Objectives

Applied
- Given the specifications for a data structure, identify the tables, columns, keys, relationships, and indexes for the structure.
- Given the tables for an unnormalized database, normalize the structure to the third normal form. We’ll do this later.

Knowledge
- Give three criteria for when a column should be indexed.
- Describe referential integrity.
- Explain how Oracle uses declarative referential integrity to prevent deletion, insertion, and update problems.
- Explain how normalizing a database to the third normal form affects database performance. (Later)

Chapter 9
How to design a database

A database system is modeled after a real-world system

The six basic steps for designing a data structure
Step 1: Identify the data elements
Step 2: Subdivide each element into its smallest useful components
Step 3: Identify the tables and assign columns
Step 4: Identify the primary and foreign keys
Step 5: Review whether the data structure is normalized (later)
Step 6: Identify the indexes (later)

The data elements on the invoice document
- Vendor name
- Vendor address
- Vendor phone number
- Vendor fax number
- Vendor AR contact name
- Vendor AR contact extension
- Invoice number
- Invoice terms
- Invoice date
- Invoice total
- Item extension
- Vendor sales contact name
- Vendor sales contact extension
- Item part number
- Item quantity
- Item description
- Item unit price

An invoice that’s used to identify data elements

<table>
<thead>
<tr>
<th>Part No.</th>
<th>Description</th>
<th>Item Unit</th>
<th>Ext Price</th>
</tr>
</thead>
<tbody>
<tr>
<td>CUST123</td>
<td>Design service, Inc</td>
<td>100.00</td>
<td>1200.00</td>
</tr>
<tr>
<td>6576</td>
<td>Bearing, shaft 3000, 5000</td>
<td>70.50</td>
<td>528.50</td>
</tr>
<tr>
<td>B1123</td>
<td>Dow tubes, black, pt</td>
<td>1.00</td>
<td>678.75</td>
</tr>
<tr>
<td>627702</td>
<td>Rubber tubing, 1/4 in, 1/2</td>
<td>0.19</td>
<td>9.54</td>
</tr>
<tr>
<td>CUST234</td>
<td>Assembly, Inc</td>
<td>75.00</td>
<td>562.50</td>
</tr>
<tr>
<td>CUST147</td>
<td>Testing, Inc</td>
<td>125.00</td>
<td>250.00</td>
</tr>
<tr>
<td>Sales Tax</td>
<td>Sales Tax</td>
<td></td>
<td>240.00</td>
</tr>
<tr>
<td>Subtotal</td>
<td>Subtotal</td>
<td></td>
<td>2035.00</td>
</tr>
<tr>
<td>Total</td>
<td>Total</td>
<td></td>
<td>2035.00</td>
</tr>
</tbody>
</table>
A name that’s divided into first and last names

Vendor sales contact name
Ruben Goldberg

Vendor sales contact first name
Ruben

Vendor sales contact last name
Goldberg

An address that’s divided into its components

Vendor address
1234 West Industrial Way, East Los Angeles, California 90022

Street and number
1234 West Industrial Way
City
East Los Angeles
State
California
Zip
90022

Possible tables and columns for an A/P system

<table>
<thead>
<tr>
<th>Vendors</th>
<th>Invoices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor name</td>
<td>Invoice number*</td>
</tr>
<tr>
<td>Vendor address</td>
<td>Invoice date</td>
</tr>
<tr>
<td>Vendor city</td>
<td>Terms*</td>
</tr>
<tr>
<td>Vendor state</td>
<td>Invoice total</td>
</tr>
<tr>
<td>Vendor zip code</td>
<td>Payment date</td>
</tr>
<tr>
<td>Vendor phone number</td>
<td>Payment total</td>
</tr>
<tr>
<td>Vendor fax number</td>
<td>Invoice due date</td>
</tr>
<tr>
<td>Vendor web address</td>
<td>Credit total</td>
</tr>
<tr>
<td>Vendor contact first name</td>
<td>Account number*</td>
</tr>
<tr>
<td>Vendor contact last name</td>
<td></td>
</tr>
<tr>
<td>Vendor contact phone</td>
<td></td>
</tr>
<tr>
<td>Vendor AR first name</td>
<td></td>
</tr>
<tr>
<td>Vendor AR last name</td>
<td></td>
</tr>
<tr>
<td>Vendor AR phone</td>
<td></td>
</tr>
<tr>
<td>Terms*</td>
<td></td>
</tr>
<tr>
<td>Account number*</td>
<td></td>
</tr>
<tr>
<td>Item part number</td>
<td></td>
</tr>
<tr>
<td>Item quantity</td>
<td></td>
</tr>
<tr>
<td>Item description</td>
<td></td>
</tr>
<tr>
<td>Item unit price</td>
<td></td>
</tr>
<tr>
<td>Item extension</td>
<td></td>
</tr>
<tr>
<td>Account number*</td>
<td></td>
</tr>
<tr>
<td>Sequence number</td>
<td></td>
</tr>
</tbody>
</table>

Tables and columns for an A/P system (cont.)

<table>
<thead>
<tr>
<th>Vendors (cont.)</th>
<th>Invoice line items</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vendor AR last name</td>
<td>Invoice number*</td>
</tr>
<tr>
<td>Vendor AR phone</td>
<td>Item part number</td>
</tr>
<tr>
<td>Terms*</td>
<td>Item quantity</td>
</tr>
<tr>
<td>Account number*</td>
<td>Item description</td>
</tr>
<tr>
<td>Item unit price</td>
<td>Item extension</td>
</tr>
<tr>
<td>Account number*</td>
<td>Sequence number</td>
</tr>
</tbody>
</table>

The notation

- Data elements that were previously identified but aren’t needed are crossed out.
- Data elements that were added are displayed in italics.
- Data elements that are related to two or more entities are followed by an asterisk.

The relationships between the tables

This shows two many-to-one relationships, with a “crow’s foot” at the many end.
Two tables with a many-to-many relationship

employees
employee_id
first_name
last_name

memberships
member_id
committee_id

committees
committee_id
committee_name

- Note that R&G would show “memberships” as a
diamond implementing the many-to-many relationship
here.
- Murach is showing what tables are needed.
- Murach says the linking table doesn’t need a primary
key (pg. 294-295), but R&G (and I agree) use a PK to
make sure no duplicate links occur. See enrolled, pg. 71
for example, and in our createdb.sql.

Operations that violate referential integrity
Deleting a row from the primary key table
If the foreign key table contains one or more rows related to the
deleted row

Inserting a row in the foreign key table
If the foreign key value doesn’t have a matching primary key value
in the related table

Updating the value of a foreign key
If the new foreign key value doesn’t have a matching primary key
value in the related table

Updating the value of a primary key
If the foreign key table contains one or more rows related to the row
that’s changed

Two tables with a one-to-one relationship

employees
employee_id
first_name
last_name

employee_photos
employee_id
employee_photo

Terms to know
- Entity
- Attribute
- Instance
- Entity-relationship (ER) modeling
- Referential integrity
- Declarative referential integrity (DRI)
- Foreign key constraints