RUSS WILLMS

It's a bird. It's a plane. IT'S AN eDBA!

With billions of dollars of revenue at stake, databases supporting e-businesses are putting new demands on DBAs. Luckily, there's a new class of super DBAs putting Web skills to work to save the e-commerce day. Do you have what it takes?

CRAIG S. MULLINS

Internet usage is increasing at a rapid pace and infiltrating every aspect of our lives. From online shopping to URLs displayed on TV commercials and billboards, the Internet is truly everywhere. And the Internet is dramatically altering the way we do business. It has created expectations for businesses to be more connected, more available, and more useful. The need to integrate the Web with traditional IT services such as database management systems places heightened expectations on those who manage the databases and keep them running — database administrators.

14 MUST-HAVE DBA SKILLS

Before DBAs can run, they've got to know how to walk. These skills form the base from which more specialized eDBA skills can grow.

1. **DATA MODELING AND DATABASE DESIGN:** the ability to create an efficient physical database design from a logical data model and application specifications

2. **METADATA MANAGEMENT AND REPOSITORY USAGE:** the ability to collect, store, manage, and query data about the data in the database

3. **DATABASE SCHEMA CREATION AND MANAGEMENT:** the ability to translate a physical database design into an actual physical database implementation and to manage that database once it has been implemented

4. **PROCEDURAL SKILLS:** the ability to design, debug, implement, and maintain stored procedures, triggers, and user-defined functions that are stored in the DBMS

5. **CAPACITY PLANNING:** the ability to predict growth based on application and data usage patterns and to implement the necessary database changes to accommodate the growth

6. **PERFORMANCE MANAGEMENT AND TUNING:** the ability to proactively monitor the database environment and to make changes to data structures, SQL, application logic, or the DBMS subsystem to optimize performance

7. **SQL CODE REVIEWS AND WALKTHROUGHS:** the ability to understand and review SQL and host language programs and to recommend changes for optimization

8. **BACKUP AND RECOVERY:** the ability to implement an appropriate database backup and recovery strategy based on data volatility and application availability requirements

9. **ENSURING DATA INTEGRITY:** the ability to design a database so that only accurate and appropriate data is entered and maintained

10. **GENERAL DATABASE MANAGEMENT:** knowledge of relational database tenets and the ability to accurately communicate them to others

11. **DATA SECURITY AND DATA INTEGRITY:** the ability to implement a rigorous security infrastructure for production and test databases to ensure that only authorized users have access to data

12. **GENERAL SYSTEMS MANAGEMENT AND NETWORKING SKILLS:** the ability to integrate database administration requirements and tasks with general systems management requirements and tasks

13. **ERP AND BUSINESS KNOWLEDGE:** the ability to understand the requirements of the application users and to administer their databases to avoid interruption of business. This includes understanding how any ERP packages impact the business and how the databases used by those packages differ from traditional relational databases

14. **EXTENSIBLE DATA TYPE ADMINISTRATION:** the ability to understand, implement, and administer any extended data types implemented in the DBMS.

When your business is online, it never closes. People expect full functionality on Web sites they visit regardless of the time of day. And the Web is worldwide. It may be two o'clock in the morning in New York City, but it is always prime time somewhere in the world. An e-business must be available and operational 24 hours a day, 7 days a week, 365 days a year. It must be prepared to engage with customers at any time or risk losing business to a company whose Web site is more accessible. Those who manage an e-business must be adept, proactive, and ever vigilant.

As companies begin to transform themselves into e-businesses, their applications will become complex, integrated systems consisting of traditional and legacy IT coupled with Web-based technologies. A key component of these applications will be access to and storage of mission-critical data. And the most reliable, time-tested method of storing persistent data is inside a database.

As good as today's DBAs might be, they will see their roles and skill requirements changing quickly. In fact, the Patricia Seybold Group estimated that a Web browser initiated half of all database queries in 1998. To meet the challenges posed by tying database technology to the Internet, you'll need to be more than a standard DBA. You'll need to become an eDBA, a database administrator capable of managing Web-based applications who understands the special challenges the Internet presents. Of course an *eDBA* also needs all of the knowledge and training of a traditional DBA (see "14 Must-Have DBA Skills"). However, eDBAs must also have a special skill set that's adapted to suit applications and databases hooked to the Web.

Coupling the Internet with database technology impacts many areas of the DBA's job. An eDBA who's up to the challenges will have four key competencies.

1 Able to keep a Web site functioning and available 24×7.

When you hook a database up to the Web, data availability requirements skyrocket. Users expect Web data to be available whenever they happen to go online. Of course, some go online early in the morning to check news or traffic, get directions, or check yesterday's stock prices. Others sign on late at night to place stock trades to be executed the next day, buy books and CDs, or do research for tomorrow's presentation. There is no offline time.

eDBAs must understand that any time the data in their databases is unavailable for any reason, it impacts the business. If a customer tries to visit your site but finds it inaccessible, the chance that the customer will return diminishes. Unavailable data decreases revenue, impacts profitability, the company's stock price, future pay increases, and perhaps

even job security.

Take the recent outages at eBay, for example. As the leading auction site on the Internet, eBay's customers are both the sellers and the buyers of items put up for bid on its Web site. The company's business model relies on the Web as a mechanism for putting buyers in touch with sellers. If buyers can not view the items up for sale, the model ceases to work. In June, eBay launched an overhauled, newly designed site. Around the same time, they experienced outages on their database server.

To resolve the problem, data had to be restored. eBay customers could not reliably access the site for several days. Auction timeframes had to be extended. Bids that might have been placed during that timeframe were lost. eBay agreed to refund all fees for all auctions on its site during the time when its systems were down. Some reports estimated that this series of outages could impact quarterly profits by $3 to $5 million. This, in turn, caused the stock to drop from a high of $234 in April to the $130 range in mid-July. eBay is a great site, a good business model, and a fine example of an e-business. But better planning and preparation for "e-database administration" could have reduced the problems.

Data may become inaccessible for many reasons. Anyone who has ever surfed the Web knows that a page that was available just five seconds ago can suddenly produce that exasperating "Error- 404 requested information is unavailable" message. This message means that the data is not available because of an external problem (for example, the ISP might have encountered problems with its hardware, software, network, or connectivity). There is nothing an eDBA can do to prevent external problems from occurring, but awareness of the potential problem is crucial. Otherwise, you could spend many hours searching for a local database-related problem that doesn't exist.

Publishing your database data on the Web brings added complexity to your environment. And some of the factors will be out of your immediate control even if you are an eDBA. Consider the number of technologies and connections required for a Web-enabled database application to function:

- The DBMS resides on physical hardware, which could be improperly configured and tuned.
- The hardware requires an operating system, which could be functioning suboptimally.
- The database schema used by the Web application could be inefficient.
- The Web application accesses data in the database using SQL, either embedded in a program or using a CLI. The host language program, the SQL, or both could be inefficient.
- The application may need to access the database via an internal network that could be improperly configured.
- The internal network requires physical hardware such as routers, hubs, switches, and cables, any of which could be inefficiently implemented.
- The Web application may rely upon CGI or Java applets, which could be inefficient.
- Domain name services (DNS) may be inaccessible.
- Your ISP may be experiencing problems.
- Internet traffic may be heavy, causing inordinate wait times to access your site. Customers may just hit the STOP button and move on instead of waiting.

Although the first six issues can be experienced by any database application, the remaining issues are unique to Web-based applications. The eDBA must quickly troubleshoot internal systems

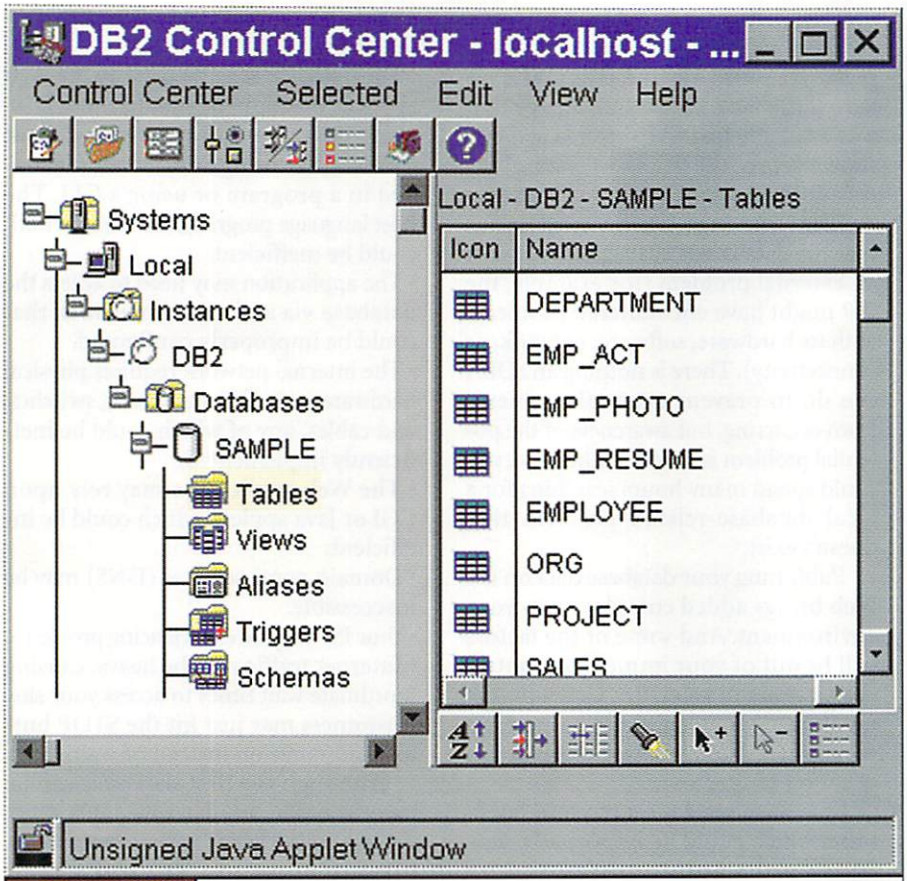


Figure 1 *IBM's DB2 Web Control Center.*

Object	Typical Size
• High-definition television	200MB/second
• Feature-length, high-resolution movie	5-6GB
• High-resolution video	3GB/hour
• Feature-length movie	2GB
• Video	1GB/hour
• Radiologic image	40-60MB
• Color image	20-40MB
• Large image	200KB-3MB
• Text	30-40KB/page
• Check image	45KB
• Small image	30-40KB

Figure 2 *LOBs mean big databases.*

and software to ensure that the problems are not internal before investigating external causes of availability problems.

Every eDBA should be armed with the appropriate tools to track down problems, whether they are internal or external. Network packet sniffers, response-time measurement tools, and transaction tracking tools can help. Examples include BMC Software's Patrol and Compuware Corp.'s EcoScope. Without these tools, tracking data availability problems can be very difficult.

The eDBA pays special attention to database backup and recovery. If data needs to be recovered, it's up to the eDBA to ensure that a relevant backup is available. Although this is true for all database applications, e-business applications have high availability requirements. Because of those requirements, eDBAs must use hot backup technologies that enable data to be online and available for query — and even modification — during the back-up. The recovery window must be as small as possible to keep data available for your 24×7 Web site.

DB2 UDB for AIX supports high availability through High Availability Cluster Multi-Processing (HACMP), for example. HACMP offers an efficient way to guard against unplanned outages and server downtime. Using HACMP for AIX, you can configure your system for standby availability so that downtime is diminished. Benefits include guarding against unplanned outages and server downtime, a simple reconfiguration process that makes the data and services of a failed node available to clients as transparently as possible, and fully automatic detection and recovery — all useful to eDBAs.

An eDBA needs to constantly monitor the volatility of the data and tweak backup and recovery plans so that backups are taken at the appropriate time and log files are available to recover to specific points in time, or up to the time of failure. Some third-party tools that can implement these procedures are Computer Associates' Recovery Analyzer and BMC Software's Recovery Manager. These tools have additional functions that enable you to more quickly build efficient recovery plans; and such utilities sometimes run faster than the traditional database copy and recover utilities, reducing the time required for any necessary data recovery.

2 Tackles performance engineering in a single bound.

If performance degrades to the point that applications become unusable (because customers won't wait for the screen to return), then the data is unavailable for all intents and purposes. Performance engineering is critical for Web-based database applications.

Many of the same factors that influence data availability also impact performance issues. Proactive performance management tools can be set up to monitor Web server processes. A Web slowdown can affect everything running on the system — including non-Web work. Using performance tools, an eDBA can configure alerts to trigger agents that detect resource usage problems and execute actions using other database and systems management tools to correct the problems.

For example, BMC Software offers Patrol Knowledge Module for Internet Servers, which provides built-in exper-

tise and automated administrative control for the Internet server environment. It provides proactive monitoring and management of Web servers to ensure peak availability and performance. In conjunction with the Patrol Knowledge Module for DB2, you can create an integrated performance management environment that ensures high availability of Web servers and DB2. Using Patrol, you can manage your DB2 and Web performance from a single console. This saves time by automating administrative tasks, tracking availability metrics, and locating and correcting potential bottlenecks and problems.

With a Web-based application, the user population is very unpredictable. Fifteen years ago, when most applications were host-based, mainframe applications, performance problems were easier to track down. For the most part, you knew the cause of the problem was somewhere or someone inside the building. Ten or so years ago, at the dawn of the client/server age, things became more difficult. Because client/server enables distributed platforms, applications, data, and processing, users were spread throughout many buildings. The set of potential problem causes increased. When a problem occurred, you knew the culprit was somewhere or someone in the company. With Web-enabled applications, the set of potential problem causes increases yet again — the best you can say is that the cause of your performance problem is somewhere or someone in the world!

An eDBA needs to be able to monitor the Web application and respond accordingly to performance problems. This may include traditional application and SQL performance monitoring and tuning, but must also include tracking the Web server, application, and data availability. You should be able to monitor your entire system from one console, which makes finding internal problems easier so you can turn your eye to potential external causes. You should also be able to track performance statistics for your Web server, ISP, and applications and have a mechanism to locate and automatically correct as many errors as possible over your complete IT system, including your Internet servers.

Every DBA knows that as databases are modified, they can become disorganized. Running a reorganization utility can resynchronize the data efficiently. An eDBA will need to consider using utilities that increase data availability, such as an online reorganization utility, which keeps the data accessible during the reorganization process.

As of DB2 for OS/390 version 5, the SHRLEVEL parameter has been added to the REORG utility, keeping data available during reorganization. As with SHRLEVEL in other DB2 utilities, the SHRLEVEL parameter controls the level of concurrent data access permitted during a reorg. There are three SHRLEVEL options for REORG:

1. **SHRLEVEL NONE** indicates that concurrent data reading is permitted while data is being unloaded, but no access is permitted during the RELOAD phase. This is the default and the manner in which REORG is executed for all versions of DB2 prior to v.5.
2. **SHRLEVEL REFERENCE** indicates that concurrent read access is permitted during both the UNLOAD and RELOAD phases of the REORG.
3. **SHRLEVEL CHANGE** indicates concurrent read and write access is available throughout most of the reorganization.

Both SHRLEVEL REFERENCE and SHRLEVEL CHANGE require a shadow copy of the object being reorganized, as do third-party online REORG utilities from Computer Associates and BMC Software. You will require twice the amount of disk storage to perform an online reorg, so you must be careful to plan for the space and make sure it's available when reorganizing highly available Web databases.

Perhaps the most exciting advantage of Web-based computing is that it can be used to assist with its own management. As more applications and tools with Web interfaces become available, the Web can be exploited for systems management. IBM's DB2 Web Control Center, a Java version of the DB2 Universal Database Control Center, is a database administration tool similar to the original DB2 Control Center, but with a more flexible network-centric administration environment (see Figure 1,). (*Note:* Control Center was recently extended to manage DB2 on OS/390 as well as on Unix, Windows and OS/2). DB2's Web Control Center is implemented as a Java applet that uses DB2's JDBC support. Web-based administration tools allow an eDBA to manage Web-based applications from anywhere.

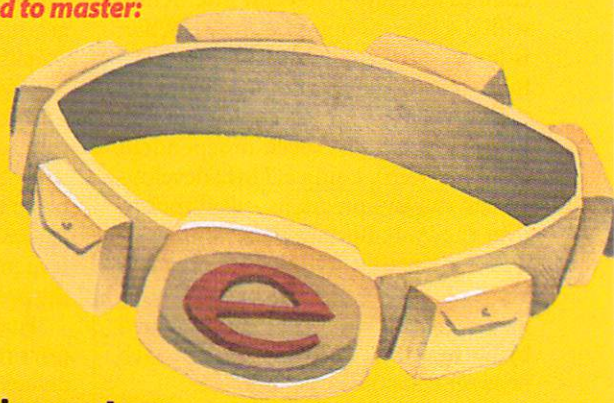
Masters application architecture faster than a speeding bullet.

One of the most significant ways the Web changes the application architecture used to access the database is through the use of Java. Already one of the most pervasive languages used on the Web, Java is predicted to gain popularity even more rapidly over the next three years. Because applications architected for the Web are coded in Java, the eDBA needs to understand how Java works, how it is coded, how SQL is used with it, and how it integrates to the Web.

But what is Java? First and foremost, Java is an object-oriented programming language. Developed by Sun Microsys-

THE E-TOOLBELT

With a complete set of Web-related skills, any DBA can be an eDBA. Here are the technologies you'll need to master:



- **HTTP**
- **FTP**
- **XML**
- **Web development tools, such as CGI**
- **Java**
- **TCP/IP**
- **Net.Data**
- **Web server operating systems**
- **Highly available storage technologies, such as RAID tools, and techniques for making database changes and perform database tuning without impacting data availability**
- **Security methods, including SSL, firewalls, database security (DCL), and network/OS security**

tems, Java was modeled after, and most closely resembles, C++. But it requires a smaller footprint and eliminates some of the more complex features of C and C++ (such as pointer management). Java enables animation for and interaction with the Web. Although Web interaction is Java's most touted feature, Java is a fully functional programming language that can be used for developing general-purpose programs (even independent from the Web). Using HTML, developers can run Java programs, called applets, over the Web. But Java is a completely different language from HTML and does not replace HTML. Java applets are automatically downloaded and executed by users as they surf the Web. The Web browser runs the Java applet.

What makes Java special is that it was designed to run on any platform, which can mean reduced development and maintenance costs, lower systems management costs, and more flexible hardware and software configurations.

An eDBA must understand Java or else will be incapable of application tuning in a code walkthrough. All DBAs know that you need to monitor not just the SQL but also the host language code (in this case Java) in which the SQL is embedded to properly tune database performance.

Although there are many benefits to Java, there are also downsides. For example, a Java program is coded and compiled into Java bytecodes to give it its hardware independence. The hardware requires a Java Virtual Machine (JVM) to interpret the bytecodes for execution of the program or a just-in-time (JIT)compiler, which interprets the bytecodes just before they are executed, allowing the code to run as machine language on the destination platform. But Java is an interpreted language — like BASIC — which causes it to run more slowly than a compiled, executable program.

eDBAs may encounter JavaBeans, which are reusable object components. Using JavaBeans, developers can assemble object-oriented components into operational applications more quickly than they could by coding everything from scratch. eDBAs need to be able to understand how the applications were built and how the performance of a JavaBean might impact multiple applications, instead of just one. Reusable components should be tuned more stringently than programs used only once within the system.

You can access databases with Java using either JDBC or SQLJ (DB2 supports both standards). JDBC uses an API to enable Java to access relational databases. Similar to ODBC, JDBC consists of a set of classes and interfaces that can be used to access relational data. Anyone familiar with application programming and ODBC (or any call-level interface) can get up and running with JDBC very quickly. SQLJ enables embedded static SQL for Java programs, giving full support for all DML, DDL, DCL, commit/rollback processing, and cursors. When using SQLJ, you use a precompiler to replace the SQL with code to access the database, after which you can compile the Java program into bytecodes and BIND the precompiler output into a package.

Furthermore, IBM provides Java support for stored procedures and user-defined functions in DB2. Undoubtedly, Java will be one of the most popular options for these features because of its ability to automate Web processes.

An eDBA needs to understand Java programming differences so that the appropriate development procedures can be implemented. Every eDBA knows that the application logic, be it 3GL or SQL, is the primary cause of most performance problems. eDBAs must understand Java and how SQL is used with Java in order to achieve optimal performance. As an eDBA, you also need to understand the nuances of how Java is executed so that an effective tuning strategy can be achieved when Java, JDBC, and SQLJ are in the mix.

Architects for different data types to take Web content up, up, and away.

Web-based applications require database architectures different from other applications. Web-based applications store more different types of data than traditional IT applications do. Pictures, maps, audio, and video are more common on the Web, and users expect to see them. But these types of data increase the size of the database and require different storage mechanisms than do traditional numeric, character, and date/time data types (see Figure 2, page 50). As database size increases, manageability decreases. In other words, the bigger the database, the more difficult it is to keep the data available for efficient access.

You can use DB2 Universal Database Extenders to enable your database applications to access data beyond the traditional numeric and character data types. DB2 Extenders let you store images, video, voice, and complex documents in DB2 UDB databases, access them using SQL, and manipulate them with powerful functions. DB2 UDB support for data type Extenders is important for Web-based applications because Web surfers expect complex, multimedia content. A data type Extender consists of a large object (LOB), a UDT to define the structure of the LOB, UDFs to manipulate the data, and triggers to maintain data integrity. DB2 UDB Extenders are available for Unix, Windows, and OS/2, and OS/390.

eDBAs must analyze the large object data type requirements of Web-based applications. Planning for appropriate storage and growth is of paramount importance. However, the eDBA needs to be aware of how LOBs change the structure of the databases being implemented. When LOBs are used in DB2, additional tablespaces are required for storing them, which means more planning, implementation, and administration.

Furthermore, Web-based applications frequently rely on triggers and stored procedures. Triggers and stored procedures consist of program logic that the DBMS manages. When code is stored in the database, different administration practices are required, including debugging, code management, and change management. eDBAs must be prepared to manage this code but should interact with the application developers to ensure the code is written optimally and accurately.

eDBAs IN DEMAND

The Web has the potential to touch every part of your business, so it would be impossible to summarize all the skills an eDBA needs in a single article. Additional concerns include:

- Data security (data access, firewalls and so on)
- Search engines (should you make your data available and, if so, how?)
- Privacy (protecting confidential customer information and transactions).

Mission-critical Web-based applications are increasingly popular because of the business benefits they provide, including rapid application development, cross-platform deployment, and robust access to data at any time, from any place. Because these are such important benefits, the number of e-businesses will continue to grow. And so will the demand for the eDBA. ●

Craig S. Mullins is president and lead consultant with CS Mullins & Associates, a firm specializing in relational database technology consulting, training, and research. He is also the author of *The DB2 Developer's Guide* (Sams Publishing, 1997). You can reach him at cmullins@compuserve.com.