Winter 12-10-2002

The Role of the Relational Database Model in E-Business

Terrence D. Quinn

The Role of the Relational Database Model in E-Business

Terrence D. Quinn
E-Business Society
California Polytechnic & State University
3801 W. Temple Ave., Pomona. CA 91768
tdquinn@csupomona.edu

Abstract

The subject of databases will be a topic we, as e-business and IT professionals will encounter quite often in the field we have chosen. There are several database formats available that I will briefly mention, but I will quickly return focus back to the most widely used and in my opinion the best out there, relational database. I believe in order for an e-business to survive or any business for that matter, it must have complete control of its database and at all phases from creation to upgrade. Not only will this paper provide an overview of the various database management system available it will also provide a real case study of how an e-business uses its relational database management system to stay competitive within its specific industry.

1. Hierarchical Database Model

The first database system developed was the hierarchical database model. The hierarchical model used physical pointers to link records (see Appendix A.) As its name implies, this type of model only allowed hierarchical relationships. This means that each entity at a lower level can only be linked to one entity at a higher level, this type of relationship is known as a "parent-child" relationship (see Appendix B.) Hierarchical database provides very efficient high-speed retrieval (NAU.) Unfortunately hierarchical database models have more disadvantages than advantages. Some of these disadvantages include: (1) Hierarchical databases are too difficult to modify and maintain as an organization and its needs grow. (2) It is almost impossible to represent non- hierarchical relationships. In order to be able to complete such a task you would have to relate many hierarchical databases together resulting in the production of redundant data.

2. Network Database Model

Another database model that is used is called the network database model. The network database model was created to represent complex data relationships more effectively than the hierarchical model could, to improve database performance, and to impose a database standard. This model is very similar to the hierarchical model and has all of it attributes plus some of its own. The network model differs from that of the hierarchical by allowing a data element to have more than one parent. What makes network database models a little less desirable is the fact that they use an even more complex set of pointer structures that can be confusing for the end user to manipulate, and its lack of structural independence, meaning if any structural changes were made to the database all application must be revalidated before they can access the database. Due to these disadvantage the network database model was widely replaced by the relational database model.

3. Relational Database Model

As mentioned before, the relational database model is the most widely used and in my opinion (if that counts) the best database management system available. The relational model, first developed by E.F. Codd (of IBM) in 1970, represented a major breakthrough for both users and designers. (DS) A fine analogy that represents this transition well states "the relational model produced an "automatic transmission" database to replace the "manual transmission" database that preceded it." (DS) Unfortunately at its time of creation, the relational model was too rich; computers lacked the power to implement the model.

Presently, a majority of computers ranging from mainframe legacy systems to the fastest of microcomputers are able to handle relational database modeled software, software's such as Sybase, Oracle, DB2, Access, SQLServer and countless others, some of which we will review in detail later. So now we must answer the question, what is a relational database and more importantly how can it be used to improve an e-business?

A relational database model uses a relational database management system to perform the same basic functions that the previously mentioned hierarchical and network models performed and multitudes more. To the end user the relational database is made up of tables that store data. The beauty of a relational database is that each table, made up of rows and columns which are analogous to records and fields, and once these rows and columns (or records and fields) have been filled with the appropriate data; all operation are done on the tables themselves. Once this data is placed in a table, each table must have what is called a primary key. A primary key is simply an attribute that uniquely identifies any given row

As you may have noticed I mention two very important concepts above, the former being that the tables are filled with ***appropriate*** data and the later being ***operations*** performed on the tables themselves. What is

meant by appropriate data is data that has been normalized and what is meant by operations is Structured Query Language (SQL) used to extract information from the data tables. We will explore these two concepts in detail next.

Before any data can be placed in a table we must make sure that the tables are normalized, causing any data inserted in these tables to be normalized as well. By normalization you are insuring three things

1. All column values are atomic
2. All column values depend on the value of the primary key
3. No column value depends on any other column value except that of the primary key.

When you have applied the three rules, you say the database is in the Third Normal Form (3NF), or that it is "normalized". A normalized database generally improves performance, lowers storage requirements, and makes it easier to change the application to add new features. Remember, most software projects change it's requirements during its development, so the time spend normalizing a database will actually mean less development time. (EDM/2)

Structured Query Language or SQL for short is the relational models standard language and is composed of about 30 commands that allow you to create database and table structures, perform various types of data manipulation, and query the database to extract useful information.

The 30 commands previously mentioned fall into two basic categories: data definition language (DDL) and data manipulation language (DML).

The DDL provides the standards or SQL statement necessary to create the database schema and create the data tables within the schema (see Appendix C). The DML provides the standards or the SQL statement necessary to retrieve data or to query specific information from the database.

4. Applications of a Relational Database in e-Business

Now that we have a clear definition and understanding of a relational database, its components and how it works, one question remains; how can a RDMS help an e-business? To answer this question, as promised earlier I will use as my example one of the RDMS software's mentioned previously. In order to have a successful business any organization must at any given time be able to account for its employees, products and customers. Many companies accomplish this task by employing the help of a relational database management system. Jaafri.com, a company I currently have the pleasure of working with as part of my senior project has allowed myself and group of fellow classmate to update their website. Jaafri.com is a small firm that sells office supplies to larger firms through traditional means as well as the Internet. Our goal by the beginning of fall 2002 is to provide Jaafri.com with a fully functional relational database management system powered by Oracle 8i software. Currently Jaafri.com is using as it RDMS Microsoft Access. Although Access is a powerful RDMS, if Jaafri.com wants to compete with other firms in its market it must chose a database management system that is scalable. We chose Oracle 8i software because Oracle has it own additions to SQL, called PL/SQL. PL/SQL stand for Procedural Language/ Structured Query Language. With this addition we will be able to write in exception handling code that will respond with a specific message if a product is out of stock, no longer available for purchase or if a substitute product is recommended.

Although very brief I hope this real life case study cast a little light on how a RDMS can help an e-business stay competitive and successful in the new economy.

Bibliography

[1] Rosenzweig, Ben and Elena Silvestrova (2000). **Oracle, PL/SQL.** New Jersey: Prentice-Hall PTR

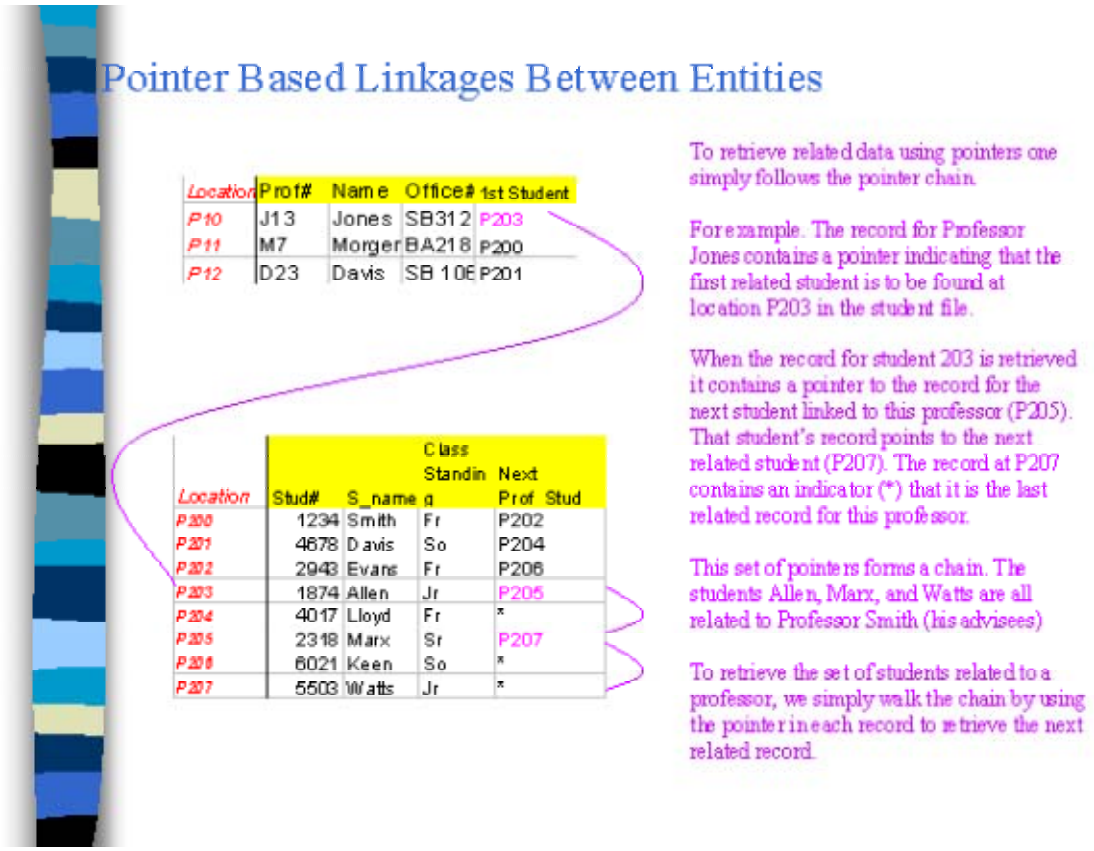
[2] Bernstein, Arthur, Michael Kifer and Philip M. Lewis (2002). **Database and Transaction Processing.** New York: Addison-Wesley

[3] Coronel, Carlos and Peter Rob (2000) **Database Systems: Design, Implementation & Management**. Massachusetts: Course Technology

[4] Lozano, F **Introduction to Relational Database.** http://www.edm2.com/0612/msql7.html Accessed July 14 2002

[5] Morgan, Jim **Alternative Database Types.** http://www.cba.nau.edu/morgan-j/class/subtop1_2/index.htm Accessed July 14 2002

APPENDIX A

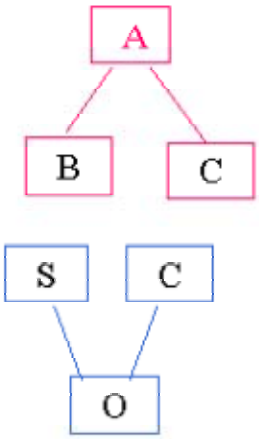


Pointer: A reference device that "points" to the location of specified data in the storage medium.

APPENDIX B

Hierarchical and Network Relationships

- A hierarchical relationship is one where each entity at a lower level of the hierarchy is related to only one type of entity at a higher level of the hierarchy (a higher level entity can be linked to two or more lower level "child" entities)
- A network relationship is one in which an entity at a lower level can be linked to two or more entities at a higher level.
  - E.G. - an order is related to both a customer who placed it and a salesperson who made the sale
- The hierarchical model does not support network relationships.
  - To capture them a second database would have to be created and linked to the first.



APPENDIX C

Creating and Manipulating the Database Structure

CREATE DATABASE <database name>; allows you to create the database

CREATE TABLE<table name>; allows you to create a table

CREATE TABLE<table name>; create table with attributes and primary key designation
<attribute1 name and attribute1characteristics
primary key designation, >

INSERT INTO <table name> allows you to insert data into tables
VALUES<attributes1 value, attributes2 value, etc… >;

SELECT* allows you to retrieve all data stored in a table
FROM <table name>;

SELECT <column(s)> allows you to retrieve a specific record of data stored in a table
FROM <table name>
WHERE<condition(s)>;