to summarize additive measure values. Other operators can also be used.
- The query should use other operators (e.g. AVG, MIN, MAX) to summarize non-additive measure values

	City	Macau		Hong Kong	
	Month	Dec 2011	Jan 2012	Dec 2011	Jan 2012
Product Type					SUM
Dairy		320	350	830	810
Drink		210	280	967	896

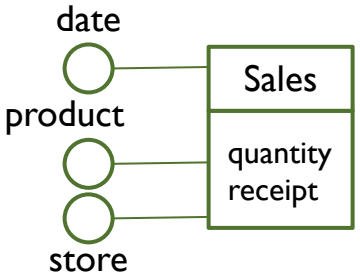
Total sales receipt by month, city and product type

Inside a secondary event

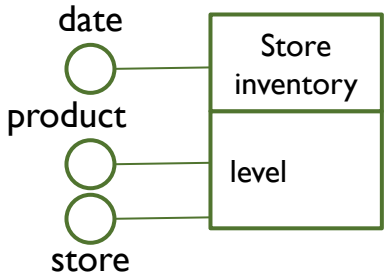
1/1, milk, hk store A, $10
1/1, cream, hk store B, $25
1/1, yogurt, hk store A, $25
1/2, milk, hk store B, $15
1/2, yogurt, hk store C, $30
1/3, milk, hk store D, $20
…
1/31, cream, hk store E, $42

Two Kinds of Fact Schemas

- Transactional fact schemas
  - Each primary event summarizes 1 or more activities in a business process
  - No primary event created if there are no corresponding business transaction
  - Use SUM to summarize the measures

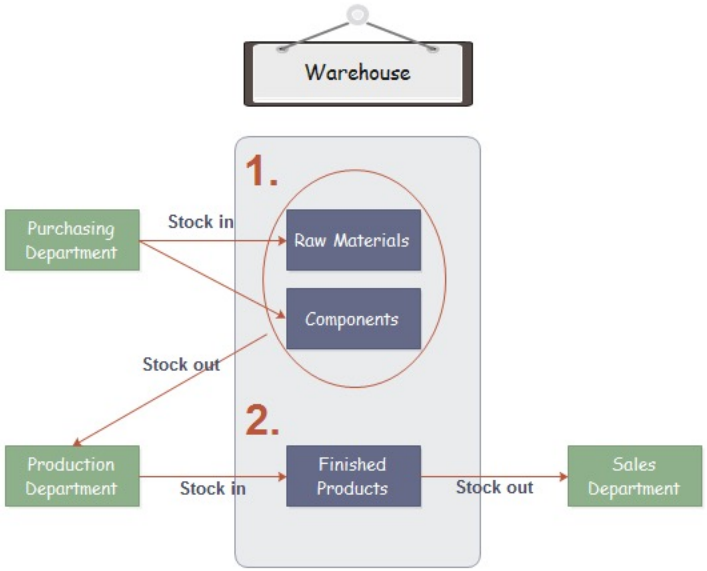


- Snapshot fact schemas
  - Each primary event records the status measurement done periodically
  - Primary events have no direct correspondence with business transaction
  - Cannot use SUM in some situations



Status Measurement

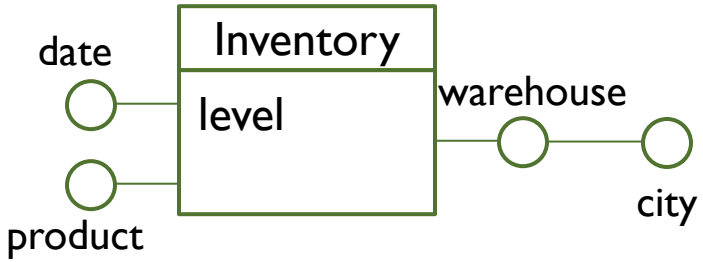
- To analyze of change of some measurement of the environment (e.g. air quality), we need to take periodic snapshot and save the data in a data mart
  - Examples: outdoor temperature, PM10 concentration, water level in a reservoir
- We can also measure the cumulative effect of a series of activities in a business process as status measurement
  - Stock in -> increase stock level
  - Stock out -> decrease stock level

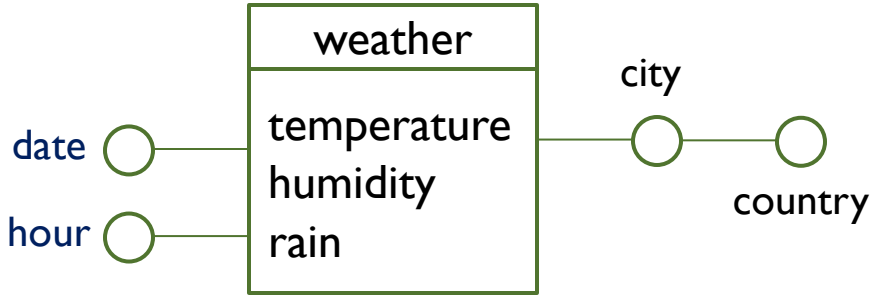


Snapshot Fact Schema

- A snapshot fact samples status measurements at a predetermined interval
  - These measurements *may* be equivalent to the cumulative effect of a series of business activities
  - A primary event refers to an instant in time and the measures are evaluated at that instant.

This snapshot fact records the number of items in stock for each product in each warehouse at the end of each day.

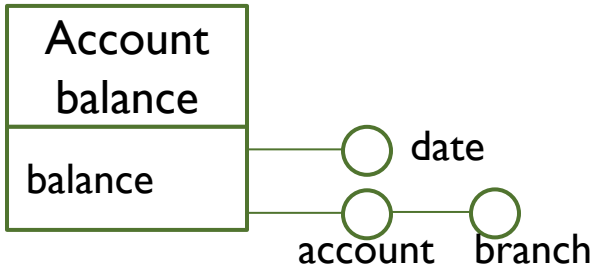




This snapshot fact records the hourly temperature, humidity and precipitation on each day in each city.

Semi-additive Measures

- Usually, periodic status measurement in a snapshot fact are semi-additive
  - are NOT additive along time dimension
    - But you may still use AVG, MIN, MAX
  - may be additive along other dimensions
  - Account A --- Mar 1st $1000, Mar 2nd $2000. $3000



It makes sense to calculate the total account balances of all accounts in a branch on a specific date. But it does not make sense to calculate the sum of balance of today and yesterday. I.e. **balance is additive along 'account' dimension, but not along 'date' dimension.**

Taking Snapshot at ETL process

- The operational system may only store the current status
  - e.g. the current inventory level of a product in a warehouse
  - Older versions of data are replaced by the new versions.
- The ETL process has to take a snapshot of the value in operational system periodically, and use the data population date as the value for the time dimension of the primary event.
  - E.g. Copy the current inventory level of each product to the reconciled database after business hour every day.

Snapshot with Zero Measurement

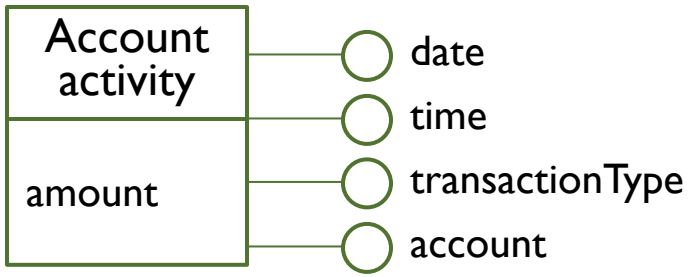
- In addition to the sampling rate, the data mart designer also needs to decide whether to store a primary event for a measurement of 0
  - (ex. 1) If it is desirable to show bank accounts with balance 0 in reports, such primary events should be recorded.
  - (ex. 2) On the other hand, outdated products may result in many primary events with balance of 0 in an inventory fact. It may be preferred to omit such primary events to avoid listing of outdated products in inventory reports.
- You need to specify it clearly in the grain statement
  - E.g. Balance of each bank account at end of each month. Inventory level of *in-stock* products in each store at the end of each day

Summary of Snapshot fact schema

- Primary events correspond to periodic status measurement
- Designer must decide whether to record primary events for zero value of measures
- Often denser than corresponding transactional fact schema
  - E.g. take inventory level every day, even if there is no sale of the product
- The measures are often semi-additive
  - Not additive along time dimension. But can use AVG, MIN, MAX to summarize
  - Maybe additive along other dimension

Calculating Balance from Transactional Facts

- A transaction fact that traces account activities is useful to measure frequency and amount involved in accounts
- But it is inefficient to calculate the account balance as on a specific day at query time



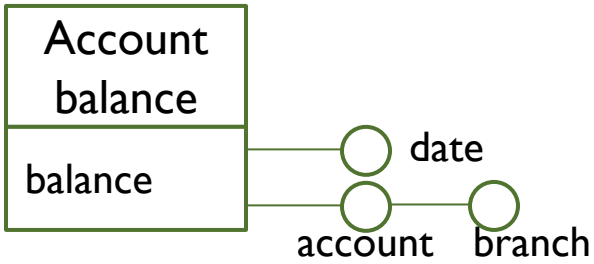
Date	Transaction type	amount
2/1/2012	Initial deposit	2000
3/1/2012	Withdrawal	(200)
3/1/2012	Check	(400)
4/1/2012	Deposit	1000
7/1/2012	Withdrawal	(300)

What are the balance of the account at end of each day?

Date	balance
2/1/2012	2000
3/1/2012	?
4/1/2012	?
…	

Using a Snapshot Fact for Tracing Balance

- A snapshot fact may trace the change of account balance more efficiently
- It is denser than the corresponding transactional fact if the frequency of transaction is small.

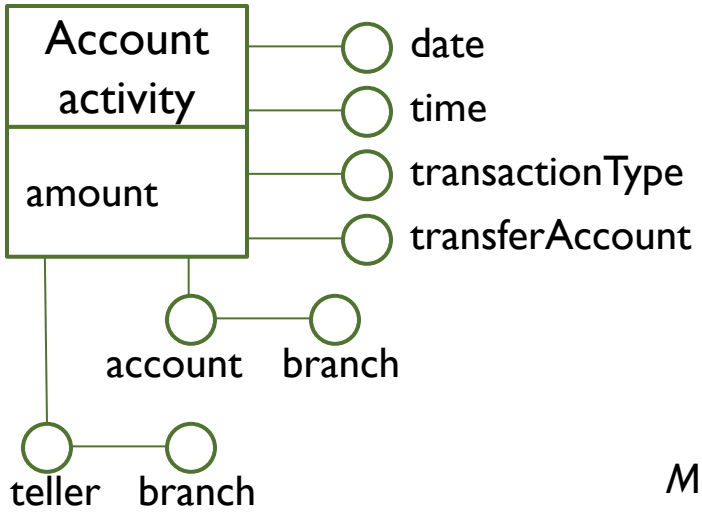


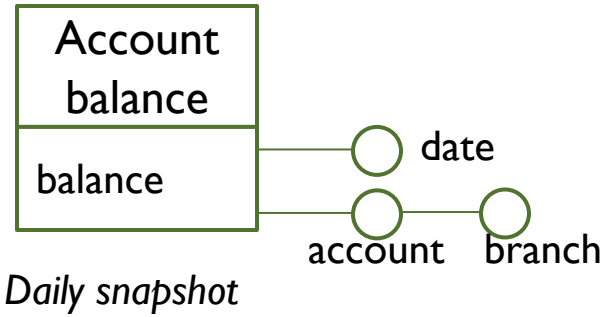
Date	balance
2/1/2012	2000
3/1/2012	1400
4/1/2012	2400
5/1/2012	2400
6/1/2012	2400
7/1/2012	2100

Note: only the events for one account are shown

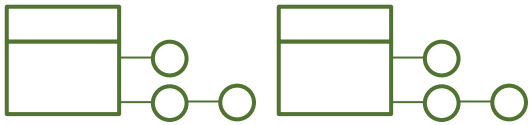
Pairing Transactional and Snapshot Facts

- The snapshot and transactional models reflect two aspects of the same process, and may be used together in a data mart
  - Transactional fact allows detailed analysis of the process activities
  - Snapshot model sacrifices some detail, but allows flexible and powerful analysis of the effect of the transactions



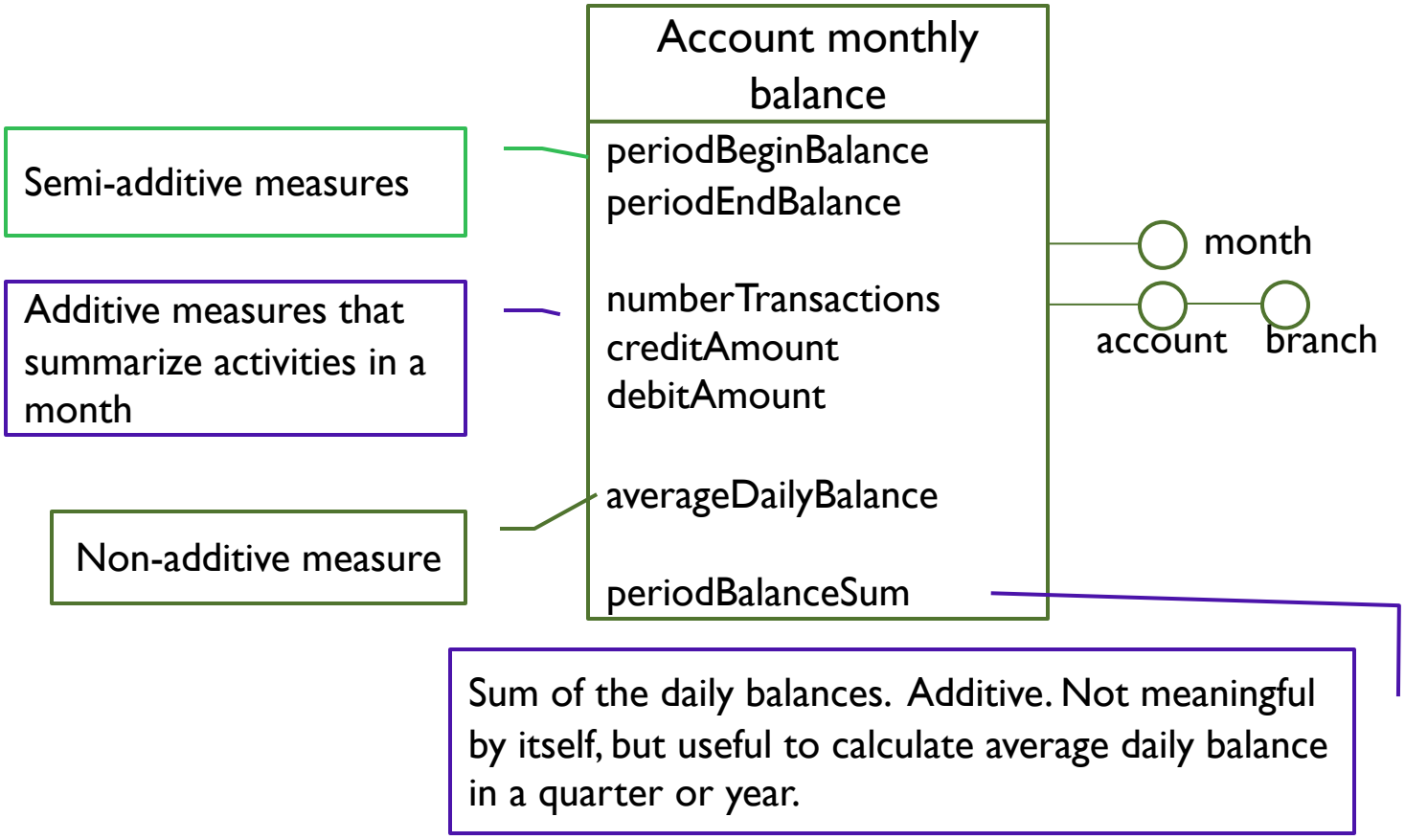


Monthly and quarterly snapshots



Additional Measures in Snapshot Facts

- You can also add additive measures to summarize transactions in a period, and add additive components to calculate averages.



Exercise

- Perform conceptual design for the case on slide no. 29.
  - What are the facts and dimensions?
  - Which facts are transactional? Which are snapshot?
  - Which measures are stored? Which are calculated?
  - Examine the additivity of the measures along each dimension. (pay special attention to 'inventory')