

Mobile Store Check List

TopGun Program

Fresher Academy - 2018



Day 2



1. Database Design Concepts

2. Database Design Best Practices

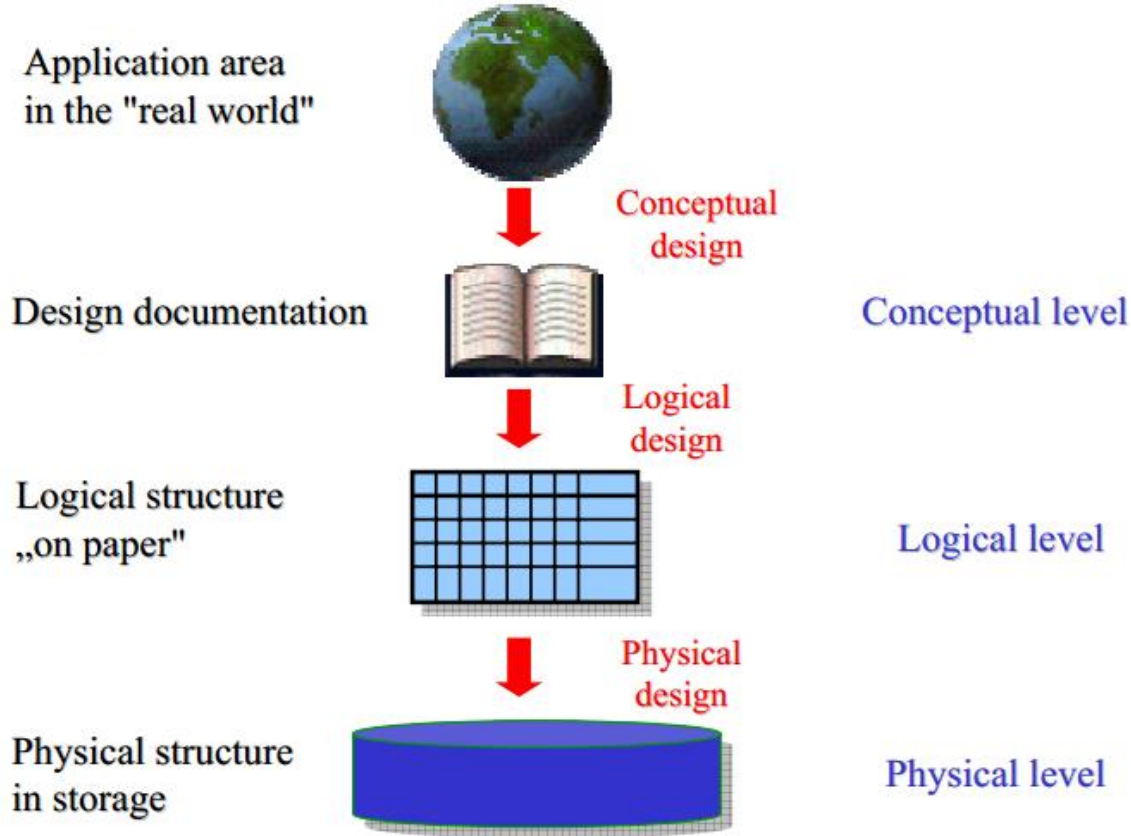
3. Database Design with OOAD

1. Database Design Concepts

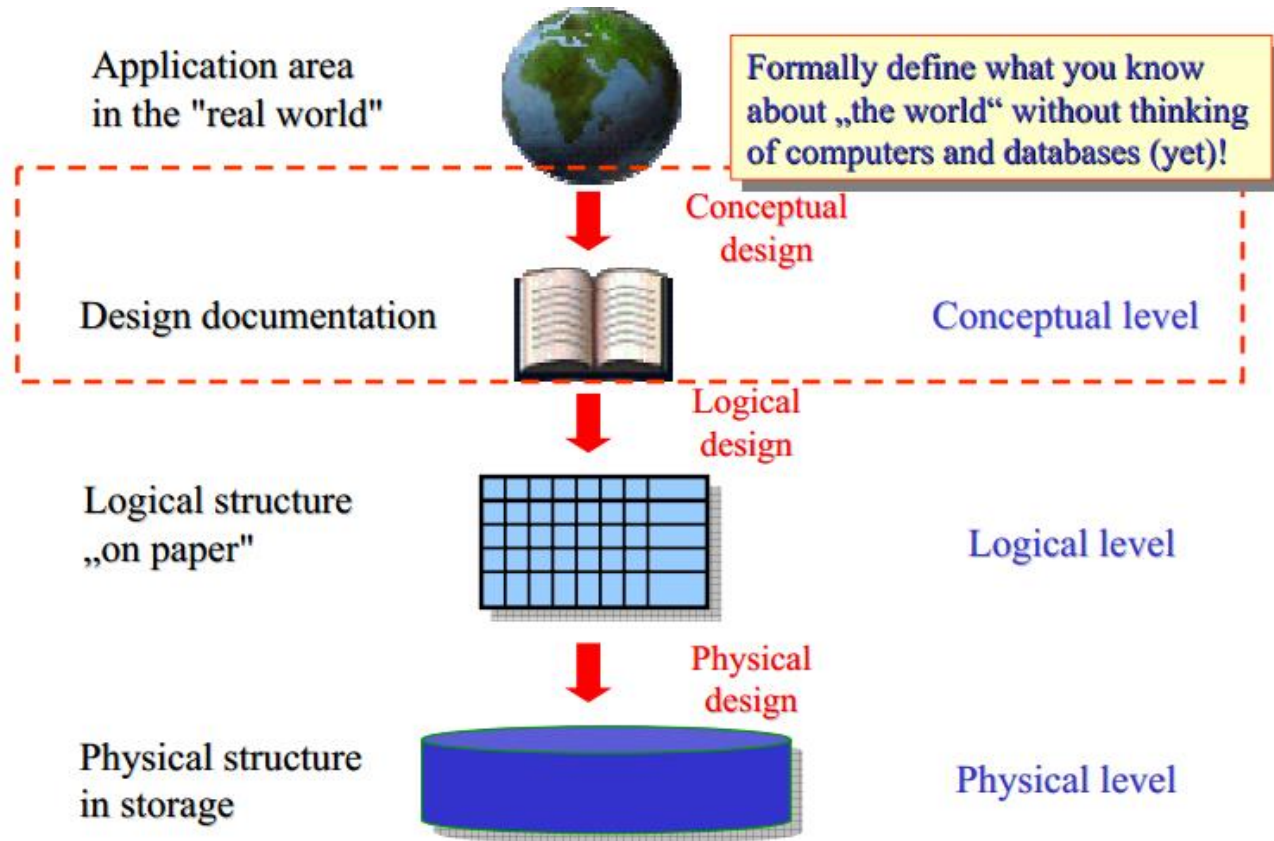


Fpt Software

Phases in DB Design

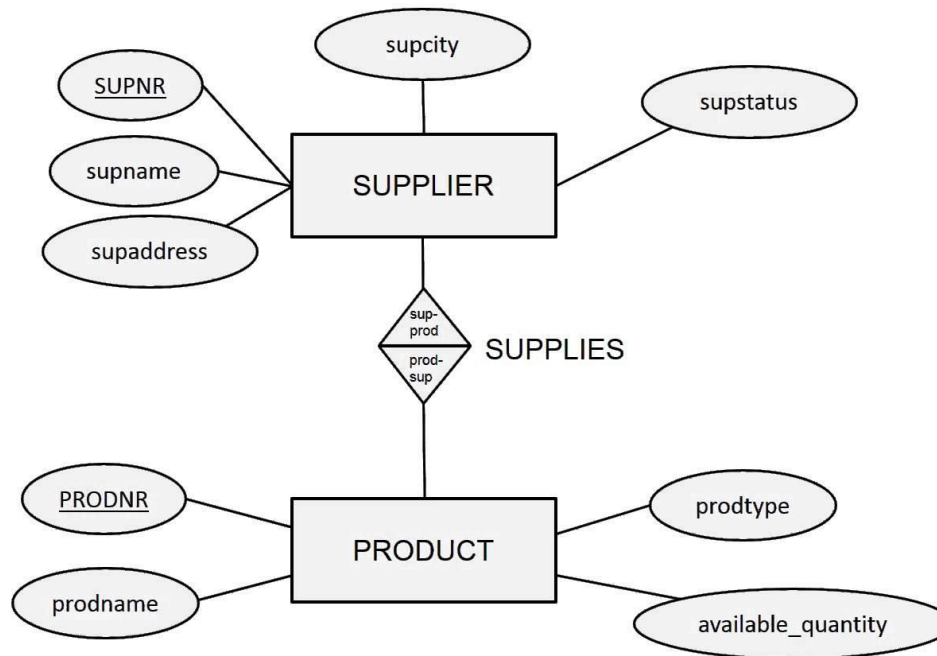


Phases in DB Design





The Entity-Relationship model



ER Model



- **Entity-Relationship data model** (ER model):
 - Proposed in 1976 in a paper by **Peter Chen**
 - **Graphical** notation for application modeling (ER diagrams)
 - Independent **semantic data model**
(aiming at the meaning of concepts in real world)
 - Predecessor of today's object-oriented data models
 - Extremely successful as a means of „**pre-design**“ of relational DBs
- The ER model offers few very simple and basic **concepts**:
 - **Entities** (objects), characterized by **attributes** (properties)
 - Binary or n-ary **relationships** between entities,
possibly characterized by attributes as well
 - Often not mentioned explicitly, but important and basic:
Values: printable symbols as values of attributes;
play a subordinate role (characterizing objects)
 - **Roles**: Names for the special meaning an entity has within a
relationship

ER Model Symbols



Entity



Attribute



Relationship







**Weak
Entity**



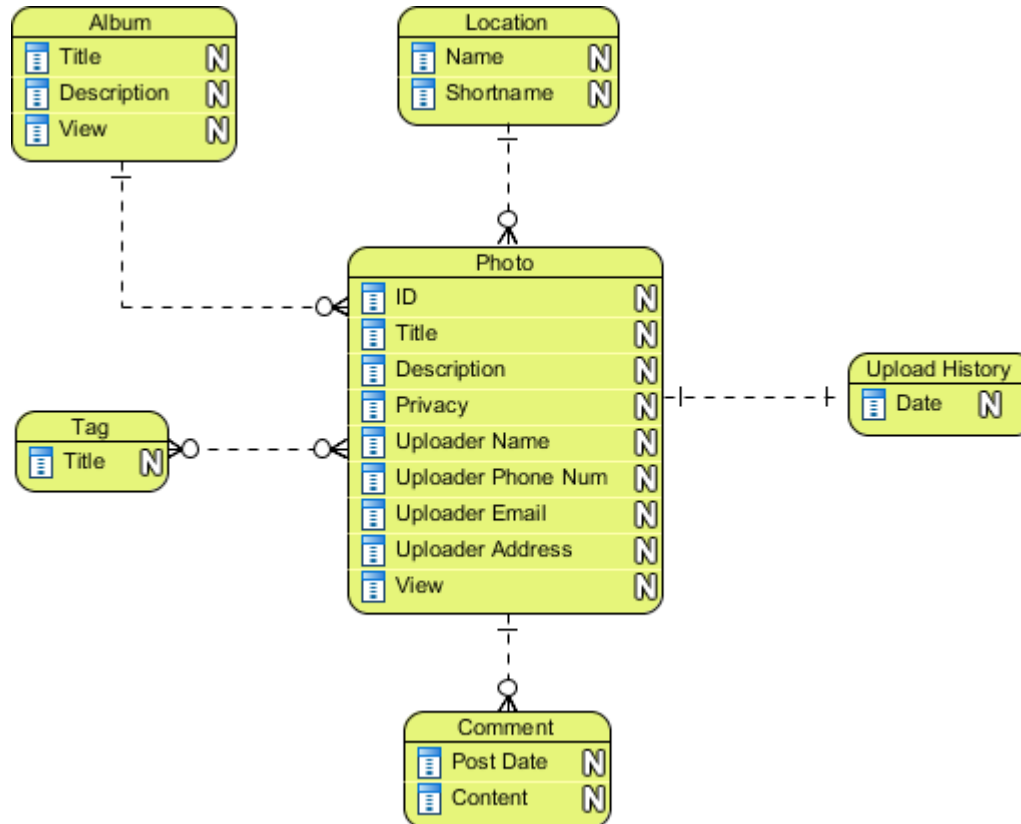
**Multivalued
Attribute**



**Weak
Relationship**

Name	Symbol	Meaning
Oval		Shows different attributes
Rectangle		Shows entity set
Diamond		Show relationship among entity set
Line		Links entity set to attributes & entity set to relationship

ERD Sample

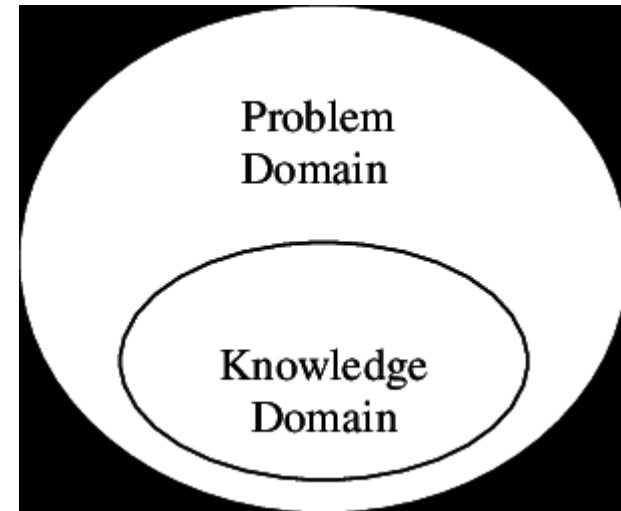
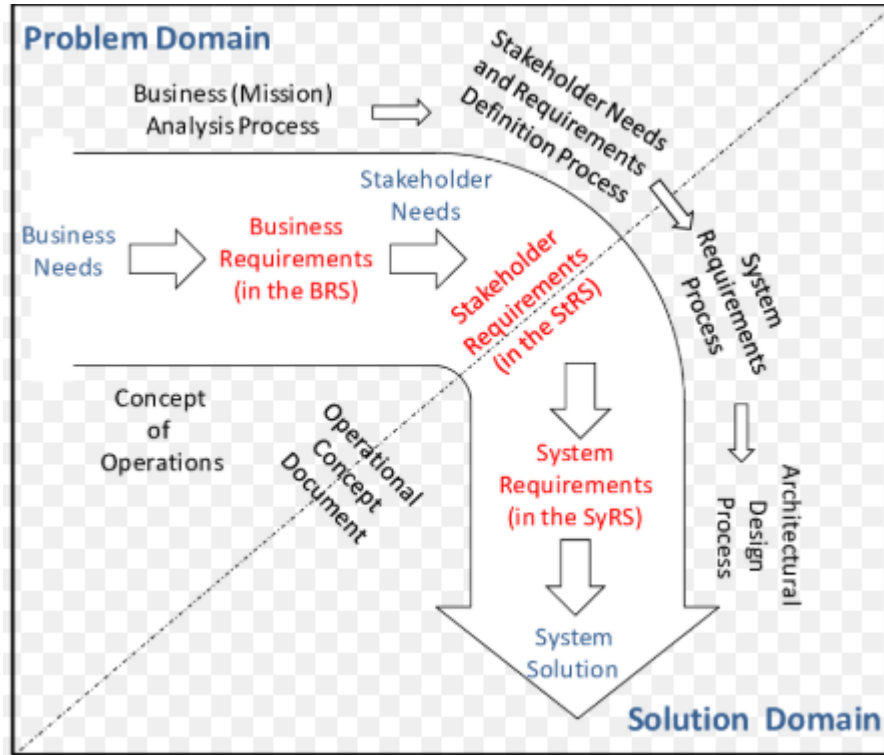


2. Best Practices in DB Design

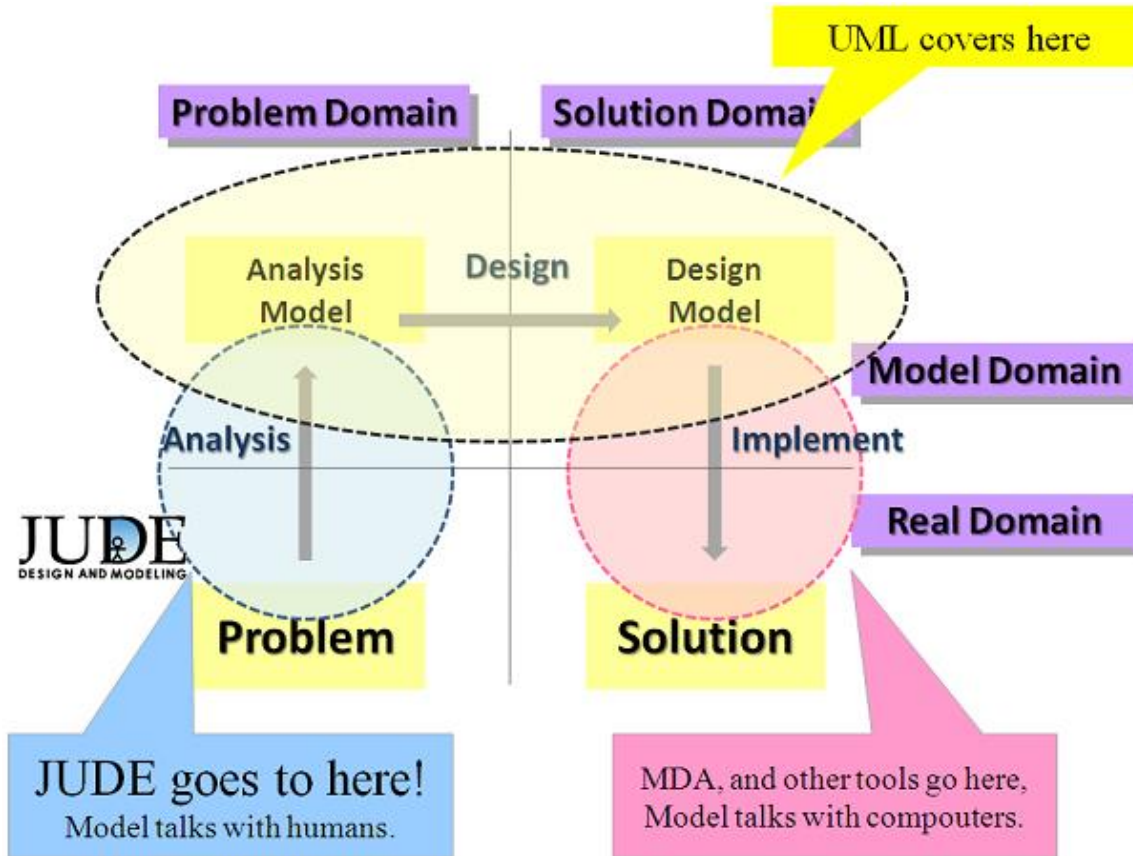


- Good knowledge about problem domain
- Good knowledge about data types
- Data normalization
- Other: apply techniques (indexes, queries...)

Problem domain



Problem domain



Data Types



	postgresql	sqlite	sqlserver	sybase
:binary	bytea	blob	image	image
:boolean	boolean	boolean	bit	bit
:date	date	date	date	datetime
:datetime	timestamp	datetime	datetime	datetime
:decimal	decimal	decimal	decimal	decimal
:float	float	float	float(8)	float(8)
:integer	integer	integer	int	int
:string	(note 1)	varchar(255)	varchar(255)	varchar(255)
:text	text	text	text	text
:time	time	datetime	time	time
:timestamp	timestamp	datetime	datetime	timestamp

Data Normalization



Database Normalization

- Well-Structured Relations (Normalization goal)
 - A relation that contains minimal data **redundancy** and allows users to insert, delete, and update rows without causing data anomalies (**inconsistencies**).
- Technical definition
 - Normalization is a formal process of eliminating redundancies and decomposing relations with **anomalies** to produce smaller, well-structured relations.

Data Normalization



1NF

- Get rid of any columns that hold the same data
- Split up data that can be
- Each Row must be unique

2NF

- Get rid of data not dependant on EVERY part Primary Key

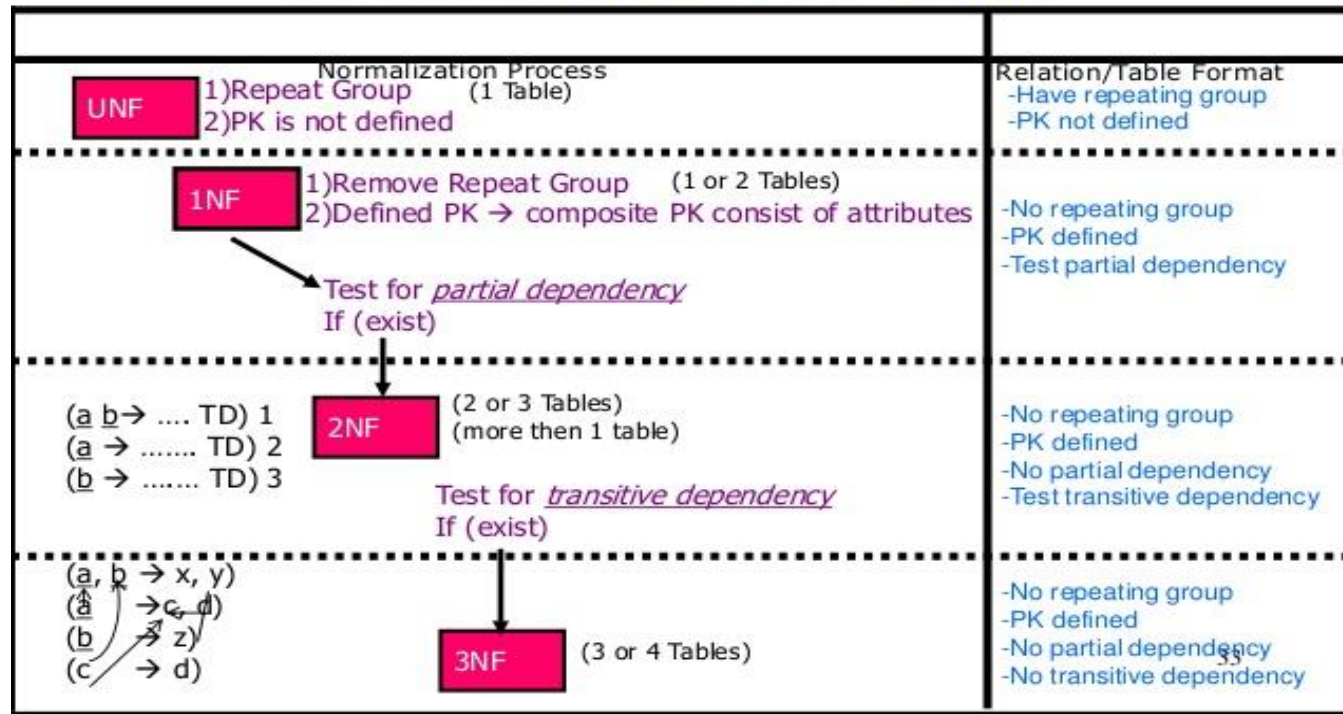
3NF

- Keep Splitting it up.
- No Non-key attribute should be dependant on another non-key attribute

Normalization Process



Normalization Process



1 NF



First Normal Form (1NF)

A table is said to be in 1NF, if

1. The data in each column **should be atomic**. No multiple values, separated by comma.
2. The table does not contain any **repeating column groups**
3. Identify each record uniquely using **primary key**.

Non Atomic Employee Column

DeptName	Employee
IT	Sam, Mike, Shan
HR	Pam

Problems of Non Atomic Columns

It is not possible to SELECT, INSERT, UPDATE and DELETE just one employee

No Repeating Column Groups

DeptName	Employee1	Employee2	Employee3
IT	Sam	Mike	Shan
HR	Pam		

Problems of Repeating Column Groups

More than 3 employees - Table structure change required
Less than 3 employees - Wasted disk space

First Normal Form
1 NF

Primary Key

Deptid	DeptName
1	IT
2	HR

Foreign Key

Deptid	Employee
1	Sam
1	Mike
1	Shan
2	Pam

First Normal Form
1 NF

1 NF - Example



1st NORMAL FORM EXAMPLE

Un-normalized Students table:

<u>Student#</u>	AdvID	AdvName	AdvRoom	Class1	Class2
123	123A	James	555	102-8	104-9
124	123B	Smith	467	209-0	102-8

Normalized Students table:

<u>Student#</u>	AdvID	AdvName	AdvRoom	Class#
123	123A	James	555	102-8
123	123A	James	555	104-9
124	123B	Smith	467	209-0
124	123B	Smith	467	102-8



Second normal form (2NF)

In each relation we need to have a key (called a primary key) on which all other attributes (column values) need to depend. For example, if the ID of a student is given, it should be possible to find the student's name.

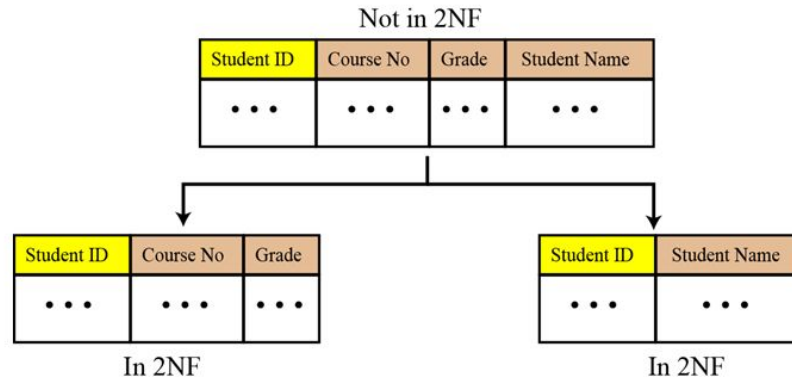


Figure 2.20 An example of 2NF

2 NF-Example



2NF Example

Divide the relation into two relations that now meet 2NF

STUDENT_ACTIVITY

<u>Student_ID</u>	<u>Activity</u>
-------------------	-----------------

Key: Student_ID and Activity

ACTIVITY_COST

<u>Activity</u>	Fee
-----------------	-----

Key: Activity

Activity → Fee

Student_ID	Activity
222-22-2020	Swimming
232-22-2111	Golf
222-22-2020	Golf
255-24-2332	Hiking

Activity	Fee
Swimming	30
Golf	100
Hiking	50

3 NF



Third Normal Form (3NF)

A table is said to be in 3NF, if the table

1. Meets all the conditions of 1NF and 2NF

2. Does not contain columns (attributes) that are **not fully dependent** upon the **primary key**

EmpId	EmployeeName	Gender	Salary	AnnualSalary	DeptId
1	Sam	Male	4500	54000	1
2	Pam	Female	2300	27600	2
3	Simon	Male	1345	16140	1
4	Mary	Female	2567	30804	2
5	Todd	Male	6890	82680	1

EmpId	EmployeeName	Gender	Salary	DeptName	DeptHead
1	Sam	Male	4500	IT	John
2	Pam	Female	2300	HR	Mike
3	Simon	Male	1345	IT	John
4	Mary	Female	2567	HR	Mike
5	Todd	Male	6890	IT	John

EmpId	EmployeeName	Gender	Salary	DeptId
1	Sam	Male	4500	1
2	Pam	Female	2300	2
3	Simon	Male	1345	1
4	Mary	Female	2567	2
5	Todd	Male	6890	1

DeptId	DeptName	DeptHead
1	IT	John
2	HR	Mike

3 NF - Example



Example in 3NF

REGISTRATION FORM

STUDENT NUMBER		S0843215	
STUDENT NAME		P. Smith	
STUDENT ADDRESS		1, South Downs Hale	
COURSE NO	COURSE	TUTOR NAME	TUTOR NO
PM951	Computing	T. Long	037428
S212	Biology	S. Short	096524

ENTITY TYPES IN 3NF

STUDENT (Student#, Student-Name, Student-Address)

ENROLMENT (Student#, Course#, Tutor-Staff#)

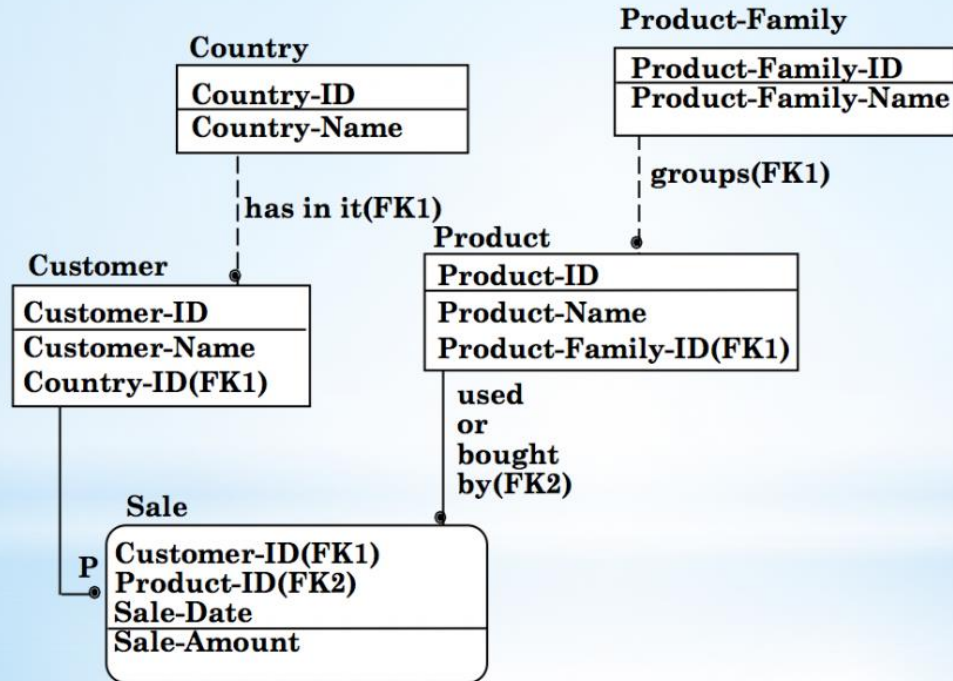
COURSE (Course#, Course-Title)

TUTOR (Tutor-Staff#, Tutor-Name)

3 NF - Example



3rd Normal Form Model





3NF and BCNF Revised

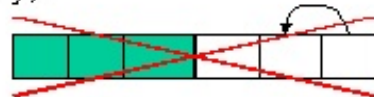
3NF: a relation in 2NF and in which no non-primary key attribute is transitively dependent on the primary key.

i.e. There are no non-key attributes with dependencies on other non-key attributes (except candidate key).

3NF



2NF and no transitive dependencies



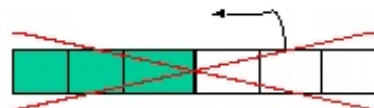
BCNF: a relation in 3NF and in which there are no dependencies of part of the compound key on another attribute.

i.e. Every determinant is a candidate key.

BCNF



3NF and all determinants are candidate key



BCNF-Example



Boyce-Codd Normal Form

- Name/phone example is not BCNF:

Name	<u>SSN</u>	Mailing-address	<u>Phone</u>
Michael	123	NY	212-111-1111
Michael	123	NY	917-111-1111

- {ssn,phone} is key
- FD: **ssn** → **name,mailing-address** holds
 - Violates BCNF: **ssn** is not a superkey
- Its decomposition is BCNF
- Only superkeys → anything else

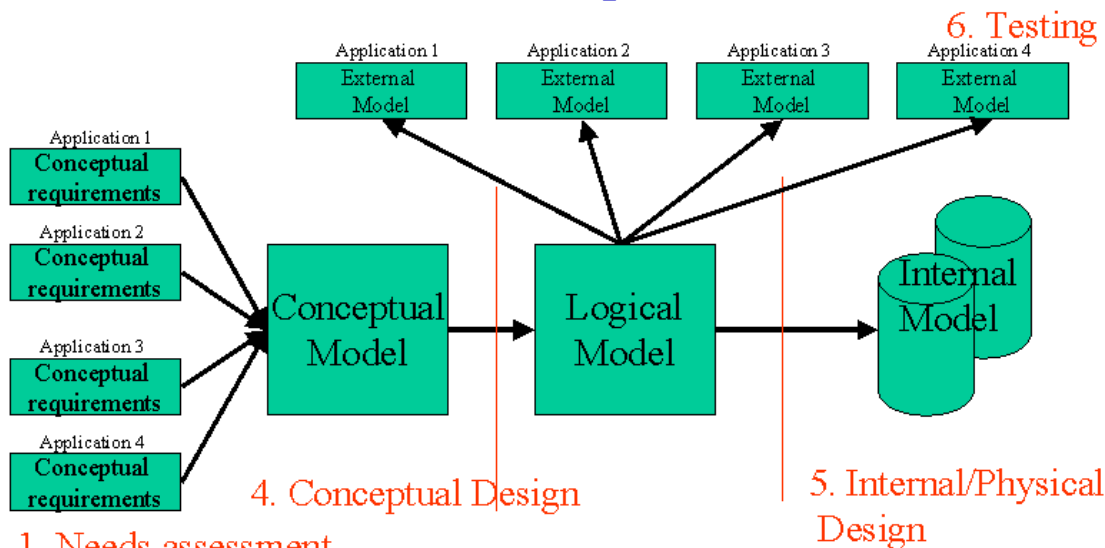
Name	<u>SSN</u>	Mailing-address
Michael	123	NY

<u>SSN</u>	<u>PhoneNumber</u>
123	212-111-1111
123	917-111-1111

DB Design Process Summary



Database Design Process



1. Needs assessment
2. Requirements analysis
3. Data identification

3. Design DB with OOAD



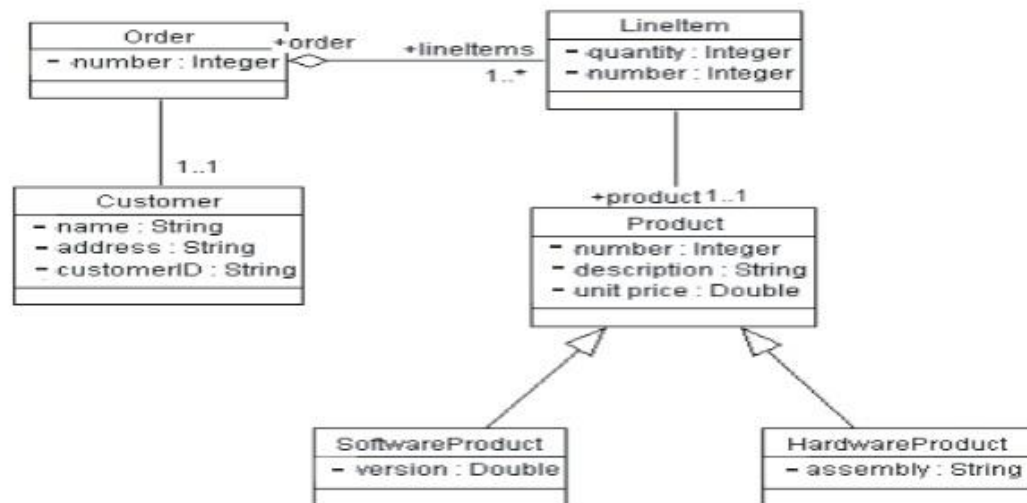
Fpt Software



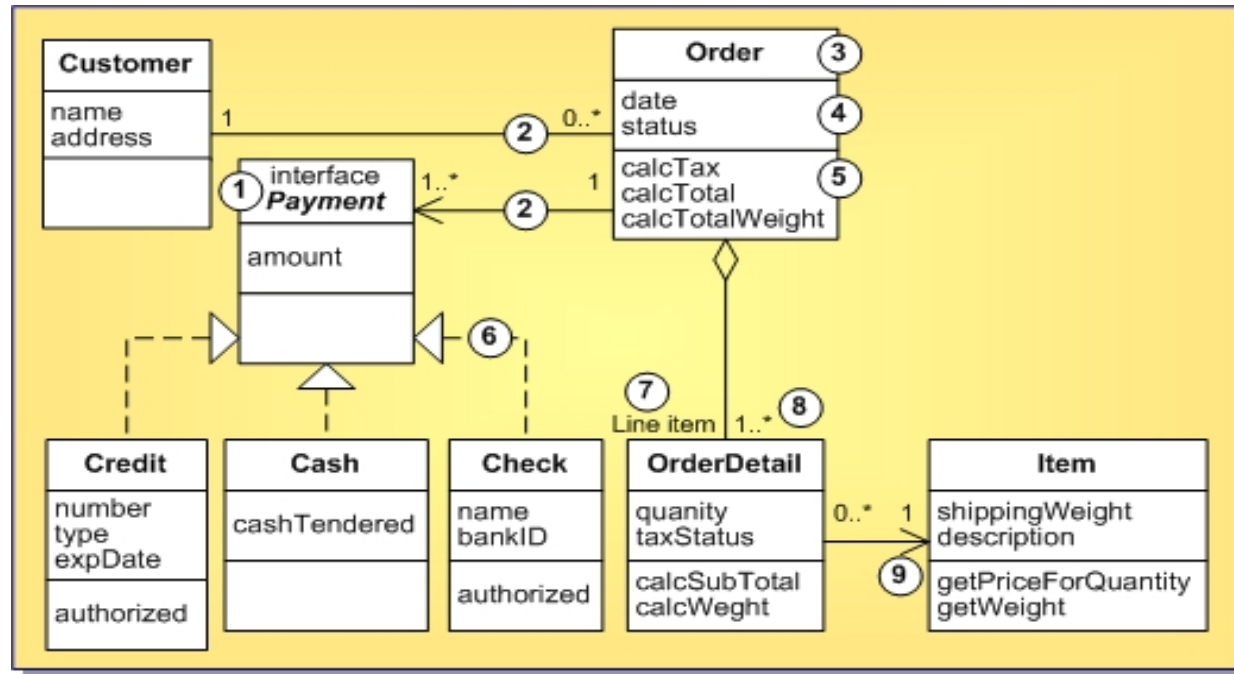
Developing list of Inputs and data requirement

Object Oriented Approach

✓ Design class diagrams

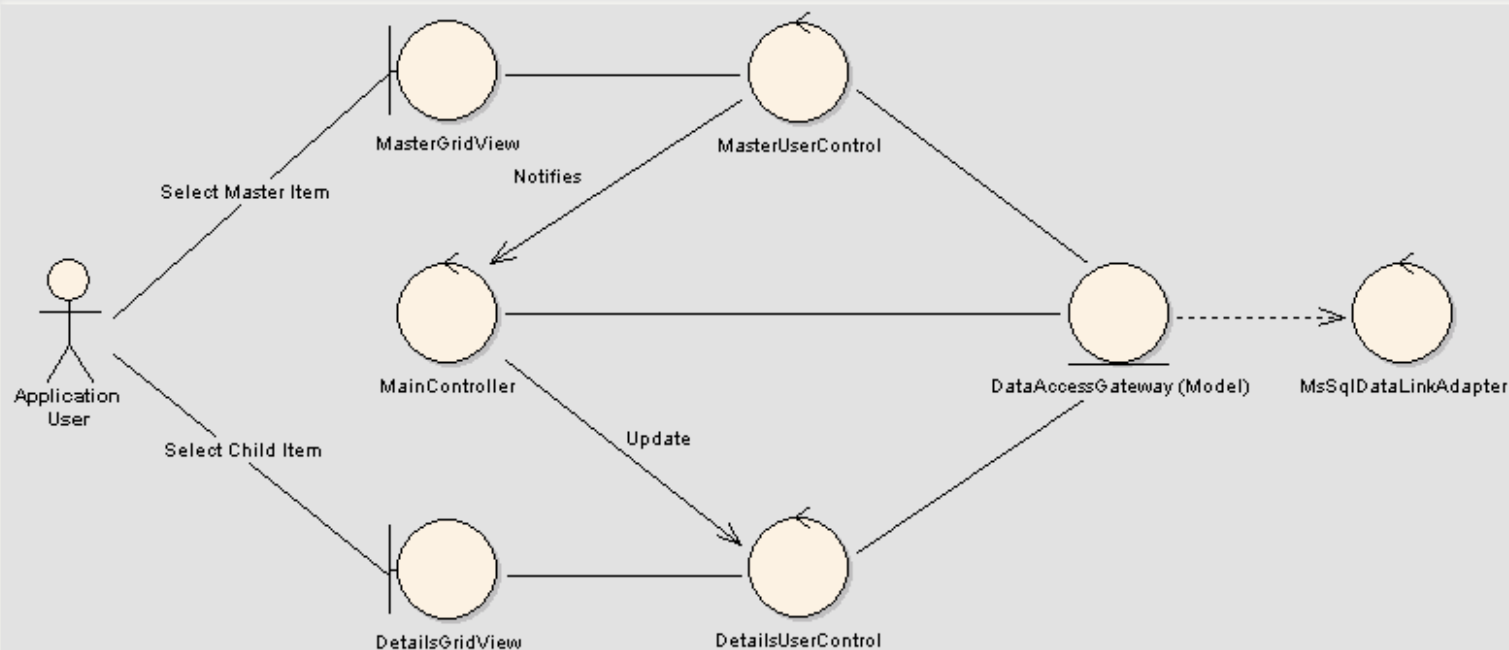


Class Diagram



- | | | | | |
|---------------|--------------|------------------|----------------|----------------|
| ① Interface | ③ Class Name | ⑤ Methods | ⑦ Role Name | ⑨ Navigability |
| ② Association | ④ Fields | ⑥ Implementation | ⑧ Multiplicity | |

Analysis Class Diagram



Robustness Analysis Diagram for Master-Details Tables Relationship

Happy Coding!



thank you!



Fpt Software