Module-3

Schema Refinement

Schema Refinement

- Guidelines for the Relational Schema
- Functional dependencies
- Normalization
  - Boyce Code Normal form
  - Multi valued dependency and Fourth Normal form
  - Join dependency and Fifth Normal form

Schema Refinement

- Informal Design Guidelines for Relational Schema

Schema Refinement

- Informal Design Guidelines for Relational Schema
    1. Semantics of the Attributes
    2. Reducing the Redundant Value in Tuples.
    3. Reducing Null values in Tuples.
    4. Dissallowing spurious Tuples.

Schema Refinement

- Informal Design Guidelines for Relational Schema
    1. Semantics of the Attributes

Schema Refinement

- Informal Design Guidelines for Relational Schema

1.Semantics of the Attributes

  - Whenever we are going to form relational schema there should be some meaning among the attributes. This meaning is called semantics.
  - This semantics relates one attribute to another with some relation.

Schema Refinement

- Informal Design Guidelines for Relational Schema

1.Semantics of the Attributes

USN No

_____	Student name	Sem

Schema Refinement

- Informal Design Guidelines for Relational Schema

2. Reducing the Redundant Value in Tuples

Schema Refinement

- Informal Design Guidelines for Relational Schema

2. Reducing the Redundant Value in Tuples

  - Mixing attributes of multiple entities may cause problems
  - Information is stored redundantly wasting storage
  - Problems with update anomalies
    - Insertion anomalies
    - Deletion anomalies
    - Modification anomalies

Schema Refinement

- Informal Design Guidelines for Relational Schema

  2. Reducing the Redundant Value in Tuples

USN No

	Student name	Sem
_____		

Dept No	Dept Name

USN No	Student name	Sem	Dept No	Dept Name

Schema Refinement

- Informal Design Guidelines for Relational Schema

  2. Reducing the Redundant Value in Tuples

  – Here whenever if we insert the tuples there may be ‘N’ students in one department, so Dept No, Dept Name values are repeated ‘N’ times which leads to data redundancy.

Redundancy

EMP_DEPT

Ename	Ssn	Bdate	Address	Dnumber	Dname	Dmgr_ssn
Smith, John B.	123456789	1965-01-09	731 Fondren, Houston, TX	5	Research	333445555
Wong, Franklin T.	333445555	1955-12-08	638 Voss, Houston, TX	5	Research	333445555
Zelaya, Alicia J.	999887777	1968-07-19	3321 Castle, Spring, TX	4	Administration	987654321
Wallace, Jennifer S.	987654321	1941-06-20	291 Berry, Bellaire, TX	4	Administration	987654321
Narayan, Ramesh K.	666884444	1962-09-15	975 FireOak, Humble, TX	5	Research	333445555
English, Joyce A.	453453453	1972-07-31	5631 Rice, Houston, TX	5	Research	333445555
Jabbar, Ahmad V.	987987987	1969-03-29	980 Dallas, Houston, TX	4	Administration	987654321
Borg, James E.	888665555	1937-11-10	450 Stone, Houston, TX	1	Headquarters	888665555

Schema Refinement

- Informal Design Guidelines for Relational Schema

  2. Reducing the Redundant Value in Tuples

  – If we delete the last student of a dept, then whole information about that department will be deleted

Schema Refinement

- Informal Design Guidelines for Relational Schema

  2. Reducing the Redundant Value in Tuples

  – Another problem is update anamolies i.e. if we insert new dept that has no students.

Schema Refinement

- Informal Design Guidelines for Relational Schema

  2. Reducing the Redundant Value in Tuples

  – If we change the value of one of the attributes of a particular table the we must update the tuples of all the students belonging to that dept else Database will become inconsistent.

Schema Refinement

- Informal Design Guidelines for Relational Schema

3. Reducing the Null Values

Schema Refinement

- Informal Design Guidelines for Relational Schema

3. Reducing the Null Values

  - Relations should be designed such that their tuples will have as few NULL values as possible
  - Attributes that are NULL frequently could be placed in separate relations (with the primary key)

Schema Refinement

- Informal Design Guidelines for Relational Schema

4. Dissallowing spurious Tuples

Emp_ Loc

Ename	Ploc
RAJA	SJT
KUMAR	TT
SIVA	SJT
.....	

Emp_ Proj

SSN	Pno	P Name	Ploc
101	44	A	SJT
102	45	B	TT
103	46	C	SJT
....			

Schema Refinement

- Informal Design Guidelines for Relational Schema

4. Dissallowing spurious Tuples

SSN	Pno	P Name	EName	Ploc
101	44	A	RAJA	SJT
102	45	B	KUMAR	TT
103	46	C	RAJA	SJT
....				

Schema Refinement

- Normalization
  - Data analysis technique to design a Database system
  - Abnormal data to normal data
  - Properties
    - No redundancy
    - No update anomaly

Schema Refinement

- Normalization
  1. Functional Dependencies
  2. Finding the key attribute

Schema Refinement

- Normalization
  1. Functional Dependencies

Schema Refinement

- Normalization
  1. Functional Dependencies
     - It is a constraint between the 2 sets of attributes from the database
     - It is a property of the semantic of the database
     - DB designers specify the semantics by functional dependencies
     - Example:

       FD: x -> y ; x, y are attributes

       x determines y or y is dependent on x

Schema Refinement

- Normalization

  2. Finding the key attribute

  Algorithm:

  Determine X+ , the closure of X under F.

  X+ = X;

  repeat

  old X+ =X+;

  for each FD Y -> Z in F do

  if X+ >= Y then x+ = X+ U Z;

  until (X+ = old X+)

Schema Refinement

- Normalization

  2. Finding the key attribute

  FD1: A -> B , C ; FD2: B -> D; FD3: C -> D

  FD4: D -> E

  Step-1: A+ = { A }

  Step-2: A+ = { A , B , C}

  Step-3: A+ = { A , B, C, D}

  Step-4: A+ = { A , B, C, D}

  Step-5: A+ = { A , B, C, D, E}

  Similarly find B+, C+, D+

Schema Refinement

- Normalization

  2. Finding the key attribute

  FD1: A -> B , C ; FD2: B -> C; FD3: C -> A

  Find A+, B+, C+

Schema Refinement

- Normalization

  2. Finding the key attribute

  R:{ A, B, C, D, E, F, G, H, I, J }

  FD1: (A, B) -> C;

  FD2: A -> D, E, C;

  FD3: B -> F, C

  FD4: F -> G, H

  FD3: D -> I, J

  Find A+, B+, C+, F+, D+

Schema Refinement

- Normalization

  2. Finding the key attribute

  R:{ A, B, C, D, E}

  FD1: A -> B, C;

  FD2: B -> D;

  FD3: C -> A

  FD4: E -> B

  Find A+, B+, C+, E+

Schema Refinement

FD1: SSN, P no - > Hours

FD2: SSN -> Ename

FD3: Pno -> Pname, Ploc

SSN	P No	Hours	Ename	P Name	P Loc
.....					

Schema Refinement

- Transitive Dependency
  - It exists when there is a intermediate functional dependency
  - Notation

    if A -> B and B -> C

    Then transitive dependency exists

    A -> B -> C

Schema Refinement

- Transitive Dependency
  - Example

Staff No	Job	Dept No	Dept Name

Staff No -> Dept No

Dept No -> Dept Name

Then

Staff No -> Dept No -> Dept Name

Schema Refinement

- Composite determinant
  - If two attributes determine one attribute
  - Notation

    A , C -> B

Schema Refinement

- Inference Rules
  - Reflexivity
  - Augmentation
  - Transitivity
  - Union
  - Decomposition
  - Pseudo Transitivity

Schema Refinement

- Inference Rules

Axiom of reflexivity

If $Y \subseteq X$, then $X \rightarrow Y$

Axiom of augmentation

If $X \rightarrow Y$, then $XZ \rightarrow YZ$ for any Z

Axiom of transitivity

If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Schema Refinement

- Inference Rules

Union

If $X \rightarrow Y$ and $X \rightarrow Z$ then $X \rightarrow YZ$

Decomposition

If $X \rightarrow YZ$ then $X \rightarrow Y$ and $X \rightarrow Z$

Pseudo transitivity

If $A \rightarrow B$ and $BC \rightarrow D$ then $AC \rightarrow D$

Schema Refinement

- Inference Rules
  - Union:

    A -> B, A-> C, then A -> BC

    Proof:

    A -> B

    AA -> AB

    A->AB

    A -> C

    AB -> BC

    Hence A -> BC

Schema Refinement

- Inference Rules
  - Pseudo Transitivity:

    A -> B, BC-> D, then AC-> D

    Proof:

    A -> B

    AC -> BC

    BC-> D

    Hence AC -> D

Schema Refinement

- Inference Rules
  - Decomposition:

    A -> BC, then A-> B, A ->C

    Proof:

    A -> BC

    BC -> B

    Hence A -> B, similarly A -> C