Database Design - Section 7

Instructor Guide
Lesson 1 - Review of Unique Identifiers

Lesson Preparation
In this lesson, we organize and add to what we’ve learned about UIDs so far -- artificial UIDs, simple vs. composite UIDs, barred relationships, and secondary UIDs.
Examples of each will be given, and students will be asked to revisit the Video Store ERD to identify UIDs.

What to Watch For
Make sure students understand that all UIDs are mandatory and that all parts of a UID are mandatory.
In the diagram, they can be denoted with #* or only with #. If a relationship participates in a UID, then it must be mandatory -- note that the barred side of a relationship is always mandatory.

Connections
None.
What Will I Learn?

In this lesson, you will learn to:

- Review the different types of unique identifiers (UIDs)
- Apply the rules of UIDs and identify UIDs for entities in an existing model
- Analyze business rules and justify the creation of an artificial UID, a composite UID, or a secondary UID
Why Learn It?

The unique identifier (UID) is very important in relational databases. It is the value or combination of values that enables the user to find that one unique item among all the rest. Identifying just the right attribute, combination of attributes, and/or relationships is a skill that any database designer must master. The unique identifier is the way to find your record in a file, a particular card in a deck of cards, a specific piece of data in a database, and your package in a warehouse.

Why Learn It?

You can supplement with your own examples of the importance of a UID:
How do you find your backpack in a pile of bags?
How do you make sure that no one but you can unlock your locker?
How do you find your grade on a posted list?
Some students may suggest a combination of first name, last name, and address. This would probably be unique.

However, it is not as convenient as a single ID. Think of the business requirements: if you went to the library, or checked out equipment at the gym, or asked for a copy of your grades, -- is it easier to just provide a student ID or to provide name and address each time?
Tell Me / Show Me

Ask students why date of performance or seat number alone does not identify a concert ticket. On any given date, there are many seats available. The same seat number is sold for many different dates.
Point out that it’s possible for two people to have the same account number, but at different banks.

Bank-to-bank transfers will always need the bank routing number and the customer account number.
In this example, the UID of PLAY LIST ITEM comes from EVENT and SONG. What tells you this? (The bars on the relationships)

Note: It is possible for an intersection entity to use an artificial attribute as the UID, instead of the barred relationships to the originating entities. For example: Each MANUFACTURER may produce one or more PRODUCTS (shoes, shirts, jeans, etc.). Each PRODUCT may be produced by one or more MANUFACTURER (Nike shoes, Adidas shoes, Levi’s jeans, Calvin Klein jeans, etc.). CATALOG ITEM resolves this many-to-many relationship. An item in a catalog can be uniquely identified by the manufacturer number and the product code. However, think of the business application. When a customer orders from a catalog, he/she does not provide the manufacturer number/name or the product code. A customer provides a catalog item number. Because of the business rules, number is created as the UID of the CATALOG ITEM entity.

The relationships are not barred, and catalog number is drawn with “#” before it.
Secondary UIDs can be useful as an alternative means of searching for data.

Examples:
- You forgot your student ID, but you have your badge with you.
- You have a frequent-shopper card at the supermarket but did not bring it with you. Some systems can find your information if you provide your telephone number.
Try It / Solve It

Assessment:
The solutions here do not cover all the possible answers. This is another opportunity for your students to be creative as they think of situations that will and will not work with the proposed UID. In all cases, students may opt to just create an artificial ID. This is valid. However, encourage them to find attributes in the entities first, or create related entities that can help compose a “real” UID.

A City
Name and country is not always sufficient. In some countries, you would also need province or state because there will be multiple cities with the same name within one country. Some countries use a unique postal code per town or city. Geographical coordinates, such as latitude and longitude, are unique but may not be very practical for the business that is requesting the design.

A Contact Person for a Customer
Name, job (or telephone number), and customer ID are probably enough.

A Train
Trains often have a unique ID and name. You could also use the time, the originating point, and the destination. The railroad company uses a code for internal use.

Assessment:
A Road
Most important roads have a name, such as Route 60 or I95. Unimportant roads might not have a name. In this case, geographical information (such as both ends of the road) can be used as the UID.
A Financial Transaction
Use the combination of from account, to account, date, and time. You need time because there may be several transactions to and from the same account on the same date.
An Award
In the case of a movie award (such as an Oscar or a Cesar), you can identify it by the recipient, the year, the movie, and the category (cinematography, supporting male role, etc.).
A Painting
Not all paintings are identified by title, painter, and year. Museums will often create an ID.
Some students may not understand why STAR BILLING has no attributes. Remind them that this is valid, as long as it has a UID (the barred relationships from MOVIE and ACTOR). You can also suggest that they come up with appropriate attributes, such as review or comments.
Some students may not understand why STAR BILLING has no attributes. Remind them that this is valid, as long as it has a UID (the barred relationships from MOVIE and ACTOR). You can also suggest that they come up with appropriate attributes, such as review or comments.
Lesson Preparation
None.

What to Watch For
Go back to the Global Foods ERD and look at all entities. Make sure they are in 3NF.

Connections
Ask students to recall the rules of First and Second Normal Form:
First Normal Form (1NF)
First Normal Form requires that there be no multivalued attributes and no repeating groups. To check for First Normal Form, validate that each attribute has a single value for each instance of the entity.
Second Normal Form (2NF) requires that any non-UID attribute be dependent on the entire UID.
This means that all attributes that are not part of the entity’s UID should be dependent on the whole UID. This specifically applies to entities that have a UID that is composed of more than one attribute or a combination of attribute(s) and relationship(s).

Explain how these three rules together allow them to achieve a normalized design -- the ultimate goal of normalization.
What Will I Learn?

In this lesson, you will learn to:

- Identify transitive dependencies in a data model
- List and define the rules of Third Normal Form of the normalization process
- View a nonnormalized entity and determine which rule (or rules) of normalization are being violated
- Apply the rule of Third Normal Form to solve a violation in the model
- Recognize and discuss the issues of identification in the real world
Why Learn It?

Your goal as a database designer is to "store information in one place and the best place." Following the rules of normalization helps you achieve this goal.

You may want to enter different kinds of information for a friend in your personal address book: phone number, address, favorite movie, name of school or place of work. If you have several friends who go to the same school, and you enter the school's street address along with each of them, you would not only be duplicating data but causing potential problems -- for instance, if the school moved and changed its address, you would have to go back and change it everywhere!

Normalization is a process to eliminate these kinds of problems.

It is useful to understand the difference between identification in the real world and identification in the database world. This helps you recognize the need for creating UIDs that meet the business requirements.
Tell Me / Show Me

Point out that in the first model, if store address changed, you would have to change the information on all the CDs that were bought at the store. In the second, normalized model, you would just change it in the one instance of the STORE entity.
Another way to look at the rule of third Normal Form is: Attributes can’t have attributes of their own. In the first model, the attribute state has an attribute -- state flower. This is a violation of Third Normal Form.
Tell Me / Show Me

You may need to explain to students what is meant by "dependent." Give them an example of what information would constitute an instance of EMPLOYEE. For example, your mother is an employee, and she has a name, an address, and a birthday. Your father can be listed as the partner.

Point out that the 1:1 relationship is optional on the EMPLOYEE end because some employees will not have partners. It is mandatory on the PARTNER end because information about a partner is tracked only if the partner is a dependent. If the partner is in the system, the partner must be a dependent.
Tell Me / Show Me

Point out that the DJ is talking about Elvis, while the teenager is thinking about a dog, who is also named Elvis.

In a database, we select a unique identifier so we can distinguish between the two. In everyday life, it is easier to confuse them in a casual conversation. This is very common if the people talking are from different backgrounds and are not thinking in the same context.

Try addressing the students by calling out their student numbers instead of their names. This is a great way to introduce the identification in the database (school records) and the real world.
Try It / Solve It

Paint color and interior color are attributes of color scheme.
Create a new entity for color scheme and create a relationship to CAR.
This can be tricky because there are two transitive dependencies: team coach depends on team, and agent commission depends on agent. Two new entities have to be created.

**Assessment:**

Team coach is an attribute of team. Agent commission is an attribute of agent. Create an entity TEAM and an entity AGENT, both with relationships to ATHLETE.
Try It / Solve It

The goal of this activity is to elicit discussion and have students think about the issue of personal identification and its implementation in a database. (Define a measurable outcome for the students: have them work in groups and write up their responses, or do Internet research in groups or alone, share ideas, and write up responses.)

This issue could be a "hot" button for some students. The Internet has many articles about national ID cards and DNA profiling (use the keywords: "identity cards," "national ID cards," "DNA profiling"). Encourage students to find data instead of just voicing unsubstantiated opinions. Possible responses for pros/cons might include:

**CONS**
- Potential loss of personal freedom
- Use of database for other purposes, such as tracking buying patterns (e.g., book purchases) of individuals who belong to certain religious groups (this is a violation of privacy)

**PROS**
- Reduce identity theft
- Reduce the number of pieces of identification one would need
Lesson 3 - Arcs

Lesson Overview

<table>
<thead>
<tr>
<th>Section</th>
<th>Groupings</th>
<th>Timings</th>
</tr>
</thead>
<tbody>
<tr>
<td>What Will I Learn?</td>
<td>Whole Class</td>
<td>1 minute</td>
</tr>
<tr>
<td>Why Learn It?</td>
<td>Whole Class</td>
<td>4 minutes</td>
</tr>
<tr>
<td>Tell Me / Show Me</td>
<td>Whole Class</td>
<td>15 minutes</td>
</tr>
<tr>
<td>Try It / Do It</td>
<td>Small Groups</td>
<td>30 minutes</td>
</tr>
</tbody>
</table>

Lesson Materials

- None

Vocabulary

- Arc
- Constraint
- Mutually exclusive relationship
- Exclusive OR relationship

Lesson Preparation

None.

What to Watch For

Some students will be confused as to when to use an arc vs. a supertype/subtype:

Use supertypes/subtypes when you want to represent classifications or types of things. For example: STUDENT (graduating, nongraduating).

Use arcs when you want to represent mutually exclusive relationships between entities. For example: STUDENT may attend a UNIVERSITY or a TRADE SCHOOL. A UNIVERSITY or a TRADE SCHOOL may be attended by one or more STUDENT. It is not a type of STUDENT.

It is also a M:1 relationship, which therefore would not be suited to a supertype/subtype construct.
Connections
None.
What Will I Learn?

When discussing arcs, point out that they can also be modeled as supertype/subtypes. Supertypes/subtypes are best used when you are trying to classify within an entity (e.g., university student: full-time, part-time). Arcs should be used when the exclusive relationships between the entities are more important to the business. The lesson will show examples of these.
Why Learn It?

Arcs in data modeling help designers clarify an exclusive OR across relationships. The more explicitly you can define the client's requirements, the more accurate your final implementation will be.
Point out the arc in the diagram. It represents the exclusive relationship across billboard and movie, product advertisement and public announcement. The arc tells the reader of the diagram that only one of these “features” will have the relationship with BILLBOARD for each instance. The optional nature of the relationships indicates that each BILLBOARD may be empty -- that is, it may not have any of the “features.”
Tell Me / Show Me

Tell Me / Show Me

The DJ business has a mutually exclusive relationship between EVENT and VENUE.
Tell Me / Show Me

Tell Me / Show Me

Tell Me / Show Me
Read the ERDish. The arc represents the exclusive OR relationship -- each EVENT must be held at one PRIVATE HOME or must be held at one PUBLIC SPACE, but not at the same time.

Point out that each entity has its own attributes. Each PUBLIC SPACE may be the venue for one or more EVENTs. Each PRIVATE HOME may be the venue for one or more EVENTs. Some students may ask why not just model one entity and make the rental fee optional? Answer: Although that may well be the decision made when the database is built, at this conceptual level, we want to capture all the business rules as closely as possible. The rental fee is mandatory for a PUBLIC SPACE so we want our model to reflect that, as well as the exclusive OR relationship between EVENT and VENUE.
Tell Me / Show Me

This would actually be a good choice here, since there are many common attributes between the entities PRIVATE HOME and PUBLIC SPACE.
Tell Me / Show Me

Note that when a supertype/subtype is redrawn as an arc, the relationships in the arc are mandatory 1:1 is/is relationships. Each PARTNER must be an EVENT PLANNER, or a DJ or a PROJECT MANAGER, or an OTHER. Each EVENT PLANNER must be a PARTNER, each DJ must be a PARTNER, each MANAGER must be a PARTNER, each OTHER must be a PARTNER. This is why supertypes and subtypes are not usually redrawn as arcs, but arcs can be often redrawn as supertypes and subtypes.
Try It / Solve It

Activity: Global Fast Foods Menu

You will be adding to the Global Fast Foods ERD. Here is what we’ve modeled so far:

This is meant to bring the students up to date on what has been modeled for the Global Fast Foods ERD so far.

It will also help students who did not arrive at an accurate solution for the previous activity on Global Fast Foods.
Try It / Solve It

Artificial Fast Food Menus

Read the following interview notes with the Global Fast Foods owner. Refine the ERD accordingly.

"We just started introducing a promotional menu. This menu features food items that are not available on the regular menu. It's a way for us to test new items and to take advantage of special events (holidays, etc.) and commercial offerings (for example, when the movie 'King Kong' was out, we offered oversized Kong burgers). Sometimes we have a little gift associated with a promotional item. For the Chinese New Year, we featured a moosecone on the promotional menu, and everyone who ordered it got a free toy dragon, because it was the year of the dragon. Each promotional menu has a theme, such as Back to School or Summer Barbecue, and it has a start and end date. There is only one promotion in effect at any given time.

"Our regular menus contain those items that customers expect when they come to Global Fast Foods. Right now we have two types: a breakfast menu, available from 6 a.m. to 11 a.m., and a lunch menu, available from 11:30 a.m. until closing time. We're still considering a separate dinner menu, but we'll have to test that first."

Try It / Solve It

One acceptable solution would be to have three entities in the arc:

PROMOTIONAL MENU
BREAKFAST MENU
LUNCH MENU

BREAKFAST MENU and LUNCH MENU would have the same attributes -- code and hours served.
Activity: Tickets

A show ticket is purchased from an agent, the box office, or the Internet. A ticket has a description, an event, a date and a price. An agent has a name and a phone number. The box office has an address and a phone number. The Internet has a URL address.

Draw the entities and represent the exclusive relationship.
**Graphic:**

```
TICKET
   # number
   * event description
   * date
   * price

purchased from
the seller of

AGENT
   # name
   # phone number

purchased from
the seller of

BOX OFFICE
   # address
   * phone number

purchased from
the seller of

INTERNET
   # url
```
Try It / Solve It

Assessment:

One acceptable solution would be to have four subtypes of MENU (BREAKFAST, LUNCH, PROMOTIONAL, OTHER). The type attribute would not be needed in the BREAKFAST and LUNCH subtypes.
Lesson Preparation

None.

What to Watch For

Be sure the class understands that hierarchy and recursive relationships essentially model the same structure. Be prepared to answer questions about pros and cons of each one.

PRO Hierarchical: Hierarchical structures are more explicit and are easier for most people to understand because they are very similar to an organizational chart. Each entity can have its own mandatory attributes and relationships, if the business requires this (instead of all optional attributes and relationships, as you would have in a recursive). In this way, your data model truly reflects the business rules.

PRO Recursive: Recursive relationships tend to be simpler because you are using only one entity. Your diagram will be less “busy.” However, they are less specific -- you cannot have mandatory attributes or relationships unless they are mandatory in all instances of the entity.

Another thing to consider when modeling these types of relationships is how often the structure changes. If it changes often, then a recursive relationship is easier to maintain. If it is fairly fixed, then you may consider the more explicit hierarchy.
A structure that doesn’t change very much would be a building with suites and floors and rooms. This is a good case for using an explicit hierarchy. On the other hand, a company that reorganizes frequently (going from a fairly flat organizational structure to one with many levels and vice versa) would probably be better modeled with a recursive relationship.

**Connections**

**Go back to the ERD with ROOM, SUITE, FLOOR, and BUILDING. Ask about the transferability of the relationships. Can a floor be moved from one building to another? How should the ERD be modified?**

**Answer:** These are fairly permanent physical structures. Although walls could be torn down or built up to reorganize rooms into different suites, a suite is not likely to move up or down a floor. Similarly, a floor cannot physically relocate to another building. The relationships between BUILDING and FLOOR and between FLOOR and SUITE are nontransferable and should be represented with a diamond. The business rules will determine whether the relationship between ROOM and SUITE is transferable or not.
What Will I Learn?

In this lesson, you will learn to:

- Define and give an example of a hierarchical relationship
- Diagram the UID relationships in a hierarchical model
- Identify business examples of recursive relationships
- Diagram a recursive relationship given a scenario
- Construct a model using both recursion and hierarchies to express the same conceptual meaning
Why Learn It?

Ask students to name other hierarchical structures.
Answers could include: church, government, school
Tell Me / Show Me

Explain that one way to model a hierarchy would be to create an entity for each level with a relationship to the next level.

The UIDs would probably be an employee ID or number for each entity.
Tell Me / Show Me

Here you have a case of the cascading UIDs: the UID of FLOOR is the combination of FLOOR number and the BUILDING id, the UID of SUITE is the combination of SUITE number and the FLOOR number and the BUILDING id, and the UID of ROOM is the combination of ROOM id and SUITE number and FLOOR number and the BUILDING id. It is important to have the students realize this.

This may be convenient in the sense that the UID of a ROOM will also tell you the SUITE and FLOOR and BUILDING that it’s in. However, this makes for a rather long UID. Unique independent, artificial codes may be more practical. Also, if the hierarchical structure changes often, use artificial identifiers. (If we were to add a level called APARTMENT in between FLOOR and SUITE, think of how that would affect the UIDs of all SUITEs and ROOMs!)
Tell Me / Show Me

Explain that the mandatory attributes specific to an entity (bonus plan, car plan, budget) become optional in the recursive model.

Why is the recursive relationship fully optional?

Answer: Otherwise, the hierarchy does not have a top or bottom. Ask students who manages the president? No one! That is why it must be fully optional. One instance of the entity does not have a manager. Hence, it is optional in the recursive model.

The ER diagramming convention that shows a recursive relationship is known as a "pig’s ear." The loop can appear on any side of the entity’s softbox.
Tell Me / Show Me

In the DJ model, the project manager has overall responsibility for an event and manages the other employees (event planner, DJ) working on the event. We chose to represent the hierarchy with a recursive relationship.

Tell Me / Show Me

Explain to students that, according to the relationship, only managers can supervise other employees.
Try It / Solve It

Activity: Sales Regions and Salespeople

Develop two ER diagrams to represent the following situation. Develop one using a hierarchical structure and one using a recursive structure.

Our company sells products throughout the United States. So we've divided the U.S. into four major sales regions: the Northern, Eastern, Southern, and Western regions. Each sales region has a unique region code. Each sales region is then divided into sales districts. For example, the Western region is divided into the Rocky Mountain, Northwest, Pacific Coast, and Pacific districts. Each district has a unique district code. Each district is made up of sales territories. The Rocky Mountain district is composed of three territories: Wyoming-Montana, Colorado, and Utah-New Mexico. The Northwest district is made up of two territories: the Washington and Oregon-Idaho territories. The Pacific Coast district is composed of two territories: the California and Nevada territories. The Pacific District includes the Hawaii territory and the Alaska territory. Each territory has a unique territory code.

Try It / Solve It

You may want to explain "quota" as an amount of sales that the salesperson must achieve.

A lot of students will also include hierarchical relationships from SALESPEOPLE, SALES MANAGER, and SALES DIRECTOR (Each SALESPEOPLE must be reporting to one and only one SALES MANAGER, each SALES MANAGER may be the manager of one or more SALESPEOPLE, etc.).
Try It / Solve It

Activity: Sales Regions and Salespeople

Then each sales territory is broken down into sales areas. For example, California is made up of two sales areas: the Front Range and the Western Slope sales areas. Each sales area has a unique sales area code.

Each salesperson is responsible for one or more sales areas and has a specific sales quota. We also have sales managers who are responsible for one or more sales districts, and sales directors who are responsible for one or more sales regions. Each sales manager is responsible for the territories within his/her district. We don't overlap our employees' responsibilities. Each sales area is always the responsibility of a single salesperson, and our managers' and directors' responsibilities don't overlap. Sometimes our salespeople, managers, and directors will have special assignments and will not be responsible for sales. We identify all our sales personnel by their employee IDs.

Try It / Solve It

They may also model a recursive relationship between SALES EMPLOYEE (Each SALES EMPLOYEE may be the manager of one or more SALES EMPLOYEES. Each SALES EMPLOYEE may be managed by one and only one SALES EMPLOYEE.) These relationships are not part of the solution, because it is not explicitly stated in the business scenario. However, it is not a bad assumption to make. Just remind them that if they make assumptions, they have to eventually check them with the customer.

Recursive Structure
Try It / Solve It

Activity: Global Fast Foods Staff Manager

You will be adding to the Global Fast Foods ERD. Here is what we’ve modeled so far:

This is meant to bring the students up to date on what has been modeled for the Global Fast Foods ERD so far. It will also help students who did not arrive at an accurate solution for the previous activity on Global Fast Foods.
Try It / Solve It

You may want to explain “target revenue” as the amount of sales that the restaurant needs to achieve.

Point out that the relationship is from MANAGER to STAFF, not STAFF to itself. This is because only a STAFF member who is a manager can supervise other employees.
Lesson 5 - Review

Lesson Preparation

Review lesson content for fifteen minutes before you administer the quiz. This is a good time to answer questions, clarify recently learned concepts and terminology, and go over some of the practice solutions from previous lessons.

What to Watch For

None.

Connections

None.
What Will I Learn?

- Demonstrate 70% mastery for Data Modeling Quiz 4
Why Learn It?

Tests give you a chance to find out how much you’ve learned and provide you with areas for improvement.
Tell Me / Show Me

### Tell Me / Show Me

**Lesson 5 - Review**

Tell Me / Show Me

<table>
<thead>
<tr>
<th>Your instructor will lead the class in a review before the quiz. This is your chance to ask additional questions, review recently learned concepts and terminology, and discuss the practice solutions from previous lessons.</th>
</tr>
</thead>
</table>

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Copyright © Oracle, 2004. All rights reserved.

Database Design - Section 7

Page 59
Assessment:

You may want to encourage students to retake the quiz until they achieve a passing score. Or you may prefer to allow only one attempt at the quiz.

Have students work in small groups to review what they missed on the quiz. Based on what types of questions they missed, have students write out the rules (e.g., rules for subtypes). Have a member from each group present a summary back to the class about what areas their group can improve.