**Unit 7 Reading**

IBM Intellas. (2010) What is a Database Management System?  
[**https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-what-is-database-management-system**](https://www.ibm.com/docs/en/zos-basic-skills?topic=zos-what-is-database-management-system)

A database management system (DBMS) is a computerised data keeping system. Users may either manipulate data in the system or manage the structure of the system themselves.

Relational databases include the following:

**Database –** a grouping of data with related table spaces and index spaces. Typically all the data associated with an application or a group of applications.

**Table –** A structure made up of rows and columns. Rows have no fixed order so can be sorted when retried. Order of columns is the order as when created by the admin. Intersection of each column / row is a value.

Base table – holds persistent data. Temporary table stores intermediate query results. Results table are the query results

A table has columns with the same data type, rows with data for a single element, values is the intersection, of the column and row. The data item for that row category.

**Indexes -** Ordered set of pointers to rows of a table, indexes always in order. It helps performance of data retrieval. It is also for uniqueness.

**Keys -** One or more columns that are identified at the creation of a table or in the definition of referential integrity.

Can only be one primary key – defines the entity. It cannot be null and it must be unique.

Primary key must be unique but can have another unique key in a table. For example, an employee ID and another type of person ID.

Foreign key – dependent on the primary or unique key in another table for referential integrity.

**Letkowski, J. (2015) Doing Database Design with MySQL. Journal of Technology Research.**

MySQL most widely used open-source relational database management system. Paper discusses two use cases, transforming a data model into a physical database and reverse engineering a database into a data model.

Database design is part of the database design process, involves understanding problem to provide details to build logical data structure. Could be initially plain language then transformed into a data model. This is a view to allow precise mapping – can often be expressed in a data definition language and shown as an entity relationship diagram. Some tools allow these diagrams to be transformed into metadata – known as forward engineering. Some can do the reverse, diagrams from a database.

Steps in database design are requirement analysis, high level conceptual design then logical data modelling.

Once requirements are collected, the conceptual schema is a description of the data requirements of the users and a description of the entity types, relationships, and constraints. They are not technical details and are usually easier to understand for non-technical users. Generally undertaken by database professionals, subject matter experts and end users.

General guidelines when structuring logical data models:

Identifying and mapping entities: entity types become tables, entity instances become records.

Describing properties of the attributes: Design may not describe everything in the final entities sets or tables but it should describe keys, or the unique identifiers. Systems often or usually generate auto IDs which are used as keys and can help with privacy for personal info such as social security number

Defining relationships between entities in database, how the system joins together. Relationship types (or cardinality) may also be identified. Examples include one to one, one to many and many to many.

Scenario 1**: Consider a problem of developing a database for an on-line election system that will be utilized to conduct election of new leaders of a non-profit organization**

**Step 1: Analysis** - the initial analysis of the problem statement should focus on identifying entities (entity types). Such statements will also help capture relationship between the objects. As new entities and relationships are added to the data model, other entities, attributes and relationships may emerge.

**Step 2: Model Building –** Example uses MySQL Workbench to build an Enhanced Entity Relationship Diagram. This describes each table / entity. It’s keys and relationships to other entities. This is described in relatively plain language.

**Step 3: Schema Generation –** The software is able to transform the EER diagram into a SQL script. This creates the tables and fields with data types.

Scenario 2: Reverse engineering. Physical databases already exist but their logical data models are lost or have never been developed. Same software can reverse engineer with minimal modifications.

**Conclusion**

Relational databases are complex data structures. When only a few entities, an experienced developer may be able to design directly in SQL. When more complex or working in a team, graphical views can be more important particularly for non-technical and end users.