**Databases**

Databases store collections of data in a way that allows the data to be viewed in multiple forms. In comparison data, a normal file is only accessible in one form whereas a database because of its structure is accessible in various ways.

Early databases were utilized to consolidate where data in an organization is stored.

Instead of each department using their own storage formats, the use of databases allowed organizations to store data uniformly and reduce duplication.

By storing data for all departments in "integrated pools", all departments can then access each other's data easily to gain insight.

Nowadays, the techniques for extracting "insight" from databases have matured with the use of data mining techniques.

Databases are used to store the majority of data on the web as well as the data required for applications and operating systems.

Nowadays databases can store not only text but images video and countless other formats of data. The structure of a database is defined by its Schema.

User permissions can be controlled by granting users access to only a subschema of the database. For example, the engineering department may not need access to accounting subschema.

**Database management systems.**

Users will interact with an application to view data from a database. The application itself communicates with a database management system and does not directly access the actual database.

The database management system provides an abstraction to the actual database, hiding the way the database was implemented, for example, is it spread across multiple machines or just one?

Data independence is achieved when access to the data can be maintained even if the underlying structure of the data is changed.

**Relational databases vs Object orientated databases**

In a relational database, the data collection is stored with relationships between the data.

relational database store data in tables with rows and columns.

In an Object, Orientated Database data is stored in the form of objects (similar to OOP). OOD'd can handle more complex data than an RDS.

Objects in an Orientated Database can encapsulate the way the data is stored using the same features as classes for OOP.

**Commit/Rollback**

When an application interacts with a database it may wish to make multiple calls to read and alter the data in the database.

The interaction may only be valid if all the calls are successful, and if stopped prematurely and calls are missed may cause invalid data in the system. As such databases offer a commit/rollback functionality whereby all calls will be committed only when requested and calls that need to be invalidated can be done so with a rollback.

Commit rollbacks to help to ensure data integrity in the system.