# Part 2

# A) Is the Schema in 1NF?

No it is not.

We want all the attributes to be atomic

### **Department Location is not atomic**

If we want to access the city of a given employee we would need to parse the string of location

### Solution:

Assuming location is only in Canada

Split Location into 4 attributes:

- address\_line
  - Street names and numbers are complicated with different units and subunits so the culmination can be put here as as string
- city
- province
- postalcode
  - Although province can most likely be found using the postalcode it is not a guarantee
  - o Assuming a string a length 6 without the dash

### Start and End date are not atomic

Since we are ignoring dates this does not matter

### **Potential Solution**

Split Dates into the following:

- start\_date into
  - o start\_date\_year
  - o start\_date\_month
  - o start\_date\_day
- end\_date into
  - o end\_date\_year
  - o end\_date\_month
  - o end\_date\_day

### **Splitting Role Level**

Although role level can technically be split into L and then the level number, it does not reduce the complexity because L is always in front a a level number. If other first letters were possible such as M1 then we would split the role\_level into 2 attributes

Part 2

# **New Relational Schema Diagram**

# Employee - empID INT(11) - emp\_fname VARCHAR(20) - emp\_initials VARCHAR(3) - emp\_lname VARCHAR(20) - job VARCHAR(100) - deptID INT(11) - salary INT(11)

```
Project
- projID INT(11)
- title VARCHAR(100)
- phase VARCHAR(20)
- budget DECIMAL(10,2)
- funds DECIMAL(10,2)
```

# Assigned - empID INT(11) - projID INT(11) - roleID INT(11) - role\_description VARCHAR(100) - role\_level VARHCAR(100) - state\_date DATE - end\_date DATE - assigned\_status VARCHAR(20)

```
Department

- deptID INT(11)
- deptName VARCHAR(100)
- address_line VARCHAR(200)
- city VARCHAR(100)
- province VARCHAR(100)
- postalcode VARCHAR(6)
```

# B1) Is the 1NF Schema in 3NF?

# **Checking 3NF**

## **Functional Dependencies**

 $\begin{aligned} \textbf{Department} & F = \{ \text{deptID} \rightarrow \text{deptID}, \text{location} \} \\ & = \{ \text{deptID} \rightarrow \text{deptID}, \text{address\_line}, \text{city}, \text{province}, \text{postalcode} \} \end{aligned}$ 

**Project**  $F = \{\text{projID} \rightarrow \text{title, phase, budget, funds}\}$ 

Employee  $F = \{\text{empID} \rightarrow \text{emp\_fname, emp\_initials, emp\_lname, job, deptID, salary}\}$ 

### **Assigned**

```
F = \{ \text{empID}, \text{projID}, \text{roleID} \rightarrow \text{role\_description}, \text{role\_level}, \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \\ \text{roleID} \rightarrow \text{role\_description}, \text{role\_level} \\ \}
```

Since Department Project and Employee FD all have their primary key on the left hand side. Thus these tables pass 3NF

For Assigned, roleID is not a super key because it cannot identify  $start_date$ ,  $end_date$ , or  $assigned_status$  thus because of FD  $roleID \rightarrow role_description$ ,  $role_level$ , Assigned is not in 3NF

# **3NF Decomposition**

We need to decompose Assigned to be in 3NF

```
R = (\mbox{empID}, \mbox{projID}, \mbox{roleID} \\ \mbox{role\_description}, \mbox{role\_level}, \mbox{start\_date}, \mbox{end\_date}, \mbox{assigned\_status})
```

```
F = \{ \text{empID, projID, roleID} \rightarrow \text{role\_description, role\_level, start\_date, end\_date, assigned\_status,} \\ \text{roleID} \rightarrow \text{role\_description, role\_level} \\ \}
```

Part 2

### **Finding Canonical Cover**

```
Proving role_description and role_level is extraneous in:
empID, projID, roleID \rightarrow role_description, role_level, start_date, end_date, assigned_status
Let:
F' = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \}
                                                                                              }
                              roleID \rightarrow role\_description, role\_level
                empID, projID, roleID \rightarrowroleID
                                                                                            \{\text{relexivity}\}\ (1)
                                    roleID \rightarrow role\_description, role\_level
                                                                                          {given in F'} (2)
                empID, projID, roleID \rightarrowrole_description, role_level {transitivity 1,2}
                                                                                          {given in F'} (4)
                empID, projID, roleID →start_date,
                                                end_date,
                                                assigned_status
                                                                                            \{union 3,4\} (5)
                empID, projID, roleID \rightarrowrole_description,
                                                role_level,
                                                start_date,
                                                end_date,
                                                assigned\_status
Thus:
F_C = \{\text{empID}, \text{projID}, \text{roleID} \rightarrow \text{start\_date}, \text{end\_date}, \text{assigned\_status}, \}
                              roleID \rightarrow role\_description, role\_level
```

### **Decomposition**

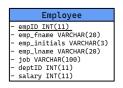
So we can split the relation into

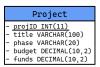
 $R_1 = (\text{empID}, \text{projID}, \text{roleID}, \text{start\_date}, \text{end\_date}, \text{assigned\_status})$ 

 $R_2 = (\text{roleID}, \text{role\_description}, \text{role\_level})$ 

Since both these relations only have super keys on the left of the FD they are both in 3NF Let  $R_1$  be named Assigned and  $R_1$  be Roles

# **New Relational Schema Diagram**





	Assigned	
F	empID INT(11)	
ŀ	projID INT(11)	
	roleID INT(11)	
ŀ	state_date DATE	
	end_date DATE	
Ŀ	assigned_status	VARCHAR(20)
_		

L	Department		
F	deptID INT(11)		
ŀ	deptName VARCHAR(100)		
ŀ	address_line VARCHAR(200)		
ŀ	city VARCHAR(100)		
ŀ	province VARCHAR(100)		
ŀ	postalcode VARCHAR(6)		

Roles
- roleID INT(11)
- role\_description VARCHAR(100)
- role\_level VARHCAR(100)

3

Part 2

Part 2 4