

## SPECIAL SECTION: INNOVATION IN ACTION

 Printable version  Send to a colleague

## In the beginning: an RDBMS history

**Follow the last quarter-century of the relational database revolution.**

by Colin White

When *Teradata Magazine* invited me to write an article about the evolution of relational database technology, I readily accepted. After all, it gave me the opportunity to reminisce about a technology that has dominated most of my working life.



[Click to enlarge](#)

It also enabled me to catch up with pioneers such as Stephen Brobst, Teradata chief technology officer; Chris Date, noted author and researcher; Jim Gray, distinguished engineer at Microsoft; and Mike Stonebraker, adjunct professor of computer science at MIT, all of whom were involved in the early stages of this technology.

I wanted to get their perspectives on the events of the past 25 years and discuss where they see the database industry going in the future. They

had so many interesting anecdotes and details of the relational database evolution that I can't possibly share them all here; however, I can give you a sense of where we've been-and where we hope to be in the years to come.

### Survival of the fittest

Before we can look at the future, we have to go back to August 1969 to discover the origins of the relational model. This was when Dr. E. F. Codd published his paper, "Derivability, Redundancy, and Consistency of Relations Stored in Large Data Banks" in an IBM Research Report.

Codd's paper had a restricted audience, but a revised version of the paper was published the following year in *Communications of the ACM*. This paper, "A Relational Model of Data for Large Shared Data Banks," received a much wider distribution and is often incorrectly credited with being the original paper on the relational model. (For a detailed review of the two papers, check out Chris Date's excellent 1998 series of articles about this topic in *Intelligent Enterprise Magazine*, entitled "The Birth of the Relational Model.")

Codd's papers led to a variety of research projects including the System R project at IBM Research in San Jose, Calif., and the INGRES relational prototype led by Michael Stonebraker at the University of California at Berkeley. But as exciting as these projects were, it would be another 10 years before the first commercial relational products appeared, and relational technology itself had to fight several battles both outside and inside IBM to gain acceptance.

Outside of IBM, the relational model came under attack from advocates of alternative solutions. Perhaps the best known was the network database proposal by the CODASYL Data Base Task Group (DBTG). Charlie Bachman championed this effort to create an industry standard database model.

During this period, there were heated interchanges between Bachman and Codd about the merits of the

### Special Section table of contents

#### Retrospective: 25 years of IT history

In 1969, E. F. Codd came up with the idea of a relational database, but it was 10 years before the first commercial RDBMS got off the ground. Here is the story of how a theory launched an industry-and changed everything.

#### Case study: Union Pacific

North America's largest railroad company did more than just streamline data. It set standards for the competition.

#### Case study: PING

From data warehouse "anomaly" to pacesetter in just 15 years, PING is clearly at the top of its game.

#### The future: eXtreme data warehousing

Where will the future lead? Skyrocketing demands for data will create bigger, faster, better data warehouses.

network vs. relational approaches to database development. Relational technology also faced stiff opposition within IBM, which was concerned about the impact of the unproven technology on the revenue base from its proven IMS transaction and database system.

Codd, aided by Date, fought hard to gain credibility for the relational model. After all, one of Codd's objectives was "to simplify the potentially formidable job of the DBA," and that's precisely what the data independence of the relational model achieved, making database technology easier to use and program. As we now know, their persistence paid off.

While System R was being developed, IBM also was busily crafting a new generation of hardware and software commonly known as FS, or Future Systems. The FS software included a successor to IMS. This new product provided a network database based on the CODASYL DBTG model, a relational database and an interface to allow existing IMS database applications to run on the new system.

The FS project failed because it was too big and too complicated, making it the most expensive development failure in the company's history, according to Emerson Pugh in the 1995 book *Building IBM*. This left IBM with the problem of what to do about its database technology. A fascinating and in-depth discussion among the various people involved in the events of this period can be found in the paper "The 1995 SQL Reunion: People, Projects, and Politics."

Despite the setback, IBM continued to develop its first commercial relational database management system (RDBMS), which was released in 1981. This product, SQL/DS for VSE, was based on the System R research effort and carefully positioned as a decision-support system in order to protect IMS. In 1983, IBM announced DB2 for MVS, which stands as the last surviving component of the costly FS fiasco. Again, DB2 was marketed initially for decision support to protect IMS.

While IBM was navigating the political and practical obstacles of RDBMS technology, the rest of the industry was busy developing commercial products. Relational Software Inc. (now Oracle Corporation) announced its Oracle RDBMS in 1979, beating IBM to market by nearly two years. Many other key RDBMS products were released during the early 1980s, including a commercial version of Stonebraker's INGRES and NonStop SQL, which was developed in part by Jim Gray (see the timeline for more industry milestones).

As commercial products began to appear, there was considerable debate about their performance and scalability. One approach to improving performance was the database machine, which used tightly integrated hardware and software to boost database performance. Database machines employed their own operating systems and were optimized for the parallel processing of database requests against large databases. Applications requiring database services ran on a host computer (typically running DEC VAX/VMS or IBM MVS) and accessed the database machine across a channel or network connection.

Two of the best-known database machine companies were Britton-Lee (later renamed ShareBase Corporation) and Teradata, both of whom focused on decision support, not transaction processing as other companies were. Of the two, Teradata was the more successful. Scott Humphrey, founder of Humphrey Strategic Communications and former public relations manager for Britton-Lee, believes Teradata's success was due to its being first to market, in 1984, with a database machine that provided IBM mainframe connectivity. Teradata acquired ShareBase in 1985. (For more Teradata history, see sidebar.)

## Milestones in RDBMS development

### 1969

Dr. E. F. Codd publishes his first paper on the relational model

### 1970

UC Berkeley INGRES prototype work begins

### 1974

IBM SEQUEL language and prototype developed

IBM System R  
Prototype work begins

### 1977

Relational Software Inc. (RSI) founded

Revised SEQUEL/2 (subsequently renamed SQL) defined

### 1979

**Teradata Corporation formed**

Britton-Lee, Inc. (later renamed ShareBase) formed

Oracle released by RSI (now Oracle Corporation)

### 1981

SQL/DS for VSE announced by IBM

INGRES for VAX/VMS announced by Oracle Corporation

Having successfully proven they could provide good performance for both decision-support and business-transaction processing, RDBMSs steadily gained market share during the late 1980s and early 1990s.

### Lingua data

The initial releases of Oracle and IBM SQL/DS guaranteed the adoption of SQL as the de facto industry language to use for defining and manipulating relational databases. (A common misconception is that Codd created SQL as part of his early RDBMS research, but SQL was developed as a part of the System R research project at IBM.)

The American National Standards Institute (ANSI) published the first official SQL standard in 1986. The story goes that, in order to progress as rapidly as possible, the SQL committee based the standard on the System R SQL documentation.

In "The 1995 SQL Reunion," Don Chamberlin, a programmer who helped develop SQL, comments, "They kept all the warts, too. They didn't try to clean any of it up." To this day, Stonebraker is quite outspoken about IBM's role in the development of SQL. As he explains, "IBM knew SQL was not well-designed, and IBM could have fixed it. Instead, they chose not to."

There is no question, however, that the move toward SQL standardization was an important step in the RDBMS evolution. As Gray points out, "The relational model has gone from concept to bedrock. The consensus on the SQL syntax and the standardization was pivotal to this."

I don't think anyone would disagree that SQL has become complex. The latest version of the ANSI SQL standard (SQL: 2003) exceeds a thousand pages. As Gray notes, "Evolution typically brings complexity—compare the blue-green algae to a tree. But it also brings much greater diversity and functionality." However, he adds, "I think something will come along that will replace SQL. After all, FORTRAN was not the last programming language."

"SQL is no longer a language for real users, if it ever was," says Date. "It has become a developer's language." Stonebraker notes, "We need a simpler interface to today's RDBMS products. Graphical interfaces and self-tuning databases may offer some solutions." Brobst agrees, saying, "This increase in complexity puts the onus on vendors to produce user interfaces that generate SQL on behalf of the developer or user."

### Object lesson

During the 1990s, relational technology once again came under attack, this time from the object-oriented database camp. For several reasons, however, object-oriented database systems were not successful. One of the key reasons was poor performance for generalized commercial database processing. Although relational databases eventually won the day, the debate did lead to object-like capabilities being added to relational products and the SQL standard.

Not everyone approves of SQL's move toward object orientation. "If you think back to the mid- to late-1990s, there were a lot of people around who were saying that object databases would replace relational ones," says Date. "That battle has gone away. The extent that people have taken the same object ideas and put them into SQL, I think, is a terrible mistake."

Gray has a different perspective. "One problem that CODASYL created and that the relational database folks perpetuated is that they separated data from algorithms (programs). The object-oriented community has been lamenting this division almost from the start. The good news is that the unification of databases and programs is

**1983**  
DB2 for MVS  
announced by IBM

**1984**  
First DBC/1012  
database machine  
shipped by Teradata

**1985**  
Teradata acquired  
Britton-Lee

**1986**  
First version of SQL  
standard released

Sybase Inc. formed

**1987**  
NonStop SQL  
announced by Tandem

**1988**  
Microsoft, Sybase and  
Ashton-Tate develop  
Sybase for OS/2

**1989**  
Teradata partners with  
NCR Corporation

**1992**  
AT&T purchases NCR  
and Teradata

**1993**  
Microsoft and Sybase  
end partnership

Microsoft rebrands  
Sybase as SQL Server  
and releases Windows  
version

**1995**  
Computer Associates  
acquires INGRES as a  
part of its Ask Group  
purchase

**1996**  
Teradata Database  
' . . . . '



now happening, and that unification should be complete by the end of the decade."

### The XML factor

As we entered the new century, RDBMSs continued to add functionality and to increase their market penetration. RDBMS vendors worked on adding analytical and data-mining functions to the database engine, improving performance (a never-ending task), providing easier and more automated administration, creating support for complex data (spatial, multimedia, etc.), adding integration with messaging software and providing support for Linux.

A recent move by RDBMS vendors is the addition of XML support to relational products. This involves supporting XML data, adding XML extensions to SQL and providing XML query (XQuery) capabilities. Some XML advocates even believe XQuery will replace SQL.

So, what do the experts think about the impact of XML and XQuery on relational DBMSs? "There is a lot of hype around XML," says Stonebraker. "I think it is great for handling messages across the network, but it's not well suited for handling the storage and manipulation of data."

Brobst has a slightly different opinion. "As RDBMS products have evolved, vendors have added support for a variety of complex data types, including XML-based data. This is an important step forward, because it enables the processing of the data to be moved into the database engine for better performance."

Both Date and Gray express reservations. "I don't think XQuery is a particularly good language. It's not a query language; it's a programming language," argues Date. "XQuery is controversial," agrees Gray. "And I suspect that something cleaner and simpler will evolve and eventually displace it."

### Time for real-time?

The rate of change in the IT industry continues unabated. I asked our experts to predict where the database industry is headed:

**Stephen Brobst:** "I think that the ability to acquire data in real-time and to do event-based decision-making will become mainstream. The storing of unstructured data in relational DBMS products will grow dramatically, given the low cost of disk space. RDBMS self-management will become even more important. Lastly, grid computing for supporting the virtualization of CPU, I/O and storage will be a key direction."

**Chris Date:** "One of my research areas is what I call the Third Manifesto. Hugh Darwen and I have been working on this for well over 10 years. The original version was a paper of about 10 pages, but it grew into a book (*Foundation for Object/Relational Databases: The Third Manifesto*). We are now working on the third edition of that book. What we are trying to do is to get people to implement the relational model right. Another area I am researching is temporal data. This work also grew into a book (*Temporal Data and the Relational Model*). Support for temporal data and queries is badly needed in products."

**Jim Gray:** "Two really significant and related things are happening. First, database systems are recognizing that they must store objects. This has huge implications for how we structure and use databases. The second major thrust is that organizations are acquiring petabytes of information (documents, mail, Web sites, photos, music, videos and more). They can capture the information and they can store the information, but they cannot find it in their archives. So, there is a move from file systems to multimedia database systems that index this information."

made available for  
UNIX

### 1997

NCR becomes independent company

### 1998

In-database OLAP and data mining appear in RDBMSs

### 1999

RDBMSs prepare to support Y2K

### 2000

RDBMSs continue to add object-oriented capabilities and support for complex data

### 2001

Native XML support is provided for the first time in an RDBMS

### 2003

W3C enhances XQuery, the XML query language

### 2004

SQL:2003 standard is published

### A dedication

Dr. E. F. Codd, who passed away on April 18, 2003, invented the relational database model. Many of us, including myself, owe our livelihood to his original work. I would, therefore, like to dedicate this article to his memory. As Stephen Brobst has said, "Dr. Codd's work provided an elegant .. .. ."

theory that created a strong foundation for commercial products."

**Michael Stonebraker:** "I think a new type of processing is beginning to appear. This is the real-time processing of stream data. Relational (processing) is great for static data, but I am looking at engines that can process bits on the wire in real-time. Another important growth area is sensor networks (including RFID technology). In the future, all significant items will be tagged electronically. Companies will want to track items in real-time and correlate the results with historical information and business plans. This will require real-time business intelligence. I also think grid computing will be big."

### **Extreme can be good**

The innovators of the past 25 years have made it possible for companies like Teradata to develop a robust data warehousing technology. But it's also true that the companies that use the technology contribute to the evolution.

Many businesses, including Union Pacific and PING, embraced data warehousing early and created innovative solutions to ordinary business problems. And as technology has progressed, so have the ways in which companies implement it.

Who knows what the future will hold? "Extreme" data warehousing may be the next new thing.

This issue's special section takes a walk through time. See how Union Pacific and PING have grown up with Teradata. Then enjoy a glimpse of the future as Teradata's Chief Technology Officer Stephen Brobst and noted consultant Richard Hackathorn explore a future of extremes. **T**

## **Teradata: 25 years of industry firsts**

For a quarter of a century, Teradata has been at the forefront of the database universe, continually redefining and extending the possibilities of this rapidly evolving field.

Launched in 1979 from a garage in Marina del Rey, Calif., by Phil Neches, Jerry Modes, Dave Hartke, Ira Moskatel and Jack Shemer, the company started with a goal of building the first massively parallel database system that could store and reuse data in different ways. At the time, the idea of using parallel processing technology to speed up decision-support applications was considered revolutionary.

By 1984, Teradata had developed its first product: the Teradata Database Computer (DBC/1012), a relational database management system (RDBMS) on a proprietary platform. (The number 1012 was no whim; 1012 represents one trillion bytes, otherwise known as a terabyte, hence the name of the database.) That system first shipped to customers Wells Fargo and AT&T.

"One of my first jobs at Amdahl was to evaluate the DBC/1012, because Amdahl was considering re-marketing the machine," says Colin White, president of BI Research. "After working on proprietary mainframe systems for many years, it was exciting to talk to Teradata co-founder Phil Neches about large parallel computers built using off-the-shelf Intel microprocessors (an uncommon practice at that time) that could handle huge amounts of data."

All of this early development paid off when *Fortune* magazine named Teradata's DBC/1012 Database Computer System its 1986 "Product of the Year."

### **Joining forces**

As business and technology continued to evolve, Teradata formed a partnership with NCR in 1989, setting a goal of building the next-generation database computer. In 1991, AT&T acquired NCR, and NCR purchased Teradata later that year.

By 1993, Teradata had become the first company to market and install a database that was capable of running or storing three terabytes of data. The following year, Gartner Group named Teradata the "Leader in Commercial Parallel Processing." And in 1995, IDC named Teradata number one in massively parallel processing (MPP) in *Computerworld* magazine.

Support for open systems became a key issue during this period, and Teradata software was made available for UNIX in 1996: the Teradata RDBMS version 2 on UNIX SVR4. A Teradata Database became the world's largest database, with 11 terabytes of data, that same year. Also in 1996, the Data Warehouse Institute presented Teradata with its Best Practices Award for Data Warehousing.

The year 1996 also saw NCR, along with Teradata, become an independent, publicly traded company. In 1997, Teradata client Wal-Mart created the world's largest production database at the time (24TB) and Teradata received the Data Warehouse Institute Best Practices Award and DBMS Reader's Choice Award. Teradata pioneered in-database data mining in 1998.

By 1999, another Teradata client created a production database with 130TB of data on 176 nodes. A significant milestone was reached in 2002, when Teradata launched Teradata Warehouse 7.0, an advanced suite of data warehousing hardware and software. That marked the first time in data warehousing history that any vendor extended decision-making beyond corporate management to all functions across the organization.

### **Forward thinking**

This year witnessed the launch of the Teradata University Network, with nearly 170 universities from 27 countries involved in the advancement of data warehousing in the academic community. Teradata also has partnered with SAP, the world's leading provider of business software solutions, to deliver analytical solutions to industries with high data-volume requirements, including telecommunications, financial services, pharmaceuticals, aerospace and others.

Another partnership with Siebel Systems, Inc. makes available a business intelligence platform exploiting specific Teradata capabilities, enabling customers to leverage the complete functionality and performance provided by the Teradata Warehouse.

Colin White, founder and president of BI Research, is known for his knowledge of business intelligence and enterprise business integration. He is a respected consultant and a frequent speaker at leading IT events.

© Teradata Magazine-September 2004

### RELATED LINKS:

- > **Chris Date's *Intelligent Enterprise* article**
- > **E. F. Codd's *Communications of the ACM* article**
- > **Emerson Pugh's book *Building IBM***

[back to top](#)