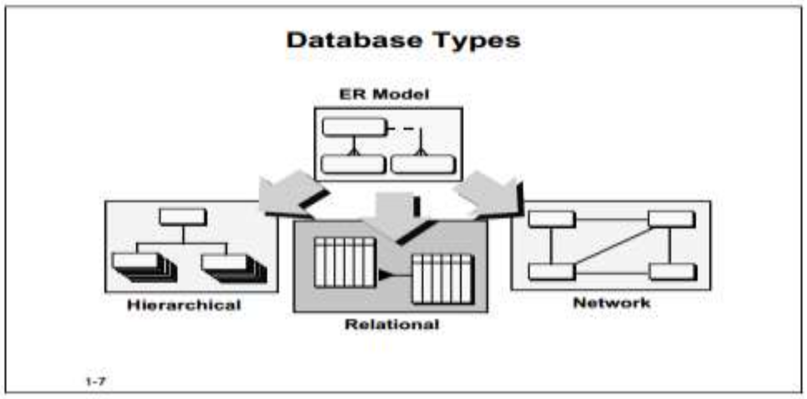
Physical Data Model and Database Design

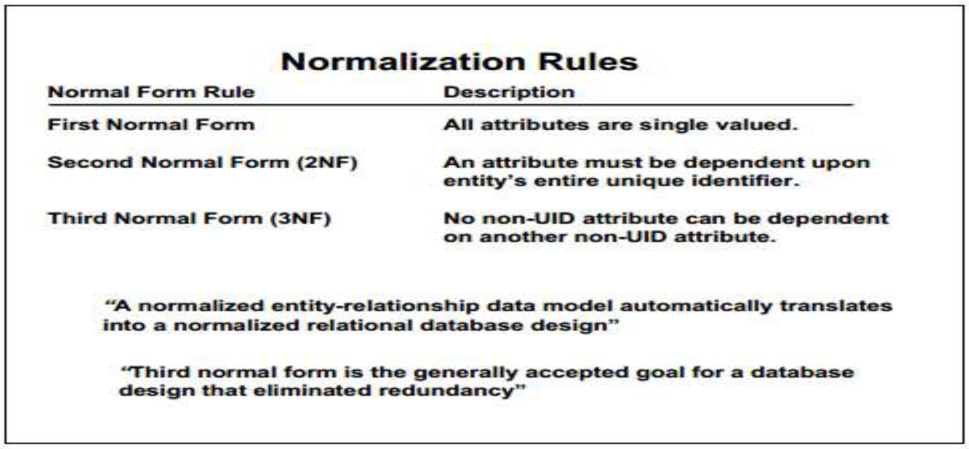
Introduction

When building an entity relationship (ER) model we tend to use it to later build different physical models of database types. Therefore a physical data model is used to implement into different technical software and hardware environments that is due to the current state of technology and is changing as technologies change. It means that physical model design done five or ten years ago is outdated today.



NORMALIZATION

It is a relationship database concept and is done in the process of building ER. If the correct entity model is being built will conform to the rules of normalization. Each rule has corresponding data model interpretation, which can be used to validate placement of attributes in ER model



Normalization Rules

Normal Form Rule	Description
First Normal Form	All attributes are single valued.
Second Normal Form (2NF)	An attribute must be dependent upon entity's entire unique identifier.
Third Normal Form (3NF)	No non-UID attribute can be dependent on another non-UID attribute.

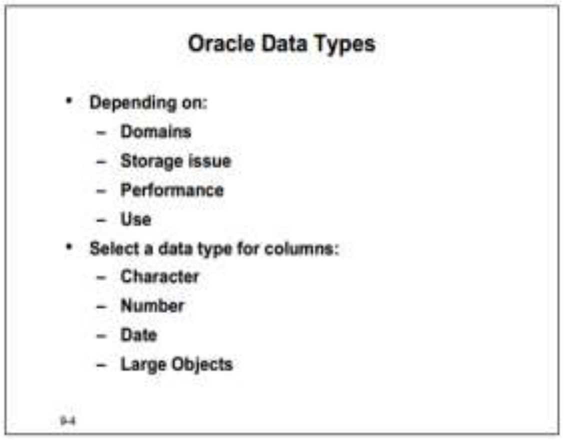
"A normalized entity-relationship data model automatically translates into a normalized relational database design"

"Third normal form is the generally accepted goal for a database design that eliminated redundancy"

First normal form （1NF)	Second normal form (2NF)	Third normal form (3NF)
	• T13_DEPARTMENT	• T13_CUSTOMER
	• T13_OFFICE	• T13_RECEIPT
	• T13_VEHICLE	• T13_APPLICATION
	• T13_NOK	• T13_STAFF
	• T13_INSURANCE_POLICY	• T13_INSURANCE_COMPANY
	• T13_CLAIM	• T13_MEMBERSHIP
	• T13_CLAIM_SETTLEMENT	• T13_PRODUCT
	• T13_PREMIUM_PAYMENT	• T13_COVERAGE
	• T13_QUOTE	• T13_VEHICLE_SERVICE
	• T13_INCIDENT_REPORT	• T13_INCIDENT
	• T13_POLICY_RENEWABLE	

PHYSICAL MODEL

When creating physical models we create tables or clusters and we must write specifications of internal data type for each of its columns.These types define generic domain of values that each column can contain

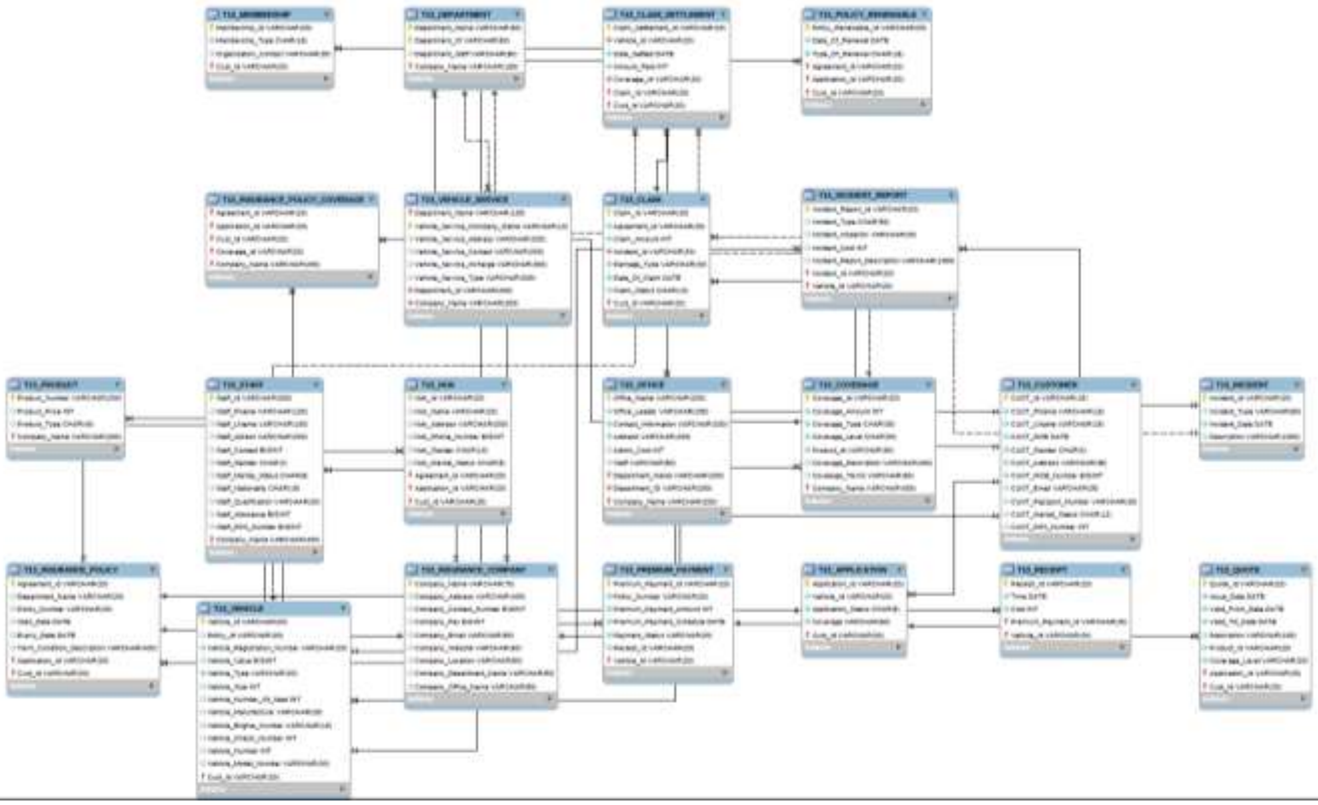


Oracle Data Types

- Depending on:
  - Domains
  - Storage issue
  - Performance
  - Use
- Select a data type for columns:
  - Character
  - Number
  - Date
  - Large Objects

9-4

Graphical presentation of PDM :-



Conclusion

We have a class project created database (DB) with all documentations and reports included. Our goal was to create a DB for Online vehicle insurance companies with code generated for Oracle and MS Access.
There were some big and small challenges but we succeeded in making a functional DB. We started to build conceptual data models (CDM) , we continued with logical data models (LDM) and then we made physical data models (PDM) all in the Erwin software program. From the physical data model we created a code to be rune in Oracle and MS Access database management system (DBMS). For better understanding for a reader and for our learning we included some theory in each phase we did and documented in the project initial document (PID) with reports of progress and work being done.