

CSCI235 Database Systems

Functional Dependencies

Dr Janusz R. Getta

School of Computing and Information Technology -
University of Wollongong

Functional dependencies

Outline

Functional dependency ? What is it ?

Functional dependencies versus classes of objects

Functional dependencies versus associations

Derivations of functional dependencies

Armstrong axioms

Other inference rules

Using inference rules

Functional dependency ? What is it ?

Let $R = (A_1, \dots, A_n)$ be a relational schema (a header of relational table) and let X, Y be the nonempty subsets of R

We say that a functional dependency $X \rightarrow Y$ is valid in a relational schema R if

for any contents of a relational table R , it is not possible that R has two rows that agree in the components for all attributes in a set X yet disagree on one or more component for the attributes in a set Y

Examples

- A warehouse is located at exactly one address: $\text{warehouse} \rightarrow \text{address}$
- An address is related to exactly one warehouse: $\text{address} \rightarrow \text{warehouse}$
- At a warehouse, the parts of the same sort have only one total quantity: $\text{warehouse, part} \rightarrow \text{quantity}$
- A car has one owner: $\text{registration} \rightarrow \text{driving license}$
- A student has one first name and one last name and one date of birth: $\text{student-number} \rightarrow \text{first-name, last-name-date-of-birth}$

[In HTML view press 'p' to see the lecture notes](#)

[TOP](#)

Created by Janusz R. Getta, CSCI235 Database Systems, Autumn 2026

3/27

Functional dependency ? What is it ?

More examples

- An employee belongs to one department:
 $\text{employee-number} \rightarrow \text{department-name}$
- A manager manages one department: $\text{manager-number} \rightarrow \text{department-name}$
- An employee has one manager: $\text{employee-number} \rightarrow \text{manager-number}$
- A student enrolls a subject one time:
 $\text{student-number}, \text{subject-code} \rightarrow \text{enrolment-date}$
- An employee is located in one building in one office:
 $\text{employee-number} \rightarrow \text{building-number}, \text{office-number}$
- An office in a building hosts one employee:
 $\text{building-number}, \text{office-number} \rightarrow \text{employee-number}$
- An office in a building at a campus hosts one employee:
 $\text{campus-name}, \text{building-number}, \text{office-number} \rightarrow \text{employee-number}$
- A department has one manager: $\text{department-name} \rightarrow \text{manager-number}$
- A department is located in one building: $\text{department-name} \rightarrow \text{building-number}$
- A department has one manager and it is located in one building:
 $\text{department-name} \rightarrow \text{manager-number}, \text{building-number}$

Functional dependency ? What is it ?

How to discover the **functional dependencies** in a relational table ?

- Is it possible to discover the **functional dependencies** in a **relational schema** (a header of relational table) **R(A, B, C, D, E)** ?
- Of course it is impossible to do it because we do not know the **semantics** (the **meanings**) of the names: **R, A, B, C, D, E**
- To discover the **functional dependencies** in a relational table we must use the **semantics** of a **relational table name** and the **names of attributes**
- For example consider a relational schema (a header of relational table) **TRIP(rego#, licence#, tdate)** of a relational table that contains information about the **trips** made by the **drivers** (**licence#**) who used the **trucks** (**rego#**) on a given **day** (**tdate**)
- Can a truck be used only one time ? If yes then **rego# → tdate**
- Can a driver make only one trip ? If yes then **licence# → tdate**
- Can a driver use more than one truck ? If no then **licence# → rego#**
- Can a truck be used by more than one driver ? If no then **rego# → licence#**
- And so on ...

[In HTML view](#) press 'p' to see the [lecture notes](#)

[TOP](#)

Created by Janusz R. Getta, CSCI235 Database Systems, Autumn 2026

5/27

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Functional dependencies versus classes of objects

A class of objects **STUDENT**

STUDENT	
s#	ID1
fname	ID2
lname	ID2
dob	ID2
average	
language	[1..*]

validates (satisfies) the following functional dependencies:

$s\# \rightarrow \text{fname}$

$s\# \rightarrow \text{lname}$

$s\# \rightarrow \text{dob}$

$s\# \rightarrow \text{average}$

$\text{fname, lname, dob} \rightarrow s\#$

$\text{fname, lname, dob} \rightarrow \text{average}$

Functional dependencies versus classes of objects

The functional dependencies:

$s\# \rightarrow \text{fname}$

$s\# \rightarrow \text{lname}$

$s\# \rightarrow \text{dob}$

$s\# \rightarrow \text{average}$

are equivalent to a functional dependency

$s\# \rightarrow \text{fname, lname, dob, average}$

The functional dependencies

$\text{fname, lname, dob} \rightarrow s\#$

$\text{fname, lname, dob} \rightarrow \text{average}$

are equivalent to a functional dependency

$\text{fname, lname, dob} \rightarrow s\#, \text{average}$

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

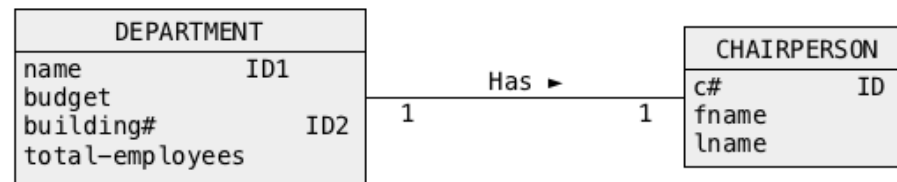
[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Functional dependencies versus associations

The classes of objects **DEPARTMENT** and **CHAIRPERSON** and association **Has**



validate (satisfy) the following functional dependencies:

$\text{name} \rightarrow \text{budget}, \text{building\#}, \text{total-employees}$

$\text{building\#} \rightarrow \text{name}, \text{budget}, \text{total-employees}$

$\text{c\#} \rightarrow \text{fname}, \text{lname}$

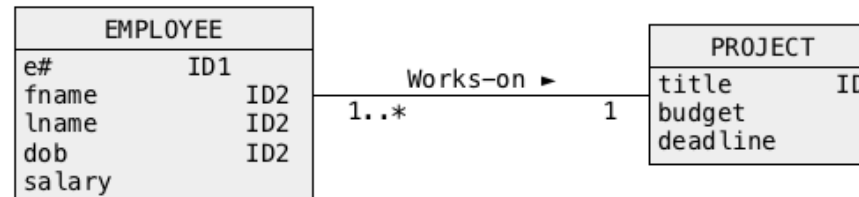
$\text{name} \rightarrow \text{c\#}, \text{fname}, \text{lname}$

$\text{building\#} \rightarrow \text{c\#}, \text{fname}, \text{lname}$

$\text{c\#} \rightarrow \text{name}, \text{building\#}, \text{budget}, \text{total-employees}$

Functional dependencies versus associations

The classes of objects **EMPLOYEE** and **PROJECT** and association **Works-on**



validate (satisfy) the following functional dependencies:

$e\# \rightarrow \text{fname, lname, dob, salary}$

$\text{fname, lname, dob} \rightarrow e\#, \text{salary}$

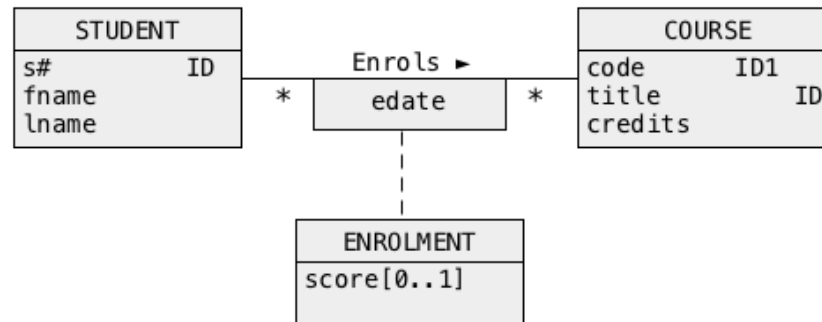
$\text{title} \rightarrow \text{budget, deadline}$

$e\# \rightarrow \text{title, budget, deadline}$

$\text{fname, lname, dob} \rightarrow \text{title, budget, deadline}$

Functional dependencies versus associations

The classes of objects **STUDENT** and **COURSE** and association **Enrols**



validate (satisfy) the following functional dependencies:

$s\# \rightarrow \text{fname}, \text{lname}$

$\text{code} \rightarrow \text{title}, \text{credits}$

$\text{title} \rightarrow \text{code}, \text{credits}$

$s\#, \text{code}, \text{edate} \rightarrow \text{score}$

$s\#, \text{title}, \text{edate} \rightarrow \text{score}$

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Derivations of functional dependencies

Consider a relational schema (a header of relational table)

`EMPLOYEE(e#, ename, department, address, chairperson)`

If $e\# \rightarrow \text{ename}$ and $e\# \rightarrow \text{department}$ then $e\# \rightarrow \text{ename, department}$

If $e\# \rightarrow \text{department}$ and $\text{department} \rightarrow \text{address}$ then $e\# \rightarrow \text{address}$

If $e\# \rightarrow \text{department}$ and $\text{department} \rightarrow \text{chairperson}$ then
 $e\# \rightarrow \text{chairperson}$

If $e\# \rightarrow \text{department}$ then $e\#, \text{ename} \rightarrow \text{department}$

If $e\#, \text{ename} \rightarrow \text{department}$ then $e\#, \text{ename, address} \rightarrow \text{department}$

It is always true that $e\# \rightarrow e\#$

Functional dependency $e\# \rightarrow e\#$ is called as a **trivial functional dependency**

It is always true that $e\#, \text{ename} \rightarrow e\#$

A functional dependency $e\#, \text{ename} \rightarrow e\#$ is also called as a **trivial functional dependency**

[In HTML view press 'p' to see the lecture notes](#)

[TOP](#)

Created by Janusz R. Getta, CSCI235 Database Systems, Autumn 2026

14/27

Derivations of functional dependencies

A **trivial functional dependency** is a functional dependency that is always true no matter what its left and right hand sides are

For example,

$e\# \rightarrow e\#$,

$\text{department} \rightarrow \text{department}$

$e\#, \text{ename} \rightarrow e\#$,

$e\#, \text{ename}, \text{department} \rightarrow e\#, \text{department}$,

and so on

Derivations of functional dependencies

Consider a relational schema $R(A, B, C)$

It is always true that $A \rightarrow A$

It is always true that $A, B \rightarrow A$

It is always true that $A, B, C \rightarrow A$

If $A \rightarrow B$ then $A, C \rightarrow B$

If $A \rightarrow B, C$ then $A \rightarrow B$ and $A \rightarrow C$

If $A \rightarrow B$ and $B \rightarrow C$ then $A \rightarrow C$

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Armstrong axioms

Let $R = (A_1, \dots, A_n)$ be a relational schema (a header of relational table)
and

let X, Y, Z be the nonempty subsets of $\{A_1, \dots, A_n\}$

(i) If $Y \subseteq X$ then $X \rightarrow Y$ (reflexivity axiom)

(ii) If $X \rightarrow Y$ then $X, Z \rightarrow Y, Z$ (augmentation axiom)

(iii) If $X \rightarrow Y$ and $Y \rightarrow Z$ then $X \rightarrow Z$ (transitivity axiom)

The axioms (i),(ii), and (iii) form a minimal and complete set of axioms

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Other inference rules

Let $R = (A_1, \dots, A_n)$ be a relational schema (a header of relational table)
and

let X, Y, Z be the nonempty subsets of $\{A_1, \dots, A_n\}$

If $X \rightarrow Y$ and $X \rightarrow Z$ then $X \rightarrow Y, Z$ (union rule)

If $X \rightarrow Y$ and $W, Y \rightarrow Z$ then $W, X \rightarrow Z$ (pseudotransitivity rule)

If $X \rightarrow Y$ and $Z \subseteq Y$ then $X \rightarrow Z$ (decomposition rule or reduce right hand side rule)

If $X \rightarrow Y$ then $X, Z \rightarrow Y$ (extend left hand side rule)

Functional dependencies

Outline

[Functional dependency ? What is it ?](#)

[Functional dependencies versus classes of objects](#)

[Functional dependencies versus associations](#)

[Derivations of functional dependencies](#)

[Armstrong axioms](#)

[Other inference rules](#)

[Using inference rules](#)

Using inference rules

Let $R = (A, B, C)$ be a relational schema

Given set of functional dependencies $F = \{A \rightarrow B, B \rightarrow C\}$ valid in R

Is it true that $A \rightarrow C$?

If $A \rightarrow B$ and $B \rightarrow C$ then application of **transitivity axiom** provides $A \rightarrow C$

Using inference rules

Let $R = (A, B, C)$ be a relational schema

Given set of functional dependencies $F = \{A \rightarrow B, C\}$ valid in R

Is it true that $A \rightarrow B$ and $A \rightarrow C$?

Reflexivity axiom provides $B, C \rightarrow C$

If $A \rightarrow B, C$ and $B, C \rightarrow C$ then **transitivity axiom** provides $A \rightarrow C$

Reflexivity axiom provides $B, C \rightarrow B$

If $A \rightarrow B, C$ and $B, C \rightarrow B$ then **transitivity axiom** provides $A \rightarrow B$

Using inference rules

Let $R = (A, B, C)$ be a relational schema

Given set of functional dependencies $F = \{A \rightarrow B, A \rightarrow C\}$ valid in R

Is it true that $A \rightarrow B, C$?

If $A \rightarrow B$ then **augmentation axiom** provides $A \rightarrow A, B$

If $A \rightarrow C$ then **augmentation axiom** provides $A, B \rightarrow B, C$

If $A \rightarrow A, B$ and $A, B \rightarrow B, C$ then **transitivity axiom** provides $A \rightarrow B, C$

Using inference rules

Let $R = (A, B, C)$ be a relational schema

Given set of functional dependencies $F = \{A \rightarrow B\}$ valid in R

Is it true that $A, C \rightarrow B$?

Reflexivity axiom provides $A, C \rightarrow A$

If $A, C \rightarrow A$ and $A \rightarrow B$ then **transitivity axiom** provides $A, C \rightarrow B$

Using inference rules

A relational schema **STUDENT**(s#, fname, lname, dob, average) validates (satisfies) the following functional dependencies:

$s\# \rightarrow \text{fname}$

$s\# \rightarrow \text{lname}$

$s\# \rightarrow \text{dob}$

$s\# \rightarrow \text{average}$

$\text{fname, lname, dob} \rightarrow s\#$

$\text{fname, lname, dob} \rightarrow \text{average}$

We proved that if $A \rightarrow B$ and $A \rightarrow C$ then $A \rightarrow B, C$

Hence,

$s\# \rightarrow \text{fname, lname, dob, average}$ and ...

$\text{fname, lname, dob} \rightarrow s\#, \text{average}$

Note, that both functional dependencies **cover** entire relational schema and **no other** functional dependencies that **do not cover** entire relational schema validate in the schema e.g. $\text{fname} \rightarrow s\#$

References

T. Connolly, C. Begg, Database Systems, A Practical Approach to Design, Implementation, and Management, Chapter 14.4 Functional Dependencies, Chapter 15.1 More on Functional Dependencies, Pearson Education Ltd, 2015