Fundamentals of Database Systems

Emily Hegge

CTU Online

CS251-1101A-03 Fundamentals of Database Systems

Phillip Goin

2/10/2012
# Contents

Project outline .................................................................................................................. 3
Description of system development life cycle ...................................................................... 4
Seven steps of SDLC ............................................................................................................ 4
Description of database design life cycle ............................................................................ 6
Description of the database design life cycle ....................................................................... 6
A description of the DDLC .................................................................................................... 6
A discussion about the three steps in database design .......................................................... 8
The entity relationship diagram ........................................................................................... 9
An entity relationship diagram ............................................................................................ 9
The logical model and normalization .................................................................................. 11
Logical data model .............................................................................................................. 11
Normalized to 3NF .............................................................................................................. 13
The Microsoft Access database .......................................................................................... 14
Your database created in Microsoft Access .......................................................................... 14
SQL SELECT statements ..................................................................................................... 17
The Microsoft Access database application .......................................................................... 19
Complete your Microsoft Access application ...................................................................... 19
Menu, forms and reports ..................................................................................................... 19
References .......................................................................................................................... 22
Project outline

Your project outline goes here for IP1
Description of system development life cycle

Seven steps of SDLC

1. Planning, feasibility study

During the planning stages the need is identified. Studies are conducted to determine what the exact needs are and set up an overview of the solution. Also at this time the cost is examined to see if it is an affordable solution. (Kay, R. 2002, Klucznik, F. 2006)

2. Systems analysis, requirements

In the systems analysis and requirements phase the goals of the project is defined. The requirements for the project are assessed and the base design is created. (Kay, R. 2002, Klucznik, F. 2006)

3. Design

At the design stage the software, database, or application (for examples) are designed in full detail. Depending on the needs of the application and the skills of the developers different applications may be written in different languages and with different tools. (Kay, R. 2002, Klucznik, F. 2006)

4. Implementation

The implementation stage is the building of the application. This is where the code is written. Throughout this phase a small portion of code is written and then checked to see if it functions properly which is continued throughout the implementation process. (Kay, R. 2002, Klucznik, F. 2006)
5. Integration and testing

During integration and testing phase the program undergoes thorough testing for every possible outcome a user may make. This insures that all areas of the program work properly and to find any bugs that may be in the program. (Kay, R. 2002, Klucznik, F. 2006)

6. Acceptance, installation, deployment

After the program has been tested and accepted it then is ready to be integrated into the business. At this point the application may be installed on all employees’ workstations. Also training takes place so that everyone knows how to use the new system. Depending on the SDLC methodology maintenance may also be in this category as well. Throughout the use of the program updates and changes may be needed to make sure that it still suits the customers’ needs and that everything continues to work properly. (Kay, R. 2002, Klucznik, F. 2006)

7. Disposal

At times a program may no longer be a feasible option. At this point it is made known that the program will be made obsolete so that its users can safeguard any data they may need to remove and switch to another program.

(Kay, R. 2002, Klucznik, F. 2006)
Description of database design life cycle

Description of the database design life cycle

A description of the DDLC

The database design life cycle is similar to the system development life cycle.

1. Planning

Planning a database requires gathering information from the customers submitting the request for the database. Also data is gathered from those who the database will affect about their needs as well. The database may affect employees, customers, vendors, and suppliers. Some of the information that needs to be gathered is the type of database needed, how much data is processed on a daily basis and the amount of people accessing the database. Some other helpful information would be what type of database is currently being used, what are some of the issues with the current database, what kind of security requirements are needed, and the budget the company has for the new database. Additional information might include the type of hardware the company has and if new hardware needs to be purchased to support the new database system.  . (Irron, S. 2009, Mahapatra, H. 2010)

2. Analysis, Requirements

After all the information is gathered and reviewed a general design is set up. There are three steps to designing a database conceptual, logical and physical modeling which will be discussed more in the next section.  . (Irron, S. 2009, Mahapatra, H. 2010)
3. Design

The developers find a design that will work best to suit the needs of the company and customers and build upon that design. The development of the database begins. (Irron, S. 2009, Mahapatra, H. 2010)

4. Implementation, Data Loading

Storage for that database may need to be created to hold the data. “These constructs usually include storage group, table spaces, data files, tables, and so on.” (Mahapatra, H. 2010) Also data is loaded into the database. Depending on the existing database the data may need to be reformatted for the new database. (Irron, S. 2009, Mahapatra, H. 2010)

5. Integration and testing

The database is then tested. Testing includes checking to make sure the system is secure, making sure that users can gain access to the data, checking for the data’s integrity (to make sure that the data stays intact and doesn’t get changed unnecessarily or duplicated), and to make sure that the database can be accessed by more than one person at a time. (Irron, S. 2009, Mahapatra, H. 2010)

6. Installation, Acceptance, Deployment

The new database is now put into operation. At this point training may be needed for users to learn how to access and modify the database. During operation new data is
added or deleted. Also more storage space may need to be created. (Irron, S. 2009, Mahapatra, H. 2010)

7. Maintenance

Over time changes may be need to keep the database functional for the company. Maintenance insures that the database will continue working for the company for years to come. This may include updating the software and hardware, backing up the data, monitoring usage, and performance tuning. (Irron, S. 2009, Mahapatra, H. 2010)

A discussion about the three steps in database design

There are three steps to database design which include conceptual, logical, and physical modeling. Conceptual modeling is the overall general view of what the database needs are. A general flow chart is created showing how the data may relate to each other. Next is logical modeling. Logical modeling is the mapping of the database. This determines the table design, how data is stored, access controls, and indexing. Finally is physical modeling. Physical modeling deals with the storage and access of the data. Physical modeling also includes the hardware that is needed for the database to run properly. (Irron, S. 2009, Mahapatra, H. 2010, McCaldin, D. 2010)
The entity relationship diagram

An entity relationship diagram

There are multiple entities and attributes within this diagram. Starting from the left side the first entity is Invoice Products with the attributes Invoice ID, Product ID, Quantity, and Discount. Invoice is another entity with the attributes Date of Invoice, Shipping Info ID, and Customer ID. There is an entity titled Product with the attributes Description and Price. Shipping Info is another entity its attributes include Shipping Method, Terms, Date Shipped, and Cost. Customer is yet another entity with the attributes First Name ID and Last Name ID. First Name and Last Name are also entities with the attribute First Name and Last Name. Another entity is Emails which contains Customer ID and Email Address. Phone numbers is also an entity with the attributes Customer ID and Phone Number. Finally there is the entity Address with the attributes Customer ID, Street Address, Street Name, City, State, and Zip Code. (Barton, T., & Christensen, B. 2004), (Hernandez, M. J. 2003)

These entities and attributes also have relationships with each other. The attribute Product ID in Invoice Products is related to the Products entity. Invoice ID also in Invoice Products has a relationship with the entity Invoice. Invoice has two more relationships one with Customer and one with Shipping Info. Customer has relationships with First Name, Last Name, Emails, Phone Numbers, and Address. (Hernandez, M. J. 2003)
The logical model and normalization

Logical data model

What is Normalization? Normalization is the process of organizing data to reduce redundancies and anomalies such as repetition or data, unwanted deletion, or other errors. Normalization has steps to follow to get the database normalized which is referred to as 1\textsuperscript{st}, 2\textsuperscript{nd}, 3\textsuperscript{rd}, 4\textsuperscript{th}, and 5\textsuperscript{th} normal form. Although there is a 4\textsuperscript{th} and 5\textsuperscript{th} normal form most databases only need 3\textsuperscript{rd} normal form. What are the steps to get to these normal forms?

First Normal form also called 1NF requires gathering all the data intended to go into the database. Once this data is gathered the data needs to be sorted through. The data should be sorted into groups of likeness these groups are referred to as topic tables. Also duplicated data should be removed. For instance if you had a list with pencil, pencil 2, pencil 3, pencil 4, pen, pen 1, pen 2, pen 3 these could be reduced to two topic tables one labeled pencil and one pen without the need to keep repeating or maybe it they could both be put under on topic table labeled writing supplies depending on what the function of the database being created is for.

Once all the data has been put into first normal form than the process of second normal form can begin. Second normal form takes the topic tables and determines what the primary key will be and what might be foreign keys. The main purpose of 2NF is to make sure that there are primary keys for each table so that the information stored in that table can be easily called upon. The primary key is used to access one table it is the identifier for that particular table. Foreign keys are keys within a table that are not the primary key for that table but are a primary key for another table. So if you are an office supply store and you’re organizing your products database you might have a writing supplies table that has an id to locate the table then within the table there might be pens, pencils, and paper which have foreign keys leading to separate tables. One
of these tables may be labeled paper which has a primary key for the table and lists all the
different brands of paper. The different brands within the paper table are also the attributes of the
table.

Next is third normal form. Before beginning 3NF everything must be in second normal form
first. The goal of 3NF is to not have anything that’s not dependent on the primary key apart of
that table. For example with the paper and the different brands let’s say each brand is supplied by
a different vendor. The vendors should be listed under a vendor table. Each vendor should have
its own primary key. Then for each brand of paper the vendor primary key can be listed for the
specific vendor that supplies that brand. This makes it so the vendors information does not need
to be repeated for each brand but only once in the vendor table.

Putting a database in third normal form can reduce errors from occurring. By assigning each
set of information its own primary key that can be called upon when needed in different areas of
the database this can save on redundancy and save space. The information would then only need
to be updated in the one area if something where to change. For example in the vendors table
mentioned above if a vendor would get a new phone number it would only need to be updated in
the vendors information table and it would be updated in every other area as well since every
other area calls upon the vendor information primary key. Also creating dates for the information
to be saved helps keep the old information from being deleted when new data is added.
Normalized to 3NF

The table meets 1NF because there are no duplicates. 2NF is established by determining primary keys. Finally 3NF is reached by getting rid of anything that doesn’t relate to the primary key. 3NF has also established the business rule for the table. Information that may be changed frequently is given its own primary key and table so that it is easier to change.

Primary Keys: Invoice Product ID, Invoice ID, Shipping Info ID, Shipping Method ID, Email ID, Shipping Cost ID, Customer ID, First Name ID, Last Name ID, Phone Number ID, Product ID, Address ID, City ID, State ID, Zip ID. Foreign Keys: Invoice Product ID, Invoice ID, Shipping Info ID, Shipping Method ID, Email ID, Shipping Cost ID, Customer ID, First Name ID, Last Name ID, Phone Number ID, Product ID, Address ID, City ID, State ID, Zip ID.
The Microsoft Access database

Your database created in Microsoft Access

Product table contains the product id, a description of the product and the price of the product. Invoice Product contains the invoice product id along with the shipping invoice id as well as product id and discount id, this table also contains quantity which determines the amount of product the customer would like to purchase. Discount contains discount id and discount percent for the percent that will be taken off of the products. Shipping invoice contains shipping info id, customer id and the date that the invoice is created for records. Shipping info contains a shipping info id, ship to id, shipping method id, shipping cost id, terms that defines specifics such as overnight delivery, and the date that the order is shipped. Ship to contains ship to id, ship to city id ship to name and ship to address. Ship to city has the ship to city id, ship to state id and ship to city name. Ship to zip code holds a ship to zip code id, ship to city id and the ship to zip code. Ship to state has ship to state id and the ship to state name. The tax rate table has tax rate id, ship to state id the tax rate percentage and the date of the tax rate since tax rates are changed frequently. Customer contains customer id first name and last name of the customer. Email has email id, customer id and email address. Phone number holds phone number id customer id and phone number. Address has address id, customer id, city id, and street address for the customers address. City id has state id and city name within it. The zip code table has zip code id, city id and the zip code. State table contains state id and state name.

The relationship between product table and invoice product has the one on the product end and many on the invoice products end which states that there can be many
product ids with in the invoice product. The invoice product table also has a many to one relationship with shipping invoice. The invoice product creates the many to many relationship needed to allow for the multiple products on one single invoice. The discount table also has a relationship with the invoice product table this relationship also allows for multiple discounts to be added to products and to a single invoice. The shipping invoice also has relationships with customer id and shipping info. Their relationships say that there can be many shipping invoices but only one customer and only one type of shipping info for each invoice. The customer table also has relations with email, phone number and address these relations are all many to one customer to allow the customers to enter multiple phone numbers, addresses, and emails for example their home phone and their cell phone. The address table has another relation with the city table where there may be only one city per address. The city table then has a relation with state and zip code. The relation between city and state says that there may be many cities per state but only one state per city. The relation between city and zip is the opposite of the relation between city and state. Each city may have multiple zip codes but there may only be one zip code per address. Besides the relation between shipping info and shipping invoice shipping info has three more relationships. These relations are with ship to, shipping method, and shipping cost. All these relations state that there can only be on ship to, shipping method and shipping cost per shipping info id. The ship to table has a relation to ship to city where there is only one city for a shipping address a relations with state where there is one state per city and a relation with zip code where there is only one zip code for the city for the shipping address. Off of the state that the product is being shipped to is the tax rate table where there is only one tax rate per state.
Query one the Invoice Product Query shows the many to many relations for product and shipping invoice. The first column shows the product ids where the second column shows they are all on the same invoice followed by a description to the product the shipping info id, customer id, date invoice was created quantity of products and the cost per product.

Query two Customer Query shows the customers id with their first name, last name, street address, city, state, zip, phone number and email.

Query three Shipping Invoice query shows the date the invoice was created the shipping info id the method which is Fed EX here terms are 3-day shipping the date the products where shipped and the shipping cost at 3% of the total weight of products.
**SQL SELECT statements**

```
FROM [Shipping Invoice] RIGHT JOIN (Product RIGHT JOIN [Invoice Product] ON Product.[ProductID] = [Invoice Product].[ProductID]) ON [Shipping Invoice].[ShippingInvoiceID] = [Invoice Product].[ShippingInvoiceID]
```

```
SELECT Customer.CustomerID, Customer.FirstName, Customer.LastName, Address.StreetAddress, City.[City Name], State.StateName, [Zip Code].ZipCode, [Phone Number].PhoneNumber, Email.EmailAddress
FROM ((Customer INNER JOIN [Phone Number] ON Customer.[CustomerID] = [Phone Number].[CustomerID]) INNER JOIN (((State INNER JOIN City ON State.[StateID] = City.[StateID]) INNER JOIN [Zip Code] ON City.[City ID] = [Zip Code].[CityID]) INNER JOIN Address ON City.[City ID] = Address.[CityID]) ON
```
Customer.[CustomerID] = Address.[CustomerID]) INNER JOIN Email ON
Customer.[CustomerID] = Email.[CustomerID];

SELECT [Shipping Invoice].DateOfInvoice, [Shipping Info].ShippingInfoID,
[Shipping Method].[Shipping Method], [Shipping Info].Terms, [Shipping
Info].DateShipped, [Shipping Cost].ShippingCost
FROM ([Shipping Method] RIGHT JOIN ([Shipping Cost] RIGHT JOIN [Shipping Info]
ON [Shipping Cost].[ShippingCostID] = [Shipping Info].[ShippingCostID]) ON
[Shipping Method].[Shipping Method ID] = [Shipping Info].[ShippingMethod ID])
RIGHT JOIN [Shipping Invoice] ON [Shipping Info].[ShippingInfoID] = [Shipping
Invoice].[ShippingInfoID];
The Microsoft Access database application

Complete your Microsoft Access application

Menu, forms and reports
(These are screen shots)
### Shipping Invoice

<table>
<thead>
<tr>
<th>ShippingInfoId</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>DateOfInvoice</td>
<td>2/2/2012</td>
</tr>
<tr>
<td>CustomerID</td>
<td>3</td>
</tr>
<tr>
<td>ShipTo</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>George Phillips</td>
</tr>
<tr>
<td>Address</td>
<td>123 North Lane</td>
</tr>
<tr>
<td>City</td>
<td>Lincoln</td>
</tr>
<tr>
<td>State</td>
<td>Nebraska</td>
</tr>
<tr>
<td>ZipCode</td>
<td>68434</td>
</tr>
</tbody>
</table>

| DateShipped   | 2/3/2012              |
| Shipping Method| Fed Ex                |
| Terms         | 3-day                 |

<table>
<thead>
<tr>
<th>ProductID</th>
<th>Description</th>
<th>Price</th>
<th>Quantity</th>
<th>Discount</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Brian in a jar</td>
<td>$10.00</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Spam Lip Balm</td>
<td>$3.00</td>
<td>3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Toast Mittens</td>
<td>$10.00</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>(New)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Customer

<table>
<thead>
<tr>
<th>CustomerID</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FirstName</td>
<td>Andy</td>
</tr>
<tr>
<td>LastName</td>
<td>Grey</td>
</tr>
<tr>
<td>StreetAddress</td>
<td>598 E 41st</td>
</tr>
<tr>
<td>CityName</td>
<td>Sioux Falls</td>
</tr>
<tr>
<td>StateName</td>
<td>South Dakota</td>
</tr>
<tr>
<td>ZipCode</td>
<td>57103</td>
</tr>
</tbody>
</table>

### Ship To Query

<table>
<thead>
<tr>
<th>ShipToName</th>
<th>George Phillips</th>
</tr>
</thead>
<tbody>
<tr>
<td>ShipToAddress</td>
<td>123 North Lane</td>
</tr>
<tr>
<td>City</td>
<td>Lincoln</td>
</tr>
<tr>
<td>ZipCode</td>
<td>68434</td>
</tr>
<tr>
<td>State</td>
<td>Nebraska</td>
</tr>
<tr>
<td>Terms</td>
<td>3-day</td>
</tr>
<tr>
<td>DateShipped</td>
<td>2/3/2012</td>
</tr>
<tr>
<td>CustomerID</td>
<td>3</td>
</tr>
<tr>
<td>DateOfInvoice</td>
<td>2/2/2012</td>
</tr>
<tr>
<td>Shipping Method</td>
<td>Fed Ex</td>
</tr>
<tr>
<td>CustomerID</td>
<td>3</td>
</tr>
<tr>
<td>ProductID</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------</td>
</tr>
<tr>
<td>6</td>
<td>Brian in a jar</td>
</tr>
<tr>
<td>5</td>
<td>Spam Lip Balm</td>
</tr>
<tr>
<td>1</td>
<td>Toast Mittens</td>
</tr>
</tbody>
</table>

Total: $23.00
References


