Some Normalization Examples

Dependencies: Definitions

- ***Multivalued Attributes*** (or ***repeating groups***): non-key attributes or groups of non-key attributes the values of which are not uniquely identified by (directly or indirectly) (not functionally dependent on) the value of the Primary Key (or its part).

STUDENT

Stud_ID	**Name**	**Course_ID**	**Units**
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Dependencies: Definitions

- ***Partial Dependency*** – when an non-key attribute is determined by a part, but not the whole, of a **COMPOSITE** primary key.

Partial Dependency

CUSTOMER

Cust_ID	**Name**	**Order_ID**
101	AT&T	1234
101	AT&T	156
125	Cisco	1250

Dependencies: Definitions

- ***Transitive Dependency*** – when a non-key attribute determines another non-key attribute.

Transitive Dependency

EMPLOYEE

<u>Emp_ID</u>	**F_Name**	**L_Name**	**Dept_ID**	**Dept_Name**
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

Normal Forms: Review

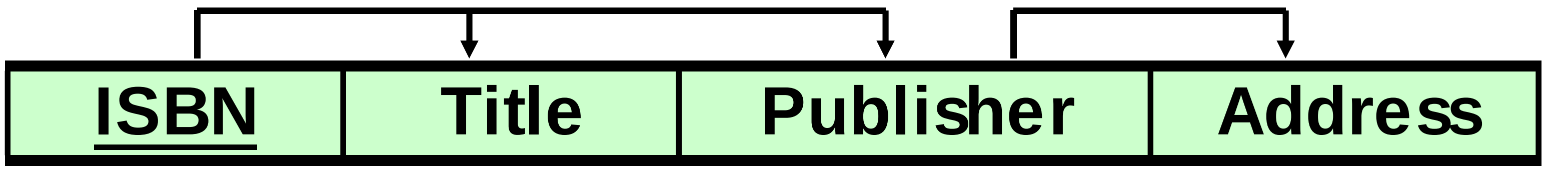
- Unnormalized – There are multivalued attributes or repeating groups
- 1 NF – No multivalued attributes or repeating groups.
- 2 NF – 1 NF plus no partial dependencies
- 3 NF – 2 NF plus no transitive dependencies

Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

All attributes are directly or indirectly determined by the primary key; therefore, the relation is at least in 1 NF

BOOK



Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

The relation is at least in 1NF.
There is no COMPOSITE primary key, therefore there can't be partial dependencies.
Therefore, the relation is at least in 2NF

BOOK

<u>ISBN</u>	Title	Publisher	Address

Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Addres

Publisher is a non-key attribute, and it determines Address, another non-key attribute. Therefore, there is a transitive dependency, which means that the relation is NOT in 3 NF.

BOOK

<u>ISBN</u>	Title	Publisher	Address

Example 1: Determine NF

- ISBN → Title
- ISBN → Publisher
- Publisher → Address

We know that the relation is at least in 2NF, and it is not in 3 NF. Therefore, we conclude that the relation is in 2NF.

BOOK

<u>ISBN</u>	Title	Publisher	Address

5

Example 1: Determine NF

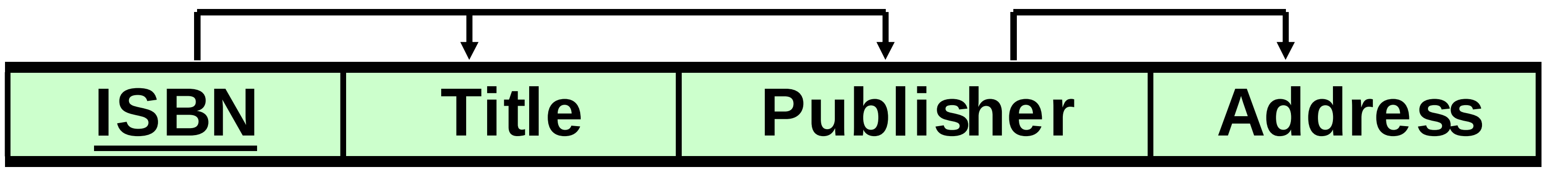
- ISBN → Title
- ISBN → Publisher
- Publisher → Address

In your solution you will write the following justification:

1) **No M/V attributes, therefore at least 1NF**
2) **No partial dependencies, therefore at least 2NF**
3) **There is a transitive dependency (Publisher → Address), therefore, not 3NF**

Conclusion: The relation is in 2NF

BOOK



Example 2: Determine NF

- Product_ID → Description

All attributes are directly or indirectly determined by the primary key; therefore, the relation is at least in 1 NF

ORDER

Order_No	Product_ID	Description

Example 2: Determine NF

- Product_ID → Description

The relation is at least in 1NF.
There is a COMPOSITE Primary Key (PK) (<u>Order_No, Product_ID</u>), therefore there can be partial dependencies. Product_ID, which is a part of PK, determines Description; hence, there is a partial dependency. Therefore, the relation is not 2NF. No sense to check for transitive dependencies!

ORDER

<u>Order_No</u>	<u>Product_ID</u>	Description

Example 2: Determine NF

- Product_ID → Description

We know that the relation is at least in 1NF, and it is not in 2 NF. Therefore, we conclude that the relation is in 1 NF.

ORDER

Order_No	Product_ID	Description

Example 2: Determine NF

- Product_ID → Description

In your solution you will write the following justification:
1) No M/V attributes, therefore at least 1NF
2) There is a partial dependency (Product_ID → Description), therefore not in 2NF
Conclusion: The relation is in 1NF

ORDER

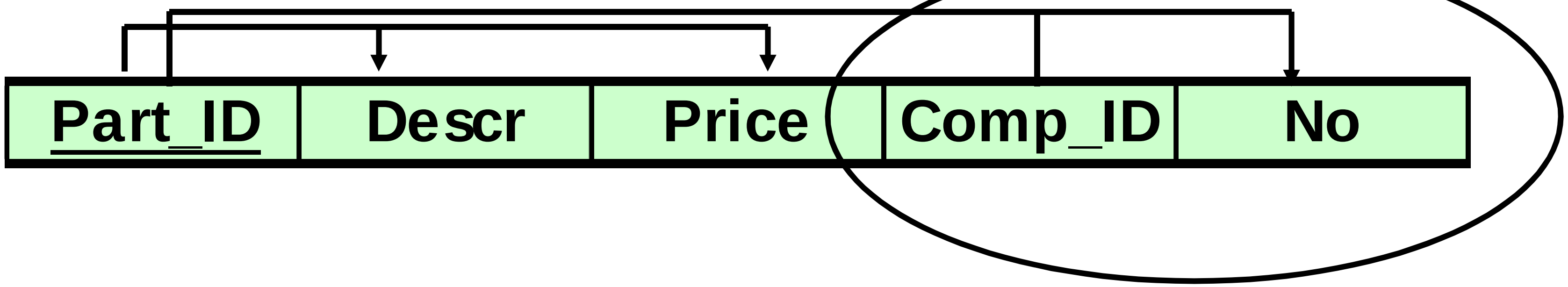
<u>Order_No</u>	<u>Product_ID</u>	Description

Example 3: Determine NF

- Part_ID → Description
- Part_ID → Price
- Part_ID, Comp_ID → No

Comp_ID and No are not determined by the primary key; therefore, the relation is NOT in 1 NF. No sense in looking at partial or transitive dependencies.

PART



Example 3: Determine NF

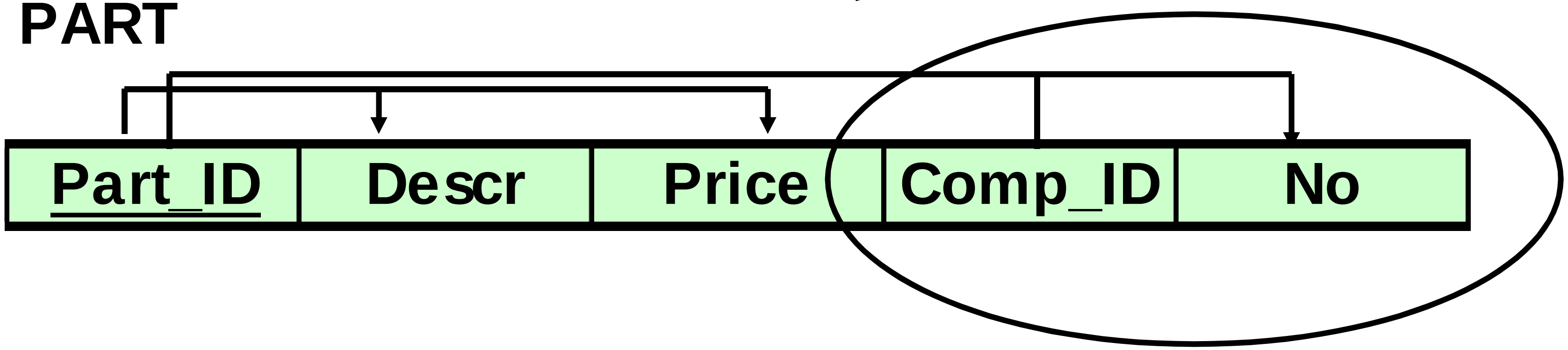
- Part_ID → Descriptio
- Part_ID → Price
- Part_ID, Comp_ID →

In your solution you will write the following justification:

1) There are M/V attributes; therefore, not 1NF

Conclusion: The relation is not normalized.

PART



Bringing a Relation to 1NF

STUDENT

<u>**Stud_ID**</u>	**Name**	**Course_ID**	**Units**
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 1NF

- Option 1: Make a determinant of the repeating group (or the multivalued attribute) a part of the primary key.

Composite Primary Key

STUDENT

Stud_ID	**Name**	**Course_ID**	**Units**
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 1NF

- Option 2: Remove the entire repeating group from the relation. Create another relation which would contain all the attributes of the repeating group, plus the primary key from the first relation. In this new relation, the primary key from the original relation and the determinant of the repeating group will comprise a primary key.

STUDENT

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 1NF

STUDENT

Stud_ID	Name
101	Lennon
125	Jonson

STUDENT_COURSE

Stud_ID	Course	Units
101	MSI 250	3
101	MSI 415	3
125	MSI 331	3

Bringing a Relation to 2NF

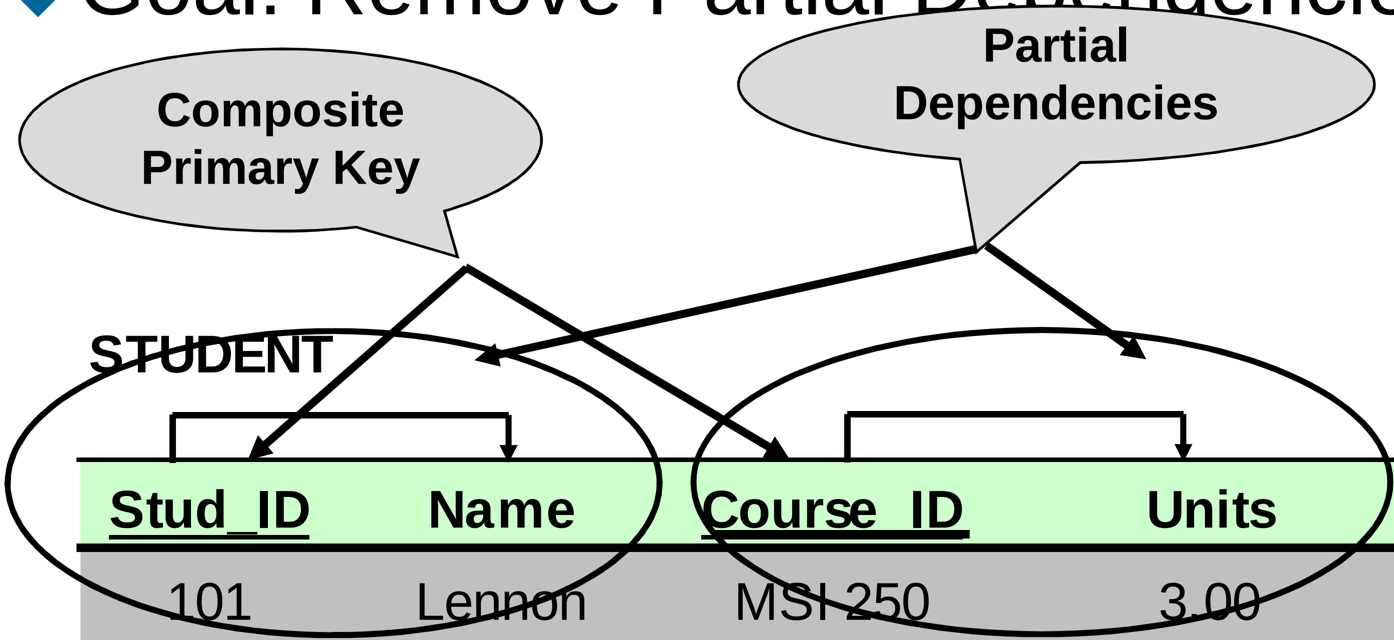
Composite Primary Key

STUDENT

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

- Goal: Remove Partial Dependencies



Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

- Remove attributes that are dependent from the part but not the whole of the primary key from the original relation. For each partial dependency, create a new relation, with the corresponding part of the primary key from the original as the primary key.

STUDENT

Stud_ID	**Name**	**Course_ID**	**Units**
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

Bringing a Relation to 2NF

5

CUSTOMER

Stud_ID	Name	Course_ID	Units
101	Lennon	MSI 250	3.00
101	Lennon	MSI 415	3.00
125	Johnson	MSI 331	3.00

STUDENT_COURSE

Stud_ID	Course_ID
101	MSI 250
101	MSI 415
125	MSI 331

STUDENT

Stud_ID	Name
101	Lennon
101	Lennon
125	Johnson

COURSE

Course_ID	Units
MSI 250	3.00
MSI 415	3.00
MSI 331	3.00

Bringing a Relation to 3NF

- Goal: Get rid of transitive dependencies.

Transitive Dependency

EMPLOYEE

<u>Emp_ID</u>	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

5

Bringing a Relation to 3NF

- Remove the attributes, which are dependent on a non-key attribute, from the original relation. For each transitive dependency, create a new relation with the non-key attribute which is a determinant in the transitive dependency as a primary key, and the dependent non-key attribute as a dependent.

EMPLOYEE

<u>**Emp_ID**</u>	**F_Name**	**L_Name**	**Dept_ID**	**Dept_Name**
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

Bringing a Relation to 3NF

EMPLOYEE

Emp_ID	F_Name	L_Name	Dept_ID	Dept_Name
111	Mary	Jones	1	Acct
122	Sarah	Smith	2	Mktg

EMPLOYEE

Emp_ID	F_Name	L_Name	Dept_ID
111	Mary	Jones	1
122	Sarah	Smith	2

DEPARTMENT

Dept_ID	Dept_Name
1	Acct
2	Mktg

1NF

Deptno	Dname	Location
10	IT	Leeds, Bradford, Kent
20	Research	Hundredfold
30	Marketing	Leeds

1NF

Deptno	Dname	Location
10	IT	Leeds, Bradford, Kent
20	Research	Hundredfold
30	Marketing	Leeds

Deptno	Dname
10	IT
20	Research
30	Marketing

Deptno	Location
10	Leeds
10	Bradfprd
10	Kent
20	Hundredfold
30	Leeds

2NF

PNo	PName	PLoc	EmpNo	EName	Salary	Address	HoursNo

Given the following FDs:

$PNo \;, EmpNo \; - - - -- > \; HoursNo$

$PNo \; - - - -- > \; Dname \;, Loc$

$EmpNo \; - - - -- > \; Name \;, Salary \;, Address$

Assuming all attributes are atomic, is the above relation in the 1NF, 2NF ?

Relation X1

PNo	PName	PLoc

Relation X3

PNo	EmpNo	HoursNo

Relation X2

EmpNo	EName	Salary	Address

Example: 1NF

Order(<u>OrderNumber</u>, OrderDate, {PartNumber, {Supplier}})

5

Example: 1NF

Order(OrderNumber, OrderDate, {PartNumber, {Supplier}})

Order(OrderNumber, OrderDate)

Order-Part(OrderNumber, PartNumber)

Part(PartNumber, {Supplier})

Example: 1NF (cont.)

Part(PartNumber, {Supplier})

Part(PartNumber)

Part-Supplier(PartNumber, SupplierNum)

Supplier(SupplierNum)

2nd Normal Form

- No partial dependencies

  No attribute depends on only some of the attributes of a concatenated key.

Order-Part

[<u>OrderNumber</u> | <u>PartNumber</u> | PartDescription]

Create a new table with PartNumber key.