Schema Refinement

R&G Ch 19

Redundancy

OID	Name	Post	Rank	Salary
1	Sheridan	Bab 5	Capt.	200,000
2	Ivanova	Bab 5	Cmdr.	180,000
3	Sinclair	Bab 5	Cmdr.	180,000
4	Keffer	Bab 5	Lt.	120,000
5	Corwin	Bab 5	Lt.	120,000

Redundancy

OID	Name	Post	Rank	Salary
1	Sheridan	Bab 5	Capt.	200,000
2	Ivanova	Bab 5	Cmdr.	180,000
3	Sinclair	Bab 5	Cmdr.	180,000
4	Keffer	Bab 5	Lt.	120,000
5	Corwin	Bab 5	Lt.	120,000

Commanders earn 180,000 credits.
Lieutenants earn 120,000 credits.

Redundancy

OID	Name	Post	Rank	Salary
1	Sheridan	Bab 5	Capt.	200,000
2	Ivanova	Bab 5	Cmdr.	180,000
3	Sinclair	Bab 5	Cmdr.	180,000
4	Keffer	Bab 5	Lt.	120,000
5	Corwin	Bab 5	Lt.	120,000

Commanders earn 180,000 credits.

Lieutenants earn 120,000 credits.

<Salary> depends on <Rank>

Redundancy

What's wrong with this picture?

- Wasted space: 2 values/officer
- How do we find the salary of new officers?
- What happens if we delete Capt. Sheridan?
- What is involved in changing the default salary for all officers of a given rank?

Redundancy

What's wrong with this picture?

- Wasted space: 2 values/officer
- How do we find the salary of new officers?
- What happens if we delete Capt. Sheridan?
- What is involved in changing the default salary for all officers of a given rank?

Eliminate Redundancy By Splitting Relations (Decomposition)

Decomposition

- Replace R(A,B,C,D) with, for example,
  - R1(A,B), R2(B,C,D) or
  - R1(A,C,D), R2(A,B,D)

- When is it useful to decompose?
  - What are the costs of decomposition?

Functional Dependencies

- A <u>functional dependency</u> $X \rightarrow Y$ holds over relation R if for every pair of tuples t_1, t_2 in R, it holds that if $\pi_X t_1 = \pi_X t_2$, then $\pi_Y t_1 = \pi_Y t_2$.
  - X and Y are sets of columns
- A FD isn't just a statement about a particular instance of R, but about application semantics.
  - We can check to see of an FD holds over R, but can't check to see if R has an FD.

Example

- Officers(oid, name, post, rank, salary)
  - Cols abbreviated ONPRS
    - i.e., the set of attributes {O, N, P, R, S}
- Some example FDs on Officers
  - oid is a key: O → ONPRS
    - If X is a key, is X → R an FD? If X → R, is X a key?
  - rating determines salary: R → S

Example-Problems

OID	Name	Post	Rank	Salary
1	Sheridan	Bab 5	Capt.	200,000
2	Ivanova	Bab 5	Cmdr.	180,000
3	Sinclair	Bab 5	Cmdr.	180,000
4	Keffer	Bab 5	Lt.	120,000
5	Corwin	Bab 5	Lt.	120,000

Update: Can we just update S in row 2

Insert: How do we insert an admiral?

Delete: What happens if we delete row 1?

Reasoning about FDs

- Armstrong's Axioms: (Implication rules for FDs)
  - **Reflexivity**: If $X \subseteq Y$ then $Y \rightarrow X$
  - **Augmentation**: If $X \rightarrow Y$ then $XZ \rightarrow YZ$ for any Z
  - **Transitivity**: If $X \rightarrow Y$, $Y \rightarrow Z$ then $X \rightarrow Z$
- If F is a set of FDs, then F^+ is the closure of F, the set of all FDs implied by F.

Reasoning about FDs

- A few additional rules
  - **Union**: If $X \rightarrow Y$ and $X \rightarrow Z$ then $X \rightarrow YZ$
  - **Decomp.**: If $X \rightarrow YZ$ then $X \rightarrow Y$, $X \rightarrow Z$
- These rules follow from Armstrong's Axioms.
- AA rules are sound and complete.

Example

Contracts(***c****id*, ***s****id*, ***j****id*, ***d****id*, ***p****id*, ***q****ty*, ***v****alue*)

1. C is a key: C → CSJDPQV
2. Pro**J**ects purchase **P**arts using a single **C**ontract: JP → C
3. **D**epts. purchase at most one **P**art from any **S**upplier: SD → P
4. (1), (2) imply that JP → CSJDPQV
5. (3) implies that SDJ → JP
6. (4), (5) imply that SDJ → CSJDPQV

Reasoning about FDs

- Computing the closure of a set of FDs can be expensive (size is exponential in #attr)
  - … but usually only need if $X \rightarrow Y \in F^+$?
- Instead, compute <u>attribute closure</u> of X (X^+)
  - Maximal A such that $X \rightarrow A \in F^+$.
- How do we compute this?

Example

- Does $F = \{A \rightarrow B, B \rightarrow C, CD \rightarrow E\}$ imply that $A \rightarrow E$?
  - … is $A \rightarrow E \in F^+$?
  - … is $E \in A^+$?

Decomposition

- Replace R(A,B,C,D) with, for example,
  - R1(A,B), R2(B,C,D) or
  - R1(A,C,D), R2(A,B,D)

- When is it useful to decompose?
  - What are the costs of decomposition?

Normal Forms

Normal Forms

- If a relation is in one of the normal forms (BCNF, 3NF) certain problems are avoided/minimized.
- Decomposition can produce relations in/closer to a normal form.
- FDs help us detect redundancy
  - For R(A,B,C), if A → B, and several tuples have the same A value, they'll all have the same Bs.

Boyce-Codd Normal Form (BCNF)

- R (with FDs F) is in BCNF if for all $X \rightarrow A \in F^+$:
  - $A \subseteq X$ (the trivial FD), or
  - X contains a key for R
- In other words, R is in BCNF if the only non-trivial FDs that hold over R are key constraints.

BCNF Isn't Always Viable

Ship	**Crew Role**	**Officer**
Enterprise	Captain	Kirk
Enterprise	Science	Spock
Enterprise	Medical	McCoy
Excelsior	Captain	Sulu

Ship, Crew Role → Officer

Officer → Crew Role

Keys: {Ship, Crew Role}, {Crew Role, Officer}

BCNF Isn't Always Viable

Ship	Crew Role	Officer
Enterprise	Captain	Kirk
Enterprise	Science	Spock
Enterprise	Medical	McCoy
Excelsior	Captain	Sulu

Ship, Crew Role → Officer

Officer → Crew Role

Keys: {Ship, Crew Role}, {Crew Role, Officer}

3rd Normal Form

- R (with FDs F) is in BCNF if for all $X \to A \in F^+$:
  - $A \subseteq X$ (the trivial FD), or
  - X contains a key for R, or
  - A is a subset of any key for R
    - Recall that keys are <u>minimal</u> sets of attributes.
- Weaker form of BCNF
  - …used when BCNF impractical, impossible.

3rd Normal Form

- If 3NF is violated by $X \rightarrow A$ then:
  - X is a subset of some key K
    - Some (X,A)s are being stored redundantly.
  - X is not a proper subset of any key
    - So there exists redundancy: $K \rightarrow X \rightarrow A$
      - But this can still happen in 3NF.

3NF Isn't Always Perfect

Ship	**Crew Role**	**Officer**
Enterprise	Captain	Kirk
Enterprise	Science	Spock
Enterprise	Medical	McCoy
Excelsior	Captain	Sulu

Ship, Crew Role → Officer

Officer → Crew Role

Keys: {Ship, Crew Role}, {Crew Role, Officer}

3NF Isn't Always Perfect

Ship	Crew Role	Officer
Enterprise	Captain	Kirk
Enterprise	Science	Spock
Enterprise	Medical	McCoy
Excelsior	Captain	Sulu

Ship, Crew Role → Officer

Officer → Crew Role

Keys: {Ship, Crew Role}, {Crew Role, Officer}

3NF Isn't Always Perfect

- BCNF can't always be decomposed (as in example)
- 3NF is a compromise:
  - Guaranteed to be possible to decompose to 3NF.
  - Not guaranteed to lack redundancy.

Decomposition

- Starting with $R(A_1, \ldots, A_n)$, a decomposition creates relations $R_1, R_2, \ldots$ such that
  - $R_i \subset R$ (R_i contains only attributes in R)
  - $R \equiv R_1 \cup R_2 \cup \ldots$ (each attribute appears at least once in a decomposed rel)
- We store instances of the R_is instead of R.

Example

- Officers(**O**id, **N**ame, **P**ost, **R**ank, **S**alary)
  - F = {O → N,P,R,S; R → S}
  - R → S violates 3NF
- **Store**: Officers'(ONPR), Salaries(RS)
  - Can we just project Officers down to O',S?
  - What problems could occur?

Decomposition Costs

- Queries become more expensive:
  - How much does Sheridan earn? (2 way join)
- May not be possible to reconstruct original relation from instances.
  - $R_1(A,B)$, $R_2(B,C)$, $R_3(A,C)$
- Checking dependencies may require joining instances of the decomposed relation.