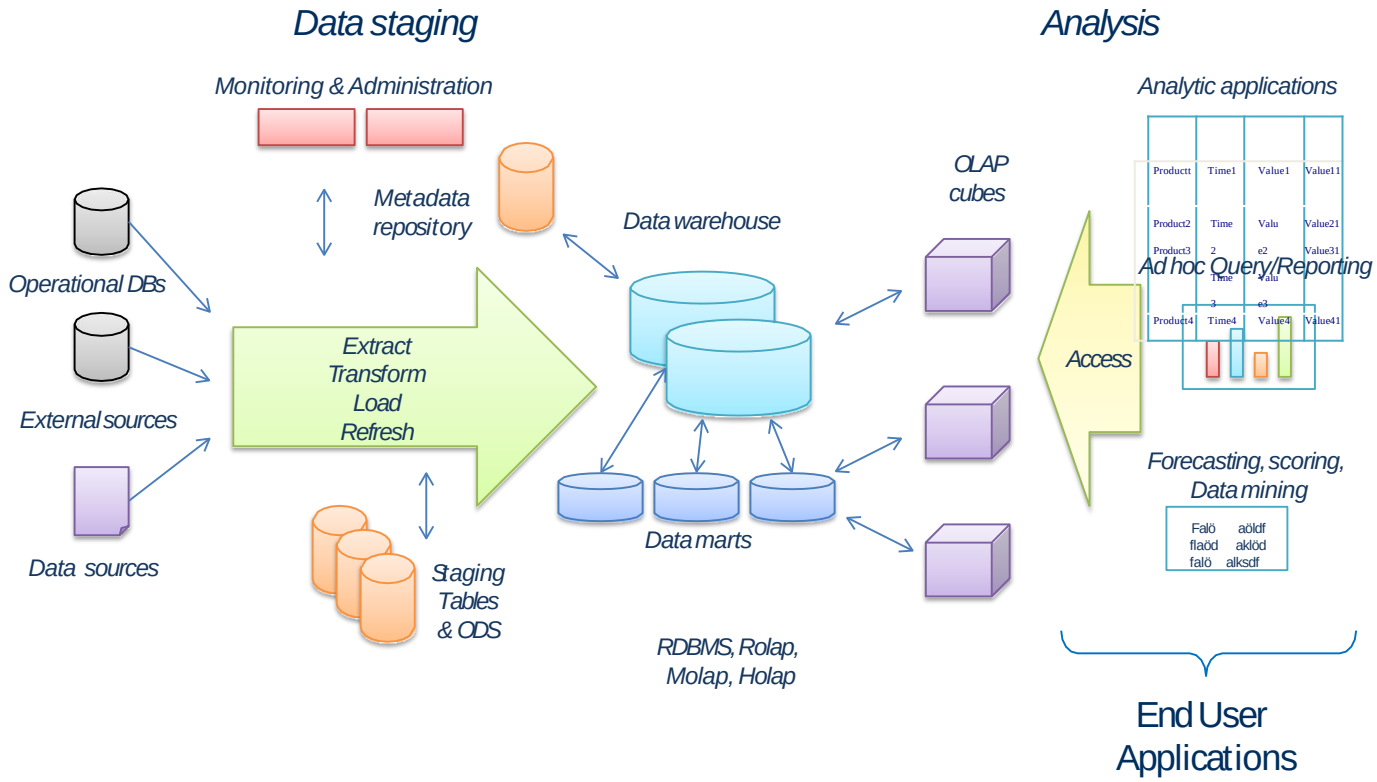
Presentation:
Dimensional Modelling 2

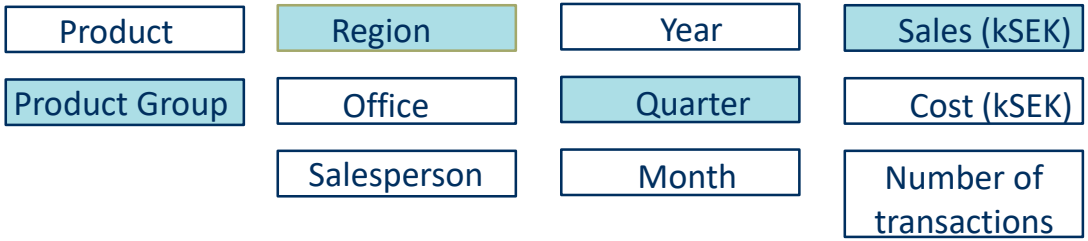
Erik Perjons

DSV, Stockholm University

DW Architecture

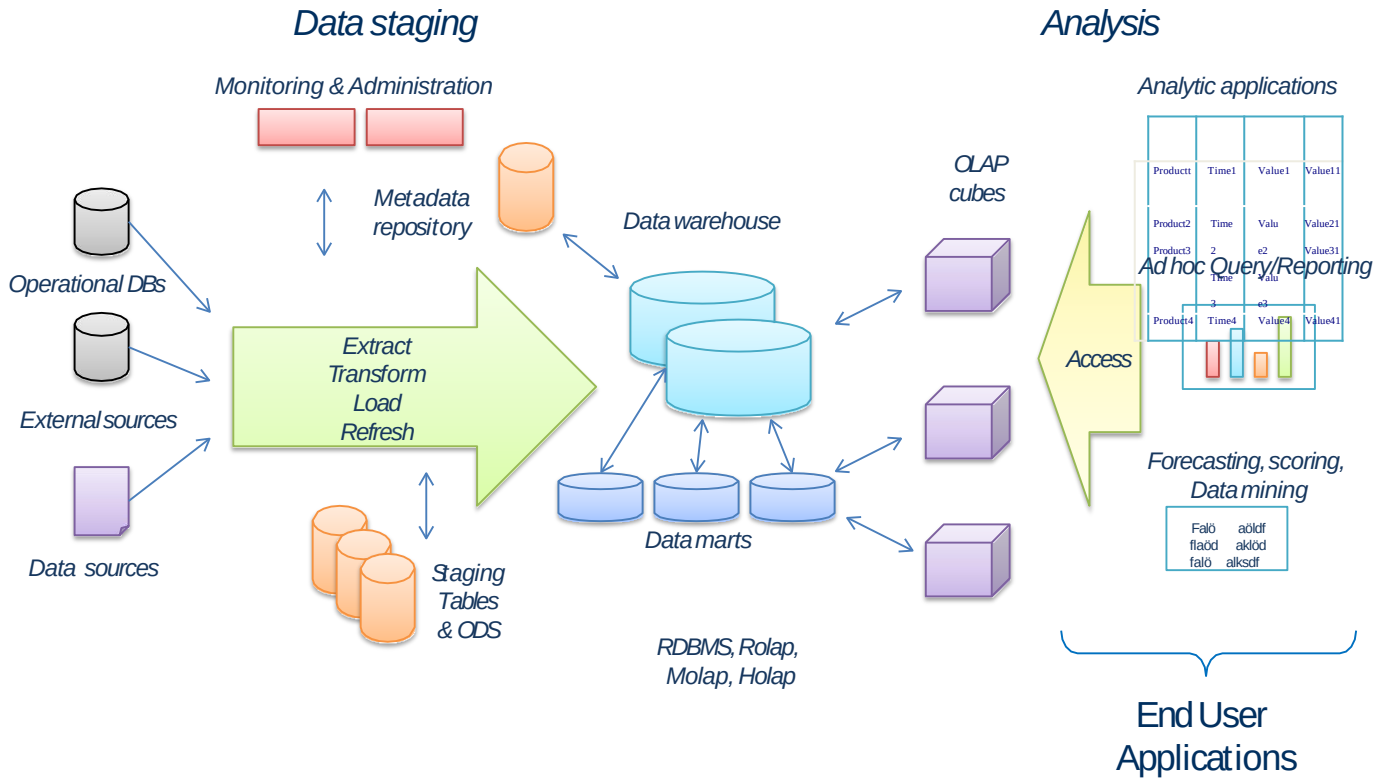


OLAP tools: User interface

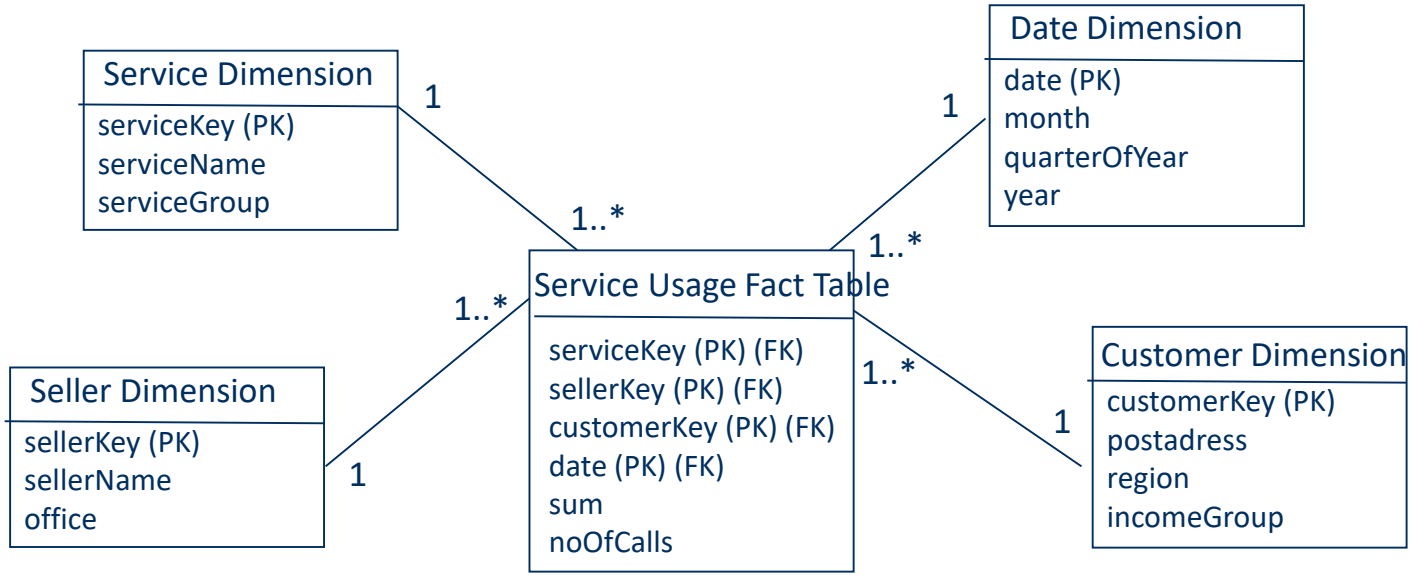


Product Group	Region	Quarter	Sales (kSEK)
Group A	Region ABC	Quarter 1 - 2018	100
Group A	Region DEF	Quarter 1 - 2018	400
Group B	Region ABC	Quarter 1 - 2018	200
Group B	Region DEF	Quarter 1 - 2018	900

DW Architecture



Dimensional modelling/Star schema



Stockholms universitet

Service Dimension

Key	Service	Service group
S1	Local call	Group A
S2	Intern. call	Group A
S3	SMS	Group B
S4	WAP	Group C

Date Dimension

Date/ Key	Month	Quarter	Year
991011	9910	4 - 99	99
991012	9910	4 - 99	99

Service Usage Fact Table

				Sum	Number of calls
C210	S1	F11	991011	25:00	3
C210	S3	F11	991011	05:00	1
C212	S2	F13	991011	89:00	1
C213	S1	F13	991011	12:00	1
C214	S4	F13	991012	08:00	1

1 1..* 1 1..*

Seller Dimension

Key	Seller	Office
F11	Anders C	Sundsvall
F12	Lisa B	Sundsvall
F13	Janis B	Kista

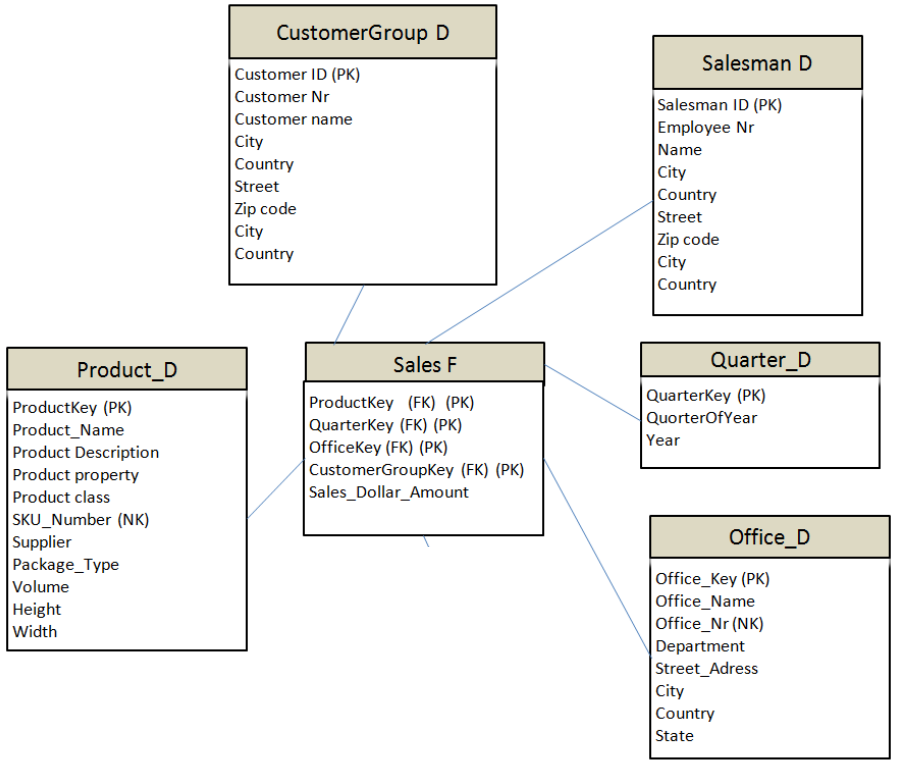
1 1..* 1 1..*

Customer Dimension

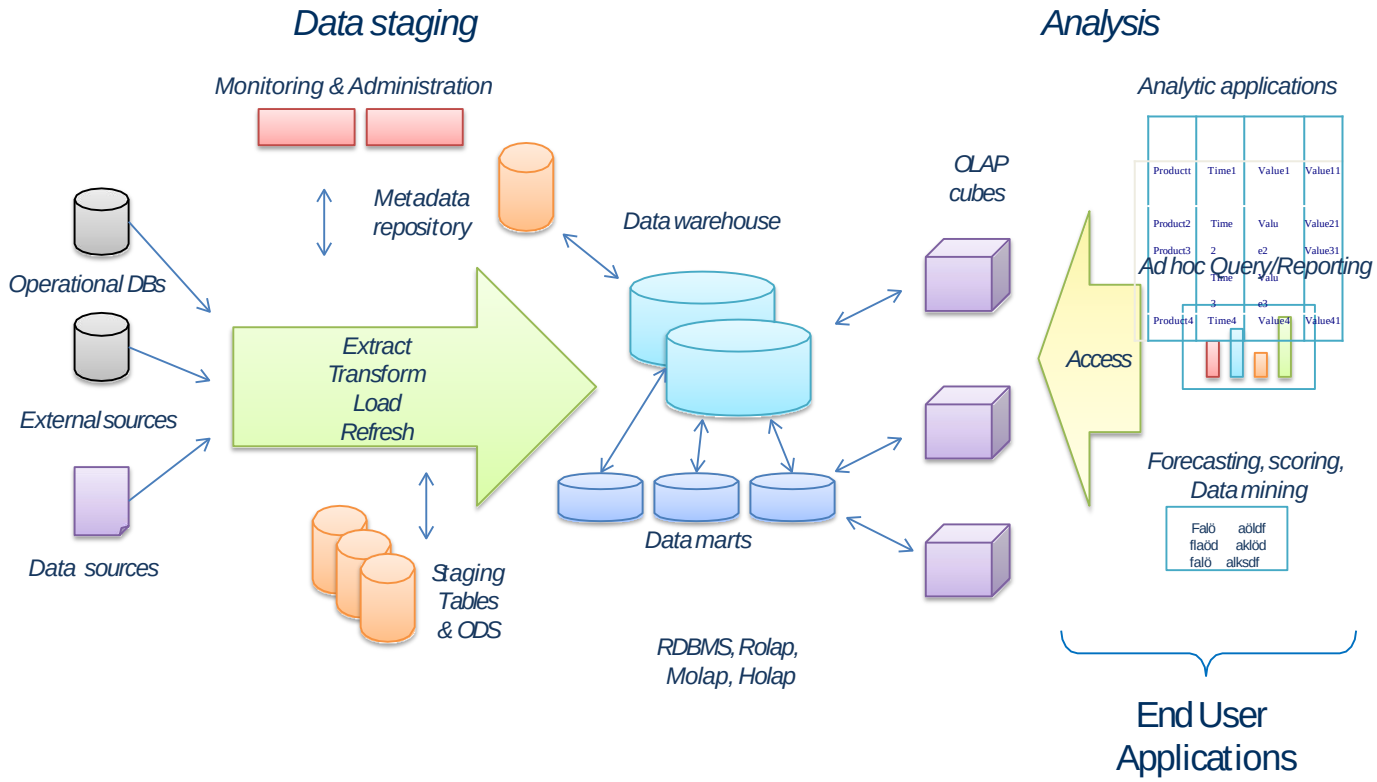
Key	Customer	Address	Region	Income group
C210	Anna N	Stockholm	Stockholm	B
C211	Lars S	Malmö	Skåne	B
C212	Erik P	Rättvik	Dalarna	C
C213	Danny B	Stockholm	Stockholm	A
C214	Åsa S	Stockholm	Stockholm	A

Towards Dimensional Modelling

- A single fact table
- A number of dimension tables
- Denormalized dimensions
- Hierarchies
- Keys
- Grain
- Use understandble labels for attributes/columns and data
- Managing queries



DW Architecture

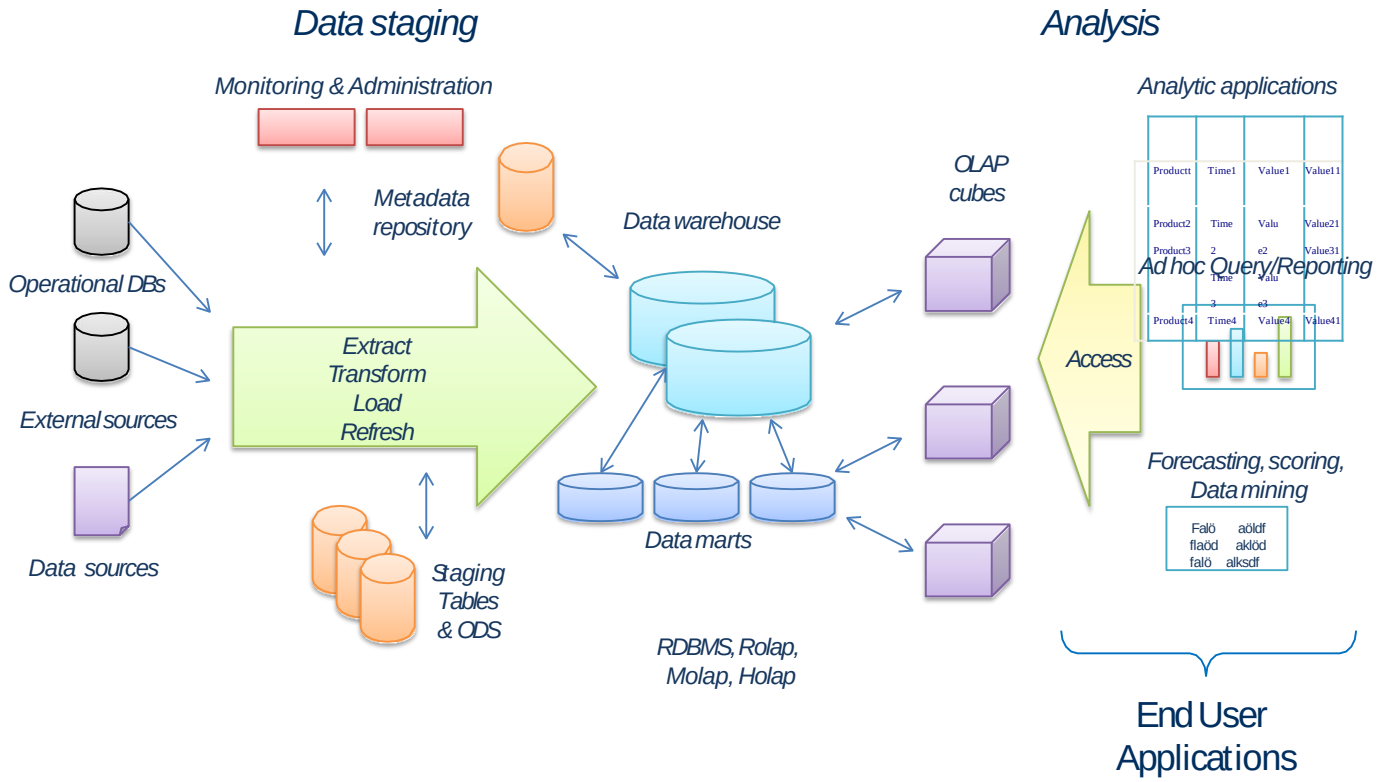


A business event/transaction

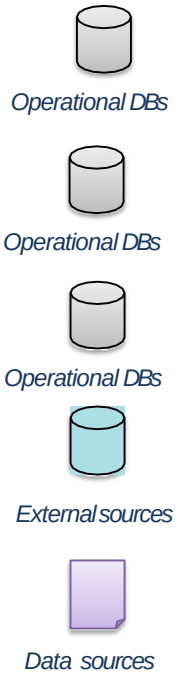
Translates into

Retail Sales Facts
Date Key (FK)
Product Key (FK)
Store Key (FK)
Promotion Key (FK)
Customer Key (FK)
Clerk Key (FK)
Transaction #
Sales Dollars
Sales Units

DW Architecture



Operational Systems

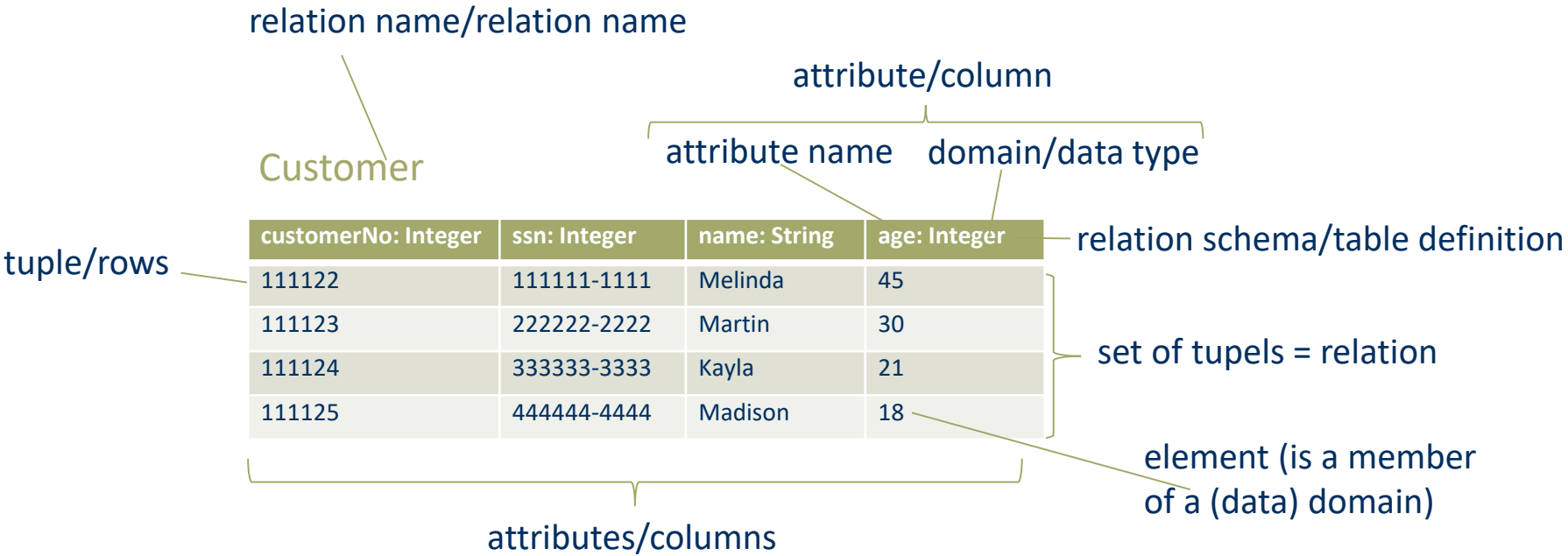


- The operational systems/transactional systems are tuned to support known daily operations
- They are normalized according to 3 NF
- They are focusing is on data integrity

Database fundamentals

- Relation and Relation schema
- Relational database schema
- Keys
- Normalisation
- Data integrity

Relation and relation schema



customerNo: Integer	ssn: Integer	name: String	age: Integer
111122	111111-1111	Melinda	45
111123	222222-2222	Martin	30
111124	333333-3333	Kayla	21
111125	444444-4444	Madison	18

Attribute value = attribute name paired with an element that is part of the attribute's domain)

Relational database schema

Stockholms universitet

Student
studentID (PK) studentName

Registration
registrationID (1..1) (PK) regDate (1..1) courseOccasionID (1..1) (FK to CourseOccasion.courseOccasionID) studentID (1..1) (FK to Student.studentID)

Course
courseID (PK) courseName

CourseOccasion
courseOccasionID (PK) startDate courseID (FK to Course.courseID),

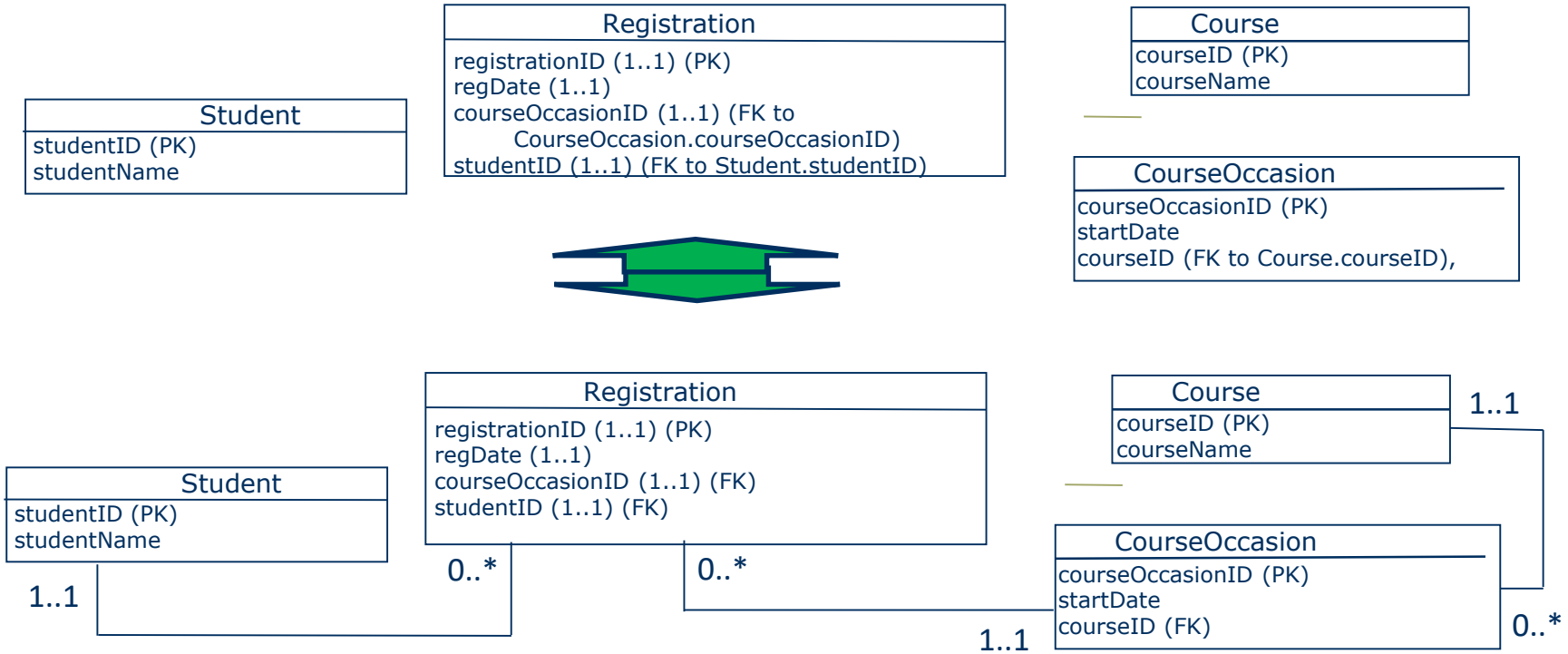
Student	
studentID (PK)	studentName

Course	
courseID (PK)	courseName

CourseOccasion		
courseOccasionID (PK)	startDate	

Registration			
registrationID (PK)	regDate	studentID	courseOccasionID

Relational database schema



Keys

- Candidate key - is a minimal set of attributes/columns necessary to uniquely identify a tuple/row
- Primary key - is a candidate key chosen to be the unique identifyer of the tuple/row
- Alternate key - is a candidate key not chosen to be the primary key
- Foreign key – is a key in a table that refers to a primary key in another table, representing the relation between two tables in a relational database schema

Keys

- Natural key – is a candidate key derived from application data (that is, formed of attribute(s) that already exist in the real world)
- Surrogate key – is a candidate key not derived from application data (that is, has no meaning outside the database environment) – assigned sequentially by the system (that is, an integer that starts by 1 and is increased by 1 when adding rows to the dimension table)
- Composite key/Concatenated key - is a candidate key that consists of two or more attributes

Keys

- Intelligent key/smart keys – is a key that includes encoded info into the key.
- For example Stock keeping unit (SKU) can contain information about manufacturer, product type and product

Normalization

- Normalization - is the process to restructuring a relational database in accordance with a series of so called normal forms in order to reduce data redundancy (that is, an activity aiming to store data in one place), prevent update, insertion, deletion anomalies, and, thereby, improve data integrity

1:st Normal Form (1NF)

- 1 NF is a property of a relation where each attribute in a tuple contains only a single value from a domain

1:st Normal Form (1NF)

Team

Name	Members
Farsta och Cobran (FOC)	Helen Andrea Lotta Maria
Farsta och Cobran (FOC)	Lars Sören Peter David
Kyrkheddinge IF (KIF)	Helen Andrea Lotta Maria
Farsta och Cobran (FOC)	Lars Sören Peter David

Member

Member	Team
Helen	FOC
Andrea	FOC
Lotta	FOC
Maria	FOC
Lars	FOC
Sören	FOC
Peter	FOC
David	FOC
Helen	KIF
Andrea	KIF
Lotta	KIF

2:nd Normal Form (2NF)

- 2 NF is a property of a relation that is in 1 NF and where ALL the attributes in a composite candidate key determines the values of the non-key attributes

2:nd Normal Form (2NF)

Stockholms universitet

Purchase

PurchaseID	ItemID	ItemName	Type	amount	Unitsof_ Measure	Price
673	46545	Cardamom	Pods	3	Kg	10 £
673	73	Glee	Frozen	1	L	5 £
673	41065	Cumin	Seeds	1	Kg	10 £
674	42775	Cinnamon	Stick	0,5	Kg	12 £
674	5534	Ginger/Garlic	Paste	10	Kg	5 £
675	311	Cashew	Nuts	5	Kg	20 £
676	41065	Cumin	Seeds	1	Kg	10 £
676	48888	Coriander	Seeds	0,5	Kg	10 £
676	46545	Cardamom	Pods	1	Kg	10 £
677	73	Glee	Package	10	L	5 £
677	151	Yoghurt	Package	50	L	1 £

Purchase

PurchaseID	amount	ItemID
673	3	46545
673	1	73
673	1	41065
674	0,5	42775
674	10	5534
675	5	311
676	1	41065
676	0,5	48888
676	1	46545
677	10	73
677	50	151

Item

ItemID	ItemName	Type	Unitsof_ Measure	Price
46545	Cardamom	Pods	Kg	10 £
73	Glee	Frozen	L	5 £
41065	Cumin	Seeds	Kg	10 £
42775	Cinnamon	Stick	Kg	12 £
5534	Ginger/Garlic	Paste	Kg	5 £
311	Cashew	Nuts	Kg	20 £
41065	Cumin	Seeds	Kg	10 £
48888	Coriander	Seeds	Kg	10 £
46545	Cardamom	Pods	Kg	10 £
73	Glee	Package	L	5 £
151	Yoghurt	Package	L	1 £

3:rd Normal Form (3NF)

- 3 NF is a property of a relation that is in 2 NF and all non-key attributes are determined only by the candidate keys (and not by any non-key attributes)

3:rd Normal Form (3NF)

Owner

Person ID	Name	Car ID	Brand	Model	Color
51-03-14	Helen	NBS-735	SAAB	Sonett	Red
88-09-16	Andrea	HSJ-888	Volvo	245	Red
73-05-05	Lotta	KDD-284	SAAB	9-3	Silver
92-04-03	Maria	HJI-223	Volvo	PV	Black
73-12-19	Tuija	KDF-638	Volvo	XC 90	Black
58-11-01	Ulrika	HYT-923	SAAB	900	White

Person

Person ID	Name
51-03-14	Helen
88-09-16	Andrea
73-05-05	Lotta
92-04-03	Maria
73-12-19	Tuija
58-11-01	Ulrika

Owner

Person ID	Car ID
51-03-14	NBS-735
88-09-16	HSJ-888
73-05-05	KDD-284
92-04-03	HJI-223
73-12-19	KDF-638
58-11-01	HYT-923

Car

Car ID	Color	Brand	Model
NBS-735	Red	SAAB	Sonett
HSJ-888	Red	Volvo	245
KDD-284	Silver	SAAB	9-3
HJI-223	Black	Volvo	PV
KDF-638	Black	Volvo	XC 90
HYT-923	White	SAAB	900

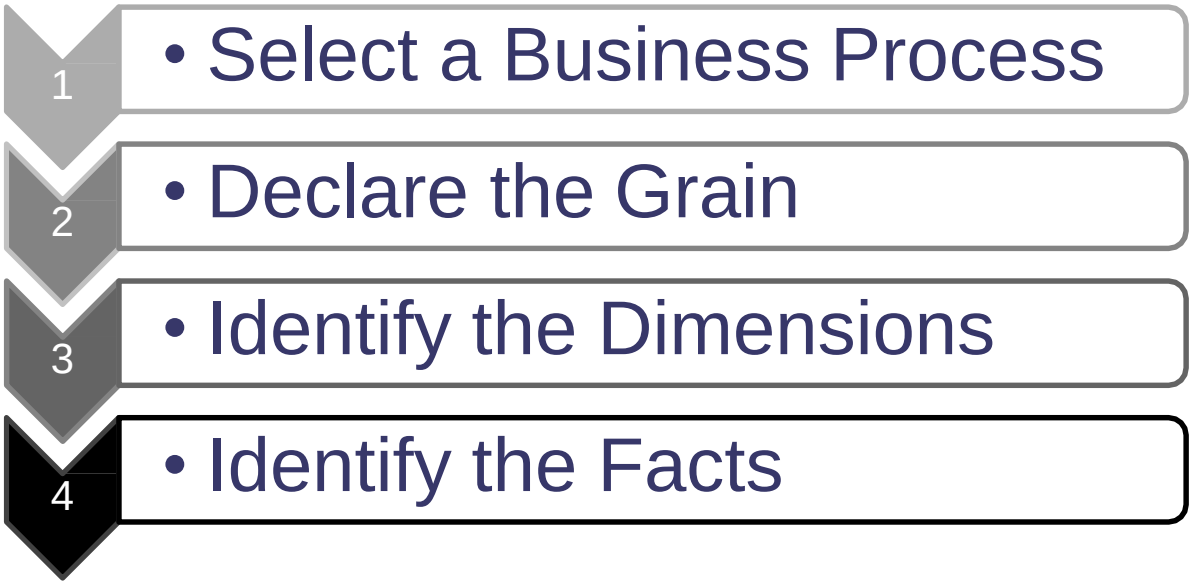
Data Integrity

- Data Integrity – is the maintenance of the accuracy and consistency of data. Data integrity is enforced by a number of rules/constraints:
  - Entity Integrity - every tuple must have a primary key and the value of the primary key must be unique
  - Referential Integrity – the foreign key value must refer to a primary key value of some table in the database
  - ...

Anomalies

- Update anomalies – if redundant data is stored in several places there is a risk that an update is not done in all this places
- Delete anomalies – data might be lost when some other data is delted if not stored properly
- Insertion anomalies – it is not possible to add new data

Four steps to design a star schema



The retail case

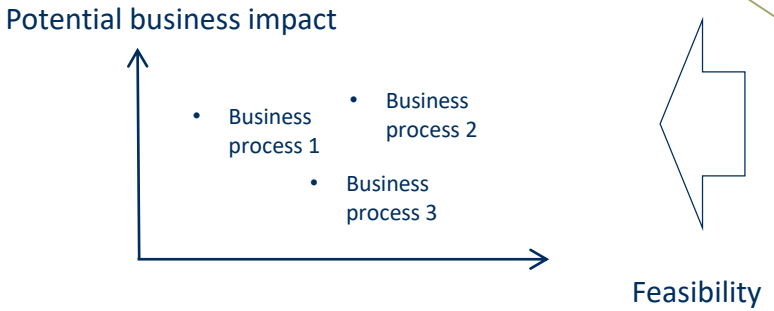
- The retail company has 100 grocery stores spread over a five state area
- Each store has a number of departments, including grocery, frozen food, dairy, meat, produce, bakery, floral, health/beauty aids
- Each store has roughly 60 000 individual products
- The individual products are called Stock Keeping Units (SKU)

The retail case

- Management is concerned with:
  - the logistics of ordering (the procurement are often done too late)
  - the stock levels (too high stock levels for some products and too low for other)
  - the selling of products (too low sales figure)
- We can collect data:
  - at the back door of retail stores, where deliveries are made
  - at the depository where we keep our stock
  - at the cash registers, called point of sale (PoS), where customer purchases products.

Step 1. Select a business process

- A business process is a “low-level activity” performed by an organization, such as taking orders, receiving payments, etc
- Kimball: ”The first dimensional model built should be the one with the most impact – it should answer the most pressing business questions and be readily accessible for data extraction.”



A dimensional model is based on a business process, therefore the focus on business process

Step 1. Select a business process

- Which Business process should we select first?
  - the *logistics of ordering* can be interpreted as *the order process*
  - the *stock levels* can be interpreted as *the inventory process*
  - the *selling of products* can be interpreted as *the sales process*

Step 1. Select a business process

- Which Business process should we select first?
  - the *logistics of ordering* can be interpreted as *the order process*
  - the *stock levels* can be interpreted as *the inventory process*
  - **the *selling of products* can be interpreted as *the sales process***

More information is needed to select the business process with most impact and highest feasibility. In this case, we assume that a sales process in an important process since the management is concerned with the low sales figures, and increased sales will probably make a larger impact than the other two processes

Step 2. Declare the grain

Stockholms universitet

- The grains is the level of detail of data of the process or event. The grain means specifying exactly what an individual fact table row represent.
- Typical grains are individual transactions, individual daily (or monthly) snapshots
- Kimball: ”The less granular model is immediately vulnerable to unexpected user requests to drill down into the detail.”
- Kimball: “Preferably you should develop dimensional models for the most atomic information captured by a business process.”

Step 2. Declare the grain

This is how you can express the most detailed level of data:

- Most granular data (the more you go into details, the more granular data you have)
- High granularity (the more you go into details, the higher granularity you have)

Step 2. Declare the grain

- What grain shall we choose for our case study?
  - An individual line item on a POS transaction, i.e. the sale of a product, per store, per transaction)
  - The sale of a product, per store, per day
  - The sale of a product, per region, per day
  - The sale of a product, per store, per month
  - The sale of a product category, per region, per month

Step 2. Declare the grain

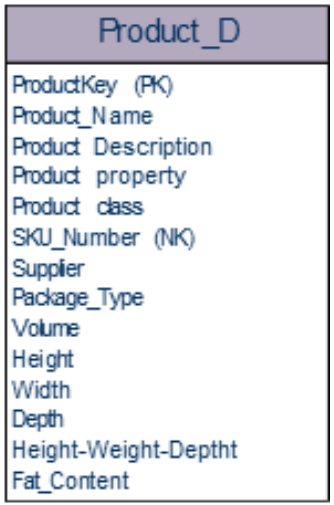
- What grain shall we choose for our case study?
  - **An individual line item on a POS transaction, i.e. the sale of a product, per store, per transaction**
  - The sale of a product, per store, per day
  - The sale of a product, per region, per day
  - The sale of a product, per store, per month
  - The sale of a product category, per region, per month

Step 3. Identify the dimensions

- Identify the dimensions that will be applied to each fact table tuple/record
  - Consider what aspects do you want to be able to slice and dice
  - Typical dimensions are date, product, customer, store, etc
- Guideline: Use the declared grain of the selected process to determine dimensions: product, store
- Guideline: Use the Who, What, When, Where, and How to identify possible dimensions

Step 3. Identify the dimensions

- An example of a dimension: Product dimension

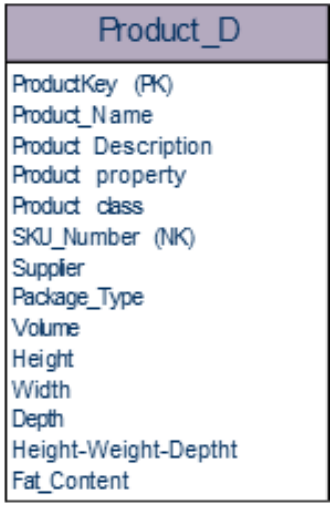


Product

ProductKey	SKU	Product Name	Product Description	Supplier	Product Class	Product Property	Package Type	Volume	Height	Widht	Depth	Height-Widtht-Depthht	Fat content
1	7 310865 001191	Blue Milk	Milk with vitamin D and 0,5% fat	Arla	Dairy	Low fat	Roof ridge pack	1 liter	23 cm	7 cm	7 cm	23x7x7	0,5 %
2	7 310865 001795	Green Milk	Milk with vitamin D and 1,5% fat	Arla	Dairy	Plain	Brick pack	1 liter	16 cm	6,5 cm	9,5 cm	16x6,5x 9,5	1,5 %
3	7 310865 000156	Red Milk	Milk with vitamin D and 3% fat	Arla	Dairy	Plain	Brick pack	1 liter	16 cm	6,5 cm	9,5 cm	16x6,5x 9,5	3 %
4	7 300170 062642	Green Milk	Local Milk with vitamin D and 1,5% fat	Coop Änglam ark	Dairy	Ecological	Roof ridge pack	1 liter	23 cm	7 cm	7 cm	23x7x7	1,5 %

Dimensions

- Use surrogate key as primary key - to keep history in if natural keys are changing in the organization
- How to identify attributes:
  - "we do not measure them, we usually know them"
  - usually text fields, with discrete values, e.g., the flavor of a product, the size of a product
- Denormalization is no problem since the ETL process ensure the data quality, and no advanced calculations (only filtering) are made on the dimensions



Dimensions

- Use column names/labels that are easy to understand
- Use names/lables of the values that are easy to understand

The end users will see these names/labels interacting with the system using an OLAP tool

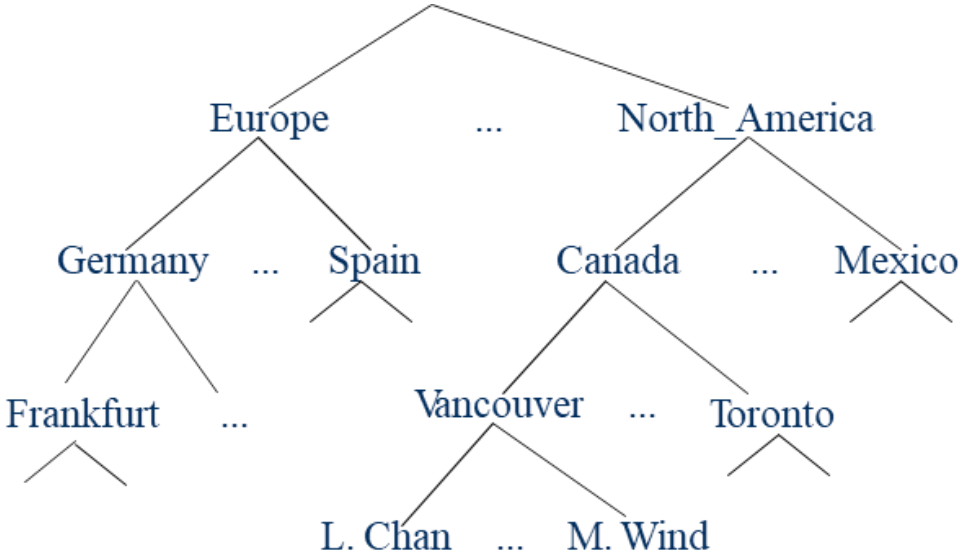
Product

ProductKey	SKU	Product Name	Product Description	Supplier	Product Class	Product Property	Package Type	Volume	Height	Widht	Depth	Height-Widtht-Depht	Fat content
1	7 310865 001191	Blue Milk	Milk with vitamin D and 0,5% fat	Arla	Dairy	Low fat	Roof ridge pack	1 liter	23 cm	7 cm	7 cm	23x7x7	0,5 %
2	7 310865 001795	Green Milk	Milk with vitamin D and 1,5% fat	Arla	Dairy	Plain	Brick pack	1 liter	16 cm	6,5 cm	9,5 cm	16x6,5x 9,5	1,5 %
3	7 310865 000156	Red Milk	Milk with vitamin D and 3% fat	Arla	Dairy	Plain	Brick pack	1 liter	16 cm	6,5 cm	9,5 cm	16x6,5x 9,5	3 %
4	7 300170 062642	Green Milk	Local Milk with vitamin D and 1,5% fat	Coop Änglam ark	Dairy	Ecological	Roof ridge pack	1 liter	23 cm	7 cm	7 cm	23x7x7	1,5 %

Step 3. Identify the dimensions

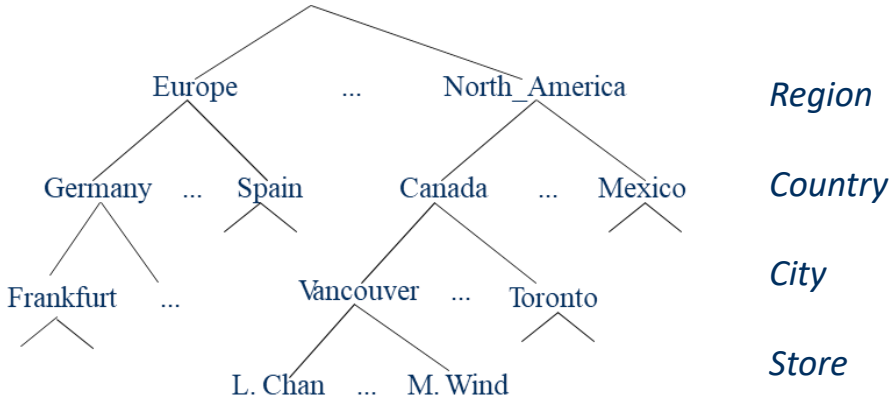
- Another dimension: Store dimension
- A geographic dimension - every store has a location
- Dimensions have hierarchies - we can roll up and drill down on geographic attributes

Dimensional hierarchies



Dimensional hierarchies

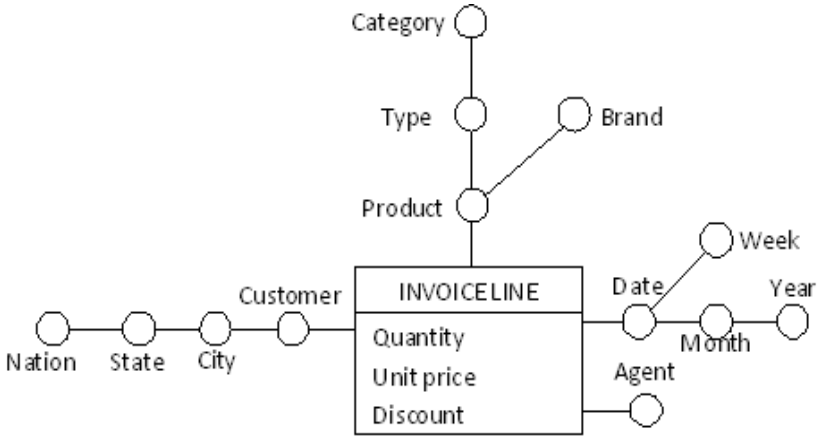
- A hierarchy is a directed tree which nodes are dimensional attribute values/elements
- There is a many-to-one relationship between the nodes in the hierarchy



Store_D
Store_Key (PK) Store_Name Store_Nr (NK) Street_Adress City Country State Zip_Code Manager

Dimensional Fact Model (DFM)

- Dimensional Fact Model (DFM) is graphical formalism showing:
  - Facts in the fact table
  - Hierarchies of the dimensional tables
- Developed by Golfarelli & Rizzo



Step 3. Identify the dimensions

- Date dimension – most star schema has a date dimension

Date Key	Date	Full Date Description	Day of Week	Calendar Month	Calendar Quarter	Calendar Year	Fiscal Year Month	Holiday Indicator	Weekday Indicator
20130101	01/01/2013	January 01, 2013	Tuesday	January	Q1	2013	F2013-01	Holiday	Weekday
20130102	01/02/2013	January 02, 2013	Wednesday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130103	01/03/2013	January 03, 2013	Thursday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130104	01/04/2013	January 04, 2013	Friday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130105	01/05/2013	January 05, 2013	Saturday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130106	01/06/2013	January 06, 2013	Sunday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130107	01/07/2013	January 07, 2013	Monday	January	Q1	2013	F2013-01	Non-Holiday	Weekday
20130108	01/08/2013	January 08, 2013	Tuesday	January	Q1	2013	F2013-01	Non-Holiday	Weekday

Date Dimension

Date Key (PK)
Date
Full Date Description
Day of Week
Day Number in Calendar Month
Day Number in Calendar Year
Day Number in Fiscal Month
Day Number in Fiscal Year
Last Day in Month Indicator
Calendar Week Ending Date
Calendar Week Number in Year
Calendar Month Name
Calendar Month Number in Year
Calendar Year-Month (YYYY-MM)
Calendar Quarter
Calendar Year-Quarter
Calendar Year
Fiscal Week
Fiscal Week Number in Year
Fiscal Month
Fiscal Month Number in Year
Fiscal Year-Month
Fiscal Quarter
Fiscal Year-Quarter
Fiscal Half Year
Fiscal Year
Holiday Indicator
Weekday Indicator
SQL Date Stamp
...

Step 3. Identify the dimensions

- The data can be populated in advance.
- It has one meaningful key (intelligent/smart key).

  That is, it does not have any surrogate key due to performance, ease of use, partitioning reasons

Date Dimension
Date Key (PK)
Date
Full Date Description
Day of Week
Day Number in Calendar Month
Day Number in Calendar Year
Day Number in Fiscal Month
Day Number in Fiscal Year
Last Day in Month Indicator
Calendar Week Ending Date
Calendar Week Number in Year
Calendar Month Name
Calendar Month Number in Year
Calendar Year-Month (YYYY-MM)
Calendar Quarter
Calendar Year-Quarter
Calendar Year
Fiscal Week
Fiscal Week Number in Year
Fiscal Month
Fiscal Month Number in Year
Fiscal Year-Month
Fiscal Quarter
Fiscal Year-Quarter
Fiscal Half Year
Fiscal Year
Holiday Indicator
Weekday Indicator
SQL Date Stamp
...

Step 3. Identify the dimensions

- A **fiscal year** (or **financial year**, or sometimes **budget year**) is the period used by governments for accounting and budget purposes, which varies between countries. It is also used for financial reporting by business and other organizations. Laws in many jurisdiction require company financial reports to be prepared and published on an annual basis, but generally do *not* require the reporting period to align with the calender year (1 January to 31 December). (Wikipedia)

Date Dimension
Date Key (PK)
Date
Full Date Description
Day of Week
Day Number in Calendar Month
Day Number in Calendar Year
Day Number in Fiscal Month
Day Number in Fiscal Year
Last Day in Month Indicator
Calendar Week Ending Date
Calendar Week Number in Year
Calendar Month Name
Calendar Month Number in Year
Calendar Year-Month (YYYY-MM)
Calendar Quarter
Calendar Year-Quarter
Calendar Year
Fiscal Week
Fiscal Week Number in Year
Fiscal Month
Fiscal Month Number in Year
Fiscal Year-Month
Fiscal Quarter
Fiscal Year-Quarter
Fiscal Half Year
Fiscal Year
Holiday Indicator
Weekday Indicator
SQL Date Stamp
...

Step 3. Identify the dimensions

- Time dimension – may exists in a dimensional model
- Time should not be part of the date dimension
- The time can be populated in advance – grain can be seconds, minutes, or hours
- If you do not need to roll-up on time, from second to minutes, from minutes to hours – the time can be stored in the fact table

Step 4. Identify the facts

- Establishing what we want to measure, that is, the process metrics or key performance indicators (KPI)
- How to identify the facts?
  - “something not known in advance” (Note, the values of the dimensions we know in advance)
  - an observation
  - many facts (but not all) have numerical values

Step 4. Identify the facts

- What facts shall we choose for our case study?
- What can we find out of a PoS?
  - The number of products?
  - The total width of the products?
  - The total sum of the products

Step 4. Identify the facts

- What facts shall we choose for our case study?
- What can we find out of a PoS?
  - **The number of products?** → Sales Quantity
  - The total width of the products?
  - **The total sum of the products** → Sales Dollar Amount

Point of Sales F			
ProductKey	:	Integer	(PK, FK)
DateKey	:	Integer	(PK, FK)
StoreKey	:	Integer	(PK, FK)
Sales Quantity	:	Integer	(fact)
Sales Dollar Amount	:	Integer	(fact)

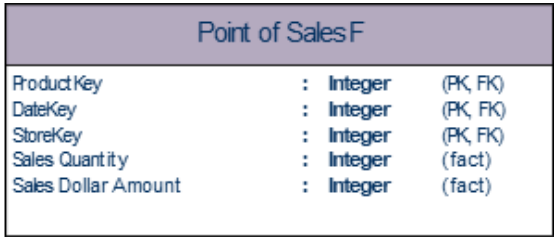
Two kind of facts

- There are two kinds of fact table attributes:
  - **Classification attributes:** Keys that captures the dimensions of the process
    - Keys are usually represented in integer form and do not require much memory space
  - **Facts:** Attributes that captures the measures of the process
    - Facts are often numeric properties and can usually be represented as integers
    - Contrast to dimensional attributes which are usually long text strings

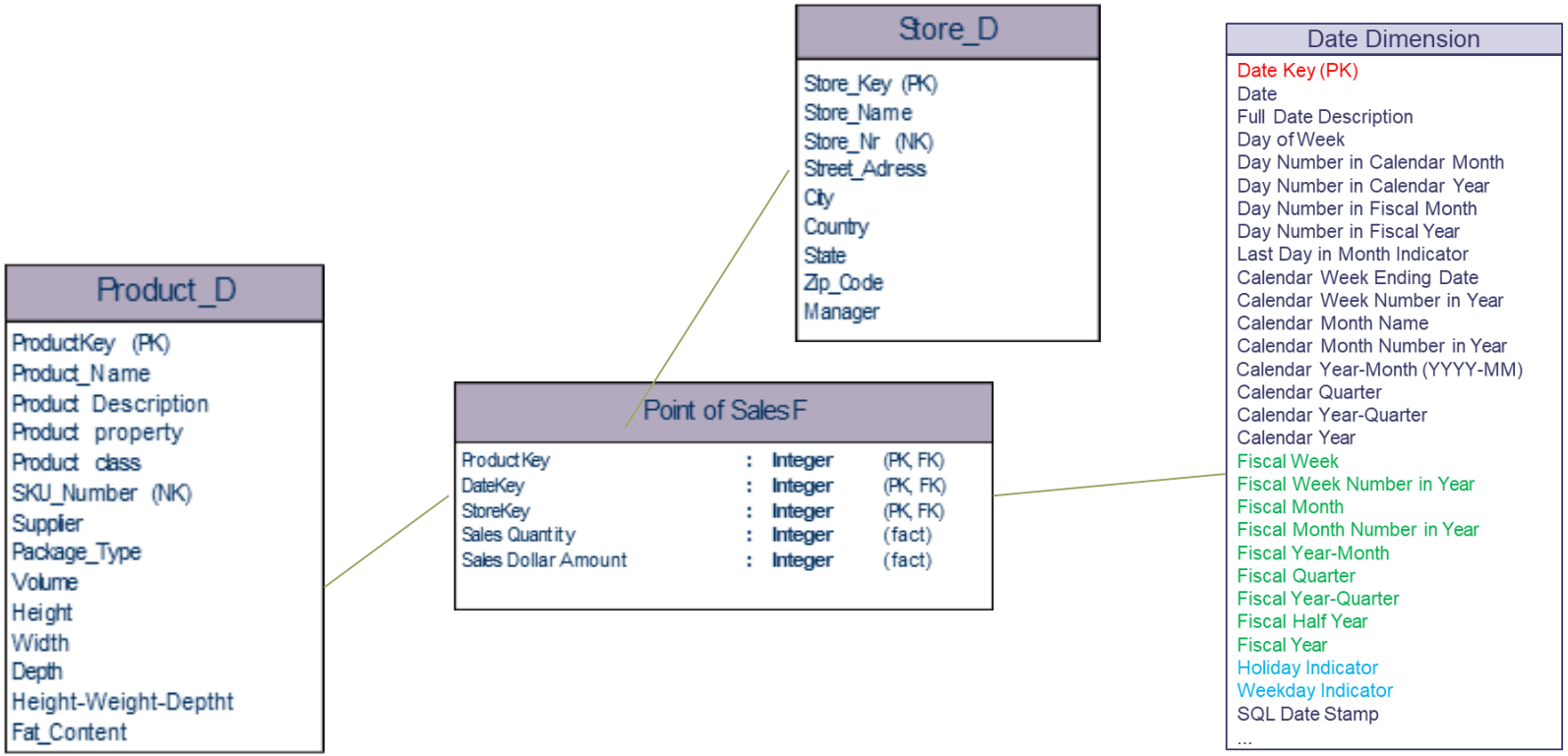
Point of Sales F			
Product Key	:	**Integer**	(PK, FK)
DateKey	:	**Integer**	(PK, FK)
StoreKey	:	**Integer**	(PK, FK)
Sales Quantity	:	**Integer**	(fact)
Sales Dollar Amount	:	**Integer**	(fact)

Sparse fact tables important

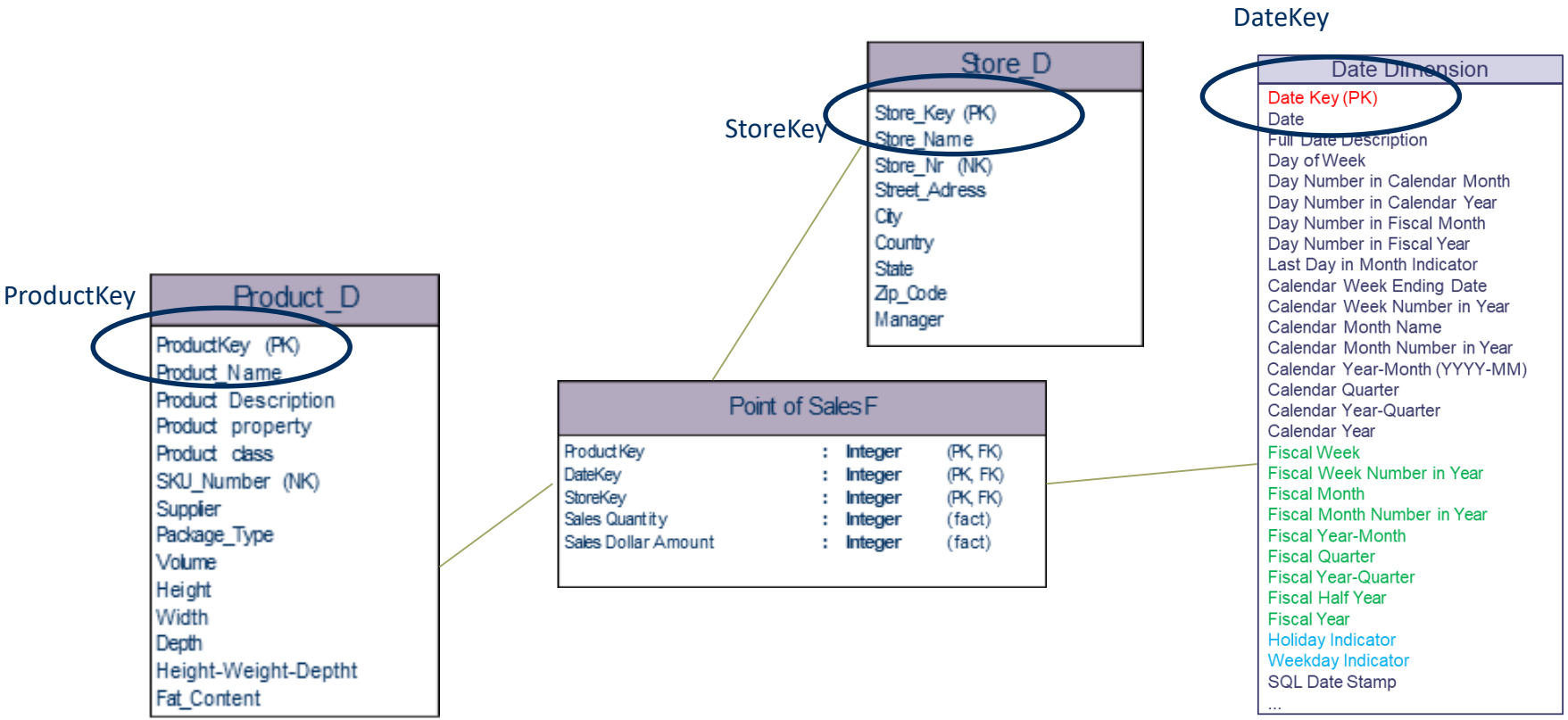
- It is important to have a sparse fact table since it is critical to the memory space consumption of the data warehouse calculations
- The Primary key for the fact table is a concatenated key consisting of the foreign keys to (several of) its dimensions
- Use surrogate keys in the dimension to avoid to end up with text/strings as foreign keys in the fact table



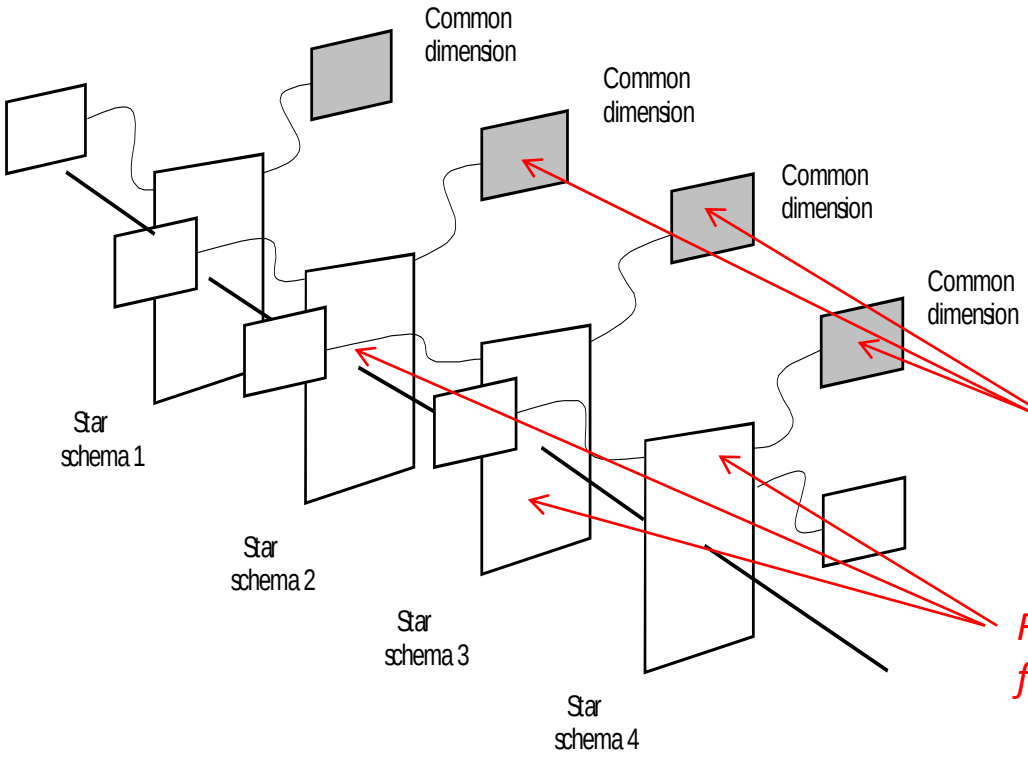
The final star schema



The final star schema



A family of stars

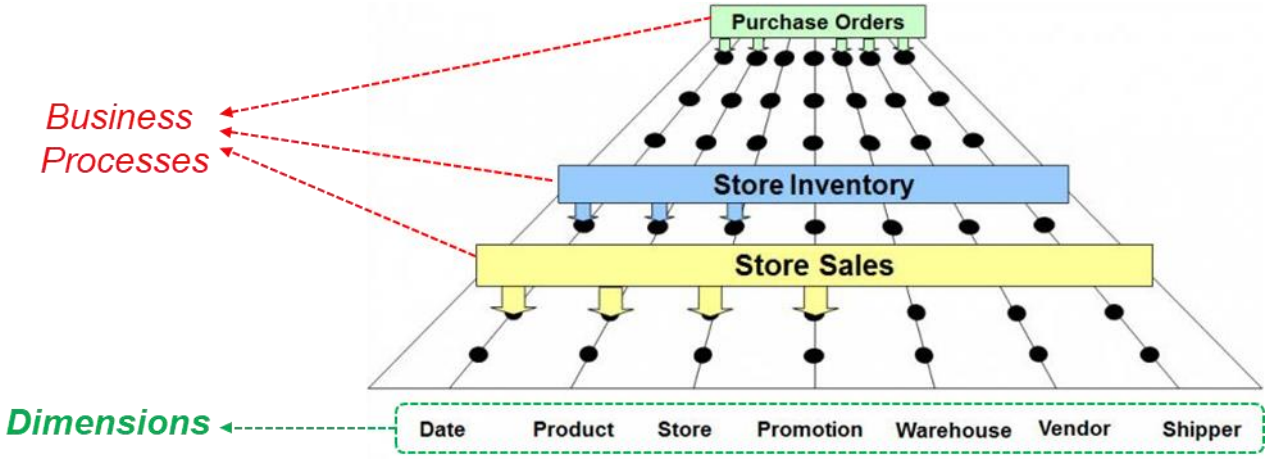


Dimension tables (in this case in the figure the same dimension for all fact tables)

Fact tables (different fact tables)

There are many processes

- There are many business processes. Therefore, we need many stars schemas. Therefore, We need an enterprise architecture, called the **Enterprise Data Warehouse Bus Architecture**



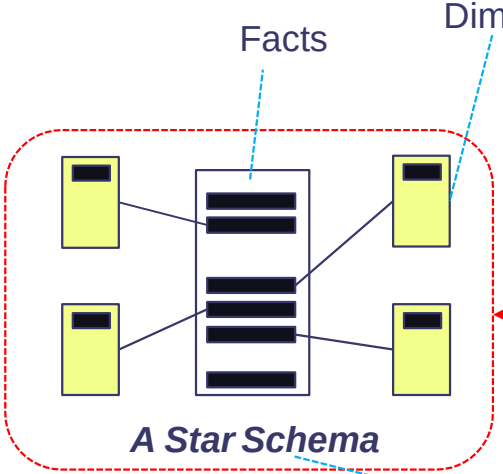
Enterprise Data Warehouse Bus Matrix

- Enterprise Data Warehouse Bus Matrix
  - Rows represent business processes
  - Columns represent dimensions
  - A dimension that is shared among several business processes is called conformed dimension
  - It is a good idea to create a list of core dimensions early
- Benefits:
  - The matrix delivers the big picture perspective, regardless of database or technology preferences.

	COMMON DIMENSIONS						
BUSINESS PROCESSES	Date	Product	Warehouse	Store	Promotion	Customer	Employee
Issue Purchase Orders	X	X	X				
Receive Warehouse Deliveries	X	X	X				X
Warehouse Inventory	X	X	X				
Receive Store Deliveries	X	X	X	X			X
Store Inventory	X	X		X			
Retail Sales	X	X		X	X	X	X
Retail Sales Forecast	X	X		X			
Retail Promotion Tracking	X	X		X	X		
Customer Returns	X	X		X	X	X	X
Returns to Vendor	X	X		X			X
Frequent Shopper Sign-Ups	X			X		X	X

Enterprise Data Warehouse Bus Matrix

- Enterprise Data Warehouse Bus Matrix



	COMMON DIMENSIONS						
BUSINESS PROCESSES	Date	Product	Warehouse	Store	Promotion	Customer	Employee
Issue Purchase Orders	X	X	X				
Receive Warehouse Deliveries	X	X	X				X
Warehouse Inventory	X	X	X				
Receive Store Deliveries	X	X	X	X			X
Store Inventory	X	X		X			
Retail Sales	X	X		X	X	X	X
Retail Sales Forecast	X	X		X			
Retail Promotion Tracking	X	X		X	X		
Customer Returns	X	X		X	X	X	X
Returns to Vendor	X	X		X			X
Frequent Shopper Sign-Ups	X			X		X	X

The star schema is designed based on one business process

Enterprise Data Warehouse Bus Matrix

- Conformed dimensions – the same dimensions are used by several facts
- Thereby, it possible to join different star schemas if needed

	COMMON DIMENSIONS						
BUSINESS PROCESSES	Date	Product	Warehouse	Store	Promotion	Customer	Employee
Issue Purchase Orders	X	X	X				
Receive Warehouse Deliveries	X	X	X				X
Warehouse Inventory	X	X	X				
Receive Store Deliveries	X	X	X	X			X
Store Inventory	X	X		X			
Retail Sales	X	X		X	X	X	X
Retail Sales Forecast	X	X		X			
Retail Promotion Tracking	X	X		X	X		
Customer Returns	X	X		X	X	X	X
Returns to Vendor	X	X		X			X
Frequent Shopper Sign-Ups	X			X		X	X