

Database Development & Design

LOGICAL DESIGN (FUNCTIONAL DEPENDENCIES)

Outline

- Functional Dependencies
- Partial Functional Dependencies
- Functional Dependencies Types

Functional Dependencies (FDs)

- A **functional dependency (FD)** is a constraint that is derived from the meaning and interrelationships of the data attributes.
- A set of attributes X functionally determines a set of attributes Y if the value of X determines a unique value for Y (i.e. if whenever two tuples have the same value for X, they *must have* the same value for Y)
- It is denoted as $X \rightarrow Y$, where X is a set of attributes that is capable of determining the value of Y.
- The attribute set on the left side of the arrow, X is called **Determinant**, while on the right side, Y is called the **Dependent**.
- Used to specify formal measures of the "goodness" of relational designs.
- In order to define the FDs, we need to understand the meaning of the attributes involved and the relationship between them.

Functional dependencies (FDs)

► From the employee table we can conclude some **valid** functional dependencies:

- $\text{number} \rightarrow \{\text{name}, \text{salary}, \text{city}\}$
 - $\text{number} \rightarrow \{\text{name}\}$
 - $\text{number} \rightarrow \{\text{salary}\}$
 - $\{\text{number}, \text{name}\} \rightarrow \{\text{salary}, \text{city}\}$
- Fully
Dependency**

Number	Name	Salary	City
1	Dana	50000	Tokyo
2	Dana	38000	London
3	Andrew	25000	Tokyo

► Here are some **invalid** functional dependencies:

- $\text{name} \rightarrow \{\text{city}\}$ employees with the same name can have different city, hence this is not a valid functional dependency.
- $\text{name} \rightarrow \{\text{number}, \text{city}\}$

FDs Example

What FDs may exist in the following example?

<u>emp SSN</u>	emp name	birthdate	address	dept number	dept name	manager SSN
123456789	John	1/9/1965	731 Fondren, Houston, TX	5	Research	333445555
333445555	Franklin	12/8/1955	638 Voss, Houston, TX	5	Research	333445555
453453453	Joyce	7/31/1972	5631 Rice, Houston, TX	5	Research	333445555
666884444	Ramesh	9/15/1962	975 Fire Oak, Humble, TX	5	Research	333445555
888665555	James	11/10/1937	450 Stone, Houston, TX	1	Head Quarters	888665555
987654321	Jennifer	6/20/1941	291 Berry, Bellaire, TX	4	Administration	987654321

Fully Dependency on the PK

- Emp SSN \rightarrow {emp name, birthdate, address, dept number, dept name, manager SSN}

Other Dependencies

- Dept number \rightarrow {dept name, manager SSN}
- Dept name \rightarrow {dept number, manager SSN}
- Manager SSN \rightarrow {dept number, dept name}

Partial Dependency

Partial Dependency occurs when a **non-prime attribute** is functionally dependent on **part of a candidate key or part of the primary key**.

<u>CustomerID</u>	<u>AccountID</u>	CustomerName	CustomerCity	AccountStatus	AccountBalance
1	1111012	Dana	Tokyo	Activated	50000
2	1112524	Dana	London	Activated	38000
3	1111968	Andrew	Tokyo	blocked	25000
4	1111968	Ali	Amman	blocked	25000
4	2222222	Ali	Amman	Activated	30000

- ▶ CustomerID → CustomerName
- ▶ AccountID → AccountStatus

CustomerName and AccountStatus are non-prime attributes, and each depends on a part of the candidate key (CustomerID, AccountID)

Partial Dependency

What FDs may exist in the following example?

<u>emp SSN</u>	<u>proj number</u>	hours	emp name	proj name	proj location
123456789	1	32.5	John	ProductX	Bellaire
123456789	2	7.5	John	ProductY	Sugarland
333445555	2	10	Franklin	ProductY	Sugarland
333445555	3	10	Franklin	ProductZ	Houston
333445555	10	10	Franklin	Computerization	Stafford
333445555	20	10	Franklin	Reorganization	Houston
453453453	1	20	Joyce	ProductX	Bellaire
453453453	2	20	Joyce	ProductY	Sugarland
666884444	3	40	Ramesh	ProductZ	Houston

Partial Dependency

- Emp SSN \rightarrow {emp name}
- Proj number \rightarrow {proj name, proj location}

Fully Dependency

- {emp SSN, Proj number} \rightarrow {hours}

FDs Types

- Trivial functional dependency
- Non-trivial functional dependency
- Multi-valued dependency
- Transitive dependency

Trivial FDs

In Trivial Functional Dependency, a **dependent** is always a subset of the **determinant**. i.e. If $X \rightarrow Y$ and Y is the subset of X , then it is called trivial functional dependency.

- ▶ $X \rightarrow Y$ is trivial functional dependency if Y is a subset of X . $\Rightarrow (XY \rightarrow Y)$
- ▶ **Examples:**
 - ▶ $\{\text{Student ID, Student Name}\} \rightarrow \{\text{Student Name}\}$ is a trivial functional dependency as Student Name is a subset of $\{\text{Student ID, Student Name}\}$. That makes sense because if we know the values of Student ID and Student Name then the value of Student ID can be uniquely determined.
 - ▶ $\{\text{number, name}\} \rightarrow \text{number}$ is a trivial functional dependency as Employee number is a subset of $\{\text{number, name}\}$.
 - ▶ $\{\text{number, name}\} \rightarrow \text{name}$ is a trivial functional dependency as Employee name is a subset of $\{\text{number, name}\}$.

Non-Trivial FDs

In Non-trivial functional dependency, the **dependent** is strictly **not** a subset of the **determinant**. i.e. If $X \rightarrow Y$ and Y is not a subset of X, then it is called Non-trivial functional dependency.

► Examples:

- $\{\text{customerID}\} \rightarrow \{\text{name, email, birthdate}\}$ is a non-trivial functional dependency as (name, email, birthdate) are not a subset of customerID.
- $\{\text{Company}\} \rightarrow \{\text{CEO}\}$ (if we know the Company, we know the CEO name). But CEO is not a subset of Company, and hence it's non-trivial functional dependency.

Multivalued FDs

In Multivalued functional dependency, entities of the dependent set are not dependent on each other. i.e. If **X** → {**Y, Z**} and **there exists no functional dependency between Y and Z**, then it is called a multivalued functional dependency.

► Example:

<u>customer IDs</u>	Name	Postal Code
1001	Ali	111111
1001	Ali	111111
1002	John	333333
1002	John	333333
1003	Ali	111111

- **customerID -> name**
- **customerID -> Postal Code**

Here, name and Postal Code are independent of each other and dependent on **customerID**. In this case, these two columns are said to be multivalued dependent on **customerID**.

Transitive FDs

In transitive functional dependency, **dependent** is indirectly dependent on **determinant**. i.e. If $X \rightarrow Y$ & $Y \rightarrow Z$, then according to axiom of transitivity, $X \rightarrow Z$.

► Example:

Book	Author name	Author Age
Book A	Author1	66
Book B	Author2	40
Book C	Author1	66

- $\{\text{Book}\} \rightarrow \{\text{Author name}\}$
- $\{\text{Author name}\} \rightarrow \{\text{Author Age}\}$
- So:
 - $\{\text{Book}\} \rightarrow \{\text{Author Age}\}$

► Example:

Company	CEO	CEO Age
Microsoft	Satya Nadella	51
Google	Sundar Pichai	46
Alibaba	Satya Nadella	51

- $\{\text{Company}\} \rightarrow \{\text{CEO}\}$
- $\{\text{CEO}\} \rightarrow \{\text{CEO Age}\}$
- So:
 - $\{\text{Company}\} \rightarrow \{\text{CEO Age}\}$

Advantages of Functional Dependency

Here some Advantages of Functional Dependency:

1. Functional Dependency avoids data redundancy. Therefore, same data do not repeat at multiple locations in that database.
2. It helps you to maintain the quality of data in the database.
3. It helps you to defined meanings and constraints of databases.
4. It helps you to identify bad designs.
5. It helps you to find the facts regarding the database design.

References

- Elmasri, R., & Navathe, S. (2017). Fundamentals of database systems (Vol. 7). Pearson
- Eng. Lina's slides
- Dr. Raneem's slides.