



DBMS Trends into the 21st Century

The editors of IDUG Solutions Journal have invited two DB2 experts to duel peacefully on key issues facing DB2 today. In this regular column, Willie Favero of BMC Software and Craig Mullins of PLATINUM technology will face off. Sometimes, they even agree.

BY WILLIE FAVERO

While dabbling on the Internet a few weeks ago, I received a note from a DBA expressing concern that her DB2 for OS/2 skills might go to waste. I was struck by the thought that I would have been more concerned about her OS/2 skills, not what she had learned about DB2 for OS/2. The note prompted me to consider yet again how quickly our industry is changing and yet how much of it stays the same.

Remember several years ago when client/server was going to revolutionize the IT world, reduce IT costs, and thereby kill the "expensive" mainframe? As we all know, the death of MVS has been greatly exaggerated. Not much later, applications began returning to the big iron. MVS, or rather OS/390 today, became the server of choice in many client/server environments. MVS can handle today's high transaction rates, has strong backup and recovery capabilities that exist in both OS/390 and the databases that run on OS/390, and it solved many client/server security issues some years ago.

What happened next? Along came the Internet, and with it Intranets and Xtranets. The Internet, it was predicted, would now replace client/server and again threaten the longevity of MVS and the mainframe. Yet the mainframe is — once again — emerging as the repository for much of the information that will be accessed over the Internet. Moreover, MIPS growth over the last two years has increased by as much as 50 percent. MVS and the mainframe's ability to survive each new era in data processing no longer surprises anyone anymore.

As many of you know, hardware is not my forte — database and programming are. Having started with IMS/360 in the early 1970s, I have watched many things come and go. Some I miss, and some I am happy to see disappear. I have

BY CRAIG S. MULLINS

The IS organization has evolved from one of centralized control to an open, heterogeneous environment. Once upon a time, the mainframe was the centralized point of control for all application development and data management activities, but today it is only one small piece of an increasingly complex puzzle.

Managing in a heterogeneous environment

The biggest challenge faced by database administration groups is the task of managing heterogeneous database environments. Most organizations have more than one DBMS and many have three or four. It is not unusual for medium to larger shops to have DB2, Sybase, Oracle, Informix, and Microsoft SQL Server installed and housing production data. The fact that each of these is a relational DBMS does little to diminish the pain of administering and managing the data stored therein.

Why is this so? First and foremost, each RDBMS vendor has implemented different procedures and syntax for implementing and managing databases. Although ANSI is working on a standard SQL, none of the vendors implements plain vanilla ANSI SQL. Each RDBMS supports a certain level of ANSI SQL compliance, but product-specific extensions abound. Try learning Sybase's Transact-SQL and see how difficult it is to translate into COBOL for DB2 stored procedures, or into Oracle PL/SQL or Informix SPL. Additionally, the user interface varies from product to product, making it difficult for the DBA to switch from, say, Oracle to DB2.

Furthermore, heterogeneity poses problems that would exist even if all of the DBMSs were 100 percent open and interoperable. How do you implement changes on multiple platforms and keep the data in multiple DBMSs synchronized? Unless a database

been with mainframe DB2 (now DB2 for OS/390) nearly 15 years, and I am happy to say that DB2 is going to be around for a while — a long while. Even more exciting is that IBM has finally gotten a chance to be a major competitor again with its DB2 Universal Database Version 5 (UDB).

UDB still seems to lack the application vendor support Oracle has always enjoyed. However, IBM has done a solid job of combining parallel processing and object support with an already strong relational database engine. With Oracle8 and IBM's UDB Version 5 looming as formidable opponents over the next few years, Microsoft's NT operating system should become the major battle front. UNIX seems unsure of what it really wants to be, with its different flavors running on different operating systems. OS/2 is still putting up a fight but is struggling for true acceptance in the marketplace. It is NT that is emerging as the clear and unchallenged leader. Although OS/2 does have features that should make it *the* dominant OS, it simply has not experienced the broad acceptance by the application vendors that Windows 95 and NT enjoy.

It's the application

These days, superior technology no longer seems to be the force



"...the larger the warehouse grows, the longer queries will run. To reduce query times...will require taking advantage of all the processing power available. Parallelism unlocks that power."

"The true value of data warehousing is that it provides more accurate information in a more timely manner...enabling businesses to quickly react to rapidly changing business conditions...gaining a competitive advantage."

driving operating system and DBMS purchase decisions. More often, it is the application that needs to be implemented. Application software packages are forcing the underlying DBMS and operating system decisions in a new top-down decision making process. Rather than choosing a package that will run in your current environment, you are making purchase decisions based on the package's functions and merits. Only after the package is chosen do you purchase and install the appropriate database and operating systems. For some, packages are becoming year 2000 solutions. Packaged software are becoming the alternative to rewriting currently installed applications for the year 2000.

CONTINUED ON PAGE 20, COLUMN 1

administration tool is used that understands the open enterprise environment of the organization, synchronization must be done manually. How do you coordinate application changes with database changes? When a data structure changes, quite obviously some application program somewhere is going to need to access that data. Thus, the program must be changed. To reduce errors and enhance integrity it would be nice to integrate the database management tool with an automated software distribution tool. This becomes increasingly complex in client/server environments where application programs need to be distributed to hundreds or thousands of client nodes across the network.

Integrated management tools

To take the previous analogy a step further, shouldn't all of our management tools be integrated so that management information is shared, the user interface is common across tools, and management is automated and proactive instead

of reactive? Integration is definitely on the horizon. Consider the following scenario. Your database performance monitor hits a threshold indicating a performance problem. The problem

requires a database to be reorganized. A job is automatically scheduled for the best time using your database reorg tool and your job scheduler (on Windows NT, UNIX, MVS, or whatever). A trouble ticket is automatically sent to the help desk indicating that performance could be troublesome until the reorg occurs, but it is scheduled at a specific indicated time. The time comes for the reorg to run, and the scheduler automatically blocks all other jobs using that database, then runs the reorg. The problem is resolved, so the trouble ticket is removed. And all of this occurs automatically and proactively with no administrator involvement. Wouldn't that be nice?

CONTINUED ON PAGE 20, COLUMN 2



(One side benefit of the approaching year 2000 is the resurrection of the COBOL programmer.)

Object orientation

If technology alone were setting trends for the 21st century, it would enhance the strength of Object Oriented application design and programming — a trend I would really like to see. Although I believe that acceptance of an OO database is a long way off, OO programming has tremendous potential to change the face of design and development once and for all (again!). Consider where many shops would be today in their year 2000 conversion efforts if they had jumped on the object bandwagon just a few years ago.

OO is also going to play an important role in the future development of relational databases. The OO idea of strong typing is needed by RDBMSs to support the different data types corporations will come to rely on in the years ahead. Spatial data, large binary and character objects, and notions as simple as supporting different kinds of dollars in the same data source will all be entrusted to user defined data types and the user defined functions that support them.

Parallel technology will also be a significant trend as data warehousing and data mining become major mainstream processes. Corporations are learning that all that information they have been gathering for years can give them an invaluable competitive edge. Yet, the larger the warehouse grows, the longer queries will run. To be useful, more and more data must be moved into the warehouse. To reduce query times, and in some cases simply to make processing a query possible, will require taking advantage of all the processing power available. Parallelism unlocks that power. Corporate decisions today need to be made in minutes, not in weeks or months.

We all know the only trend we can really count on: tomorrow is not going to look anything like today. Which, by the way, is good. One of the things that makes this industry so exciting is that it is always changing. ☺

ABOUT THE AUTHOR

Willie Favero has been a database professional for more than 20 years, the last 14 years primarily with DB2. He has been a software consultant for BMC Software, Inc. and a senior DB2 instructor for IBM. Favero is the author of numerous articles, a contributor to several IBM Redbooks, and a regular speaker at regional, national, and international conferences. He can be reached via email at willie_favero@compuserve.com.



Packaged applications

With technology changing so rapidly, time is not always available to develop applications internally. As a result, third party applications are increasing in popularity. Just look at the rapid growth of SAP into one of the top five independent software vendors. And SAP is not alone; other popular application vendors include Peoplesoft and Baan. Five years ago, almost no one had even heard of them. Today's organizations are wrestling with managing the database structures that ship with these applications. The databases must be generic so that any of the major RDBMS products can be used with the application. The down side, however, is that the design is not optimized for the product. Moreover, code changes cannot be made to the product — the result is that organizations are struggling with how to optimize these applications and what, if anything, can be changed.

Data growth and data warehousing

Perhaps the hottest trend in database management today is data warehousing. Data warehousing separates production transactions from ad hoc, analytical queries. Data warehousing imposes a development methodology involving data refining (scrubbing and transformation), data movement (replication and propagation), and data documentation (storing metadata in a corporate repository). The true value of data warehousing is not only that it provides businesses with more information, but more accurate information in a more *timely* manner. In this way the data warehouse enables businesses to quickly react to rapidly changing business conditions and thereby gain a competitive advantage. So the data warehouse becomes as critical in terms of administration and management as the production systems are. And, you won't be surprised to learn, the data warehouse is often implemented in incremental data marts using multiple database structures on multiple platforms. By increasing heterogeneity, the job of the DBA is further complicated.

An offshoot of the data warehousing trend is data mining. Data mining is the process of heuristically discovering heretofore unknown information and patterns lurking within your organization's data. Just imagine what happens to data access and database performance when automated data mining tasks are executed against legacy data.

All this warehousing and mining is at the core of another key trend: an ever-increasing amount of data being stored, managed, and manipulated. Multi-terabyte databases are common and petabyte databases will be just as common by the turn of the century. Look what's next! — exabytes, zettabytes, and even yottabytes (*see chart*). We are indeed moving from VLDBs to VHDBs (very huge data-

CONTINUED ON PAGE 21

bases). The net result of huge databases is the increasing difficulty of database administration and management. How do you back up a terabyte of data? Indeed, a Gartner Group study shows that the largest production database at any given point in time is not administrable — given the technology and tools at hand!

Abbrev.	Term	Amount
KB	Kilobyte	1,024 bytes
MB	Megabyte	1,024 KB
GB	Gigabyte	1,024 MB
TB	Terabyte	1,024 GB
PB	Petabyte	1,024 TB
EB	Exabyte	1,024 PB
ZB	Zettabyte	1,024 EB
YB	Yottabyte	1,024 ZB

Databases are going online

The rapid acceptance of the Internet and the World Wide Web as a source for information is forcing companies to hook their databases into the Internet. Web browsers are fast becoming the de facto method for gathering and sharing information outside (Internet) and inside (Intranet) corporations. Now that Java can be used to write applications to be executed on the Inter- and Intranet by your favorite Web browser, many administrative quagmires will develop. How do you ensure that Java applets can access your corporate databases over a vast network? Furthermore, how do you tune a database that could potentially be accessed by every Web user in the world? And how is down-time managed? Anyone who has surfed the web knows that a site can be available one second, but not necessarily the next. These are new issues with which DBAs are not accustomed to dealing.

A side effect of the Internet is the ready availability of a virtual user group — usenet. Usenet newsgroups provide online access to a vast number of database experts, with newsgroups available for DB2, Informix, Oracle, Sybase, Microsoft SQL Server, and Ingres.

Extending the database

Traditionally, DBMS products stored data and nothing else. But all of today's major RDBMS products support procedural logic in the form of triggers, functions, and stored procedures, otherwise known as Server Code Objects (or SCOs for short). Of course, not all RDBMSs support every SCO, nor do they implement SCOs in the same way. This greatly complicates the job of the DBA. But storing procedural logic in the database is here to stay, because it enhances performance of client/server applications, eases security, and promotes reusability.

Last, but definitely not least, is the freight train known as object orientation, or OO. The basic idea behind OO is that

process should be encapsulated with data, thereby creating complex objects (instead of the simple rows and columns we use with RDBMSs). Objects increase the possibility of reuse and provide structures for dealing with the complicated data with which relational technology has traditionally struggled (CAD/CAM drawings, bill-of-material hierarchies, for example). Several pure object-oriented DBMS products are available, but none of them provides the full functionality of the major RDBMSs, making it unlikely that we will have to “unlearn” relational technology and learn OO technology in its place. All of the major RDBMS products are being re-vamped to support objects in upcoming releases, creating what is being called a Universal DBMS. However, the advent of Universal DBMS products will further complicate database administration, because objects will need to be modeled, implemented, maintained, and administered — just how, for example, do you reorg an object?

Synopsis

As I peer into the future, I see a Java application running within a Web browser accessing object relational data from heterogeneous databases. Down the hall the marketing department has just discovered a new customer purchasing pattern in the sales database using an automated data mining tool. The DBAs are invoking changes to multi-terabyte DB2 and Sybase databases from a single management console with no database down time. And my database administration tools are integrated with my systems management tools so everyone uses the same interface. Or am I just dreaming? ☺

ABOUT THE AUTHOR

Craig S. Mullins is vice president of marketing and operations for the database tools division of PLATINUM technology, inc. He is also the author of the popular book, *DB2 Developer's Guide*, which will soon be available in its third edition; the book will include tips, techniques, and guidelines for DB2 through Version 5.

What a Picture!...SILVERRUN and DB2

and DB2/AIX and DB2/400 and Oracle and Sybase and Informix and Access and so many others you manage...

Call for information on SILVERRUN modeling with DB2 and the RDBMS sub-schemas of your choice, with your VSAM conversions, and synchronizing your model and database.

Larry Kintisch, MS, CSP • Able Information Services

DB2 / SILVERRUN / Data Modeling Consulting & Training

(914) 353-3809 • e-mail: lkint@ix.netcom.com