Featured Database Articles

SQL etc

Posted Aug 16, 1998

Introduction to Databases for the Web: Pt. 1

By Selena Sol

What is a database?

Once upon a time, in the primitive and barbarian days before computers, the amount of information shepherded by a group of people could be collected in the wisdom and the stories of its older members. In this world, storytellers, magicians, and grandparents were considered great and honored storehouses for all that was known.

Apparently, and according to vast archeological data, campfires were used (like command-line middleware) by the younger members of the community to access the information stored in the minds of the elders using API's such as

public String TellUsAboutTheTimeWhen(String s);.

And then of course, like a sweeping and rapidly-encompassing viral infection, came agriculture, over-production of foodstuffs, and the origins of modern-day commerce.

Dealing with vast storehouses of wheat, rice, and maize became quite a chore for the monarchs and emperors that developed along with the new economy. There was simply too much data to be managed in the minds of the elders (who by now were feeling the effects of hardware obsolescence as they were being pushed quietly into the background).

And so, in order to store all the new information, humanity invented the technology of writing. And though great scholars like Aristotle warned that the invention of the alphabet would lead to the subtle but total demise of the creativity and sensibility of humanity, data began to be stored in voluminous data repositories, called books.

As we know, eventually books propogated with great speed and soon, whole communities of books migrated to the first real "databases", libraries.

Unlike previous versions of data warehouses (people and books), that might be considered the australopithecines of the database lineage, libraries crossed over into the modern-

day species, though they were incredibly primitive of course.

Specifically, libraries introduced "standards" by which data could be stored and retrieved.

After all, without standards for accessing data, libraries would be like my closet, endless and engulfing swarms of chaos. Books, and the data within books, had to be quickly accessible by anyone if they were to be useful.

In fact, the usefulness of a library, or any base of data, is proportional to its data storage and retrieval efficiency. This one corollary would drive the evolution of databases over the next 2000 years to its current state.

Thus, early librarians defined standardized filing and retrieval protocols. Perhaps, if you have ever made it off the web, you will have seen an old library with its cute little indexing system (card catalog) and pointers (Dewey decimal system).

And for the next couple thousand years libraries grew, and grew, and grew along with associated storage/retrieval technologies such as the filing cabinet, colored tabs, and three ring binders.

All this until one day about half a century ago, some really bright folks including Alan Turing, working for the British government were asked to invent an advanced tool for breaking German cryptographic "Enigma" codes.

> Readers: In response to the above sentence, a concerned reader wrote in with the following comments, which I have verified online as true. I have left the original text in tact, but add his comments...
>
> > *"The historical part of your story isn't correct, I'm afraid. In your article, you mention how the first computers were invented by the English to break the German enigma code. It seems that you are not aware of the fact that the Nazi's used IBM computers to manage the Holocaust in the most orderly fashion. All data about the Holocaust victims were carefully stored in their American databases, using punch-cards! (Indeed, American IBM engineers have travelled to Germany all throughout WWII to manage the Nazi ICT system).*
> >
> > *More information on this chapter of history can be found in "Wallstreet and the Rise of Hitler" by professor Anthony Sutton.*
> >
> > *I thought this information should not be neglected for future generations.*

That day the world changed again. That day the computer was born.

The computer was an intensely revolutionary technology of course, but as with any technology, people took it and applied it to old problems instead of using it to its revolutionary potential.

Almost instantly, the computer was applied to the age-old problem of information storage and retrieval. After all, by World War Two, information was already accumulating at rates beyond the space available in publicly supported libraries. And besides, it seemed somehow cheap and tawdry to store the entire archives of "The Three Stooges" in the Library of Congress. Information was seeping out of every crack and pore of modern day society.

Thus, the first attempts at information storage and retrieval followed traditional lines and metaphors. The first systems were based on discrete files in a virtual library. In this file-oriented system, a bunch of files would be stored on a computer and could be accessed by a computer operator. Files of archived data were called "tables" because they looked like tables used in traditional file keeping. Rows in the table were called "records" and columns were called "fields".

Consider the following example:

First Name	Last Name	Email	Phone

Name			
Eric	Tachibana	erict@eff.org	213-456-0987
Selena	Sol	selena@eff.org	987-765-4321
Li Hsien	Lim	hsien@somedomain.com	65-777-9876
Jordan	Ramacciato	nadroj@otherdomain.com	222-3456-123

The "flat file" system was a start. However, it was seriously inefficient.

Essentially, in order to find a record, someone would have to read through the entire file and hope it was not the last record. With a hundred thousands records, you can imagine the dilemma.

What was needed, computer scientists thought (using existing metaphors again) was a card catalog, a means to achieve random access processing, that is the ability to efficiently access a single record without searching the entire file to find it.

The result was the indexed file-oriented system in which a single index file stored "key" words and pointers to records that were stored elsewhere. This made retrieval much more efficient. It worked just like a card catalog in a library. To find data, one needed only search for keys rather than reading entire records.

However, even with the benefits of indexing, the file-oriented system still suffered from problems including:

- Data Redundancy - the same data might be stored in different places
- Poor Data Control - redundant data might be slightly different such as in the case when Ms. Jones changes her name to Mrs. Johnson and the change is only reflected in some of the files containing her data
- Inability to Easily Manipulate Data - it was a tedious and error prone activity to modify files by hand
- Cryptic Work Flows - accessing the data could take excessive programming effort and was too difficult for real-users (as opposed to programmers).

Consider how troublesome the following data file would be to maintain.

Name	**Address**	**Course**	**Grade**
Mr. Eric Tachibana	123 Kensigton	Chemistry 102	C+
Mr. Eric Tachibana	123 Kensigton	Chinese 3	A
Mr. Eric Tachibana	122 Kensigton	Data Structures	B
Mr. Eric Tachibana	123 Kensigton	English 101	A
Ms. Tonya Lippert	88 West 1st St.	Psychology 101	A
Mrs. Tonya Ducovney	100 Capitol Ln.	Psychology 102	A
Ms. Tonya Lippert	88 West 1st St.	Human Cultures	A
Ms. Tonya Lippert	88 West 1st St.	European Governments	A

What was needed was a truly unique way to deal with the age-old problem, a way that reflected the medium of the computer rather than the tools and metaphors it was replacing.

Enter the database.

Simply put, a database is a computerized record keeping system. More completely, it is a system involving data, the hardware that physically stores that data, the software that utilizes the hardware's file system in order to 1) store the data and 2) provide a standardized method for retrieving or changing the data, and finally, the users who turn the data into information.

Databases, another creature of the 60s, were created to solve the problems with file-oriented systems in that they were compact, fast, easy to use, current, accurate, allowed the easy sharing of data between multiple users, and were secure.

A database might be as complex and demanding as an account tracking system used by a bank to manage the constantly changing accounts of thousands of bank customers, or it could be as simple as a collection of electronic business cards on your laptop.

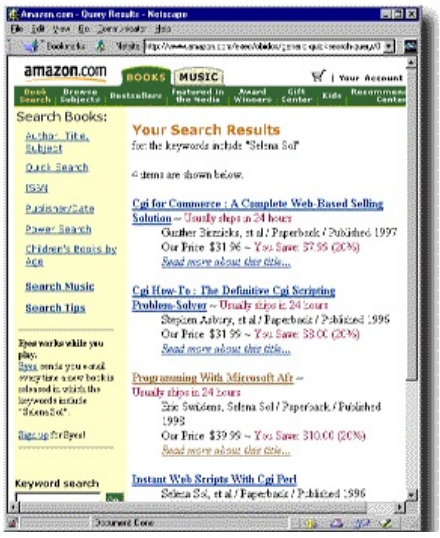
The important thing is that a database allows you to store data and get it or modify it when you need to easily and efficiently regardless of the amount of data being manipulated. What the data is and how demanding you will be when retrieving and modifying that data is simply a matter of scale.

Traditionally, databases ran on large, powerful mainframes for business applications. You will probably have heard of such packages as Oracle 8 or Sybase SQL Server for example.

However with the advent of small, powerful personal computers, databases have become more readily usable by the average computer user. Microsoft's Access is a popular PC-based engine.

More importantly for our focus, databases have quickly become integral to the design, development, and services offered by web sites.

Consider a site like Amazon.com that must be able to allow users to quickly jump through a vast virtual warehouse of books and compact disks.



How could Amazon.com create web pages for every single item in their inventory and how could they keep all those pages up to date? Well the answer is that their web pages are created on-the-fly by a program that "queries" a database of inventory items and produces an HTML page based on the results of that query.

The goal of this tutorial is to give you a rough and ready introduction to databases and give you the tools you need to get to work using the database tools available to you.

We will begin by focussing on some of the more theoretical aspects of databases so that you will have a good feel for the generic subject before we start in on all the specifics.

Go to page: 1 2 3 4 5 6 7 8 9 10 11 Next

SQL etc Archives

0 Comments (click to add your comment)

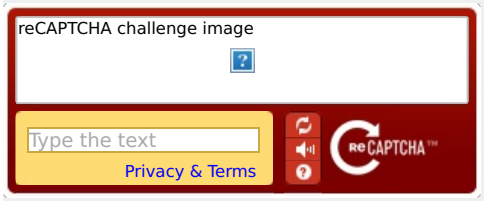
Comment and Contribute

Your name/nickname

Your email

Subject

(Maximum characters: 1200). You have 1200 characters left.



Submit Your Comment

SQL etc Forum

Topic	By	Replies	Updated
MySQL rollback	UAL225	0	August 21st, 09:56 PM
Complex Search Query	Galway	0	May 20th, 10:04 PM
change collation at once	supercain	2	May 15th, 06:18 AM
SQL Features, tools and utilities question	Neomite	1	April 10th, 09:13 AM