Warm-up Exercise

(See exercise sheet. You can start before class.)

a) Given: \( X \rightarrow Y \) and \( YW \rightarrow Z \)
\( XW \rightarrow YW \) (augmentation)
\( XW \rightarrow Z \) (transitivity)

b) \( E \) is a candidate key, \( D \) is not

Goals for Today

• Learn how to decompose a relation to adhere to Boyce-Codd Normal Form (BCNF)

• Reason about issues that result even if a decomposition is loss-less

• Understand the tradeoffs between BCNF and Third Normal Form (3NF)
### Loss-less Decomposition

- Decomposition of $R$ into $X$ and $Y$ is **lossless** w.r.t. a set of FDs $F$ if, for every instance $r$ that satisfies $F$:
  $$\pi_X(r) \Join \pi_Y(r) = r$$
- Decomposition of $R$ into $X$ and $Y$ is **lossless with respect to $F$** if and only if $F^+$ contains:
  - $X \cap Y \rightarrow X$, or
  - $X \cap Y \rightarrow Y$

**Corollary**: If $Z \rightarrow W$ holds over $R$ and $Z \cap W$ is empty, then decomposition of $R$ into $R-W$ and $ZW$ is loss-less.

- In “Oversees” example, decomposing into $\{E,P\}$ and $\{D,P\}$ is **lossy** because the intersection (i.e., Project) is not a key of either resulting relation.

### Reasoning about BCNF

- Relation $R$ with FDs $F$ is in BCNF if, for all $X \rightarrow A$ in $F^+$
  - $A \in X$ (a trivial FD), or
  - $X$ is a superkey for $R$

Also recall that relations are *sets* of tuples.

### Loss-less Decomposition into BCNF

- Relation $R$ has FDs $F$. If $Z \rightarrow W$ in $F$ violates BCNF:
  - decompose $R$ into $R - W$ and $ZW$
  (guaranteed to be loss-less)

### Reasoning about BCNF

- If relation $R$ is in BCNF, then each field of a tuple provides a fact *that cannot be inferred using FDs alone*

- Suppose we are told that the FD $X \rightarrow A$ holds for this relation:

<table>
<thead>
<tr>
<th>X</th>
<th>Y</th>
<th>A</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>y1</td>
<td>a</td>
</tr>
<tr>
<td>x</td>
<td>y2</td>
<td>?</td>
</tr>
</tbody>
</table>

  You can likely guess the value of the missing attribute!

  *We can infer missing value using FD... this relation is *not* in BCNF.*
Example 1: Is Hourly_Emps in BCNF?

Hourly_Emps

<table>
<thead>
<tr>
<th>S</th>
<th>N</th>
<th>L</th>
<th>R</th>
<th>W</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-22-3666</td>
<td>Attishoo</td>
<td>48</td>
<td>8</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>231-31-5368</td>
<td>Smiley</td>
<td>22</td>
<td>8</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>131-24-3650</td>
<td>Smethurst</td>
<td>35</td>
<td>5</td>
<td>7</td>
<td>30</td>
</tr>
<tr>
<td>434-26-3751</td>
<td>Guldu</td>
<td>35</td>
<td>5</td>
<td>7</td>
<td>32</td>
</tr>
<tr>
<td>612-67-4134</td>
<td>Madayan</td>
<td>35</td>
<td>8</td>
<td>10</td>
<td>40</td>
</tr>
</tbody>
</table>

- SNLRWH has FDs
  
  S → SNLRWH
  R → W

In BCNF??

Example 2: Is Bar_Sells in BCNF?

- Combing Bars and Sells
  Bar_Sells (bar_name, beer_name, address, price)

  FDs (for just Bar_Sells):
  bar_name → address
  bar_name, beer_name → price

In BCNF?? (Exercise 2)

Examples: BCNF Decomposition

- Hourly_Emps

<table>
<thead>
<tr>
<th>S</th>
<th>N</th>
<th>L</th>
<th>R</th>
<th>W</th>
<th>H</th>
</tr>
</thead>
<tbody>
<tr>
<td>123-22-3666</td>
<td>Attishoo</td>
<td>48</td>
<td>8</td>
<td>10</td>
<td>40</td>
</tr>
<tr>
<td>231-31-5368</td>
<td>Smiley</td>
<td>22</td>
<td>8</td>
<td>10</td>
<td>30</td>
</tr>
<tr>
<td>131-24-3650</td>
<td>Smethurst</td>
<td>35</td>
<td>5</td>
<td>7</td>
<td>30</td>
</tr>
<tr>
<td>434-26-3751</td>
<td>Guldu</td>
<td>35</td>
<td>5</td>
<td>7</td>
<td>32</td>
</tr>
<tr>
<td>612-67-4134</td>
<td>Madayan</td>
<td>35</td>
<td>8</td>
<td>10</td>
<td>40</td>
</tr>
</tbody>
</table>

- Bar_sells

  Bars(bar_name, address) Sells(bar_name, beer_name, price)

Transitive Dependencies

- Violating FD involves attribute(s) depending on non-key attribute(s)

Hourly_Emps violating FD: R → W
Partial Dependencies

• Violating FD involves attribute(s) depending on attribute(s) that are proper subset of a key

Repeated Decomposition

• Repeated decomposition
  – May be necessary to yield set of relations that are in BCNF
  – Okay to confirm BCNF for original relation R using only FDs F, but each decomposed relation R_i must be checked for violating each [relevant] FD in F^+

  • Using attribute closure to check decomposed R_i
    – For each subset of attributes α in R_i, check that α^+ (under F):
      • Contains no attributes of R_i – α, or
      • Contains all attributes of R_i

Exercise 3: BCNF Decomposition

• Candidate key={id, advisorId}
• FD violation? Both!
• Decomposed into three relations:
  – R1 = {id, name, dorm}
  – R2 = {advisorId, advisorName}
  – R3 = {id, advisorId}

An Aside: Multiple Candidate Keys

• For relation Bars(bar_name, address), suppose we knew:
  – bar_name → address
  – address → bar_name

  Either attribute could serve as primary key!

• When creating a relation in SQL, use one candidate key as the primary key
  – Enforce others using UNIQUE key word
  – Commonly used when use surrogate key as a primary key
Dependency Preservation

- Decomposed example from “Oversees”:
  - $E \rightarrow P$ (an employee oversees only one project)
  - $D \rightarrow P$ (a dept works on only one project)
  - $E \rightarrow D$ (an employee only works with one dept for these projects)

- Can we still check $E \rightarrow P$? (an employee oversees only one project)

<table>
<thead>
<tr>
<th>Project</th>
<th>Department</th>
</tr>
</thead>
<tbody>
<tr>
<td>Comet</td>
<td>Physics</td>
</tr>
<tr>
<td>Comet</td>
<td>Astronomy</td>
</tr>
<tr>
<td>Genomics</td>
<td>Biology</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Department</th>
<th>Employee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physics</td>
<td>Alice</td>
</tr>
<tr>
<td>Astronomy</td>
<td>Bob</td>
</tr>
<tr>
<td>Biology</td>
<td>Carl</td>
</tr>
<tr>
<td>Biology</td>
<td>Denise</td>
</tr>
</tbody>
</table>

Dependency Preserving Decomposition

- **Dependency preserving decomposition** (Intuition):
  - If $R$ is decomposed into $X$, $Y$ and $Z$, and we enforce the FDs that hold individually on $X$, on $Y$ and on $Z$, then all FDs that were given to hold on $R$ must also hold

- The **projection of $F$ on attribute set $X$** (denoted $F_X$):
  - the set of FDs $U \rightarrow V$ in $F^+$ (closure of $F$, not just $F$!) such that all of the attributes on both sides of the FD are in $X$
  - *That is: $U$ and $V$ are subsets of $X*

Exercise 4: Dependency Preservation

Note that the closure of $F$ ($F^+$) also includes $A \rightarrow C$

a. Lossless. They intersect on attribute B and B is a key for BC. Also dep. preserving.
   $F_{AB} = \{ A \rightarrow B \}, F_{BC} = \{ B \rightarrow C \}$ so we see $(F_{AB} \cup F_{BC})^+ = F^+$

b. Lossless. They intersect on attribute A and A is a key for AB. Not dep. preserving
   $F_{AB} = \{ A \rightarrow B \}, F_{AC} = \{ A \rightarrow C \}$. No way to get $B \rightarrow C!$
Exercise 5: Movie showings

a. \{movie, city\} and \{movie, theater\}
b. Theater → city
c. Can’t preserve the FD movie, city → theater

Movie showings: decomposition issue

- Showings (movie, theater, city)
- FDs
  - movie, city → theater
  - theater → city

Viola Sng # FD!

Decompose...

Above decomposition could allow this to happen!

Violates FD movie, city → theater

Third Normal Form (3NF)

- Definition: for all \( X \rightarrow A \) in \( F^+ \)
  - \( A \in X \) (called a trivial FD), or
  - \( X \) is a superkey for \( R \), OR
  - \( A \) is a part of some candidate key for \( R \)
- Allows FDs like non-key → partial key
- 3NF but not BCNF?
  - have overlapping composite candidate keys

Always possible to get a loss-less, dependency-preserving decomposition into 3NF!

(may contain redundancy)

Alternate Formulation of 3NF & BCNF

Every non-key attribute must describes a fact about “the key, the whole key, and nothing but the key, so help me Codd”

- Normal forms increasingly restrictive
  - 1st NF ⊇ 2nd NF ⊇ 3rd NF ⊇ Boyce-Codd NF
Refining an ER Diagram

- 1st diagram becomes:
  \[ \text{Workers}(S,N,L,D,Si) \]
  \[ \text{Departments}(D,M,B) \]
  
  - Lots associated w/ workers

- Suppose all workers in a dept are assigned the same lot: \( D \rightarrow L \)

- Redundancy; fixed by:
  \[ \text{Workers}_2(S,N,D,Si) \]
  \[ \text{Dept}_{\text{Lots}}(D,L) \]
  \[ \text{Departments}(D,M,B) \]

- Can fine-tune this:
  \[ \text{Workers}_2(S,N,D,Si) \]
  \[ \text{Departments}(D,M,B,L) \]

Redundancy Despite BCNF

- A teacher T can teach course C, and book B is a recommended text for the course
  
  - Schema: \((\text{course}, \text{teacher}, \text{book})\)
  
  - Any functional dependencies? BCNF?

- Suppose first two tuples exist... what other tuples “should” exist?

<table>
<thead>
<tr>
<th>Course</th>
<th>Teacher</th>
<th>Book</th>
</tr>
</thead>
<tbody>
<tr>
<td>Physics 101</td>
<td>Green</td>
<td>Mechanics</td>
</tr>
<tr>
<td>Physics 101</td>
<td>Brown</td>
<td>Optics</td>
</tr>
<tr>
<td>Physics 101</td>
<td>Green</td>
<td>Optics</td>
</tr>
<tr>
<td>Physics 101</td>
<td>Brown</td>
<td>Mechanics</td>
</tr>
</tbody>
</table>

Idea: course books are independent of course instructors

Also should exist

Redundancy?