Decomposition algorithm

The content of the slides is authored by Prof. Maria de Marsico, Sapienza University of Rome
Part of the material is taken from the book: Atzeni, Ceri, Paraboschi, Torlone Basi di dati - Modelli e linguaggi di interrogazione , McGraw-Hill, 2002.

Introduction

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- we now show that, given a relation schema R and a set of functional dependencies F there **always** exists a decomposition $\rho = R_1, R_2, \ldots, R_k$ of R such that:
  - for each $i = 1, \ldots, k$, $R_i \in \rho$, R_i is in 3NF
  - ρ preserves F
  - ρ has a lossless join

- and that such a decomposition can be computed in polynomial time

How do we proceed?

- the decomposition algorithm, given a relational schema *R* and a set of functional dependencies *F* on *R*, which is a **minimal cover**, allows us to **compute** in polynomial time a decomposition
    -
        - $\rho = R_1, R_2, \ldots, R_k$ of *R* such that:
    -
- for each $i = 1, \ldots, k$, R_i is in 3NF
- ρ preserves *F*

We can consider **any** minimal cover F defined on the schema *R* if there are more than one.

Decomposition Algorithm

Algorithm - decomposition of a relational schema

Input: a relational schema R and a set F of functional dependencies on R, which is a **minimal cover**

Output: a decomposition of R that preserves F and such that for each subschema in is in 3NF

begin

S = ∅

ρ = ∅

for each A∈R, such that A is not involved in any functional dependency in F **do**

  S = S ∪ A

if S ≠ ∅ **then:**

  R = R - S

  ρ = ρ ∪ {S}

if there is a functional dependency in F that involves all the attributes in R

  then: ρ = ρ ∪ {R}

else:

  for each X → A∈F **do**

    ρ = ρ ∪ {XA}

end

Theorem

Sapienza Università di Roma
Dipartimento di Informatica

Theorem: let R be a relational schema and F a set of functional dependencies on R, which is a minimal cover; the Decomposition Algorithm computes (in polynomial time) a decomposition ρ of R such that:

- each relational schema in ρ is in 3NF
- ρ preserves F

Proof

ρ preserves F

let $G=\cup_{i=1}^{k} \pi_{R_i}(F)$; since for each **functional dependency $X \rightarrow A \in F$ (all of them!)** we have **that $XA \in \rho$ (**it is one of the subschemas), we have that this dependency **of F will be in G**, hence $F \subseteq$ **G** and $F^+ \subseteq$ **G^+**; the inclusion $G^+ \subseteq F^+$, as we already know, is trivially true as, **by definition**, **$G \subseteq F^+$**

Theorem

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- **each schema in is in 3NF.**

let's analyze the different cases that can arise:

1. if $S \in \rho$, each attribute in S is part of the key, and so trivially, S is in 3NF

2. if $R \in \rho$, **there is** a functional dependency in F **involving all attributes of R;** as **F is a minimal cover,** this dependency will have the form **$R\text{-}A \rightarrow A$**; as **F is a minimal cover**, it cannot exist a functional dependency **$X \rightarrow A$ in F, such that $X \subseteq R\text{-}A$** and, then, **R-A is a key of** R; let **$Y \rightarrow B$** be a **dependency in F**; if B=A and F is a minimal cover, then Y=R-A (i.e., Y is a superkey); if $B \neq A$, then $B \in R\text{-}A$ and so B is prime

3. if $XA \in \rho$, as **F is a minimal cover**, it cannot exist a functional dependency **$X' \rightarrow A$ in F, such that $X' \subset X$** and, hence, **X is key of XA;** let **$Y \rightarrow B$** be any dependency in F, such that **$YB \subseteq XA$**; if B=A then, as F is a minimal cover, Y=X (i.e., Y is a superkey); if $B \neq A$ then $B \in X$, so, B is prime

it can be either 1+2, or 1+3, or 3 only

Anything missing?

- what about the lossless join?

- it can be demonstrated that, to have a lossless join, we just need to add to ρ a subschema containing one of the keys of R

To sum up

SAPIENZA
UNIVERSITÀ DI ROMA
DIPARTIMENTO DI INFORMATICA

- there are schemas that are not "good"
- they are those in which many concepts are represented in a single schema:
  Curriculum (Matr, TC, SurN, Name, DateB, Town, Prov, C#, Tit, Teacher, DateP, Grade)
- they have redundancy, update, insertion and deletion anomalies

QUESTION 1:

- is it possible to formalize the concept of a "good" relational schema?

QUESTION 2:

- is it always possible to represent the reality of interest with a relational DB in which each relation schema is "good"?

Answer 1

yes!

a schema is good
if it is in 3rd Normal Form (3NF)

Answer 2

Sapienza Università di Roma
Dipartimento di Informatica

yes!

there is a polynomial algorithm that, given a relational schema R and a set of dependencies F on R , computes a decomposition of R such that:

- each relation schema in the decomposition is in 3NF
- the decomposition preserves F
- the decomposition has a lossless join