DOI:10.1145/1831407.1831409

How to Celebrate Codd's RDBMS Vision

While we were pleased *Communications* celebrated E.F. Codd's seminal article "A Relational Model of Data for Large Shared Data Banks" (June 1970) in "Happy Birthday, RDBMS!" by Gary Anthes (May 2010), we were also dismayed by its inaccuracies and misrepresentations, including about more than just pre-RDBMS history.

For example, saying "Codd's relational model *stored* data in rows and columns…" (emphasis added) is completely at odds with Codd's goal that "Future users of large data banks must be protected from having to know how data is organized in the machine." Rows and columns are the canonical representation of Codd's relations, not a constraint on physical data structures. Getting this wrong completely undermines Codd's contribution. Moreover, no viable commercial RDBMS has stored data purely in rows and columns, nor has any vendor completely implemented the logical and physical data independence his theory made possible.

Other inaccuracies and misleading statements abound:

DB2 did not "edge out IMS and IDMS." It took a long time for the transaction rates of any commercial RDBMS to compete with those of IMS, which remains an important commercial DBMS;

Ingres and its derivatives did not have the "DEC VAX market to themselves." Interbase, Oracle, and Rdb/VMS were early players (1980s), and Ingres was initially available on VAX/VMS but like many RDBMS products that preceded the IBM products introduced on Unix;

The "database wars" raged for almost two decades. Relational repeatedly had to prove itself against network, hierarchical, and object-oriented DBMSs, continuing with XML and Hadoop contenders;

Map/Reduce is a non declarative programmer's distributed query template, and the Hadoop Distributed File System is a storage model. Neither rises to the level of data model or programming language;

Whether it was "easier to add the key features of OODBs to the relational model than start from scratch with a new paradigm" never happened. At best, features were added to SQL and SQL-based products, but these misguided additions did violence to the relational model's way of achieving desired capabilities, namely extensible domain support;

"Querying geographically distributed relational databases" is not unsolved. Implementing the relational model's physical data independence solved it;

Since 1980, numerous RDBMS products have provided partial implementation of physical data independence and been widely used in industry. Perhaps David DeWitt [cited by Anthes and director of Microsoft's Jim Gray Systems Laboratory at the University of Wisconsin-Madison] was referring to the problems of querying heterogeneous, distributed data with inadequate metadata, since he was quoted saying databases "created by different organizations" and "almost but not quite alike"; and

Database scalability has always been about numbers of concurrent users and locations, user variety, and manageability, not just data volumes. One of us (McGoveran) published (late 1980s, 1990s) studies evaluating scalability of commercial products along these lines.

David McGoveran, Boulder Creek, CA
C.J. Date, Healdsburg, CA

Author's Response:

E.F. Codd's model let users "see" their data as if it were stored in ordinary tables, rows, and columns. This was easier for them to understand than the pointers and hierarchical trees used in other models. Such simplification was one reason the RDBMS model edged out IMS and IDMS, though IMS is still used in a few narrow (but important) niches. Alas, vendors did not implement Codd's rules in their purest form, as McGoveran and Date point out.

Gary Anthes, Arlington, VA

[redacted]

Communications welcomes your opinion. To submit a Letter to the Editor, please limit your comments to 500 words or less and send to letters@cacm.acm.org.