

## A Short Database History

**Ancient to modern:** The origins go back to libraries, governmental, business, and medical records. There is a very long history of information storage, indexing, and retrieval. Don't ignore this history, there is usually something to learn from these folks and their success and failure. Lots of online stuff (and there is lots) does not guarantee quality of data or search technique. Good design principles goes way back and lots is known now about how to make good designs that lead to better reliability and performance.

**1960's:** Computers become cost effective for private companies along with increasing storage capability of computers. Two main data models were developed: network model (CODASYL) and hierarchical (IMS). Access to database is through low-level pointer operations linking records. Storage details depended on the type of data to be stored. Thus adding an extra field to your database requires rewriting the underlying access/modification scheme. Emphasis was on records to be processed, not overall structure of the system. A user would need to know the physical structure of the database in order to query for information. One major commercial success was SABRE system from IBM and American Airlines.

**1970-72:** E.F. Codd proposed relational model for databases in a landmark paper on how to think about databases. He disconnects the schema (logical organization) of a database from the physical storage methods. This system has been standard ever since.

**1970's:** Several camps of proponents argue about merits of these competing systems while the theory of databases leads to mainstream research projects. Two main prototypes for relational systems were developed during 1974-77. These provide nice example of how theory leads to best practice.

Ingres: Developed at UCB. This ultimately led to Ingres Corp., Sybase, MS SQL Server, Britton-Lee, Wang's PACE. This system used QUEL as query language.

System R: Developed at IBM San Jose and led to IBM's SQL/DS & DB2, Oracle, HP's Allbase, Tandem's Non-Stop SQL. This system used SEQUEL as query language.

The term Relational Database Management System (RDBMS) is coined during this period.

**1976:** P. Chen proposed the Entity-Relationship (ER) model for database design giving yet another important insight into conceptual data models. Such higher level modeling allows the designer to concentrate on the use of data instead of logical table structure.

**Early 1980's:** Commercialization of relational systems begins as a boom in computer purchasing fuels DB market for business.

**Mid-1980's:** SQL (Structured Query Language) becomes "intergalactic standard". DB2 becomes IBM's flagship product. Network and hierarchical models fade into the background, with essentially no development of these systems today but some legacy systems are still in use. Development of the IBM PC gives rise to many DB companies and products such as RIM, RBASE 5000, PARADOX, OS/2 Database Manager, Dbase III, IV (later Foxbase, even later Visual FoxPro), Watcom SQL.

**Early 1990's:** An industry shakeout begins with fewer surviving companies offering increasingly complex products at higher prices. Much development during this period centers on client tools for application development such as PowerBuilder (Sybase), Oracle Developer, VB (Microsoft), etc. Client-server model for computing becomes the norm for future business decisions. Development of personal productivity tools such as Excel/Access (MS) and ODBC. This also marks the beginning of Object Database Management Systems (ODBMS) prototypes.

## A Short Database History

**Mid-1990's:** Kaboom! The usable Internet/WWW appears. A mad scramble ensues to allow remote access to computer systems with legacy data. Client-server frenzy reaches the desktop of average users with little patience for complexity while Web/DB grows exponentially.

**Late-1990's:** The large investment in Internet companies fuels tools market boom for Web/Internet/DB connectors. Active Server Pages, Front Page, Java Servlets, JDBC, Enterprise Java Beans, ColdFusion, Dream Weaver, Oracle Developer 2000, etc are examples of such offerings. Open source solution come online with widespread use of gcc, cgi, Apache, MySQL, etc. Online Transaction processing (OLTP) and online analytic processing (OLAP) comes of age with many merchants using point-of-sale (POS) technology on a daily basis.

**Early 21st century:** Decline of the Internet industry as a whole but solid growth of DB applications continues. More interactive applications appear with use of PDAs, POS transactions, consolidation of vendors, etc. Three main (western) companies predominate in the large DB market: IBM (buys Informix), Microsoft, and Oracle.

**Future trends:** Huge (terabyte) systems are appearing and will require novel means of handling and analyzing data. Large science databases such as genome project, geological, national security, and space exploration data. Clickstream analysis is happening now. Data mining, data warehousing, data marts are a commonly used technique today. More of this in the future without a doubt. Smart/personalized shopping using purchase history, time of day, etc.

Successors to SQL (and perhaps RDBMS) will be emerging in the future. Most attempts to standardize SQL successors has not been successful. SQL92, SQL2, SQL3 are still underpowered and more extensions are hard to agree upon. Most likely this will be overtaken by XML and other emerging techniques. XML with Java for databases is the current poster child of the "next great thing". Check in tomorrow to see what else is news.

Mobile database use is a product now coming to market in various ways. Distributed transaction processing is becoming the norm for business planning in many arenas.

Probably there will be a continuing shakeout in the RDBMS market. Linux with Apache supporting MySQL (or even Oracle) on relatively cheap hardware is a major threat to high cost legacy systems of Oracle and DB2 so these have begun pre-emptive projects to hold onto their customers.

Object Oriented Everything, including databases, seems to be always on the verge to sweeping everything before it. Object Database Management Group (ODMG) standards are proposed and accepted and maybe something comes from that.

Ethical/security/use issues tend to be diminished at times but always come back. Should you be able to consult a database of the medical records/genetic makeup of a prospective employee? Should you be able to screen a prospective partner/lover for genetic diseases? Should amazon.com keep track of your book purchasing? Should there be a national database of convicted sex offenders/violent criminals/drug traffickers? Who is allowed to do Web tracking? How many times in the last six months did you visit a particular sex chat room/porn site/political satire site? Who should be able to keep or view such data? Who makes these decisions?