

# DB2<sup>®</sup> magazine

Quarter 1, 2001 • Volume 6 Number 1

*Sponsored by IBM Corp.*

## POWERING UP: DB2 UTILITIES



Database  
Security

Mobile  
Business



GC26-9029-16

PRINTED IN USA

Subscribe online at [www.db2mag.com](http://www.db2mag.com)





**Batch  
window  
constraints?**

***Not with Smart/RESTART***

Smart/RESTART lets your applications restart from near the point of failure – after abends, recompiles, even system IPLs.

***Restart made simple™***

Your applications can run restartably – often without source changes or even recompiling! Smart/RESTART guarantees that changes to DB2, IMS and MQSeries stay in synch with your program's working storage, sequential file position and VSAM updates. So you can restart fast with assured integrity.


For a free trial call  
800 776-0771

***www.relarc.com***

DB2, IMS and MQSeries are registered trademarks of IBM Corp.

***fai*** Relational  
Architects  
Internationa





*It's easier  
to find a needle  
in a haystack when  
the haystack comes with  
a multicriteria,  
searchable database.*

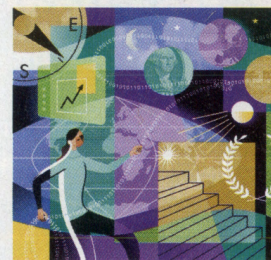
Presenting the **Global Solutions Directory** by PartnerWorld for Developers

Finally, there's a single source for business software solutions. The Global Solutions Directory is a comprehensive online directory of applications, tools and services created by some of the industry's leading developers using IBM technologies. You can search this directory free of charge – anytime, day or night – to find the ideal solutions for your business.

With thousands of entries, you're bound to find what you're looking for. Each listing gives you product descriptions, company information and even direct links to company Web sites and e-mail addresses. Free-form text searches help you evaluate and com-

pare solutions. And you can search by an incredibly wide variety of parameters including industry, technology, language, solution type and country of availability.

Why wait another minute to find the solution you need? Just log on to the address below, select a language and start searching the Global Solutions Directory. It's a quick click from there to the perfect software for your business.



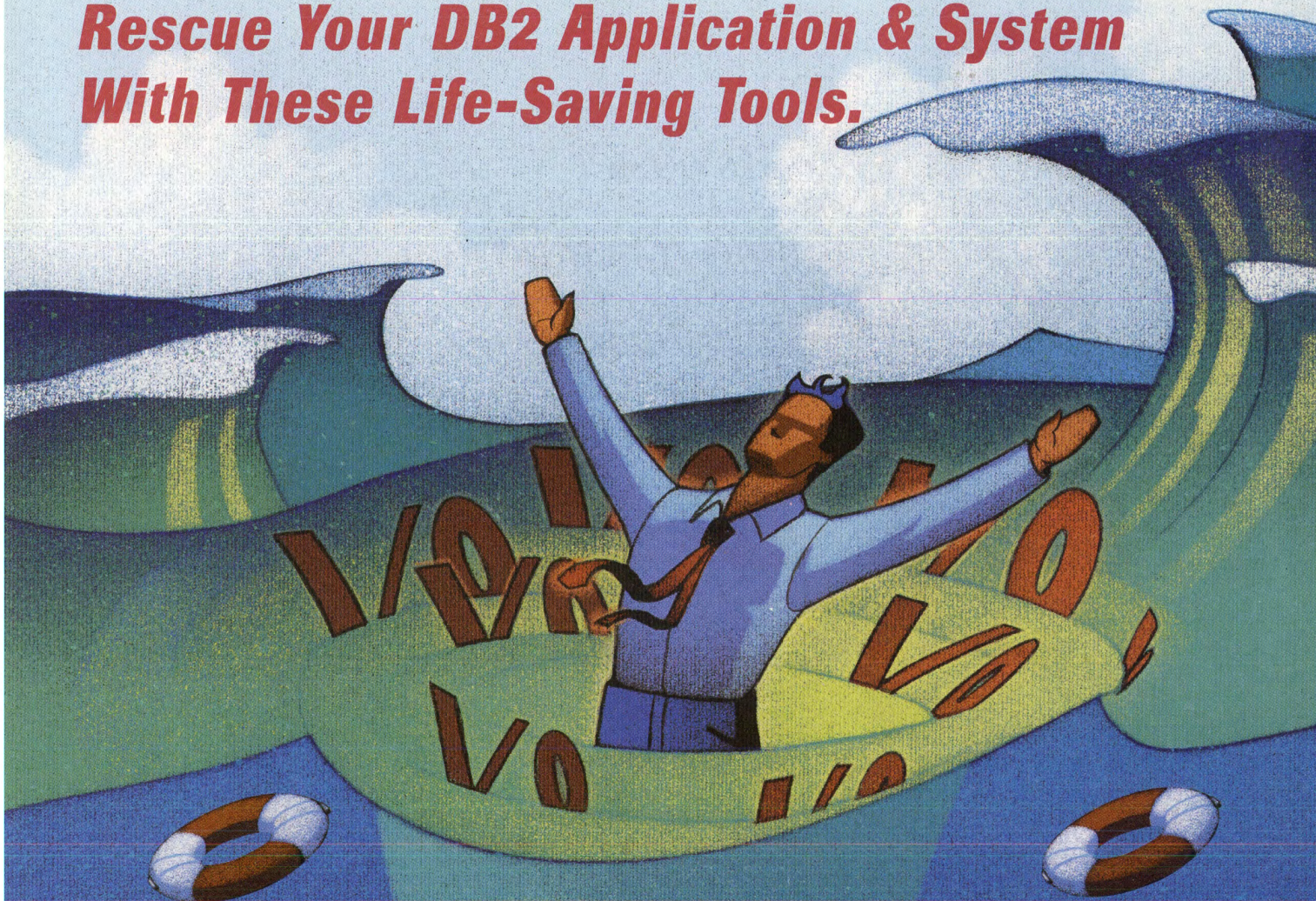
**If you have any questions, call 1 800 627-8363 in the U.S. & Canada; + 1 770 835-9902 worldwide.**

[www.software.ibm.com/solutions/isv](http://www.software.ibm.com/solutions/isv)

**IBM**



# Rescue Your DB2 Application & System With These Life-Saving Tools.



## **Buffer Pool Tool<sup>®</sup>**

Feeling overwhelmed by poor DB2 performance? A system or application not tuned for maximum performance can drown in avoidable IOs, slow application response times, and unnecessary costs due to poor memory utilization. Save your sanity and your DB2 system with the Buffer Pool Tool.

Instead of just treading water to get by everyday, take action by obtaining statistical detail about your DB2 buffer pool in easy to understand graphical formats. The ability to *predict* the effect of tuning changes to pool sizes and object placement, allows evaluation without impacting the operating DB2 system. A cost-saving bonus.

## **DASD/Xpert<sup>™</sup>**

Feeling frustrated by baffling DB2 problems? In spite of the promise of new hardware and software technologies, you are still over your head with basic problems. However, the demanding process of detecting and analyzing performance problems for DB2 objects is now solved with DASD/Xpert.

At the touch of a button, powerful graphics, and instant analysis capabilities are deployed to reduce both transaction response times and batch window requirements due to DASD contention. By quickly detecting and highlighting current problems caused by poor data set placement you gain improved performance at reduced processing costs.



# DB2<sup>®</sup> magazine

QUARTER 1, 2001

VOLUME 6 • NUMBER 1

## features

### Powering Up: DB2 Utilities

**RICHARD YEVICH**

DB2 UDB OS/390 version 7 utilities are challenging the best offerings from third-party vendors in terms of performance, price, and function.

14

### The Database Security Blanket

**PAUL ZIKOPOULOS**

Hackers and crackers and spies, oh my! Understanding database security is the first step to companywide peace of mind.

20

### Dial M for Business

**MARKO TEERIOJA**

How mobile and Internet technologies helped a bricks-and-mortar company escape the paper past for the digital future.

28

### Giving Business a Sporting Chance

**DAN GIBSON**

The business intelligence functions that power Olympic coverage can work for your company.

38

### Top 10 Performance Tips

**SCOTT HAYES**

A database first-aid plan that will help you find and repair almost any performance problem.

46

## columns

### 8 • Data Miner

*By Stefanos Manganaris*

### 12 • Content Management

*By Rita Lochner & Karen Ross*

### 52 • Web Developer

*By Jeff Norton*

### 56 • DB2 DBA

*By Robert Catterall*

### 60 • Programmers Only

*By Bonnie Baker*

### 64 • View from IDUG

*By Brenda Castiel*

## departments

### 4 • Editor's Note

### 6 • Product Update

## online extra feature

### • DB2 Knows Baseball

*By Dan Gibson*

DB2's object/relational capabilities at work in the Olympic world.

*For more information on the products mentioned in this issue, point your Web browser to:*

[ibm.com/software/data](http://ibm.com/software/data)

IBM Data Management software

[ibm.com/software/data/db2](http://ibm.com/software/data/db2)

The DB2 family

[ibm.com/software/data/db2/os390](http://ibm.com/software/data/db2/os390)

DB2 for OS/390

[ibm.com/software/data/db2imstools](http://ibm.com/software/data/db2imstools)

IBM Data Management Tools

[ibm.com/software/bi](http://ibm.com/software/bi)

IBM Business Intelligence software



## Back to Business

**I**N THE FIRST ISSUE OF A NEW year, it's always tempting to prognosticate about what the next months might bring. But I'm hesitant to do so: The year 2000 held enough surprises to make anyone think twice about trying to tell the future.

Most recently, political pundits in the United States took a drubbing during an electoral fracas in which the twists and turns were impossible to predict. And at the beginning of the year, despite (or because of, depending on who you ask) dire predictions, the Y2K bug barely flapped a wing.

Then there's the so-called "new economy." The Internet bubble seemed to spring a leak. Even high-profile Wall Street faves folded or ditched their B2C models in favor of the next new thing, which seemed for a time to be B2B e-commerce. What happens next? I'm not about to guess.

Despite the turbulence that marked the past year, there are still folks willing to put the past in perspective and take a stab at the future. In fact, IBM Chairman and CEO Louis V. Gerstner Jr. is one.

In his keynote address at the e-Business Conference and Expo in New York City on Dec. 12, Gerstner summed up the e-business craze, which he says parallels the hype that surrounded the introduction of other new technologies:

"First, there's a period of wild enthusiasm — intoxicating optimism that the new technology is going to rewrite the laws of competition and economics ... to create whole new wealth, wipe out old industries, create new ones.

"Predictably that fever passes. ... People open their eyes. They don't see new industries. They don't see radically new

business life-forms. And they say: Bah, forget it.

"That passes, too, and the world finally gets down to the important work of taking the technology and integrating it into the structure and fabric of society and business. ... Today, e-business is just business — real business. And real business is serious work."

If 2001 turns out to be less about enthusiasm and more about work, then it's fitting that *DB2 Magazine's* cover story focuses on DB2 utilities. They may not be glamorous, but where would you be without them? Certainly more glamorous applications, "e" or otherwise, rely on a database with top-notch performance and resource utilization. And with a job market that's still tight, anything that makes the DBA's life easier is a bonus. You can read about how the new batch of utilities for DB2 for OS/390 in the article by Richard Yevich on page 14.

Other articles in this issue share the "no-nonsense" vein. No matter what the new, new economy looks like, data is sure to remain such a vital ingredient that protecting it is more important than ever. Paul Zikopoulos explains how to do so in DB2 on page 20, and attorney Jeffrey Matsuura provides the legal perspective on page 24.

Of course, we wouldn't want to focus so closely on the details that we miss the larger picture. These articles speak to a data management foundation that can integrate with exciting, innovative solutions such as the mobile and Internet solution for supply chain management described on page 28.

The party may be over, but we've survived the morning after. Now let's get on with business. ●

EDITOR **Kim Moutsos**

650-513-4319 or [kmoutsos@cmp.com](mailto:kmoutsos@cmp.com)

ASSISTANT EDITOR **Olga Zundel**

650-513-4292 or [ozundel@cmp.com](mailto:ozundel@cmp.com)

EXECUTIVE MANAGING EDITOR **Annalisa Chamberlain**

DESIGN DIRECTOR **Jim Shinnick**

DESIGNER **Earl Otsuka**

SENIOR PRODUCTION EDITOR **Katy German**

PRODUCTION EDITOR **Shannon McArdle**

EDITORIAL BOARD OF DIRECTORS

**Doreen Coyne, Don Haderle, Jeff Jones, Jim Kelly, Tom Kendra, Charles Lickel, Janet Perna, Pat Selinger, David Stodder**

CONTRIBUTING WRITERS

**Bonnie Baker, Brenda Castiel, Robert Catterall, Dan Gibson, Scott Hayes, Rita Lochner, Stefanos Manganaris, Jeff Norton, Karen Ross, Marko Teerijoja, Richard Yevich, Paul Zikopoulos**

EDITORIAL DIRECTOR **David Stodder**

PUBLISHER **BJ Ghiglione**

GROUP PUBLISHER **Veronica Costanza**

ADVERTISING SALES **Angela Duarte**

415-389-9632 or [aduarte@cmp.com](mailto:aduarte@cmp.com)

SALES MANAGER, ANCILLARY PRODUCTS **Gae Reck**

650-513-4245 or [greck@cmp.com](mailto:greck@cmp.com)

ADVERTISING COORDINATOR **Amy Santana**

650-513-4216 or [asantana@cmp.com](mailto:asantana@cmp.com)

ASSISTANT CIRCULATION DIRECTOR **Evelyn Adenau**

650-513-4431 or [eadenau@cmp.com](mailto:eadenau@cmp.com)



PRESIDENT AND CEO **Gary Marshall**

CORPORATE PRESIDENT AND COO **John Russell**

CFO **John Day**

GROUP PRESIDENT, BUSINESS TECHNOLOGY GROUP

**Adam K. Marder**

GROUP PRESIDENT, SPECIALIZED TECHNOLOGIES GROUP

**Regina Starr Ridley**

GROUP PRESIDENT, CHANNEL GROUP **Pam Watkins**

GROUP PRESIDENT, ELECTRONICS GROUP **Steve Weitzner**

SENIOR VICE PRESIDENT, HUMAN RESOURCES **Leah Landro**

SENIOR VICE PRESIDENT, GLOBAL SALES AND MARKETING

**Bill Howard**

SENIOR VICE PRESIDENT, BUSINESS DEVELOPMENT

**Vittoria Borazio**

GENERAL COUNSEL **Sandra L. Grayson**

SENIOR VICE PRESIDENT, SOFTWARE STRATEGIES GROUP

**Peter Hutchinson**

Back issues may be purchased for \$9.00 per copy (includes shipping and handling). For issue availability, call 800-444-4881 in the U.S. and Canada. All other countries call (785) 841-1631. Send email to [orders@cmp.com](mailto:orders@cmp.com). Back issue orders must be prepaid. Please send payment to: DB2 Magazine, 1601 W. 23rd St., Suite 200, Lawrence, KS 66046-2700. For change of address, write DB2, 2800 Campus Drive, San Mateo, CA 94403.

DB2 is a registered trademark of the International Business Machines Corporation and is used by CMP Media under license.

All material published in *DB2 Magazine* copyright © 2000, CMP Media ALL RIGHTS RESERVED. Reproduction of material appearing in *DB2 Magazine* is forbidden without permission.

PRINTED IN THE USA





e-business



It takes a powerful  
**database**  
to recognize your  
customers  
no matter how they  
make contact.  
That's the software  
**IBM** makes.



**DB2® with Siebel eBusiness Applications** delivers superior response across all customer touch points, from partners to the Web to sales support. Which may be why Siebel Systems now recommends DB2 Universal Database™. Visit us at [www.ibm.com/software/data/siebel](http://www.ibm.com/software/data/siebel)

**Software is the soul of e-business.™**

IBM, the e-business logo, DB2, DB2 Universal Database and Software is the soul of e-business. are trademarks of International Business Machines Corporation in the United States and/or other countries. All other products or company names are or may be trademarks or registered trademarks of their respective companies. © 2000 IBM Corp. All rights reserved.



## New IBM Support For Linux

IBM announced the general availability of new Linux support, including:

- **DB2 Enterprise-Extended Edition (EEE)**, a solution for multiterabyte databases supporting multiple processors through server clustering

- **DB2 Enterprise Edition (EE)** for the IBM zSeries and S/390 platforms

- **DB2 Connect**, which provides access to DB2 data on the mainframe from applications using DB2 OS/390 and DB2 for Linux on the same server

- **WebSphere Application Server, Advanced Edition** version 3.5, which supports high-performance, scalable Web sites, for zSeries, S/390, and Intel-based servers

- **WebSphere Application Server, Standard Edition**, Version 3.5 for Linux running on Intel servers.

Linux distributors Red Hat, SuSE, and TurboLinux support the new offerings and bundle DB2 with their Linux solutions. In addition, the three distributors use DB2 and WebSphere as their standard infrastructure software.

For more information, go to [www.ibm.com/software/data/db2/linux](http://www.ibm.com/software/data/db2/linux) or read "Serving Up Linux" in the Quarter 4, 2000: Winter *DB2 Magazine*, available online at [www.db2mag.com](http://www.db2mag.com).

## J-Store Integrates DB2

J-Commerce Inc. will integrate IBM middleware (including DB2 UDB, WebSphere Application Server, and MQSeries) with its Web-enabled Java point-of-sale appli-

cation, J-Store. J-Store lets users remotely manage retail operations. This integration will let J-Store's customers integrate their point-of-sale over the Web with business-critical systems such as inventory, order processing, and shipping.

J-Store is object-based, modifiable, extendable, and can be integrated with business-critical systems, including ERP, inventory management, financial management, and supply chain management.

For more information, see [www.j-commerce.com](http://www.j-commerce.com).

## SuSE – DB2 Database

SuSE announced an enterprise database solution for Enterprise Applications. SuSE Linux Database Server combines SuSE Linux 7.0 and DB2 UDB for a cross-platform solution that will support data types such as images, videos, audio files, and text documents and allow easy integration with Internet applications.

SuSE Linux Database Server includes:

- SuSE Linux Server Edition
- DB2 UDB Workgroup Edition
- Installation and configuration support

For more information, see [www.suse.com](http://www.suse.com).

## DB2 Everyplace 7.1.1

IBM announced the general availability of DB2 Everyplace version 7.1.1, a compact relational database that supports the Palm platform and four new platforms, including Microsoft Windows CE/Pocket PC, EPOC, Neutrino, and embedded Linux. Version 7.1.1 also supports Bluetooth and IR standards, enabling true wireless synchronization.

The four new platforms run on devices such as cellular phones, handheld

PCs, and personal digital assistant (PDA) devices. DB2 Everyplace will also enable enterprises to build and manage database-enabled applications on stand-alone mobile devices in addition to storing and synchronizing with enterprise data.

DB2 Everyplace Enterprise Edition version 7.1 includes a Sync Server, for data synchronization with other members of the DB2 family, and national language and Double Byte Character support. Supported languages include simplified and traditional Chinese, English, French, German, Italian, Japanese, Korean, and Spanish.

For more information about DB2 Everyplace, go to [www.ibm.com/software/data/db2/everyplace](http://www.ibm.com/software/data/db2/everyplace).

## Alysis WorkOut

Alysis Technologies announced that it will bundle WorkOut, its electronic billing software, with the IBM e-business solutions DB2 UDB and WebSphere Application Server.

WorkOut aims to effectively manage Internet billing without compromising service or security.

For more information, go to [www.alysis.com](http://www.alysis.com).

## Content Exchange

QuantumStream announced a new content exchange infrastructure built on IBM technology, including RS/6000 Unix servers on AIX, WebSphere application server, and DB2 UDB.

An integration infrastructure for universal content placement, exchange, and insertion, quantumStream's solution will help content originators, aggregators,



distributors, and consumers interact in real time.

The initial rollout of quantumStream infrastructure will support companies interested in Web and wireless applications. Future implementations will support emerging audio, broadband, and interactive television applications.

For more information, see [www.quantumstream.com](http://www.quantumstream.com).

## Quest Central for DB2

Quest Software Inc. announced Quest Central, an integrated suite of database management solutions designed to extend the performance of DB2.

Quest Central helps manage the DB2 environment by ensuring availability, improving overall response time and performance, increasing DBA productivity, and easing administration. The product line includes four core components: performance monitoring, SQL tuning, space management, and database administration.

For more information, go to [www.quest.com](http://www.quest.com).

## B2B E-Marketplaces

CommerceQuest has teamed with IBM to deliver five e-marketplaces to customers in a broad range of industries by the end of 2000. The customers — ICG Commerce, AviEx, E-Chemicals, Plastics-Net and UIP — are using enableNet (CommerceQuest's online managed trading service) to integrate their trading communities over the Internet.

CommerceQuest's enableNet service helps customers achieve reduced supply chain costs and increased speed. The technology platform consists of IBM S/390 servers, DB2 UDB, MQSeries middleware, and Domino software. CommerceQuest and IBM plan to deliver 60 more such e-marketplaces by the end of 2000.

For more information, go to [www.commercequest.com](http://www.commercequest.com).

## Open Market to Support DB2

Open Market Inc. a provider of integrated enterprise e-business solutions, announced that its content management products will offer native support for IBM's WebSphere Application Server and DB2 UDB.

Open Market's Content Server and associated applications, Content Centre,

Personalization Centre, Catalog Centre, and Marketing Studio, already offer native support for a variety of J2EE compliant application servers.

Open Market and IBM are working together on solutions for the financial, new media, and manufacturing markets.

For more information, go to [www.openmarket.com](http://www.openmarket.com).

## IBM-ESRI Alliance

IBM and Environmental Systems Research Institute Inc. (ESRI) announced a global strategic alliance designed to serve the emerging marketplace for geographic information system (GIS) technology, which is expected to reach \$60 billion by 2005.

GIS technology combines data visualization, spatial analysis, and mapping with database management system integration to help users understand relationships, enhance decision making, and solve problems.

Both companies will jointly market, sell, and provide integration services for ESRI Enterprise Solutions for local government and telecommunication customers. The GIS Enterprise Solutions will include ESRI's GIS software technology combined with IBM software, hardware, and services, including DB2 UDB with Spatial Extender and WebSphere operating on IBM eServer, pSeries, and xSeries.

For more information, go to [www.esri.com](http://www.esri.com).

## Linux for Small Business

IBM announced the Small Business Suite for Linux, an integrated software suite that delivers tools for messaging and collaboration, productivity, Web site creation and design, and data management. The suite gives small businesses necessary e-business elements, including database, email, and Web application server software.

Small Business Suite for Linux includes:

- Lotus Domino Release 5.04, a messaging and collaboration solution that offers a simple administrator interface for desktop and mobile email, Web access, group scheduling, bulletin boards/news-groups, workflow, and database access.

- WebSphere Application Server, Standard Edition version 3, which delivers an open Web application runtime environment by allowing Java servlets to run on top of an HTTP server and makes it easy

to build security-enhanced Web sites.

- DB2 UDB version 7, a relational database for a wide variety of data types, including text, data, voice, image, and binary objects. Data from third-party applications that support DB2 can be stored, retrieved, and reported on from DB2, routed or shared in a teamroom with Lotus Domino, or made accessible to customers or suppliers over the Internet with WebSphere Application Server.

- Web design and productivity tools, including WebSphere Homepage Builder, IBM Suites Installer, WebSphere Studio and Domino Designer.

The Small Business Suite for Linux is supported on Linux distributions from Caldera, Red Hat, SuSE and TurboLinux.

For more information, go to [www.ibm.com/linux/r/sbs/2](http://www.ibm.com/linux/r/sbs/2).

## February

### DCI's eCustomer Conference & Exposition

February 13-15  
Chicago  
[www.dci.com](http://www.dci.com)

### DCI's Data Warehousing Conference & Exposition

February 13-15  
Chicago  
[www.dci.com](http://www.dci.com)

### TDWI World Conference; Winter 2001

February 25 - March 2  
Palm Springs  
[www.dw-institute.com/conferences/palmsprings2001](http://www.dw-institute.com/conferences/palmsprings2001)

## March

### META Group and DCI's Enterprise Architectures Conference

March 27-29  
Boston  
[www.dci.com](http://www.dci.com)

## May

### IDUG 2001; North America

May 13-17  
Orlando, Fla.  
[www.idug.org](http://www.idug.org)



# Reading Between The Lines

*Refining your text mining techniques.*

**B**Y SOME ESTIMATES, THE WEB now contains close to 2.1 billion pages, a number that roughly doubles every 12 months. Worldwide, more than 225 million people send and receive email messages. AOL alone conveys 760 million messages a day. The Internet's phenomenal growth and the widespread use of computers to store, process, and communicate text have created the need for tools that help individuals and businesses find the information they need in the most effective and efficient manner possible.

Such documents as Web pages, news postings, and email messages contain information in unstructured free-form text or streams of characters that make up words and sentences that conform to the syntax and grammar rules of a specific computing language. In sharp contrast, traditional databases are rigidly structured collections of tables that contain records representing specific instances of entities, relationships between entities, and columns representing the various records' attributes. For example, a table of customers could have one record per customer and the following columns: customer ID, customer name, and customer address.

Data mining extracts previously unknown, comprehensible, and ultimately actionable information from structured databases. Using various analytical techniques that fall under the umbrella of

data mining, you can perform knowledge discovery tasks, including:

- Discovering patterns
- Revealing hidden relationships
- Detecting unusual behaviors
- Organizing entities into similar groups
- Inducing models that can explain the underlying rules that govern a process
- Formulating models that can accurately predict a process.

Such capabilities are often critical components in solutions to business problems across industries, ranging from demand forecasting in supply chain management to targeted marketing in e-commerce, to fraud and abuse detection in processing medical claims. I've focused on several of these analytic techniques in previous issues (see, for example, the Data Miner column in the Quarter 3, 2000: Fall issue, available online at [www.db2mag.com](http://www.db2mag.com)).

But what about knowledge discovery in unstructured data collections? Are there analytical techniques for extracting previously unknown actionable information from unstructured data collections, which might include text documents, images, video, and audio? What business problems could you solve using knowledge discovery on unstructured data?

## MINING UNSTRUCTURED DATA

You can perform knowledge discovery on unstructured data with the help of tools such as IBM Intelligent Miner (IM)

for Text and IBM ViaVoice for continuous speech audio. ViaVoice facilitates knowledge discovery with its voice recognition capabilities. Other technologies, such as handwriting recognition, content-based querying for images and time series (such as IBM's Query by Image Content), image classification, and video indexing, are in various stages of readiness for widespread use.

There are three fundamental text mining operations: clustering, categorization, and information retrieval.

**Clustering.** You can use clustering techniques to impose an organizational structure on a collection of text documents by clustering together groups that are related or similar based on their content. The clustering induces the number and type of thematic categories from the data; the organizational structure is data-driven and you don't have to prespecify document categories. Many clustering techniques produce flat organizational structures in which document groups are disjoint, and each document is assigned to a single thematic category. (See Figure 1, page 10.) Other clustering techniques produce hierarchical structures in which groups may be decomposed recursively into subgroups corresponding to refined or lower-level thematic categories. In any case, clustering turns unstructured document collections into thematically organized groups that provide a summary view of the documents in that group and



facilitate effective and efficient navigation.

**Categorization.** Some document collections already have an organizational structure, either imposed by a document clustering tool or crafted manually by identifying thematic categories and assigning documents to them. In such situations, categorization techniques can uncover the principles governing the assignment of documents to categories. By analyzing the content of the documents and their assigned categories, categorization techniques produce classification models that detail the discriminating features of the various categories. Such models explain the key differences between categories and can automatically classify new documents and incorporate them into the collection and the existing structure.

**Information retrieval.** In addition to organizing document collections and categorizing documents, users must be able to retrieve pertinent information from the collection. With structured data, users perform a database query (which is not usually considered a data mining operation because the results of a query are explicitly stored in the database, and no new information is produced.) With unstructured data, querying is considerably more difficult because effective information retrieval requires analysis of document content — in other words, text mining. Simply matching the text of a query to documents is not very useful. You might want to retrieve relevant documents even when they don't contain any of the text found in the query. In addition, the fact that some of the text in the query is present in a document does not always mean that the document is relevant. The key characteristics of a good information retrieval engine are query flexibility, effective retrieval, and computational efficiency.

A data miner will immediately recognize that clustering text is similar to clustering data points, while text categorization is similar to inducing classification models. Indeed, the basic ideas in the algorithms used to perform these operations are the same, but the devil is in the details. The free-form nature of text makes handling it difficult. You have to analyze the semantics of words and sentences in the context they appear in order to derive *features* before you can apply any knowledge discovery techniques. Features are significant items in text, such as names and technical terms. Which features are significant in a document may depend on the content of other documents in the collection and

on the text-mining task at hand. Thus, text-mining algorithms must incorporate sophisticated text analysis tools.


IBM IM for Text offers a rich set of text analysis tools, including a feature extraction component that can automatically derive a vocabulary that captures key terms and concepts appropriate for the document collection analyzed. Elements of the vocabulary can be multiword terms; names of people, organizations, and places; abbreviations; and key numeric figures,

such as currency amounts and dates. IM uses algorithms that are sophisticated enough to recognize “credit facility,” “credit line,” “Credit Lyonnais,” and “Credit Suisse” as four separate concepts, while recognizing “Bill Clinton” and “President Clinton” as the same entity, distinct from “Clinton, N.J.” A similar tool extracts significant sentences from a document to create a summarized version. Figures 2 and 3 (see page 10) show examples of feature extraction.



Nerds  
Wanted!™

JustDB2Jobs.com

brought to you by  
 justtechjobs.com

DB2 is a registered  
trademark of IBM corp.



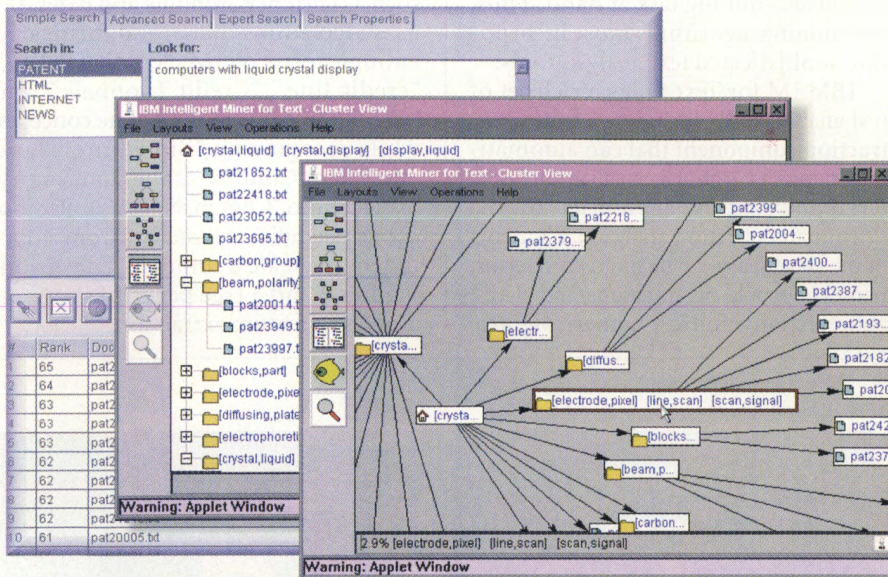


Figure 1: Hierarchical document clustering.

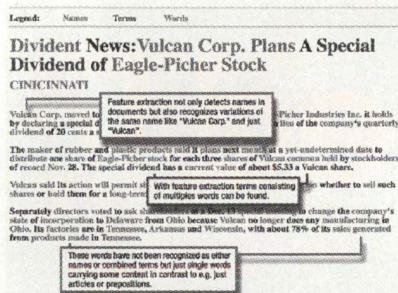


Figure 2: Sample feature extraction.

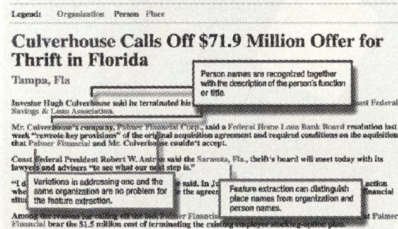


Figure 3: Sample feature extraction.

Another IM component can automatically identify the language of a document — an important feature in text mining. You can implement this capability using IM for Text categorization tools that also make the product trainable and extensible to other languages.

The feature extraction and language identification components in IM for Text are available for developing custom text-mining applications. IM for Text's sophisticated search engine also uses them for information retrieval. The search engine includes an indexing tool that performs in-depth linguistic analysis of the documents in the collection to prepare a data structure that facilitates fast and

effective information retrieval. Although most of the indexing is performed offline, IM for Text is capable of updating the index on the fly, while it's processing queries. Client applications typically submit queries to the search engine's server. Using the index and other optional resources, such as dictionaries and thesaurus files, the server can efficiently search very large collections of documents written in any of 16 different languages, including double-byte character-set languages (such as Japanese, Chinese, and Korean), and stored in various file formats.

Figure 4 shows a typical client/server configuration. The query language in this example allows the use of free text; Boolean expressions with conjunction, disjunction, and exclusion of search terms; and a hybrid combination of the two. The search engine supports several paradigms, including precise-term searching, probabilistic retrieval, phonetic searches, and fuzzy searching.

Document collections are often decentralized and are constantly changing (the Internet is a prime example). Therefore, you need tools that automatically explore the collection to track the changes and identify all documents in it. IM for Text includes a Web crawler toolkit, which you can use to develop customized agents that monitor collections of Web pages on the Internet or an intranet.

### EMAIL AND BEYOND

Let's now consider a practical example of text mining: coping with email. If you are like most people, you receive more email messages in a day than hard-copy

correspondence. You read some, delete many, and file the rest. You probably have devised some rules for filing: you may file some messages by sender, others by topic. A text-mining application for email could employ categorization tools to automatically determine which incoming messages should be kept for reading, immediately deleted, or filed unread. Messages to be filed could also be categorized automatically and assigned to the appropriate folder using a second categorization engine. The categorization algorithm can discover the rules that govern your particular message routing and filing scheme automatically, by analyzing your actions as you process your email. Information retrieval tools can enhance the primitive string search capabilities of most email reading tools. Linguistic analysis and feature extraction in the collection of accumulated email would allow sophisticated text queries to identify pertinent email messages based on their content. (For example, that kind of analysis would let you find all archived messages that are related in content to the one you are reading.) You can use clustering to further organize your email notes thematically or to identify predominant themes within a category of filed notes.

Clearly, the potential of text mining goes beyond email. News, patents, and other intellectual capital, and knowledge management can also benefit from such capabilities. Analysts estimate that 80 percent of the information an enterprise possesses is in unstructured form, vs. 20 percent structured. Faster, better access to this information can certainly have a positive impact on a business. Internally, text-mining solutions can help store, manage, retrieve, and deliver the intellectual capital of business communications. Examining external sources, text mining can help track developments in an industry and analyze the competition by pulling and analyzing relevant news, reports, and patents. The IBM Business Intelligence organization has a worldwide consulting practice that specializes in text mining applications, with experience helping customers in various industries develop solutions that leverage their most important asset: information.

### GLOBAL USES FOR TEXT MINING

Beyond facilitating access to information, text-mining solutions can help improve operations in profound ways. In the context of customer relationship management, text mining can help glean use-



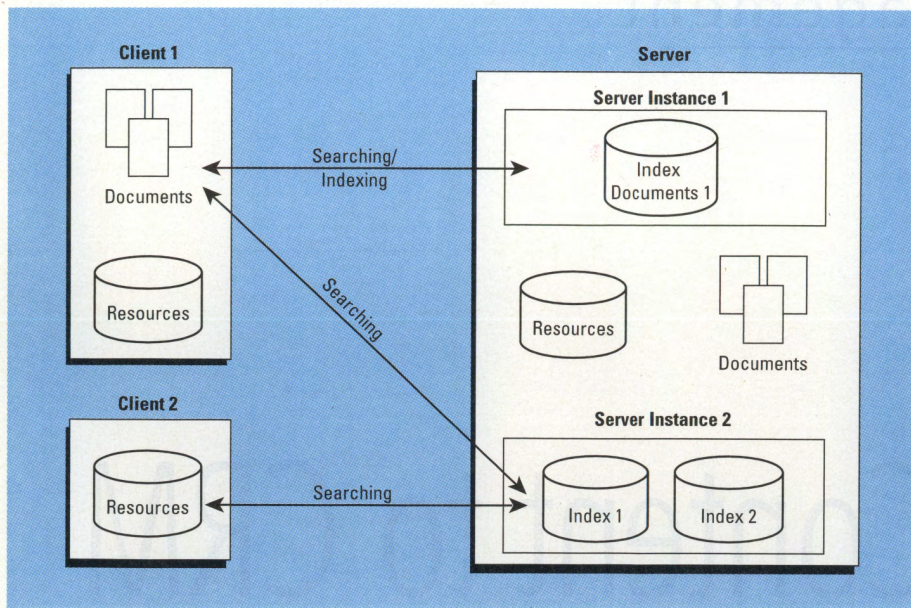


Figure 4: An information retrieval solution with two instances of the search engine on the server and two client applications, one of which has the capability to update the indexes for newly available documents.

ful information about customers. Customer communications encompass more than placing orders and processing returns. A lot of useful information about a customer and the business arrives in the form of praises, complaints, desires,

and suggestions. Feature extraction can also be used to automatically assess customers' satisfaction with their relationship to the business. Data mining techniques are used to profile customers or to assess their propensity to purchase or

their risk for attrition (among many other things) based on structured data, such as demographics, psychographics, and the customer's past transactions with the company. Powerful solutions combine data and text mining capabilities and expand these analyses to factor in customer attributes inferred from nontransactional communications. ●

**Stefanos Manganaris** is a senior consultant in the Knowledge Discovery Consulting group at IBM. He is responsible for data mining solutions in consulting engagements and research in knowledge discovery with applications to business problems across industries. You can reach him by email at stefanos@us.ibm.com.

#### RESOURCES:

Read more about :

#### IBM Intelligent Miner for Text

[www.ibm.com/software/data/iminer/fortext](http://www.ibm.com/software/data/iminer/fortext)

#### IBM ViaVoice

[www.ibm.com/software/speech](http://www.ibm.com/software/speech)

#### IBM Business Intelligence organization

[www.ibm.com/software/data/busn-intel](http://www.ibm.com/software/data/busn-intel)

## Web Links Made Simple



**OLE DB, ODBC, JDBC without complexity.**

Small footprint  
Easy to configure  
Simple to support

**HIT**SOFTWARE  
Open Up Your Data

[www.hitsw.com](http://www.hitsw.com) • 1-408-345-4001 • [info@hitsw.com](mailto:info@hitsw.com)

## The Database Monitors with Zero Impact Win!

eData Usage Analyzer  
Zero Impact Sql Monitor  
Zero Impact Top N Sql Analyzer  
Zero Impact Service Level Monitor



Scales to 10,000+ users with no impact on the server, network or end-users!

- **Data usage** in SELECT, WHERE, ORDER BY, GROUP BY clauses by **user** - aggregated over time with user service levels.
- Identifies production SQL that **executes most frequently** or uses the **most resources** with end-user response time service levels.
- Monitors the cause of **80% of database poor performance** incidents, i.e. SQL statements. 24x7 captures production SQL performance with text. Pinpoint or reconstruct SQL 15+ ways.
- **Service level monitor** pro-actively identifies SQL performance issues prior to **end-user complaints**. Measures average end user and database server response time, plus network time in real-time.
- Requires **no intrusive** database server agents, audits, traces, middleware, database connections or shared memory reads.
- **Free copy** for consultants to build a performance tuning practice.

**Sql Power Tools**

[www.sqlpower.com](http://www.sqlpower.com) 702.990.3209



Rita Lochner & Karen Ross

## Adding Content to CRM

*Giving customer service representatives access to all customer information requires a solution that can handle unstructured data.*

**W**HEN CUSTOMERS CONTACT a customer service center, they are often anxious or agitated. It is a moment of truth between the company and its customer. The company can either sustain the customer's loyalty or lose it.

What will make the difference? Prompt and professional treatment will help strengthen a customer's loyalty. But consumers today have high expectations for customer service. They want a company to recognize them as a customer in every channel they use, whether they purchase from a bricks-and-mortar location, a toll-free number, a Web site, or a combination of all three.

That's not an unreasonable demand, and most e-business companies recognize the need for a solution. That's one reason why customer relationship management (CRM) is a core component of most e-businesses. Companies are striving to meet and exceed customer expectations in order to build and sustain long-term customer loyalty. At the same time, they want to reduce costs by avoiding redundant recruitment campaigns. The customer information generated through call-center and online interactions is crucial for both of these goals. But companies often hit a glitch when attempting to store and analyze this kind of information: Much of it arrives in the form of paper and email correspondence. This unstructured data isn't stored in any database. And even the most expensive CRM solutions can fal-

ter when faced with the challenges unstructured data presents.

The content of the email and paper correspondence may hold the key to prompt and accurate resolutions — but only if the content is readily available and integrated with the call center and CRM applications. CRM-enabled content management fosters successful moments of truth for customer service representatives (CSRs) and customers. Ideally, a CSR working with access to the case history, correspondence, and reference information (for example, statements and invoices) could resolve most customer calls during the initial contact.

Introduced in March 2000, IBM's Content Manager products were designed for storing and accessing unstructured data. Content Manager lets you store, index, and retrieve documents in many different formats, such as TIFF files (for document scans or faxes), Word documents, and spreadsheets. OnDemand handles capture, presentment, storage and retrieval of computer output to laser disc (COLD) documents, such as statements and bills. (You can read an overview of Content Manager's features in the recent *DB2 Magazine* article "Controlling the Digital Deluge," by Blaine Lucyk, Quarter 2, 2000: Summer, available online at [www.db2mag.com](http://www.db2mag.com).)

To help companies implement a CRM-enabled content management solution, IBM Global Services (IGS) and IBM business partners often integrate Content

Manager and Content Manager OnDemand with CRM software products. IBM Global Services developed an offering that tightly integrates Content Manager and Content Manager OnDemand with Siebel Call Center — the first to be Siebel 2000 validated.

The IGS offering brings to light a number of features that are important for any CRM-content management solution. We'll walk you through those features, using examples drawn from the Content Manager-Siebel solution.

### INBOUND/OUTBOUND DOCUMENT CAPTURE

A complete CRM-content management solution needs to capture both structured and unstructured customer correspondence, both inbound and outbound. Inbound customer correspondence received via mail, fax, or email can be stored in Content Manager. Outbound customer bills and statements captured as COLD documents for future reference can be stored in Content Manager OnDemand. Outbound correspondence can be stored in Content Manager in the native format of the word processor or document-editing program. In the IGS solution, a single logon connects the CSR to Siebel Call Center, Content Manager, and OnDemand. From Siebel Call Center the CSR has access to mail, fax, and outbound correspondence stored in Content Manager and bills, statements, and other COLD documents stored in OnDemand.



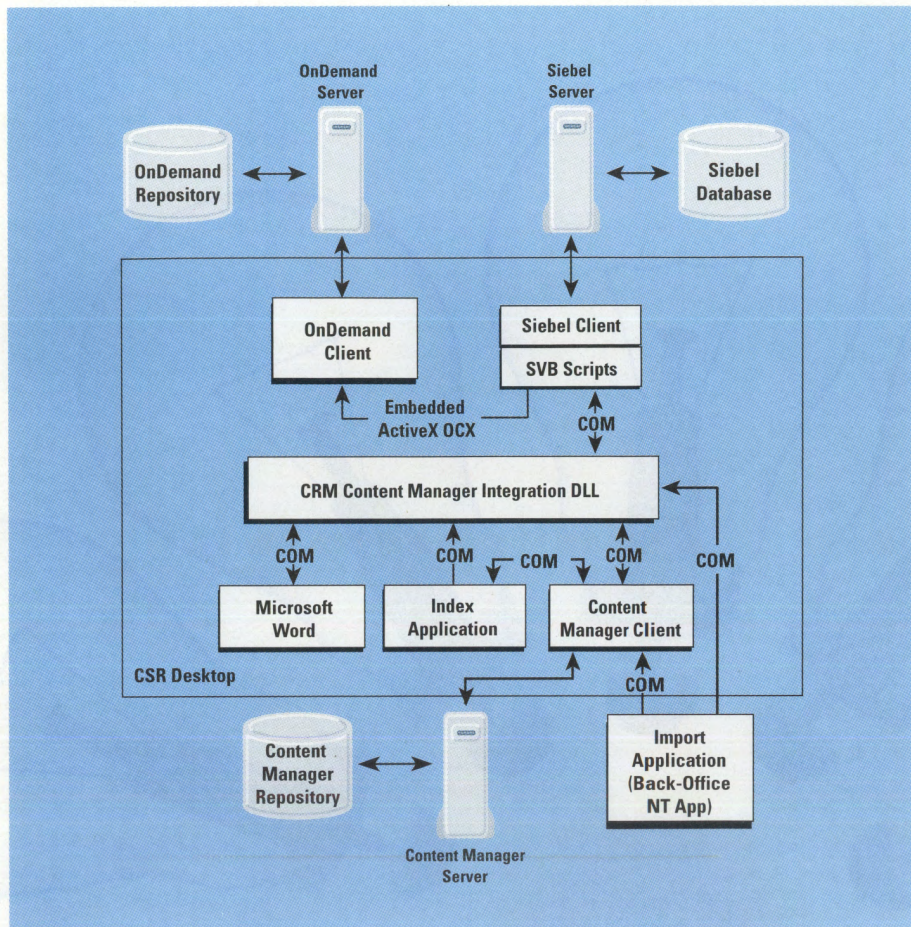


Figure 1: A CRM-enabled content management solution integrated with Siebel Call Center.

Point-and-click access lets the CSR simply locate and view the document referenced in the Siebel application. Easy access to all documents related to a service request, account, or contact helps CSRs respond to customer queries quickly.

In the IGS solution, the OnDemand client document viewer is embedded in the Siebel client. References to OnDemand documents are stored in the Siebel database. The Content Manager client provides a Common Object Model (COM) Automation interface. References to Content Manager documents are maintained in the Siebel database. The document list can be viewed from the Siebel client without connecting to Content Manager. When the user wants to view a document, an API in the CRM Content Manager (CM) Integration DLL is called and connects to Content Manager. The document is then displayed in the Content Manager viewer. Figure 1 shows the architecture of the IGS offering.

### PROCESSING INBOUND CORRESPONDENCE

Linking customers' inbound mail or fax documents to their records makes sure that CSRs have access to all pertinent in-

formation when a customer call comes in. The IGS offering handles this process through an application that lets any user in the organization access inbound mail or fax documents from prioritized queues (in this case, Content Manager Workbaskets) and index them to a Siebel service request. The Index Application runs on the client desktop and can be launched from the Siebel Call Center toolbar. The application interfaces directly with Content Manager via the COM automation interface to the Content Manager client.

### CREATING OUTBOUND CORRESPONDENCE

Correct handling of outbound correspondence is just as important to the customer relationship as correct processing of inbound correspondence. But how often have you gotten mail from a company in which your name is spelled wrong? CSRs are human — they make mistakes. The IGS offering eliminates some opportunity for error by automating much of the process. From a service request, a CSR can select from a list of Microsoft Word templates. The application then launches Word and prefills the template with data such as customer name, ad-

dress, service request number, and CSR name. If the CSR modifies the template, it can be queued for review; otherwise it is queued for a batch print process. The document is then automatically archived in Content Manager, linked to the Siebel service request, and made available to any agent who views the service request.

The IGS offering uses the Microsoft Word COM interface. When the CSR completes the document, a Word macro is executed. The macro calls the CRM CM Integration DLL to archive the document to Content Manager and link it to the service request.

The IGS offering also lets CSRs reassign correspondence from one service request to another. Once reassigned, the document stored in Content Manager is updated to be consistent with the new service request.

### Document Capture Integration.

The IGS offering contains an import application that uses the Content Manager COM automation interface, which is a back-office application that imports documents and index data from a file server as they become available. The documents and index data can come from several sources including:

- A front-end document capture system, such as Kofax Ascent Capture
- A fax server, such as FaxPlus/Open
- A third-party vendor
- Converted legacy documents.

### HAPPY CUSTOMERS, HAPPY EMPLOYEES

No matter whether you create your own content management-CRM solution or use the IGS or other offering, the goal is the same: making sure that employees have all the information they need to meet escalating customer service expectations. Meeting that goal will help you build customer loyalty and may even help you retain your frontline employees. ●

**Rita Lochner** is project executive at IBM E-Business Knowledge and Content Management in Boulder, Colo.

**Karen Ross, Ph.D.**, is CRM market offering manager in IBM Content Management Marketing in St. Simons Island, Ga.

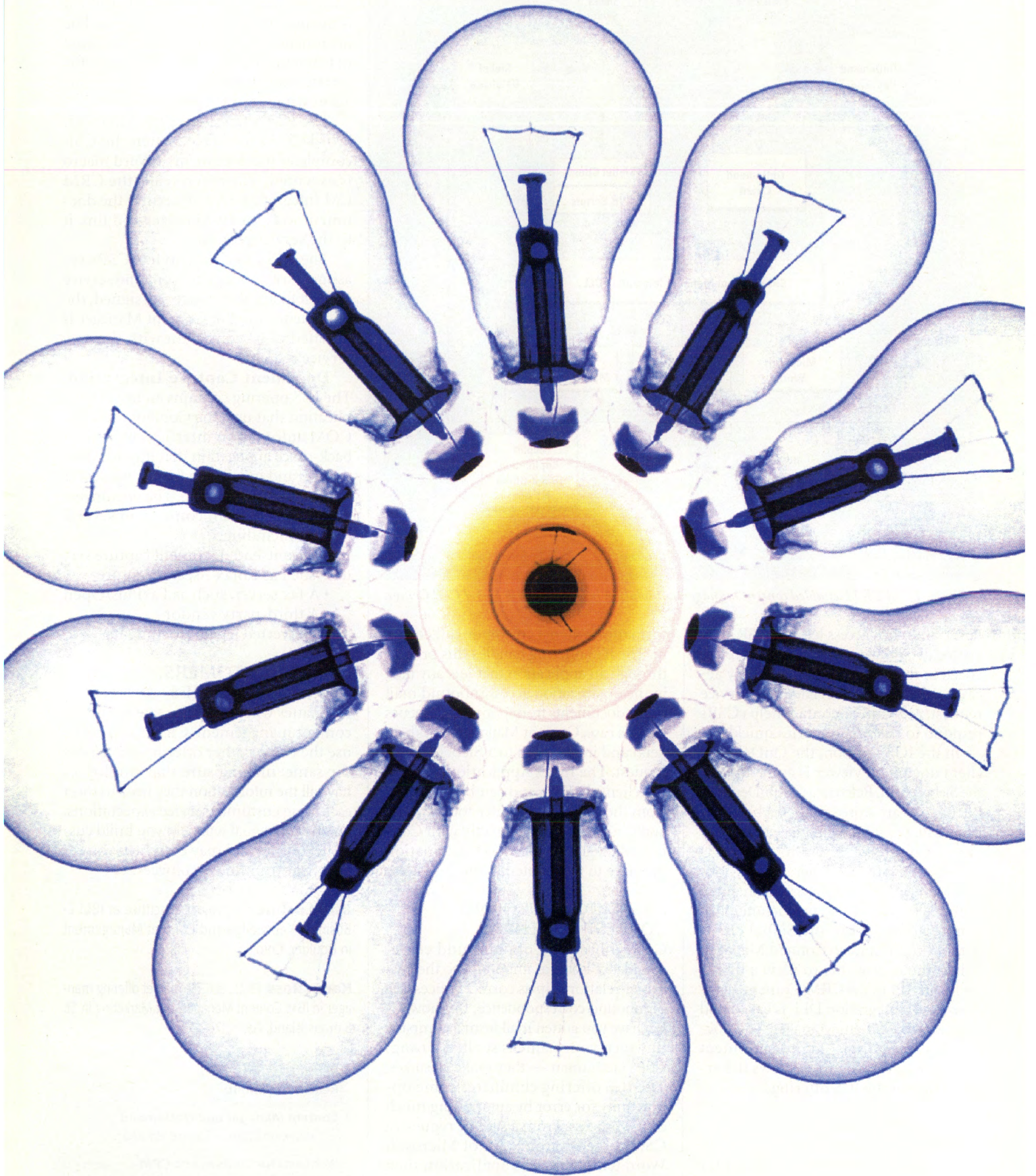
### RESOURCES:

Read more about:

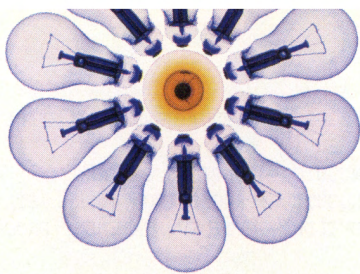
**Content Manager and OnDemand**  
[www.ibm.com/software/data/cm](http://www.ibm.com/software/data/cm)

**Webcasts for Content and CRM**  
[www.ibm.com/software/data/cm/events](http://www.ibm.com/software/data/cm/events)



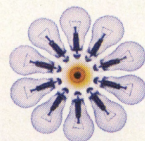






# POWERING UP: DB2 UTILITIES

*Once more functional than fabulous, the utilities available for DB2 UDB OS/390 version 7 are challenging the best offerings from third-party vendors in terms of performance, price, and function.*



In the beginning there was DB2 for OS/390, which included the utilities necessary to support DB2's day-to-day needs.

These utilities weren't particularly fast and they weren't particularly feature rich. Around 1996, IBM began to raise the bar by making enhancements (in both performance and function) that allowed the utilities to start to compete with independent software vendor (ISV) offerings. Then came the DB2 Universal Database (UDB) for OS/390 version 7 announcement. How things have changed.



With that announcement, IBM separated most of the utilities from the base product and imbued them with features and performance on par with the best products from ISVs. These stand-alone utilities compete in terms of price for performance and function, ROI, and ongoing maintenance.

In addition, version 7's core utilities (those that work directly with the base product and are "must haves" for DB2 operation) need to be best of breed. A basic set of core utilities has always been included with DB2 to provide basic services. Due to customer demand, IBM's investment in improving its utilities includes enhancements to these core utilities.

The enhancements to the utilities provide improved functionality. The most significant changes are grouped around wildcarding, dynamic allocation, and templates, which cross the individual utility bounds, and the new UNLOAD utility. Wildcarding, dynamic allocation, and templates provide for better, more automated administration and usage of all the utilities. The new UNLOAD utility now includes many of the features, functions, and improved performance that users have been looking for.

## OBJECT WILDCARDING AND DYNAMIC ALLOCATION

Support for utility wildcarding lets you execute utilities against a list of objects matching a specified pattern that can include wildcard designations. You can execute a utility for a specified list of objects and can INCLUDE and/or EXCLUDE certain objects from these lists by using the new LISTDEF statement. You also have the option to create a utility procedure that would let you run several utilities against several objects with one command. Data set templates that automatically generate data sets based on a defined set of criteria let you use dynamic allocations instead of Job Control Language (JCL) for all datasets required for the utilities.

The Control Center in version 7 supports both utility wildcarding and the ability to execute a utility for a specified list of objects. The enhanced Control Center can also restart utilities from the last committed phase (phase) or the last commit point (current), but only for utilities originally started in the Control Center. The Control Center supports utility IDs and lets you create a utility ID template using a variety of variables (such as USERID and UTILNAME).

Listing 1 shows how you can test your utility and how to use these new features.

```

DSNU000I  DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = TEMP
DSNU050I  DSNUGUTC - OPTIONS PREVIEW
DSNU1000I DSNUGUTC - PROCESSING CONTROL STATEMENTS IN PREVIEW MODE
DSNU1035I DSNUILDR - OPTIONS STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - TEMPLATE KBR DSN
                SY1AA.&DB..&TS..D&JDATE..COPY&IC.&LOCREM.&PRIBAC.
DSNU1035I DSNUJTDR - TEMPLATE STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - LISTDEF KLAAS INCLUDE TABLESPACE DBKLAAS.*
DSNU1035I DSNUILDR - LISTDEF STATEMENT PROCESSED SUCCESSFULLY
DSNU1020I + DSNUILSA - EXPANDING LISTDEF KLAAS
DSNU1021I + DSNUILSA - PROCESSING INCLUDE CLAUSE TABLESPACE DBKLAAS.*
DSNU1022I + DSNUILSA - CLAUSE IDENTIFIES 1 OBJECTS
DSNU1023I + DSNUILSA - LISTDEF KLAAS CONTAINS 1 OBJECTS
DSNU1010I DSNUGPVV - LISTDEF KLAAS EXPANDS TO THE FOLLOWING OBJECTS:
                LISTDEF KLAAS - 00000001 OBJECTS
                INCLUDE TABLESPACE DBKLAAS.TSKLAAS

DSNU050I  DSNUGUTC - COPY LIST KLAAS COPYDDN(KBR,KBR)
DSNU1009I DSNUGPVV - TEMPLATE KBR DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLP
DSNU1007I DSNUGPVV - DATE/TIME VALUES MAY CHANGE BEFORE EXECUTION
DSNU1009I DSNUGPVV - TEMPLATE KBR DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLB
DSNU1007I DSNUGPVV - DATE/TIME VALUES MAY CHANGE BEFORE EXECUTION
DSNU010I  DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=4

```

*Listing 1: Testing a utility using new features.*

```

DSNU000I  DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = TEMP
DSNU050I  DSNUGUTC - TEMPLATE KBR DSN
                SY1AA.&DB..&TS..D&JDATE..COPY&IC.&LOCREM.&PRIBAC.
DSNU1035I DSNUJTDR - TEMPLATE STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - LISTDEF KLAAS INCLUDE TABLESPACE DBKLAAS.*
DSNU1035I DSNUILDR - LISTDEF STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - COPY LIST KLAAS COPYDDN(KBR,KBR)
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR
                DDNAME=SYS00001
                DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLP
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR
                DDNAME=SYS00002
                DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLB
DSNU400I  DSNUBBID - COPY PROCESSED FOR TABLESPACE DBKLAAS.TSKLAAS
                NUMBER OF PAGES=3
                AVERAGE PERCENT FREE SPACE PER PAGE = 36.33
                PERCENT OF CHANGED PAGES = 12.50
                ELAPSED TIME=00:00:02
DSNU428I  DSNUBBID - DB2 IMAGE COPY SUCCESSFUL FOR TABLESPACE DBKLAAS.TSKLAAS
DSNU010I  DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0

```

*Listing 2: Utility test output.*

Listing 2 shows the actual output of the execution.

Another nice feature is the EVENT (ITEMERROR, SKIP) option. If any objects in a list fail (for example due to duplicate dataset name), the list continues with the next object. You can't ignore errors. The utility stops after this utility statement and does not process any other utility statements in the input (see Listing 3).

## UNLOAD

UNLOAD is a new utility that delivers better performance and more flexibility than DSNTIAUL. It supports both Tablespace and Image Copy as the source and allows unloading of multiple partitions in parallel. You can provide field selection for the unloaded data and the ordering and formatting options via SQL-like syntax. UNLOAD also provides for record sampling and charac-



```

DSNU000I DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = TEMP

DSNU050I DSNUGUTC - OPTIONS EVENT(ITEMERROR,SKIP)
DSNU1035I DSNUILDR - OPTIONS STATEMENT PROCESSED SUCCESSFULLY

DSNU050I DSNUGUTC - TEMPLATE KBR DSN
SY1AA.&DB..&TS..D&JDATE..COPY&IC.&LOCREM.&PRIBAC.
DSNU1035I DSNUJTDR - TEMPLATE STATEMENT PROCESSED SUCCESSFULLY

DSNU050I DSNUGUTC - LISTDEF KLAAS INCLUDE TABLESPACE DBKLAAS.*
DSNU1035I DSNUILDR - LISTDEF STATEMENT PROCESSED SUCCESSFULLY

DSNU050I DSNUGUTC - COPY LIST KLAAS COPYDDN(KBR,KBR)
DSNU1015I DSNUGDYN - ERROR ALLOCATING DATA SET
DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLP
CODE=X'47040416'
IKJ56229I DATA SET SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLP NOT ALLOCATED,
DADSM ERROR+
IKJ56229I DUPLICATE DATA SET NAME ON VOLUME
IGD17001I DUPLICATE DATA SET NAME ON VOLUME OS39M1 FOR DATA SET
SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLP
DSNU1038I DSNUGDYN - DATASET ALLOCATED. TEMPLATE=KBR
DDNAME=SYS00002
DSN=SY1AA.DBKLAAS.TSKLAAS2.D2000318.COPYFLP
DSNU1038I DSNUGDYN - DATASET ALLOCATED. TEMPLATE=KBR
DDNAME=SYS00003
DSN=SY1AA.DBKLAAS.TSKLAAS2.D2000318.COPYFLP
DSNU400I DSNUBBID - COPY PROCESSED FOR TABLESPACE DBKLAAS.TSKLAAS2
NUMBER OF PAGES=3
AVERAGE PERCENT FREE SPACE PER PAGE = 36.33
PERCENT OF CHANGED PAGES = 12.50
ELAPSED TIME=00:00:02
DSNU428I DSNUBBID - DB2 IMAGE COPY SUCCESSFUL FOR TABLESPACE
DBKLAAS.TSKLAAS2
DSNU012I DSNUGBAC - UTILITY EXECUTION TERMINATED, HIGHEST RETURN CODE=8

```

*Listing 3: The EVENT (ITEMERROR.SKIP) option.*

ter encoding. Listing 4 (page 18) shows the new UNLOAD utility with dynamic allocation of both sysrec and syspunch datasets (dynalloc sysrec and syspunch). Since a tablespace was unloaded and no tables were specified, the OBID will also be unloaded.

The output from syspunch looks like this:

```

LOAD DATA INDDN SYSREC LOG NO
RESUME YES
EBCDIC CCSID(00000,00000,00000)
INTO TABLE "P390A " ".TBKLAAS "
WHEN(00001:00002 = X'0003')
( "MONTH " POSITION(
00003:00012) CHAR(010)
, "PROFIT " POSITION(
00013:00016) INTEGER
)

```

Listing 5 (page 19) also shows UNLOAD execution. However, this example uses imagecopy and normal dd statements in the JCL (the defaults are sysrec and syspunch).

### ONLINE REORG IMPROVEMENTS

With two new enhancements to online reorg, its use in achieving 24x7 availability in a production environment is

# SIGN UP TODAY!

★★★★ THE NEW ★★★★★

## DB2<sup>®</sup> magazine Email Newsletter

★★★ BRINGS YOU: ★★★

**ONLINE ONLY ARTICLES  
SITE UPDATE ALERTS  
THE LATEST DB2 NEWS**

BE THE FIRST TO KNOW AT  
**WWW.DB2MAG.COM**

## Get Your IDUG<sup>®</sup> eSubscription Today!

Simply visit [www.idug.org/esubscription/](http://www.idug.org/esubscription/) and register for your **FREE Trial eSubscription** to IDUG's all-new user-focused gateway to DB2 UDB for OS/390, UNIX, Windows, and OS/2 information, education, and best-practice sharing.

If you are a DB2 database management professional, you can have any-time, anywhere electronic access to the information and resources you need to succeed through the IDUG eSubscription. **Take a couple of minutes and subscribe online for your FREE eSubscription. This offer will not be available indefinitely so subscribe today!**

[www.idug.org/esubscription/](http://www.idug.org/esubscription/)



401 N. Michigan Avenue  
Chicago, IL 60611-4267 USA  
Tel: +1.312.644-6610  
Fax: +1.312.321.5158  
Email: [idug@idug.org](mailto:idug@idug.org)  
[www.idug.org](http://www.idug.org)

*DB2 From the Users' View*



```

DSNU000I  DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = TEMP
DSNU050I  DSNUGUTC - TEMPLATE KBR DSN SY1AA.&DB..&TS..D&JDATE..UNLOAD
DSNU1035I DSNUJTDR - TEMPLATE STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - TEMPLATE KBR2 DSN SY1AA.&DB..&TS..D&JDATE..PUNCH
DSNU1035I DSNUJTDR - TEMPLATE STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - LISTDEF KLAAS INCLUDE TABLESPACE DBKLAAS.*
DSNU1035I DSNUILDR - LISTDEF STATEMENT PROCESSED SUCCESSFULLY

DSNU050I  DSNUGUTC - UNLOAD LIST KLAAS UNLDDN(KBR) PUNCHDDN(KBR2)
                SHRLEVEL(CHANGE) ISOLATION(UR)
DSNU1033I DSNUGULM - PROCESSING LIST ITEM: TABLESPACE DBKLAAS.TSKLAAS
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR2
                DDNAME=SYS00001
                DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.PUNCH
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR
                DDNAME=SYS00002
                DSN=SY1AA.DBKLAAS.TSKLAAS.D2000318.UNLOAD
DSNU253I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
                UNLOADED=24 FOR TABLE SY1AA.TBKLAAS
DSNU252I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
                UNLOADED=24 FOR TABLESPACE DBKLAAS.TSKLAAS
DSNU250I  DSNUNLND - UNLOAD PHASE COMPLETE, ELAPSED TIME=00:00:01
DSNU1033I DSNUGULM - PROCESSING LIST ITEM: TABLESPACE DBKLAAS.TSKLAAS2
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR2
                DDNAME=SYS00003
                DSN=SY1AA.DBKLAAS.TSKLAAS2.D2000318.PUNCH
DSNU1038I DSNUGDYN - DATASET ALLOCATED.  TEMPLATE=KBR
                DDNAME=SYS00004
                DSN=SY1AA.DBKLAAS.TSKLAAS2.D2000318.UNLOAD
DSNU253I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
                UNLOADED=12 FOR TABLE SY1AA.TBKLAAS2
DSNU252I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
                UNLOADED=12 FOR TABLESPACE DBKLAAS.TSKLAAS2
DSNU250I  DSNUNLND - UNLOAD PHASE COMPLETE, ELAPSED TIME=00:00:01
DSNU010I  DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0

```

**Listing 4:** The new unload utility with sysrec and syspunch datasets.

now more of a reality. These enhancements lessen the impact of some of the ONLINE REORG processes that prior to version 7 still caused outages that were unacceptable to those who needed true 24x7 availability.

The first improvement is the new FASTSWITCH keyword. This keyword for ONLINE REORG basically replaces the approximately three-second outage associated with the renaming of original and shadow dataset copies with a memory-speed switch of MVS catalog entries.

The second long-awaited enhancement is the improvement in elapsed time in the BUILD2 phase of REORG (SHRLEVEL CHANGE or REFERENCE). This improvement is useful when reorging one partition or a range of partitions. DB2 now creates multiple logical partitions in the nonpartitioned index (NPI) keys and then updates these logical partitions in parallel, making online reorg for partitions less disruptive

and causing less of an outage. DB2 will dispatch multiple subtasks to process the logical partitions asynchronously. In contrast, the current BUILD2 phase process is synchronous. Each subtask will perform updates to a logical partition. Lack of virtual memory, insufficient number of available threads, or insufficient processors can constrain the number of subtasks.

### LOAD PARTITION PARALLELISM

Support for LOAD parallelism against several partitions helps when you're faced with short windows of time in which to load a lot of data. In the past, these short windows caused problems, especially when NPIs (which I like to say stands for "numerous problems involved") were involved. You'd have to drop and recreate the NPIs in order to get the loads done. This enhancement lets you submit a single job with several input files to be loaded in parallel, re-

sulting in much faster performance and eliminating the contention on the NPI. The number of parallel load tasks will be determined by the number of CPUs and the number of threads available.

An enhanced -DISPLAY UTIL command shows the number of records updated for all logical parts. A new DSNU message (DSNU1114I) issued during the BUILD2 phase of the REORG utility indicates the number of logical partitions updated in parallel.

### ONLINE LOAD RESUME

The new online LOAD RESUME utility lets DB2 effectively combine the speed and performance of the LOAD utility with the availability and access offered by normal SQL INSERT processing. The online LOAD (LOAD RESUME YES SHRLEVEL CHANGE) operates much like a SQL INSERT program. It performs claims instead of drains, allowing for concurrency.

Online LOAD RESUME also attempts to maintain the clustering order of the data. If you use LOG YES only, COPY will not be required afterward. Locking problems are avoided through internal monitoring of the commit scope, although just how this is monitored is proprietary information. The Online LOAD RESUME can also be run in parallel for partitioned tablespaces.

### STATISTICS HISTORY

Version 7 offers better elapsed time reporting and statistics for performance evaluations over time. And, it lets you keep a history of statistics, allowing for better proactive performance analysis capabilities and better object monitoring. With these improvements, you can monitor growth over time and (with the help of additional information) determine if objects need to change. Statistics history is supported via nine new catalog tables. The RUNSTATS, REORG, LOAD, and REBUILD utilities use the new keyword HISTORY to update the statistics history. And, you can delete old statistics from the catalog history tables using the new MODIFY STATISTICS utility.

The new catalog tables that support historical statistics include:

- SYSIBM.SYSCOLDIST\_HIST
- SYSIBM.SYSCOLUMNS\_HIST
- SYSIBM.SYSINDEXPART\_HIST
- SYSIBM.SYSINDEXES\_HIST
- SYSIBM.SYSINDEXSTATS\_HIST
- SYSIBM.SYSLOBSTATS\_HIST
- SYSIBM.SYSTABLEPART\_HIST



```

DSNU000I  DSNUGUTC - OUTPUT START FOR UTILITY, UTILID = TEMP

DSNU050I  DSNUGUTC - UNLOAD TABLESPACE DBKLAAS.TSKLAAS
DSNU650I  + DSNUGMS - FROMCOPY SY1AA.DBKLAAS.TSKLAAS.D2000318.COPYFLB
DSNU253I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
UNLOADED=12 FOR TABLE SY1AA.TBKLAAS
DSNU252I  DSNUNLND - UNLOAD PHASE STATISTICS - NUMBER OF RECORDS
UNLOADED=12 FOR TABLESPACE DBKLAAS.TSKLAAS
DSNU250I  DSNUNLND - UNLOAD PHASE COMPLETE, ELAPSED TIME=00:00:01
DSNU010I  DSNUGBAC - UTILITY EXECUTION COMPLETE, HIGHEST RETURN CODE=0

```

**Listing 5:** UNLOAD execution using imagecopy and normal dd.

- SYSIBM.SYSTABLES\_HIST
- SYSIBM.SYSTABSTATS\_HIST

The new syntax (pseudo) looks like:

- REORG INDEX...HISTORY  
(ALL|ACCESSPATH|SPACE|NONE)
- RUNSTATS TABLESPACE...HISTORY  
(ALL|ACCESSPATH|SPACE|NONE)

### UTILITY PACKAGING

Version 7 utilities are grouped into three categories. Some utilities are shared across both options in order to provide a complete functional set. The packages include:

- **DB2 core utilities**, delivered with

the core DB2 product at no additional charge

- **DB2 operational utilities**, which handle routine DB2 operations and are available for an additional charge

- **DB2 recovery and diagnostic utilities**, which are necessary in case of emergency and are available for an additional charge. For details about which utilities are included in each package, go to [ibm.com/software/data/db2/imstools](http://ibm.com/software/data/db2/imstools).

### AUTOMATION FOR THE DBAs

The many other enhancements to the utilities are simply too numerous to mention here. Of course, with the templates and associated features, the stored

procedure DSNUTILS takes on new meaning and power. With the ability to have triggers set up to watch events, and with the threshold controls in some of the normal operational utilities (like REORG), it would be possible to truly automate some of the more mundane DBA functions. Some of the power of these new utilities derives from their integration into the Control Center, because it lets you easily generate utility jobs without having to know all the various parameters involved. You can simply select the options required, save the generated output for future use, or simply add it into the operational schedule. ●

*The author acknowledges Susan Lawson and Klaas Brant for their assistance with this article.*

**Richard Yevich** is a consultant specializing in enterprise information systems. Formerly a senior consultant with Codd and Date Inc. and RYC Inc., Richard is currently a principal with Yevich, Lawson & Associates Inc., a firm specializing in database performance, technical research, and advanced education. He is a widely published author and coauthor of books and magazine articles. You can reach him by email at [Richard\\_Yevich@YLAssoc.com](mailto:Richard_Yevich@YLAssoc.com) or by telephone at 1-888-246-5049.



***Having trouble transferring your DB2 database to the E-World?***

**Get a boost!**

**SIVCO Inc can help!**

**Our offerings are:**

WebSphere Integration

Data Warehouse, Data Mining

Leading your team to build the E-Business solution  
(B2B, B2C)

Calgary, AB \*\*\* Olympia, WA

[Http://www.sivco.com](http://www.sivco.com)

Toll Free Call: 1-877-700-2600 e-mail: [contact@sivco.com](mailto:contact@sivco.com)







# The Database Security Blanket

*Hackers and crackers and spies, oh my! Understanding database security is the first step to companywide peace of mind.*

**L**et's face it: The topic of database security just doesn't turn heads the way benchmarking world records and reports of ever-shorter downtimes do. When was the last time you read a scintillating article about security tokens and encryption? But security breaches do turn heads — and can undermine customer confidence, as last year's well-publicized thefts of credit card numbers from a few e-businesses showed. Even if it's not the most exciting topic, security is a vital consideration for any business that uses a database management system. And, as more businesses participate in the e-space, it becomes particularly important to separate private from public data.

The database system at any given company probably collects, stores, and analyzes millions of rows of information ranging in nature from public to private. With all that responsibility, the database must enable DBAs to grant and restrict access appropriately. And it must offer ways to keep unauthorized users from accessing classified data.



But sometimes, database security information is hard to come by or difficult to understand. Although you often hear about how scalable and robust DB2 Universal Database (UDB) is, how often do you hear details about DB2's security features?

Because securing data is one of the most important DBA responsibilities, you shouldn't try to learn about database security by trial and error. Securing your database involves:

- Preventing unauthorized access to classified data by anyone without a business need to know
- Preventing unauthorized users from committing mischief through malicious deletion or tampering of data
- Monitoring user access of data through auditing techniques.

In this article, I'll walk you through the security features in DB2 UDB v.7.1 for Windows, Unix, and OS/2 and describe some internal controls that can help you maximize your security.

## AUTHENTICATION

One of the most basic concepts in database security is *authentication*, which is quite simply the process by which a system verifies a user's identity. A user can respond to a request to authenticate by providing a proof of identity, or an *authentication token*.

You're probably already familiar with this concept. If you have ever been asked to show a photo ID (for example, when opening a bank account), you have been presented with a request for authentication. You proved your identity by showing your driver's license (or other photo ID). In this case, your driver's license served as your authentication token.

Despite what you see in the movies, most software programs can't use futuristic systems such as face recognition for authentication. Instead, most authentication requests ask you to provide a user ID and a password. Your user ID represents your claim to being a person authorized to access the environment, and the password provides your authentication. Of course, this kind of authentication assumes that your password is well protected and you are the only person who knows it.

User authentication is completed by a security facility outside DB2, often part of the operating system or a separate product. In fact, security is not just a database issue; operating system vendors spend a great deal of time, money, and care to ensure that their products are secure. However, some operating systems,

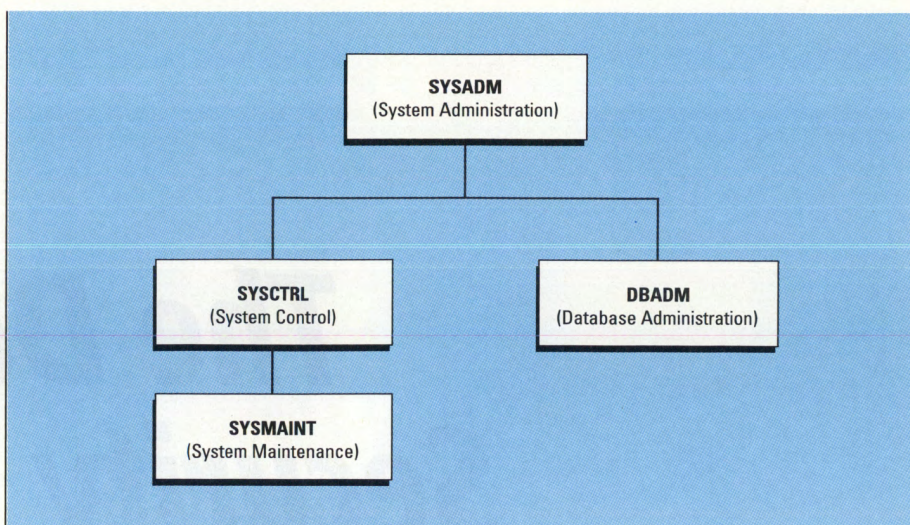


Figure 1: DB2 authorization roles.

# DBAs must have the power to grant and restrict access.

including Microsoft Windows 95 and 98, have no native security mechanism. If you use an operating system that lacks security mechanisms, you can configure your environment to rely on the DB2 server on a more secure system to provide the security. For example, you can use trusted client options, which I'll discuss more later in the article. (For more information, refer to the DB2 Administration Guide.)

You can also use third-party products, such as Distributed Computing Environment (DCE) Security Services defined by the Open Group, to add a layer of security to your environment. DB2 can coordinate these external security efforts with its security initiatives to protect a transaction or analytic environment.

Once a user is successfully authenticated, DB2 makes note of the user's identity and other relevant security information, such as the user's group list. The user must be identified to DB2 using a SQL authorization name or `authid`, which can be the same as the user ID or a mapped value. This connection information is kept for the duration of the user's connection.

## AUTHENTICATION OPTIONS

Because authentication can be handled either by the operating system or by third-party products, DB2 offers different authentication options that you can set in the database manager configuration (`dbm cfg`) file using the `AUTHEN-`

`TICATION` parameter. DB2 uses this parameter to determine how and where authentication should take place.

The many settings for the `dbm cfg AUTHENTICATION` parameter can be logically grouped into four distinct categories: Server, Client, DCE, and Kerberos.

**Server authentication.** This group offers two main options:

- **SERVER**, the default security mechanism, specifies that the authentication is to occur on the server using the server's operating system. If a user ID and password are specified during the connection, DB2 calls an operating system function to validate the submitted user ID and password. (In a Windows-based environment, a user ID is often referred to as a *username*. The combination of a username and password is often referred to as a *user account*.)

- **SERVER\_ENCRYPT** is essentially the same as the default option, except that the password passed from client to server is encrypted. DB2 uses 56-bit single DES encryption of the password and the Diffie-Hellman algorithm to generate a key for the encryption algorithm at connect time. The RSA BSAFE toolkit provides this support.

**Client authentication.** This group's only option, `CLIENT`, specifies that the authentication is to take place at the client. If this client resides on an operating system that has inherent security features (for example, AIX), it is considered a *trusted client*. In general, all clients





e-business software

# To move at the speed of business, UPS runs on DB2.

It's a different kind of world.  
You need a different kind of software.

When you track over 12 million packages a day, you need powerful, reliable software. By choosing IBM DB2® Universal Database,™ this is exactly what UPS has done. Not only are customers happy about this arrangement, so is UPS. DB2 is part of the IBM portfolio of e-business software. To learn more, visit [ibm.com/software/different/db2](http://ibm.com/software/different/db2)

IBM and DB2 are registered trademarks and DB2 Universal Database and the e-business logo are trademarks of International Business Machines Corporation in the United States and/or other countries. Other company, product, and service names may be trademarks or service marks of others. © 2000 IBM Corporation. All rights reserved.



# A DOUBLE-EDGED SWORD

When it comes to data and security, recent legal developments strengthen company rights and remedies for managing database content. But there are also a growing number of legal obligations database managers should know about. These rights and responsibilities exist not only in the United States, but also in many other nations around the world.

Increasingly, data is considered a form of property. The law has long recognized the right of owners of land or personal property to protect that property from unauthorized use by other parties. As these same basic notions of property ownership rights are being extended to the database developers, property law and regulations are evolving into a tool to help database owners control data access and provide protection against theft and sabotage.

The law now appreciates more fully the commercial value of data collections. In the United States, laws protecting trade secrets at the state level and the Economic Espionage Act at the federal level provide protection for proprietary data. U.S. courts are beginning to consider information a form of property that can be protected by law, as is the case in the ongoing controversy between eBay and Bidder's Edge, which involves limits on access to online data by intelligent agent software. The legal concepts of commercial torts and unfair trade practices are now applied by state courts, federal courts, and regulatory agencies such as the U.S. Federal Trade Commission to protect data. Database professionals can raise legal claims based on commercial harm resulting from the unauthorized use of data, and those claims can lead to court awards of monetary compensation or court-ordered changes in conduct.

But the law also enforces an obligation on database professionals to apply reasonable measures to ensure the security of database content. Therefore, security measures that control database

access are essential to minimize the risk of liability. If data is compromised and the database operators did not apply reasonable security measures, those operators face liability for commercial harm resulting from the misuse of the data. In addition, if the compromised data is protected by specific regulations (for example, personal data about individuals, financial data, or medical records) database operators would face additional liability.

Database professionals also face liability for the content of their databases. To protect themselves from liability claims from parties who provide data for the database, administrators must ensure that all material included in databases is used in a manner consistent with any intellectual property rights or contract obligations associated with that material. The database administrator must clearly outline and enforce the authorized terms of use. And, to protect themselves from liability claims from database users, administrators must make sure that any limitations or special terms associated with data content are clearly expressed (for instance, disclaimers that state how current or accurate the data is).

The law establishes enforceable rights that make the work of database professionals a bit easier and binding obligations that make their job a little more complex. Prudent professionals appreciate both sides of that sword and work closely with their legal team to make the law work for them.

For an overview of the ever changing legal environment, look for Web resources such as Tech Law Journal ([www.techlawjournal.com](http://www.techlawjournal.com)).

**Jeffrey Matsuura** is an attorney with the Alliance Law Group, a technology law firm in Vienna, Va. He is author of the book *A Manager's Guide to the Law and Economics of Data Networks* (Artech House, 2000) and coauthor of *Law of the Internet* (Aspen Publishers, 1997). He can be reached at [jmatsuura@alliancelawgroup.com](mailto:jmatsuura@alliancelawgroup.com).

are trusted except for Microsoft Windows 95 and 98, which are considered untrusted.

If a server receives requests from trusted and untrusted clients, the `TRUST_ALLCLNTS` and `TRUST_CLNTAUTH` options allow trusted clients to gain access using client authentication while untrusted clients must provide a password for successful authentication. Please see the DB2 Administration Guide for details.

**DCE authentication options.** Some administrators choose to implement DCE security services because DCE provides centralized administration of users and passwords, no transmission of clear text passwords and user IDs, and a single sign-on for users. DB2 provides integrated support for DCE security services using third-party DCE products. You can choose between two settings:

- **DCE** specifies that the user is authenticated using DCE security services. A DB2 client that has logged into DCE can obtain an encrypted "ticket" it can use to authenticate itself with the DB2 server.

- **DCE\_SERVER\_ENCRYPT** specifies that the server will accept DCE tickets or user ID and encrypted password as proof of authentication, as selected by the DB2 client.

#### **Kerberos authentication options.**

Kerberos is a new authentication mechanism added to DB2 UDB v.7.1 as part of its tight integration with Microsoft Windows 2000. Using Kerberos security, DB2 authentication can be accomplished with single sign-on facilities. Once authenticated, a user will not be challenged again by any server that exists in the Kerberos environment. This method of authentication can only be used when both the DB2 client and the DB2 server are on Windows 2000.

DCE and Kerberos use essentially the same underlying technologies. When a client logs into the Kerberos security environment, the DB2 client can obtain an encrypted Kerberos ticket for use in authenticating itself with the specified DB2 server.

You can choose between two settings:

- **KERBEROS** specifies that the user is to be authenticated using Kerberos security services only.

- **KRB\_SERVER\_ENCRYPT** specifies that the server is to accept either Kerberos tickets or user ID and encrypted password as proof of authentication, as selected by the client.

#### **AUTHORIZATION**

An authenticated user goes through the second DB2 layer of security, *authoriza-*



tion. Authorization is the process through which DB2 obtains information about the authenticated DB2 user, including which database operations that user may perform and which data objects that user may access.

Your driver's license is a perfect example of an authorization document. Though it can be used for authentication purposes, it also authorizes you to drive a certain class of car. Furthermore, the type of authorization you have gives you more or fewer privileges as far as driving a vehicle goes.

In Ontario, Canada, for example, a class G license gives you the privilege to drive a car practically wherever and whenever. The class G license sits at the top of the hierarchical structure of authorizations for driving cars. The lower-level G2 license allows a user to drive a car at any time of day, but with certain restrictions. (For example, a user with this authorization scheme is not allowed to drive after consuming any alcoholic beverages whatsoever.) Below the G2 license is the G1 license, which includes many restrictions (for example, the driver must be accompanied by a class-G licensed passenger, may not drive on highways, or drive after dark).

Your driver's license authorizes you to drive a vehicle and limits access to certain "objects" (for example, you cannot drive on a highway with a G2 license). DB2 works in much the same way.

Authorization can be broken down into two distinct categories: *authorities* and *privileges*.

**Authorities.** Authorities provide a method of grouping privileges and control over higher-level database manager maintenance and utility operations. Database-specific authorities are stored in the database catalogs; system authorities are associated with group membership and are stored in the database manager configuration file for a given instance.

DB2 has four predefined authority levels: SYSADM, SYSCTRL, SYSMANT, and DBADM. Figure 1 (page 22) illustrates the hierarchical scheme of the predefined authorization levels used in DB2.

SYSADM, SYSCTRL, and SYSMANT operate at the instance level and have a serverwide scope. Each has its own grouped privileges and authorized-access rules (much like the class G license). These authorities are defined in the database manager configuration file for each instance.

The DBADM authorization level is linked to a specific database within the

server instance and is automatically granted to a user who creates a database. A DBADM has all possible grouped privileges on the database and all the objects within that database.

By default, a SYSADM has all possible grouped privileges for the entire system, including databases (it has implicit DBADM authority).

DB2 uses more than one vertical authorization stream. With each user request, there may be more than one au-

thorization check, depending on the objects and operations involved. Authorization is performed using DB2 facilities. DB2 system catalogs contain a record of the privileges associated with each authorization name. The authorization name of an authenticated user, and the groups to which the user belongs, are compared with their recorded privileges. Based on this comparison, DB2 decides whether or not to allow the requested access.

# Problem DB2 Threads?

Cancel them  
gracefully, safely  
and with minimal impact



## Thread/SENTRY

embeds DBA intelligence in a rule-based, expert system that cancels threads and issues warnings automatically. Thread/SENTRY lets you pre-define DB2 'problem' threads in terms of over 100 constraints such as idleness and CPU time.

## Thread/STOPPER

provides a set of manual facilities to deal with DB2 problem threads

- from within ISPF
- from the MVS console
- via batch commands

## Features

- Cancels threads holding locks on the DB2® objects (tables, tablespaces, etc.) you specify
- Cancels DB2 editors without placing the TSO session in a wait state
- Supports user exits to handle special requirements

Resolve DB2 problems fast,  
before users start calling the help desk

DB2® is a registered trademark of IBM Corp.

**Relational  
Architects  
International**

800 776-0771  
201 420-0400 (Tel)  
201 420-4080 (Fax)

[www.relarc.com](http://www.relarc.com)



```

C:\DB2> update dbm cfg using catalog_noauth no
SYSADMN "TESTING" does not have the authority to change the database manager
configuration file.
C:\DB2 ? eq15081n
SQL5081N "Authorization-ID" does not have the authority to
change the database manager configuration file.
Explanation: The user attempted to update or reset the database
manager configuration file without having SYSADM authority.
The requested change is not made.
User Response: Do not attempt to change the database manager
configuration file without appropriate authorization. Contact a
user with SYSADM authority if a change is required.
C:\DB2>

```

**Figure 2:** An error message returned to a user not authorized to perform a specific command.

For example, Figure 2 shows the results of entering a command that requires a user to have a particular authorization privilege.

The DB2 security mechanism blocks the TESTING user ID because it knows that this user does not have the authorization to perform such a command. In this case, it is obvious that TESTING is not a SYSADM, since that authority level is required to execute the command shown in Figure 2.

You gain SYSADM, SYSCTRL, and SYSMANT authorities in DB2 by being a member of the group specified by the SYSADM\_GROUP, SYSCTRL\_GROUP, and SYSMANT\_GROUP database manager configuration parameters, respectively.

For a complete listing of all the authorization privileges granted to a user belonging to any of the DB2 predefined authorization groups, refer to the DB2 Administration Guide.

**Privileges.** A privilege defines a single permission for an authorization name, enabling a user to modify or access database resources. Privileges are stored in the database catalogs. Whereas authority groups have a predefined group of privileges that are implicitly granted to their members, privileges are single permissions.

DB2 can take advantage of the user groups maintained by an operating system's security feature. Groups help reduce the database system's total cost of ownership by allowing the database administrator to assign privileges to groups.

For example, if you wanted to allow the privileges of creating tables (CREATETAB) and connecting to the database (CONNECT) to 50 users, it would be easier to create a group and add the privileges to it than to explicitly grant each privilege to each individual user. And, anytime you needed to add or revoke privileges, you could do it once for the group and all the members of the group would be affected.

In general, group membership is applicable for dynamic SQL and nondatabase object authorizations (such as instance level commands and utilities). Dynamic SQL is SQL that is generated ad hoc, or on the fly. Group membership (except the PUBLIC group) is not applicable for static SQL. Static SQL is known to DB2 before execution and the access plan for the SQL has already been generated (by the DB2 optimizer) and stored as a package in the catalogs.

Each privilege can be granted or revoked from a specific authid, a specific group (PUBLIC) to which all users automatically belong, or more than one group. Figure 3 gives an example of some of the different privileges available in DB2. (Note: In the Tables and Views group, the ALTER, INDEX, REFERENCES, and UPDATE Column privileges are only available for tables.) For an in-depth discussion of these privileges, see the DB2 Administration Guide.

Access to all sorts of operations and objects is controlled through privileges. You must have the privilege to do some-

thing before DB2 lets you do it.

## ACCESS CONTROL METHODS

Now that you understand how DB2 manages authentication and authorization, let's look at the framework that DB2 provides to combine these options for strong access control. Access control methods are used to create subsets of the contents of information, so that the user can only see and access data that is relevant to their needs. You can use a number of different access control methods in DB2. Access controls exist for everything you do in a database.

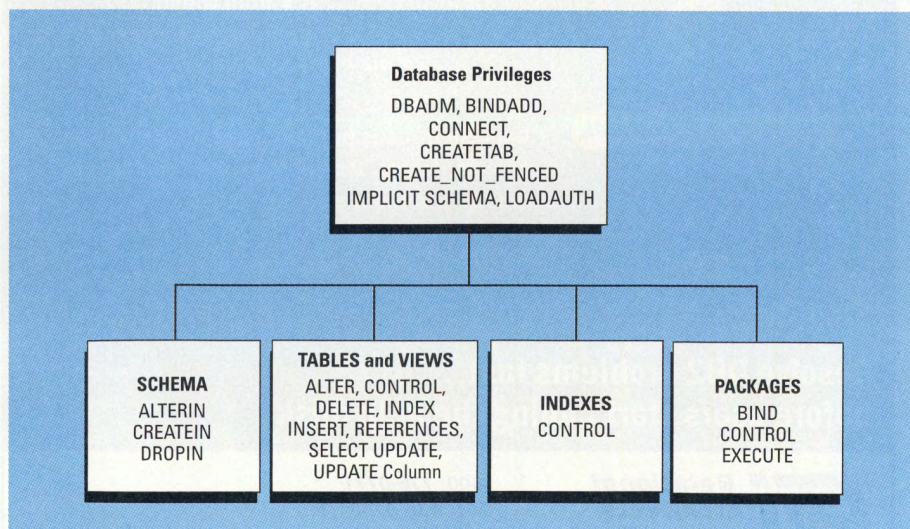
Consider the row of a view in Table 1. You may want all users in the company to have access to the L\_Name, F\_Name, Phone, email, and Title columns. However, to minimize the risk of email spam, you may want users accessing a company directory over the Web to only see the L\_Name, F\_Name, and Phone columns. Finally, employees who are part of the Human Resources department should have access to all the rows in this table. Access controls provide a very important framework in securing your data in DB2 and providing row-level access to your data.

### Access controls using packages.

Packages, collections of information that pertain to one or more SQL statements, are the primary access control point for SQL within DB2. Packages include such information as the access plan generated by the optimizer and the authorization model. Any statement issued to the database engine is related to a specific package.

When a package is created, it is bound to the database with specific privileges. The person who creates a package must have the privileges required to execute all the static SQL statements in the package. Users who run the package must also have the EXECUTE privilege for the package, but they don't have to have the individual privileges for executing the static statements contained in it. For example, without the CREATETAB privilege, a user can't create a table in the database. However, a user with the privilege to run a package containing a static CREATE TABLE statement could create a table that way. Packages play a key role in controlling the access that users have to database objects.

**Access control using views.** Views are another major access control method to restrict low-level (also called row-level) access to data. Using a view, you can hide sensitive rows and columns of information that reside in the original table from SQL statements. You can give a user ac-



**Figure 3:** Sample privileges available in DB2.



L_Name	F_Name	Phone	email	Title	Salary	Bonus	Total Salary
Godfrey	Mike	2447337	mk@money.com	Mgr	\$23,000	\$122,000	\$145,000

Table 1: A row of a view.

cess to information by granting privileges on a view. Because these privileges are only for the view and do not affect the base table, the user's access is confined to the view, which is generated by creating a subset of the data contained in required table. The WITH CHECK option provides even more security because it will not let a particular SQL statement change rows that the user is not privileged to read in the view.

**Access control using triggers.** Using a trigger, you can create a more complex security mechanism that gets initiated whenever a certain event happens. An INSERT statement on a table is one example of an event that can be used to fire a trigger. Furthermore, the trigger can fire before or after a specified event, to provide even more rigorous security checks. If the user's statement fails to pass a security check in a trigger, an error can be signaled from within the body of the trigger to prevent the table from being modified.

**Access control using the USER special register.** DB2 provides a special register, called USER, which contains the user ID used to connect to the database for the current session. You can use the value stored in the special register in a view to make customizations for the view for a particular user. Using the special register, you can make the view, which is based on a table, look different from user to user. You can also use this technique with triggers and SQL statements.

#### AUDITING FEATURES

DB2's auditing facility lets you maintain an audit trail of events that occurred within an instance. Successful monitoring of data access attempts and subsequent analysis can lead to improvements in the control of data access and the ultimate prevention of malicious or careless unauthorized access to the data. These recorded events can then be extracted into a report for analysis. Auditing is detailed in the DB2 Administration Guide.

#### TRUST THE DATABASE

DB2's many features and mechanisms let you trust the database engine with even your most secure and confidential data. When your data is stored and managed by DB2, you can be confident that your business can run in a scalable, robust, and secure environment. And, if you're in charge of database security, that means a better night's sleep for you. ●

**Paul Zikopoulos** is a database specialist with the DB2 Sales Support team. He has written many magazine articles and books about DB2. Most recently, Paul coauthored the books *A DBA's Guide to Databases on Linux* (Syngress Media, 2000) and *DB2 for Dummies* (IDG Books, 2000). Paul is a DB2 Certified Advanced Technical Expert (DRDA and Cluster/EEE) and a DB2 Certified Solutions Expert (Business Intelligence and Database Administration). You can reach him at paulz\_ibm@yahoo.com.

#### RESOURCES:

For more information about DB2 UDB security features, see the DB2 Administration Guide at:

[www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v7pubs.d2w/main](http://www.ibm.com/cgi-bin/db2www/data/db2/udb/winos2unix/support/v7pubs.d2w/main)

## Make e-Business Happen Faster!

This is your Enterprise Application...

This is the costly SQL that is slowing down your business...

This is SQL-GUY...

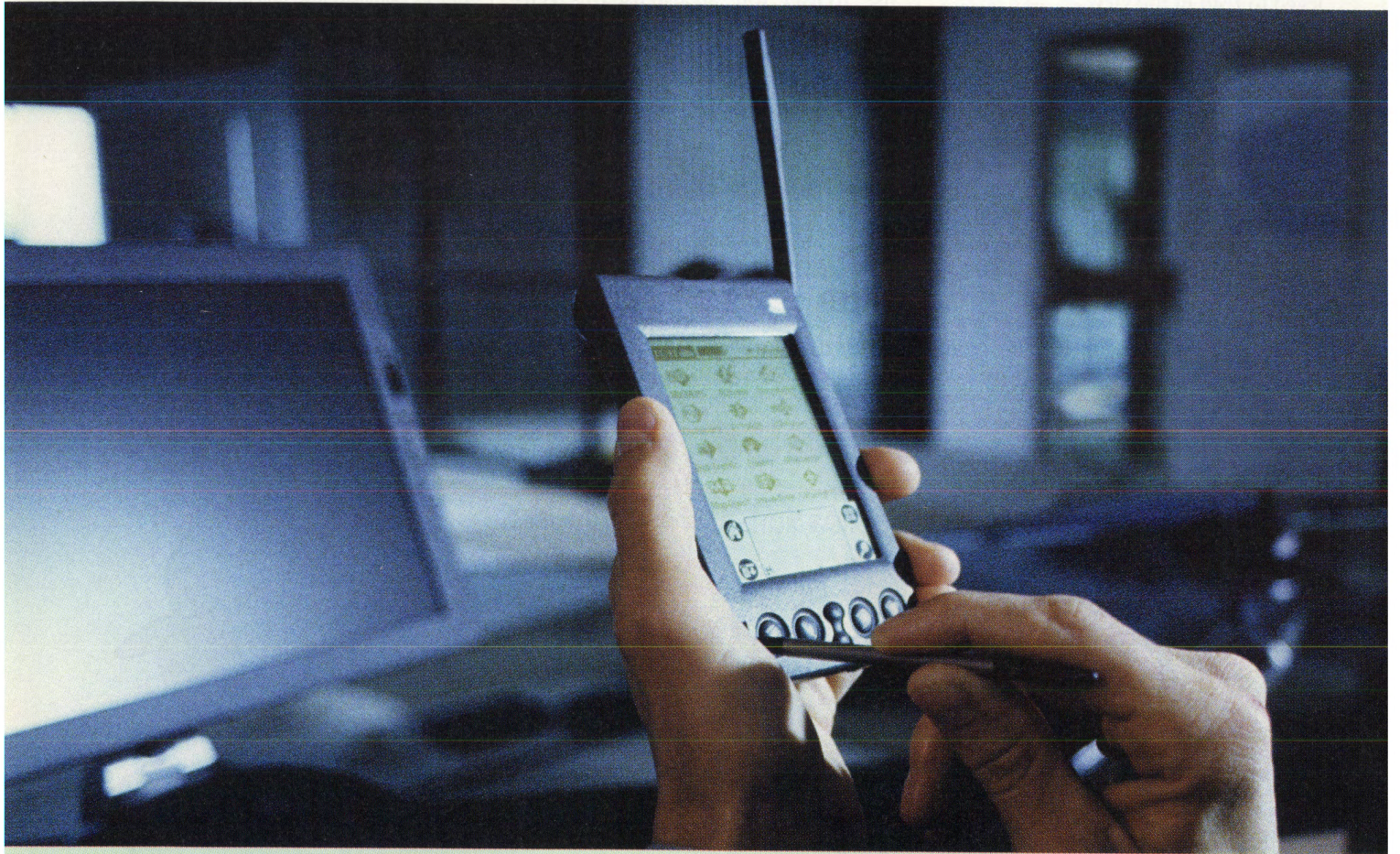
Any Questions?  
[www.Database-GUYS.com/domore/](http://www.Database-GUYS.com/domore/)



Performance Management Tools, Consulting, and Support  
**DB2 UDB for UNIX... It's all we think about!**  
[www.Database-GUYS.com](http://www.Database-GUYS.com)



# Dial M for Business





## *How a combination of mobile and Internet technologies helped a bricks-and-mortar company escape the paper past for the digital future.*

**W**hen a large wholesaler in Finland approached my company, Rissa Solutions, for a way to achieve the widest possible access to its enterprise resource planning (ERP) system from Web clients and PDA devices, I had the chance to explore the true possibilities and limitations of mobile technology today. In designing the solution, which relies on DB2 Universal Database (UDB) version 7.1, my colleagues and I had to overcome credibility, technology, and functionality challenges. Although customers in Finland are familiar with mobile systems, they still want to see a proven technology backbone that will let them escape potential bottlenecks. Our experiences in overcoming these challenges might prove useful to other companies designing mobile solutions.

### INCREASING ACCESS AND INTEGRATING SYSTEMS

While looking for new opportunities to increase market share, the wholesale company realized it needed to provide a better way for businesses in its own distribution network and independent retailers (such as general stores, kiosks, gas stations, commercial kitchens, and so on) to place orders. The company, which specializes in daily goods (such as groceries and household items) and sporting goods, sells about 60,000 different items. It relies on an SAP R/3 ERP system and an older ERP system (which will soon be upgraded to SAP or another major vendor's product) for centralized pricing and other management. Suppliers can deliver goods either to the warehouses or directly to the stores (see Figure 1, page 30).



Despite the centralized control the ERP systems provide, the company's order process suffers from a lack of integration. Currently, all stores place their orders to the central management or to the warehouses, and all independent retailers place their orders to a service center by making a phone call or sending a fax. Employees in the service center use native SAP clients, which lack a connection to the warehouses' information systems. Purchase orders are placed to the SAP system, and pick lists are then printed and delivered to the warehouse (see Figure 2).

This lack of integration creates a number of problems. First, there is no way to confirm that items are in stock, which causes delays and probably frustration for the retailers. Second, it is almost impossible to control prices centrally. Although the SAP system provides the desired centralized pricing control, the company doesn't currently have the ability to gather information relevant to pricing from the various shops or from the warehouses. Clearly, the company needed an order process that would support strong, centralized supply chain management and maximize the company's existing ERP processes.

### E-BUSINESS SOLUTION OVERVIEW

The wholesaler needed new channels, specifically Internet and mobile clients, for its own and the independent retailers. To connect and integrate the ERP systems with the service center and the Internet and mobile clients, we used a Rissa Solutions middleware product called Netseller, which enables connections to many different ERP systems at the same time (see Figure 3). (Plumtree Software offers a product with similar capabilities.)

Integrating the separate ERP systems in order to share information was only the first step. The next step proved to be the most interesting: The wholesaler decided to provide personal digital assistant (PDA) devices to all retailers and to the warehouses. The solution is designed to handle up to 10,000 PDA devices. Each device can read barcodes and can be used in many environments, including WAN or LAN and Global System for Mobile Communications (GSM) and General Packet Radio Service (GPRS) networks. With these PDA devices, warehouse employees can do inventory, create purchase orders, and check pick lists from the SAP system. Retailers can also create purchase orders, do inventory, check over their orders, and calculate

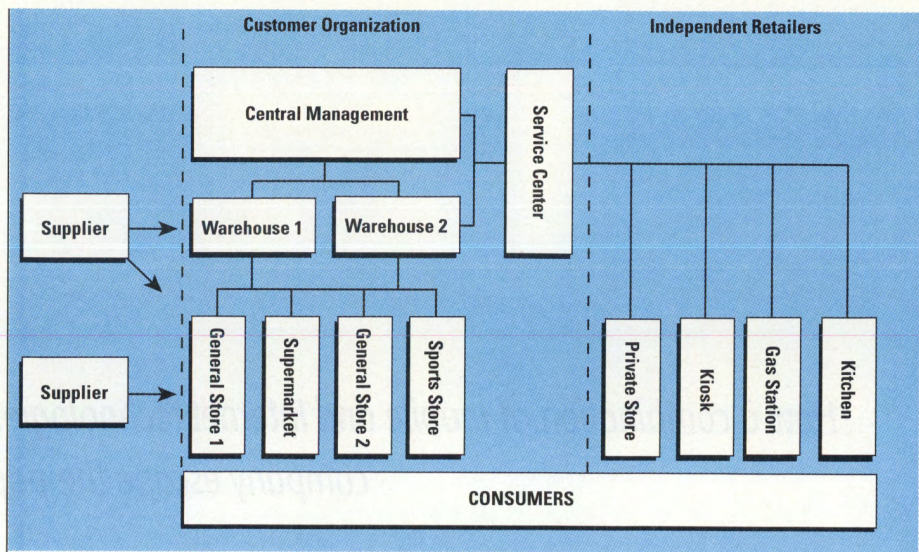


Figure 1: The Finnish wholesaler's distribution network.

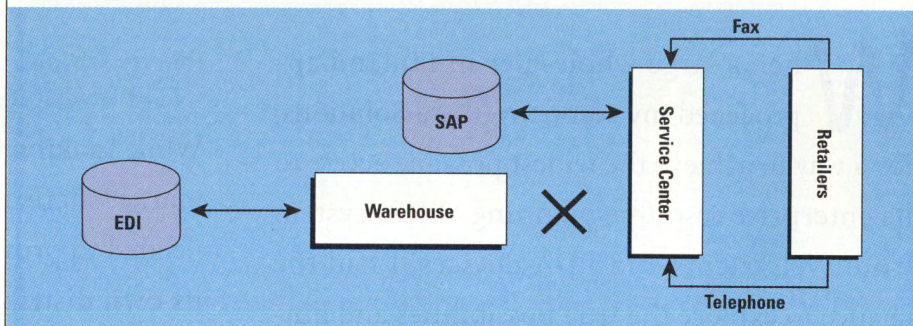


Figure 2: The current order process, which relies on phone and fax.

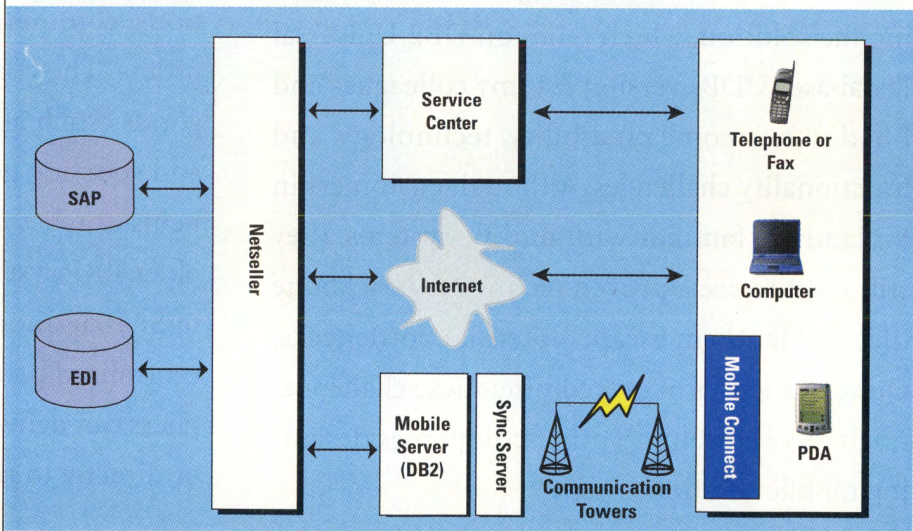


Figure 3: The new order process, which allows access from mobile and Internet sites.

deficits. In addition to the mobile aspects, the solution also includes a Web interface for retailers and consumers.

This solution lets the wholesaler offer their customers multiple channels for communication while keeping centralized control. It also gives the company a competitive edge, as they are among the first to implement such a solution. Which company would you rather work with,

one that offers you a PDA and Internet access or one that still relies on paper, pencil, and fax machine?

### DESIGNING THE SOLUTION

When designing this mobile solution, we had to keep in mind several factors, including:

**Network coverage.** In Finland, the current geographical coverage of mobile



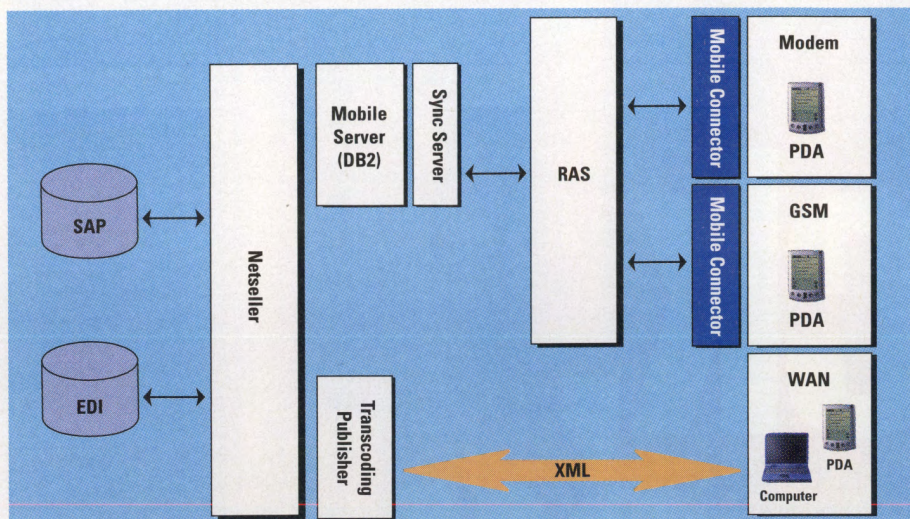


Figure 4: Different solutions for different kinds of access.

networks such as GSM is almost 100 percent. Nevertheless, there are many places where no network service is available (because radio waves are not able to penetrate everywhere). Basements, some areas inside buildings, and areas with low population are a few examples. Yet these places might support commercial activities: A store or office could be located in a basement, a gas station might exist in an area with poor network coverage,

and so on. We had to take these possibilities into account in our design.

**Available bandwidth.** All mobile networks have restrictions in regards to available bandwidth. The standard GSM data system allows 9600 bps. GPRS will have, in theory, 171.2 Kbps and WAN/LAN can have 2Mbps, 11Mbps (IEEE 802.11b) or 52Mbps (IEEE 802.11a). However, these numbers represent the total amount of what these networks can

handle. In real life, you're likely to get about 50 to 65 percent of the bandwidth — and you have to share this amount with other users. So, bandwidth is a problem now and will continue to be in the future. Only WAN or LAN technology provides relief in small geographical areas or locations.

**Functional requirements.** When you have 60,000 rows of product information, you don't want to deliver it to each and every single client, especially if you have regular analog modem lines or a 9600 bps mobile network. To avoid sending all the data to everyone, we defined groups based on the information that the groups need. Using groups, we can determine which subset of information to send to each group rather than sending the entire 60,000 rows to each device. The application we designed generates transactions (including purchase orders, inventories, price changes, and so on) from the ERP system and back to the ERP system. As you can probably imagine, the warehouse requirements are quite different from those of the gas stations. The warehouses typically have about 30,000 different items in stock; supermarkets have 12,000 to 15,000 items; general stores 2,000 to 3,000; and gas sta-



## DB2<sup>®</sup> Performance Journal

- Quarterly fee based subscription
- 6-8 in-depth papers
- Devoted to Performance Topics
- Written by top-notch experts
- Additional benefits

# Yevich, Lawson and Associates, Inc.

## Offering the Finest in DB2<sup>®</sup> Education

- Version 6 and 7 Transition
- SQL Performance and Coding
- Advanced and Complex SQL
- High Performance Design and Tuning
- Datasharing Performance and Implementation
- Advanced Programming Techniques
- Object Relational DB2 Programming
- Data Warehousing Design and Performance
- DB2 UDB for Linux, Unix and Windows
- ....and much more!



Visit our website for details! • [www.ylassoc.com](http://www.ylassoc.com)



**Quick,  
Who Offers  
The Best  
Tools For  
DB2?**



Computer Associates. That's who.

CA is one of the industry's leading vendors of tools for DB2 for OS/390, and is a true, one-stop shop for all of your IT needs. CA has a broad range of solutions that enable you to manage your applications and maximize the return on your DB2 investment in Performance Management and Analysis, Database Administration, Data Availability, Backup and Recovery, and Application Development.

CA provides proven solutions for DB2, that will ensure successful implementation and management of all of your business-critical applications, including third-party ERP applications running on DB2 for OS/390.

## Solutions for DB2 for OS/390

REDUCE COSTS	GAIN COMPETITIVE ADVANTAGE		PROVIDE PROACTIVE MANAGEMENT	
<b>Database Administration</b>	<b>Data Availability</b>	<b>Backup and Recovery</b>	<b>Application Development</b>	<b>Performance Management</b>
Streamline administration tasks	Reduce or eliminate downtime	Keep data secure and reliable	Accelerate development cycles	Monitor critical resources
Concurrently manage multiple databases	Perform online reorganization	Recover specific objects or entire databases	Integrate legacy data with new applications	Improve data retrieval
Alter and migrate database objects	Quickly load and unload data	Maintain business profitability	Generate error-free SQL code	Optimize database tuning efforts

CA Solutions for DB2 for OS/390 give you complete control of your mission-critical information.

CA recently completed its portfolio of DBA tools by adding IMS database solutions to its product portfolio. Don't be held hostage by



other tool vendors when CA can offer a single vendor solution across a multiple database environment, complete with an aggressive value proposition, and conversion services. Just send an e-mail to [rescueprogram@cai.com](mailto:rescueprogram@cai.com) for information on CA's Rescue Program and your no obligation product conversion analysis.

And remember, when you think DB2, think CA.



**For more information, call 1-800-850-7528 Ext. 100, or visit our website at [www.cai.com](http://www.cai.com)**

**COMPUTER ASSOCIATES**<sup>®</sup>  
Software superior by design.



tions have 200 items or less. Therefore, we had to design different kinds of solutions for different kinds of users (see Figure 4, page 31). For those businesses connecting via modem or GSM, we designed a platform that uses a mobile server (based on DB2) and IBM Sync Server to handle clients using offline replication with Netseller. For those businesses with a WAN or LAN connection, we decided to use IBM Transcoding Publisher, which lets devices that have online connections receive XML messages from the server. Transcoding Publisher basically generates different XSL stylesheets for different PDA devices. These stylesheets are used to display the XML information correctly in every different PDA device.

Adding to the complexity, all the warehouses and the supermarkets have LAN or WAN connections to the central management. Some of the general stores have a connection as well, but no independent retailer does. In a few cases, retailers don't even have a fixed telephone line, they only use a cellular phone.

**Users.** Most of the people using this solution are familiar with cash registers, but they may not be familiar with computers. Therefore, the solution we designed had to look like an extension to a cash register more than a mobile e-business solution. We brought in usability experts to study the way people actually work and what kind of user interface they prefer.

## ONLINE VS. OFFLINE TECHNOLOGY

All the warehouses and supermarkets have a LAN or a WAN. Depending on the bandwidth of the WAN, we will either use simple browser technology with IBM Transcoding Publisher or we will use offline technology (see Figure 5). All the PDA devices will have their own IBM DB2 Everyplace databases and IBM Mobile Connect clients installed. Using offline technology in a WAN or LAN environment lets you spare bandwidth and build solutions that use a relatively fat client. Using online technology through Web browsers requires the solution to handle all the information on the server side, requiring more from both the server and the WAN or LAN connection.

Using IBM Transcoding Publisher is extremely easy. We export all information from Netseller in XML format and then create XSL style sheets for every client. However, we often like to use offline technology even with online connections. If you rely on pure browser tech-

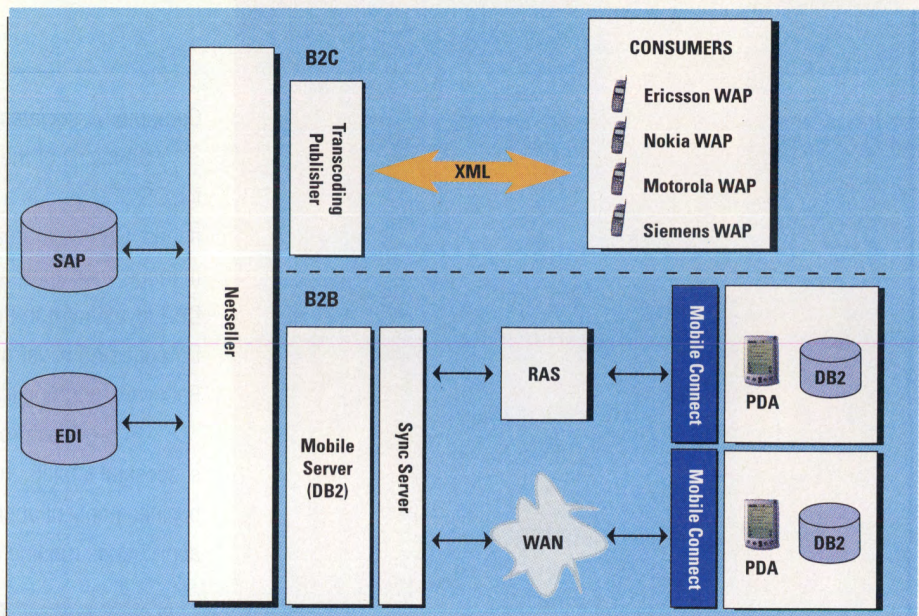


Figure 5: Architectures for online and offline access.

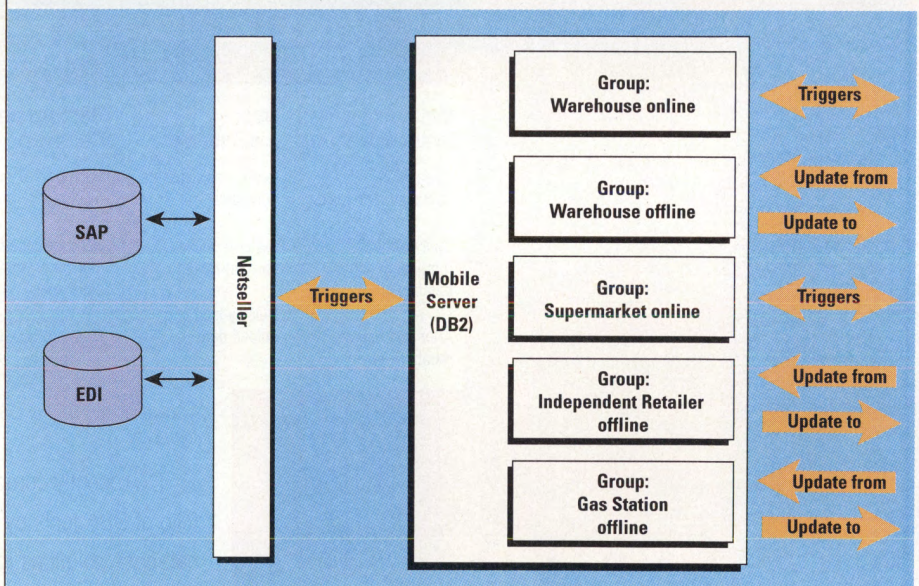


Figure 6: Data replication for different groups.

nology, you depend on the back-end services all the time. If there are any problems with these services, you'll run into trouble when your application won't work. People might have to revert to using paper and fax machines again — the exact situation we set out to remedy. Then there's the issue of administering all the devices. When you distribute up to 10,000 PDA devices, you want to make sure you have some kind of control over them.

But offline technology comes with its own set of challenges. I'll walk through some of the considerations for the ones I encountered when building the solution.

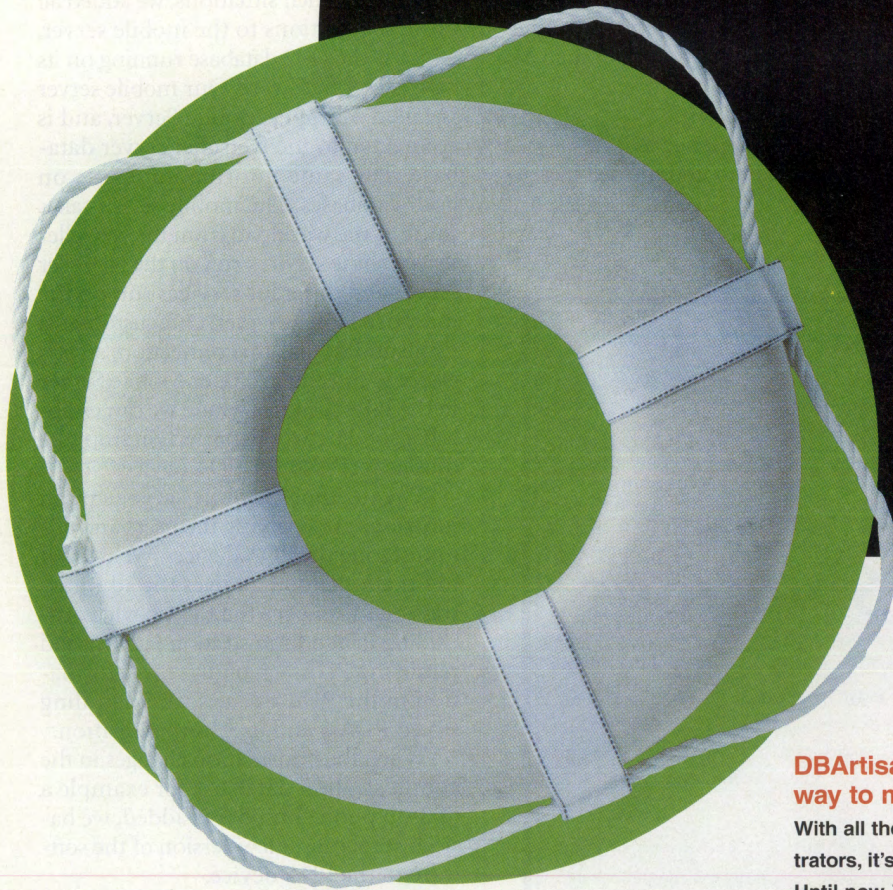
**Data replication.** Data replication in our solution is based on groups. We separated users into groups based on the

item information they need. The DB2 database handles the administration, and IBM Sync Server handles the actual replication. In the group management, we define whether the PDA belongs to the group that has an online access to our system or a group that has to make dial-in connections. If the PDA device belongs to the online access group, replication is launched every time certain information (for example, price or stock status) changes either on the PDA database or the mobile server (DB2) database. Some administration information (for example, if the PDA is marked stolen, we would overwrite the local database in the PDA) can also trigger the connection (see Figure 6).

If only dial-up connections are pos-



Dive into any database administration challenge. Without the risk.



**DBArtisan. The fastest, easiest—and safest—way to manage multiple databases.**

With all the responsibilities facing today's database administrators, it's nearly impossible to keep your head above water. Until now. The Embarcadero DBArtisan® product family for database administration gives you everything you need to master any database in no time at all. Plus, DBArtisan delivers the hottest features available—like migration and table alter—for faster, more accurate performance.

This means you no longer need to be an expert to look like one. Embarcadero takes this performance even farther with tools like DBArtisan Change Manager for migrating application schemas across heterogeneous database environments and our powerful PL/SQL Debugger and Profiler add-on modules

for debugging Oracle-programmable objects and detecting performance bottlenecks. All this makes DBArtisan the ideal administration solution for Oracle, DB2, Sybase, and SQL Server DBAs. Even if they're all you. Make the leap to better performance. Download your free trial version of DBArtisan today. [www.embarcadero.com/download/artisan.htm](http://www.embarcadero.com/download/artisan.htm)



Do more now in even less time with Embarcadero's new DBArtisan Job Scheduler:

[www.embarcadero.com/downloads/downloadjobscheduler.asp](http://www.embarcadero.com/downloads/downloadjobscheduler.asp)

"Products like DBArtisan change the way we do our jobs. Garden.com and e-business in general are functioning better because of them."

Andy Martin  
CTO, Gardens.com



do more now

©2000 Embarcadero Technologies, Inc. All rights reserved. Embarcadero Technologies and DBArtisan are registered trademarks of Embarcadero Technologies, Inc. All other products are the trademarks of their respective companies.



sible, the only way to handle true offline replication and avoid conflicts that could occur if old information remains on the PDA is to force all dial-up users to replicate the information from the back-end system before they can start a purchase order. The system administrator can determine the time gap that is allowed between replications.

This group replication feature is one of the main benefits of using Sync Server: You can constantly control what is repli-

cated and where. And DB2 and other IBM technology lets you concentrate on actual functionality issues rather than on coding problems.

**Administration.** With 10,000 PDA devices in circulation, administration and support can be a major challenge. You have to plan for "what ifs," such as:

- A device is stolen.
- A device is broken.
- New devices are needed.

You don't want to find yourself in a sit-

uation where all the users are calling your help desk just because they don't know how to install a new version of software or they don't know how to change the Remote Access Server (RAS) phone number.

To avoid such situations, we added all of these functions to the mobile server, which is the DB2 database running on its own server hardware. Our mobile server is called mZipper Mobile Server, and is connected to the Netseller server database. The same information exists on both databases. The mobile server takes some of the workload from the Netseller server; some services run on the Netseller server and different services run on the mZipper Mobile Server. Our customer is using an ISP dial-up connection, so it is easier to put the mZipper Mobile Server in the ISP's facilities where we can establish a good network connection from the mobile server to the PDA devices.

The mZipper Mobile server's main functions are verifying the users and administering the PDA devices. We've built in functionality that gives the administrator control over the PDA devices, including PDA administration (adding and removing PDAs, changing the information in the PDA devices, and handling stolen PDAs) and user authentication.

When the information changes in the administration database, for example a dial-up phone number is added, we basically replicate a new version of the software to the PDA device.

**Authentication and information security.** All the devices are authenticated via serial number. When a new device is added, the administrator adds the serial number and the group information to the system. Only the devices are authenticated, not the users. When the device is stolen, the administrator marks the device as stolen. Someone can still connect to the system with a stolen device, but the only transaction that can occur is that the local database inside the PDA will be overwritten. At the moment, data security software for the PDA devices is almost nonexistent.

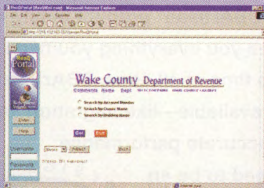
**Phone number administration.** Our customer has a nationwide deal with a telecom operator, which delivers the RAS service. All the PDA devices call one number, but the administrator has the power to list a primary and secondary phone number. There is also a service phone number for special purposes.

The need for two phone numbers is obvious. Most of the purchase orders come in at closing time. The cost for thousands of dedicated phone lines for devices that need the connection only

## Leading the Way in Web-Enabling and Rejuvenation Solutions



Take your AS/400 or Mainframe



To a GUI or the Web



and your Palm

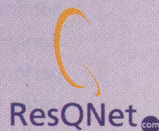
...in two days

ResQNet is a technology leader in the rapid deployment of rejuvenation, web-enabling, and self-service portal solutions. Our patented products provide graphical connectivity to host applications via the Internet, intranet, and wireless.

ResQNet's patented drag-and-drop products require no scripting or programming and have been proven on tens of thousands of desktops worldwide. Our ease of installation and ability to customize is unsurpassed.

After an exhaustive search for the best rejuvenation solution on the market, IBM chose ResQNet and has private-labeled the product for over two years. The ResQNet line offers limitless applications and is the perfect solution for verticals.

**FINANCIAL INSTITUTIONS • CALL CENTERS  
FEDERAL, STATE & LOCAL GOVERNMENTS  
INSURANCE COMPANIES • TELECOM  
MANUFACTURING • RESERVATIONS  
LIBRARIES AND COLLEGES • HEALTHCARE**



Visit [www.resqnet.com](http://www.resqnet.com) for more information, or call 212.537.4800.



for few minutes once or twice a day is enormous. The secondary phone number acts as an overflow mechanism. The service phone number is used when the two first numbers are not available (for example, if a line is being repaired). The administrator can give a phone number that is valid for only certain period of time. All the administration information is replicated to the PDA devices when the information changes.

Administration is easily forgotten when designing mobile systems. You can do without a decent mechanism for handling all the devices until the first system update, when you'll have to make all the changes manually. In our system, all the updates to the configuration and new software versions are made, managed, and delivered from one place.

**PDA OS restrictions.** We decided to use Palm OS instead of Windows CE mainly because of system reliability. It also easier to disguise the Palm desktop to look like a cash register extension than it is with Windows CE. Of course, the limitations of the Palm system mean you have only one process running, you can't save information to device in file formats such as CSV or XML, and you don't have any kind of directory structure.

In a way, the OS restrictions made our life easier. When we first thought about this project, we planned to take information out of Netseller, put it into an XML file, then send it to the PDA device. Imagine the impact of creating an XML file with 60,000 rows of item information and sending it to PDA devices. Then imagine the impact of several thousand PDA devices sending XML files to your back-end system. Because the Palm OS doesn't support XML, we had to design a better solution using the mobile server. It's not enough to just replace the paper system. You have to make sure that your application has a good response time.

**Life cycle.** When distributing thousands of PDA devices to its customers, our client wanted to make sure its investment would be safe. DB2 lets you deploy a solution to different platforms (such as Windows CE and EPOC) within days if necessary. And, the solution doesn't depend only on PDA devices or mobile technologies. We can offer it to basically every network and client device. Such a flexible solution lets us guarantee that the basic infrastructure can last for the next five to 10 years. And this client plans to recoup its investment within a year.

## THE PAPERLESS OFFICE AT LAST?

Mobile technologies let companies change the whole supply chain and help automate processes that previously were manual. The technology is here and it's mature. And, it may finally bring us closer to the elusive paperless office — you don't have to print everything out to get something done. The mobile world is here, if you don't want to join it, you can always use the off button. ●

**Marko Teerijä** is a senior consultant at Rissa Solutions in Finland. You can reach him at [marko.teerija@rissasolutions.com](mailto:marko.teerija@rissasolutions.com).

### RESOURCES:

#### Read more about DB2 Everyplace:

#### You Can Take it With You: Mobile Databases:

[www.db2mag.com/db\\_area/archives/2000/q3/mullins.shtml](http://www.db2mag.com/db_area/archives/2000/q3/mullins.shtml)

#### and

[ibm.com/software/data/everyplace](http://ibm.com/software/data/everyplace)

#### Read more about IBM Mobile Connect:

[ibm.com/pvc/products/mobile\\_connect/index.shtml](http://ibm.com/pvc/products/mobile_connect/index.shtml)

*"AlphaBlox's unique ability to analyze relational data and OLAP data on the same page, through the same user interface, will let us package a more complete portal-like view of our business, allowing management to make faster, more informed decisions."*

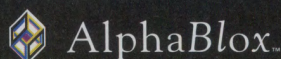
- Paul Hopkins, Project Leader for Equifax

## Nothin' But Net!

To survive and win in the Net economy, you need to identify and capitalize on your most profitable opportunities. With the right analytic infrastructure, your information assets become a strategic weapon. You make smarter, faster decisions and can respond instantly to rapidly changing market conditions.

Improve your bottom line with an integrated, Web-based analytic infrastructure that leverages AlphaBlox, IBM DB2 and the DB2 OLAP Server.

To learn more, call 1.888.BLOX.NOW or visit [www.alphablox.com](http://www.alphablox.com)



[www.alphablox.com](http://www.alphablox.com)



# **GIVING BUSINESS A SPORTING CHANCE**



TONY STONE IMAGES



## HOW THE BUSINESS INTELLIGENCE FUNCTIONS THAT POWERED OLYMPIC COVERAGE CAN WORK FOR YOU.

**W**atching sportscasts may be good for your business. Need convincing? Parents have long known that figuring out baseball statistics such as earned run averages and runs batted in helps kids learn math. Understanding the process behind the seemingly endless supply of relevant statistics sportscasters have at their fingertips can help your company learn about quickly delivering relevant statistics about your business. Aside from entertainment value, the analysis that goes into these sports statistics is the same kind your business could most likely benefit from. After all, the process of answering the question “Who are the top five athletes from any country to have won more than one gold medal?” isn’t significantly different from the process for answering “What are the top 25 bestsellers being ordered in the northeastern part of the United States?” Or “How many suits have we sold in California and Arizona during the last two quarters?”





Take, for example, the Olympic Games that took place in Sydney last fall. The built-in business intelligence features in DB2 Universal Database (UDB) that helped crunch athlete rankings and other statistics for live Web reports and television broadcasts at the Olympics are the same ones that help businesses crunch their numbers every day. These built-in functions let you perform ranking, statistical, and multidimensional analysis just by using SQL.

I'll explain how to do this using examples of the kinds of analysis DB2 UDB provided for the Olympics.

### WHO'S NUMBER ONE?

Athlete ranking is one of the most common functions used at the Olympics. After every heat or event, viewers are shown a list of who finished first, second, third, and so on. DB2 UDB provides two ranking functions for presenting this information: RANK and DENSERANK.

When you specify the RANK function, the rank of a row is defined as "1 plus the number of rows that strictly precede the row." So, if two rows are not distinct with respect to the ordering (for example, in the case of tied scores), there will be a gap in the sequential rank numbering.

If you specify the DENSERANK function, the rank of a row is defined as "1 plus the number of rows preceding the row that are distinct with respect to the ordering." Therefore, there will be no gaps in the sequential rank numbering.

Suppose for a single event (let's say diving) you want to rank the athletes from first place onward. Based on the information in Table 1, you could use this query to rank the athletes by score:

```
SELECT event, firstname, score,
       city,
       rank over (order by score desc)
       as Rank
FROM medals
ORDER BY Rank;
```

That query would return the results shown in Table 2: Notice that Dave and Gene are tied for second place. However, Rob is ranked fourth rather than third. Remember, the RANK function defines rank as 1 plus the number of rows that strictly precede the row. Because three rows precede Rob's row, he is listed as fourth.

In order for Rob to appear in third place, you would use the DENSERANK function:

```
SELECT event, firstname, score,
       city,
```

# DB2's built-in functions let you perform analysis using just SQL.

```
denserank over (order by score
desc)
as Rank
FROM medals
ORDER BY Rank;
```

Note that in the DENSERANK output shown in Table 3, Rob is correctly ranked third. I'll use DENSERANK for all examples that involve ranking.

### WHICH ATHLETES WON THE GOLD, SILVER, AND BRONZE?

Although the DENSERANK query I just showed you provides the correct ranking

of every athlete, it does not explicitly state who won the medals. You can display the medal-winning athletes by using the CASE expression, as follows:

```
SELECT event, firstname, score,
       city,
       denserank() over (order by score
desc) as Rank,
       CASE denserank() over (order by
score desc)
       WHEN 1 then 'Gold'
       WHEN 2 then 'Silver'
       WHEN 3 then 'Bronze'
END as Medal
```

Event	Athlete	City	Score
Diving	Dan	Sydney	10.0
Diving	Dave	Sydney	9.9
Diving	Rob	Sydney	9.8
Diving	Grant	Melbourne	9.7
Diving	Gene	Melbourne	9.9
Diving	Leon	Melbourne	7.0
Diving	Ed	Perth	9.0
Diving	Bob	Perth	9.5

Table 1: Unranked scores in a diving event.

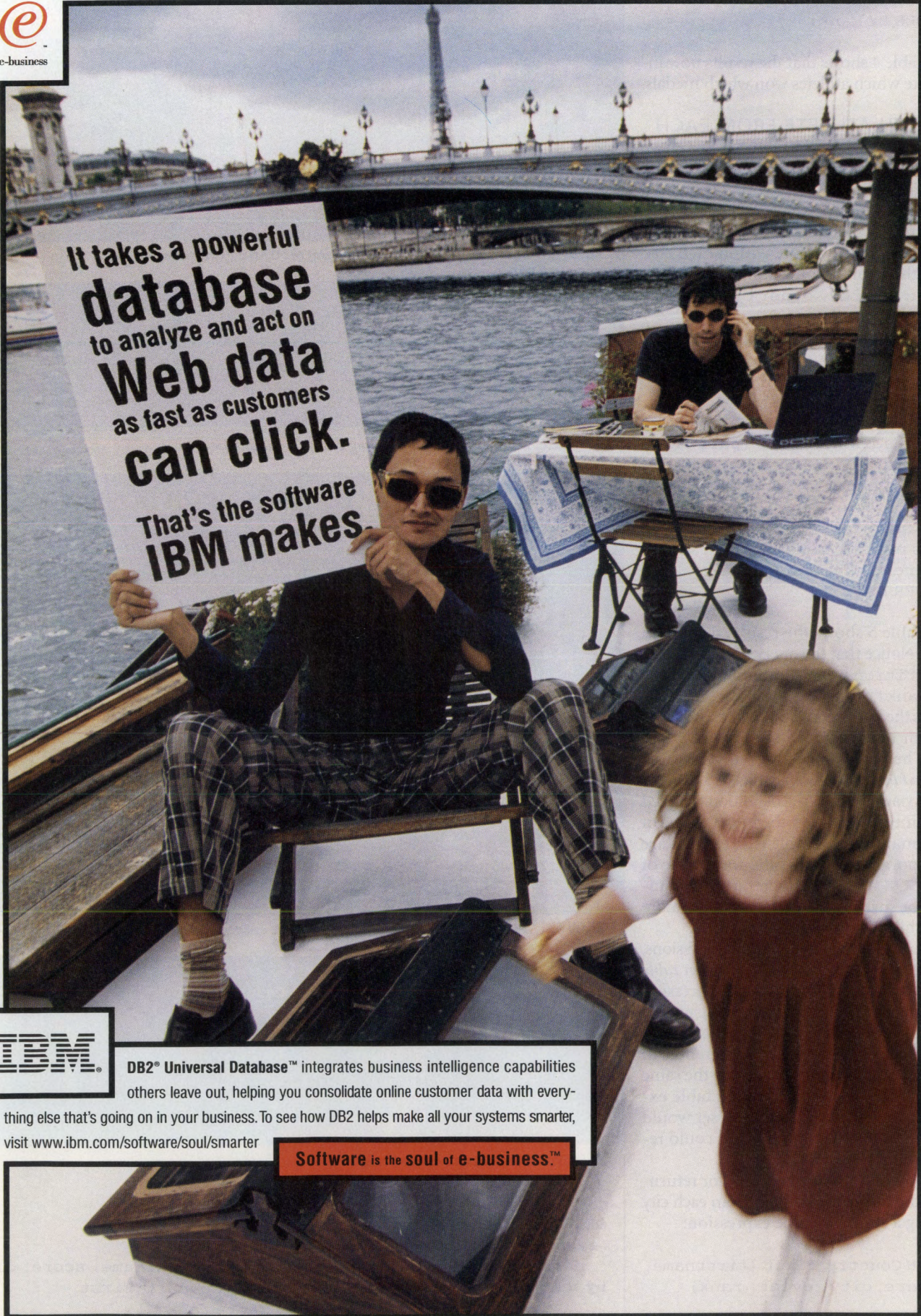
Event	Athlete	Score	City	Rank
Diving	Dan	10.0	Sydney	1
Diving	Dave	9.9	Sydney	2
Diving	Gene	9.9	Melbourne	2
Diving	Rob	9.8	Sydney	4
Diving	Grant	9.7	Melbourne	5
Diving	Bob	9.5	Perth	6
Diving	Ed	9.0	Perth	7
Diving	Leon	7.0	Melbourne	8

Table 2: Athletes' scores ranked using the RANK function.

Event	Athlete	Score	City	Rank
Diving	Dan	10.0	Sydney	1
Diving	Dave	9.9	Sydney	2
Diving	Gene	9.9	Melbourne	2
Diving	Rob	9.8	Sydney	3
Diving	Grant	9.7	Melbourne	4
Diving	Bob	9.5	Perth	5
Diving	Ed	9.0	Perth	6
Diving	Leon	7.0	Melbourne	7

Table 3: Athletes' scores ranked using the DENSERANK function.





It takes a powerful  
**database**  
to analyze and act on  
**Web data**  
as fast as customers  
**can click.**  
That's the software  
**IBM makes.**



**DB2® Universal Database™** integrates business intelligence capabilities others leave out, helping you consolidate online customer data with everything else that's going on in your business. To see how DB2 helps make all your systems smarter, visit [www.ibm.com/software/soul/smarter](http://www.ibm.com/software/soul/smarter)

**Software is the soul of e-business.™**

IBM, DB2, DB2 Universal Database, the e-business logo and Software is the soul of e-business, are trademarks of International Business Machines Corporation in the United States and/or other countries. © 2000 IBM Corporation. All rights reserved.



```
FROM medals
ORDER BY Rank;
```

Table 4 shows that the results now indicate which athletes won which medals.

### WHICH ATHLETE FROM EACH CITY HAS THE HIGHEST SCORE?

So far we have been returning all rows of the table. But suppose you only wanted to know which person from each city has the highest score? To find out, you would need to order by score within each city. In other words, you would use `partition by city, order by score`:

```
Select firstname, score, city,
rank from
(
SELECT event, firstname, score,
city,
denserank() over (partition
by city
order by score desc) as
Rank
FROM medals
) as nested_medals
Where rank = 1
ORDER BY Rank;
```

Table 5 shows the results from each city. Notice that the query uses the clause `partition by` in addition to the `denserank()` function. `Partition by` defines the partition within which the ranking or numbering will be applied.

The query I just showed you uses a *nested table expression*, which is a temporary view in which the definition is nested (defined directly) in the `FROM` clause of the main query. Table expressions are temporary and only valid for the life of the SQL statement; therefore, they cannot be shared. However, they allow more flexibility than views.

In addition to nested table expressions, DB2 UDB also supports *common table expressions*. A common table expression is a named result table defined using the `WITH` keyword prior to the beginning of a `fullselect`. Repeated references to a common table expression use the same result set. If you used nested table expressions or views, the result set would be regenerated each time and could return different results.

You could write the query for returning the top-scoring athlete from each city using a common table expression:

```
WITH Country_First (firstname,
score, city, event, rank)
AS
```

Event	Firstname	Score	City	Rank	Medal
Diving	Dan	10.0	Sydney	1	Gold
Diving	Gene	9.9	Melbourne	2	Silver
Diving	Dave	9.9	Sydney	2	Silver
Diving	Rob	9.8	Sydney	3	Bronze
Diving	Grant	9.7	Melbourne	4	-----
Diving	Bob	9.5	Perth	5	-----
Diving	Ed	9.0	Perth	6	-----
Diving	Leon	7.0	Melbourne	7	-----

Table 4: *Displaying medal winners using the CASE expression.*

Firstname	Score	City	Rank
Bob	9.5	Perth	1
Dan	10.0	Sydney	1
Gene	9.9	Melbourne	1

Table 5: *Displaying the top-scoring athlete from each city.*

Event	Athlete	City	Score
Diving	Dan	Sydney	10.0
Diving	Dave	Sydney	9.8
Diving	Rob	Sydney	9.8
Diving	Grant	Melbourne	9.7
Diving	Gene	Melbourne	9.9
Diving	Leon	Melbourne	7.0
Diving	Ed	Perth	9.0
Diving	Bob	Perth	9.5
Gymnastics	Jack	Sydney	8.0
Gymnastics	Chris	Sydney	9.8
Gymnastics	Tim	Sydney	9.0
Gymnastics	Bill	Melbourne	10.0
Gymnastics	George	Melbourne	9.9
Gymnastics	Phil	Melbourne	7.0
Gymnastics	Terry	Perth	9.8
Gymnastics	Gary	Perth	9.5

Table 6: *Athlete information from two separate events.*

DAY	CITY	Count
Mon	Sydney	3
Mon	Melbourne	1
Tue	Perth	1
Tue	Sydney	1
Tue	Melbourne	2

Table 7: *Total medals won by each city.*

```
(
SELECT firstname, score, city,
event,
denserank() over (partition
by city
order by score desc) as
Rank
FROM medals
)
Select firstname, score, city
from Country_First
Where rank = 1
ORDER BY Rank;
```



DAY	CITY	COUNT	
Mon	Sydney	3	
Mon	Melbourne	1	
Mon	--	4	Subtotal of medals for Monday
Tue	Perth	1	
Tue	Sydney	1	
Tue	Melbourne	2	
Tue	--	4	Subtotal of medals for Tuesday
--	--	8	Total number of medals awarded

Table 8: Total medals won by each city and total medals awarded each day.

DAY	CITY	COUNT	
Mon	Sydney	3	
Mon	Melbourne	1	
Tue	Perth	1	
Tue	Sydney	1	
Tue	Melbourne	2	
--	Perth	1	Total medals for Perth
--	Sydney	4	Total medals for Sydney
--	Melbourne	3	Total medals for Melbourne
--	--	8	Grand total medal count

Table 9: Total medals by city by day and overall total.

```

WITH MEDAL_INFO (day, score, city, event, medal)
AS
(
SELECT day, score, city, event,
CASE denserank() over (partition by event order by
score desc)
WHEN 1 then 'Gold'
WHEN 2 then 'Silver'
WHEN 3 then 'Bronze'
END as Medal
FROM medals)

Select day, city, medal, count (medal) as Count
from medal_info where medal is not null
group by rollup (day, (city, medal))
order by day, city;

Results:
DAY CITY MEDAL COUNT
Mon Sydney Bronze 2
Mon Sydney Gold 1
Mon Melbourne Silver 1
Mon -- -- 4 Total awarded Monday
Tue Perth Bronze 1
Tue Sydney Bronze 1
Tue Melbourne Gold 1
Tue Melbourne Silver 1
Tue -- -- 4 Total awarded Tuesday
-- -- -- 8

```

Listing 1: Total gold, silver, and bronze medals won per city and the medal total for each day.

## BY CITY, BY EVENT, BY DAY

So far, I've only shown you queries that involved a single sport. Suppose you wanted to view multiple sports by various categories? For example, suppose you wanted to know the total number of medals that each city won each day, and the total number of medals awarded so far? Or, how many gold, silver, and bronze medals were awarded each day to each city, and the total number of all medals awarded each day to each city?

Notice that these questions are asking for information from various categories (medals, city, and day). These categories are commonly called *dimensions*. Dimensions let you organize data in way that makes answering the kinds of questions I mentioned easier. For example, using the dimensions medals, city, and day, the resulting cube depicts the intersection of three axes: Product (medals), Geography (city), and Time (day).

The phrase online analytic processing (OLAP) describes the kind of complex queries mentioned. DB2 UDB's built-in OLAP capability supports such queries by extending the GROUP BY clause to support "super groups." One type of super group is a ROLLUP group, a result set that contains subtotal and "overall total" rows in addition to the regular grouped rows. Another type of super group is a CUBE group, a result set that contains cross-tabulation rows in addition to all the rows that would be in a ROLLUP group for the same columns.

Table 6 contains information about diving and gymnastics events. To answer the question "What are the total number of medals each city has won?" you would use this query:

```

WITH MEDAL_INFO (day, score,
city, event, medal)
AS
(
SELECT day, score, city, event,
CASE denserank() over (parti-
tion by event order by score
desc)
WHEN 1 then 'Gold'
WHEN 2 then 'Silver'
WHEN 3 then 'Bronze'
END as Medal
FROM medals)

Select day, city, count (medal)
from medal_info where medal is
not null
group by (day, city)
order by day, city;

```

Table 7 lists the total number of



medals won by each city. But suppose you wanted to answer the question “What is the total number of medals each city has won each day, and what is the total number of medals awarded so far?” To include these subtotals (by day) as well as the final total, you would use the ROLLUP function:

```
WITH MEDAL_INFO (day, score,
city, event, medal)
AS
(
SELECT day, score, city, event,
CASE denserank() over (parti-
tion by event order by score desc)
WHEN 1 then 'Gold'
WHEN 2 then 'Silver'
WHEN 3 then 'Bronze'
END as Medal
FROM medals)
```

```
Select day, city, count (medal) as
COUNT
from medal_info where medal is
not null
group by ROLLUP (day, city)
order by day, city;
```

In addition to the rows returned in the first query, notice the additional rows returned in Table 8 (see page 43).

Now, watch what happens if I use the same query except with ROLLUP(city,day) instead of ROLLUP(day, city):

```
WITH MEDAL_INFO (day, score,
city, event, medal)
AS
(
SELECT day, score, city, event,
CASE denserank() over (parti-
tion by event order by score
desc)
WHEN 1 then 'Gold'
WHEN 2 then 'Silver'
WHEN 3 then 'Bronze'
END as Medal
FROM medals)
```

```
Select day, city, count (medal) as
COUNT
from medal_info where medal is not
null
group by ROLLUP (city, day)
order by day, city;
```

Notice that the results in Table 9 (see page 43) show totals by city versus totals by day.

You can perform many variations of these queries depending on your information needs. To find out how many gold, silver, and bronze medals athletes

```
WITH MEDAL_INFO (day, score, city, event, medal)
AS
(
SELECT day, score, city, event,
CASE denserank() over (partition by event order by
score desc)
WHEN 1 then 'Gold'
WHEN 2 then 'Silver'
WHEN 3 then 'Bronze'
END as Medal
FROM medals)
```

```
Select day, city, medal, count (medal) as Count
from medal_info where medal is not null
group by cube (day, (city, medal))
order by day, city;
```

Results:

DAY	CITY	MEDAL	COUNT
Mon	Sydney	Bronze	2
Mon	Sydney	Gold	1
Mon	Melbourne	Silver	1
Mon	--	--	4
Tue	Perth	Bronze	1
Tue	Sydney	Bronze	1
Tue	Melbourne	Gold	1
Tue	Melbourne	Silver	1
Tue	--	--	4
--	Perth	Bronze	1
--	Sydney	Bronze	3
--	Sydney	Gold	1
--	Melbourne	Gold	1
--	Melbourne	Silver	2
--	--	--	8

**Listing 2:** The number of gold, silver, and bronze medals each city won, the total number of medals awarded each day, and the total number of medals each city has won overall.

from each city won and the total number of medals awarded each day, see Listing 1 (page 43). To find out how many gold, silver, and bronze medals each city won, the total number of medals awarded each day, and the total number of medals each city has won overall, see Listing 2.

### THE COMPETITIVE EDGE

Business intelligence means using your data assets to make better business decisions. In this article I've shown just a few of the many different ways you might want to look at sports results. Imagine how many more ways you'd want to look at your business data. DB2 UDB's built-in business intelligence functions give you easier access to your data assets. ●

**Dan Gibson** works on development of DB2 UDB for Intel and Unix platforms at the IBM Toronto Lab. An 11-year IBM veteran, he has participated in DB2 UDB projects including data-

base porting efforts, customer benchmarks, and skills transfer and has supported the Olympics for the past seven years. You can reach him at [dgibson@ca.ibm.com](mailto:dgibson@ca.ibm.com).

### RESOURCES:

For more information about DB2 at the Olympics, check out these articles online:

#### An Olympics for the Internet Age

[www.db2mag.com/db\\_area/archives/2000/q4/olympics.shtml](http://www.db2mag.com/db_area/archives/2000/q4/olympics.shtml)

#### Let the Games Begin

[www.db2mag.com/db\\_area/archives/2000/q3/nelson.shtml](http://www.db2mag.com/db_area/archives/2000/q3/nelson.shtml)

#### DB2 Knows Baseball

[www.db2mag.com](http://www.db2mag.com)

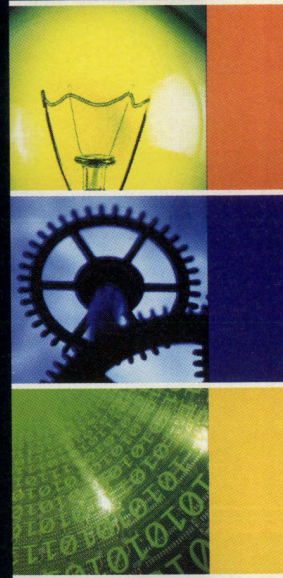
#### For more information about DB2 UDB's built-in OLAP functions, go to

[www.ibm.com/software/data/db2/db2olap](http://www.ibm.com/software/data/db2/db2olap)



# IDUG® 2001 North America

## DB2 Powered by You



may 13-17, 2001

orlando world center marriott

orlando, florida USA

### INTERNATIONAL DB2 USERS GROUP



**IDUG Headquarters**  
401 North Michigan Avenue  
Chicago, Illinois 60611-4267 USA

+1.312.644.6610 phone

+1.312.321.5158 fax

idug@idug.org email

Join the International DB2 Users Group for IDUG 2001 — North America. Build on your knowledge and enhance your ability to maximize the value of the DB2 family of products.

#### The conference provides:

- access to four full days of high quality education, presented by industry leaders
- the opportunity to learn the latest news on DB2 features and advances direct from IBM's top developers and executives
- an exhibition where you can learn about the latest DB2 products and services from today's leading vendors, in one place, at one time
- optional one-day seminars (additional fee required) on Sunday, May 13.

**Register today. Visit our Web site at [www.idug.org](http://www.idug.org).**



# TOP TEN TIPS







# Top 10 Performance Tips

*Database performance woes? Don't panic. This database first-aid plan will help you find and repair almost any performance problem.*

Every few weeks or so, we get performance calls from DBAs in distress. “Our Web site is crawling,” they bemoan. “We’re losing customers, and the situation is dire. Can you help?” To answer these calls, I’ve developed an analytics process for my consulting company that lets us rapidly find the cause of performance problems and develop remedial actions and tuning suggestions. Rarely do these callers inquire about fees or costs — they’re only interested in stopping the bleeding. When a DB2 or e-business application isn’t performing up to expectations, the entire organization and financial bottom line measurably suffers.



To help DB2 DBAs avoid performance fires and achieve high performance on their own, I've summarized a troubleshooting process for our clients, customers, and fellow DB2 professionals. The top 10 performance tips for e-business OLTP applications in DB2 UDB for Unix, Windows, and OS/2 environments are:

## 10. MONITOR SWITCHES

Make sure the monitor switches are turned on. If they aren't, you won't have access to the performance information you need. To turn the monitor switches on, issue the command:

```
db2 "update monitor switches using
lock ON sort ON bufferpool ON uow ON
table ON statement ON"
```

## 9. AGENTS

Make sure there are enough DB2 agents to handle the workload. To find out, issue the command

```
db2 "get snapshot for database
manager"
```

and look for the following lines:

```
High water mark for agents
registered = 7
High water mark for agents
waiting for a token = 0
Agents registered = 7
Agents waiting for a token = 0
Idle agents = 5
Agents assigned from pool = 158
Agents created from empty
pool = 7
Agents stolen from another
application = 0
High water mark for
coordinating agents = 7
Max agents overflow = 0
```

If you find either Agents waiting for a token or Agents stolen from another application, increase the number of agents available to the database manager (MAXAGENTS and/or MAX\_COORDAGENTS as appropriate).

## 8. MAXIMUM OPEN FILES

DB2 tries to be a good citizen within the constraints of operating system resources. One of its "good citizen" acts is putting a ceiling, or upper limit, on the maximum number of files open at any one time. The MAXFILOP database configuration parameter stipulates the maximum number of files that DB2 can have open concurrently. After it reaches that

point, DB2 will start closing and opening its tablespace files (including raw devices). Opening and closing files slows SQL response times and burns CPU cycles. To find out if DB2 is closing files, issue the command

```
db2 "get snapshot for database on
DBNAME"
```

and look for the line that reads:

```
Database files closed = 0
```

If files are being closed, increase the value of MAXFILOP until the opening and closing stops. Use the command:

```
db2 "update db cfg for DBNAME using
MAXFILOP N"
```

# A high number of overflows probably means you need to reorganize the table.

## 7. LOCKS

The default value for LOCKTIMEOUT is -1, which means that there will be no lock timeouts — a situation that can be catastrophic for OLTP applications. Nevertheless, I all too frequently find many DB2 users with LOCKTIMEOUT = -1. Set LOCKTIMEOUT to a very short value, such as 10 or 15 seconds. Waiting on locks for extended periods of time can have an avalanche effect on locks.

First, check the value of LOCKTIMEOUT with this command

```
db2 "get db cfg for DBNAME"
```

and look for the line containing this text:

```
Lock timeout (sec) (LOCKTIMEOUT) = -1
```

If the value is -1, consider changing it to 15 seconds by using the following command (be sure to consult with the application developers or your vendor first to make sure the application is prepared to handle lock timeouts):

```
db2 "update db cfg for DBNAME using
LOCKTIMEOUT 15"
```

You should also monitor the number of lock waits, lock wait time, and amount of lock list memory in use. Issue the command:

```
db2 "get snapshot for database on
DBNAME"
```

Look for the following lines:

```
Locks held currently = 0
Lock waits = 0
Time database waited on
locks (ms) = 0
Lock list memory in
use (Bytes) = 576
Deadlocks detected = 0
Lock escalations = 0
Exclusive lock escalations = 0
Agents currently waiting
on locks = 0
Lock Timeouts = 0
```

If the Lock list memory in use (Bytes) exceeds 50 percent of the de-

finer LOCKLIST size, then increase the number of 4K pages in the LOCKLIST database configuration.

## 6. TEMPORARY TABLESPACES

Temporary tablespaces should have at least three containers on three different disk drives in order to help DB2 perform parallel I/O and improve performance for sorts, hash joins, and other database activities that use TEMPSPACE.

To find out how many containers your temporary tablespace has, issue the command:

```
db2 "list tablespaces show detail"
```

Look for the TEMPSPACE tablespace definition similar to this example:

```
Tablespace ID = 1
Name = TEMPSPACE1
Type = System
managed
space
Contents = Temporary
data
State = 0x0000
Detailed explanation:
Normal
Total pages = 1
Useable pages = 1
Used pages = 1
Free pages = Not
```



	applicable
High water mark (pages)	= Not applicable
Page size (bytes)	= 4096
Extent size (pages)	= 32
Prefetch size (pages)	= 96
Number of containers	= 3

Notice that Number of containers has the value 3, and that Prefetch size is three times Extent size. For best parallel I/O performance, it is important for Prefetch size to be a multiple of Extent size. The multiple should be equal to the number of containers.

To find the definitions for the containers, issue the command:

```
db2 "list tablespace containers for 1 show detail"
```

The 1 refers to tablespace ID #1, which is TEMPSPACE1 in the example just given.

## 5. SORT MEMORY

OLTP applications should not be performing large sorts. They are too costly in terms of CPU, I/O, and elapsed time and will slow down any OLTP application. Therefore, the default SORTHEAP size of 256 4K pages (1MB) should be more than adequate. You should also know the number of sort overflows and the number of sorts per transaction.

Issue the command

```
Db2 "get snapshot for database on DBNAME"
```

and look for the following lines:

```
Total sort heap allocated = 0
Total sorts = 1
Total sort time (ms) = 8
Sort overflows = 0
Active sorts = 0
Commit statements attempted = 3
Rollback statements attempted = 0
Let transactions = Commit statements attempted + Rollback statements attempted
Let SortsPerTX= Total sorts / transactions
Let PercentSortOverflows = Sort overflows * 100 / Total sorts
```

If PercentSortOverflows ((Sort overflows \* 100) / Total sorts) is greater than 3 percent, there may be

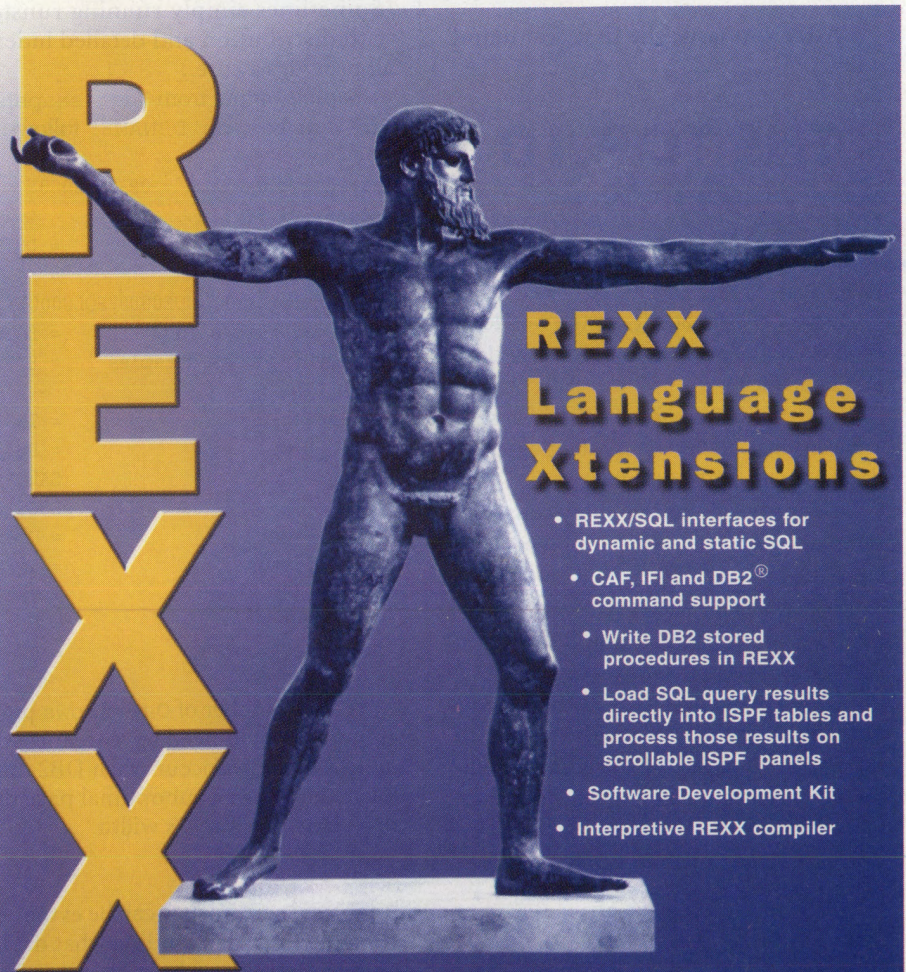
serious and unexpected sort problems in the application SQL. Because the very presence of overflows indicates that large sorts are occurring, finding zero sort overflows, or at least a percentage less than one, would be ideal.

If excessive sort overflows are present, the "band aid" solution is to increase the size of SORTHEAP. However, doing so only masks the real performance problem. Instead, you should identify the SQL that is causing the sorts and change the

SQL, indexes, or clustering to avoid or reduce the sort cost.

If SortsPerTX is greater than 5 (as a rule of thumb), the number of sorts per transaction may be high. Some application transactions perform dozens of small composite sorts (which do not overflow and have very short durations), but consume excessive CPU. When SortsPerTX is high, my experience indicates that these machines are typically CPU bound. Identifying the SQL that is

# REXX



## REXX Language Xtensions

- REXX/SQL interfaces for dynamic and static SQL
- CAF, IFI and DB2<sup>®</sup> command support
- Write DB2 stored procedures in REXX
- Load SQL query results directly into ISPF tables and process those results on scrollable ISPF panels
- Software Development Kit
- Interpretive REXX compiler

**Unsurpassed productivity with REXX and RLX!**

**Develop applications, tools, even MVS system exits with incredible speed and ease.**

**Prototype any OS/390 requirement.**

**Let us show you how.**

**Relational  
Architects  
International**

800 776-0771  
201 420-0400 (Tel)  
201 420-4080 (Fax)

[www.relarc.com](http://www.relarc.com)

DB2, ISPF, and OS/390 are registered trademarks of IBM Corp.



causing the sorts and improving the access plans (via indexes, clustering, or SQL changes) is paramount to improving transaction throughput rates.

#### 4. TABLE ACCESS

For each table, identify how many rows DB2 is reading for each transaction. You must issue two commands:

1. db2 "get snapshot for database on DBNAME"
2. db2 "get snapshot for tables on DBNAME"

After you issue the first command,

## THE TOP 10 AT A GLANCE

1. Use sufficient agents for the workload.
2. Do not allow DB2 to needlessly close and open files.
3. Do not allow extended lock waits.
4. Ensure parallel I/O capabilities to the database `TEMPSPACE` tablespace.
5. Manage DB2 sort memory conservatively and don't mask sort problems with large `SORTHEAPS`.
6. Analyze table access activity and identify tables with unusually high rows read per transaction or overflow counts.
7. Analyze the performance characteristics of each tablespace, and seek to improve the performance of the tablespaces with the slowest read times, longest write times, highest physical I/O read rates, worst hit ratios, and access attributes that are inconsistent with expectations.
8. Create multiple buffer pools, and make purposeful assignments of tablespaces to buffer pools such that access attributes are shared.
9. Examine DB2 UDB SQL Event Monitor information to discover which SQL statements are consuming the largest proportions of computing resources, and take corrective actions.
10. Reevaluate configuration and physical design settings once high cost SQL is eliminated.

determine how many transactions have occurred (by taking the sum of `Commit statements attempted` plus `Rollback statements attempted` — see tip 3).

After issuing the second command, divide the number of rows read by the number of transactions (`RowsPerTX`). OLTP applications should typically read one to 20 rows from each table per transaction. If you discover that hundreds or thousands of rows are being read for each transaction, scans are taking place and indexes may need to be created. (Sometimes simply running `runstats` with distribution and detailed indexes all provides a cure.)

Sample output from "get snapshot for tables on DBNAME" follows:

```
Snapshot timestamp = 09-25-2000
14:47:09.970811
Database name           = DGIDB
Database path           =
/fs/inst1/inst1/NODE0000/SQL00001/
Input database alias    = DGIDB
Number of accessed tables = 8
Table List
Table Schema           = INST1
Table Name             = DGI_
                      SALES_
                      LOGS_TB
Table Type              = User
Rows Written           = 0
Rows Read              = 98857
Overflows              = 0
Page Reorgs           = 0
```

A high number of `Overflows` probably means you need to reorganize the table. `Overflows` occur when DB2 must locate a row on a suboptimal page due to a change in a row's width.

#### 3. TABLESPACE ANALYSIS

A tablespace snapshot can be extremely valuable to understanding what data is being accessed and how. To get one, issue the command:

```
db2 "get snapshot for tablespaces on
DBNAME"
```

For each tablespace, answer the following questions:

- What is the average read time (ms)?
- What is the average write time (ms)?
- What percentage of the physical I/O is asynchronous (prefetched) vs. synchronous (random)?
- What are the buffer pool hit ratios for each tablespace?
- How many physical pages are being read each minute?

• How many physical and logical pages are being read for each transaction?

For all tablespaces, answer the following questions:

• Which tablespaces have the slowest read and write times? Why? Containers on slow disks? Are container sizes unequal?

• Are the access attributes, asynchronous versus synchronous access, consistent with expectations? Randomly read tables should have randomly read tablespaces, meaning high synchronous read percentages, usually higher buffer pool hit ratios, and lower physical I/O rates.

For each tablespace, make sure that the prefetch size is equal to the extent size multiplied by the number of containers. Issue the command:

```
db2 "list tablespaces show detail"
```

The prefetch size can be altered for a given tablespace if necessary. Container definitions can be checked by using the command

```
db2 "list tablespace containers for
N show detail"
```

in which N is the tablespace ID number.

#### 2. BUFFER POOL OPTIMIZATION

All too often I find DB2 UDB sites where the machines have 2, 4, or 8GB of memory, yet the DB2 database has one buffer pool, `IBMDEFAULTBP`, which is only 16MB in size!

If this is the case at your site, create a buffer pool for the `SYSCATSPACE` catalog tablespace, one for the `TEMPSPACE` tablespace, and at least two more buffer pools: `BP_RAND` and `BP_SEQ`. Tablespaces that are accessed randomly should be assigned to a buffer pool with random objectives, `BP_RAND`. Tablespaces that are accessed sequentially (with asynchronous prefetch I/O) should be assigned to a buffer pool with sequential objectives, `BP_SEQ`. You can create additional buffer pools depending on performance objectives for certain transactions; for example, you could make a buffer pool large enough to store an entire "hot," or very frequently accessed, table. When large tables are involved, some DB2 users have great success placing the indexes for important tables into an index, `BP_IX`, buffer pool.

Buffer pools that are too small result in excessive, unnecessary, physical I/O. Buffer pools that are too large put a system at risk for operating system paging



and consume unnecessary CPU cycles managing the overallocated memory. Somewhere between “too small” and “too large” lies the size that is just right. The right size exists where the point of diminishing returns is reached. If you’re not using a tool to automate the diminishing returns analysis, you should scientifically test buffer pool performance (hit ratios, I/O times, physical I/O read rates) at incremental sizes until an optimum size is reached. Because businesses

## OLTP applications should not be performing large sorts.

constantly change and grow, the “optimum size” decision should be reevaluated periodically.

### 1. SQL COST ANALYSIS

One bad SQL statement can ruin your whole day. Time and time again I’ve seen a single, relatively simple SQL statement bring a finely tuned database and machine to its knees. For many of these statements, there isn’t a DB2 UDB configuration parameter under the sun (or in the doc) that can make right the high cost of an errant SQL statement.

Making matters worse, the DBA’s hands are frequently tied: You can’t change the SQL can because it’s provided by an application vendor (such as SAP, PeopleSoft, or Siebel). This leaves the DBA three courses of action:

1. Change or add indexes
2. Change clustering
3. Change catalog statistics.

What’s more, today’s robust applications are made up of hundreds or thousands of different SQL statements. These statements are executed at varying rates of frequency depending on application functionality and the business needs du jour. A SQL statement’s true cost is the resource cost to execute it once multiplied by the number of times it is executed.

The monumental task that confronts each DBA is the challenge of identifying the SQL statements with the highest “true cost,” and working to reduce the costs of these statements.

You can find out the resource cost to execute a SQL statement once from native DB2 Explain utilities, a number of tools from third-party vendors, or the DB2 UDB SQL Event Monitor data. But the frequency of statement execution can only be learned through careful and

time-consuming analysis of DB2 UDB SQL Event Monitor data.

In researching problem SQL statements, the standard procedure used by DBAs is:

1. Create an SQL Event Monitor, write to file:

```
$> db2 "create event monitor SQLCOST
for statements write to ..."
```

2. Activate the event monitor (be sure

ample free disk space is available):

```
$> db2 "set event monitor SQLCOST
state = 1"
```

3. Let the application run.

4. Deactivate the event monitor:

```
$> db2 "set event monitor SQLCOST
state = 0"
```

5. Use the DB2-supplied db2evmon tool to format the raw SQL Event Monitor data (hundreds of megabytes of free disk space may be required depending on SQL throughput rates):

```
$> db2evmon -db DBNAME -evm SQLCOST
> sqltrace.txt
```

6. Browse through the formatted file scanning for unusually large cost numbers, a time-consuming process:

```
$> more sqltrace.txt
```

7. Undertake a more complete analysis of the formatted file that attempts to identify unique statements (independent of literal values), each unique statement’s frequency (how many times it occurred), and the aggregate of its total CPU, sort, and other resource costs. Such a thorough analysis could take a week or more on just a 30-minute sample of application SQL activity.

To reduce the time it takes to identify SQL statements with high costs, you can consider many sources of available information:

• From Tip 4, be sure to compute the number of rows read from each table per transaction. If the resulting number seems high, the DBA may be able to identify problem statements by search-

ing the SQL Event Monitor formatted output for the table name in question (this will narrow the search, and possibly save some time).

• From Tip 3, be sure to compute the asynchronous read percentage and physical I/O read rates for each tablespace. If a tablespace has a very high asynchronous read percentage and way above average physical I/O read rates, one or more of the tables in the tablespace is probably being scanned. Query the catalog and find out which tables are assigned to the suspect tablespaces (one table per tablespace provides the best performance instrumentation), then search the SQL Event Monitor formatted output for the tables. This, too, may help narrow the search for costly SQL statements.

• Try to look at DB2 Explain information for each of the SQL statements being executed by the application. However, I’ve found that high frequency, lower-cost statements often rob a machine of its capacity and ability to provide desired performance.

• If analysis time is short and maximum performance is critical, consider vendor tools that can quickly automate the process of identifying resource intensive SQL statements.

### STAYING IN TUNE

Optimum performance requires not only eliminating high cost SQL statements, but also making sure that appropriate physical infrastructures are in place. Peak performance results when all the tuning knobs are set just right, memory is allocated to pools and heaps effectively, and I/O is evenly balanced across disks. Although it takes time to measure and make adjustments, the DBA who performs these 10 suggestions will be very successful at satisfying internal and external DB2 customers. As e-businesses change and grow, even the best-administered database will need regular fine-tuning. The DBA’s job is never done! ●

**Scott Hayes** is president of Database-Guys Inc., a performance tools and consulting company specializing in DB2 UDB on Unix platforms. He is also a member of the IBM DB2 Gold Consultant Group and a DB2 UDB Advanced Certified Technical Expert. Scott has worked with DB2/AIX since V1, and his diverse experiences have taught him a number of DB2 UDB best practices. A frequent speaker at IDUG and DB2 technical conferences, Scott has published a number of articles in DB2-related magazines. For more information, visit [www.database-guys.com](http://www.database-guys.com) or write [scott@database-guys.com](mailto:scott@database-guys.com).



# JDBC Explained

*Creating DB2 applications for the JDBC API.*

**T**ODAY'S DEVELOPERS LOOK for solutions that are scalable, secure, and flexible enough to keep pace with the ever-transforming Internet. That's why so many developers use Java. Java's flexibility gives you several ways to connect and interact with databases. In this column, I'll provide an overview of one of those methods: developing Java applications and applets for DB2 that use the Java database connectivity (JDBC) API.

## UNDERSTANDING JDBC

Both Microsoft's Open Database Connectivity (ODBC) and JDBC are based on the X/Open SQL call-level interface specifications. ODBC has one major flaw: It relies on the multiple use of void pointers and other C features that are not natural in Java.

A vendor-neutral dynamic SQL inter-

face, JDBC lets you access virtually any tabular data source from the standardized Java programming language. The JDBC program is similar to the DB2 CLI in that you do not have to precompile or bind a JDBC program. JDBC provides cross-DBMS connectivity to a wide range of SQL databases and provides access to other tabular data sources, such as spreadsheets or flat files. JDBC applications provide increased portability because of their vendor-neutral standard.

The JDBC interface lets you write code for your applications in a heterogeneous environment.

JDBC consists of two layers (see Figure 1, page 54). The first layer is called the JDBC API, and its job is to communicate SQL commands to the second layer, called JDBC manager driver API. The manager should communicate with the various third-party drivers that actually

connect to the database and return the information from the query or perform the action specified by the query.

JDBC is especially useful in developing applications that run across the Internet. You can use Java to connect via the JDBC API to a database using a standard network connection. You can also create JDBC stored procedures that live on the server and can be executed with simple programming calls and return information to the client via the Internet or any other network connection.

Your Java code passes SQL statements as function calls to the DB2 JDBC driver. The driver then handles the JDBC API calls from your client Java code.

Because of Java's portability, you only have to worry about the client's ability to handle Java applications. Almost any modern Web browser is Java-enabled and can handle the Java applications.





# Migrate to a Better Place



## Database Chaos?

How do you combine *automated code reengineering* and *database conversion* with *faster, more efficient*, and *accurate results*?

ManTech's SQL Conversion Workbench<sup>®</sup> (SQL-CW) offers a realistic approach for businesses looking to convert mainframe and mid-tier databases to DB2. SQL-CW allows the user to determine which application code to convert, permitting an evolutionary conversion where it is necessary. While some data may have been migrated to DB2, other data may remain in legacy repositories. With the selective conversion of data management calls supported by the SQL-CW, phased conversions are fully supported.

Alternatively, WORx converts such legacy systems as ADABAS-Natural, Datacom-IDEAL, and IDMS-ADS/O to IBM's DB2. The transition from older, mainframe computer systems to today's enterprise server technologies is smoother, too. With the ability to concurrently perform Year 2000 (Y2K) mitigation as an integral part of the overall conversion project, our full-service solutions result in a savings of time and money.

With ManTech's conversion tools, your organization can take advantage of the latest hardware and software environments without changing vital business processes. This provides for added capabilities, such as e-commerce and Web-based data processing.

**MANAGEMENT  
TECHNOLOGY MSSC**  
MANTECH SYSTEMS SOLUTIONS CORPORATION





Java lets you create small, byte-size code called *applets*. An applet is a small Java program that can be embedded within an HTML page. The applet is executed by the browser's Java Virtual Machine (JVM). In DB2 the applet does not require a DB2 client connection. When you open an HTML document with an embedded Java applet the browser will download the applet to the client machine along with the Java class file and the DB2's JDBC driver. The DB2 JDBC then will issue a command for a separate network connection to the DB2 database through the JDBC applet server.

Figure 2 illustrates how the JDBC applet driver, also known as the net driver, works. The driver consists of a JDBC client and a JDBC server, db2jd. The JDBC client driver is loaded on the Web browser along with the applet. When the applet requests a connection to a DB2 database, the client opens a TCP/IP socket to the JDBC server on the machine where the Web server is running. After a connection is set up, the client sends each of the subsequent database access requests from the applet to the JDBC server through the TCP/IP connection. The JDBC server then makes corresponding CLI (ODBC) calls to perform the task. Upon completion, the JDBC server sends the results back to the client through the connection.

SQLJ applets add the SQLJ client driver on top of the JDBC client driver, but otherwise work the same as JDBC applets.

Figure 3 illustrates how a DB2 JDBC application works. You can think of a DB2 JDBC application as a DB2 CLI application, only written in Java. Calls to JDBC are translated to calls to DB2 CLI through Java native methods. JDBC requests flow from the DB2 client through DB2 CLI to the DB2 server.

SQLJ applications use this JDBC support, but they also require the SQLJ runtime classes to authenticate and execute any SQL packages that were bound to the database at the precompiling and binding stage.

The JDBC API lets you write Java programs that make dynamic SQL calls to databases. SQLJ applications use JDBC as a foundation for such tasks as connecting to databases and handling SQL errors, but can also contain embedded static SQL statements in the SQLJ source files. You must translate a SQLJ source file with the SQLJ translator before you can compile the resulting Java source code. That way, you'll have a smooth transition from the raw JDBC into Java. ●

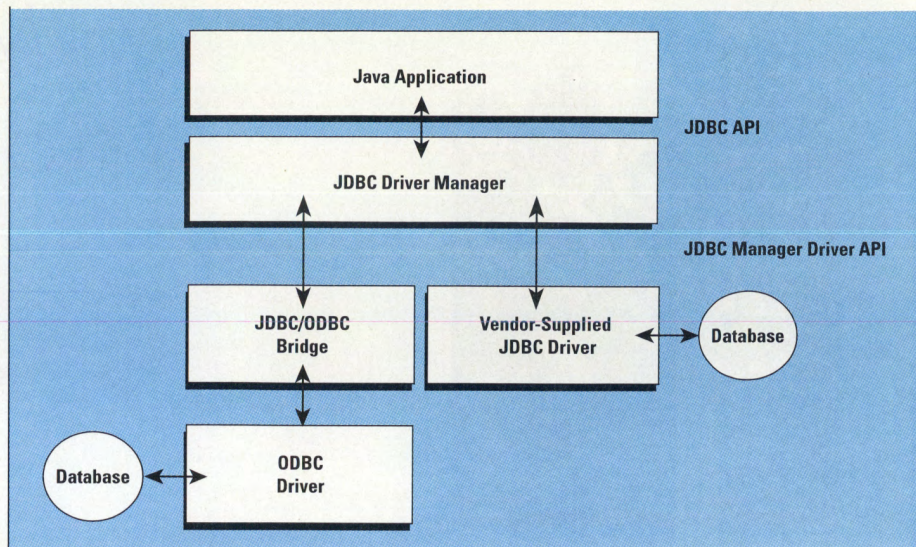


Figure 1: JDBC to database communication path.

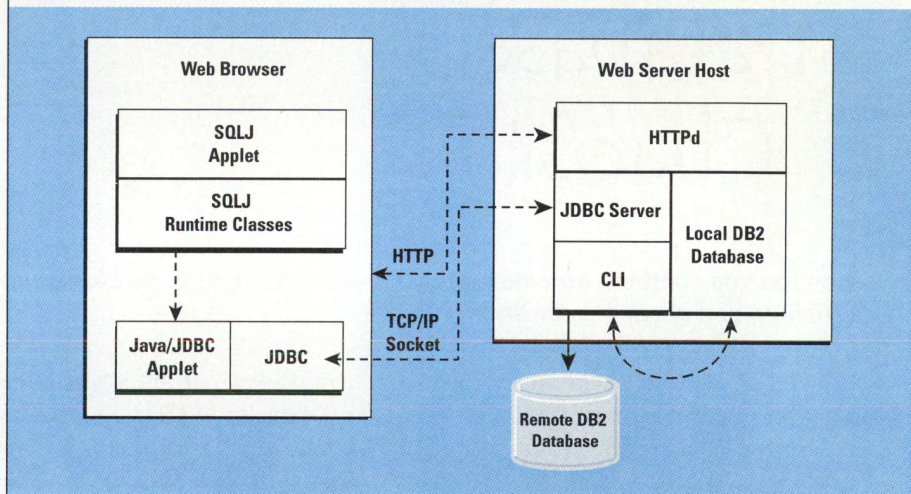


Figure 2: The JDBC applet driver.

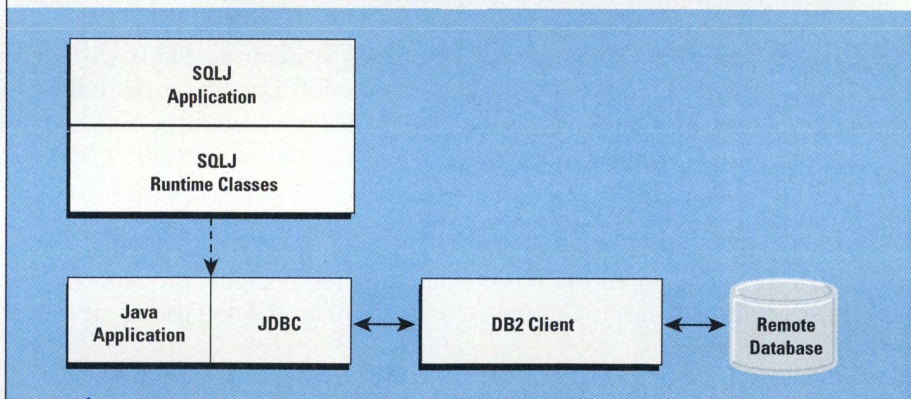


Figure 3: The DB2 JDBC application.

**Jeff Norton** is the president of the Windy City Linux User Group and chief technical officer of The PDA Group ([www.thepdagroup.com](http://www.thepdagroup.com)), which develops database solutions and applications for pocket PC and wireless technologies. In his spare time, he mentors young computer enthusiasts and writes computer fiction crime novels. You can reach him at [jnorton@thepdagroup.com](mailto:jnorton@thepdagroup.com).

**RESOURCES:**  
Read more about Java:

- [www.ibm.com/software/data/db2/java/](http://www.ibm.com/software/data/db2/java/)
- [java.sun.com/products/jdbc/index.html](http://java.sun.com/products/jdbc/index.html)
- [java.sun.com/docs/books/tutorial/jdbc/TOC.html#basics](http://java.sun.com/docs/books/tutorial/jdbc/TOC.html#basics)





e-business

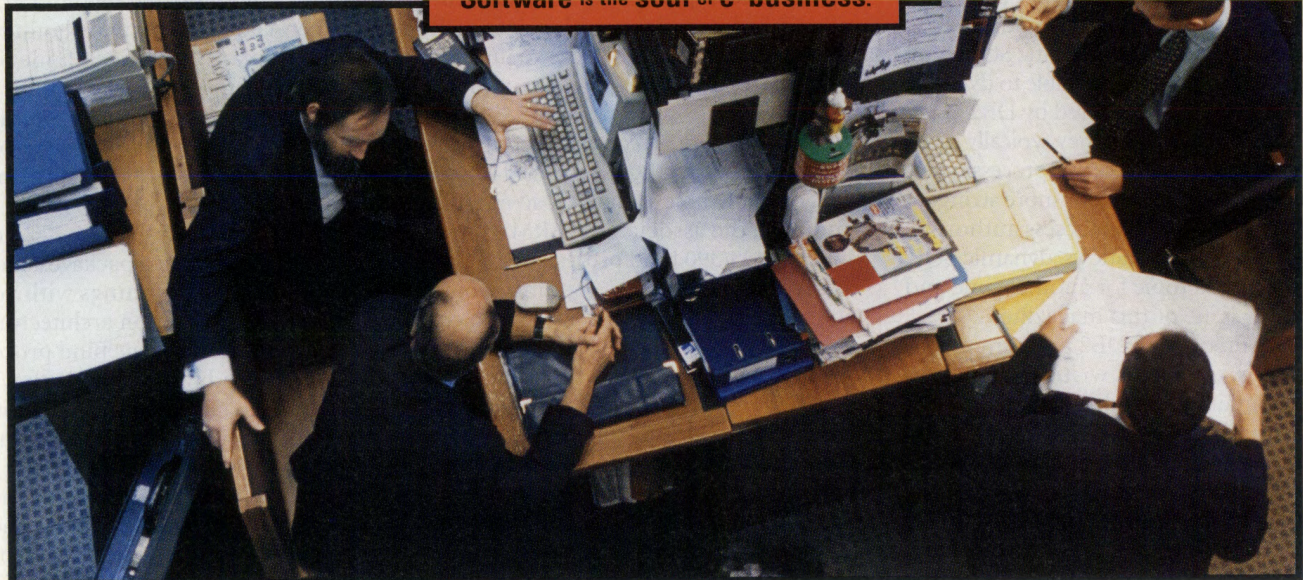


It takes powerful software to transform 7 years of unstructured data into an e-business resource. That's the software IBM makes.



**New Content Management solutions** from IBM can help you capture, manage and share information in any form – from documents to photos to Web pages – turning unstructured data into assets in the digital economy. Seeing is believing. Visit [www.ibm.com/software/data/cm](http://www.ibm.com/software/data/cm)

**Software is the soul of e-business.™**



IBM, the e-business logo and Software is the soul of e-business. are trademarks of International Business Machines Corporation in the United States and/or other countries. ©2000 IBM Corp. All rights reserved.



# Programs and Packages, Plans and Collections

*The ins and outs of DB2 program preparation and execution.*

**I**F YOU ARE A DB2 FOR OS/390 SYSTEMS programmer, DBA, or application developer (or a consultant who works with such folks), you are probably very familiar with the terms *package*, *collection*, and *plan*. But could you explain these elements of program preparation and execution to a DB2 newbie? And do you know how something as seemingly innocuous as a plan's package list can affect a program's performance? Or how DB2 really finds what it needs when it's time to execute a static SQL statement? This stuff really is more interesting than you might think. Don't believe me? Read on, and see for yourself.

## THE BASICS

In this column, I'll talk about programs through which static (prebound) SQL statements are executed. Static SQL statements are generally preferable to dynamic SQL statements (prepared by DB2 at execution time) because they typically make more efficient use of server processing resources and can allow for more stringent rules concerning data access authorization. The issue of static vs. dynamic SQL, however, is a topic for another time and another issue of this magazine.

The first step in DB2 program preparation is, of course, writing a program that contains embedded SQL statements. You could use COBOL, C (or C++), Assembler, FORTRAN, PL/I, or Java. (You may think that Java implies only dynamic SQL, but it doesn't if you take advantage of SQLJ. For a good introduction to SQLJ, see the Web Database column in the Spring

1999 issue, available online at [www.db2mag.com](http://www.db2mag.com).) You could write the program to run on the system on which the DB2 for OS/390 subsystem resides, or to access the DB2 database from a remote client via distributed relational database architecture (DRDA).

Once you've written the program, the DB2 precompiler processes it and generates two outputs:

**1. A modified program source module.** The precompiler comments out each of the program's embedded SQL statements, and inserts a call to DB2 for each statement.

**2. A database request module (DBRM).** A DBRM contains the SQL statements found in the program source.

The precompiler places a unique identifier, called a *consistency token*, into each of these outputs. I'll say more about consistency tokens when I get to program execution.

Following the precompile process, you compile and link-edit the modified source program into an executable load module and bind the associated DBRM. In the DB2 for OS/390 bind process, such tasks as access path selection (optimization), access authorization, and database object validation are performed. The output of the bind process is a control structure that DB2 will use to execute the SQL statements when the application program is run. The control structure will either be part of a plan (if the DBRM is bound directly into a plan) or contained within a package that will be executed via a plan.

How many of you work at a DB2 for

OS/390 site that still binds DBRMs directly into plans? I wish I could see a show of hands. I would expect (and hope) that few hands would be raised. The DB2 package bind process has been around for quite some time (since DB2 version 2 release 3), and it offers some important advantages over plan-direct binds.

At CheckFree Corp., we've been using package bind exclusively for quite some time. I strongly recommend that you do the same if you're not already doing so, for these reasons:

- **Improved availability.** If you change a SQL statement in a program, you only have to rebind one package. You can rebind one package quickly, and it's important to do so: A package cannot be executed while it's being rebound. If, on the other hand, programs are bound directly into plans, a change of one SQL statement requires that the plan be rebound. If you've bound a large number of DBRMs into the plan, the rebound could take a fair amount of time, during which the plan cannot be executed.

- **Improved flexibility.** As I'll show, binding programs into packages allows you to do some nifty things with your database and application architecture.

If you use the package bind process, you have to bind the package into what is called a *collection*. How do you create a collection? Pretty simple: You bind a package into it.

Of course, even if you're using packages, you still need to bind one or more plans if the program in question will run on the local or a remote DB2 for OS/390

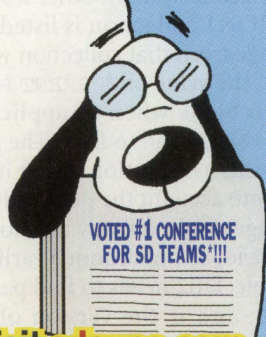
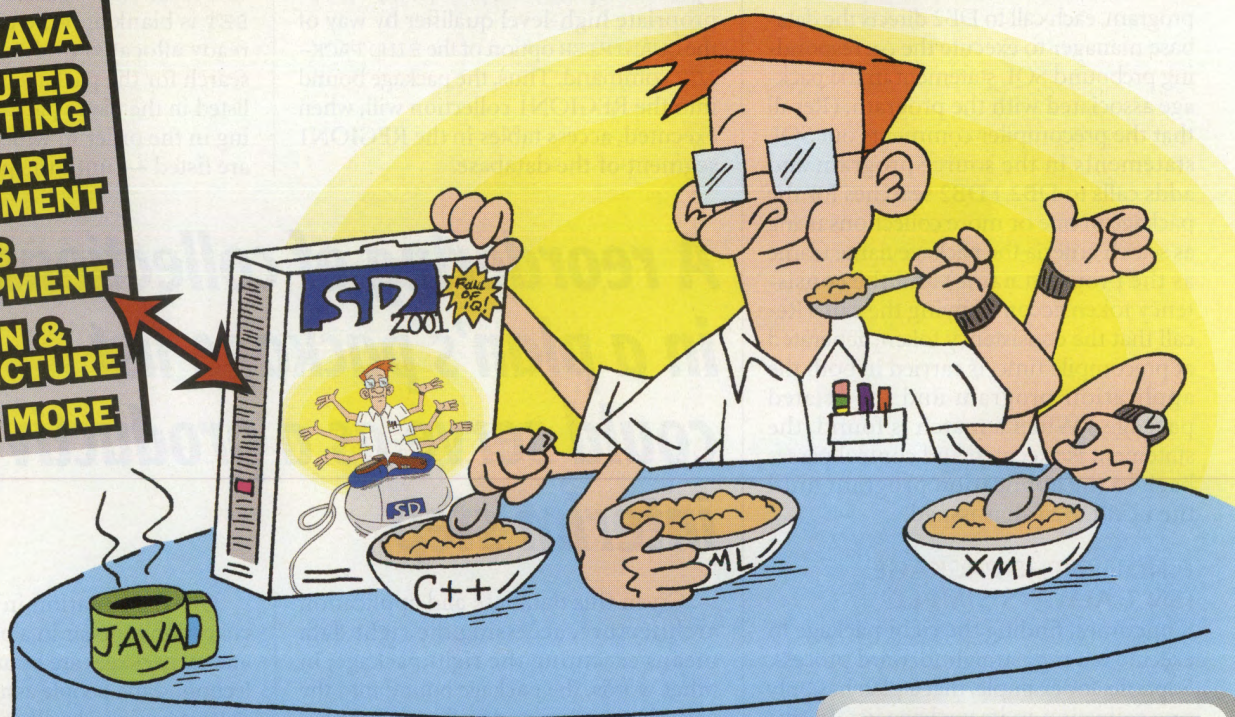


**SD**  
2001

**Software Development  
Conference & Expo  
invites you to...**

# FEED YOUR HEAD

**XML & JAVA  
DISTRIBUTED  
COMPUTING  
SOFTWARE  
MANAGEMENT  
WEB  
DEVELOPMENT  
DESIGN &  
ARCHITECTURE  
C++ & MORE**



Visit [sdexpo.com](http://sdexpo.com)  
for more information.

The SD Conference and Expo is chock-full of essential ingredients for the professional developer. Only SD offers a curriculum that is wide, deep and relevant to help solve your daily development challenges. From languages like Java, C++, XML to cross platform integration to architecture, design and management - SD has it all.

For the complete menu, visit our website at [www.sdexpo.com](http://www.sdexpo.com) or call 1.800.441.8826.

**PREMIERING THIS YEAR**

**WEB SERVICES  
WORLD**

A two day conference-within-a-conference devoted to the open standards of service-oriented architecture in support of the UDDI initiative.

*Don't miss this  
ground-breaking event!*

**Coming April 8 - 12, 2001  
San Jose, California**

*Serious content for serious developers*

\* In an independent research study by Evans Marketing Services for the year 2000, the Software Development Conference and Expo was voted the number one conference for software development teams.



subsystem. Programs that run on other remote clients and access DB2 via DRDA use a default plan called `DISTSERV`. You can execute a particular package using a plan if the collection into which you've bound the package appears in what is called the plan's *package list* (a list of one or more collections specified via the `PKLIST` option of the `BIND PLAN` command). The program, in turn, invokes the plan through a specification in the resource control table (or a `DB2ENTRY` if you're using resource definition online) for a CICS transaction, via the application program load module name for an IMS transaction, or with a control statement in the job control language (JCL) for a batch job.

When you execute the application program, each call to DB2 directs the database manager to execute the corresponding prebound SQL statement in the package associated with the program. (Recall that the precompiler comments out SQL statements in the source program and adds calls to DB2.) DB2 searches for the package in one or more collections using as search criteria the package name (same as the program name) and the consistency token accompanying the call. (Recall that the consistency token, generated at precompile time, is carried in both the application program and the related package.) When a match is found, the statement is executed and control passes back to the application program until the next DB2 call is issued.

### FINDING THE PACKAGE ISN'T ALWAYS SIMPLE

Sometimes, finding the right package to execute is a very straightforward process. Suppose, for example, that a plan has only one collection in its package list. In this case, it's obvious that DB2 will use packages bound into this collection.

Now, take the situation in which a plan's package list contains multiple collections. Why might this be the case? Two possible reasons are database segmentation and package versioning. By database segmentation, I refer to the practice of dividing at least a portion of the data among sets of otherwise identical tables within a single DB2 for OS/390 subsystem (or a single data sharing group). The division of data could be along geographic lines (for example, `REGION1` and `REGION2` tables) or by client or groups of client organizations (`COMPANY1` and `COMPANY2` tables, for example). Some DB2 shops with very large databases (including CheckFree) undertake such segmentation for database manageability and

availability reasons. You distinguish tables in the various database segments by way of a segment-related high-level qualifier. For example, sales data would be divided between tables with fully qualified names such as `REGION1.SALES`, `REGION2.SALES`, `REGION3.SALES`, and so on. In this type of scenario, package bind allows you to develop one program that you can use in each database segment, as follows:

First, the program is written using unqualified table names (for example, `SELECT TERRITORY FROM SALES`). Then the program is bound into multiple collections, one for each database segment (`REGION1` collection, `REGION2` collection, and so on). For each of these multiple bind operations, you specify the appropriate high-level qualifier by way of the `QUALIFIER` option of the `BIND PACKAGE` command. Thus, the package bound into the `REGION1` collection will, when executed, access tables in the `REGION1` segment of the database.

## A reordering of collections in a plan's package list could serve as a productive interim step.

Given this database and application architecture, accessing the right data means executing the right package; in other words, the package bound into the collection associated with the database segment of interest. Given a plan with a multicollection package list, how does DB2 know where to look for the package when executing an SQL statement? The search process is as follows (assuming that programs are not bound directly into the plan):

- DB2 first checks to see whether a special register called `CURRENT PACKAGESET` (one such register is maintained for each DB2 thread) contains a nonblank value. If it does, DB2 will search for the package in the collection specified (and will find the package, assuming — as is probably true — that each segment-related collection contains the same set of packages, distinguished only by the high-level qualifier specified at bind time). The value of `CURRENT PACKAGESET` is blank at the beginning of a trans-

action or batch job and can be updated by way of the SQL statement `SET CURRENT PACKAGESET`. Thus, if a program needs to access data in the `REGION2` database segment, it can do so by issuing the statement `SET CURRENT PACKAGESET = 'REGION2'`.

- If the value of `CURRENT PACKAGESET` is blank, DB2 will check to see whether the package is already allocated to the thread. This could be the case if, for example, the thread is reused by multiple transactions (an example being a CICS-DB2 protected thread) and the package in question was bound with `RELEASE- (DEALLOCATE)`. If the package is already allocated to the thread, DB2 will use that package.

- If the value of `CURRENT PACKAGESET` is blank and the package is not already allocated to the thread, DB2 will search for the package in the collections listed in the plan's package list — searching in the order in which the collections are listed — until the package is found.



the plan's package list. If, for performance reasons, you drive thread reuse and bind your most frequently executed programs with `RELEASE (DEALLOCATE)`, you'd better not think of any collection in a multicollection package list as the default. Instead, explicitly direct DB2 to the desired collection by way of `SET CURRENT PACKAGESET`. Even if all of your packages are bound with `RELEASE (COMMIT)`, you should probably use `SET CURRENT PACKAGESET` to get where you want to go, in terms of collections. It's simply good programming practice.

### COLLECTION ORDER'S IMPACT ON PROGRAM PERFORMANCE

Let's look at multicollection package lists from another angle. Suppose that, for example, you use a single plan in a production DB2 environment, which is a viable possibility with package bind. Then suppose that you group different packages into different collections by functionality. If each program is bound into only one collection, you might think that the order of collections in the "big plan's" package list is not very important — that DB2 will eventually find the package it needs as it searches the collections in package list order. In fact, collection sequence can matter, certainly in terms of program performance.

At CheckFree, we used to have a transaction that invoked a DB2 plan for which we'd specified a multicollection package list. We noticed that the execution of this transaction drove a large number of `GETPAGES` for buffer pool `BP0`, in spite of the fact that the transaction executed nothing in the way of dynamic SQL (our `BP0` basically is used only for the DB2 catalog and directory — user objects are assigned to other buffer pools). Upon investigating further, we noticed that a package the transaction executed very frequently was in a collection that was several spots behind the front of the plan's package list. Could these `BP0 GETPAGES` be DB2 directory accesses associated with unsuccessful searches for the frequently executed package in the first few collections in the package list? To find out, we moved the collection with the frequently executed package to the front of the plan's collection list, and voila! The `BP0 GETPAGES` dropped dramatically. Fewer `GETPAGES` means less CPU and elapsed time.

So, in a multicollection situation such as the one I've just described, put the collections holding the most frequently executed packages at the front of the package list. Alternatively, you can use `SET CURRENT PACKAGESET` to point DB2

right at the collection that you should search. This, as I've said previously, is good programming practice. The practice may take some time to implement, as is often the case with even relatively simple code changes. A reordering of collections in a plan's package list could serve as a productive interim step.

### A FEW WORDS ON PACKAGE VERSIONING

Allowing more than one version of a package (say, the newest and the next-most-recent versions) to exist within the package list of a plan can help in providing a quick backout capability in the event that a new version of a package causes problems. If such a situation occurs, you simply free the just-added version of the package (the one associated with a new and problematic version of the application program). When the previous (and well-behaved) version of the program is executed, the right package will be found because the previous version of the package is already in the collection list. If the previous version of the package had been removed from the collection list upon the bind of the new version, falling back to the old version would have required a rebind — an additional step that could delay the backout process and perhaps make the process more error-prone.

If you want to use package versioning in this way, you have at least two choices:

1. You could keep the two versions of the package in two separate collections, with the newer version going in the collection listed first in the package list of the plan. (I will refer to the two collections as `CURRENT` and `FALLBACK`.) Before a new version of a package is bound, the version of the package in the `FALLBACK` collection is freed. Then, you move the version of the package in the `CURRENT` collection from that collection to the `FALLBACK` collection. Then you bind the new version of the package into the `CURRENT` collection. If a problem necessitating a fallback occurs, the version of the package in the `CURRENT` collection is freed. When you execute the previous version of the program, DB2 will search for the package in the `CURRENT` collection, since it is listed first in the plan's package list. On not finding the package in the collection (because it was freed), DB2 will search for the package in the `FALLBACK` collection, resulting in the previous version of the package being found and utilized for SQL statement execution.

2. Alternatively, you could keep multiple versions of the package in the same

collection, with the version ID (specified at precompile time) serving to distinguish one version of the package from another. (Version ID is intended primarily for user management of package versions; DB2 for OS/390 uses the consistency token to find the correct version of a package at program execution time, as I mentioned previously.) In this case, you'd want to ensure that you periodically remove old and unneeded package versions (those older than the current and next-most-recent versions) from the collection. You could accomplish this goal with a user-written program, in which case you could opt for a DB2-supplied version ID. (Specifying `VERSION AUTO` at precompile time causes DB2 to use a timestamp value as the package version ID.) The program would use the version ID to identify and free packages older than the current and next-most-recent versions. If you want to remove old package versions manually, you might want to go for a version ID that is shorter and easier to specify than the timestamp value generated by DB2 with `VERSION AUTO`. (You can specify your own version ID when a program is precompiled.)

I think both approaches are viable, so go with the package versioning approach that looks best to you — there isn't a right or wrong answer.

### EXPLOITATION: IT'S A GOOD THING

I find the subject of packages, collections, and plans quite interesting and could go on and on. But before you tell me to get a life, I'd better wrap up. By taking a closer look at these and other tried-and-true DB2 for OS/390 capabilities, you can learn how to really exploit the product to benefit your business. ●

**Robert Catterall** is a strategic technology engineer with Atlanta-based CheckFree Corp. where he analyzes and evaluates database technology in order to facilitate the integration of new database capabilities with CheckFree application systems. You can reach him at [rcatterall@checkfree.com](mailto:rcatterall@checkfree.com).

### RESOURCES:

Read more about:

#### DB2 for OS/390

[www.ibm.com/software/data/db2/os390](http://www.ibm.com/software/data/db2/os390)

#### Five Reasons for Using SQLJ in Your Java Programs

[www.db2mag.com/db\\_area/archives/1999/q1/99sp\\_webdb.shtml](http://www.db2mag.com/db_area/archives/1999/q1/99sp_webdb.shtml)



## Predicate Evaluation: Part 2

*Stage 1 is better than Stage 2, and don't even think about using Stage 3. But Stage 0 predicates are the best solution of all.*

**I**N MY MOST RECENT PROGRAMMERS Only column (available online at [www.db2mag.com](http://www.db2mag.com)), I wrote about four reasons to avoid complex predicates, those predicates that are evaluated within DB2 at Stage 2 by “the Doctor,” DB2’s Relational Data Store (RDS) component. Those four reasons were that the more complex predicates consume more CPU, that they may cause unselected columns to be passed up from the Data Manager to the RDS, that they are not indexable, and that they may cause repetitive, non-productive program calls between the RDS and the Data Manager.

In addition to these reasons, there are two more reasons why you may not want to code Stage 2 RDS predicates. And all of these reasons, all six of them, if they are found to cause performance problems, will encourage you to rewrite your SQL to simplify it and, if possible, make

all of the predicates Stage 1. Or even Stage 0. Yes, that’s right. Zero. And never, or hardly ever, Stage 3. Yes, three. A total of four stages to consider.

So, in Part 2 we’ll look at these other reasons to avoid Stage 2 predicates and consider some alternative ways of coding predicates. And finally, we’ll consider the positives of Stage 0 predicates and the negatives of Stage 3 predicates.

So, what are the other two reasons? And, if we discover that our SQL is not performing well because of a Stage 2 predicate, what are some solutions?

### LATCHING OVERHEAD

Since Version 3 of DB2 for OS/390, the Buffer Manager has included a component to help with data integrity and the serialization of access to table data. This component, the Latch Manager, ensures that pages are “readable” (that is, physi-

cally stable) and, in some cases, that those pages contain only “clean” (that is, committed) data. In many situations, table pages can be read using just a latch to keep the page stable (and the rows clean) while the Data Manager examines the rows on the page and applies any Stage 1 predicates to those rows. In this way, latches, not locks, are used to ensure that the data being read is stable, readable, and — unless you requested uncommitted reads (UR) — clean. And, because latches are one-tenth the CPU overhead of locks, resources are thereby saved and locking contention is reduced or, in some cases, virtually eliminated.

Assume that you have a read-only query against a table that uses page-level locking and that the query is reading the table sequentially. And assume that there are 20 clean, committed rows per page and that you have bound your program



# Advertiser Index

COMPANY	PHONE NUMBER	WEB ADDRESS	PAGE NUMBER
AlphaBlox	1.888.BLOX.NOW	www.alphablox.com	37
Computer Associates Inc.	1.800.850.7528 EXT. 100	www.cai.com	32-33
Database-GUYS Inc.	1.877.4DB.GUYS	www.Database-GUYS.com/domore/	27
Embarcadero Technologies Inc.	415.834.3131	www.embarcadero.com/download/artisan.htm	35
HiT Software	408.345.4001	www.hitsw.com	11
IBM	1.800.627.8363	www.software.ibm.com/solutions/isv	1
IBM	1.800.627.8363	www.ibm.com/software/data/siebel	5
IBM	1.800.627.8363	www.ibm.com/software/different/db2	23
IBM	1.800.627.8363	www.ibm.com/software/soul/smarter	41
IBM	1.800.627.8363	www.ibm.com/software/data/cm	55
IBM	1.800.627.8363	www.db2mag.com	61
IBM	1.800.627.8363	www.ibm.com/software/different/db2imstools	C3
IDUG	312.644.6610	www.idug.org/esubscription/	17
IDUG 2001 North America	312.644.6610	www.idug.org	45
JustTechJobs.com	1.877.906.6373	JustDB2Jobs.com	9
Mantech Systems Solutions Corp.	877.GO 2 MSSC	www.mssc-mantech.com	53
Quest Software	1.800.930.7731	www.quest.com	C4
Relational Architects International	1.800.776.0771	www.relarc.com	C2, 25, 49
Responsive Systems	732.972.1261	www.responsivesystems.com	2
ResQNet.com	212.537.4800	www.resqnet.com	36
Sivco Inc.	1.877.700.2600	www.sivco.com	19
Software Development Conference & Expo	1.800.441.8826	www.sdexpo.com	57
Sql Power Tools	702.990.3209	www.sqlpower.com	11
Yevich, Lawson & Associates Inc.	1.800.484.8158	www.ylassoc.com	31

# DB2<sup>®</sup> magazine

## Attention Vendors

If you would like to see your ad in the next issue of *DB2 Magazine*, contact:  
Angela Duarte  
(415) 389-9632



**SUBSCRIBE**

To get your free subscription for DB2 Magazine, visit our Web site at

**www.db2mag.com**

and fill out the online subscription form.



with either `ISOLATION UR` (readable, stable, and either clean or dirty data is OK without locking overhead) or `ISOLATION CS` with `CURRENTDATA NO` (readable, stable, and clean data is OK without locking overhead — use locks for dirty data only). Whenever the Buffer Manager determines that the page is in the buffer pool, the Latch Manager attempts to obtain a latch on the page. If the page is unstable, meaning unreadable (I did *not* say “unclean”), then you will endure a latch suspension and will have to wait until the page is stable again and the latch is available. You will not timeout, but you may wait. Once the latch is obtained, then the Data Manager can read the page. (Remember our assumption is that there are only clean, committed rows on this particular table.)

The Data Manager addresses Page 1 in the buffer pool and evaluates Row 1, then Row 2, and so on. As long as rows disqualify by Stage 1 predicates, the Data Manager just continues reading rows on

readable. Wait and endure a latch suspension. And if the program was bound with `ISOLATION CS`, wait while the page is evaluated to see if it now contains uncommitted data. If it does, then it is also an opportunity to wait for a lock request, a lock suspension, and possibly a lock timeout. All opportunities that we could do without.

Of course, if the Stage 2 predicate has a very low filter factor, if you want almost every row that is passed to the RDS, then relatching will be rare and not a big problem. But if the Stage 2 predicate is highly filtering and has a high rejection rate, then releasing and reacquiring up to 20 latches per page is overhead that is best avoided.

#### LESS EFFICIENT LOCK AVOIDANCE

Again, assume that you have a read-only query that is reading the table sequentially and assume that there are 20 clean, committed rows per page. But this time

sion). An opportunity to timeout. And whenever the lock is successfully acquired it will provide other users with the opportunity to have locking contention with you. All of these are opportunities that you could probably do without.

Once the lock is acquired (*if* it is acquired), then the row can move from the Data Manager to the RDS. And while the lock is maintained, the latch on the page is released.

If there are no Stage 2 predicates, then the row will continue on to either become input to a sort or find its way back to your program. But if there are Stage 2 predicates, they must now be evaluated. And whenever any of the Stage 2 predicates cause the row to be rejected, then the RDS must call the Data Manager again to get the next row, off the *same* page — the page you no longer have a latch on. And this means that the latch must be reacquired. But it also means that you now have a lock that you did not need on this page. No row *really* qualified. The row only qualified by the Stage 1 predicates. But it is Stage 1, the Data Manager, who handles lock requests. And you did not know that the row actually *disqualified* until the Stage 2 predicates were applied.

In other words, the Stage 2 predicates caused you to prematurely lock the page before you really, completely knew for sure that the row qualified. And if no row ever qualifies (once all predicates are applied) then you have acquired and held a lock on a page when no row even qualified on that page. Not a good thing.

Of course, again, if the Stage 2 predicate has a very low filter factor, if you want almost every row that is passed to the RDS, then you have acquired locks that you need and want using `ISOLATION CS` and `CURRENTDATA YES`.

#### WHY RDS PREDICATES MAY BE PERFECTLY OK

I do not mean to give you the impression that all Stage 2 predicates are bad. I used to think that they should be avoided at all costs. In fact, I would move them out of my SQL and into my COBOL program to be evaluated after the call to DB2, after the row was returned to my program. Today, now that I understand a little more about *why* these predicates may cause problems, I understand that my solution those many years ago was not a good one. I am embarrassed to tell you that I made my Stage 2 predicates into Stage 3 predicates. I put the qualifying logic in the form of “if, then, else” logic in my COBOL program to be applied to the DB2 row *after* the row was returned to

## If there is anything worse than a Stage 2 predicate, it's a Stage 3 predicate.

the page using the same latch. But whenever a row qualifies (either because all of the Stage 1 predicates qualify the row or because there are, in fact, no Stage 1 predicates), then the row must move up to the RDS. At the point when the row moves from the Data Manager to the RDS, the latch on the page is released.

If there are no Stage 2 predicates, then the row will continue on to either become input to a sort or find its way back to your program. But if there are Stage 2 predicates, they must now be evaluated. And whenever any of the Stage 2 predicates cause the row to be rejected, the RDS must call the Data Manager again to get the next row, possibly off the same page, the page you no longer have a latch on. And this means that the latch must be reacquired.

During the time that there was no latch on the page, there was an opportunity for someone to make the page unstable, unreadable, or “dirty” (it now contains an uncommitted updated, deleted, or inserted row). And the attempt to reacquire the latch is an “opportunity” to wait. Wait for the page to become stable and

assume that you have bound your program with `ISOLATION CS` with `CURRENTDATA YES` (readable, stable, clean data can be read and evaluated, but once a row qualifies a lock must be acquired). The Latch Manager latches Page 1 (immediately or after a short wait) and determines that the page is clean. The Data Manager addresses Page 1 in the buffer pool and evaluates Row 1, then Row 2, and so on. As long as rows disqualify by Stage 1 predicates, the Data Manager just continues reading rows on the page using the same latch. But whenever a row qualifies (either because all of the Stage 1 predicates qualify the row or because there are, in fact, no Stage 1 predicates), then a lock must be acquired before the row can move up to the RDS.

The DB2 Lock Manager, the Internal Resource Lock Manager (IRLM), is located in a different address space from the RDS and the Data Manager. Therefore, a lock request requires a cross-memory services call from one address space to another address space. And each lock request is an opportunity to be told “no.” An opportunity to wait (a lock suspen-



my program. Not smart. If there is anything worse than a Stage 2 predicate, it is a Stage 3 predicate.

### WHY STAGE 3 PREDICATES ARE WORSE

I find it interesting that I see far more Stage 3 predicates than Stage 2 predicates. Why? Because often data is moved from VSAM or flat files into DB2 tables without the benefit of having gone through any true relational logical and physical design phases. And then the programs that used to access the VSAM files are converted to access those new tables (and I use the word “tables” very loosely here). Generally I/O modules that used to access VSAM files are changed to access DB2 tables. The program logic stays the way it was when the file access was VSAM

## Sometimes the best-performing SQL is the SQL that you do not do.

(and no WHERE clause was available). In other words, COBOL “if, then, else” logic is used after the I/O module call to DB2 to either qualify or disqualify the ersatz row is retrieved. Often no one takes the time to look at the program logic to determine what logic could be moved into the SQL WHERE clause.

Any time a row is rejected in your program *after* a call to DB2, the rejection is Stage 3. And I guarantee you that, regardless of the simplicity or complexity of the criteria, the Stage 3 predicate is not indexable and that every rejected row is going to cause a nonproductive call from your COBOL program in its address space to DB2 in its address space. And latches have been released and locks may have been acquired, all for a row that that may ultimately be rejected by the program logic.

Remember the analogy that I used in the previous column? The fruitless trips to the airport to get balls out of the box? Balls that were rejected in the city? The impact of Stage 3 predicates is worse. You go to the airport to get me a ball out of that box and bring it back to me at the hotel in Kansas City where I then get in my car and drive to Minnesota and hand it to someone there who says, “I don’t want that ball.” At least you and I, the airport, and the hotel are in the same state,

just as the Data Manager and the RDS are in the same DB2 address space. But the COBOL program is in Minnesota, a completely different address space.

Stage 2 predicates are better than Stage 3. And Stage 1 predicates are better than Stage 2. And Stage 0 predicates are the best of all.

### WHAT IS A STAGE 0 PREDICATE?

Sometimes predicates can be evaluated prior to calling DB2, and that is best of all. My friend Prem Mehra used to say that “the best-performing SQL is the SQL that you do not do,” and that is what Stage 0 predicates are all about.

Consider the following SQL statement: `Select cola, colb from taba where :hvflag = 'y' and colc = :hvc.` This

statement has a predicate in which a host variable is being compared to a literal. This predicate is Stage 2 and not indexable, and it will be done last, not first. DB2 will retrieve a page, qualify a row using the Stage 1 predicate (`where colc = :hvc`) and then pass that row up to the RDS. The RDS will check the host variable flag to see if it is *y*. It would be much better if the flag were checked in the program at Stage 0 and the SQL were performed only when the flag is *y*.

Another example of a predicate in which some of the work could be done at Stage 0 follows: `Select cola, colb from taba where qty_onhand > :ohqty * 1.0765.`

Assume that this predicate will be applied to one million rows. Wouldn’t it be better to do the multiplication in your program before the call to DB2? You could then change your WHERE clause to `where qty_onhand > :calc_qty` and the predicate could be done at Stage 1, rather than Stage 2. While you may not have an index on `qty_onhand`, by doing the mathematical calculation in your program before the call to DB2 you have eliminated the need for rows to be passed from the Data Manager back to the RDS for evaluation.

In the prior column I gave you the following example: “Let’s look at a predicate

in the format `WHERE COLA = :HV1 + :HV2.` This predicate is easily converted into a Stage 1 predicate. All you have to do is compute `:HV1 + :HV2` and move the value into `:HV3.` If you do this *before* the call to DB2, then the SQL predicate can be `WHERE COLA = :HV3,` a simple Stage 1 predicate.” I was simply using the Stage 0 concept to make the SQL as simple as possible. Why make a predicate difficult when it can be simple? Why be clever when simplicity performs better? Why do a mathematical computation for each row when it can be done once *before* calling DB2? What I did was remember the rule that Stage 0 is better than Stage 1, and Stage 1 is better than Stage 2. While I could have made the predicate Stage 1 and indexable by using various other techniques, all of those techniques would have been less efficient than the Stage 0 technique.

Also, in the prior column I gave you the following example: “Suppose you want rows `WHERE trandate + 5 days > :hvdate.` You have an index on `tran-date,` but DB2 cannot use it because the predicate is Stage 2. If you add an SQL statement to calculate `hvdate - 5 days,` (`Select date(:hvdate) - 5 days into :hvnewdate from Sysibm.Sysdummy1,`) then you can change the WHERE clause on the big table to `WHERE tran-date > :hvnewdate,` which is an indexable Stage 1 predicate.” I could have used a technique that would have made the predicate Stage 1 and indexable, but that technique would have caused the date math to be done for every single row accessed. Why make DB2 do the work multiple times when your program can do the work once?

### SOME SOLUTIONS ARE BETTER THAN OTHERS

There are often many different ways of coding the SQL that will return the rows you need, but all SQL solutions are not equal. Some are better than others. And you get no points for a solution that increases the workload and CPU overhead unnecessarily. Clever does not always equal efficient. The programmer who writes SQL that enables minimal I/O, uses minimal CPU power, and is simple and easily understood gets the blue ribbon. ●

**Bonnie Baker** is a consultant and educator specializing in application performance issues on the DB2 for OS/390 platform. She is a five-time winner of the IDUG Best Speaker award and is best known for her series of seminars entitled “Things I Wish They’d Told Me 8 Years Ago.” She can be reached through Bonnie Baker Corp. at 800-992-7192 or [bkbaker@bonniebaker.com](mailto:bkbaker@bonniebaker.com).



## Got Utilities?

*The decision about which utility to use just got more interesting.*

**T**HERE ARE ALWAYS CHANGES in the marketplace with respect to vendor product packaging, pricing, new products, and new versions. Change is simply a fact of life in the technology industry. Vendors strive for innovation and product improvements to better meet the buyers' needs, and (of course) to generate additional revenue. Pricing models also change, as we have seen in per-CPU pricing, user-based pricing, and numerous other variations.

IBM's recent change in utility and tool packaging with DB2 Universal Database (UDB) for OS/390 version 7 was one of the hot topics of discussion at the recent IDUG conference in Geneva last October. In recent years, IBM has continued to enhance its products in the tools area, and the new products and changed packaging with Version 7 are among the results. Specifically, the group of DB2 utilities (for example, the operational, recovery, and diagnostic utilities) that were previously included with the base product at no additional charge will be separately priced in the future.

Of course, independent tool vendors have long offered alternatives to the IBM utilities. These options cost more, but provided additional features, usability, or speed.

IDUG reviewed this change in the landscape and has worked to accommodate it, keeping in mind the best interests of our members and the DB2 community at large. We reviewed our position in the light of our mission statement:

*The International DB2 Users Group is an independent, not-for-profit, user-run organization whose mission is to support and strengthen the information systems community by providing the highest quality education and services designed to promote the effective utilization of the DB2 family of products.*

In order to keep providing you with the best education and services, we've decided to include in this year's IDUG conferences technical presentations on the independent tool vendors' utilities, as well as those from IBM. Each of the

leading utility vendors will have an equal opportunity to present their offerings. These will be educational (not marketing) presentations put together to inform and update our attendees. We hope these sessions will be helpful as you make your product decisions.

Now, I'm going to get on my soapbox and ask you for something. Almost one year ago, IDUG launched the Web site e-Subscription, a free online information source geared specifically toward users of DB2 UDB for Unix, Windows, and OS/2. We need volunteers to help with this site by adding to its content and keeping it fresh and meaningful. Because it's online, it's a perfect opportunity for people around the world to get involved. (World peace through DB2!) If you'd like to help, please email us at [idug@idug.org](mailto:idug@idug.org). ●

**Brenda Castiel** is president of the International DB2 Users Group (IDUG). She has many years experience in IT development, marketing, and consulting and currently works for an Internet start-up in Los Angeles. You can reach her at [idug@idug.org](mailto:idug@idug.org).





e-business software

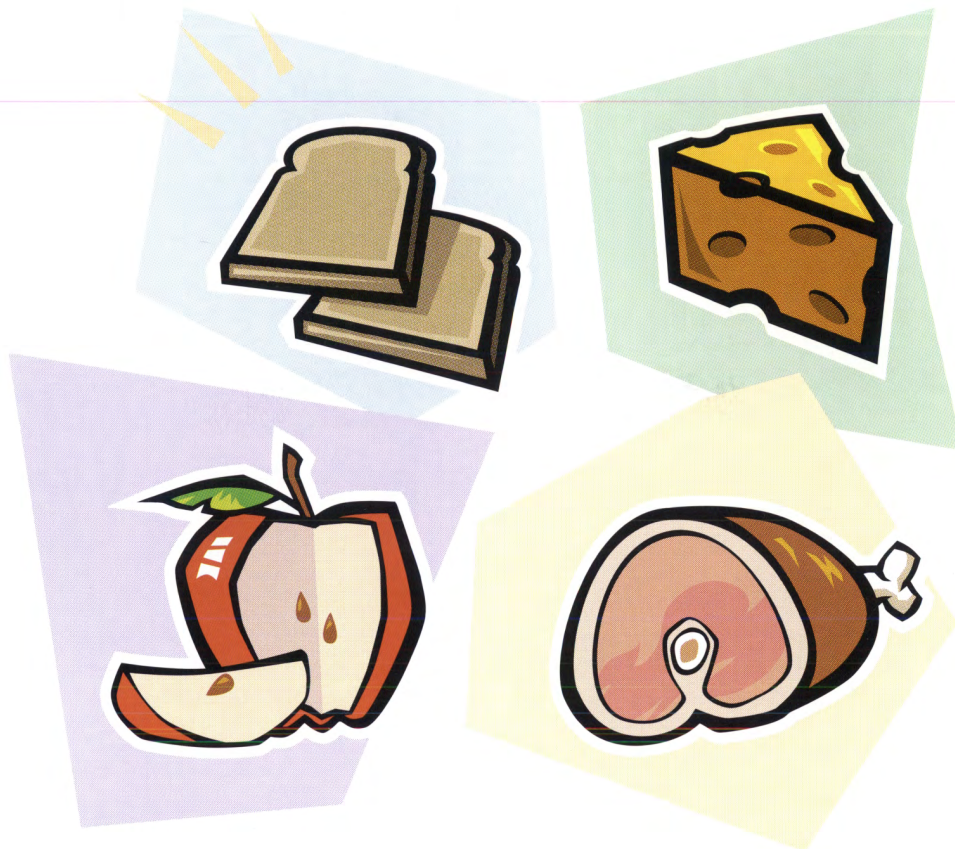
# A database manager can't have too many power tools.

It's a different kind of world.  
You need a different kind of software.

IBM has dozens of new and enhanced DB2<sup>®</sup> and IMS<sup>™</sup> tools to optimize your system utilization. Our tools can lower your cost of ownership while helping you manage your database environments. To hear top IBM tools managers discuss IBM S/390<sup>®</sup> software trends and cost management strategies, visit us at [ibm.com/software/different/db2imstools](http://ibm.com/software/different/db2imstools)



# Satisfy your Hunger for DB2 Tools with Quest Central™ for DB2



## Providing the four essential food groups for a healthy database

Quest Software introduces the first ever, well-rounded database management solution for DB2 UDB. Quest Central for DB2 combines the four essential food groups for today's DBA — performance monitoring, database administration, SQL tuning and space management — in one compact, easy-to-use interface.

Once you've gotten a taste of our smart integration, you'll never want to go anywhere else. Quest Central accommodates your workflow, so you can work better, smarter and faster. Now that's easy to swallow.

Fortify your database today. Download a free trial version at [www.quest.com](http://www.quest.com), or call us at 800.930.7731.

