Computer Aided Building Modeling and Design (CT-314)

Database, its Structure and Content

Manufacturing Database

Creating Manufacturing Database

Data Base

A database is an organized collection of data in computer, stored and accessed electronically which can be accessed in various ways. Data is given in structured way, grouped into tables which is further divided into columns & rows.

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a database management system (DBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just “database”.

Examples: NADRA database, Students Result database.

- There are various kinds of database software like analytical database software (allows analysis), Data warehouse software (takes data from various sources and store in central location), Distributed database software (like Google cloud storage), End user database software (like MS Excel spreadsheets), External database software (where users can access e.g. from internet), Operational database software (allows user to edit them like for customer or finances like in banks)
- Software examples: MS Excel, MS Access, Oracle database.

Database Management System

- A database management system (DBMS) is a software package designed to define, manipulate, retrieve and manage data in a database. A DBMS generally manipulates the data itself, the data format, field names, record structure and file structure. It also defines rules to validate and manipulate this data.
- The earliest databases only handled individual single pieces of specially formatted data. Today's more evolved systems can handle different kinds of less formatted data as well.
- The earliest types of database management systems consisted mainly of hierarchy and network models.
- A database model shows the logical structure of a database, including the relationships and constraints that determine how data can be stored and accessed. Individual database models are designed based on the rules and concepts of whichever broader data model the designers adopt.
- The hierarchy model is one where each node or component has a child/parent relationship with one other node or component.
- In the network model, the difference is that a single component can have multiple relationships – think of this as a single node being able to "multicast" connections.

DBMS & Database Model

- However, over time, these models became overtaken by something called a **relational database**. In the relational database model, individual components have attributes that are linked to their identities through a database table design. The rows and columns of an individual database table include those identities and attributes in such a way that traditional Structured Query Language or SQL can be used to pull various kinds of information on these relational models.
- Other types of DBMS models include a graph database model. These offer further alternatives to traditional relational database design.
- Some of the newest types of DBMS can be used where a data center may have a wide disparity of differently formatted or relatively unformatted or “raw” data to work with, where records are not normalized in the conventional way. This and other types of advances have made the world of the DBMS more complex for which DB engineers and administrators are required.

DBMS & Database Model

- Other types of DBMS models include a graph database model. These offer further alternatives to traditional relational database design.
- Some of the newest types of DBMS can be used where a data center may have a wide disparity of differently formatted or relatively unformatted or “raw” data to work with, where records are not normalized in the conventional way. This and other types of advances have made the world of the DBMS more complex for which DB engineers and administrators are required.

Database Normalization

- **Normalization** of data is the process of organizing data in a database. This includes creating tables and establishing relationships between those tables according to rules designed both to protect the data and to make the database more flexible by eliminating redundancy and inconsistent dependency.
- **Why to do Normalization?**
- **Redundant** data wastes disk space and creates maintenance problems. If data that exists in more than one place must be changed, the data must be changed in exactly the same way in all locations. A customer address change is much easier to implement if that data is stored only in the Customers table and nowhere else in the database.
- I**nconsistent dependency** can be explained with the example that in a table of addresses of customer, it may not make sense to look for the salary of the employee who calls on that customer. The employee's salary is related to, or dependent on, the employee and thus should be moved to the Employees table. Inconsistent dependencies can make data difficult to access because the path to find the data may be missing or broken.

Difference between a Database and a Spreadsheet

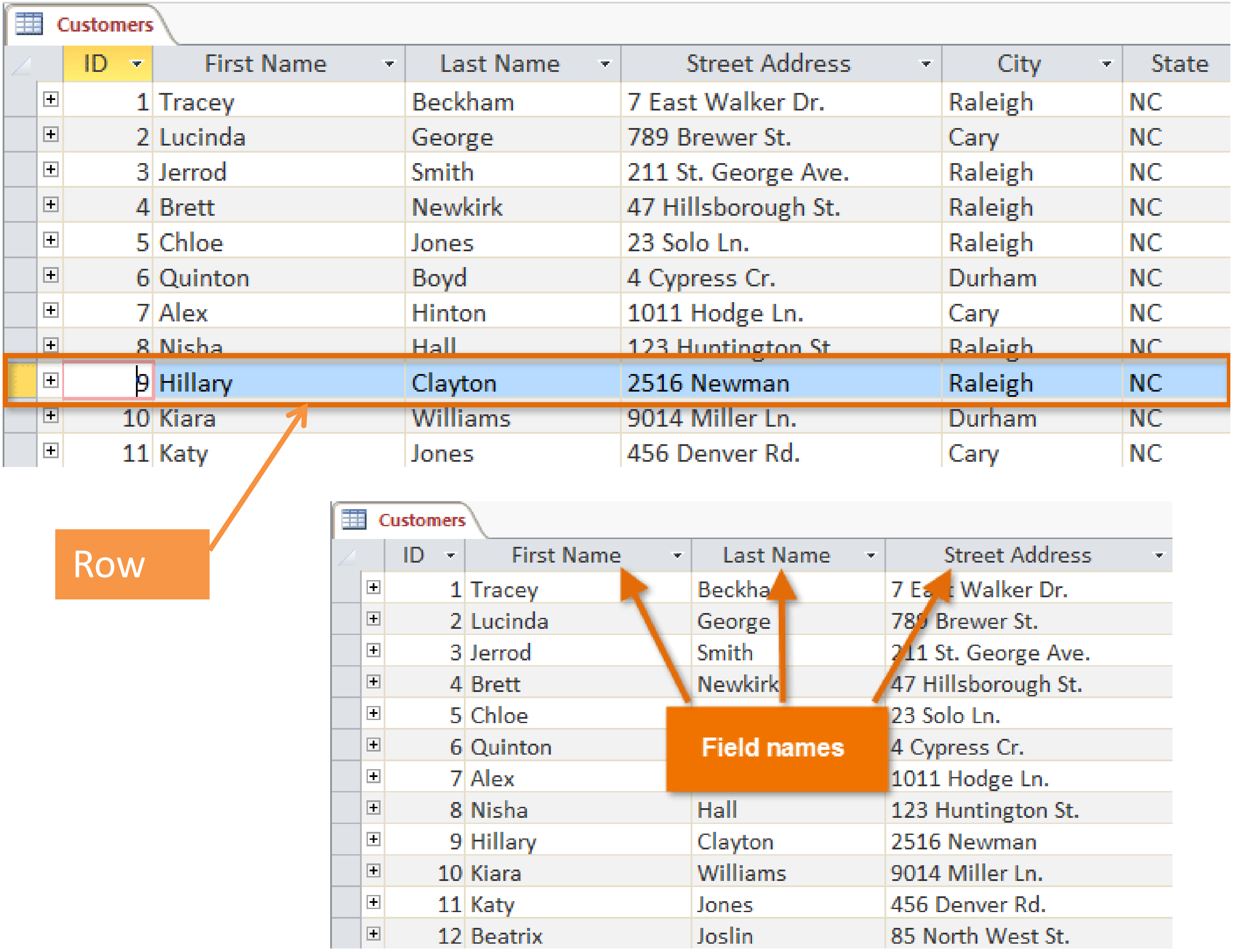
- Databases and spreadsheets (such as Microsoft Excel) are both convenient ways to store information. The primary differences between the two are:
  - How the data is stored and manipulated
  - Who can access the data
  - How much data can be stored
- Spreadsheets were originally designed for **one** user, and their characteristics show that. They are great for a single user or small number of users who don’t need to do a lot of complicated data manipulation.
- Databases, on the other hand, are designed to hold much larger collections of organized information and sometimes even in massive amounts. Databases allow **multiple** users at the same time to quickly and securely access and query the data using highly complex logic and language.

Structure of Database

- Within a database (relational model), related data are grouped into tables, each of which consists of rows (also called tuples) and columns, like a spreadsheet.
- Each row of a table is called a record. Records include data about something or someone, such as a particular customer. By contrast, columns (also known as fields or attributes) contain a single type of information that appears in each record, such as the addresses of all the customers listed in the table.
- To keep the data consistent from one record to the next, the appropriate data type is assigned to each column. Common data types include:
  - CHAR - a specific length of text
  - VARCHAR - text of variable lengths
  - TEXT - large amounts of text
  - INT - positive or negative whole number
  - FLOAT, DOUBLE - can also store floating point numbers
  - BLOB - binary data
- Some database management systems also offer the Autonumber data type, which automatically generates a unique number in each row.

Structure of Database

- A **field** is more than just a column; it's a way of organizing information by the **type** of data it is. Every piece of information within a field is of the same **type**. For example, every entry in a field called **First Name** would be a name, and every entry in field called **Street Address** would be an address.
- Likewise, a **record** is more than just a row; it's a unit of information. Every cell in a given row is part of that row's record



ID	First Name	Last Name	Street Address	City	State
1	Tracey	Beckham	7 East Walker Dr.	Raleigh	NC
2	Lucinda	George	789 Brewer St.	Cary	NC
3	Jerrod	Smith	211 St. George Ave.	Raleigh	NC
4	Brett	Newkirk	47 Hillsborough St.	Raleigh	NC
5	Chloe	Jones	23 Solo Ln.	Raleigh	NC
6	Quinton	Boyd	4 Cypress Cr.	Durham	NC
7	Alex	Hinton	1011 Hodge Ln.	Cary	NC
8	Nisha	Hall	123 Huntington St	Raleigh	NC
9	Hillary	Clayton	2516 Newman	Raleigh	NC
10	Kiara	Williams	9014 Miller Ln.	Durham	NC
11	Katy	Jones	456 Denver Rd.	Cary	NC

ID	First Name	Last Name	Street Address
1	Tracey	Beckham	7 East Walker Dr.
2	Lucinda	George	789 Brewer St.
3	Jerrod	Smith	211 St. George Ave.
4	Brett	Newkirk	47 Hillsborough St.
5	Chloe		23 Solo Ln.
6	Quinton		4 Cypress Cr.
7	Alex		1011 Hodge Ln.
8	Nisha	Hall	123 Huntington St.
9	Hillary	Clayton	2516 Newman
10	Kiara	Williams	9014 Miller Ln.
11	Katy	Jones	456 Denver Rd.
12	Beatrix	Joslin	85 North West St.

Structure of Database

- One or more attributes will serve as the primary key for each table. A **primary key** (PK) is a unique identifier for a given entity, meaning that you could pick out an exact customer even if you only knew that value.
- Attributes chosen as primary keys should be unique, unchanging, and always present (never NULL or empty). For this reason, order numbers and usernames make good primary keys, while telephone numbers or street addresses do not. You can also use multiple fields in conjunction as the primary key (this is known as **a composite key**).

Structure of Database

Relationships between Entities in a Database

- **One-to-one relationships (1:1)**

When there's only one instance of Entity A for every instance of Entity B, they are said to have a one-to-one relationship (often written 1:1).

- **One-to-many relationships** (1:M)

These relationships occur when a record in one table is associated with multiple entries in another. For example, a single customer might have placed many orders, or a patron may have multiple books checked out from the library at once.

- **Many-to-many relationships (M:N)**

When multiple entities from a table can be associated with multiple entities in another table, they are said to have a many-to-many (M:N) relationship. This might happen in the case of students and classes, since a student can take many classes and a class can have many students.

Unfortunately, it's not directly possible to implement this kind of relationship in a database. Instead, you have to break it up into two one-to-many relationships.

Content of Database

- Depending upon the purpose of use, the content of database will vary. Many databases consists of natural-language texts of documents; number-oriented databases primarily contain information such as statistics, tables, financial data, and scientific & technical data.
- Typical commercial database applications include airline reservations, production management functions, medical records in hospitals, and legal records of insurance companies. The largest databases are usually maintained by governmental agencies, business organizations, and universities. These databases may contain texts of such materials as abstracts, reports, legal statutes, wire services, newspapers and journals, encyclopaedias, and catalogs of various kinds. Reference databases contain bibliographies or indexes that serve as guides to the location of information in books, periodicals, and other published literature. Thousands of these publicly accessible databases now exist, covering topics ranging from law, medicine, and engineering to news and current events, games, classified advertisements, and instructional courses.

Content of Database in CAD

- Depending upon the purpose of use, the content of database will vary. Many databases consists of natural-language texts of documents; number-oriented databases primarily contain information such as statistics, tables, financial data, and scientific & technical data.
- Typical commercial database applications include airline reservations, production management functions, medical records in hospitals, and legal records of insurance companies. The largest databases are usually maintained by governmental agencies, business organizations, and universities. These databases may contain texts of such materials as abstracts, reports, legal statutes, wire services, newspapers and journals, encyclopaedias, and catalogs of various kinds. Reference databases contain bibliographies or indexes that serve as guides to the location of information in books, periodicals, and other published literature. Thousands of these publicly accessible databases now exist, covering topics ranging from law, medicine, and engineering to news and current events, games, classified advertisements, and instructional courses.

Manufacturing Database

- With the advent of the CAD and CAM software, there has been integration of designing and manufacturing processes. Just like computer aided designing (CAD) we have concept of computer aided manufacturing (CAM). Software enables direct link between CAD & CAM.
- On its part, CAD enables automation of designing, while CAM enables automation of manufacturing processes. The combination of CAD and CAM enables automated transition from designing to manufacturing. For the product that has been designed using the CAD software on computer, all the process planning and management of the manufacturing operations for the manufacture of the product can be done by the computer systems. All the data from the CAD systems can be directly used for the CAM systems.
- **The database created by the integration of CAD/CAM is also called as manufacturing database.** It includes all the data about the product generated during design like shape and dimensions, bill of materials and part lists, material specifications etc. It also includes additional data required for the manufacturing purposes.
- Thus in the integrated CAD/CAM system the two processes of designing and manufacturing are combined together. There is no time gap between the two processes and there is no duplication of efforts required on the parts of designer and the production personnel.

Database in Manufacturing Industry

- http://misclassblog.com/database-application/databases-in-manufacturing/

Creating a Database

Database Design Process

A well-structured database:

- Saves disk space by eliminating redundant data.
- Maintains data accuracy and integrity.
- Provides access to the data in useful ways.

Designing an efficient, useful database is a matter of following the proper process, including these phases:

- Requirements analysis, or identifying the purpose of your database:
- Organizing data into tables
- Specifying primary keys and analyzing relationships
- Normalizing to standardize the tables

These phases have been explained in the next slides.

Database Design Process

Requirements analysis, or identifying the purpose of your database:

- Understanding the purpose of your database will inform your choices throughout the design process. Make sure you consider the database from every perspective. For instance, if you were making a database for a public library, you'd want to consider the ways in which both patrons and librarians would need to access the data.
- Here are some ways to gather information before creating the database:
  - Interview the people who will use it
  - Analyze business forms, such as invoices, timesheets, surveys
  - Comb through any existing data systems (including physical and digital files)
  - Start by gathering any existing data that will be included in the database. Then list the types of data you want to store and the entities, or people, things, locations, and events, that those data describe, like this:
  - Customers

    Name, Address, City, State, Zip, Email address,
  - Products

    Name, Price, Quantity in stock, Quantity on order, Orders
- This information will later become part of the data dictionary, which outlines the tables and fields within the database. Be sure to break down the information into the smallest useful pieces. For instance, consider separating the street address from the country so that you can later filter individuals by their country of residence. Also, avoid placing the same data point in more than one table, which adds unnecessary complexity.
- Once you know what kinds of data the database will include, where that data comes from, and how it will be used, you're ready to start planning out the actual database.

Database Design Process

- **Organizing data into tables**
- The next step is to lay out a visual representation of your database
- Within a database (relational), related data are grouped into tables, each of which consists of rows (also called tuples) and columns, like a spreadsheet.
- **Specifying primary keys and analyzing relationships**

Primary keys are assigned and relationships (1:1, 1:M, M:N) established.

- **Normalizing to standardize the tables**

Once you have a preliminary design for your database, you can apply normalization rules to make sure the tables are structured correctly. Think of these rules as the industry standards.

Creating Manufacturing Database

- Keep the functionality small like create, report, update, and delete.
- Use an SQL-based database. This provides scalability and long-term maintainability of the application. Many manufacturing applications will have a long lifetime and only a standards-based database will provide long-term supportability.
- Use a database server. While small applications often can be run on one computer, locating the database on a database server provides a supportable solution. The application's database may be just one of many databases on the server, but the server will have backup support, and redundant power supplies.
- Use integrity constraints. Most SQL databases support some form of not-null, unique, and range checking functions. This ensures that **only valid data is added** to the database and provides checks that do not have to be included in code.
- Create tables without integrity constraints to hold "abnormal" situations. If an error occurs in normal operation, the operator often will not have enough information to solve the problem. Create "abnormal" tables that have the same structure as normal tables, but without integrity constraints, and store the abnormal cases in these tables. Use this information for analysis and correction of problems.