CSC 443 – Database Management Systems

Lecture 5 – Normalization

Normalization

• We want our database to be a clear representation of the data, its relationships and constraints
• We can identify relationship using a technique called normalization.
• Normalization is a bottom-up technique where we examine the relationship between attributes and reconfigure the tables accordingly.
Purpose of Normalization

• Characteristics of a suitable set of relations include:
  – the minimal number of attributes necessary to support the data requirements of the enterprise;
  – attributes with a close logical relationship are found in the same relation;
  – minimal redundancy with each attribute represented only once with the important exception of attributes that form all or part of foreign keys.

Purpose of Normalization

• The benefits of using a database that has a suitable set of relations is that the database will be:
  – easier for the user to access and maintain the data;
  – take up minimal storage space on the computer.
How Normalization Supports Database Design

- Major aim of relational database design is to group attributes into relations to minimize data redundancy.

Data Redundancy and Update Anomalies

- Use top-down approach such as ER modeling
- ER model is mapped to a set of relations
- Use normalization as a validation technique to check structure of relations
- Use normalization as a bottom-up technique to create set of relations

Approach 1

Approach 2

Set of well-designed relations

Sources describing the enterprise such as data dictionary and corporate data model

Forms/reports that are used or generated by the enterprise

Users' requirements specification

Users
Data Redundancy and Update Anomalies

• Potential benefits for implemented database include:
  – Updates to the data stored in the database are achieved with a minimal number of operations thus reducing the opportunities for data inconsistencies.
  – Reduction in the file storage space required by the base relations thus minimizing costs.

Data Redundancy and Update Anomalies

• Problems associated with data redundancy are illustrated by comparing the Staff and Branch relations with the Staff-Branch relation.
Data Redundancy and Update Anomalies

- Problems associated with data redundancy are illustrated by comparing the Staff and Branch relations with the Staff-Branch relation.
Data Redundancy and Update Anomalies

- Staff-Branch relation has redundant data; the details of a branch are repeated for every member of staff.
- In contrast, the branch information appears only once for each branch in the Branch relation and only the branch number (branchNo) is repeated in the Staff relation, to represent where each member of staff is located.

Data Redundancy and Update Anomalies

- Relations that contain redundant information may potentially suffer from update anomalies.
- Types of update anomalies include
  - Insertion
  - Deletion
  - Modification
Our Example

• The DreamHome Customer Rental Details form holds details about property rented by a given customer.
  – To simplify things, we will assume that a renter rents a given property once and only one property at a time.

<table>
<thead>
<tr>
<th>CustNo</th>
<th>Cname</th>
<th>PropNo</th>
<th>PAddr</th>
<th>RntSt</th>
<th>RntFnsd</th>
<th>Rent</th>
<th>OwnerNo</th>
<th>OName</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>John Kay</td>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>7/1/10</td>
<td>8/31/06</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>9/1/06</td>
<td>9/1/08</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>9/1/02</td>
<td>6/10/04</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PG36</td>
<td>2 Manor Rd, Scarsdale</td>
<td>8/1/04</td>
<td>12/1/05</td>
<td>750</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td></td>
<td></td>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>8/1/06</td>
<td>9/1/10</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
</tbody>
</table>
First Normal Form (1NF)

• Unnormalized – A table with one or more repeating groups.
• First Normal Form (1NF) – A relation in which the intersection of each row and column contains one and only one value

Repeating Groups

• Any collection of attributes that repeat provides a complication for a database, both in terms of storing it (how many repeating groups would you allow for) as well as querying them.
• It is necessary to recognize them so we can eliminate them.
• E.g.,
  Repeating Group = (Property_no, Paddress, RentStart, RentFinish, Rent, Owner_No, OName)
## Our Table in 1NF

<table>
<thead>
<tr>
<th>CustNo</th>
<th>CName</th>
<th>PropNo</th>
<th>PAddr</th>
<th>RntSt</th>
<th>RntFnsh</th>
<th>Rent</th>
<th>OwnerN o</th>
<th>OName</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>John Kay</td>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>7/1/10</td>
<td>8/31/06</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td>CR76</td>
<td>John Kay</td>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>9/1/06</td>
<td>9/1/08</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>9/1/02</td>
<td>6/10/04</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
<td>PG36</td>
<td>2 Manor Rd, Scarsdale</td>
<td>1/1/04</td>
<td>12/1/05</td>
<td>750</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>8/1/06</td>
<td>9/1/08</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
</tbody>
</table>

## Candidate Keys

- A candidate key for a given table is
  - unique (only one row has that value or combination of values)
  - irreducible (there is no subset of the candidate that is unique).
- Our candidate keys are:
  - (Customer_No, Property_No)
  - (Customer_No, RentStart)
  - (Property_No, RentStart)
The Customer_Rental Relation

Customer_Rental(Customer_No, Property_No, Cname, Paddress, RentStart, RentFinish, Rent, Owner_No, Oname)

Primary Key Fields

Functional Dependency

- If A and B are attributes of Relation R, B is functionally dependent on A (A→B) if each value of A is associated with one and only one value of B.
- B is fully functionally dependent on A if B is functional dependent on A and not on a proper subset of A.
- B is partially functionally dependent on A if there is some attribute that can be removed from A and the dependence still holds.
Listing All The Functional Dependencies

1. Cust_No, Prop_no → RentStart, RentFinish (Primary Key)
2. CustNo → Cname (Partial Dependency)
3. Prop_no → Paddress, Rent, Owner_No, Oname (Partial Dependency)
4. Owner_No → Oname (Transitive Dependency)
5. CustNo, RentStart → PropNo, Paddress, RentFinish, Rent, Owner_No, Oname (Candidate Key)
6. Prop_No, RentStart → CustNo, Cname, RentFinish (Candidate Key)

Functional Dependencies in Graphical Form

- CustNo → (Primary Key)
- PropNo → (Partial Dependency)
- CName → (Partial Dependency)
- PAddr → (Partial Dependency)
- RentStart → (Partial Dependency)
- RentFinish → (Partial Dependency)
- Rent → (Partial Dependency)
- OwnerNo → (Transitive Dependency)
- OName → (Candidate Key)
- CustNo → (Candidate Key)
- Cname → (Candidate Key)
Functional Dependency in Our Table

• We have three relations with the following functional dependencies:
  – CustNo, PropNo → RentStart, RentFinish
  – CustNo → CustName
  – PropNo → Paddress, Rent, OwnerName, Oname
• Therefore, we have:
  – Customer(CustNo, Cname)
  – Rental(CustNo, PropNo, RentStart, RentFinish)
  – Property_Owner(PropNo, Paddress, Rent, OwnerNo, OName)

Second Normal Form (2NF)

• A relation is in 2NF if it is in 1NF and every non-primary key attribute is fully functionally dependent on the primary key
Our Database in 2NF

Customer Relation

<table>
<thead>
<tr>
<th>CustNo</th>
<th>CName</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>John Kay</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
</tr>
</tbody>
</table>

Rentals Relation

<table>
<thead>
<tr>
<th>CustNo</th>
<th>PropNo</th>
<th>RentStart</th>
<th>RentFinish</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>PG4</td>
<td>7/1/10</td>
<td>8/31/06</td>
</tr>
<tr>
<td>CR76</td>
<td>PG16</td>
<td>9/1/06</td>
<td>9/1/08</td>
</tr>
<tr>
<td>CR56</td>
<td>PG4</td>
<td>9/1/02</td>
<td>6/10/04</td>
</tr>
<tr>
<td>CR56</td>
<td>PG36</td>
<td>1/1/04</td>
<td>12/1/05</td>
</tr>
<tr>
<td>CR56</td>
<td>PG16</td>
<td>8/1/06</td>
<td>9/1/10</td>
</tr>
</tbody>
</table>
Our Database in 2NF

Property-Owner Relation

<table>
<thead>
<tr>
<th>PropNo</th>
<th>PAddr</th>
<th>Rent</th>
<th>OwnerNo</th>
<th>OName</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>700</td>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td>PG36</td>
<td>2 Manor Rd, Scarsdale</td>
<td>750</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
<tr>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>900</td>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
</tbody>
</table>

Transitive Dependency

- If A, B, and C are attributes of a relation R such that if A → B and B → C, then C is transitively dependent on A via B.
Third Normal Form

- A relation is in 3NF if it is 2NF and there are no non-primary-key attributes that are transitively dependent on the primary key.

Functional Dependencies in 2NF

- Customer
  - CustNo → Cname

- Rental
  - CustNo, PropNo → RentStart, RentFinish
  - PropNo, RentStart → CustNo, RentFinish

- PropertyOwner
  - PropNo → Paddr, Rent, OwnerNo, OName
  - OwnerNo → Oname (Oname is not f.d. on PropNo)
Our 3NF Relations

We have 4 relations:

• Customer(CustNo, Cname)
• Rental(CustNo, PropNo, RentStart, RentFinish)
• Property_For_Rent(PropNo, Paddress, Rent, OwnerNo)
• Owner(OwnerNo, OName)

Our Database in 3NF

<table>
<thead>
<tr>
<th>CustNo</th>
<th>CName</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>John Kay</td>
</tr>
<tr>
<td>CR56</td>
<td>Aline Stewart</td>
</tr>
</tbody>
</table>
Our Database in 3NF

Rentals Relation

<table>
<thead>
<tr>
<th>CustNo</th>
<th>PropNo</th>
<th>RentStart</th>
<th>RentFinish</th>
</tr>
</thead>
<tbody>
<tr>
<td>CR76</td>
<td>PG4</td>
<td>7/1/10</td>
<td>8/31/06</td>
</tr>
<tr>
<td>CR76</td>
<td>PG16</td>
<td>9/1/06</td>
<td>9/1/08</td>
</tr>
<tr>
<td>CR56</td>
<td>PG4</td>
<td>9/1/02</td>
<td>6/10/04</td>
</tr>
<tr>
<td>CR56</td>
<td>PG36</td>
<td>1/1/04</td>
<td>12/1/05</td>
</tr>
<tr>
<td>CR56</td>
<td>PG16</td>
<td>8/1/06</td>
<td>9/1/10</td>
</tr>
</tbody>
</table>

Our Database in 3NF

Property-for-Rent Relation

<table>
<thead>
<tr>
<th>PropNo</th>
<th>PAddr</th>
<th>Rent</th>
<th>OwnerNo</th>
</tr>
</thead>
<tbody>
<tr>
<td>PG4</td>
<td>6 Lawrence St, Elmont</td>
<td>700</td>
<td>CO40</td>
</tr>
<tr>
<td>PG16</td>
<td>5 Nova Dr, East Meadow</td>
<td>900</td>
<td>CO93</td>
</tr>
<tr>
<td>PG36</td>
<td>2 Manor Rd, Scarsdale</td>
<td>750</td>
<td>CO93</td>
</tr>
</tbody>
</table>
Our Database in 3NF

Owner Relation

<table>
<thead>
<tr>
<th>OwnerNo</th>
<th>OName</th>
</tr>
</thead>
<tbody>
<tr>
<td>CO40</td>
<td>Tina Murphy</td>
</tr>
<tr>
<td>CO93</td>
<td>Tony Shaw</td>
</tr>
</tbody>
</table>