Normalisation

The development of the logical data model (i.e. the tables) involves continuous ***validation*** of the model against the user requirements

Normalisation is the process of testing the correctness of a logical data model.

What does normalisation ensure?

- No data redundancy
- thus removing the possibility of ***update anomalies***

Normalisation helps to identify a suitable set of relations to represent data in the database

Definition

Normalisation is a technique for producing a set of relations with desirable properties, given the data requirements of an organisation

The normalisation process was developed by E.F. Codd (1972)

- Informally defined as a ***series of tests***
- Applied to one (or more) relations
- Determines if the relations satisfy or violate the requirements of a given normal form

Three normal forms were initially proposed:

- 1NF
- 2NF
- 3NF

A stronger definition of 3NF was subsequently defined:

Boyce-Codd Normal Form (BCNF)

This normal form examines candidate keys and determinants.

Higher normal forms were later defined (4NF, 5NF). These deal with situations which are very rare

A database ***schema*** is said to consist of a group of relations.

A ***relation*** is said to consist of a set of attributes.

When the **data requirements** of an organisation are identified, how are these attributes grouped into suitable relations ?

This decision is made based on the following:

- the common sense of the database designer
- by mapping ER diagrams onto relations

Some formal method is required to help the database designer to identify the optimal groupings of attributes for each relation in the schema

All normal forms are based on ***functional dependencies*** among the attributes of a relation

Attributes are classed as either:

- Key attributes
- Non-key attributes

Normalisation is a formal method

It identifies relations based on:

- Primary key (candidate key in BCNF)
- and the functional dependencies between their attributes

Normalisation presents a series of tests applied to ***individual*** relations so that a schema can be normalised to a specific form to prevent possible ***update anomalies***

Data Redundancy & Update Anomalies

The primary aim of normalisation is to ***minimise*** data redundancy thus reducing file storage space.

Remember, storage space is an expensive commodity! The less we need the better.

Consider the data requirements of an organisation are such that it needs to store details of all its branches and all staff working in those branches.

An initial representation of this data might be:

StaffBranch(StaffNo,Sname,Saddr,Position,Salary, Bno,Baddr,TelNo)

A sample table might be:

Staff_Branch

Staff_No	SName	SAddress	Position	Salary	Branch_No	Baddress	Tel_No
1001	John White	19 Taylor Street, London	Manager	€30,000.00	B5	22 Deer Road, London	0171-886-1212
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	€12,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1003	David Ford	63 Ahsby Street, Glasgow	Deputy	€18,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	€9,000.00	B7	16 Argyll Street, Aberdeen	01224-67111
1005	Susan Brand	5 Gt Western Rd, Glasgow	Manager	€24,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1006	Julie Lee	29 Malvern Street, Kilburn	Assistant	€9,000.00	B5	22 Deer Road, London	0171-886-1212

Look at the sample data in the table.

In particular we notice that some branches (B3, B5) have more than one staff member .

The details of each branch are ***repeated*** for each staff member at that branch

=> **Data Redundancy**

A <u>better</u> representation would be:

STAFF(<u>Staffno</u>,Sname,Saddr,Position,Salary,Bno)
BRANCH(<u>Bno</u>,Baddr,TelNo)

Staff

StaffNo	SName	SAddress	Position	Bno	Salary
1001	John White	19 Taylor Street, London	Manager	B5	€30,000.00
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	B3	€12,000.00
1003	David Ford	63 Ashby Street, Glasgow	Deputy	B3	€18,000.00
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	B7	€9,000.00
1005	Susan Brand	5 Gt Western Road, Glasgow	Manager	B3	€24,000.00
1006	Julie Lee	28 Malvern Street, Kilburn	Assistant	B5	€9,000.00

Branch

Bno	BAddress	Tel_No
B3	163 Main Street, Glasgow	0141-339-2178
B5	22 Deer Road, London	0171-886-1212
B7	16 Argyll Street, Glasgow	01224-67125

- Only the branch number is repeated (in a controlled way)
- Tells us where each staff member is located

Redundant data leads to Update Anomalies

This has serious implications for data consistency and data integrity.

There are three categories of update anomalies:

- ***Insertion*** anomalies
- ***Deletion*** anomalies
- ***Modification*** anomalies

Insertion Anomalies

Insert a new staff member into the relation **StaffBranch** :

- Adding a record for a staff member working at branch B3 must also include details of branch B3
- Must be consistent with all other entries for branch B3

Possible inconsistency

Staff_Branch

Staff_No	SName	SAddress	Position	Salary	Branch_No	Baddress	Tel_No
1001	John White	19 Taylor Street, London	Manager	€30,000.00	B5	22 Deer Road, London	0171-886-1212
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	€12,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1003	David Ford	63 Ahsby Street, Glasgow	Deputy	€18,000.00	B3	63 Main St, Glasgow	0141-339-2178
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	€9,000.00	B7	16 Argyll Street, Aberdeen	01224-67111
1005	Susan Brand	5 Gt Western Rd, Glasgow	Manager	€24,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1006	Julie Lee	29 Malvern Street, Kilburn	Assistant	€9,000.00	B5	22 Deer Road, London	0171-886-1212

The individual staff and branch relations ensure that

- branch details are recorded only once and
- thus prevents potential inconsistency

Staff

StaffNo	SName	SAddress	Position	Bno	Salary
1001	John White	19 Taylor Street, London	Manager	B5	€30,000.00
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	B3	€12,000.00
1003	David Ford	63 Ashby Street, Glasgow	Deputy	B3	€18,000.00
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	B7	€9,000.00
1005	Susan Brand	5 Gt Western Road, Glasgow	Manager	B3	€24,000.00
1006	Julie Lee	28 Malvern Street, Kilburn	Assistant	B5	€9,000.00

Branch

Bno	BAddress	Tel_No
B3	163 Main Street, Glasgow	0141-339-2178
B5	22 Deer Road, London	0171-886-1212
B7	16 Argyll Street, Glasgow	01224-67125

Insert a branch that currently has no members of staff into the StaffBranch relation

To do this you must enter NULL in the attributes of staff (since no staff details are available).

NULL Values

Staff_Branch

Staff_No	SName	SAddress	Position	Salary	Branch_No	Baddress	Tel_No
1001	John White	19 Taylor Street, London	Manager	€30,000.00	B5	22 Deer Road, London	0171-886-1212
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	€12,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1003	David Ford	63 Ahsby Street, Glasgow	Deputy	€18,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	€9,000.00	B7	16 Argyll Street, Aberdeen	01224-67111
1005	Susan Brand	5 Gt Western Rd, Glasgow	Manager	€24,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1006	Julie Lee	29 Malvern Street, Kilburn	Assistant	€9,000.00	B5	22 Deer Road, London	0171-886-1212
					B6	12 Mayfield Ave, London	0171-779-1542

Since Staff_No is the primary key, this is not allowed (Primary key may not be NULL).

The individual ***Staff*** and ***Branch*** relations avoid this problem.

Staff

StaffNo	SName	SAddress	Position	Bno	Salary
1001	John White	19 Taylor Street, London	Manager	B5	€30,000.00
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	B3	€12,000.00
1003	David Ford	63 Ashby Street, Glasgow	Deputy	B3	€18,000.00
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	B7	€9,000.00
1005	Susan Brand	5 Gt Western Road, Glasgow	Manager	B3	€24,000.00
1006	Julie Lee	28 Malvern Street, Kilburn	Assistant	B5	€9,000.00

Branch

Bno	BAddress	Tel_No
B3	163 Main Street, Glasgow	0141-339-2178
B5	22 Deer Road, London	0171-886-1212
B7	16 Argyll Street, Glasgow	01224-67125
B6	12 Mayfield Ave, London	0171-779-1542

Delete Anomalies

If we remove a staff member from the StaffBranch relation we also remove information about the branch at which they work.

If the staff member happened to be the ***last*** member at this branch, we lose all details of that branch from the database

Removing details for customer 1004 (Mary Howe) also removes the last remaining entry for branch B7.

Staff_Branch

Staff_No	SName	SAddress	Position	Salary	Branch_No	Baddress	Tel_No
1001	John White	19 Taylor Street, London	Manager	€30,000.00	B5	22 Deer Road, London	0171-886-1212
1002	Ann Beech	81 George Street, Glasgow	Snr Asst	€12,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1003	David Ford	63 Ahsby Street, Glasgow	Deputy	€18,000.00	B3	63 Main St, Glasgow	0141-339-2178
1004	Mary Howe	2 Elm Place, Aberdeen	Assistant	€9,000.00	B7	16 Argyll Street, Aberdeen	01224-67113
1005	Susan Brand	5 Gt Western Rd, Glasgow	Manager	€24,000.00	B3	163 Main Street, Glasgow	0141-339-2178
1006	Julie Lee	29 Malvern Street, Kilburn	Assistant	€9,000.00	B5	22 Deer Road, London	0171-886-1212

Modification Anomalies

Change the telephone number for branch B3 in the StaffBranch relation

- We must update the rows of all staff located at branch B3
- If some of the rows are not updated, this results in inconsistent data

Conclusions

The relations Staff and Branch have more desirable properties than the relation StaffBranch

We will see how normalisation can be used to derive such relations.

Refer to the PDF document ***Normalisation_Chapter*** on X:\ drive.

Functional Dependencies

Describes the relationship between attributes in a relation

If A and B are attributes of a relation R

B is ***functionally dependent*** on A (A->B)

if each value of A in R is associated with exactly one value of B in R

A -> B

A is said to be the ***determinant***

B is said to be the ***dependent***

Look at the dependencies of the StaffBranch relation:

StaffNo -> Sname, Saddr, Position, Salary, Bno, Baddr, TelNo

Bno -> Baddr, TelNo
Baddr -> Bno, TelNo
TelNo -> Bno, Baddr

Un-normalised Form

A relation which contains one or more repeating groups

i.e. attribute values are non-atomic

This also means that the record length varies.

1NF

A relation is in **1NF** if

- it contains no repeating groups
- all non-key attributes are functionally dependent on the primary key

2NF

A relation is in **2NF** if

- it is in 1NF
- All non-key attributes are ***fully*** functionally dependent on the primary key

B is ***fully*** functionally dependent on A if:

B is functionally dependent on A and not any subset of A

B is ***partially*** dependent on A if some attribute can be removed from A and the dependency still holds

Applies only to relations with composite keys

3NF

A relation is in **3NF** if

- it is in 2NF
- Non-key attributes are not transitively dependent on the primary key

If A, B and C are attributes of a relation

and

A->B and B->C

then C is ***transitively*** dependent on A via B

The StaffBranch relation contains the following dependencies:

StaffNo -> Bno

Bno -> Baddr

Clearly there is a transitive dependency

Staffno -> Baddr (via Bno)

Example

Consider a Real Estate system which maintains property rental details.

A property has an owner and is rented to a customer.

A customer can rent more than one property but not simultaneously

A customer never rents the same property twice

Such a relation might contain the following attributes:

CustNo, Cname
PropNo, Paddr
RentStart, RentFinish, Rent
OwnerNo, Oname

Show the data as a set of normalised (3NF) relations