

COMPS321F

Advanced Database and Data Warehousing

Lecture 2

NORMALIZATION PART I

Dr. Wyman Wang

School of Science and Technology

Objectives

The potential problems associated with redundant data in base relations.

The concept of functional dependency, which describes the relationship between attributes.

How to identify functional dependencies for a given relation.

How functional dependencies identify the primary key of a relation.

Normalization

A technique of organizing the data in the database.

A systematic approach of decomposing tables to eliminate data redundancy (unnecessary repetition) and undesirable characteristics like update anomalies.

Purpose of Normalization

The benefits of using a database that has a suitable set of relations are that the database will:

- be easy for the user to access and maintain the data;
- take up minimal storage space on the computer.

Purpose of Normalization

Characteristics of a suitable set of relations include:

- the minimal number of attributes necessary to support the data requirements of the enterprise;
- attributes with a close logical relationship are found in the same relation;
- minimal redundancy with each attribute represented only once, except for attributes that form all or part of foreign keys.

Data Redundancy and Update Anomalies

Major aim of relational database design is to group attributes into relations to minimize data redundancy.

Potential benefits for implemented database include:

- Updates to the data stored in the database are achieved with a minimal number of operations thus reducing the opportunities for data inconsistencies.
- Reduction in the file storage space required by the base relations thus minimizing costs.

Data Redundancy and Update Anomalies

Problems associated with data redundancy are illustrated by comparing the Staff and Branch relations with the StaffBranch relation.

Data Redundancy and Update Anomalies

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Branch

branchNo	bAddress
B005	22 Deer Rd, London
B007	16 Argyll St, Aberdeen
B003	163 Main St, Glasgow

Staff Branch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Data Redundancy and Update Anomalies

StaffBranch relation has redundant data; the address of a branch is repeated for every member of staff.

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Data Redundancy and Update Anomalies

In contrast, the branch information appears only once for each branch in the Branch relation, and only the branch number (branchNo) is repeated in the Staff relation to represent where each member of staff is located.

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Branch

branchNo	bAddress
B005	22 Deer Rd, London
B007	16 Argyll St, Aberdeen
B003	163 Main St, Glasgow

Data Redundancy and Update Anomalies

Relations that contain redundant information may potentially suffer from update anomalies.

Types of update anomalies include

- Insertion
- Deletion
- Modification

Update Anomalies

Insertion

- insert staff: need to include branch address
- insert new branch: need to include staff information

Deletion

- delete last staff of the branch: branch information lost

Modification

- change a branch information: need to change all tuples having that branch

Staff Branch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Lossless-join and Dependency Preservation Properties

Two important properties associated with decomposition of a larger relation into smaller relations.

- Lossless-join property enables us to find any instance of the original relation from corresponding instances in the smaller relations.
- Dependency preservation property enables us to enforce a constraint on the original relation by enforcing some constraints on each of the smaller relations.

Functional Dependency

Important concept associated with normalization.

Functional dependency describes relationship between attributes.

For example, if A and B are attributes of relation R, B is functionally dependent on A (denoted $A \rightarrow B$), if each value of A in R is associated with exactly one value of B in R.

Characteristics of Functional Dependency

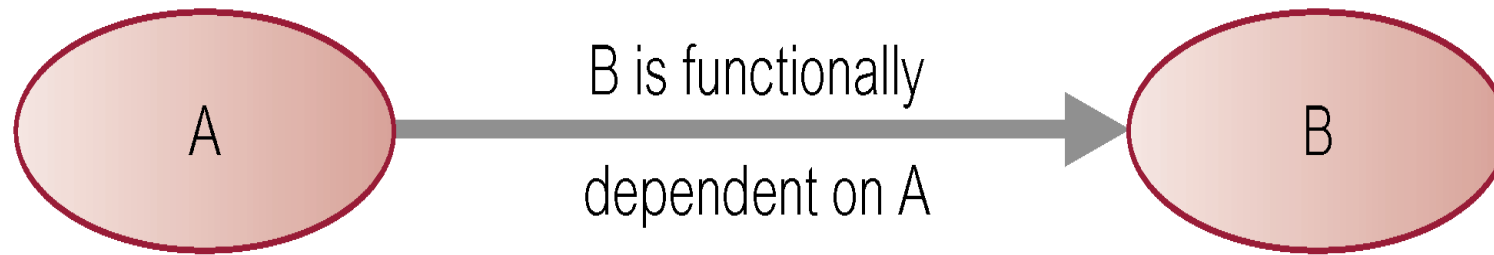
Main characteristics of functional dependency used in normalization:

- There is a one-to-one relationship between the attribute(s) on the left-hand side (determinant) and those on the right-hand side of a functional dependency.
- Holds for all time.
- The determinant has the minimal number of attributes necessary to maintain the dependency with the attribute(s) on the right hand-side.

Characteristics of Functional Dependency

Property of the meaning or semantics of the attributes in a relation.

Diagrammatic representation.



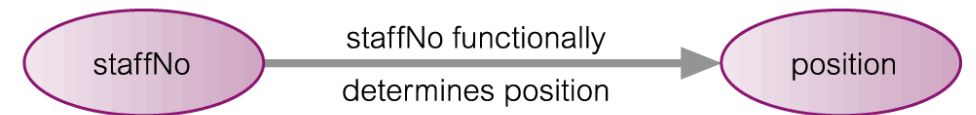
The determinant of a functional dependency refers to the attribute or group of attributes on the left-hand side of the arrow.

Example - Functional Dependency

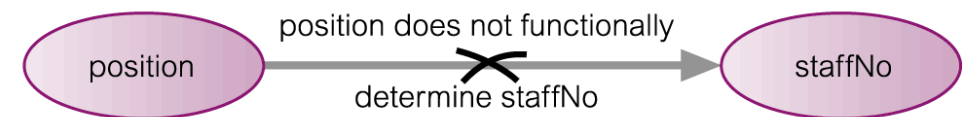
Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

- staffNo → position ✓
- position → staffNo ✗



Staff number SL21 → Manager
(a)



Manager → Staff number SL21
Manager → Staff number SG5
(b)

Example - Functional Dependency that holds for all Time

Consider the values shown in staffNo and sName attributes of the Staff relation.

Based on sample data, the following functional dependency appear to hold.

staffNo \rightarrow sName



sName \rightarrow staffNo



Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Example - Functional Dependency that holds for all Time

However, the only functional dependency that remains **true for all possible values** for the staffNo and sName attributes of the Staff relation is:

staffNo \rightarrow sName ✓
sName \rightarrow staffNo ✗

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Example - Functional Dependency that holds for all Time

The only functional dependency that remains **true for all possible values** for the branchNo and bAddress attributes of the Branch relation is:

branchNo \rightarrow bAddress

bAddress \rightarrow branchNo

(this one will usually not be selected as using branchNo is more convenient within the organization.)

Branch

branchNo	bAddress
B005	22 Deer Rd, London
B007	16 Argyll St, Aberdeen
B003	163 Main St, Glasgow

Full Functional Dependency

Determinant should have the **minimal** number of attributes necessary to maintain the functional dependency with the attribute(s) on the right hand-side.

This requirement is called **full functional dependency**.

Full functional dependency indicates that 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 on** any proper subset of A.

Example - Full Functional Dependency

Exists in the Staff relation:

$\text{staffNo} \rightarrow \text{sName, position, salary, branchNo}$

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

True - each value of staffNo is associated with a single value of (sName, position, salary, branchNo).

Example - Full Functional Dependency

This is NOT a full functional dependency:

$\text{staffNo}, \text{sName} \rightarrow \text{branchNo}$

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Even though it is true for each value of (staffNo, sName) is associated with a single value of branchNo.

However, branchNo is also functionally dependent on a **subset** of (staffNo, sName), namely **staffNo**, i.e.

$\text{staffNo} \rightarrow \text{branchNo}$

Partial Functional Dependency

Partial functional dependency indicates that if A and B are attributes of a relation, B is partially functionally dependent on A, if B is functionally dependent on A, but **also on** any proper subset of A.

Example - Partial Functional Dependency

Exists in the Staff relation

$\text{staffNo}, \text{sName} \rightarrow \text{branchNo}$

branchNo is also functionally dependent on a **subset** of (staffNo, sName), namely **staffNo** i.e.

$\text{staffNo} \rightarrow \text{branchNo}$

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Question

Is the following a full or partial functional dependency?

branchNo, position → salary

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Answer

This is a **full** functional dependency

branchNo, position \rightarrow salary

salary is **NOT** functionally dependent on a subset of (branchNo, position), i.e.

branchNo \rightarrow salary **×**

position \rightarrow salary **×**

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Question

Is the following a full or partial functional dependency?

staffNo, position → salary

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Answer

This is a **partial** functional dependency

staffNo, position \rightarrow salary

salary is also functionally dependent on a **subset** of (staffNo, position), i.e.

staffNo \rightarrow salary

Staff

staffNo	sName	position	salary	branchNo
SL21	John White	Manager	30000	B005
SG37	Ann Beech	Assistant	12000	B003
SG14	David Ford	Supervisor	18000	B003
SA9	Mary Howe	Assistant	9000	B007
SG5	Susan Brand	Manager	24000	B003
SL41	Julie Lee	Assistant	9000	B005

Transitive Dependency

Important to recognize a transitive dependency because its existence in a relation can potentially cause update anomalies.

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

Example - Transitive Dependency

Consider functional dependencies in the StaffBranch relation

- staffNo \rightarrow sName, position, salary, branchNo, bAddress
- branchNo \rightarrow bAddress

bAddress is transitively dependent on staffNo via branchNo.

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Identifying a set of functional dependencies for the StaffBranch relation

Examine semantics of attributes in StaffBranch relation. Assume that position and branch determine a member of staff's salary.

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Version 1

With sufficient information available, identify the functional dependencies for the StaffBranch relation as:

staffNo \rightarrow sName, position, salary, branchNo, bAddress

branchNo \rightarrow bAddress

branchNo, position \rightarrow salary

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

Version 2

staffNo → sName, position, salary, branchNo, bAddress

branchNo → bAddress

bAddress → branchNo

branchNo, position → salary

bAddress, position → salary

StaffBranch

staffNo	sName	position	salary	branchNo	bAddress
SL21	John White	Manager	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Assistant	12000	B003	163 Main St, Glasgow
SG14	David Ford	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Mary Howe	Assistant	9000	B007	16 Argyll St, Aberdeen
SG5	Susan Brand	Manager	24000	B003	163 Main St, Glasgow
SL41	Julie Lee	Assistant	9000	B005	22 Deer Rd, London

[Note] Version 1 and Version 2 are both valid, Version 1 is more preferable! More discussions in the next lecture.

Reference

Chapter 14 of Connolly, T and Begg, C, Database Systems: A practical Approach to Design, Implementation, and Management (6th ed.), Boston: Pearson Education.