Design Theory for Relational DB
(Functional Dependencies and Normal Forms)

David Tanzer
Copyright 2000
A Relational Design with Flaws

• Consider the table \( R(\text{Person}, \text{Zip}, \text{City}) \):

<table>
<thead>
<tr>
<th>P</th>
<th>Z</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe</td>
<td>10024</td>
<td>New York</td>
</tr>
<tr>
<td>Mac</td>
<td>10024</td>
<td>New York</td>
</tr>
<tr>
<td>Jim</td>
<td>02111</td>
<td>Boston</td>
</tr>
<tr>
<td>Sid</td>
<td>02555</td>
<td>Boston</td>
</tr>
</tbody>
</table>

• Redundancy: “New York” is stored many times

• Update anomaly: Possible to update inconsistently

• Deletion anomaly: We may lose the City for a Zip
**Solution: Decompose the Relation**

- Suppose a relation $R(A)$ over the attributes $A$ suffers from these design flaws

- To remedy, we look for a good decomposition of $R(A)$ into relations $R_1(A_1), \ldots, R_n(A_n)$, where:
  1. Each $A_i \subseteq A$, and $A_1 \cup \cdots A_n = A$
  2. $R_i = \pi_{A_i}(R)$

- A decomposition of $R(\text{Person}, \text{Zip}, \text{City})$:
  - $R_1(P, Z) = \pi_{P,Z}(R)$
  - $R_2(Z, C) = \pi_{Z,C}(R)$

- A compact notation is $\{PZ, ZC\}$
What are the Pros and Cons of:

1. PC,ZC
   - $R_1 = \pi_{P,C}(R)$
   - $R_2 = \pi_{Z,C}(R)$

2. P,Z,C
   - $R_1 = \pi_P(R)$
   - $R_2 = \pi_Z(R)$
   - $R_3 = \pi_C(R)$

3. PZ,ZC
   - $R_1 = \pi_{P,Z}(R)$
   - $R_2 = \pi_{Z,C}(R)$
GOALS AND RESULTS OF RELATIONAL DESIGN THEORY

1. Define important properties for a decomposition:
   (a) Irredundancy
   (b) Expressiveness
   (c) Preservation of constraints

2. Algorithms for decomposing relations into various “normal forms”:
   • With BCNF, we can get (a) and (b)
   • With 3NF, we can get (b) and (c)

The foundation is the theory of functional dependencies.
Functional Dependencies

- Functional dependencies generalize the concept of many-one relationships
- Let $R$ be a relation with attributes $A$, and suppose that $X \subseteq A$ and $Y \subseteq A$
- Dependency $X \rightarrow Y$ asserts an integrity constraint:
  - If two tuples of $R$ are equal on $X$, then they are equal on $Y$
- Some dependencies for $R(\text{Person}, \text{Zip}, \text{City})$:
  - $\text{Person} \rightarrow \text{Zip}$
  - $\text{Zip} \rightarrow \text{City}$
More Examples of Functional Dependencies

- $R(Addr,City,Zip)$:
  $Addr,City \rightarrow Zip$
  $Zip \rightarrow City$

- $R(Course,Teacher,Hour,Room)$:
  $Course \rightarrow Teacher$
  $Hour,Room \rightarrow Course$
  $Hour,Teacher \rightarrow Room$
Using Dependencies to Define Keys

• In the ER model, we assessed the keys of entity-sets
• Here we use FDs to define the keys of relations
• For attributes $A$ with functional dependencies $F$, $K \subseteq A$ is a key if $(K \rightarrow A)$ follows from $F$, and no other subset of $K$ has this property

• Example:
  – $F = \{(Person \rightarrow Zip), (Zip \rightarrow City)\}$
  – $\{Person\}$ is a key for $\{Person, Zip, City\}$

• To prove: $\{(P \rightarrow Z), (Z \rightarrow C)\}$ implies $(P \rightarrow PZC)$
Implied Dependencies

• Claim: \( \{(P \rightarrow Z), (Z \rightarrow C)\} \) implies \((P \rightarrow PZC)\)

• \( \{f_1, \ldots , f_k\} \) implies \(g\) means:
  – Every relation that satisfies all the dependencies \(f_1, \ldots , f_k\) also satisfies \(g\)
  – Hence, \(g\) is an objective consequence of \(f_1, \ldots , f_k\)

• \( \{(A \rightarrow B), (B \rightarrow C)\} \) implies \((A \rightarrow C)\)

• \( \{(A \rightarrow B)\} \) does not imply \((B \rightarrow A)\)
Some Basic Implications

- \{(A \rightarrow B), (B \rightarrow C)\} \models (A \rightarrow C)
- \{(A \rightarrow X), (B \rightarrow Y)\} \models (AB \rightarrow XY)
- \{(A \rightarrow X_1 \cdots X_n)\} \models (A \rightarrow X_i)

The trivial dependencies:
(A \rightarrow B) holds whenever \(B \subseteq A\)

Every true implication can be proven from these basics.
Proof of an Implication

Proof of \((P \rightarrow PZC)\) from \(\{(P \rightarrow Z), \ (Z \rightarrow C)\}\):

1. \(P \rightarrow Z\) \hspace{1.5cm} \text{(given)}
2. \(Z \rightarrow C\) \hspace{1.5cm} \text{(given)}
3. \(P \rightarrow C\) \hspace{1.5cm} \text{(transitivity on 1,2)}
4. \(P \rightarrow P\) \hspace{1.5cm} \text{(trivial)}
5. \(P \rightarrow PZC\) \hspace{1.5cm} \text{(clustering on 1,3,4)}
**Closure of a Set of Attributes**

- Let $F$ be functional dependencies over attributes $A$, and $S \subseteq A$.
- The closure $S^+$ is the set of attributes $X$ such that $(S \rightarrow X)$ can be proven from $F$.
- Examples:
  - $P^+ = PZC$
  - $Z^+ = ZC$
  - $(PZ)^+ = PZC$
- $S \subseteq S^+$
- $S$ contains a key iff $(S^+) = A$
Computation of the Closure $S^+$

Res := $S$
loop
  if for some dependency $A \rightarrow X$,
    we have $A \subseteq Res$ and $X \notin Res$
then
  Res := Res $\cup \{X\}$
else break
endloop
S+ = Res
The BCNF Irredundancy Condition

• Let $R(A)$ be a relation schema with functional dependencies

• $R(A)$ is irredundant, or in Boyce-Codd Normal Form, if the left side of every dependency contains a key
  – i.e., if whenever $S \rightarrow T$, then $S \rightarrow A$

• Which of these are in BCNF?
  – $R$(Person,Name,Age)
  – $R$(Person,Zip,City)
  – $R$(City,Addr,Zip)

• Why does BCNF prevent redundancy?
**Why BCNF Prevents Redundancy**

- Consider \( R(\text{City}, \text{Addr}, \text{Zip}) \)
- Two keys: \( \{\text{City}, \text{Addr}\} \), and \( \{\text{Addr}, \text{Zip}\} \)
- \( (\text{Zip} \rightarrow \text{City}) \) violates BCNF
- Fact: 10024 is in NYC
- For irredundancy, Fact should be stored just once
- But since \( \{\text{Zip}\} \) does not contain a key, there can be many records with Zip=10024
- Thus the Fact can be stored in many records
Recomposing a Decomposed Relation

• Suppose $R$ is replaced by $R_1, \ldots , R_n$

• We should be able to reconstruct $R$ from $R_1, \ldots , R_n$

• Because $R_1, \ldots , R_n$ share attributes, we connect them by natural join $\Join$

• $R$ is represented by $R_1, \ldots , R_n$ if $R = R_1 \Join \cdots \Join R_n$

• Can $R$(Person,Zip,City) be represented as $R_1$(Person,Zip) $\Join R_2$(Zip,City)?

• Can $R$(Person,Zip) be represented as $R_1$(Person) $\Join R_2$(Zip)?