10/20/2020

CS301-DBMS- V semester , IIIT Dharwad

Database design for a Vehicle Insurance Company

Table of Contents

Figures

Figures LDM

Tables

Tables LDM

Appendix

Project Initiation Document

ABOUT A PROJECT

Section 1: What is the project all about?

Project title:
A database for a Vehicle Insurance Company

Background:
The aim of the Project is to maximize the applied learning of each student as taking courses in DBMS in IIIT Dharwad in 2020 – 2021 study year.

Project Definition

Purpose:
This module provides a comprehensive discussion of, and practical experience in, advanced entity modelling; normalisation; transactional relational database design; SQL and PL/SQL coding; and generation of data backed management reports. Students gain practical experience using contemporary database modelling and design tools and technologies, and apply sound design principles for creating effective decision support solutions for realistic business scenarios.

Objectives (and how they will be measured):

Students are expected to interact with their group colleagues to develop a database in three steps: PART A: Conceptual Data Model (CDM) - 10%; PART B: Logical Data Model (LDM) – 15%; PART C: Physical Data Model (PDM) – 25% of the project mark. The onus is on each group to develop their own style of Project report but learning should be applied to this task from the CS301 – DBMS Course.

Project scope:
Boundaries of the project are: knowledge of the students delivering the outputs; their back ground in different fields; learning the process of Data Base (DB) design; not knowing how to use the tools to design DB; what kind of DB to be deliver as regarding to marks given by module; time management;

Exclusions from scope:

Cost calculations of project, administration of DB when project ends, loos of data because of working in cloud.

Deliverables (including dates of completion):

Part A - Submit it to the appropriate folder on Google Classroom by ***5pm on Wednesday 25th of November, 2020 for evaluation***

- GroupX_CDM
- GroupX_CDM_Report

Part B - Submit the model and report to the appropriate folder on Google Classroom by ***5pm on Wednesday 25th of November, 2020 for evaluation***

- GroupX_LDM
- GroupX_LDM_Report

Part C - Submit the Report and the Database Code to the appropriate folder on Google Classroom by ***5pm on Wednesday 25th of November, 2020 for evaluation***
GroupX_PDM

- GroupX_PDM_Report
- GroupX_DatabaseCode
- GroupX_report

Constraints:

External variables that we must manage but are out of our control: other module schedules changes; work been done for other modules that could affect time spent on a project for design of DB; weather conditions – being out of electricity as we use computers to design DB; logistics not coming on time; or internet failure; lost or stolen disk or computer containing the project outputs; team members not working hard.

Assumptions:

Delivering outputs on time that might be influenced by schedule of other modules projects and work being done.

Section 2: Why should this project is important?

Business case:

Project Benefits:

To make a good model of DB for car insurance and get the highest mark. To make a good DB, that could be used with analytical tools and faster in delivering the right data at the right time for better decision making.

Cost and Timescale:

The only cost we have is time of a students used to implement this project.

Sample Business Case analysis :

Assume, for the whole project there is estimated 500 hours for the team of 10 members Team.

Cost/Benefit Analysis:

Costs: 500 hours * (10 members) = 5,000 hours
If each hour costs 1,000 Rs
Total cost of the Project = 5000* 1,000 = 50,00,000 Rs = 50 L

Benefits:

- Learn to build a DB – FREE cost = paying fee = 10000 Rs for DBMS course
- With DB learning skill, an average student may be able to earn a DB job with minimum salary of 50,000/month = 6 L/year.
- For 10 students in a team = 6 L* 10 members = 60 L

Total cost / total revenue (or benefits) = 50 L / 60 L = .8

Since Cost invested in less than expected revenue of the Total team, overall project is beneficial to work.

Risk Analysis:

Risk Identification:

Time management, time scheduling and not sufficient time to do the output, loss of data or outputs, not enough information to build the right DB, delegating the tasks to the right person in the Team.

Risk Prevention:

Make a good time scheduling, make a good research and preparation on the tasks ahead, save data in different locations not to get lost or stolen, use more computers if one will get broken down.

Risk Management:

If there are risks that we could not foresaw or we did not take a precaution to prevent it to happen, we will take a meeting und discuss it inside the group what action to take to eliminate it..

Section 3: Who will work on the project?

Roles and Responsibilities

Project Organization Chart

Names of Project Team:

Name	Roles/job descriptions	Student Number	Phone Number.	e-mail address

Section 4: How and when will the project be delivered? (optional)

Project Plan

Assignments/Milestones:

Schedule (Gantt chart):

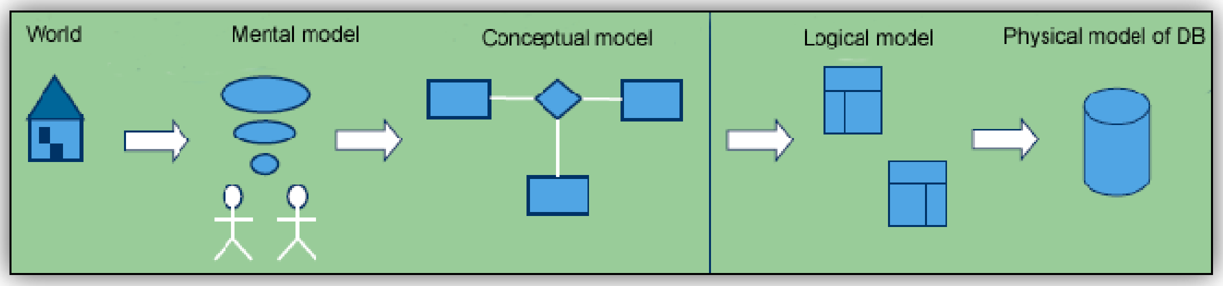
Section 5: Case: XYZ Car Insurance

XYZ is a global service company in 17 countries that has around 43 million customers with insurance, savings and investment products. Aviva Ireland provides general insurance, Life & Pensions and Health Insurance. Larger portion of their business is done online or as they say 'Digital will be central how XYZ operates' and they gave us the chance to build a Data base model for vehicle insurance

Section 6: Data Base Modelling

Data modeling is an activity that makes physical world become digital stored in data base, as seen in Figure 1, how the process from real world become Data base model.

Figure 1: Process of data base building



The output we get from this activity is the data model as static information structure in terms of data entities and their relationship using architectural style.

The data model in principal describes the structure of data entities and their relationships. Like in our case – Vehicle insurance, entities will include Customers, Insurance policy, Employees. Customer has several attributes, as costumer number, address, status and number of insurance policy.

- *Conceptual.* The model focuses on the entities and their relationships and properties that are imbedded in the problem. Best use for communication with stakeholders.
- *Logical.* Is a step from conceptual data model to a data management technology (relation databases) and is subject of normalization.

- *Physical.* This is a model with implementation of data entities. With optimizations that have partitioning or merging entities, duplicating data, creating identification keys and indexes.

Figure 2: Graphical presentation of Conceptual model (Fig.2), Logical model (Fig.3) and Physical model (Fig.4)

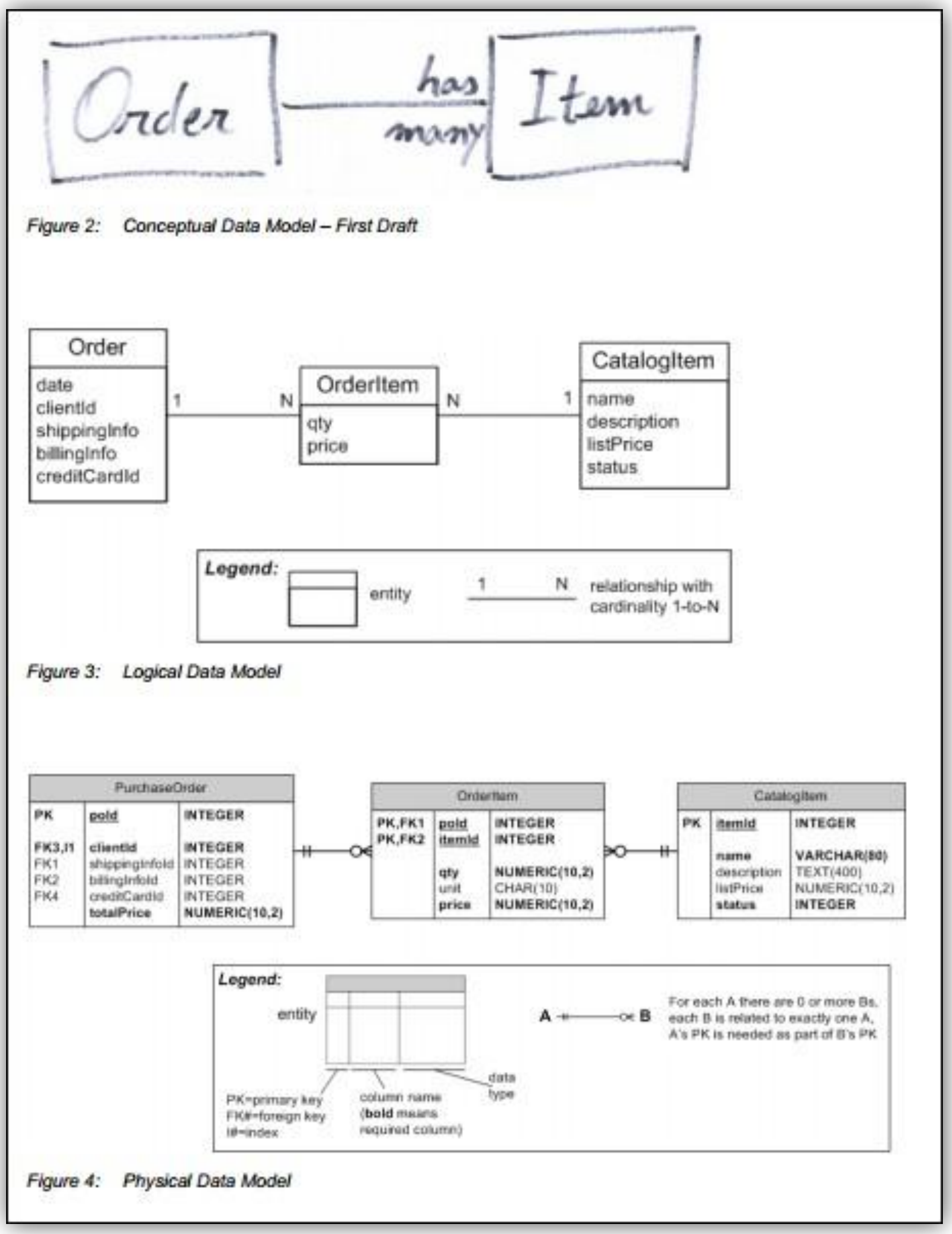


Figure 2: Conceptual Data Model – First Draft

Figure 3: Logical Data Model

Figure 4: Physical Data Model

(Source: http://www.sei.cmu.edu/, (Merson, 2009))

Part A: Conceptual Data Model (10%)

Design Rules

To design our car insurance database conceptual data model we first needed to decide what characteristics underpin the model under investigation. As a group we decided on various rules that need to be implemented in order for the model to be consistent and precise. Table 2 below illustrates these rules.

Table 2: Design rules for CDM model of car insurance database

Design Rule	Description	Example
Rule 1	All individual entity types must be in capital letters	CUSTOMER
Rule 2	An underscore is used to label an entity type with more than one word	TERMS_CONDITIONS
Rule 3	Plurals are not used when labelling entity types	APPLICATION
Rule 4	No abbreviations are used when labelling entity types	QUOTE

Assumptions

To design the Conceptual Data Model (CDM) we certain set of assumptions. These assumptions will help shape our model to allow consistency within our design. Table 3 presents the assumptions used in this model.

Table 3: assumptions used in car insurance database model

Assumption	Description
Assumption 1	Customer must be a permanent international driving licence
Assumption 2	The online insurance has no physical high-street presence
Assumption 3	The online insurance is given to customers over 18 years of age
Assumption 4	The online insurance needs some driving history of customer
Assumption 5	The online insurance needs to know type of car customer drives
Assumption 6	The online insurance needs to know about insurance history of customer

Entity Types

All of the entity types that we feel are relevant in our CDM are illustrated in Table 4, below.

Table 4: Entity types used in car insurance data base system CDM model.

Entity Type	Description
CUSTOMER	Records all the personal details about the customer
APPLICATION	Records details of the insurance cover requested by Customer
QUOTE	Records details of customer potential cost of the insurance product
INSURANCE POLICY	Records details of Insurance agreement
PREMIUM	Records details of customer payments
VEHICLE	Records details of Vehicle model, cost and registration
CLAIMS	Records details of customer claims in case of an incident
SETTLEMENTS	Records details of settlement made on claims
STAFF	Records details of employees
DEPARTMENT	Records details of the various departments
OFFICE	Records details of different office locations
MEMBERSHIP	Records details of customer membership
SERVICE	Records details of different car services offered
NOK	Records details of the next o kin
TERMS_CONDITIONS	Records all terms and conditions in regard to the policy
VEHICLE INSURANCE DEPARTMENT	Records details of vehicle insurance cover
RECEIPT	Records details of Receipt of Premiums
COMPANY	Details of the Insurance organization giving the insurance cover

Relationships in CDM

Applying Relationships to Entities

To apply relationships to our entity types we formed certain assumptions to simplify and determine connections between entity types. These assumptions and explanations are illustrated in Table 5 below.

Table 5: CDM relationship of entities for car insurance database

Entity type	Related To Entities	Relationship
QUOTE	APPLICATION	**one to one**
APPLICATION	INSURANCE POLICY CUSTOMER	**one to many** **one to many**
CUSTOMER	MEMBERSHIP PREMIUM CLAIMS VEHICLE	**many to many** **one to many** **one to many** **one to one, one to many**
INSURANCE POLICY	VEHICLE INSURANCE DEPARTMENT TERM AND CONDITION NOK	**one to many** **many to many** **one to many**
PREMIUM	RECEIPT	**one to many**
CLAIMS	SETTLEMENT	**one to one**
VEHICLE INSURANCE DEPARTMENT	DEPARTMENT SERVICE	**one to one, one to many** **one to many**
DEPARTMENT	OFFICE COMPANY	**many to many** **one to many**
COMPANY	STAFF	**many to many**

Graphical presentation of CDM

The Conceptual Data Model that will be used as a starting point in designing our online car insurance database system can be seen in Figure 3 (with no entities relationships) and Figure 4 (with entities relationships), done in ERwin software.

Figure 3: Conceptual Data Model for Car insurance - character graphic presentation (no entities relationships)

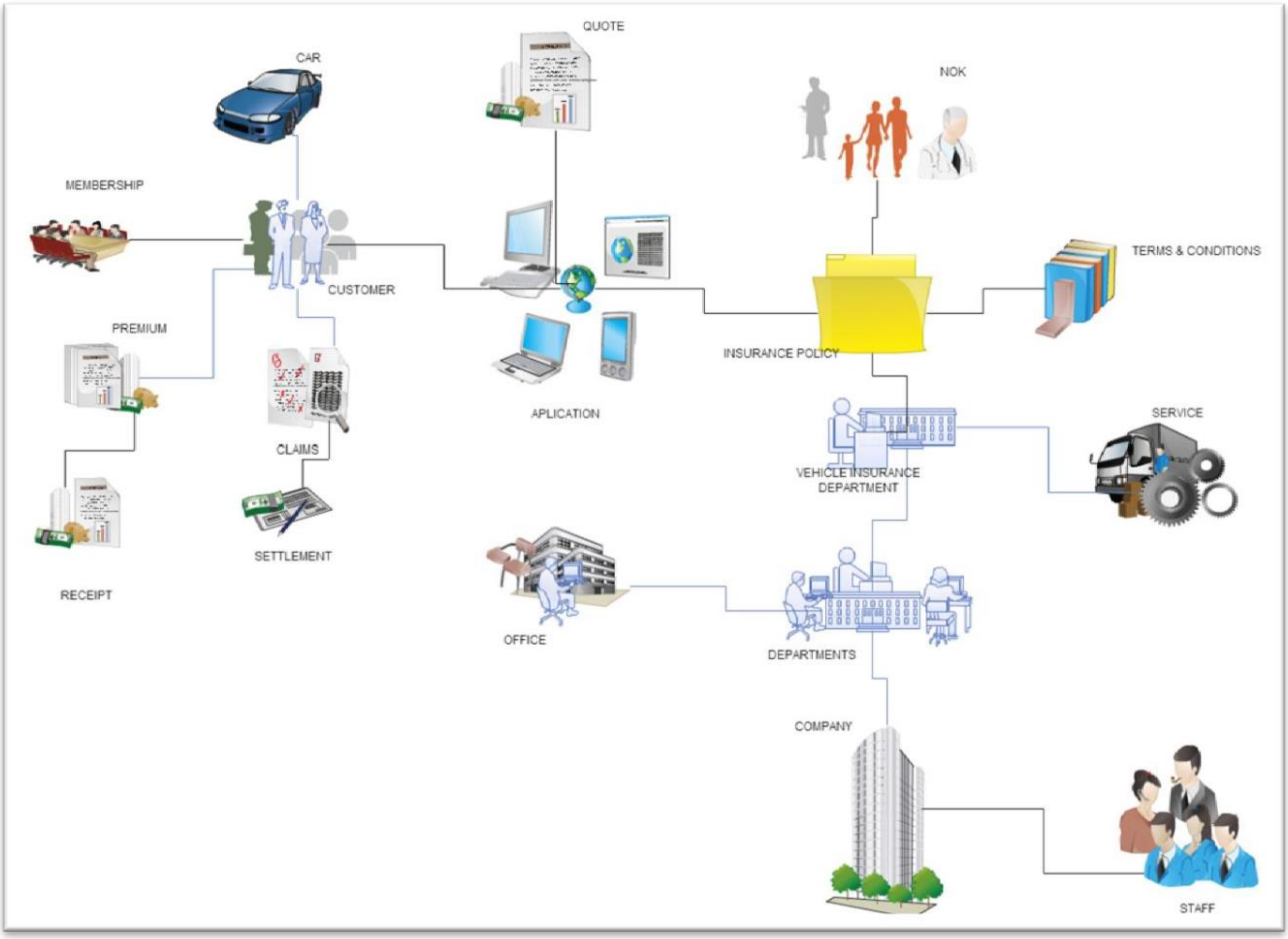
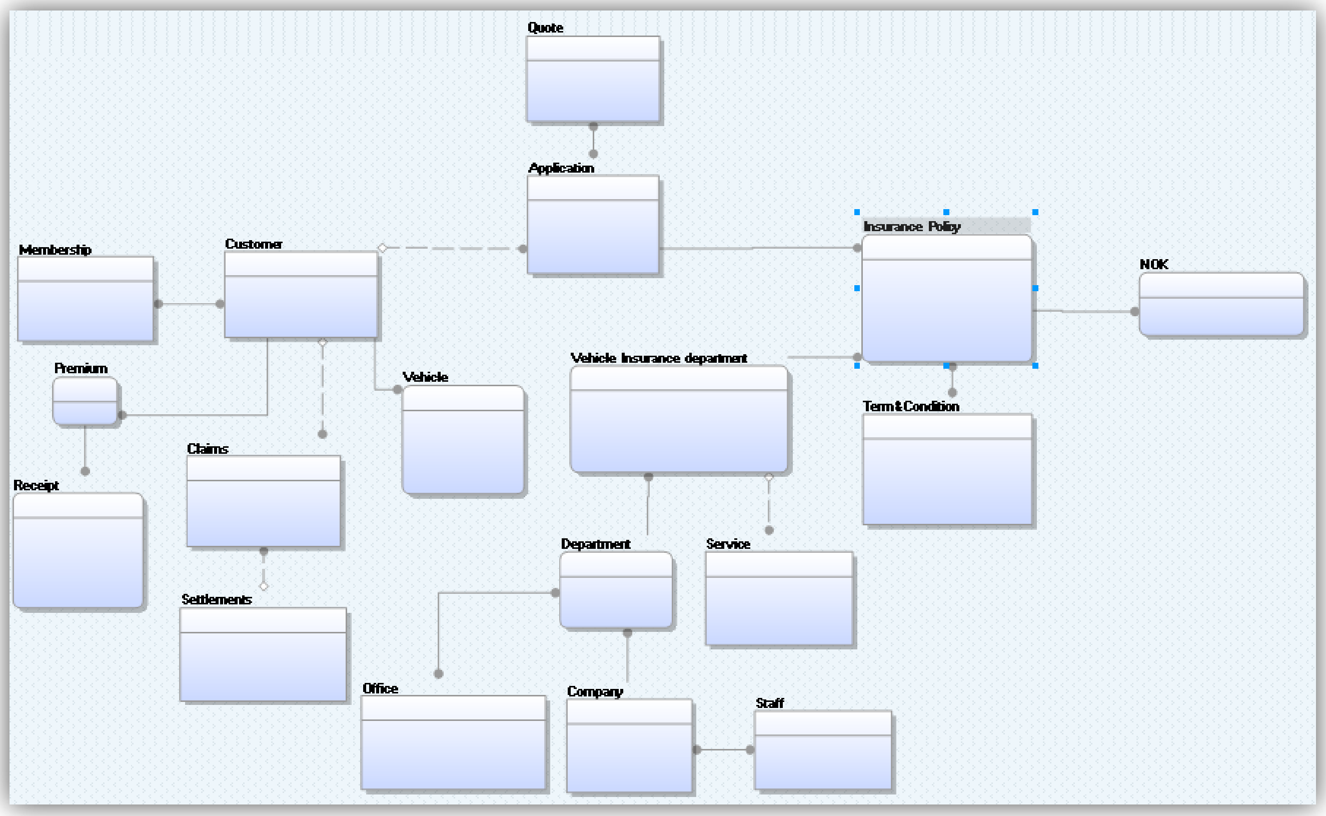


Figure 4: Conceptual data model of Car insurance for AVIVA Ltd. (with entities relationships)



Part B: Logical Data Model (15%)

Introduction

Part B of this report includes design of logical data model (LDM) for vehicle insurance company XYZ Ltd.

First we had to make some changes and improvements to the conceptual data model which are explained and presented in last part of this report. Then we identify all the attributes in old and new entities and assigned them the primary (PK) and foreign keys (FK) and made relationships with them in Erwin to make a full LDM. We identified each variable and the data types that this LDM could be used to design the database in Access or Oracle data base management systems.

Introduction of terms used for constructing LDM

Elements

Elements in data model are named entities. This is any distinguishable object that presents part of data base. It can be related to any object in real world such as: a car, a customer (person), a policy, a company, etc. with respective attributes that are relevant to the software system.

Properties of **entities** can have values:

- Name
- Description of the meaning and significance
- Weather entity is dependent or non-dependent
- List of **attributes** (Car entity: year, manufacturer, model, mileage, owner, licence, book of maintenance) with properties (data type, size, is it required or not).
- The attributes (or attribute) are used to precisely identify an entity (primary key – PK, foreigner key – FK, …)
- Constraints of individual or combined attributes values (e.g. date of issue of new policy can't be prior to renewal date of policy)
- Rules to grant permission to users or user groups to access the entity
- Expected number of entity instances and expected growth rate

Or additional:

- List of attributes to be indexed to optimize access time
- List of attributes to be encrypted or compressed
- Weather entity should become a database view or a table

- Weather entity should become a materialized view
- List of database triggers to be implemented for that entity.

Relations

Relationship - Designates logical association between entities, with cardinality of the participant entities: one-to-one, one-to-many, or many-to-many relationships. Relationships can be identifying or non-identifying (identifying A-B; existence of B depends on existence of A).

Generalization/specialization – Indicates an "is a" relationship between entities. For example department entity is a generalization of different types of departments; at the same time vehicle insurance department or travel insurance department is specialization of department entity.

Aggregation - is an abstraction that turns relationship between entities into an aggregate entity, rarely used. Example: "customer-insurance advisor -date" can be an aggregate entity called Appointment.

Constraints

The database normalization technique is used to impose restrictions on data model that is based no dependencies between entities and their attributes. Normalization is used with the goal objective to avoid duplication of information in order to safe guard the consistency (integrity) of the data.

Data types

When we assign attributes to entities with primary keys and foreign keys do the normalization, we identify each attribute with data type for each data management system – Access and Oracle as seen in example Table 6, below.

Table 6: Example of data types in Access and Oracle DMS

Entity type	Attributes	Data type Access	Data type Oracle
CUSTOMER	FIRST_NAME	Text	varchar2(size)
	LAST_NAME	Text	varchar2(size)
	DATE_OF_BIRTH	Date/Time	date

Business Rules, Attributes, Data types and Primary/Foreign Keys

This section of the report identifies all of the attributes, data types and primary and foreign keys for our system LDM. For better overview we presents the table number to the corresponding entity type, followed by the business rule of what we wish the entity type to capture, posted below in Table 7.

Table 7: Table number entity type

Table Number	Entity Type	Business Rules
1	CUSTOMER	Records all the personal details about the customer
2	APPLICATION	Records details of the insurance cover requested by customer
3	QUOTE	Records details of customer potential cost of the insurance product
4	INSURANCE_POLICY	Records details of Insurance agreement
5	PREMIUM_PAYMENT	Records details of customer cost of payments
6	VEHICLE	Records details of Vehicle model, cost and registration
7	CLAIM	Records details of customer claims in case of an incident
8	CLAIM_SETTLEMENT	Records details of settlement made on claims
9	STAFF	Records details of employees
10	DEPARTMENT	Records details of the various departments
11	OFFICE	Records details of different office locations
12	MEMBERSHIP	Records details of customer membership, clubs, societies
13	VEHICLE_SERVICE	Records details of different vehicle services offered
14	NOK	Records details of the next of kin
15	INSURANCE_COMPANY	Details of the Insurance organization giving the insurance cover
16	POLICY_RENEWABLE	Records details of due date of insurance policy
17	INCIDENT	Records details of the accident, theft, fire, etc.
18	INCIDENT_REPORT	Records details of the individual incident
19	COVERAGE	Records all terms and conditions in regard to the policy
20	PRODUCT	Records details of the products offered by insurance company
21	RECEIPT	Details of premium payments to customer
22	INSURANCE_POLICY_COVERAGE	It shows agreement and coverage details

Table 8: Abbreviation table of attributes manes used in LDM

LDM Attributes	Column Names Abbreviations
CUST_ID	CUSTOMER_IDENTIFICATION
CUST_FNAME	CUSTOMER_FIRST_NAME
CUST_LNAME	CUSTOMER_LAST_NAME
CUST_DOB	CUSTOMER_DATEOFBIRTH
CUST_PPS_NUMBER	CUSTOMER_PERSONALPUBLICSERVICE_NUMBER
STAFF_FNAME	STAFF_FIRSTNAME
STAFF_LNAME	STAFF_LASTNAME
STAFF_PPS_NUMBER	STAFF_PERSONALPUBLICSERVICE_NUMBER
ADMIN_COST	ADMINISTRATION_COST
NOK_ID	NEXTOFKIN_IDENTIFICATION

Table-LDM 1: CUSTOMER

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
CUST_ID	TEXT	VARCHAR2(20)	PK	The CUSTOMER attributes record all the essential personal details of the customer. The CUST_ID is the unique primary key.
CUST_FNAME	TEXT	VARCHAR2(10)		
CUST_LNAME	TEXT	VARCHAR2(10)		
CUST_DOB	DATE	DATE		
CUST_GENDER	TEXT	CHAR(2)		
CUST_ADDRESS	TEXT	VARCHAR2(20)		
CUST_MOB_NUMBER	NUMBER	INTEGER		
CUST_EMAIL	TEXT	VARCHAR2(20)		
CUST_PASSPORT_NUMBER	TEXT	VARCHAR2(20)		
CUST_MARITAL_STATUS	TEXT	CHAR(8)		
CUST_PPS_NUMBER	NUMBER	INTEGER		

Table-LDM 2:APPLICATION

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
APPLICATION_ID	TEXT	VARCHAR2（20）	PK	The APPLICATION attributes record all the essential application details of the customer. The APPLICATION_ID is the unique primary key and the CUST_ID is a foreign key linking the table back to the entity type CUSTOMER.
CUST_ID	TEXT	VARCHAR2（20）	FK	
VEHICLE_ID	TEXT	VARCHAR2（20）		
APPLICATION_STATUS	TEXT	CHAR（8）		
COVERAGE	TEXT	VARCHAR2（50）		

Table-LDM 3: QUOTE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
QUOTE_ID	TEXT	VARCHAR2（20）	PK	The QUOTE attributes record all the essential quotation details of the customer. The QUOTE_ID is the unique primary key and APPLICATION_ID and CUST_ID is a foreign key linking the table back to the respective entities
APPLICATION_ID	TEXT	VARCHAR2（20）	FK	
CUST_ID	TEXT	VARCHAR2 (20)	FK	
ISSUE_DATE	DATE/TIME	DATE		
VALID_FROM_DATE	DATE/TIME	DATE		
VALID_TILL_DATE	DATE/TIME	DATE		
DESCRIPTION	TEXT	VARCHAR2（100）		
PRODUCT_ID	TEXT	VARCHAR2（20）		
COVERAGE_LEVEL	TEXT	VARCHAR2（20）		

Table-LDM 4:INSURANCE_POLICY

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
AGREEMENT_ID	TEXT	VARCHAR2（20）	PK	The INSURANCE POLICY attributes record all the essential policy details of the customer. The AGREEMENT_ID is the unique primary key and the CUST_ID, and APPLICATION_ID are linked to the other corresponding entities through their foreign keys.
APPLICATION_ID	TEXT	VARCHAR2（20）	FK	
CUST_ID	TEXT	VARCHAR2（20）	FK	
DEPARTMENT _NAME	TEXT	VARCHAR2（20）		
POLICY_NUMBER	TEXT	VARCHAR2（20）		
START_DATE	DATE/TIME	DATE		
EXPIRY_DATE	DATE/TIME	DATE		
TERM_CONDITION_DESCRIPTION	TEXT	VARCHAR2（100）		

Table-LDM 5:PREMIUM_PAYMENT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
PREMIUM_PAYMENT_ID	TEXT	VARCHAR2(20)	PK	The PREMIUM_PAYMENT attributes record all the essential policy premium payments details of the customer. The PREM_PAYMENT_ID is the unique primary key and the CUST_ID is the Foreign key linking table to CUSTOMER entity.
CUST_ID	TEXT	VARCHAR2(20)	FK	
POLICY_NUMBER	TEXT	VARCHAR2(20)		
PREMIUM_PAYMENT_SCHEDULE	DATE/TIME	DATE		
PREMIUM_PAYMENT_AMOUNT	NUMBER	INTEGER		
RECEIPT_ID	TEXT	VARCHAR2(20)		

Table-LDM 6: VEHICLE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
VEHICLE_ID	TEXT	VARCHAR2(20)	PK	The VEHICLE attributes record all the essential VEHICLE details belonging to the customer. The VEHICLE_ID is the unique primary key and the CUST_ID is the foreign key linking table to CUSTOMER entity.
CUST_ID	TEXT	VARCHAR2(20)	FK	
POLICY_ID	TEXT	VARCHAR2 (20)		
DEPENDENT_NOK_ID	TEXT	VARCHAR2(20)		
VEHICLE_REGISTRATION_NUMBER	TEXT	VARCHAR2(20)		
VEHICLE_VALUE	NUMBER	INTEGER		
VEHICLE_TYPE	TEXT	VARCHAR2(20)		
VEHICLE_SIZE	NUMBER	INTEGER		
VEHICLE_NUMBER_OF_SEAT	NUMBER	INTEGER		
VEHICLE_MANUFACTURER	TEXT	VARCHAR2(20)		
VEHICLE_ENGINE_NUMBER	NUMBER	INTEGER		
VEHICLE_CHASIS_NUMBER	NUMBER	INTEGER		
VEHICLE_NUMBER	TEXT	VARCHAR2(20)		
VEHICLE_MODEL_NUMBER	TEXT	VARCHAR2(20)		

Table-LDM 7: CLAIM

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
CLAIM_ID	TEXT	VARCHAR2(20)	PK	The CLAIM attributes record all the essential CLAIM details of the customer in case of an incident. The CLAIM_ID is the unique primary key and the CUST_ID is foreign key linking table to CUSTOMER entity.
CUST_ID	TEXT	VARCHAR2(20)	FK	
AGREEMENT_ID	TEXT	VARCHAR2(20)		
CLAIM_AMOUNT	NUMBER	INTEGER		
INCIDENT_ID	TEXT	VARCHAR2(20)		
DAMAGE_TYPE	TEXT	VARCHAR2(20)		
DATE_OF_CLAIM	DATE/TIME	DATE		
CLAIM_STATUS	TEXT	CHAR(10)		

Table-LDM 8: CLAIM_SETTLEMENT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
CLAIM_SETTLEMENT_ID	TEXT	VARCHAR2(20)	PK	The CLAIM_SETTLEMENT attributes record all the essential claim settlement details of the customer after an incident. The CLAIM_SETTLEMENT_ID is the unique primary key and the CUST_ID and CLAIM_ID are the foreign keys that link the table to the corresponding entity.
CLAIM_ID	TEXT	VARCHAR2(20)	FK	
CUST_ID	TEXT	VARCHAR2(20)	FK	
VEHICLE_ID	TEXT	VARCHAR2(20)		
DATE_SETTLED	DATE/TIME	DATE		
AMOUNT_PAID	NUMBER	INTEGER		
COVERAGE_ID	TEXT	VARCHAR2(20)		

Table-LDM 9: STAFF

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
STAFF_ID	TEXT	VARCHAR2(20)	PK	The STAFF attributes record all the essential staff details working in the insurance company. The STAFF_ID is the unique primary key and the COMPANY_NAME is a foreign key linking the table back to the entity type COMPANY.
COMPANY_NAME	TEXT	VARCHAR2(20)	FK	
STAFF_FNAME	TEXT	VARCHAR2(10)		
STAFF_LNAME	TEXT	VARCHAR2(10)		
STAFF_ADDRESS	TEXT	VARCHAR2(20)		
STAFF_CONTACT	NUMBER	INTEGER		
STAFF_GENDER	TEXT	CHAR(2)		
STAFF_MARITAL_STATUS	TEXT	CHAR(8)		
STAFF_NATIONALITY	TEXT	CHAR(15)		
STAFF_QUALIFICATION	TEXT	VARCHAR2(20)		
STAFF_ALLOWANCE	NUMBER	INTEGER		
STAFF_PPS_NUMBER	NUMBER	INTEGER		

Table-LDM 10: DEPARTMENT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
DEPARTMENT_NAME	TEXT	VARCHAR2(20)	PK	The DEPARTMENT attributes record all the essential company department details within the insurance company. The DEPARTMENT_NAME is the unique primary key and the COMPANY_NAME is a foreign key linking the table back to the entity type COMPANY.
COMPANY_NAME	TEXT	VARCHAR2(20)	FK	
OFFICE	TEXT	VARCHAR2(20)		
CONTACT_INFORMATION	TEXT	VARCHAR2(20)		
DEPARTMENT_STAFF	TEXT	VARCHAR2(50)		
DEPARTMENT_LEADER	TEXT	VARCHAR2(20)		

Table-LDM 11:OFFICE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
OFFICE_NAME	TEXT	VARCHAR2(20)	PK	The OFFICE attributes record all the essential office details within the insurance company. The OFFICE_NAME is the unique primary key and the DEPARTMENT_NAME and COMPANY_NAME are foreign keys linking the table back to the respective entity types.
DEPARTMENT_NAME	TEXT	VARCHAR2(20)	FK	
COMPANY_NAME	TEXT	VARCHAR2(20)	FK	
OFFICE_LEADER	TEXT	VARCHAR2(20)		
CONTACT_INFORMATION	TEXT	VARCHAR2(20)		
ADDRESS	TEXT	VARCHAR2(20)		
ADMIN_COST	NUMBER	INTEGER		
STAFF	TEXT	VARCHAR2(50)		

Table-LDM 12: MEMBERSHIP

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
MEMBERSHIP_ID	TEXT	VARCHAR2(20)	PK	The MEMBERSHIP attributes record all the essential membership details available for insured customer. The MEMBERSHIP_ID is the unique primary key and the CUST_ID is a foreign key linking the table back to the entity type CUSTOMER.
CUST_ID	TEXT	VARCHAR2(20)	FK	
MEMBERSHIP_TYPE	TEXT	CHAR(15)		
ORGANISATION_CONTACT	TEXT	VARCHAR2(20)		

Table-LDM 13: VEHICLE_SERVICE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
VEHICLE_SERVICE	TEXT	VARCHAR2(20)	PK	The VEHICLE_SERVICE attributes record all the essential vehicle services offered to insured customer details. The VEHICLE_SERVICE is the unique primary key and the CUST_ID and VEHICLE_ID are linked to the other corresponding entities.
VEHICLE_ID	TEXT	VARCHAR(20)	FK	
CUST_ID	TEXT	VARCHAR(20)	FK	
DEPARTMENT_NAME	TEXT	CHAR(20)		
VEHICLE_SERVICE_ADDRESS	TEXT	VARCHAR2(20)		
VEHICLE_SERVICE_CONTACT	TEXT	VARCHAR2(20)		
VEHICLE_SERVICE_INCHARGE	TEXT	CHAR(20)		
VEHICLE_SERVICE_TYPE	TEXT	VARCHAR2(20)		

Table-LDM 14:NOK

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
NOK_ID	TEXT	VARCHAR2(20)	PK	The NOK attributes record information on the next of kin details. NOK_ID is the unique primary key here. AGREEMENT_ID, APPLICATION_ID, and CUST_ID are foreign keys linking back information to their respective entities.
AGREEMENT_ID	TEXT	VARCHAR2(20)	FK	
APPLICATION_ID	TEXT	VARCHAR2(20)	FK	
CUST_ID	TEXT	VARCHAR2(20)	FK	
NOK_NAME	TEXT	VARCHAR2(20)		
NOK_ADDRESS	TEXT	VARCHAR2(20)		
NOK_PHONE_NUMBER	NUMBER	INTEGER		
NOK_MARITAL_STATUS	TEXT	CHAR(8)		
NOK_GENDER	TEXT	CHAR(2)		

Table-LDM 15: INSURANCE_COMPANY

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
COMPANY_ NAME	TEXT	VARCHAR2(20)	PK	The INSURANCE COMPANY attributes record all the essential company details of the customer. The COMPANY_ID is the unique primary key
COMPANY_ADDRESS	TEXT	VARCHAR2(20)		
COMPANY_CONTACT_NUMBER	NUMBER	INTEGER		
COMPANY_FAX	NUMBER	INTEGER		
COMPANY_ EMAIL	TEXT	VARCHAR2(20)		
COMPANY _WEBSITE	TEXT	VARCHAR2(20)		
COMPANY _LOCATION	TEXT	VARCHAR2(20)		
COMPANY _DEPARTMENT_NAME	TEXT	VARCHAR2(20)		
COMPAN_ OFFICE_NAME	TEXT	VARCHAR2(20)		

Table-LDM 16:POLICY_RENEWABLE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
POLICY_RENEWABLE_ID	TEXT	VARCHAR2(20)	PK	The POLICY RENEWABLE attributes record all the essential policy renewal details of the insured customer. The POLICY_RENEWABLE_ID is the unique primary key and the AGREEMENT_ID, APPLICATION_ID and CUST_ID are foreign keys linking the table back to the respective entities.
AGREEMENT_ID	TEXT	VARCHAR2(20)	FK	
APPLICATION_ID	TEXT	VARCHAR2(20)	FK	
CUST_ID	TEXT	VARCHAR2(20)	FK	
DATE_OF_RENEWAL	DATE	DATE		
TYPE_OF_RENEWAL	TEXT	CHAR(15)		

Table-LDM 17: INCIDENT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
INCIDENT_ID	TEXT	VARCHAR2(20)	PK	The INCIDENT attributes record all the essential incident details such as Accident and theft on the insured customer vehicle. The INCIDENT_ID is the unique primary key.
INCIDENT_TYPE	TEXT	VARCHAR2(30)		
INCIDENT_DATE	DATE	DATE		
DESCRIPTION	TEXT	VARCHAR2(100)		

Table-LDM 18: INCIDENT_REPORT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
INCIDENT_REPORT_ID	TEXT	VARCHAR2(20)	PK	The INCIDENT_REPORT_ID attributes record all the essential incident occurrences on the customer vehicle. The INCIDENT_REPORT_ID is the unique primary key and the CUST_ID, AND INCIDENT_ID are foreign keys linking the table back to their respective entity types.
INCIDENT_ID	TEXT	VARCHAR2(20)	FK	
CUST_ID	TEXT	VARCHAR2(20)	FK	
INCIDENT_INSPECTOR	TEXT	VARCHAR2(20)		
INCIDENT_COST	NUMBER	INTEGER		
INCIDENT_TYPE	TEXT	CHAR(10)		
INCIDENT_REPORT_DESCRIPTION	TEXT	VARCHAR2(100)		

Table-LDM 19: COVERAGE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
COVERAGE_ID	TEXT	VARCHAR2(20)	PK	The COVERAGE attributes record all the essential coverage details of the insurance policy to the customer. The COVERAGE_ID is the unique primary key and the COMPANY_NAME is a foreign key linking the table back to the entity type COMPANY.
COMPANY_NAME	TEXT	VARCHAR220)	FK	
COVERAGE_AMOUNT	NUMBER	INTEGER		
COVERAGE_TYPE	TEXT	CHAR(10)		
COVERAGE_LEVEL	TEXT	CHAR(15)		
PRODUCT_ID	TEXT	VARCHAR2(20)		
COVERAGE_DESCRIPTION	TEXT	VARCHAR2(100)		
COVERAGE_TERMS	TEXT	VARCHAR2(50)		

Table-LDM 20: PRODUCT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
PRODUCT_NUMBER	TEXT	VARCHAR2(20)	PK	The PRODUCT attributes record all the essential company products details offered by the Insurance company. The PRODUCT_NUMBER is the unique primary key and COMPANY_NAME is the foreign key linking table to COMPANY entity.
COMPANY_NAME	TEXT	VARCHAR2(20)	FK	
PRODUCT_PRICE	NUMBER	INTEGER		
PRODUCT_TYPE	TEXT	CHAR(15)		

Table-LDM 21: RECEIPT

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
RECEIPT_ID	TEXT	VARCHAR2(20)	PK	The RECEIPT attributes record all the essential payments done by CUSTOMERS to Insurance company. The RECEIPT_ID is the unique primary key and PREMIUM_PAYMENT_ID and CUST_ID are Foreign keys linking table to their respective entities.
PREMIUM_PAYMENT_ID	TEXT	VARCHAR2(20)	FK	
CUST_ID	TEXT	VARCHAR2(20)	FK	
COST	NUMBER	INTEGER		
TIME	DATE	DATE		

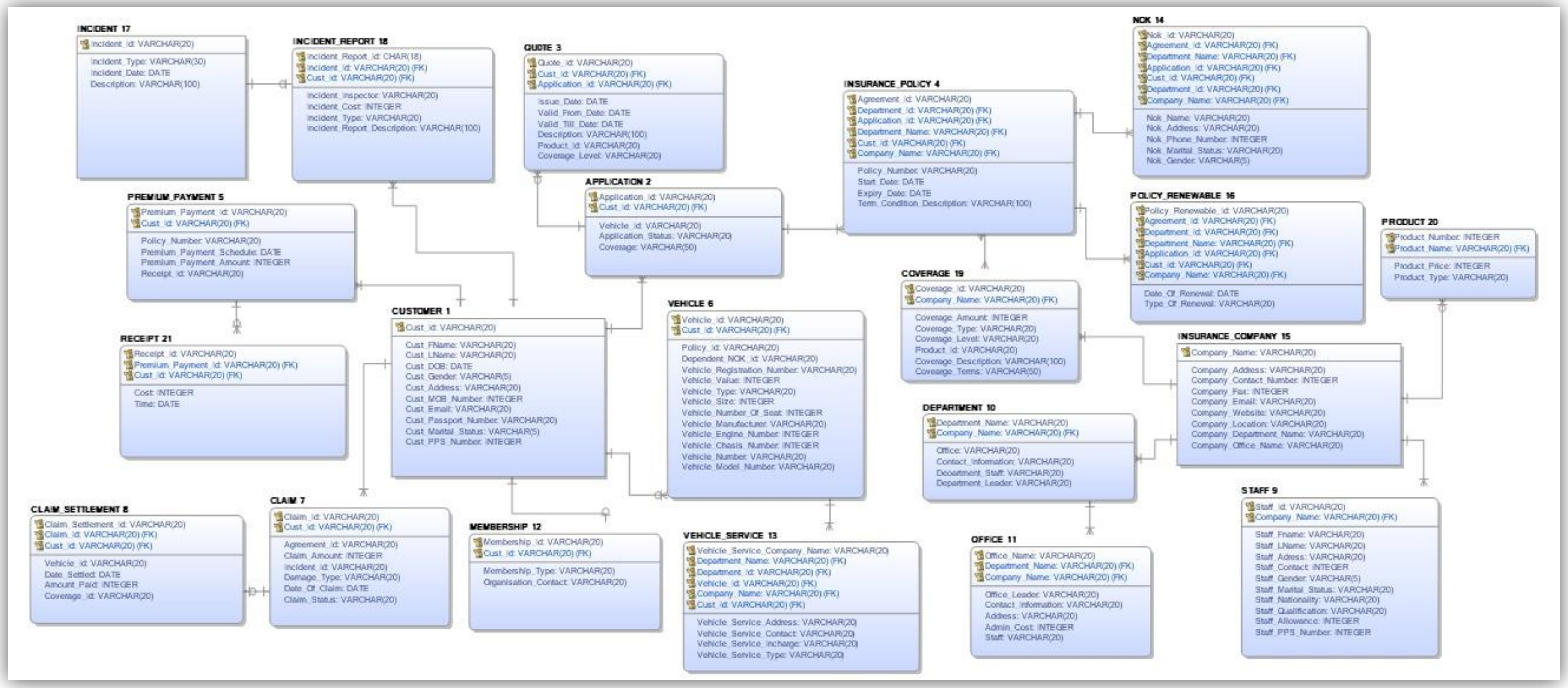
Table-LDM 22: INSURANCE_POLICY_COVERAGE

Attributes	Data Type Access	Data Type Oracle	Primary and Foreign Keys	Explanation
AGREEMENT_ID	TEXT	VARCHAR2(20)	PK	The INSURANCE_POLICY_COVERAGE records details of the Vehicle policy that entails Terms Conditions of the Contract. AGREEMENT_ID is the unique primary key and COVERAGE_ID is the Foreign key linking this table to COVERAGE Entity.
COVERAGE_ID	TEXT	VARCHAR2(20)	FK	

Graphical presentation of LDM

The Logical Data Model (LDM) that we have designed for this part of report in graphical Figure-LDM 1. It has all the entity types, attributes and relationships that are valid and pertinent in designing our online vehicle insurance database system

Figure-LDM 1: Logical data model of Car insurance for AVIVA Ltd.


INCIDENT 17
Incident_id: VARCHAR(20)
Incident_Type: VARCHAR(30)
Incident_Date: DATE
Description: VARCHAR(100)
INCIDENT_REPORT 18
Incident_Report_id: CHAR(18)
Incident_id: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Incident_Inspector: VARCHAR(20)
Incident_Cost: INTEGER
Incident_Type: VARCHAR(20)
Incident_Report_Description: VARCHAR(100)
QUOTE 3
Quote_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Application_id: VARCHAR(20) (FK)
Issue_Date: DATE
Valid_From_Date: DATE
Valid_Till_Date: DATE
Description: VARCHAR(100)
Product_id: VARCHAR(20)
Coverage_Level: VARCHAR(20)
INSURANCE_POLICY 4
Agreement_id: VARCHAR(20)
Department_id: VARCHAR(20) (FK)
Application_id: VARCHAR(20) (FK)
Department_Name: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Company_Name: VARCHAR(20) (FK)
Policy_Number: VARCHAR(20)
Start_Date: DATE
Expiry_Date: DATE
Term_Condition_Description: VARCHAR(100)
NOK 14
Nok_id: VARCHAR(20)
Agreement_id: VARCHAR(20) (FK)
Department_Name: VARCHAR(20) (FK)
Application_id: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Department_id: VARCHAR(20) (FK)
Company_Name: VARCHAR(20) (FK)
Nok_Name: VARCHAR(20)
Nok_Address: VARCHAR(20)
Nok_Phone_Number: INTEGER
Nok_Marital_Status: VARCHAR(20)
Nok_Gender: VARCHAR(5)
PREMIUM_PAYMENT 5
Premium_Payment_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Policy_Number: VARCHAR(20)
Premium_Payment_Schedule: DATE
Premium_Payment_Amount: INTEGER
Receipt_id: VARCHAR(20)
APPLICATION 2
Application_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Vehicle_id: VARCHAR(20)
Application_Status: VARCHAR(20)
Coverage: VARCHAR(50)
POLICY_RENEWABLE 16
Policy_Renewable_id: VARCHAR(20)
Agreement_id: VARCHAR(20) (FK)
Department_id: VARCHAR(20) (FK)
Department_Name: VARCHAR(20) (FK)
Application_id: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Company_Name: VARCHAR(20) (FK)
Date_Of_Renewal: DATE
Type_Of_Renewal: VARCHAR(20)
PRODUCT 20
Product_Number: INTEGER
Product_Name: VARCHAR(20) (FK)
Product_Price: INTEGER
Product_Type: VARCHAR(20)
COVERAGE 19
Coverage_id: VARCHAR(20)
Company_Name: VARCHAR(20) (FK)
Coverage_Amount: INTEGER
Coverage_Type: VARCHAR(20)
Coverage_Level: VARCHAR(20)
Product_id: VARCHAR(20)
Coverage_Description: VARCHAR(100)
Coveage_Terms: VARCHAR(50)
RECEIPT 21
Receipt_id: VARCHAR(20)
Premium_Payment_id: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Cost: INTEGER
Time: DATE
CUSTOMER 1
Cust_id: VARCHAR(20)
Cust_FName: VARCHAR(20)
Cust_LName: VARCHAR(20)
Cust_DOB: DATE
Cust_Gender: VARCHAR(5)
Cust_Address: VARCHAR(20)
Cust_MOB_Number: INTEGER
Cust_Email: VARCHAR(20)
Cust_Passport_Number: VARCHAR(20)
Cust_Marital_Status: VARCHAR(5)
Cust_PPS_Number: INTEGER
VEHICLE 6
Vehicle_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Policy_id: VARCHAR(20)
Dependent_NOK_id: VARCHAR(20)
Vehicle_Registration_Number: VARCHAR(20)
Vehicle_Value: INTEGER
Vehicle_Type: VARCHAR(20)
Vehicle_Size: INTEGER
Vehicle_Number_Of_Seat: INTEGER
Vehicle_Manufacturer: VARCHAR(20)
Vehicle_Engine_Number: INTEGER
Vehicle_Chasis_Number: INTEGER
Vehicle_Number: VARCHAR(20)
Vehicle_Model_Number: VARCHAR(20)
INSURANCE_COMPANY 15
Company_Name: VARCHAR(20)
Company_Address: VARCHAR(20)
Company_Contact_Number: INTEGER
Company_Fax: INTEGER
Company_Email: VARCHAR(20)
Company_Website: VARCHAR(20)
Company_Location: VARCHAR(20)
Company_Department_Name: VARCHAR(20)
Company_Office_Name: VARCHAR(20)
DEPARTMENT 10
Department_Name: VARCHAR(20)
Company_Name: VARCHAR(20) (FK)
Office: VARCHAR(20)
Contact_Information: VARCHAR(20)
Deoartment_Staff: VARCHAR(20)
Department_Leader: VARCHAR(20)
CLAIM_SETTLEMENT 8
Claim_Settlement_id: VARCHAR(20)
Claim_id: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Vehicle_id: VARCHAR(20)
Date_Settled: DATE
Amount_Paid: INTEGER
Coverage_id: VARCHAR(20)
CLAIM 7
Claim_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Agreement_id: VARCHAR(20)
Claim_Amount: INTEGER
Incident_id: VARCHAR(20)
Damage_Type: VARCHAR(20)
Date_Of_Claim: DATE
Claim_Status: VARCHAR(20)
MEMBERSHIP 12
Membership_id: VARCHAR(20)
Cust_id: VARCHAR(20) (FK)
Membership_Type: VARCHAR(20)
Organisation_Contact: VARCHAR(20)
VEHICLE_SERVICE 13
Vehicle_Service_Company_Name: VARCHAR(20)
Department_Name: VARCHAR(20) (FK)
Department_id: VARCHAR(20) (FK)
Vehicle_id: VARCHAR(20) (FK)
Company_Name: VARCHAR(20) (FK)
Cust_id: VARCHAR(20) (FK)
Vehicle_Service_Address: VARCHAR(20)
Vehicle_Service_Contact: VARCHAR(20)
Vehicle_Service_Incharge: VARCHAR(20)
Vehicle_Service_Type: VARCHAR(20)
OFFICE 11
Office_Name: VARCHAR(20)
Department_Name: VARCHAR(20) (FK)
Company_Name: VARCHAR(20) (FK)
Office_Leader: VARCHAR(20)
Contact_Information: VARCHAR(20)
Address: VARCHAR(20)
Admin_Cost: INTEGER
Staff: VARCHAR(20)
STAFF 9
Staff_id: VARCHAR(20)
Company_Name: VARCHAR(20) (FK)
Staff_Fname: VARCHAR(20)
Staff_LName: VARCHAR(20)
Staff_Adress: VARCHAR(20)
Staff_Contact: INTEGER
Staff_Gender: VARCHAR(5)
Staff_Marital_Status: VARCHAR(20)
Staff_Nationality: VARCHAR(20)
Staff_Qualification: VARCHAR(20)
Staff_Allowance: INTEGER
Staff_PPS_Number: INTEGER

Modifications to CDM

We have done some changes and modification to Part A: Conceptual data model seen in Figure-LDM 2 and Figure-LDM 3, with the description:

New entities added to the revised CDM Model

1. INCIDENT
2. INCIDENT_REPORT
3. POLICY_RENEWABLE
4. PRODUCT

Changes made in previous CDM entities

1. PREMIUM **to** PREMIUM_PAYMENT
2. TERMS & CONDITION **to** COVERAGE
3. SETTLEMENT **to** CLAIM_SETTLEMENT
4. SERVICE **to** VEHICLE_SERVICE
5. COMPANY **to** INSURANCE_COMPANY

Figure-LDM 2: Previous CDM

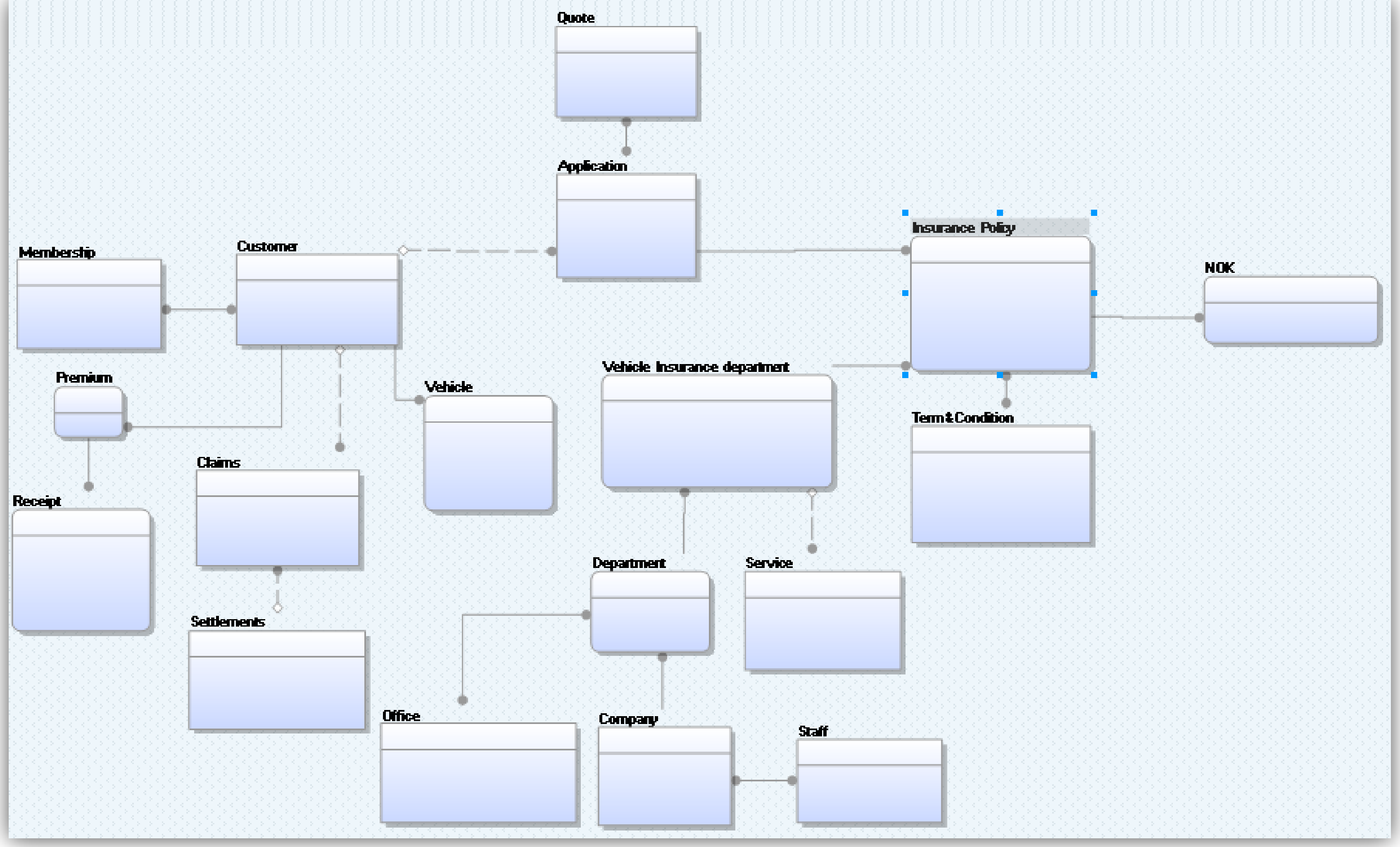
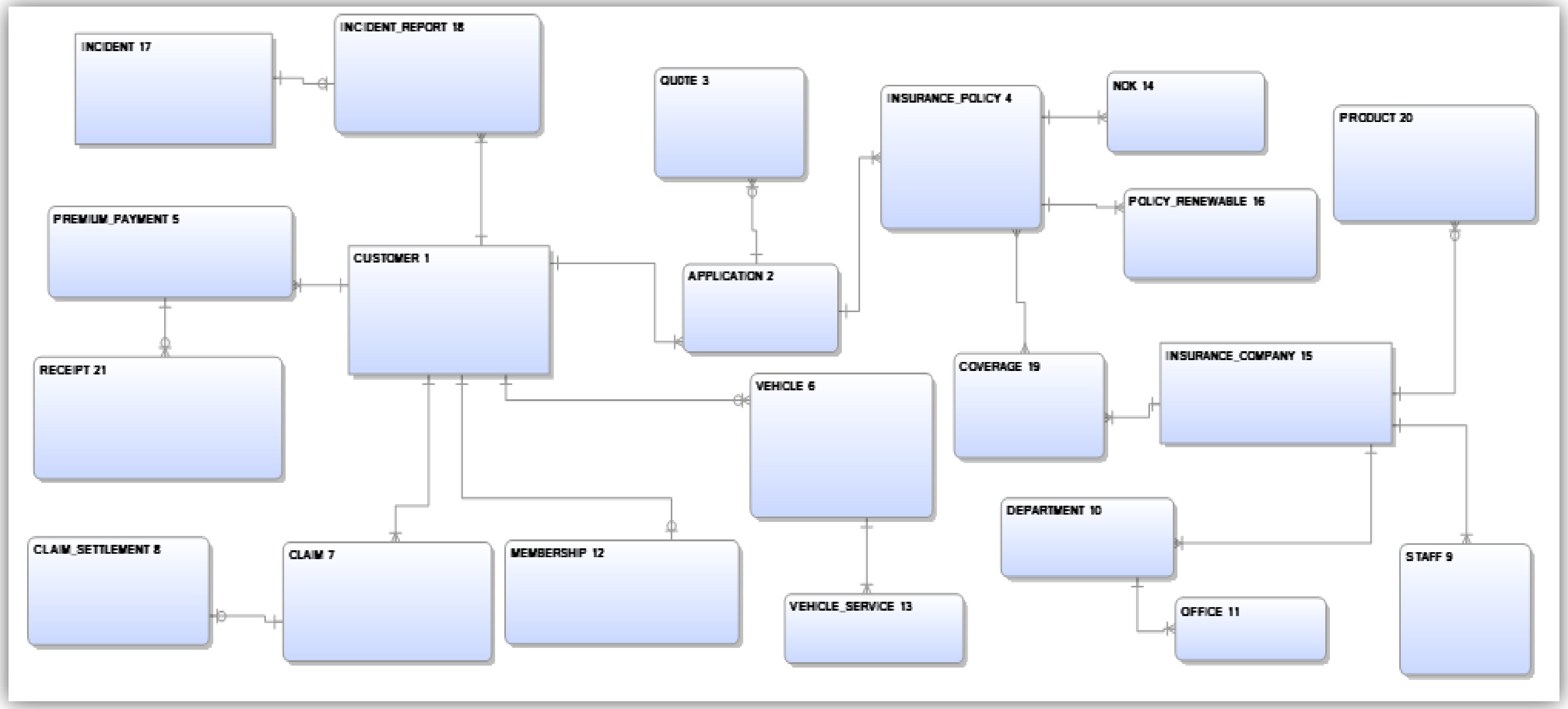


Figure-LDM 3: Revised CDM


INCIDENT 17
INCIDENT_REPORT 18
QUOTE 3
INSURANCE_POLICY 4
NOK 14
PRODUCT 20
POLICY_RENEWABLE 16
PREMIUM_PAYMENT 5
CUSTOMER 1
APPLICATION 2
RECEIPT 21
COVERAGE 19
INSURANCE_COMPANY 15
VEHICLE 6
DEPARTMENT 10
CLAIM_SETTLEMENT 8
CLAIM 7
MEMBERSHIP 12
VEHICLE_SERVICE 13
OFFICE 11
STAFF 9

Part C: Physical Data Model and Database Design (25%)

Introduction

When building an entity relationship (ER) model we tend to use it to later build different physical models of database types, see Figure 5. Therefore physical data model is used to implement into different technical software and hardware environments that is due to current state of technology and is changing as technologies change.

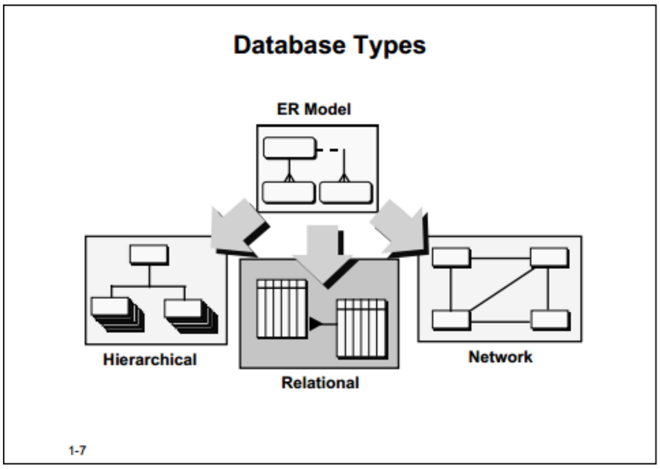
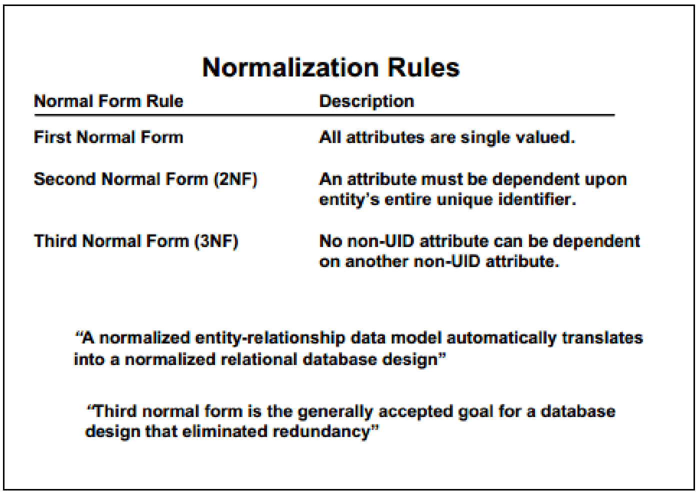


Figure 5: Database types (Source: Data Modelling and Relational Database Design (Speelpenning , et al., 2001))

NORMALIZATION

It is a relationship database concept and is done in process of building ER. If the correct entity model is being build 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, see Figure 6 (Speelpenning , et al., 2001).



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"

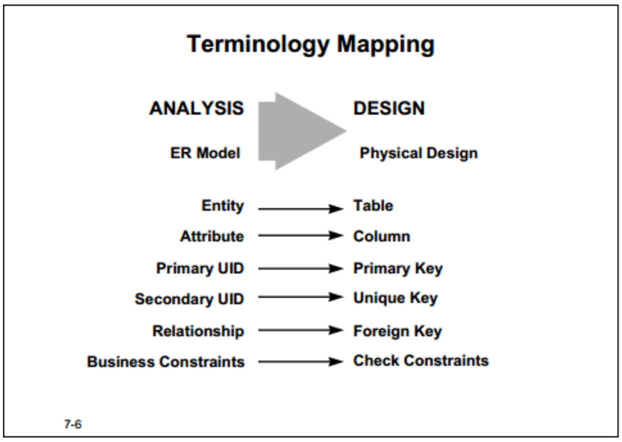
Figure 6: Normalization rules (Source: Data Modelling and Relational Database Design (Speelpenning , et al., 2001))

Table 9: Normalization of vehicle insurance company

Normal form	Table
First normal form （1NF）	
Second normal form (2NF)	DEPARTMENT OFFICE VEHICLE NOK INSURANCE_POLICY CLAIM CLAIM_SETTLEMENT PREMIUM_PAYMENT QUOTE INCIDENT_REPORT POLICY_RENEWABLE
Third normal form (3NF)	CUSTOMER RECEIPT APPLICATION STAFF INSURANCE_COMPANY MEMBERSHIP PRODUCT COVERAGE VEHICLE_SERVICE INCIDENT

TERMINOLOGY

The relational data model consists of mathematical rules, that later translate its syntax to physical model, but there are not always correspondent to syntax of physical model. Therefore you have to keep track of them and find how to implement them in correct way, see Figure 7. This change also means change of terminology (Speelpenning , et al., 2001).



Terminology Mapping

ANALYSIS	DESIGN
ER Model	Physical Design
Entity	Table
Attribute	Column
Primary UID	Primary Key
Secondary UID	Unique Key
Relationship	Foreign Key
Business Constraints	Check Constraints

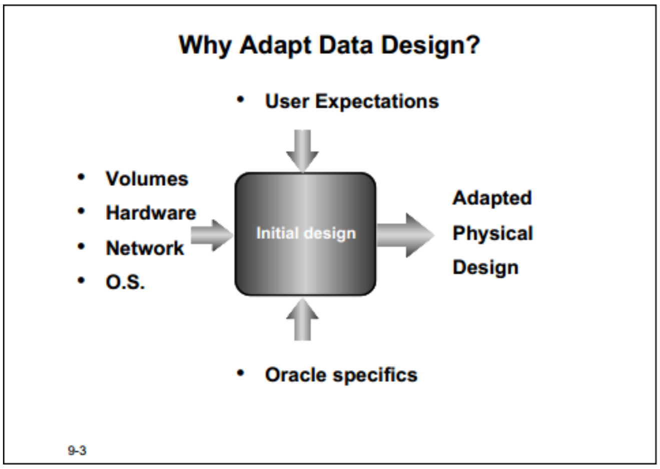
7-6

Figure 7: Terminology mapping – ER Model to Physical design (Source: Data Modelling and Relational Database Design (Speelpenning , et al., 2001))

RDBMS

As we start doing to build relational data management system (RDBMS) we need a large number of parameters to obtain a correct adapted physical model. Example to get best performance in Oracle RDBMS, see Figure 8. You must be aware that there is no absolute truth here. Some of most important points of creating physical models for RDBMS (Speelpenning , et al., 2001):

- Expected volume of tables, the hardware characteristics (CPU speed, memory size, number of disks and corresponding space), the architecture-client/server or three size, the network bandwidth, speed and operating systems are important determinants.
- User experience second big issue (response time, the GUI and frequency of use of modules).
- Depending which version of oracle you use as some elements may or may not exist.



Why Adapt Data Design?

- User Expectations

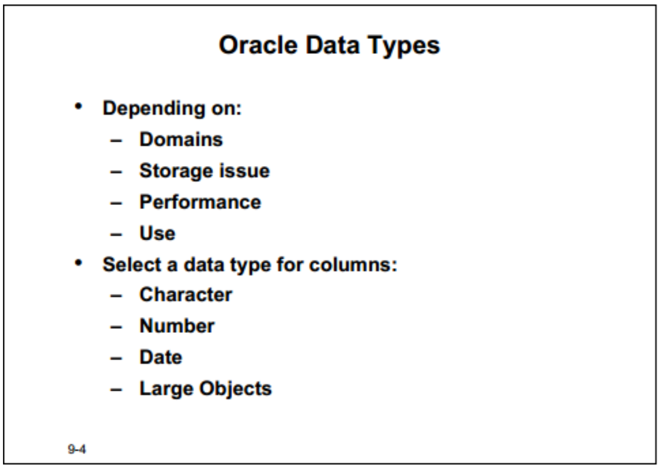
- Volumes
- Hardware
- Network
- O.S.

- Oracle specifics

Figure 8: Adapting data design in Oracle RDBMS (Source: Data Modelling and Relational Database Design (Speelpenning , et al., 2001))

PHYSICAL MODEL

When creating physical models we create tables or clusters and we must write specifications of internal data type for each its columns. These types define generic domain of values that each column can contain, see Figure 9.



Oracle Data Types

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

9-4

Figure 9: Oracle data types (Source: Data Modelling and Relational Database Design (Speelpenning , et al., 2001))

These are some concerns using data types in Oracle RDBMS building a physical model (Speelpenning , et al., 2001).

- Data types can have a narrow focus (number, date).
- Some are general purpose data types (various character data types).
- Data types can allow for variable length or not. Choosing a large fixed length for a column to only store a few bytes per row, makes a large table. This may affect performance specially if stored on multiple blocks, resulting in great number of I/O's and so affecting performance.
- Large data object types are not advised to be used in where clause as they are only retrievable against other columns.

For our assignment we created data types for **Oracle** and **MS Access** RDBMS.

Most Commonly-Used Oracle Data Types

Here you can see most used Oracle data types (Speelpenning , et al., 2001):

• **CHAR (size**) these are fixed-length character data of length-sized bytes. Maximum size is 2000 bytes.

Typical use: for official International Currency Codes which are a fixed three characters in length such as USD, FFR.

• **VARCHAR2 (size)** Variable-length character string having maximum length-sized bytes. Maximum size is 4000, and minimum is 1. This is the most commonly-used data type and you should use it if you are not sure which one to use. It replaces the old Oracle version 6 CHAR data type.

Typical use: for storing individual ASCII text lines of unlimited length ASCII texts on which you need to be able to search using a wildcard.

• **NUMBER** This data type is used for numerical values, with or without a decimal, of virtually unlimited size. This data type is used for data on which calculation or sorting should be possible. Avoid its use for numbers like a phone number, where the value does not have any meaning.

Typical use: amount of money, quantities, generated unique key values.

• **DATE** Valid date range from January 1, 4712 BC to December 31, 4712 AD. A date data type also contains time components. You should use it only when you know the full date including day, month, and year. The time component is often set to 00:00 (midnight) in normal use of dates.

Typical use: any date where the full date is known.

• **LONG** Character data of variable length up to 2 gigabytes. Obsolete since Oracle8. Was used for ASCII text files where you do not need to search using the wildcard or substring functionality. Use CLOB data type instead.

Typical use: for storing the source code of HTML pages.

- **LONG RAW** Raw binary data of variable length up to 2 gigabytes. Obsolete since Oracle8. Was used for large object types where the database should not try to interpret the data. Use BLOB data type instead.

Typical use: images or video clips.

- **CLOB** Character large object type. Replaces LONG. Major difference: a table can have more than one CLOB column where there was only one LONG allowed. Maximum size is 4 gigabytes.

Typical use: see LONG.

- **BLOB** Character large object type replaces LONG RAW. Major difference: a table can have more than one BLOB column where there was only one LONGRAW allowed. Maximum size is 4 gigabytes.

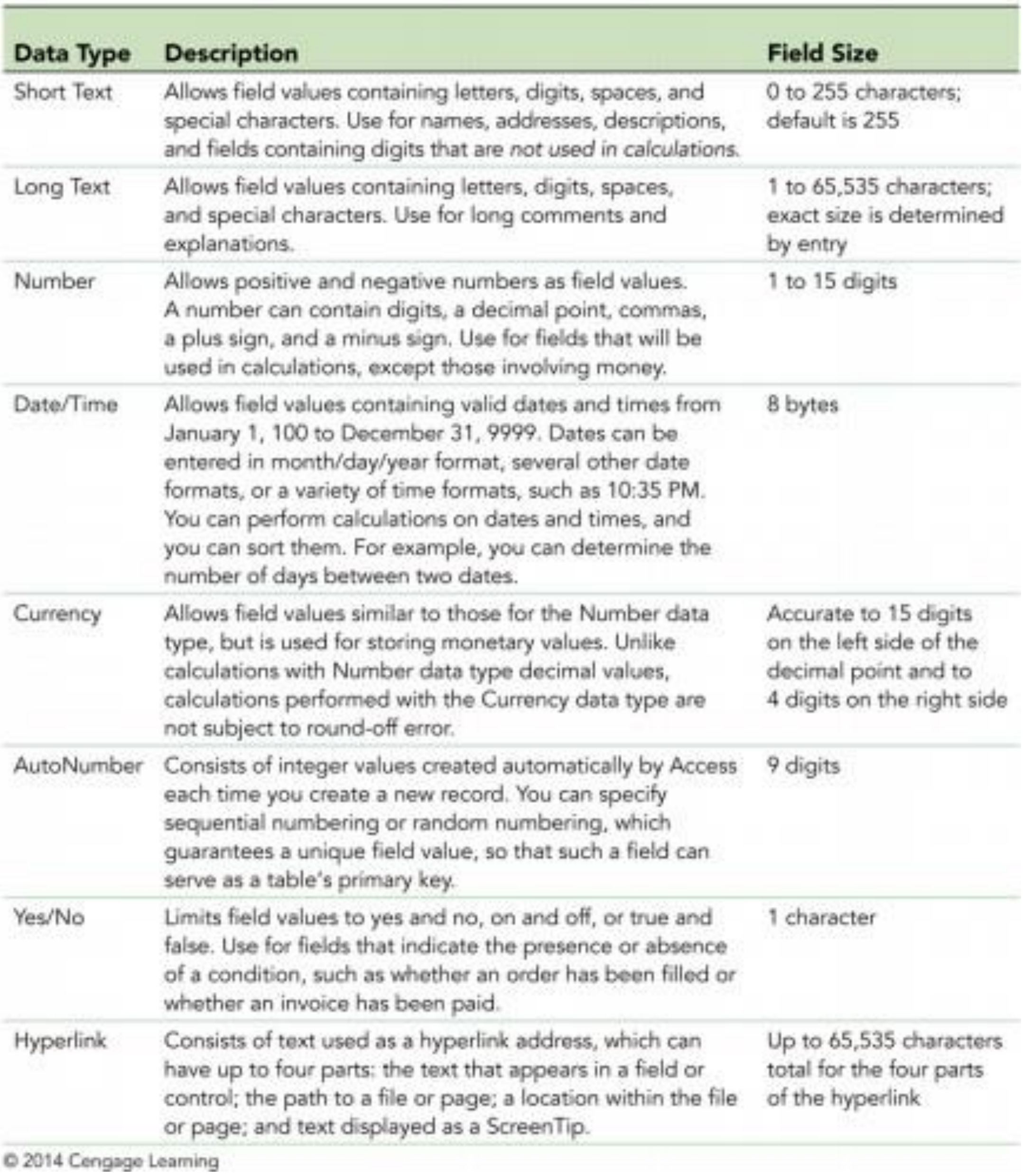
Typical use: see LONG RAW.

- **BFILE** Contains a locator to a large binary file stored outside the database to enable byte stream I/O access to external LOBs residing on the database server.

Typical use: movies

Most Commonly-Used MS Access Data Types

In Figure 10 you see also most used MS Access data types (Zickos, 2014).



Data Type	Description	Field Size
Short Text	Allows field values containing letters, digits, spaces, and special characters. Use for names, addresses, descriptions, and fields containing digits that are *not used in calculations*.	0 to 255 characters; default is 255
Long Text	Allows field values containing letters, digits, spaces, and special characters. Use for long comments and explanations.	1 to 65,535 characters; exact size is determined by entry
Number	Allows positive and negative numbers as field values. A number can contain digits, a decimal point, commas, a plus sign, and a minus sign. Use for fields that will be used in calculations, except those involving money.	1 to 15 digits
Date/Time	Allows field values containing valid dates and times from January 1, 100 to December 31, 9999. Dates can be entered in month/day/year format, several other date formats, or a variety of time formats, such as 10:35 PM. You can perform calculations on dates and times, and you can sort them. For example, you can determine the number of days between two dates.	8 bytes
Currency	Allows field values similar to those for the Number data type, but is used for storing monetary values. Unlike calculations with Number data type decimal values, calculations performed with the Currency data type are not subject to round-off error.	Accurate to 15 digits on the left side of the decimal point and to 4 digits on the right side
AutoNumber	Consists of integer values created automatically by Access each time you create a new record. You can specify sequential numbering or random numbering, which guarantees a unique field value, so that such a field can serve as a table's primary key.	9 digits
Yes/No	Limits field values to yes and no, on and off, or true and false. Use for fields that indicate the presence or absence of a condition, such as whether an order has been filled or whether an invoice has been paid.	1 character
Hyperlink	Consists of text used as a hyperlink address, which can have up to four parts: the text that appears in a field or control; the path to a file or page; a location within the file or page; and text displayed as a ScreenTip.	Up to 65,535 characters total for the four parts of the hyperlink

© 2014 Cengage Learning

Figure 10: Field properties – data types in MS Access (Source: Access Tutorial 2 (Zickos, 2014))

DATABASE SECURITY

Database security is described by the following aspects ：

Data independence

Data independence is an important method ensures data security; it can be divided into logical independence and physical independence. Physical independence means applications and data are store independent of each other data is managed by DBMS and application not need understand it, application just need processing the data when the physical storage of data changing application without changing.

Data Security

1. Isolation protect important files in database
2. Using authorization rules, such as access control method and accounts, passwords permissions control.

   In oracle there are three different system privileges:

   DBA: have all the privileges, it is the highest system privileges and only the DBA can create the database structure.

   RESOURCE: Users with resource privilege can create an entity in database, but it cannot to create and change database structure

   CONNECT: Connect privilege is the least privilege of database, in oracle 10G connect privilege only can login database and create session.

3. Data Encryption

Data Integrity

Data integrity includes:

1. Data validation: ensure clean, correct and useful data.
2. Data consistency: Different users are using the same data should be identical
3. Data correctness: The input value of the data should be consistent with data in database

Concurrency Control

Concurrency Control means the database is a shared resource for multi-use， When multiple user concurrent access to data, in the database will have multiple transactions simultaneously access the same data. If not controlled for concurrent operation may be cause incorrect to read and store data, destroy the consistency of the database. Locking mechanism of the database can effectively protect the database achieve concurrency control.

Oracle database provides three different types of locks:

1. DML lock: DML locks used to protect data integrity; DML locks mainly include TM lock and TX lock. TM lock called table lock, TX called transaction locks or row locks.
2. DDL lock: DDL lock protects the structure of the database objects. In the DDL operation is automatically added DDL lock to the object, to protect these objects will not be modified by other sessions.
3. Internal locks and Latches: Protect the internal structure of the database.

Recovery

When the database fails DBMS need to find faults and fix problems, thus preventing data corruption. Moreover database should regularly back up and establish a spare machine, makes the database can be restored as quickly as possible from the fault.

Changes made from LDM to PDM (MS Access - Oracle)

We made some syntax changes to LDM model for MS Access and Oracle database. Change from the use of "text" for MS Access and changed "VARCHAR" to "VARCHAR2" or "CHAR". We added another entity INSURANCE_POLICY_COVERAGE as to better performance of database model and normalization. As we had all entities and attributes finished we made also data type definition report, with description of entities, attributes within each entities, attributes definitions and data types for MS Access and Oracle database.

PDM for vehicle insurance service

We have prepared two codes for creation of database in Oracle SQL and MS Access, see Table 10. With ERwin Physycal data model see Appendix 7 and data type identification report see Appendix 4.

Table 10: PDM code for Oracle and MS Access

Nu.	Oracle	MS Access
1	CREATE TABLE INCIDENT (Incident_Id VARCHAR2(20) NOT NULL , Incident_Type VARCHAR2(30) NULL , Incident_Date DATE NOT NULL , Description VARCHAR2(100) NULL , CONSTRAINT XPKINCIDENT_17 PRIMARY KEY (Incident_Id)); CREATE UNIQUE INDEX XPKINCIDENT_17 ON INCIDENT (Incident_Id ASC);	CREATE TABLE INCIDENT (Incident_Id TEXT NOT NULL , Incident_Type TEXT NULL , Incident_Date DATE NOT NULL , Description TEXT NULL , CONSTRAINT XPKINCIDENT_17 PRIMARY KEY (Incident_Id)); CREATE UNIQUE INDEX XPKINCIDENT_17 ON INCIDENT (Incident_Id ASC);
2	CREATE TABLE CUSTOMER (Cust_Id VARCHAR2(20) NOT NULL , Cust_FName VARCHAR2(10) NOT NULL , Cust_LName VARCHAR2(10) NOT NULL , Cust_DOB DATE NOT NULL , Cust_Gender CHAR(2) NOT NULL , Cust_Address VARCHAR2(20) NOT NULL , Cust_MOB_Number INTEGER NOT NULL , Cust_Email VARCHAR2(20) NULL , Cust_Passport_Number VARCHAR2(20) NULL , Cust_Marital_Status CHAR(8) NULL , Cust_PPS_Number INTEGER NULL , CONSTRAINT XPKCUSTOMER_1 PRIMARY KEY (Cust_Id));	CREATE TABLE CUSTOMER (Cust_Id TEXT NOT NULL , Cust_FName TEXT NOT NULL , Cust_LName TEXT NOT NULL , Cust_DOB DATE NOT NULL , Cust_Gender TEXT NOT NULL , Cust_Address TEXT NOT NULL , Cust_MOB_Number NUMBER NOT NULL , Cust_Email TEXT NULL , Cust_Passport_Number TEXT NULL , Cust_Marital_Status TEXT NULL , Cust_PPS_NumberNUMBER NULL , CONSTRAINT XPKCUSTOMER_1 PRIMARY KEY (Cust_Id));

	CREATE UNIQUE INDEX XPKCUSTOMER_1 ON CUSTOMER
	(Cust_Id ASC);
		CREATE UNIQUE INDEX XPKCUSTOMER_1 ON CUSTOMER
		(Cust_Id ASC);

3	CREATE TABLE INCIDENT_REPORT (Incident_Report_Id VARCHAR2(20) NOT NULL , Incident_Type CHAR(10) NULL , Incident_Inspector VARCHAR2(20) NULL , Incident_Cost INTEGER NULL , Incident_Report_Description VARCHAR2(100) NULL , Incident_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKINCIDENT_REPORT_18 PRIMARY KEY (Incident_Report_Id,Incident_Id,Cust_Id), CONSTRAINT R_83 FOREIGN KEY (Incident_Id) REFERENCES INCIDENT (Incident_Id), CONSTRAINT R_86 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKINCIDENT_REPORT_18 ON INCIDENT_REPORT (Incident_Report_Id ASC,Incident_Id ASC,Cust_Id ASC);	CREATE TABLE I_REPORT (Incident_Report_Id TEXT NOT NULL , Incident_Type TEXT NULL , Incident_Inspector TEXT NULL , Incident_Cost NUMBER NULL , Incident_Report_Description TEXT NULL , Incident_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKINCIDENT_REPORT_18 PRIMARY KEY (Incident_Report_Id,Incident_Id,Cust_Id), CONSTRAINT R_83 FOREIGN KEY (Incident_Id) REFERENCES INCIDENT (Incident_Id), CONSTRAINT R_86 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKINCIDENT_REPORT_18 ON INCIDENT_REPORT (Incident_Report_Id ASC,Incident_Id ASC,Cust_Id ASC); 4
4	CREATE TABLE INSURANCE_COMPANY (Company_Name VARCHAR2(20) NOT NULL , Company_Address VARCHAR2(20) NULL , Company_Contact_Number INTEGER NULL , Company_Fax INTEGER NULL , Company_Email VARCHAR2(20) NULL , Company_Website VARCHAR2(20) NULL , Company_Location VARCHAR2(20) NULL ,	CREATE TABLE INSURANCE_COMPANY (Company_Name TEXT NOT NULL , Company_Address TEXT NULL , Company_Contact_Number NUMBER NULL , Company_Fax NUMBER NULL , Company_Email TEXT NULL , Company_Website TEXT NULL , Company_Location TEXT NULL ,

	Company_Department_Name VARCHAR2(20) NULL ,
	Company_Office_Name VARCHAR2(20) NULL ,
	CONSTRAINT XPKINSURANCE_COMPANY_15 PRIMARY KEY (Company_Name)
	);
	
	CREATE UNIQUE INDEX XPKINSURANCE_COMPANY_15 ON INSURANCE_COMPANY
	(Company_Name ASC);
		Company_Department_Name TEXT NULL ,
		Company_Office_Name TEXT NULL ,
		CONSTRAINT XPKINSURANCE_COMPANY_15 PRIMARY KEY (Company_Name)
		);
		
		CREATE UNIQUE INDEX XPKINSURANCE_COMPANY_15 ON INSURANCE_COMPANY
		(Company_Name ASC);

5	CREATE TABLE DEPARTMENT (Department_Name VARCHAR2(20) NOT NULL , Department_ID CHAR(18) NOT NULL , Department_Staff CHAR(18) NULL , Department_Offices CHAR(18) NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKDEPARTMENT PRIMARY KEY (Department_Name,Department_ID,Company_Name), CONSTRAINT R_56 FOREIGN KEY (Department_Name, Company_Name) REFERENCES DEPARTMENT (Department_Name, Company_Name)); CREATE UNIQUE INDEX XPKDEPARTMENT ON DEPARTMENT (Department_Name ASC,Department_ID ASC,Company_Name ASC);	CREATE TABLE DEPARTMENT (Department_Name TEXT NOT NULL , Department_Leader TEXT NULL , Office TEXT NOT NULL , Contact_Information TEXT NOT NULL , Department_Staff TEXT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKDEPARTMENT_10 PRIMARY KEY (Department_Name,Company_Name), CONSTRAINT R_103 FOREIGN KEY (Company_Name) REFERENCES INSURANCE_COMPANY (Company_Name)); CREATE UNIQUE INDEX XPKDEPARTMENT_10 ON DEPARTMENT (Department_Name ASC,Company_Name ASC);
6	CREATE TABLE VEHICLE_SERVICE (Department_Name VARCHAR2(20) NOT NULL , Vehicle_Service_Company_Name VARCHAR2(20) NOT NULL , Vehicle_Service_Address VARCHAR2(20) NULL , Vehicle_Service_Contact VARCHAR2(20) NULL , Vehicle_Service_Incharge VARCHAR2(20) NULL , Vehicle_Service_Type VARCHAR2(20) NULL ,	CREATE TABLE VEHICLE_SERVICE (Department_Name TEXT NOT NULL , Vehicle_Service_Company_Name TEXT NOT NULL , Vehicle_Service_Address TEXT NULL , Vehicle_Service_Contact TEXT NULL , Vehicle_Service_Incharge TEXT NULL , Vehicle_Service_Type TEXT NULL ,

	Department_Id VARCHAR2(20) NOT NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKVEHICLE_SERVICE PRIMARY KEY (Vehicle_Service_Company_Name,Department_Name), CONSTRAINT R_50 FOREIGN KEY (Department_Name, Department_Id, Company_Name) REFERENCES DEPARTMENT (Department_Name, Department_ID, Company_Name)); CREATE UNIQUE INDEX XPKVEHICLE_SERVICE ON VEHICLE_SERVICE (Vehicle_Service_Company_Name ASC,Department_Name ASC);	Department_Id TEXT NOT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKVEHICLE_SERVICE PRIMARY KEY (Vehicle_Service_Company_Name,Department_Name), CONSTRAINT R_50 FOREIGN KEY (Department_Name, Department_Id, Company_Name) REFERENCES DEPARTMENT (Department_Name, Department_Id, Company_Name)); CREATE UNIQUE INDEX XPKVEHICLE_SERVICE ON VEHICLE_SERVICE (Vehicle_Service_Company_Name ASC,Department_Name ASC);
7	CREATE TABLE VEHICLE (Vehicle_Id VARCHAR2(20) NOT NULL , Policy_Id VARCHAR2(20) NULL , Dependent_NOK_Id VARCHAR2(20) NULL , Vehicle_Registration_Number VARCHAR2(20) NOT NULL , Vehicle_Value INTEGER NULL , Vehicle_Type VARCHAR2(20) NOT NULL , Vehicle_Size INTEGER NULL , Vehicle_Number_Of_Seat INTEGER NULL , Vehicle_Manufacturer VARCHAR2(20) NULL , Vehicle_Engine_Number INTEGER NULL , Vehicle_Chasis_Number INTEGER NULL , Vehicle_Number VARCHAR2(20) NULL , Vehicle_Model_Number VARCHAR2(20) NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKVEHICLE_6 PRIMARY KEY (Vehicle_Id,Cust_Id), CONSTRAINT R_92 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKVEHICLE_6 ON VEHICLE	CREATE TABLE VEHICLE (Vehicle_Id TEXT NOT NULL , Policy_Id TEXT NULL , Dependent_NOK_Id TEXT NULL , Vehicle_Registration_Number TEXT NOT NULL , Vehicle_Value NUMBER NULL , Vehicle_Type TEXT NOT NULL , Vehicle_Size NUMBER NULL , Vehicle_Number_Of_Seat NUMBER NULL , Vehicle_Manufacturer TEXT NULL , Vehicle_Engine_Number NUMBER NULL , Vehicle_Chasis_Number NUMBER NULL , Vehicle_Number TEXT NULL , Vehicle_Model_Number TEXT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKVEHICLE_6 PRIMARY KEY (Vehicle_Id,Cust_Id), CONSTRAINT R_92 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKVEHICLE_6 ON VEHICLE

	(Vehicle_Id ASC,Cust_Id ASC);	(Vehicle_Id ASC,Cust_Id ASC);
8	CREATE TABLE PREMIUM_PAYMENT (Premium_Payment_Id VARCHAR2(20) NOT NULL , Policy_Number VARCHAR2(20) NOT NULL , Premium_Payment_Amount INTEGER NOT NULL , Premium_Payment_Schedule DATE NOT NULL , Receipt_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKPREMIUM_PAYMENT_5 PRIMARY KEY (Premium_Payment_Id,Cust_Id), CONSTRAINT R_85 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKPREMIUM_PAYMENT_5 ON PREMIUM_PAYMENT (Premium_Payment_Id ASC,Cust_Id ASC);	CREATE TABLE PREMIUM_PAYMENT (Premium_Payment_Id TEXT NOT NULL , Policy_Number TEXT NOT NULL , Premium_Payment_Amount NUMBER NOT NULL , Premium_Payment_Schedule DATE NOT NULL , Receipt_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKPREMIUM_PAYMENT_5 PRIMARY KEY (Premium_Payment_Id,Cust_Id), CONSTRAINT R_85 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKPREMIUM_PAYMENT_5 ON PREMIUM_PAYMENT (Premium_Payment_Id ASC,Cust_Id ASC);
9	CREATE TABLE RECEIPT (Receipt_Id VARCHAR2(20) NOT NULL , Time DATE NOT NULL , Cost INTEGER NOT NULL , Premium_Payment_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKRECEIPT_21 PRIMARY KEY (Receipt_Id,Premium_Payment_Id,Cust_Id), CONSTRAINT R_84 FOREIGN KEY (Premium_Payment_Id, Cust_Id) REFERENCES PREMIUM_PAYMENT (Premium_Payment_Id, Cust_Id)); CREATE UNIQUE INDEX XPKRECEIPT_21 ON RECEIPT (Receipt_Id ASC,Premium_Payment_Id ASC,Cust_Id ASC);	CREATE TABLE RECEIPT (Receipt_Id TEXT NOT NULL , Tim DATE NOT NULL , Cost NUMBER NOT NULL , Premium_Payment_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKRECEIPT_21 PRIMARY KEY (Receipt_Id,Premium_Payment_Id,Cust_Id), CONSTRAINT R_84 FOREIGN KEY (Premium_Payment_Id, Cust_Id) REFERENCES PREMIUM_PAYMENT (Premium_Payment_Id, Cust_Id)); CREATE UNIQUE INDEX XPKRECEIPT_21 ON RECEIPT (Receipt_Id ASC,Premium_Payment_Id ASC,Cust_Id ASC);

10	CREATE TABLE APPLICATION (Application_Id VARCHAR2(20) NOT NULL , Vehicle_Id VARCHAR2(20) NOT NULL , Application_Status CHAR(8) NOT NULL , Coverage VARCHAR2(50) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKAPPLICATION_2 PRIMARY KEY (Application_Id,Cust_Id), CONSTRAINT R_93 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKAPPLICATION_2 ON APPLICATION (Application_Id ASC,Cust_Id ASC);	CREATE TABLE APPLICATION (Application_Id TEXT NOT NULL , Vehicle_Id TEXT NOT NULL , Application_Status TEXT NOT NULL , Coverage TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKAPPLICATION_2 PRIMARY KEY (Application_Id,Cust_Id), CONSTRAINT R_93 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKAPPLICATION_2 ON APPLICATION (Application_Id ASC,Cust_Id ASC);
11	CREATE TABLE INSURANCE_POLICY (Agreement_id VARCHAR2(20) NOT NULL , Department_Name VARCHAR2(20) NULL , Policy_Number VARCHAR2(20) NULL , Start_Date DATE NULL , Expiry_Date DATE NULL , Term_Condition_Description VARCHAR2(100) NULL , Application_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKINSURANCE_POLICY_4 PRIMARY KEY (Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_95 FOREIGN KEY (Application_Id, Cust_Id) REFERENCES APPLICATION (Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKINSURANCE_POLICY_4 ON INSURANCE_POLICY (Agreement_id ASC,Application_Id ASC,Cust_Id ASC);	CREATE TABLE INSURANCE_POLICY (Agreement_id TEXT NOT NULL , Department_Name TEXT NULL , Policy_Number TEXT NULL , Start_Date DATE NULL , Expiry_Date DATE NULL , Term_Condition_Description TEXT NULL , Application_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKINSURANCE_POLICY_4 PRIMARY KEY (Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_95 FOREIGN KEY (Application_Id, Cust_Id) REFERENCES APPLICATION (Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKINSURANCE_POLICY_4 ON INSURANCE_POLICY (Agreement_id ASC,Application_Id ASC,Cust_Id ASC);
12	CREATE TABLE POLICY_RENEWABLE	CREATE TABLE POLICY_RENEWABLE

	(Policy_Renewable_Id VARCHAR2(20) NOT NULL , Date_Of_Renewal DATE NOT NULL , Type_Of_Renewal CHAR(15) NOT NULL , Agreement_id VARCHAR2(20) NOT NULL , Application_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKPOLICY_RENEWABLE_16 PRIMARY KEY (Policy_Renewable_Id,Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_101 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKPOLICY_RENEWABLE_16 ON POLICY_RENEWABLE (Policy_Renewable_Id ASC,Agreement_id ASC,Application_Id ASC,Cust_Id ASC);	(Policy_Renewable_Id TEXT NOT NULL , Date_Of_Renewal DATE NOT NULL , Type_Of_Renewal TEXT NOT NULL , Agreement_id TEXT NOT NULL , Application_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKPOLICY_RENEWABLE_16 PRIMARY KEY (Policy_Renewable_Id,Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_101 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKPOLICY_RENEWABLE_16 ON POLICY_RENEWABLE (Policy_Renewable_Id ASC,Agreement_id ASC,Application_Id ASC,Cust_Id ASC);
13	CREATE TABLE MEMBERSHIP (Membership_Id VARCHAR2(20) NOT NULL , Membership_Type CHAR(15) NOT NULL , Organisation_Contact VARCHAR2(20) NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKMEMBERSHIP_12 PRIMARY KEY (Membership_Id,Cust_Id), CONSTRAINT R_91 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKMEMBERSHIP_12 ON MEMBERSHIP (Membership_Id ASC,Cust_Id ASC);	CREATE TABLE MEMBERSHIP (Membership_Id TEXT NOT NULL , Membership_Type TEXT NOT NULL , Organisation_Contact TEXT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKMEMBERSHIP_12 PRIMARY KEY (Membership_Id,Cust_Id), CONSTRAINT R_91 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKMEMBERSHIP_12 ON MEMBERSHIP (Membership_Id ASC,Cust_Id ASC);
14	CREATE TABLE QU0TE (	CREATE TABLE QU0TE (

	Quote_Id VARCHAR2(20) NOT NULL , Issue_Date DATE NOT NULL , Valid_From_Date DATE NOT NULL , Valid_Till_Date DATE NOT NULL , Description VARCHAR2(100) NULL , Product_Id VARCHAR2(20) NOT NULL , Coverage_Level VARCHAR2(20) NOT NULL , Application_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKQU0TE_3 PRIMARY KEY (Quote_Id,Application_Id,Cust_Id), CONSTRAINT R_94 FOREIGN KEY (Application_Id, Cust_Id) REFERENCES APPLICATION (Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKQU0TE_3 ON QU0TE (Quote_Id ASC,Application_Id ASC,Cust_Id ASC);	Quote_Id TEXT NOT NULL , Issue_Date DATE NOT NULL , Valid_From_Date DATE NOT NULL , Valid_Till_Date DATE NOT NULL , Description TEXT NULL , Product_Id TEXT NOT NULL , Coverage_Level TEXT NOT NULL , Application_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKQU0TE_3 PRIMARY KEY (Quote_Id,Application_Id,Cust_Id), CONSTRAINT R_94 FOREIGN KEY (Application_Id, Cust_Id) REFERENCES APPLICATION (Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKQU0TE_3 ON QU0TE (Quote_Id ASC,Application_Id ASC,Cust_Id ASC);
15	CREATE TABLE STAFF (Staff_Id VARCHAR2(20) NOT NULL , Staff_Fname VARCHAR2(10) NULL , Staff_LName VARCHAR2(10) NULL , Staff_Adress VARCHAR2(20) NULL , Staff_Contact INTEGER NULL , Staff_Gender CHAR(2) NULL , Staff_Marital_Status CHAR(8) NULL , Staff_Nationality CHAR(15) NULL , Staff_Qualification VARCHAR2(20) NULL , Staff_Allowance INTEGER NULL , Staff_PPS_Number INTEGER NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKSTAFF_9 PRIMARY KEY (Staff_Id,Company_Name), CONSTRAINT R_105 FOREIGN KEY (Company_Name) REFERENCES	CREATE TABLE STAFF (Staff_Id TEXT NOT NULL , Staff_Fname TEXT NULL , Staff_LName TEXT NULL , Staff_Adress TEXT NULL , Staff_Contact NUMBER NULL , Staff_Gender TEXT NULL , Staff_Marital_Status TEXT NULL , Staff_Nationality TEXT NULL , Staff_Qualification TEXT NULL , Staff_Allowance NUMBER NULL , Staff_PPS_Number NUMBER NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKSTAFF_9 PRIMARY KEY (Staff_Id,Company_Name), CONSTRAINT R_105 FOREIGN KEY (Company_Name) REFERENCES

	INSURANCE_COMPANY (Company_Name)
	);
	
	CREATE UNIQUE INDEX XPKSTAFF_9 ON STAFF
	(Staff_Id ASC,Company_Name ASC);
		INSURANCE_COMPANY (Company_Name)
		);
		
		CREATE UNIQUE INDEX XPKSTAFF_9 ON STAFF
		(Staff_Id ASC,Company_Name ASC);

16	CREATE TABLE NOK (Nok_Id VARCHAR2(20) NOT NULL , Nok_Name VARCHAR2(20) NULL , Nok_Address VARCHAR2(20) NULL , Nok_Phone_Number INTEGER NULL , Nok_Gender CHAR(2) NULL , Nok_Marital_Status CHAR(8) NULL , Agreement_id VARCHAR2(20) NOT NULL , Application_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKNOK_14 PRIMARY KEY (Nok_Id,Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_99 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKNOK_14 ON NOK (Nok_Id ASC,Agreement_id ASC,Application_Id ASC,	CREATE TABLE NOK (Nok_Id TEXT NOT NULL , Nok_Name TEXT NULL , Nok_Address TEXT NULL , Nok_Phone_Number NUMBER NULL , Nok_Gender TEXT NULL , Nok_Marital_Status TEXT NULL , Agreement_id TEXT NOT NULL , Application_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKNOK_14 PRIMARY KEY (Nok_Id,Agreement_id,Application_Id,Cust_Id), CONSTRAINT R_99 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id)); CREATE UNIQUE INDEX XPKNOK_14 ON NOK
17	CREATE TABLE PRODUCT (Product_Price INTEGER NULL , Product_Type CHAR(15) NULL , Product_Number VARCHAR2(20) NOT NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKPRODUCT_20 PRIMARY KEY (Product_Number,Company_Name), CONSTRAINT R_107 FOREIGN KEY (Company_Name) REFERENCES	CREATE TABLE PRODUCT (Product_Price NUMBER NULL , Product_Type TEXT NULL , Product_Number TEXT NOT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKPRODUCT_20 PRIMARY KEY (Product_Number,Company_Name), CONSTRAINT R_107 FOREIGN KEY (Company_Name) REFERENCES

	INSURANCE_COMPANY (Company_Name)
	);
	
	CREATE UNIQUE INDEX XPKPRODUCT_20 ON PRODUCT
	(Product_Number ASC,Company_Name ASC);
		INSURANCE_COMPANY (Company_Name)
		);
		
		CREATE UNIQUE INDEX XPKPRODUCT_20 ON PRODUCT
		(Product_Number ASC,Company_Name ASC);

18	CREATE TABLE OFFICE (Office_Name VARCHAR2(20) NOT NULL , Office_Leader VARCHAR2(20) NOT NULL , Contact_Information VARCHAR2(20) NOT NULL , Address VARCHAR2(20) NOT NULL , Admin_Cost INTEGER NULL , Staff VARCHAR2(50) NULL , Department_Name VARCHAR2(20) NOT NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKOFFICE_11 PRIMARY KEY (Office_Name,Department_Name,Company_Name), CONSTRAINT R_104 FOREIGN KEY (Department_Name, Company_Name) REFERENCES DEPARTMENT (Department_Name, Company_Name)); CREATE UNIQUE INDEX XPKOFFICE_11 ON OFFICE (Office_Name ASC,Department_Name ASC,Company_Name ASC);	CREATE TABLE OFFICE (Office_Name TEXT NOT NULL , Office_Leader TEXT NOT NULL , Contact_Information TEXT NOT NULL , Address TEXT NOT NULL , Admin_Cost NUMBER NULL , Staff TEXT NULL , Department_Name TEXT NOT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKOFFICE_11 PRIMARY KEY (Office_Name,Department_Name,Company_Name), CONSTRAINT R_104 FOREIGN KEY (Department_Name, Company_Name) REFERENCES DEPARTMENT (Department_Name, Company_Name)); CREATE UNIQUE INDEX XPKOFFICE_11 ON OFFICE (Office_Name ASC,Department_Name ASC,Company_Name ASC);
19	CREATE TABLE COVERAGE (Coverage_Id VARCHAR2(20) NOT NULL , Coverage_Amount INTEGER NOT NULL , Coverage_Type CHAR(10) NOT NULL , Coverage_Level CHAR(15) NOT NULL , Product_Id VARCHAR2(20) NOT NULL , Coverage_Description VARCHAR2(100) NULL , Covearge_Terms VARCHAR2(50) NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKCOVERAGE_19 PRIMARY KEY	CREATE TABLE COVERAGE (Coverage_Id TEXT NOT NULL , Coverage_Amount NUMBER NOT NULL , Coverage_Type TEXT NOT NULL , Coverage_Level TEXT NOT NULL , Product_Id TEXT NOT NULL , Coverage_Description TEXT NULL , Covearge_Terms TEXT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKCOVERAGE_19 PRIMARY KEY

	(Coverage_Id,Company_Name),
	CONSTRAINT R_102 FOREIGN KEY (Company_Name) REFERENCES INSURANCE_COMPANY (Company_Name)
	);
	
	CREATE UNIQUE INDEX XPKCOVERAGE_19 ON COVERAGE (Coverage_Id ASC,Company_Name ASC);
		(Coverage_Id,Company_Name),
		CONSTRAINT R_102 FOREIGN KEY (Company_Name) REFERENCES INSURANCE_COMPANY (Company_Name)
		);
		
		CREATE UNIQUE INDEX XPKCOVERAGE_19 ON COVERAGE (Coverage_Id ASC,Company_Name ASC);

20	CREATE TABLE INSURANCE_POLICY_COVERAGE (Agreement_id VARCHAR2(20) NOT NULL , Application_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , Coverage_Id VARCHAR2(20) NOT NULL , Company_Name VARCHAR2(20) NOT NULL , CONSTRAINT XPKINSURANCE_POLICY_4_COVERAGE PRIMARY KEY (Agreement_id,Application_Id,Cust_Id,Coverage_Id,Company_Name), CONSTRAINT R_97 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id), CONSTRAINT R_98 FOREIGN KEY (Coverage_Id, Company_Name) REFERENCES COVERAGE (Coverage_Id, Company_Name)); CREATE UNIQUE INDEX XPKINSURANCE_POLICY_4_COVERAGE ON INSURANCE_POLICY_COVERAGE (Agreement_id ASC,Application_Id ASC,Cust_Id ASC,Coverage_Id ASC,Company_Name ASC);	CREATE TABLE INSURANCE_POLICY_COVERAGE (Agreement_id TEXT NOT NULL , Application_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , Coverage_Id TEXT NOT NULL , Company_Name TEXT NOT NULL , CONSTRAINT XPKINSURANCE_POLICY_4_COVERAGE PRIMARY KEY (Agreement_id,Application_Id,Cust_Id,Coverage_Id,Company_Name), CONSTRAINT R_97 FOREIGN KEY (Agreement_id, Application_Id, Cust_Id) REFERENCES INSURANCE_POLICY (Agreement_id, Application_Id, Cust_Id), CONSTRAINT R_98 FOREIGN KEY (Coverage_Id, Company_Name) REFERENCES COVERAGE (Coverage_Id, Company_Name)); CREATE UNIQUE INDEX XPKINSURANCE_POLICY_4_COVERAGE ON INSURANCE_POLICY_COVERAGE (Agreement_id ASC,Application_Id ASC,Cust_Id ASC,Coverage_Id ASC,Company_Name ASC);
21	CREATE TABLE CLAIM (Claim_Id VARCHAR2(20) NOT NULL , Agreement_Id VARCHAR2(20) NOT NULL , Claim_Amount INTEGER NOT NULL , Incident_Id VARCHAR2(20) NOT NULL , Damage_Type VARCHAR2(20) NOT NULL ,	CREATE TABLE CLAIM (Claim_Id TEXT NOT NULL , Agreement_Id TEXT NOT NULL , Claim_Amount NUMBER NOT NULL , Incident_Id TEXT NOT NULL , Damage_Type TEXT NOT NULL ,

	Date_Of_Claim DATE NOT NULL , Claim_Status CHAR(10) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKCLAIM_7 PRIMARY KEY (Claim_Id,Cust_Id), CONSTRAINT R_88 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKCLAIM_7 ON CLAIM (Claim_Id ASC,Cust_Id ASC);	Date_Of_Claim DATE NOT NULL , Claim_Status TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKCLAIM_7 PRIMARY KEY (Claim_Id,Cust_Id), CONSTRAINT R_88 FOREIGN KEY (Cust_Id) REFERENCES CUSTOMER (Cust_Id)); CREATE UNIQUE INDEX XPKCLAIM_7 ON CLAIM (Claim_Id ASC,Cust_Id ASC);
22	CREATE TABLE CLAIM_SETTLEMENT (Claim_Settlement_Id VARCHAR2(20) NOT NULL , Vehicle_Id VARCHAR2(20) NOT NULL , Date_Settled DATE NOT NULL , Amount_Paid INTEGER NOT NULL , Coverage_Id VARCHAR2(20) NOT NULL , Claim_Id VARCHAR2(20) NOT NULL , Cust_Id VARCHAR2(20) NOT NULL , CONSTRAINT XPKCLAIM_SETTLEMENT_8 PRIMARY KEY (Claim_Settlement_Id,Claim_Id,Cust_Id), CONSTRAINT R_90 FOREIGN KEY (Claim_Id, Cust_Id) REFERENCES CLAIM (Claim_Id, Cust_Id)); CREATE UNIQUE INDEX XPKCLAIM_SETTLEMENT_8 ON CLAIM_SETTLEMENT (Claim_Settlement_Id ASC,Claim_Id ASC,Cust_Id ASC);	CREATE TABLE CLAIM_SETTLEMENT (Claim_Settlement_Id TEXT NOT NULL , Vehicle_Id TEXT NOT NULL , Date_Settled DATE NOT NULL , Amount_Paid NUMBER NOT NULL , Coverage_Id TEXT NOT NULL , Claim_Id TEXT NOT NULL , Cust_Id TEXT NOT NULL , CONSTRAINT XPKCLAIM_SETTLEMENT_8 PRIMARY KEY (Claim_Settlement_Id,Claim_Id,Cust_Id), CONSTRAINT R_90 FOREIGN KEY (Claim_Id, Cust_Id) REFERENCES CLAIM (Claim_Id, Cust_Id)); CREATE UNIQUE INDEX XPKCLAIM_SETTLEMENT_8 ON CLAIM_SETTLEMENT (Claim_Settlement_Id ASC,Claim_Id ASC,Cust_Id ASC);

Who has access to certain parts of database

“A CRUD matrix is a table showing the functions in an application containing SQL statement affecting parts of a database” (Williams, 2001). It is a great way to show us what kind interaction appears between user and tables in database. This analysis of possible user scenarios and shows us tables that are very used and those that are not used, and it brings us the view how database is burdened with possible – bottleneck in system performance (Williams, 2001).

Using four SQL statements:

- Create – INSERTE – to store new data
- Read – SELECT – to retrieve data
- Update – UPDATE – to change or modify data
- Delete – DELETE – delete or remove data

More you see in Table 11.

Table 11: CRUD Matrix of database (Create - C, Read-R, Update-U and Delete-D)

MODULES ENTITIES	Customer	Manger of insurance	Insurance agent	Accountant	HR department	Damage inspector	Database administrator	Finance department	
CUSTOMER	CR	R	CRUD	R	-	-	R	-	
APPLICATION	R	R	CRD	R	R	-	CRD	R	
QUOTE	R	R	CRUD	R	R	-	R	R	
INSURANCE_POLICY	R	R	CRUD	R	-	R	CRD	R	
PREMIUM_PAYMENT	-	-	CRUD	CRUD	-	-	RU	CRUD	
VEHICLE	-	R	CRUD	-	-	-	RU	-	
CLAIM	-	R	CRUD	CRD	-	-	RU	CRUD	
CLAIM_SETTLEMENT	R	CRUD	CR	-	-	-	RU	CRUD	
STAFF	-	CR	R	R	CRUD	-	RU	-	
DEPARTMENT	-	R	R	-	CRUD	-	RU	R	
OFFICE	R	R	R	R	CRUD	-	RU	R	
MEMBERSHIP	CR	R	CRD	-	-	-	RU	R	
VEHICLE_SERVICE	R	CRD	CRU	-	-	-	RU	R	
NOK	R	R	CRUD	-	-	-	RU	R	
INSURANCE_COMPANY	R	R	R	R	R	-	RU	R	
POLICY_RENEWABLE	R	CRUD	CRU	-	-	-	RU	-	
INCIDENT	-	CRD	R	-	-	R	RU	R	
INCIDENT_REPORT	R	CRD	R	R	-	CRUD	RU	R	
COVERAGE	R	R	CRD	R	-	-	RU	R	
PRODUCT	R	CRUD	R	R	R	R	RU	R	
RECEIPT	R	CRUD	CRD	CRUD	-	-	RU	CRUD	

(Source: Database answer (Williams, 2001))

Conclusion

We have as class project created database (DB) with all documentations and repots included. Our goal was to create DB for Online vehicle insurance company 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 model (CDM) we continued with logical data model (LDM) and then we made physical data model (PDM) all in Erwin software program. From physical data model we created a code to be rune in Oracle and MS Access data base management system (DBMS). For better understanding for a reader and for our learning we included some theory in each faze we done and documented in project initial document (PID) with reports of progress and work being done.

Bibliography

Aviva, 2014. *About us.* [Online]
Available at: http://www.aviva.com/about-us/
[Accessed 22 October 2014].

Green, D., 2014. *Normalization in Database (With Example).* [Online]
Available at: https://www.udemy.com/blog/normalization-in-database-with-example/
[Accessed 15 11 2014].

Merson, P., 2009. *Reports.* [Online]
Available at: http://www.sei.cmu.edu/reports/09tn024.pdf
[Accessed 22 October 2014].

Silverston, L., 2001. *The data Model Resource Book.* Revised Edition ed. s.l.:John Wiley & Sons.

Speelpenning , J., Daux , P. & Gallus , J., 2001. *Data Modeling and Relational Database Design.* [Online]
Available at: http://www.darkopetrovic.com/pdf/Data-Modeling-and-Relational-Database-Design.pdf
[Accessed 27 November 2014].

Williams, B., 2001. *A CRUD Matrix.* [Online]
Available at: http://www.databaseanswers.org/data_migration/crud_matrix.htm
[Accessed 1 12 2014].

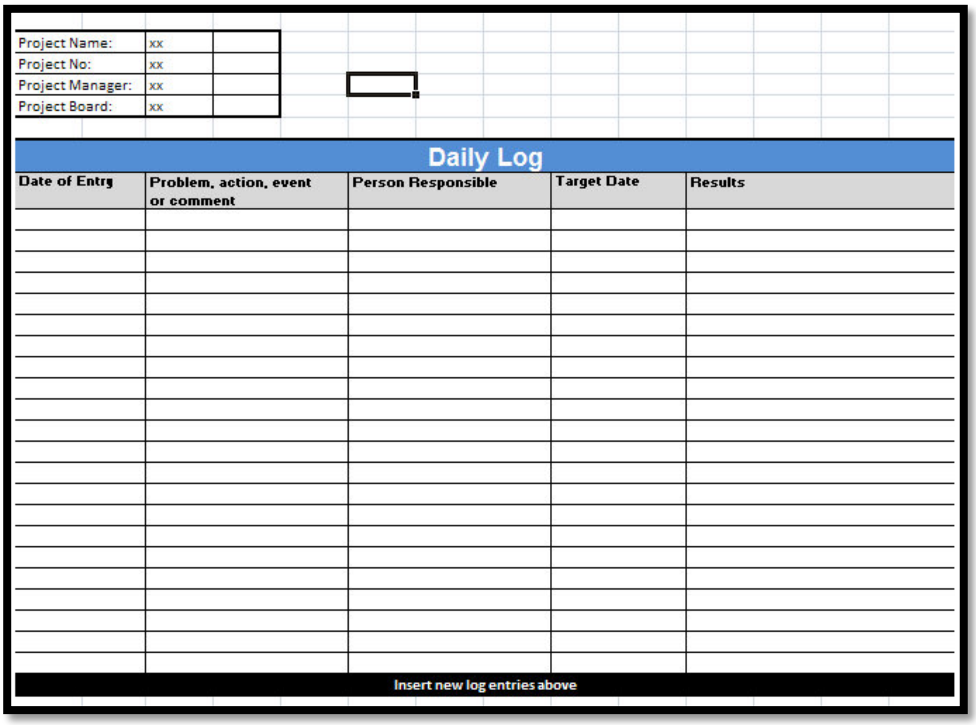
Zickos, C., 2014. *Access Tutorial 2.* [Online]
Available at:
http://www.cs.olemiss.edu/~cbzickos/download_files/191/ppts/Access2013_T02_PPT.pdf
[Accessed 27 November 2014].

Appendices

Appendix 1 : REPORT - Meeting 1 (example)

MEETING 1 - PART A: CDM – Model of DB		
Date	Time	
Meeting called by		
Type of meeting	Discussing about the CDM.	
Facilitator		
Note taker		
Timekeeper		
Attendees		
Tasks		
45 minutes		
Conclusions		
Task delegated, Team knows what to do.		
Action Items	Person Responsible	Deadline
Preparing entities for CDM report.	all TEAM members	
CDM		
5 minutes		
Discussion	On CDM model	
Conclusions		
Discussed about the entities.		
Action Items	Person Responsible	Deadline
Research on the car insurance preparing entities& their relationships.		

Appendix 2: Daily Log Report – template1



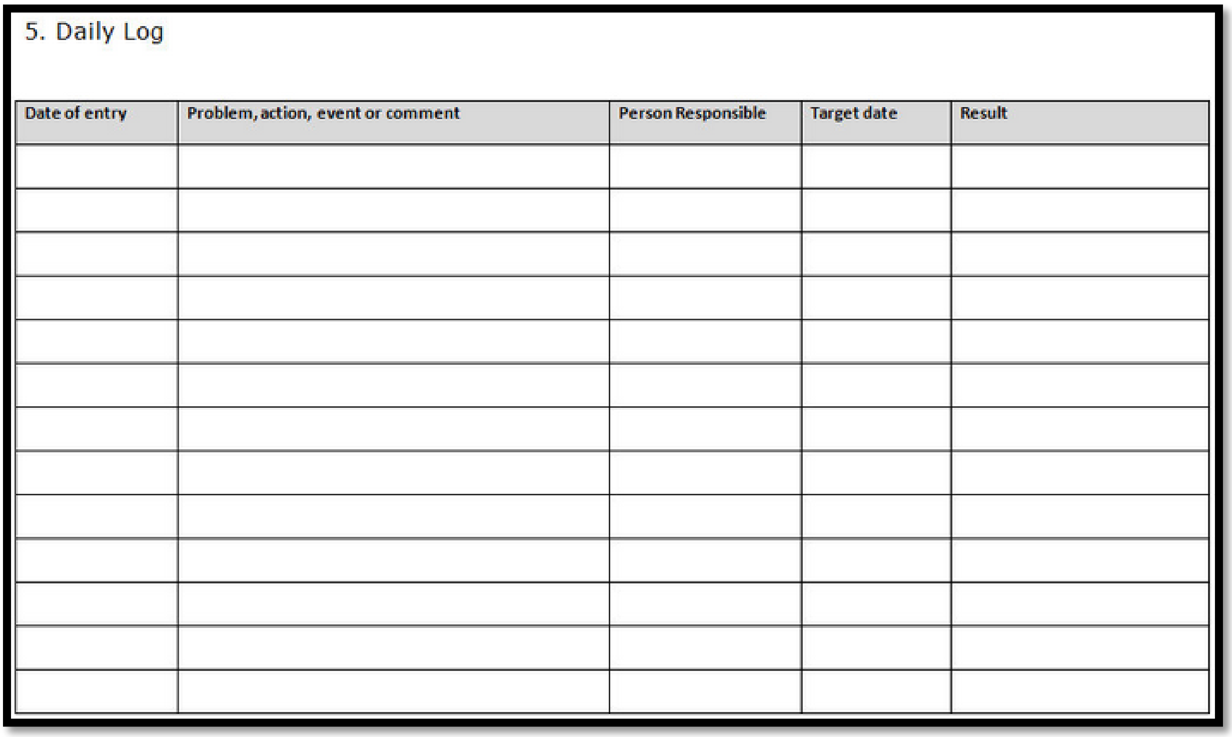
Project Name:	xx	
Project No:	xx	
Project Manager:	xx	
Project Board:	xx	

Daily Log

Date of Entry	Problem, action, event or comment	Person Responsible	Target Date	Results

Insert new log entries above

Appendix 3: Daily Log Report – template2



5. Daily Log

Date of entry	Problem, action, event or comment	Person Responsible	Target date	Result

Appendix 4: Data type identification report

Entity Types	Attributes within each entity types	Attribute definition	MS Access data type	Oracle Data Type
CUSTOMER	CUST_ID	This is the customer's unique identifier and is part of the compound primary key. It is a primary key of the Customer entity type.	TEXT	VARCHAR(20)
	CUST_FNAME	Customer’s first name.	TEXT	VARCHAR(10)
	CUST_LNAME	Customer’s last name.	TEXT	VARCHAR(10)
	CUST_DOB	Customer’s date of birth.	DATE	DATE
	CUST_GENDER	Customer’s gender.	TEXT	CHAR(2)
	CUST_ADDRESS	Address of customer - account holder assigned to insurance policy contract .	TEXT	VARCHAR(20)
	CUST_MOB_NUMBER	Customer’s mobile number.	NUMBER	INTEGER
	CUST_EMAIL	Customer’s email address.	TEXT	VARCHAR(20)
	CUST_PASSPORT_ NUMBER	Customer’s number of passport for identification purposes.	TEXT	VARCHAR(20)
	CUST_MARITAL_STATUS	Customer’s marital status.	TEXT	CHAR(8)
	CUST_PPS_NUMBER	Customer’s personal public number.	NUMBER	INTEGER
APPLICATION	APPLICATION_ID	This is the application unique identifier and is part of the compound primary key. It is a primary key of the application entity type. It records number of application for the insurance being made by customer.	TEXT	VARCHAR2（20）
	CUST_ID	Customer's unique identifier.	TEXT	VARCHAR2（20）
	VEHICLE_ID	Vehicle unique identifier.	TEXT	VARCHAR2（20）
	APPLICATION_STATUS	Status of customer applying for coverage.	TEXT	CHAR（8）
	COVERAGE	What kind of coverage can customer choose from.	TEXT	VARCHAR2（50）
QUOTE	QUOTE_ID	This is the quote unique identifier and is part of the compound primary key. It is a primary key of the quote entity type	TEXT	VARCHAR2（20）
	APPLICATION_ID	Application unique identifier.	TEXT	VARCHAR2（20）

INSURANCE_POLICY	CUST_ID	Customer’s Unique Identifier.	TEXT	VARCHAR2 (20)
	ISSUE_DATE	Date when Quote was issued to Customer.	DATE/TIME	DATE
	VALID_FROM_DATE	Beginning date when Quote remains Valid.	DATE/TIME	DATE
	VALID_TILL_DATE	End date of the Quote validity.	DATE/TIME	DATE
	DESCRIPTION	Any additional information regarding the Quote.	TEXT	VARCHAR2（100）
	PRODUCT_ID	This is the unique Product identifier.	TEXT	VARCHAR2（20）
	COVERAGE_LEVEL	This defines level of coverage the customer has choosen.	TEXT	VARCHAR2（20）
	AGREEMENT_ID	This is Agreement unique identifier. It is also the primary key of Agreement entity type.	TEXT	VARCHAR2（20）
	APPLICATION_ID	This is a unique Application Identifier.	TEXT	VARCHAR2（20）
	CUST_ID	This identifies the Customer.	TEXT	VARCHAR2（20）
	DEPARTMENT _NAME	This defines the different departments within the company by their names.	TEXT	VARCHAR2（20）
	POLICY_NUMBER	This is a unique identifier of the Policy document.	TEXT	VARCHAR2（20）
	START_DATE	The date when the Insurance policy started as legal.	DATE/TIME	DATE
	EXPIRY_DATE	The end date of the Insurance policy as per the contract.	DATE/TIME	DATE
	TERM_CONDITION_DESCRIPTION	Defines details of the Policy document with unique requirements.	TEXT	VARCHAR2（100）
PREMIUM_PAYMENT	PREMIUM_PAYMENT_ID	This is a unique identifier of premium payment paid in regards to the insurance policy. It is also a primary key.	TEXT	VARCHAR2(20)
	CUST_ID	This is the unique identifier for the customer.	TEXT	VARCHAR2(20)
	POLICY_NUMBER	This uniquely identifies the policy.	TEXT	VARCHAR2(20)
	PREMIUM_PAYMENT_SCHEDULE	This defines the different stages that premium payments are made by the customer.	DATE/TIME	DATE
	PREMIUM_PAYMENT_	This defines the amount paid by the customer in regard to the insurance policy	NUMBER	INTEGER

	AMOUNT	taken.		
	RECEIPT_ID	This identifies the amount of premium received by the insurance company	TEXT	VARCHAR2(20)
VEHICLE	VEHICLE_ID	This is a unique identifier of the Vehicle insured. It is also a primary key.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's Unique Identifier.	TEXT	VARCHAR2(20)
	POLICY_ID	Policy unique identifier.	TEXT	VARCHAR2 (20)
	DEPENDENT_NOK_ID	Identifies the dependent next of kin.	TEXT	VARCHAR2(20)
	VEHICLE_REGISTRATION_NUMBER	Defines the Vehicle insured.	TEXT	VARCHAR2(20)
	VEHICLE_VALUE	This shows the value of the insured vehicle in amounts.	NUMBER	INTEGER
	VEHICLE_TYPE	This defines the vehicle insured by the type.	TEXT	VARCHAR2(20)
	VEHICLE_SIZE	This defines the vehicle insured by the size.	NUMBER	INTEGER
	VEHICLE_NUMBER_OF_SEAT	This defines the vehicle insured by the number of seats.	NUMBER	INTEGER
	VEHICLE_MANUFACTURER	This defines the vehicle insured by the manufacturer.	TEXT	VARCHAR2(20)
	VEHICLE_ENGINE_NUMBER	This defines the vehicle insured by the engine number.	NUMBER	INTEGER
	VEHICLE_CHASSIS_NUMBER	This defines the vehicle insured by the Chassis number	NUMBER	INTEGER
	VEHICLE_NUMBER	This explains the number of vehicles insured under one customer	TEXT	VARCHAR2(20)
	VEHICLE_MODEL_NUMBER	This defines the vehicle by the model number.	TEXT	VARCHAR2(20)
CLAIM	CLAIM_ID	Unique identifier where each customer will get an id number for their claim.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique identifier number.	TEXT	VARCHAR2(20)
	AGREEMENT_ID	Agreement unique identifier number.	TEXT	VARCHAR2(20)
	CLAIM_AMOUNT	Records of customer claimed amount.	NUMBER	INTEGER
	INCIDENT_ID	Unique identifier of incidents happened with customers.	TEXT	VARCHAR2(20)
	DAMAGE_TYPE	Records types of damage to the customers vehicles.	TEXT	VARCHAR2(20)
	DATE_OF_CLAIM	Records the date on which customer claimed for amount.	DATE/TIME	DATE
	CLAIM_STATUS	Customer can see their claimed status.	TEXT	CHAR(10)
CLAIM_SETTLE	CLAIM_SETTLEMENT_ID	This ID will be generated at the time of settlement of the claim.	TEXT	VARCHAR2(20)

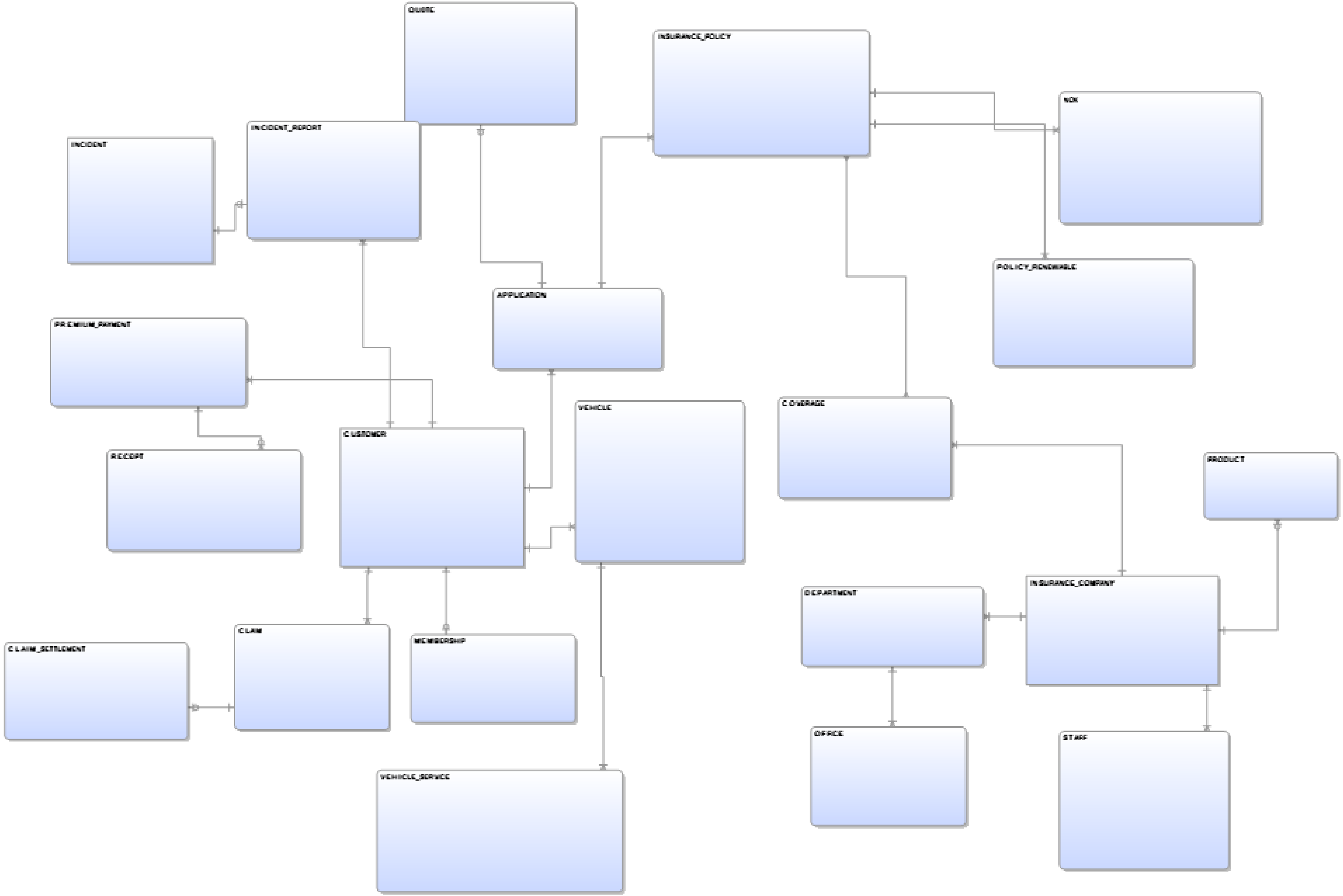
MENT	CLAIM_ID	This is the foreign key from CLAIM entity. Given to the customer claimed for amount.	TEXT	VARCHAR2(20)
	CUST_ID	Customer’s unique identifier number.	TEXT	VARCHAR2(20)
	VEHICLE_ID	Unique vehicle identifier number.	TEXT	VARCHAR2(20)
	DATE_SETTLED	Keep in record the date on which the claim is settled.	DATE/TIME	DATE
	AMOUNT_PAID	Keep in record the amount paid to the customers for their claim.	NUMBER	INTEGER
	COVERAGE_ID	Covers the amount and has unique identifier number.	TEXT	VARCHAR2(20)
STAFF	STAFF_ID	Every staff has their own unique identifier number.	TEXT	VARCHAR2(20)
	COMPANY_NAME	Has the name of the company on staff and customers records.	TEXT	VARCHAR2(20)
	STAFF_FNAME	Staff first name.	TEXT	VARCHAR2(10)
	STAFF_LNAME	Staff last name.	TEXT	VARCHAR2(10)
	STAFF_ADDRESS	Staffs addresses.	TEXT	VARCHAR2(20)
	STAFF_CONTACT	Staffs contact number.	NUMBER	INTEGER
	STAFF_GENDER	Staffs gender.	TEXT	CHAR(2)
	STAFF_MARITAL_ STATUS	Staffs marital status.	TEXT	CHAR(8)
	STAFF_NATIONALITY	Staffs nationality.	TEXT	CHAR(15)
	STAFF_QUALIFICATION	Records all the details of staff’s qualifications.	TEXT	VARCHAR2(20)
	STAFF_ALLOWANCE	Records the allowance given to the staffs.	NUMBER	INTEGER
	STAFF_PPS_NUMBER	Unique Identifier number of the staffs.	NUMBER	INTEGER
DEPARTMENT	DEPARTMENT_NAME	Has the name of the department among many.	TEXT	VARCHAR2(20)
	COMPANY_NAME	Every department has connected to a company.	TEXT	VARCHAR2(20)
	OFFICE	Records the details of the office.	TEXT	VARCHAR2(20)
	CONTACT_ INFORMATION	Records the details of the contacts of the department.	TEXT	VARCHAR2(20)
	DEPARTMENT_STAFF	Include the details of the staffs of particular department.	TEXT	VARCHAR2(50)
	DEPARTMENT_LEADER	Every department has their own leader.	TEXT	VARCHAR2(20)
OFFICE	OFFICE_NAME	This includes the name of related office from several offices.	TEXT	VARCHAR2(20)
	DEPARTMENT_NAME	This is the Foreign Key from department.	TEXT	VARCHAR2(20)
	COMPANY_NAME	This is the foreign key from staff.	TEXT	VARCHAR2(20)
	OFFICE_LEADER	Every office has their own leader.	TEXT	VARCHAR2(20)

	CONTACT_INFORMATION	Records all the contacts of the different office and departments.	TEXT	VARCHAR2(20)
	ADDRESS	Details of the office address.	TEXT	VARCHAR2(20)
	ADMIN_COST	Records the details of the administration cost incurred.	NUMBER	INTEGER
	STAFF	Details of the staffs from the related office.	TEXT	VARCHAR2(50)
MEMBERSHIP	MEMBERSHIP_ID	This is the customer membership's unique identifier and is part of the compound primary key. It is a primary key of the Membership entity type.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique ID it is a foreign key from customer entity.	TEXT	VARCHAR2(20)
	MEMBERSHIP_TYPE	Membership’s type customer has.	TEXT	CHAR(15)
	ORGANISATION_CONTACT	Contact Details of organization.	TEXT	VARCHAR2(20)
VEHICLE_SERVICE	VEHICLE_SERVICE	This is the customer vehicle service's unique identifier and is part of the compound primary key. It is a primary key of the vehicle service entity.	TEXT	VARCHAR2(20)
	VEHICLE_ID	Vehicle's unique ID it is a foreign key from vehicle entity.	TEXT	VARCHAR(20)
	CUST_ID	Customer's unique ID it is a foreign key from customer entity.	TEXT	VARCHAR(20)
	DEPARTMENT_NAME	Name of vehicle service department.	TEXT	CHAR(20)
	VEHICLE_SERVICE_ADDRESS	Vehicle service department's address.	TEXT	VARCHAR2(20)
	VEHICLE_SERVICE_CONTACT	Vehicle service department's contact details.	TEXT	VARCHAR2(20)
	VEHICLE_SERVICE_INCHARGE	Vehicle service department's leader.	TEXT	CHAR(20)
	VEHICLE_SERVICE_TYPE	Vehicle service department's type.	TEXT	VARCHAR2(20)
NOK	NOK_ID	This is the NOK's unique identifier and is part of the compound primary key. It is a primary key of the NOk entity.	TEXT	VARCHAR2(20)
	AGREEMENT_ID	Agreement's unique ID it is a foreign key from insurance policy entity.	TEXT	VARCHAR2(20)
	APPLICATION_ID	Application's unique ID it is a foreign key from application entity.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique ID it is a foreign key from customer entity.	TEXT	VARCHAR2(20)
	NOK_NAME	NOK's name.	TEXT	VARCHAR2(20)
	NOK_ADDRESS	NOK's address.	TEXT	VARCHAR2(20)
	NOK_PHONE_NUMBER	NOK's phone number.	NUMBER	INTEGER

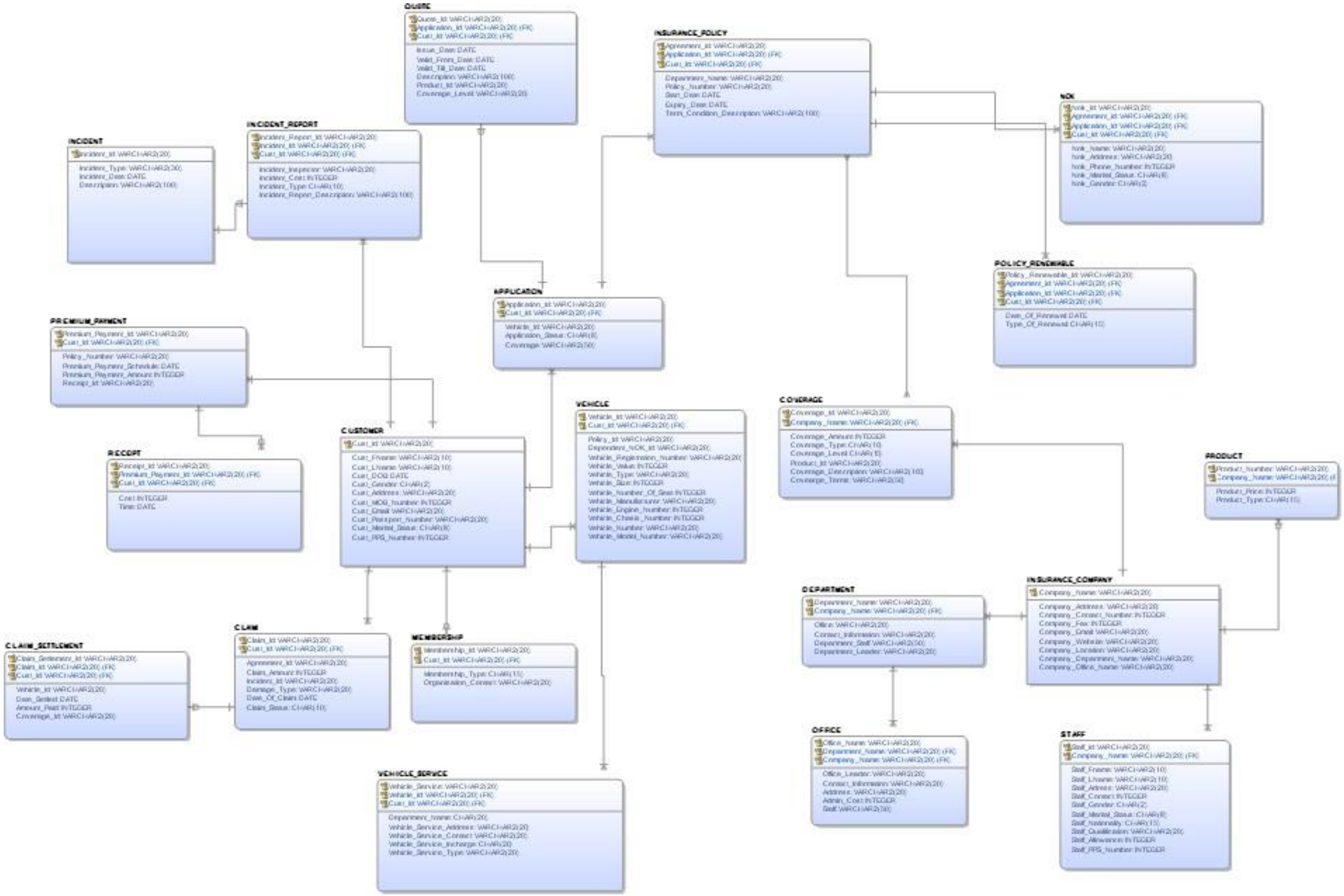
	NOK_MARITAL_STATUS	Marital status of NOK.	TEXT	CHAR(8)
	NOK_GENDER	NOK's gender.	TEXT	CHAR(2)
INSURANCE_C O MPANY	COMPANY_ NAME	This is the company's unique identifier and is part of the compound primary key. It is a primary key of the company entity.	TEXT	VARCHAR2(20)
	COMPANY_ADDRESS	Company's address.	TEXT	VARCHAR2(20)
	COMPANY_CONTACT_ NUMBER	Company's contact number.	NUMBER	INTEGER
	COMPANY_FAX	Company's fax number.	NUMBER	INTEGER
	COMPANY_ EMAIL	Company's email address.	TEXT	VARCHAR2(20)
	COMPANY _WEBSITE	Website address of company.	TEXT	VARCHAR2(20)
	COMPANY _LOCATION	Company's location.	TEXT	VARCHAR2(20)
	COMPANY _DEPARTMENT_NAME	Department name of company.	TEXT	VARCHAR2(20)
	COMPAN_ OFFICE_NAME	Office name of company.	TEXT	VARCHAR2(20)
POLICY_RENE W ABLE	POLICY_RENEWABLE_ID	This is the policy renewable's unique identifier and is part of the compound primary key. It is a primary key of the policy renewable entity.	TEXT	VARCHAR2(20)
	AGREEMENT_ID	Agreement's unique ID it is a foreign key from insurance policy entity.	TEXT	VARCHAR2(20)
	APPLICATION_ID	Application's unique ID it is a foreign key from application entity.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique ID it is a foreign key from customer entity.	TEXT	VARCHAR2(20)
	DATE_OF_RENEWAL	Date of renewable.	DATE	DATE
	TYPE_OF_RENEWAL	Type of renewable policy.	TEXT	CHAR(15)
INCIDENT	INCIDENT_ID	This is the incident's unique identifier and is part of the compound primary key. It is a primary key of the incident entity.	TEXT	VARCHAR2(20)
	INCIDENT_TYPE	Type of incident.	TEXT	VARCHAR2(30)
	INCIDENT_DATE	Date of incident.	DATE	DATE
	DESCRIPTION	Description of incident.	TEXT	VARCHAR2(100)
INCIDENT_ REPORT	INCIDENT_REPORT_ID	Incident report Unique Identifier.	LONG	VARCHAR2(20)
	INCIDENT_ID	This identifies the incident.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique identifier.	TEXT	VARCHAR2(20)
	INCIDENT_INSPECTOR	This shows the details of the inspector who handled the particular incident.	TEXT	VARCHAR2(20)

	INCIDENT_COST	This explains the cost spent on that particular customer vehicle.	NUMBER	INTEGER
	INCIDENT_TYPE	This shows the type of the incident in that vehicle.	TEXT	CHAR(10)
	INCIDENT_REPORT_ DESCRIPTION	This details the essential incident occurrences on the customer vehicle.	TEXT	VARCHAR2(100)
COVERAGE	COVERAGE_ID	Coverage Unique Identifier	TEXT	VARCHAR2(20)
	COMPANY_NAME	This shows the name of the insurance company	TEXT	VARCHAR220)
	COVERAGE_AMOUNT	This records the coverage amount.	NUMBER	INTEGER
	COVERAGE_TYPE	This defines the coverage insured by the type.	TEXT	CHAR(10)
	COVERAGE_LEVEL	This explains the levels in the coverage.	TEXT	CHAR(15)
	PRODUCT_ID	Product Unique Identifier.	TEXT	VARCHAR2(20)
	COVERAGE_ DESCRIPTION	This explains all the essential coverage details of the insurance policy to the customer.	TEXT	VARCHAR2(100)
	COVERAGE_TERMS	This explains the unique policies with regard to the coverage.	TEXT	VARCHAR2(50)
PRODUCT	PRODUCT_NUMBER	This shows the number given to the product.	NUMBER	INTEGER
	COMPANY_NAME	This shows the name of the insurance company	TEXT	VARCHAR2(20)
	PRODUCT_PRICE	This shows the price of the product.	NUMBER	INTEGER
	PRODUCT_TYPE	This defines the product type.	TEXT	CHAR(15)
RECEIPT	RECEIPT_ID	Receipt Unique Identifier this records the payments from customer to the insurance company.	TEXT	VARCHAR2(20)
	PREMIUM_PAYMENT_ ID	This is a unique identifier of premium payment paid in regards to the insurance policy. It is also a primary key.	TEXT	VARCHAR2(20)
	CUST_ID	Customer's unique identifier.	TEXT	VARCHAR2(20)
INSURANCE _ POLICY_ COVERAGE	AGREEMENT_ID	This defines the terms and conditions of the contract.	TEXT	VARCHAR2(20)
	COVERAGE_ID	Records the details of the Vehicle policy coverage.	TEXT	VARCHAR2(20)

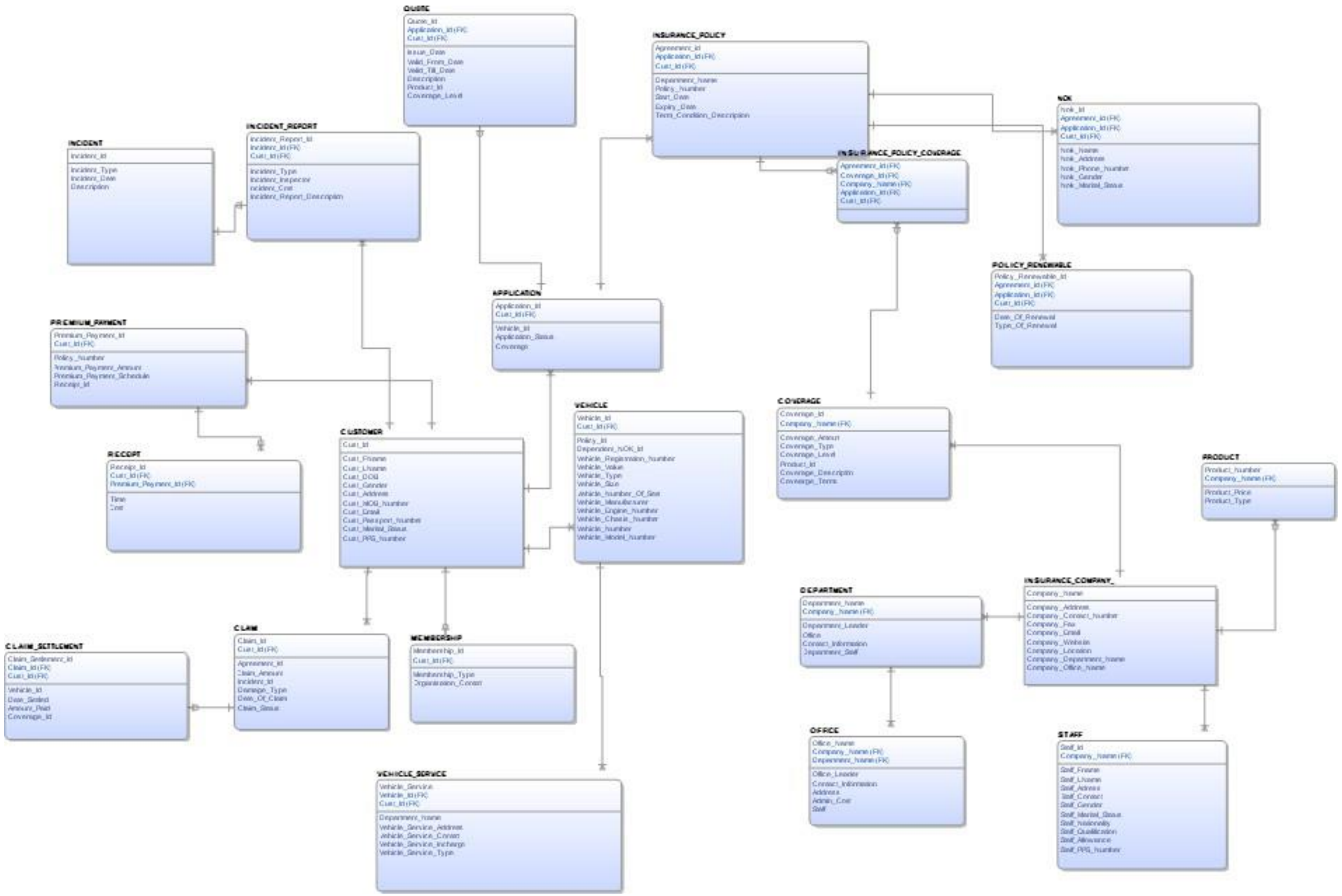
Appendix 5: CDM model – Erwin



Appendix 6: LDM model – Erwin


QUOTE
Quote_id: VARCHAR2(20)
Application_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Issue_Date: DATE
Valid_From_Date: DATE
Valid_Till_Date: DATE
Description: VARCHAR2(100)
Product_id: VARCHAR2(20)
Coverage_Level: VARCHAR2(20)
INSURANCE_POLICY
Agreement_id: VARCHAR2(20)
Application_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Department_Name: VARCHAR2(20)
Policy_Number: VARCHAR2(20)
Start_Date: DATE
Expiry_Date: DATE
Term_Condition_Description: VARCHAR2(100)
NOK
Nok_id: VARCHAR2(20)
Agreement_id: VARCHAR2(20) (FK)
Application_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Nok_Name: VARCHAR2(20)
Nok_Address: VARCHAR2(20)
Nok_Phone_Number: INTEGER
Nok_Marital_Status: CHAR(8)
Nok_Gender: CHAR(2)
INCIDENT
Incident_id: VARCHAR2(20)
Incident_Type: VARCHAR2(20)
Incident_Date: DATE
Description: VARCHAR2(100)
INCIDENT_REPORT
Incident_Report_id: VARCHAR2(20)
Incident_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Incident_Inspector: VARCHAR2(20)
Incident_Cost: INTEGER
Incident_Type: CHAR(10)
Incident_Report_Description: VARCHAR2(100)
POLICY_RENEWABLE
Policy_Renewable_id: VARCHAR2(20)
Agreement_id: VARCHAR2(20) (FK)
Application_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Date_Of_Renewal: DATE
Type_Of_Renewal: CHAR(10)
APPLICATION
Application_id: VARCHAR2(20)
Cust_id: VARCHAR2(20) (FK)
Vehicle_id: VARCHAR2(20)
Application_Status: CHAR(8)
Coverage: VARCHAR2(50)
PREMIUM_PAYMENT
Premium_Payment_id: VARCHAR2(20)
Cust_id: VARCHAR2(20) (FK)
Policy_Number: VARCHAR2(20)
Premium_Payment_Schedule: DATE
Premium_Payment_Amount: INTEGER
Receipt_id: VARCHAR2(20)
VEHICLE
Vehicle_id: VARCHAR2(20)
Cust_id: VARCHAR2(20) (FK)
Policy_id: VARCHAR2(20)
Dependent_NOK_id: VARCHAR2(20)
Vehicle_Registration_Number: VARCHAR2(20)
Vehicle_Value: INTEGER
Vehicle_Type: VARCHAR2(20)
Vehicle_Size: INTEGER
Vehicle_Number_Of_Seat: INTEGER
Vehicle_Manufacturer: VARCHAR2(20)
Vehicle_Engine_Number: INTEGER
Vehicle_Chasis_Number: INTEGER
Vehicle_Number: VARCHAR2(20)
Vehicle_Model_Number: VARCHAR2(20)
COVERAGE
Coverage_id: VARCHAR2(20)
Company_Name: VARCHAR2(20) (FK)
Coverage_Amount: INTEGER
Coverage_Type: CHAR(10)
Coverage_Level: CHAR(5)
Product_id: VARCHAR2(20)
Coverage_Description: VARCHAR2(100)
Coverage_Terms: VARCHAR2(50)
CUSTOMER
Cust_id: VARCHAR2(20)
Cust_Fname: VARCHAR2(10)
Cust_Lname: VARCHAR2(10)
Cust_DOB: DATE
Cust_Gender: CHAR(2)
Cust_Address: VARCHAR2(20)
Cust_MOB_Number: INTEGER
Cust_Email: VARCHAR2(20)
Cust_Passport_Number: VARCHAR2(20)
Cust_Marital_Status: CHAR(8)
Cust_PPS_Number: INTEGER
RECEIPT
Receipt_id: VARCHAR2(20)
Premium_Payment_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Cost: INTEGER
Time: DATE
PRODUCT
Product_Number: VARCHAR2(20)
Company_Name: VARCHAR2(20) (F
Product_Price: INTEGER
Product_Type: CHAR(15)
DEPARTMENT
Department_Name: VARCHAR2(20)
Company_Name: VARCHAR2(20) (FK)
Office: VARCHAR2(20)
Contact_Information: VARCHAR2(20)
Department_Staff: VARCHAR2(50)
Department_Leader: VARCHAR2(20)
INSURANCE_COMPANY
Company_Name: VARCHAR2(20)
Company_Address: VARCHAR2(20)
Company_Contact_Number: INTEGER
Company_Fax: INTEGER
Company_Email: VARCHAR2(20)
Company_Website: VARCHAR2(20)
Company_Location: VARCHAR2(20)
Company_Department_Name: VARCHAR2(20)
Company_Office_Name: VARCHAR2(20)
CLAIM_SETTLEMENT
Claim_Settlement_id: VARCHAR2(20)
Claim_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Vehicle_id: VARCHAR2(20)
Date_Settled: DATE
Amount_Paid: INTEGER
Coverage_id: VARCHAR2(20)
CLAIM
Claim_id: VARCHAR2(20)
Cust_id: VARCHAR2(20) (FK)
Agreement_id: VARCHAR2(20)
Claim_Amount: INTEGER
Incident_id: VARCHAR2(20)
Damage_Type: VARCHAR2(20)
Date_Of_Claim: DATE
Claim_Status: CHAR(10)
MEMBERSHIP
Membership_id: VARCHAR2(20)
Cust_id: VARCHAR2(20) (FK)
Membership_Type: CHAR(15)
Organisation_Contact: VARCHAR2(20)
OFFICE
Office_Name: VARCHAR2(20)
Department_Name: VARCHAR2(20) (FK)
Company_Name: VARCHAR2(20) (FK)
Office_Leader: VARCHAR2(20)
Contact_Information: VARCHAR2(20)
Address: VARCHAR2(20)
Admin_Cost: INTEGER
Staff: VARCHAR2(50)
STAFF
Staff_id: VARCHAR2(20)
Company_Name: VARCHAR2(20) (FK)
Staff_Fname: VARCHAR2(10)
Staff_Lname: VARCHAR2(10)
Staff_Adress: VARCHAR2(20)
Staff_Contact: INTEGER
Staff_Gender: CHAR(2)
Staff_Marital_Status: CHAR(8)
Staff_Nationality: CHAR(15)
Staff_Qualification: VARCHAR2(20)
Staff_Allowance: INTEGER
Staff_PPS_Number: INTEGER
VEHICLE_SERVICE
Vehicle_Service: VARCHAR2(20)
Vehicle_id: VARCHAR2(20) (FK)
Cust_id: VARCHAR2(20) (FK)
Department_Name: CHAR(20)
Vehicle_Service_Address: VARCHAR2(20)
Vehicle_Service_Contact: VARCHAR2(20)
Vehicle_Service_Incharge: CHAR(20)
Vehicle_Service_Type: VARCHAR2(20)

Appendix 7: PDM model –ERWin


QUOTE
Quote_Id
Application_Id (FK)
Cust_Id (FK)
Issue_Date
Valid_From_Date
Valid_Till_Date
Description
Product_Id
Coverage_Level
INSURANCE_POLICY
Agreement_Id
Application_Id (FK)
Cust_Id (FK)
Department_Name
Policy_Number
Start_Date
Expiry_Date
Term_Condition_Description
NOK
Nok_Id
Agreement_Id (FK)
Application_Id (FK)
Cust_Id (FK)
Nok_Name
Nok_Address
Nok_Phone_Number
Nok_Gender
Nok_Marital_Status
INCIDENT
Incident_Id
Incident_Type
Incident_Date
Description
INCIDENT_REPORT
Incident_Report_Id
Incident_Id (FK)
Cust_Id (FK)
Incident_Type
Incident_Inspector
Incident_Cost
Incident_Report_Description
INSURANCE_POLICY_COVERAGE
Agreement_Id (FK)
Coverage_Id (FK)
Company_Name (FK)
Application_Id (FK)
Cust_Id (FK)
POLICY_RENEWABLE
Policy_Renewable_Id
Agreement_Id (FK)
Application_Id (FK)
Cust_Id (FK)
Date_Of_Renewal
Type_Of_Renewal
APPLICATION
Application_Id
Cust_Id (FK)
Vehicle_Id
Application_Status
Coverage
PREMIUM_PAYMENT
Premium_Payment_Id
Cust_Id (FK)
Policy_Number
Premium_Payment_Amount
Premium_Payment_Schedule
Receipt_Id
VEHICLE
Vehicle_Id
Cust_Id (FK)
Policy_Id
Dependent_NOK_Id
Vehicle_Registration_Number
Vehicle_Value
Vehicle_Type
Vehicle_Size
Vehicle_Number_Of_Seat
Vehicle_Manufacturer
Vehicle_Engine_Number
Vehicle_Chasis_Number
Vehicle_Number
Vehicle_Model_Number
COVERAGE
Coverage_Id
Company_Name (FK)
Coverage_Amount
Coverage_Type
Coverage_Level
Product_Id
Coverage_Description
Coverage_Terms
CUSTOMER
Cust_Id
Cust_Fname
Cust_Lname
Cust_DOB
Cust_Gender
Cust_Address
Cust_MOB_Number
Cust_Email
Cust_Passport_Number
Cust_Marital_Status
Cust_PPS_Number
RECEIPT
Receipt_Id
Cust_Id (FK)
Premium_Payment_Id (FK)
Time
Cost
PRODUCT
Product_Number
Company_Name (FK)
Product_Price
Product_Type
DEPARTMENT
Department_Name
Company_Name (FK)
Department_Leader
Office
Contact_Information
Department_Staff
INSURANCE_COMPANY_
Company_Name
Company_Address
Company_Contact_Number
Company_Fax
Company_Email
Company_Website
Company_Location
Company_Department_Name
Company_Office_Name
CLAIM_SETTLEMENT
Claim_Settlement_Id
Claim_Id (FK)
Cust_Id (FK)
Vehicle_Id
Date_Settled
Amount_Paid
Coverage_Id
CLAIM
Claim_Id
Cust_Id (FK)
Agreement_Id
Claim_Amount
Incident_Id
Damage_Type
Date_Of_Claim
Claim_Status
MEMBERSHIP
Membership_Id
Cust_Id (FK)
Membership_Type
Organisation_Contact
OFFICE
Office_Name
Company_Name (FK)
Department_Name (FK)
Office_Leader
Contact_Information
Address
Admin_Cost
Staff
STAFF
Staff_Id
Company_Name (FK)
Staff_Fname
Staff_Lname
Staff_Adress
Staff_Contact
Staff_Gender
Staff_Marital_Status
Staff_Nationality
Staff_Qualification
Staff_Allowance
Staff_PPS_Number
VEHICLE_SERVICE
Vehicle_Service
Vehicle_Id (FK)
Cust_Id (FK)
Department_Name
Vehicle_Service_Address
Vehicle_Service_Contact
Vehicle_Service_Incharge
Vehicle_Service_Type