DATABASE NORMALIZATION

Normalization: process of efficiently organizing data in the DB.

↓

RELATIONS
(attributes grouped together)

↓

Accurate representation of data, relationships and constraints.

Goal: - Eliminate redundant data in a DB.
- Ensure data dependencies make sense.

Guidelines for ensuring that DBs are normalized → *normal forms*: 1NF, 2NF, 3NF, BCNF.

↓

Normalization: series of tests on a relation to determine whether it satisfies or violates the requirements of a normal form.

Note: meet practical business requirements.

↓

Normalization: A technique for producing a set of relations with desirable properties, given the data requirements of an enterprise.

Reason for normalization: to prevent possible corruption of DB stemming from update anomalies (insertion, deletion, modification).

EXAMPLE: REDUNDANCY ANOMALIES

STAFF RELATION

Staff_No	*SName*	*SAddress*	*Position*	*Salary*	*Branch_No*
SL21	John White	19 Taylor Street, London	Manager	30000	B5
SG37	Ann Beech	81 George Street, Glasgow	Snr Asst	12000	B3
SG14	David Ford	63 Ashby Street, Glasgow	Deputy	18000	B3
SA9	Mary Howe	2 Elm Place, Aberdeen	Assistant	9000	B7
SG5	Susan Brand	5 Gt Western Road, Glasgow	Manager	24000	B3
SL41	Julie Lee	28 Malvern Street, Kilburn	Assistant	9000	B5

BRANCH RELATION

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

Figure 6.1 Staff and Branch relations.

STAFF_BRANCH RELATION

Staff_No	*SName*	*SAddress*	*Position*	*Salary*	*Branch_No*	*BAddress*	*Tel_No*
SL21	John White	19 Taylor Street, London	Manager	30000	B5	22 Deer Road, London	0171-886-1212
SG37	Ann Beech	81 George Street, Glasgow	Snr Asst	12000	B3	163 Main Street, Glasgow	0141-339-2178
SG14	David Ford	63 Ashby Street, Glasgow	Deputy	18000	B3	163 Main Street, Glasgow	0141-339-2178
SA9	Mary Howe	2 Elm Place, Aberdeen	Assistant	9000	B7	16 Argyll Street, Aberdeen	01224-67111
SG5	Susan Brand	5 Gt Western Rd, Glasgow	Manager	24000	B3	163 Main Street, Glasgow	0141-339-2178
SL41	Julie Lee	28 Malvern Street, Kilburn	Assistant	9000	B5	22 Deer Road, London	0171-886-1212

FUNCTIONAL DEPENDENCIES

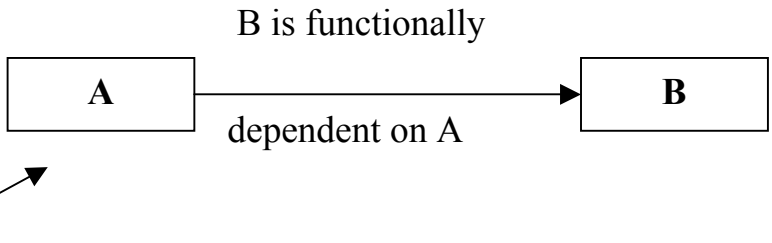
Normalization: a formal method that identifies relations based on their primary key and the *functional dependencies* among their attributes.

↓

Constraint between attributes.

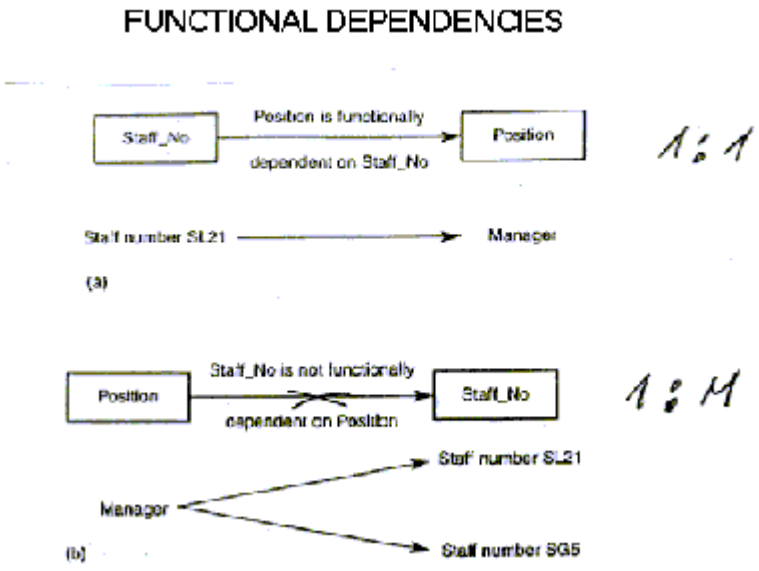
Functional dependency: Describes the relationship between attributes in a relation. If A and B are attributes of a relation R, B is functionally dependent on A (den. A → B), if

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

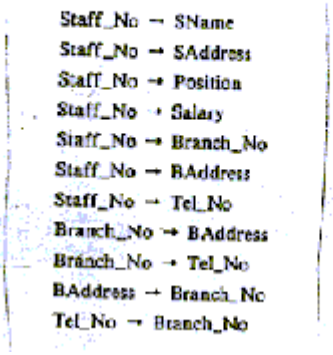


<u>Determinant</u>: attribute or set of attributes on the left hand side of the arrow.

<u>Identify</u> the <u>candidate key</u> for a relation: recognise the attribute (group of attributes) that uniquely identifies each row in a relation. All of the attributes that are not part of the primary key (non-primary key attributes) should be functionally dependent on the key.



FUNCTIONAL DEPENDENCIES OF STAFF_BRANCH RELATION



ALTERNATIVE FORMAT

Staff_No → SName, SAddress, Position, Salary, Branch_No, BAddress, Tel_No
Branch_No → BAddress, Tel_No
BAddress → Branch_No
Tel_No → Branch_No

PROCESS OF NORMALIZATION

Unnormalized form (UNF): A table that contains one or more repeating groups.

Repeating group: an attribute or group of attributes within a table that occurs with multiple values for a single occurrence of the nominated key attributes of that table.

INFORMATION HELD AS A FORM

Page 1 **DreamHome** Customer Rental Details Date 7-Oct-95

Customer Name John Kay Customer Number CR76

Property Number	Property Address	Rent Start	Rent Finish	Rent	Owner Number	Owner Name
PG4	6 Lawrence Street, Glasgow	1-Jul-93	31-Aug-95	350	CO40	Tina Murphy
PG16	5 Novar Drive, Glasgow	1-Sep-95	1-Sep-96	450	CO93	Tony Shaw

Figure 6.5
Dreamhome Customer Rental Details form.

CREATE TABLE

REPEATING GROUP

Nominated key

CUSTOMER RENTAL TABLE

Customer_No	*CName*	*Property_No*	*PAddress*	*RentStart*	*RentFinish*	*Rent*	*Owner_No*	*ONome*
CR76	John Kay	PG4	6 Lawrence Street, Glasgow	1-Jul-93	31-Aug-95	350	CO40	Tina Murphy
		PG16	5 Novar Drive, Glasgow	1-Sep-95	1-Sep-96	450	CO93	Tony Shaw
CR56	Alisa Stewart	PG4	6 Lawrence Street, Glasgow	1-Sep-92	10-June-93	350	CO40	Tina Murphy
		PG36	2 Manor Road, Glasgow	10-Oct-93	1-Dec-94	375	CO93	Tony Shaw
		PG16	5 Novar Drive, Glasgow	1-Jan-95	10-Aug-95	450	CO93	Tony Shaw

TABLE FORMAT IN UNF

<u>First normal form (1NF)</u>: A relation in which the intersection of each row and column contains one and only one value.

<u>UNF → 1NF</u>: remove repeating groups:

- **Entering appropriate data in the empty columns of rows.**
- **Placing repeating data along with a copy of the original key attribute in a separate relation. Identifying a primary key for each of the new relations.**

UNF -> 1NF

Method 1

Customer_Rental (Customer_No, Property_No, CName, PAddress, RentStart, RentFinish, Rent, Owner_No, OName)

CUSTOMER_RENTAL RELATION

Customer_No	*Property_No*	*CName*	*PAddress*	*RentStart*	*RentFinish*	*Rent*	*Owner_No*	*OName*
CR76	PG4	John Kay	6 Lawrence Street, Glasgow	1-Jul-93	31-Aug-95	350	CO40	Tina Murphy
CR76	PG16	John Kay	5 Novar Drive, Glasgow	1-Sep-95	1-Sep-96	450	CO93	Tony Shaw
CR56	PG4	Aline Stewart	6 Lawrence Street, Glasgow	1-Sep-92	10-Jun-93	350	CO40	Tina Murphy
CR56	PG36	Aline Stewart	2 Manor Road, Glasgow	10-Oct-93	1-Dec-94	375	CO93	Tony Shaw
CR56	PG16	Aline Stewart	5 Novar Drive, Glasgow	1-Jan-95	10-Aug-95	450	CO93	Tony Shaw

1NF

Method 2

CUSTOMER RELATION

Customer_No	*CName*
CR76	John Kay
CR56	Aline Stewart

1NF

PROP_RENTAL_OWNER RELATION

Customer_No	*Property_No*	*PAddress*	*RentStart*	*RentFinish*	*Rent*	*Owner_No*	*OName*
CR76	PG4	6 Lawrence Street, Glasgow	1-Jul-93	31-Aug-95	350	CO40	Tina Murphy
CR76	PG16	5 Novar Drive, Glasgow	1-Sep-95	1-Sep-96	450	CO93	Tony Shaw
CR56	PG4	6 Lawrence Street, Glasgow	1-Sep-92	10-Jun-93	350	CO40	Tina Murphy
CR56	PG36	2 Manor Road, Glasgow	10-Oct-93	1-Dec-94	375	CO93	Tony Shaw
CR56	PG16	5 Novar Drive, Glasgow	1-Jan-95	10-Aug-95	450	CO93	Tony Shaw

Customer (Customer_No, CName)

Prop_Rental_Owner (Customer_No, Property_No, PAddress, RentStart, RentFinish, Rent, Owner_No, OName)

Full functional dependency: If A and B are attributes of a relation, B is fully functionally dependent on A if B is functionally dependent on A, but not any proper subset of A.

A → B is partially dependent if there is some attribute that can be removed from A and the dependency still holds.

Ex.

Staff_No, SName → Branch_No **partial**

Staff_No → Branch_No **full**

<u>Second normal form (2NF)</u>: A relation that is in 1NF and every non-primary key attribute is fully functionally dependent on the primary key.

Note: applies to relations with composite keys (primary key composed of two or more attributes). A relation with a single attribute primary key is in at least 2NF.

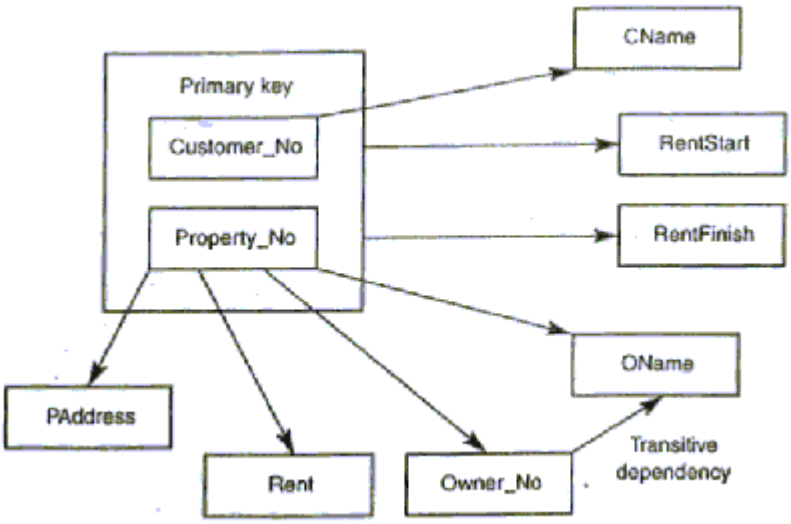
<u>1NF → 2NF</u>: remove partial dependencies: the functionally dependent attributes are removed from the relation by placing them in a new relation along with a copy of their determinant.

CUSTOMER_RENTAL RELATION

Customer_No	*Property_No*	*CName*	*PAddress*	*RentStart*	*RentFinish*	*Rent*	*Owner_No*	*OName*
CR76	PG4	John Kay	6 Lawrence Street, Glasgow	1-Jul-93	31-Aug-95	350	CO40	Tina Murphy
CR76	PG16	John Kay	5 Novar Drive, Glasgow	1-Sep-95	1-Sep-96	450	CO93	Tony Shaw
CR56	PG4	Aline Stewart	6 Lawrence Street, Glasgow	1-Sep-92	10-Jun-93	350	CO40	Tina Murphy
CR56	PG36	Aline Stewart	2 Manor Road, Glasgow	10-Oct-93	1-Dec-94	375	CO93	Tony Shaw
CR56	PG16	Aline Stewart	5 Novar Drive, Glasgow	1-Jan-95	10-Aug-95	450	CO93	Tony Shaw

FUNCTIONAL DEPENDENCIES IN CUSTOMER_RENTAL RELATION

Customer_No, Property_No → RentStart, RentFinish
Customer_No → CName
Property_No → PAddress, Rent, Owner_No, OName
Owner_No → OName



CUSTOMER RELATION

Customer_No	*CName*
CR76	John Kay
CR56	Aline Stewart

2NF

RENTAL RELATION

Customer_No	*Property_No*	*RentStart*	*RentFinish*
CR76	PG4	1-Jul-93	31-Aug-95
CR76	PG16	1-Sep-95	1-Sep-96
CR56	PG4	1-Sep-92	10-Jun-93
CR56	PG36	10-Oct-93	1-Dec-94
CR56	PG16	1-Jan-95	10-Aug-95

2NF

PROPERTY_OWNER RELATION

Property_No	*PAddress*	*Rent*	*Owner_No*	*OName*
PG4	6 Lawrence Street, Glasgow	350	CO40	Tina Murphy
PG16	5 Novar Drive, Glasgow	450	CO93	Tony Shaw
PG36	2 Manor Road, Glasgow	375	CO93	Tony Shaw

2NF

Customer (Customer_No, CName)
Rental (Customer_No, Property_No, RentStart, RentFinish)
Property_Owner (Property_No, PAddress, Rent, Owner_No, OName)

Transitive dependency: A condition where A, B and C are attributes of a relation such that if A → B and B → C, then C is transitively dependent on A via B (provided that A is not functionally dependent on B or C).

Third normal form (3NF): A relation that is in 1NF and 2NF, and in which no non-primary key attribute is transitively dependent on the primary key.

2NF → 3NF: remove transitive dependencies: the transitively dependent attributes are removed from the relation by placing them in a new relation along with a copy of their determinant.

FUNCTIONAL DEPENDENCIES

Customer Relation
Customer_No → CName
Rental Relation
Customer_No, Property_No → RentStart, RentFinish
Property_Owner Relation
Property_No → PAddress, Rent, Owner_No, OName
Owner_No → OName TRANSITIVE

PROPERTY_FOR_RENT RELATION

Property_No	*PAddress*	*Rent*	*Owner_No*
PG4	6 Lawrence Street, Glasgow	350	CO40
PG16	5 Novar Drive, Glasgow	450	CO93
PG36	2 Manor Road Glasgow	375	CO93

OWNER RELATION

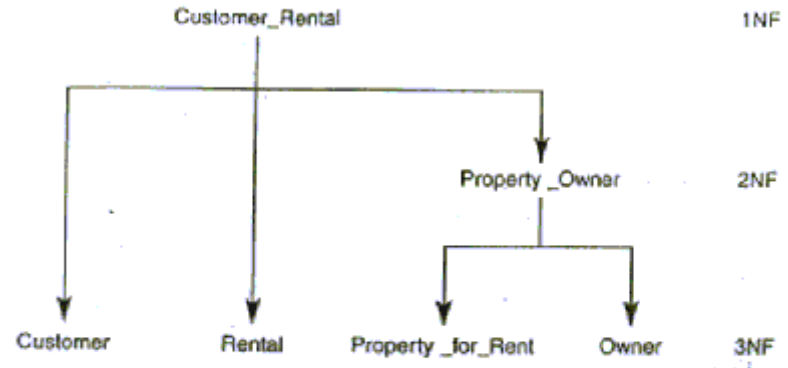
Owner_No	*OName*
CO40	Tina Murphy
CO93	Tony Shaw

3NF

Property_for_Rent (Property_No, PAddress, Rent, Owner_No)
Owner (Owner_No, OName)

The normalization process decomposes the original relation using a series of relation algebra projections. This result in a nonloss (lossless) decomposition, which is reversible using the natural join operation (primary key / foreign key mechanism).

NORMALIZATION DECOMPOSES THE ORIGINAL RELATION



The resulting 3NF relations have the form:

Customer	(Customer_No, CName)
Rental	(Customer_No, Property_No, RentStart, RentFinish)
Property_for_Rent	(Property_No, PAddress, Rent, Owner_No)
Owner	(Owner_No, OName)

CUSTOMER RELATION

Customer_No	*CName*
CR76	John Kay
CR56	Aline Stewart

RENTAL RELATION

Customer_No	*Property_No*	*RentStart*	*RentFinish*
CR76	PG4	1-Jul-93	31-Aug-95
CR76	PG16	1-Sep-95	1-Sep-96
CR56	PG4	1-Sep-92	10-Jun-93
CR56	PG36	10-Oct-93	1-Dec-94
CR56	PG16	1-Jan-95	10-Aug-95

PROPERTY_FOR_RENT RELATION

Property_No	*PAddress*	*Rent*	*Owner_No*
PG4	6 Lawrence Street, Glasgow	350	CO40
PG16	5 Novar Drive, Glasgow	450	CO93
PG36	2 Manor Road Glasgow	375	CO93

OWNER RELATION

Owner_No	*OName*
CO40	Tina Murphy
CO93	Tony Shaw

<u>Boyce-Codd Normal Form (BCNF)</u>: A relation is in BCNF if and only if every determinant is a candidate key.

<u>Notes</u>:

BCNF is a stronger form of 3NF

BCNF => 3NF

3NF ≠> BCNF

Violation of BCNF happen under specific conditions:

- **A relation contains two (or more) composite candidate keys,**
- **which overlap and share at least one attribute in common.**

<u>Transformation to BCNF:</u> remove violating functional dependencies by placing them in a new relation.

EXAMPLE

CLIENT_INTERVIEW RELATION

Client_No	*Interview_Date*	*Interview_Time*	*Staff_No*	*Room_No*
CR76	13-May-95	10.30	SG5	G101
CR56	13-May-95	12.00	SG5	G101
CR74	13-May-95	12.00	SG37	G102
CR56	1-Jul-95	10.30	SG5	G102

A B C D E

The Client_Interview relation has the following functional dependencies:

Client_No, Interview_Date → Interview_Time, Staff_No, Room_No
Staff_No, Interview_Date, Interview_Time → Client_No
Staff_No, Interview_Date → Room_No

INTERVIEW RELATION

Client_No	*Interview_Date*	*Interview_Time*	*Staff_No*
CR76	13-May-95	10.30	SG5
CR56	13-May-95	12.00	SG5
CR74	13-May-95	12.00	SG37
CR56	1-Jul-95	10.30	SG5

STAFF_ROOM RELATION

Staff_No	*Interview_Date*	*Room_No*
SG5	13-May-95	G101
SG37	13-May-95	G102
SG5	1-Jul-95	G102

Interview (<u>Client_No</u>, <u>Interview_Date</u>, Interview_Time, Staff_No)
Staff_Room (<u>Staff_No</u>, <u>Interview_Date</u>, Room_No)