Personal Systems

IBM's MAGAZINE FOR TECHNICAL COORDINATORS

JULY/AUGUST 1995

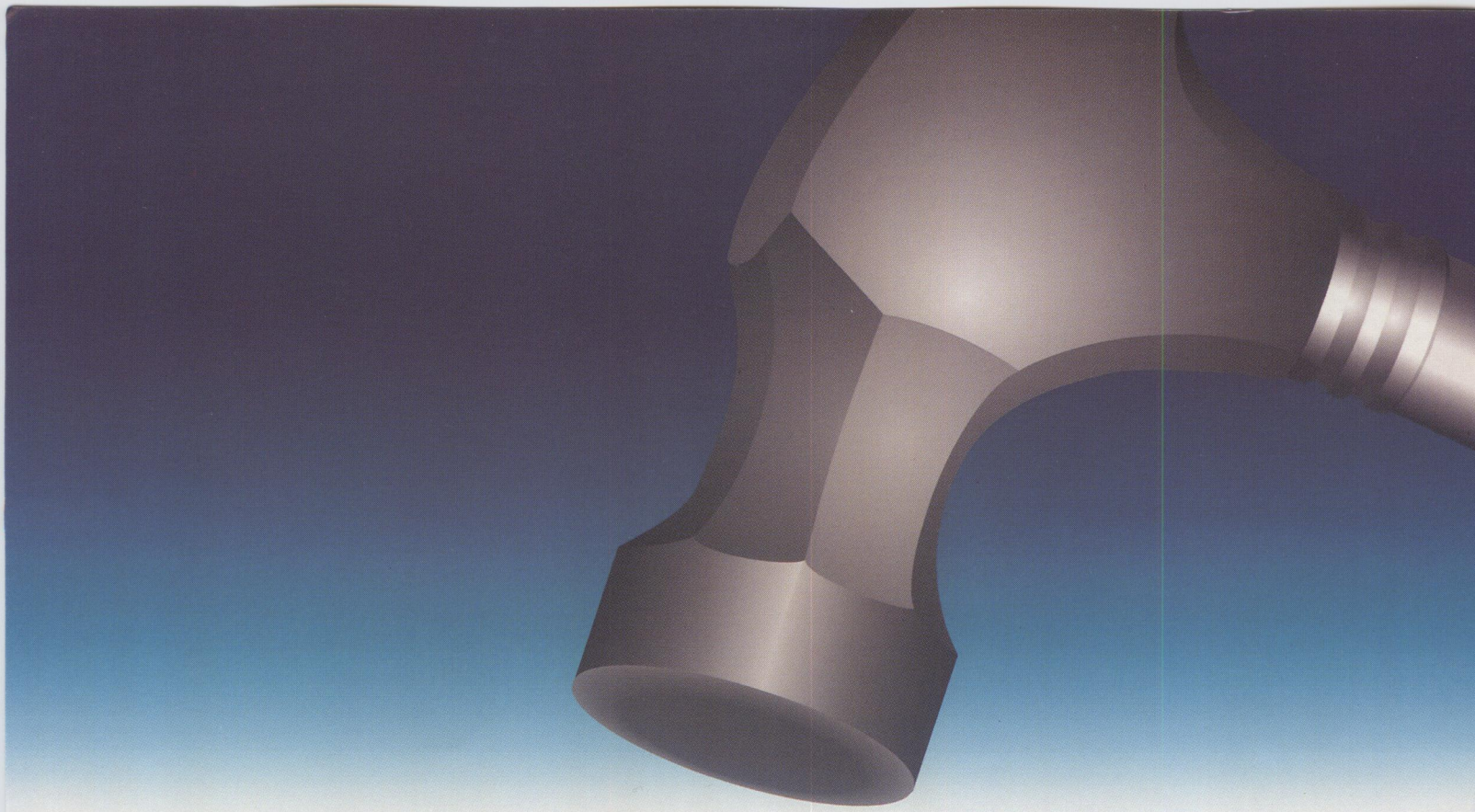
DATABASE 2 for OS/2:

- Performance
- Tools
- Recovery
- Connectivity

Visualizer Development

RPL OS/2 Warp from NetWare

IBM



A PROGRAMMING TOOL THAT WON'T SCREW YOUR HEAD UP

Splat! There goes another masterpiece. It was going to be a work of art, a monument to elegance and flexibility. It was going to be your finest creation. What happened? It got beaten into unrecognisable garbage by some clumsy GUI builder. And the controls ended up as subtle as a flying sledgehammer thanks to a resource editor with an attitude.

It's enough to turn a decent living programmer into a serial killer. Somebody out there had better come up with a solution.

We just did. It's called Prominare, a professional's programming tool for GUI creation that has all the answers. Take a look at the code it generates. It's exactly the way you would have written it yourself. Define your naming convention and Prominare will stick to it to the letter.

To create custom controls, use Prominare as the resource editor and you'll have enough options to give you total freedom. It will also handle all major object libraries, PEN and multimedia. It handles all versions of OS/2 and understands your old resources or Windows resources as well.

Prominare is also intelligent. When you make modifications to an application you won't be hindered by unnecessary generation phases, it only regenerates the parts that have been modified.

Now here's the best bit. A shareware version called Prominare Lite is freely available on the Internet. Help yourself to it and see how it beats the stink out of anything else. Then get your head around this question. If the shareware version is this good, what will the full version do?

AVAILABLE THROUGH:
INDELIBLE BLUE TEL: 1-800-776-8284 (OUTSIDE USA 1-919-878-9700)
OS/2 EXPRESS TEL: 1-800-672-5945 (OUTSIDE USA 1-612-823-6255)
EGGHEAD SOFTWARE TEL: 1-800-344-1123 (OUTSIDE USA 1-509-922-7031)
IMAGESOFT INC. TEL: 1-800-245-8840 (OUTSIDE USA 1-516-767-2233)
PROGRAMMERS PARADISE TEL: 1-800-441-1511 (OUTSIDE USA 1-908-389-9228)

FREE
Prominare Lite
AVAILABLE ON THE INTERNET
TRIAL

Prominare

Tel: +1 (416) 363 2292 Fax +1 (416) 363 6157
Tel +44 (117) 987 2679 Fax +44 (117) 972 8600
e-mail: designer@prominare.com
Connect to our Web site at http://www.prominare.com/prominare

Please circle #24 on reader service card.

LAN "Intensive Care Utilities" For IBM LAN Server 3.0/4.0

ICU FOR YOUR LAN

INCLUDES INDUSTRIAL STRENGTH TOOLS TO:

- ***Eliminate flying blind or worrying about corrupted domains***
  Export your entire domain or just a part into simple and easy to read text files. These text files can then be imported in part or as a whole.

- ***Fix your domain using artificial intelligence techniques***
  Our LAN analyzer goes through your NET.ACC and DCDB files looking for corruption and inconsistencies. Using a rule base, the utility can correct most of the problems encountered. All problems are logged.

- ***Put into place a corporate wide security policy on LAN Server***
  Our LAN analyzer will check each and every account against a rule-based profile and alert you to accounts that contain access and application assignments beyond their privilege and memberships. Accounts that are missing assignments are also flagged. Extra or missing assignments can be corrected on-line immediately. Check hundreds of users in a matter of minutes.

- ***Balance your loading***
  Reports are produced providing cross reference of alias usages as well as permissions by user and group for better control of your LAN.

- ***Build and update your LAN FAST!***
  Rapid add/delete/update of users, groups, aliases, and applications using simple ASCII text files. Templates allow you to define prototype resources or users. Templates allow you to type in only those fields that are unique for users and resources.

ORDER TODAY! *And receive your copy of LAN Intensive Care Utilities electronically in minutes.*

$970.00 *SITE LICENSING AVAILABLE AT SUBSTANTIAL DISCOUNTS*

Use your American Express, VISA, MasterCard or Discover Card.

Free demo disk available or download the demo from our BBS, CompuServe or IBMLink.

Lieberman and Associates
Design and Engineering Group
221 N. Robertson Blvd. / Suite C / Beverly Hills, CA 90211

800-829-6263

Phone: (310) 550-8575 IBMMail: USMVHLVH
FAX: (310) 550-1152 CompuServe: 76426,363
BBS: (310) 550-5980 OS2BBS1: LANUTIL
Internet: 76426.363@compuserve.com

Please circle #15 on reader service card.

Available and Ready for IBM LAN

Available and Ready for OS/2

Authors—Our Greatest Resource

Personal Systems magazine is only as good as its authors, and we think they're pretty darned good. As a matter of fact, good enough to win regional and national awards from the American Society of Business Press Editors (ASBPE)!

Todd Watson, who you know mostly for his lighthearted fun-poking at anything and everything, showed his very serious and technically competent side in the May/June 1994 issue with his "Wrightsizing at USAir" article. This thoroughly researched and well written article won first place in the case study category in the regional ASBPE contest and then went on to win at their national competition. We congratulate Todd, who has agreed to continue to contribute his special talents to *Personal Systems* even as he moves on to a new opportunity within IBM.

At the risk of appearing to pat ourselves on the back even more (which is exactly what we're doing!), we are pleased to tell you that *Personal Systems'* contents pages also won awards for design in ASBPE's regional and national competitions. Our thanks goes to Terry Pinkston of Corporate Graphics in Arlington, Texas for his clear, functional design of the contents page.

Database Focus

Another of *Personal Systems'* long-time authors, Nancy Miller, put her special touch on this issue. Thanks to her coordination efforts, as well as her written contributions, the features and enhancements of IBM's DB2 for OS/2 product are thoroughly documented in this issue.

Nancy brought together technical experts from IBM support centers in Dallas, Texas and Toronto, Canada to share their expertise with you. You'll find in-depth discussions on DB2 performance, tools, recovery, connectivity, object-oriented programming, and much more. Of special interest is the voting kiosk application described in the Focus section. Think of the possibilities!

Success in New Orleans

We got to meet a lot of *Personal Systems* readers at IBM's Technical Interchange in New Orleans last month as they participated in special Technical Coordinator Program recognition events. We hosted over 1,100 current and new technical coordinators at a reception and breakfast.

The conference was a huge success. If you missed it, make your plans now to attend next year as the Technical Interchange and special Technical Coordinator Program events move to Nashville in April for some country fun!

Free Stuff!

Since we introduced our "What's New for OS/2?" feature earlier this year, you've told us it's one of your favorite parts of *Personal Systems*. It's where you go to find out about new products—what you need to do your job. Because we have limited space to tell you about all the new products, we provide a method by which you can get more information about the products in which you are interested. Each product has a Reader Service Card number at the end of its review. The Reader Service Card is located between pages 56 and 57. Circle the numbers of the reviewed products as well as the numbers of the advertised products and mail the postage-paid card. It's all free!

Betty Hawkins

Betty Hawkins, Editor

New OS/2 software sends messages to wireless pagers

"It's so easy with ChipChat"

Innovative software combines the power of OS/2 with the freedom of wireless pagers to improve your communications and enhance your productivity

INTRODUCING THE ChipChat® Wireless Communicator, an exciting new software product that sends text messages to pagers directly from OS/2®.

ChipChat works with pagers from virtually *any* paging company, including Airtouch®, Ameritech®, MobileComm®, PageNet®, SkyTel® and many others.

Remarkably easy to use

ChipChat is object-oriented - just like OS/2. It's remarkably easy and natural to use. In fact, ChipChat looks and acts like it was "built right in" to OS/2.

Bill's Pager

For example, say you want to send pager messages to 'Bill'. You start by dragging a 'pager object' out of a 'pager template' then dropping it on your OS/2 desktop. Name it something like "Bill's Pager". Then whenever you want to send Bill a message, you just 'double click' on "Bill's Pager".

Up pops a friendly notepad. You simply write your message and press the 'Send' button. Using your modem and phone line, ChipChat calls the paging company and sends your message. Within moments your message is transmitted about town or around the country to Bill's actual pager.

"ChipChat's great! It's easy to understand and use, and helps me to squeeze more out of each day."

Dr. Laura Williams M.D., Canton Ohio

"ChipChat is a terrific product! It's helped us to vastly improve our service. It's also saved us money."

Wm Winger, CEO Winger Inc, Mpls

Page-Enable your applications

You can also send pager messages from other programs via ChipChat. This is called 'page-enabling' your applications. It's possible from REXX programs, from C or C++ programs, from the OS/2 command line, even from some spreadsheet and database programs.

As an example, ChipChat's ideal for LAN administrators who want to receive an immediate detailed pager alert whenever there's a problem on the network.

Just think how page-enabling your applications with ChipChat might make your organization more responsive, more competitive, and more profitable.

ChipChat puts you in control

ChipChat can keep a record of all your paging activities: dates, times, names, messages. It's easy to use this information to your advantage. Simply 'copy & paste' this data into a spreadsheet. Then instantly analyze your paging activity.

You can easily verify your paging bills and intelligently select the most cost-effective paging service. You can take control of your paging costs and maximize your paging effectiveness.

Robust & Reliable

ChipChat is advanced multi-threaded 32-bit object software based on IBM's SOM technology. It's been "through the ringer" with extensive corporate beta testing. It's passed a suite of rigorous tests set by IBM®, and has earned the IBM certifications: "Available and Ready for OS/2" and also "Available and Ready for LAN Server".

How much? Just $79

ChipChat is an incredibly good piece of software, and at our very competitive price of only $79 it's also an incredibly good deal!

See for yourself -

Completely risk-free!

The ChipChat Guarantee

If you're dissatisfied with ChipChat for any reason, if it isn't everything we say it is and more, then return it within 30 days for a prompt, cheerful refund.

Order today and send messages to pagers the easy ChipChat way!

"ChipChat-Cawthon Software understands the OS/2 workplace shell better than some of the biggest software companies, and it shows."

Tim Sipples, Team OS/2, Chicago

☑ ***To Order ChipChat Phone 313-565-4000 or Fax 313-565-4001***

The ChipChat Wireless Communicator sends messages to wireless pagers. It's remarkably easy to use and can also 'page-enable' your applications. ChipChat costs **only $79** + $3 shipping and comes with a **30 day moneyback guarantee**. We accept **Visa, MasterCard, & American Express**.

Send to: _____ Card number: _____

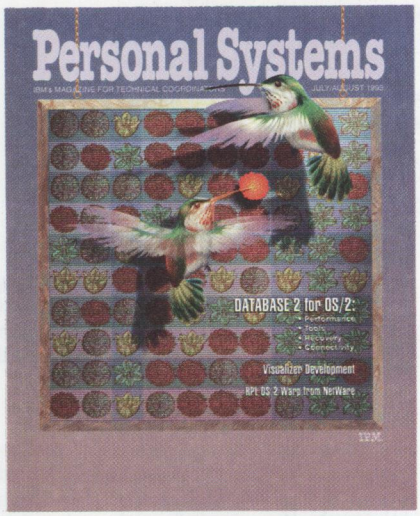
_____ Expiration date: _____

_____ Signature: _____

ChipChat-Cawthon Software, 24224 Michigan Avenue, Dearborn, Michigan 48124 USA
ChipChat-Japan, Fukuoka Japan, phone: 81-(092)-943-0798 fax: 81-(092)-944-2253

ChipChat Cawthon Software®

Please circle #16 on reader service card.



ABOUT THE COVER
Databases make order out of chaos, weaving like and unlike bits of information into a fabric from which data can be extracted in whatever form you need. Dallas artist Bill Carr uses this analogy, along with his inimitable style, to illustrate the cover of our database issue.

Printed on recycled paper and may be recycled.
Printed in U.S.A.

Contents

FOCUS

TECHNICAL

LITTLE SOLUTIONS

IBM Personal Systems Technical Solutions is published bimonthly by Personal Systems Competency Center, International Business Machines Corporation, Roanoke, Texas, U.S.A. Send any correspondence and address changes to *Personal Systems* at:
IBM Corp. Mail Stop 40-B3-04
One East Kirkwood Blvd.
Roanoke, TX 76299-0015

Personal Systems can be found on the Internet's World-Wide Web at: http://pscc.dfw.ibm.com/psmag/

Note to IBM personnel: IBM employees in divisions 12, 23, and 72 can subscribe to this publication using OV/VM and the INEWS facility: select hard copy for the PS_SOL category. IBMers in other divisions in the United States can subscribe via SLSS to GBOF-7532. IBMers outside of the United States can subscribe via SLSS to GBOF-6007.

BPA International Consumer Publication Membership applied for April, 1994.

Editor and Publisher
Betty Hawkins
(817) 961-7525
bhawkins@vnet.ibm.com

Business Manager
Donna Su
(817) 961-6268
psts@vnet.ibm.com

Editorial Assistant/ Reprint Coordinator
Lia Wilson
(817) 961-6267
lia@vnet.ibm.com

Database Administrator
Van Landrum
(817) 961-6436
vlandrum@vnet.ibm.com

Subscription Coordinator
Rose McAlister
(817) 961-7008
rmcalister@vnet.ibm.com

Production Coordinator
Elisa Davis

Publication Services, Typesetting, and Design
Terry Pinkston/Corporate Graphics
Arlington, Texas

Illustrator
Bill Carr
Dallas, Texas

Printing
Dave Willburn/Motheral Printing
Fort Worth, Texas

Editorial Services
Mike Engelberg/Studio East
Boca Raton, Florida

Executive Publishers
Beau Sinclair and Dan Russell

What's New For OS/2?

Remote Control for OS/2

The latest release of International Software Solutions' ***PolyPM/2*** remote control product has been renamed ***Remote Services Management*** and comes in three flavors:

- Remote Services Management Lite Edition
- Remote Services Management Advanced Edition
- Remote Services Management Professional Edition

The ***Lite Edition*** supports an OS/2 workstation (called a manager) controlling access to another OS/2 or Windows 3.1 workstation (called a client) connected through null-modem, modem, NetBIOS, internet packet exchange/sequenced packet exchange (IPX/SPX), and transmission control protocol/internet protocol (TCP/IP). This product is ideal for the casual home user who requires quick, easy access to another workstation such as an office PC.

The ***Advanced Edition*** supports an OS/2 manager workstation controlling access to DOS, Windows, or OS/2 client workstations. The connections are the same as the Lite Edition plus advanced program-to-program communications/advanced peer-to-peer communications (APPC/APPN), and X.25. This product is ideal for the corporate user with complex local area network (LAN) and wide area network (WAN) architecture.

The ***Professional Edition*** supports an OS/2 manager workstation controlling access to 16-bit or 32-bit OS/2 client workstations as well as support for DOS and Windows clients. The connections are the same as the Advanced Edition, plus the RSM script language is included for users requiring sophisticated, automated tasks such as software distribution.

For additional information, circle 1 on the Reader Service Card.

Analysis and Design Tool for OS/2

MultiQuest Corporation's ***S-CASE 2.0 for OS/2***, an object-oriented analysis and design tool, uses the Booch notation to graphically illustrate and model software systems. S-CASE satisfies the demand for visualizing complex software architecture necessary in today's demanding applications. You can generate high quality C++ code directly from the models. S-CASE allows iteration through the design, code, and test cycles while keeping the models and code synchronized. This iterative approach lets you enhance your application design through successive refinement without worrying about outdating your model.

S-CASE 2.0 features:

- Booch's latest notation (1994)
- Iterative C++ code generation
- Class specification reporting
- Real-time rule checking
- Hierarchical project management
- Heterogeneous multiuser support

For additional information, circle 2 on the Reader Service Card.

Centralized, Automated Systems Control of Distributed Environments

With its announcement of ***CONTROL-O/PC 2.0***, 4th Dimension Software Ltd. has enhanced its offerings of comprehensive solutions for distributed systems management within the data center and across the enterprise.

CONTROL-O/PC has been uniquely designed with a powerful set of facilities to successfully automate systems operations in a multiplatform environment:

- Expert systems-based rules automation engine
- Powerful graphical interface for automation design and systems management
- Comprehensive notification facilities
- Heterogeneous platform consolidation and management
- Multiple thread design

Running on an OS/2 workstation, CONTROL-O/PC enables you to quickly and easily automate complex tasks across a wide variety of platforms. Acting as a focal point for the enterprise, CONTROL-O/PC interacts with all components of systems management software to gather operations data, then uses this data to drive the data center's decision-making process. You can extend automation throughout the enterprise by implementing client/server technology.

CONTROL-O/PC includes completely automated inbound and outbound voice messaging to alert personnel of critical events at work or at home. This rapid notification enables you to respond quickly to problems, thereby minimizing down time. All functions can be handled remotely using fill-in-the-blank forms to define what to look for and what actions to take. CONTROL-O/PC's Definition Notebooks store instructions and rules to validate operator actions and ensure correct responses.

For additional information, circle 3 on the Reader Service Card.

Four Compilers for OS/2

Microway is now shipping four compilers for OS/2: ***NDP Fortran-77, NDP Fortran-90, NDP C/C++,*** and ***NDP Pascal***. Each language includes ***IBM's Toolkit***, with the IBM OS/2 ***WorkFrame*** optional. The ***NDP***

Pentium OS/2 compiler includes advanced numeric optimizations such as loop unrolling, numeric register caching, and numeric register coloring. Additional support for Pentium scheduling, the use of FXCH instructions to streamline x87 stack accesses, and a new peepholer that results in smaller code is also evident.

Included in the compilers is ***MGX***, Microway's device-independent, vector-based graphics and plotting package. It contains low and high level routines. The low level routines draw objects made up of lines, filled panels, ellipses, font-based characters, and lines of text. They can also create images that combine graphics and text. The high level routines produce two- and three-dimensional plots and charts.

The first 32-bit Fortran on the 386 market, NDP Fortran has excellent code generation quality. In addition to offering the standard global optimizations performed by a number of C compilers, it adds optimizations that improve numeric-intensive applications.

NDP C/C++ is a full AT&T 2.1 compliant C++ compiler, also translating the ANSI and K&R dialects of C as validated by Plum Hall. In addition to making it possible to build mixed applications that call Fortran or Pascal from C or C++, this product provides numeric optimizations not usually found in C compilers, such as loop unrolling and register caching.

For additional information, circle 4 on the Reader Service Card.

Easy OS/2 Desktop Protection—for Home or Office

Pinnacle Technology has released a new level of protection and administration for OS/2 computers. ***Kid Proof/2*** allows you to create an ideal desktop, take a picture of it, "hide" certain applications (or simply restrict capabilities such as copy and delete), and associate the desktops with certain users. When you sign on to an OS/2 machine with Pinnacle Technology's products, you get your desktop; other people, of course, get theirs–it's up to you, the administrator.

Parents can set up their machines to access their personal finance and other private applications, while their children can access–on the same computer–only their multimedia games, word processors, and the like without any capability to delete, shred, or even access the `C:` prompt. Kid Proof/2 protects your vital applications at home or at work.

Editor's Note: See the "Easily Load and Lock Desktops" article in this issue for more information about Pinnacle Technology's desktop protection products.

For additional information, circle 5 on the Reader Service Card.

File Distribution System for LAN Server

Client Server Networking's ***CONDUIT*** is a no-nonsense LAN file distribution system that facilitates administration, distribution, and inventory management for OS/2 and DOS administrators and clients.

CONDUIT automates tedious and time consuming desktop management of OS/2 and DOS platforms while defining and controlling data and program distribution.

Without a mainframe or mini-computer, CONDUIT can install, upgrade, control versions, and supervise licenses on any number of networked client personal computers. Using a unique network-independent design, CONDUIT operates with IBM, Novell, Banyan, Lantastic, and Microsoft products. Its built-in security uses identification and password safeguards, creates directories, remotely executes programs, develops client profiles, and maintains inventory.

Client profile data allows you to reconstruct corrupted `CONFIG.SYS`, `AUTOEXEC.BAT`, or `STARTUP.CMD` files. Daily virus scanning is easy, eliminating anxiety about losing mission critical data. Distribution tracking allows consolidated or separate logs. The log tracks user access, distribution status, and error conditions such as missing files, ID, password, or space.

For additional information, circle 6 on the Reader Service Card.

Checking and Watching LAN Server

Also from Client Server Networking, ***CHECKIT*** and ***WATCHIT*** optimize **IBM LAN Server** capacity and performance.

CHECKIT "takes charge" of 1,000+ servers in up to five domains using LAN Server (LS). You'll be the first to know when a failure occurs. Continually updated graphics display availability. You will be notified of any server failure during the polling process via a broadcast LS alert message and/or a phone call to your pager.

Now you can simply click on an icon to get the information formerly obtained by entering the conventional `NET STATISTICS`, `NET WHO`, `NET SHARE`, `NET SESSION`, and `NET VIEW` commands on the command line. At the click of a button, you can check up on a user's activity across multiple servers or verify an alias' availability.

To help you quickly and easily improve performance and anticipate capacity problems, WATCHIT automatically collects LAN Server capacity and performance statistics. Collection may be automated at the server or manually controlled from your desktop.

Thresholds will generate warning messages during data collection or notification via generic alerts, `NET.ERR` logging, or simple messages. WATCHIT graphs mean response, sessions, shares, bytes transferred, print jobs, big buffer, and request buffer allocations. Graphs direct you to the detailed information needed to improve performance and expand capacity.

For additional information, circle 6 on the Reader Service Card.

New Communications Software and OS/2 Warp Products

IBM has introduced three new terminal emulation products providing high performance, ease of use, and the most comprehensive data and application access available for users of OS/2 and OS/2 Warp.

The new IBM emulators allow a PC to seamlessly interact with host computer applications, which typically provide mission critical functions because of that environment's security and data integrity. When you run an emulator on OS/2, a true multitasking operating system, you can, for example, use the emulator to process host database transactions while a workstation application simultaneously prints a lengthy document.

Now joining IBM's family of emulators are ***Personal Communications AS/400 for OS/2, Personal Communications/ 3270 for OS/2,*** and ***Personal Communications AS/400 and 3270 for OS/2***. The first two products connect a PC with AS/400 and System/390 computers, respectively, while the third product offers connectivity to both host systems in a single package. Since these products also offer access to local area network applications, your users can benefit from greater desktop productivity.

These new, next generation software emulators for PCs were expressly designed to take advantage of the fast 32-bit architecture, multitasking, and multithreading capabilities of OS/2 Warp or OS/2 2.1 and 2.11.

The new 32-bit Personal Communications family for OS/2 supports the most robust features available in today's market, including dynamic data exchange (DDE) for linking data in different applications as well as data compression for moving large quantities of information across a network.

In addition to comprehensive host platform access and a robust feature set, the new emulators support credit card-sized Personal Computer Memory Card International Association (PCMCIA) adapters, a rich set of local area network and wide area network environments including NetBIOS, TCP/IP, IPX/SPX, and mobile computing through asynchronous dial-up connections.

For greater ease of use, IBM's Personal Communications family takes advantage of workstation graphical user interfaces to provide a consistent look and feel across the entire family. Furthermore, interface features familiar to the host user, such as AS/400 shared data folders, have been integrated for increased usability. Consistency across the family minimizes migration training when moving from older 16-bit operating systems to the next generation 32-bit OS/2 versions.

Developers can also realize greater productivity by taking advantage of the user interface and features in the Personal Communications family in combination with other IBM products. For example, developers can construct a user-friendly shell around the host application or integrate the application into a client/ server architecture using IBM's APPC.

For more information, contact your local IBM representative.

Tuning and Utility Kit for OS/2

Clear & Simple, Inc. has announced the third version of their book/disk combination called ***Performance Plus***, a tuning and utility kit for OS/2 2.X and Warp. Sections for everyone from the novice to the advanced user show how to get the best performance from an OS/2 system.

The companion diskette contains several OS/2 utility programs for performance tuning:

- `OPTIMIZER`–Displays performance-related `CONFIG.SYS` entries and recommends optimum values for your system.
- `DOS BLACKBOX`–Optimizes DOS/Windows application settings.
- `SIMPLECT`–Displays performance statistics graphically.

- STATS--Shows performance increases or decreases resulting from tuning adjustments, plus identifies bottlenecks.

Utility programs are also provided for viewing OS/2 bitmaps, saving the Workplace Shell desktop, creating an emergency boot diskette or partition, monitoring swap file growth, making automatic file backups, mapping disk drives, managing directories, and more. A copy of ***CPU Monitor Plus*** by BonAmi Software is included for monitoring CPU, memory, communication port, and disk usage. A bonus diskette contains a collection of OS/2 bitmaps.

For additional information, circle 7 on the Reader Service Card.

Loading External Images Under OS/2

A company specializing in converting documents to image-based systems, microMEDIA Imaging Systems, Inc., has introduced ***Visual Input***, an object-oriented software program that operates under OS/2.

Images scanned from microfilm, microfiche, aperture cards, slides, engineering documents, and other offline scanners can now be input into IBM's ***VisualInfo***. To accomplish this, Visual Input can use information from DOS, Windows, UNIX, and other systems. Document indexing can also be done offline. Specialized systems that automatically capture data can be operated independently from the OS/2 system. Additionally, low-cost key entry from non-U.S. providers is supported.

Visual Input is easy to use. Functioning in the background, it can be scheduled to operate when system usage is low. Multiple OS/2 clients can simultaneously load images and indices. Extensive internal controls monitor the process in detail. If you have unique or high volume document input requirements to IBM's VisualInfo, you can avoid complex and expensive integration by using microMEDIA's Visual Input program.

For additional information, circle 8 on the Reader Service Card.

Easier and More Productive OS/2 Desktop

The latest version of SmartSuite, ***SmartSuite 2.0 for OS/2*** from Lotus Development Corporation, comprises the new productivity tools pioneered in ***Value Pack for SmartSuite for OS/2*** and the latest versions of Lotus' four native 32-bit OS/2 applications: ***Ami Pro 3.0b, 1-2-3 2.1, Freelance Graphics 2.1***, and a ***cc:Mail Desktop 1.03 for OS/2*** Workplace Shell single-user license and software.

Featuring the innovative and well-received Lotus ***SmartCenter for OS/2***, Value Pack boosts your day-to-day efficiency by tightening the integration between SmartSuite applications and the Workplace Shell, as well as among the SmartSuite applications themselves. SmartCenter for OS/2 provides a central location for launching and switching among SmartSuite applications, other applications, and desktop tools. It also exploits the OS/2 Workplace Shell to deliver additional benefits such as drag-and-drop functionality for customization and easy access to OS/2 tools such as Find and Lockup.

For additional information, circle 9 on the Reader Service Card.

OS/2 Imaging Toolkit

AccuSoft has announced the first independent development tool for reading Kodak Photo CD image formats. The entire line of AccuSoft's ***Image Format Library*** now includes not only Photo CD but also PhotoShop, ASCII to raster, MacPrint, Windows ICO, Sun Raster, X-Windows, Brooktrout, IOCA, and 28 other formats. Application developers can now easily incorporate high performance raster imaging capabilities for 36 formats into their own products with this software library.

The library, which comes in OS/2, DOS, NT, Macintosh, Visual Basic, and UNIX versions, now has even greater performance and capability, while still supporting the very popular raster image formats: JPEG, TIFF, PCX, DIB, TGA, GIF, WMF, PICT, DCX, WPG, EPS, BMP. Also added is a new color reduction technology, automatic thumbnails, new compression algorithms, and much more.

For additional information, circle 10 on the Reader Service Card.

OS/2 Warp Unleashed

The latest in the ***OS/2 Unleashed*** series of books published by SAMS Publishing, ***OS/2 Warp Unleashed*** is written by 14 worldwide OS/2 experts, including featured authors David Moskowitz and David Kerr. According to Productivity Solutions' president and author David Moskowitz, "We created a book which we believe is unique among the Warp books in that it contains the most thorough and comprehensive information about Warp. OS/2 Warp is by far the best OS/2 product yet. There are still lots of 'undocumented features and issues' which we uncovered in our testing and from thousands of users with whom we communicate online. We attempted to cover as many of these as we could at the time we went into production."

The 1,205-page book has 20 chapters, a foreword by Lee Reiswig, president, IBM Personal Software Products, and an appendix containing worldwide bulletin board contacts in 19 countries and user resources for online information, publications, and technical support. A companion CD-ROM contains over 100 MB of commercial demonstration software, test-drives, utilities, shareware, sample REXX programs, and resource information.

OS/2 Warp Unleashed: ISBN 0-672-30545-3. For more information about SAMS Publishing, a division of Macmillan, circle 11 on the Reader Service Card.

Navigating the Internet with OS/2 Warp

The latest book from author Herb Tyson and SAMS Publishing, ***Navigating the Internet with OS/2 Warp***, is a step-by-step guide to using OS/2 Warp's new BonusPak Internet Access Kit (IAK). It covers connecting to the Internet using the IBM Internet Connection (Advantis) and other Internet providers as well as the BonusPak tools provided for using the service: Ultimedia Mail/2, NewsReader/2, Gopher, WebExplorer, FTP-PM, and Telnet. The book also documents other BonusPak Internet tools such as ping, nslookup, hostname, and finger.

The book is an excellent companion for IBM's OS/2 Warp BonusPak, providing detailed installation, configuration, usage, and troubleshooting advice for all OS/2 Warp users.

Navigating the Internet with OS/2 Warp: ISBN 0-672-30719-7 or IBM order number SR28-5665. For more information about SAMS Publishing, a division of Macmillan, circle 11 on the Reader Service Card.

Using OS/2 Warp

The fourth book in Que's *Using OS/2* series, ***Using OS/2 Warp*** is a comprehensive guide on the productive use of OS/2. Author Barry Nance is a columnist for *BYTE* Magazine, a programmer, and the author of the popular books *Introduction to Networking* and *Client/Server LAN Programming.* He has used OS/2 extensively since 1987.

Using OS/2 Warp includes OS/2 basics as well as advanced Warp topics. A tutorial and a features-and-functions reference are also provided. The book begins with the configuration and installation of OS/2 and then discusses troubleshooting, the Workplace Shell, the command line interface, the Drives object, built-in applications (including the BonusPak), using and tuning the DOS and Windows (WinOS2) components, and printing. Suggestions for installing and using additional applications are included, as are REXX programming lessons for those who want to take charge of their OS/2 environment.

Using OS/2 Warp: ISBN 0-7897-0088-3. For more information about Que, a division of Macmillan, circle 11 on the Reader Service Card.

Inside OS/2 Warp

Inside OS/2 Warp, Version 3, the latest in New Riders Publishing's *Inside OS/2* series, is both a tutorial and a reference for intermediate to advanced OS/2 users. One of several authors, Mark Minasi is a consultant and educator in PC technology and advanced operating systems. He is a contributing editor to *OS/2 Magazine*, *Compute, AI Expert,* and *BYTE.*

The book helps you optimize your system, providing details not found in the OS/2 manuals but necessary for fine tuning. Topics range from common procedures such as planning for and installing OS/2 to more advanced subjects such as using systems enhancement utilities and Internet access. The book is arranged in five sections: Getting Started, Working with OS/2, Exploiting OS/2's Power, Optimizing OS/2, and Troubleshooting OS/2. A companion disk contains the latest OS/2 shareware and utilities.

Inside OS/2 Warp: ISBN 1-56205-378-7 or IBM order number SR28-5655. For more information about New Riders Publishing, a division of Macmillan, circle 11 on the Reader Service Card.

Programming the OS/2 Warp GPI

This new book from John Wiley & Sons addresses the graphics programming interface (GPI) for OS/2 Warp, providing step-by-step instructions and accompanying coding samples. Written by Stephen Knight and Jeffrey Ryan, ***Programming the OS/2 Warp GPI*** covers the full range of OS/2 GPI functions:

- Drawing primitives
- Working in coordinate spaces
- Character fonts, metrics, sizing, and positioning
- Metafiles
- Transformations

It contains over 60 screen shots, drawings, and tables and comes with a disk including a graphics editor application, text browser file, query printer information, and other utilities.

Programming the OS/2 Warp GPI: ISBN 0-471-10718-2 or IBM order number SR28-5681. For more information about John Wiley & Sons, circle 12 on the Reader Service Card.

OS/2 Warp for Dummies

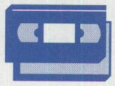
OS/2 Warp for Dummies by Andy Rathbone is IDG Books' latest addition to their best-selling *Dummies* series. This 336-page book is written primarily for new OS/2 users but also includes some tips and techniques for experienced users.

The book covers:

- Setting up the Workplace Shell desktop
- Running DOS and Windows programs under OS/2
- Using OS/2 Warp's free applications: LaunchPad, BonusPak, and the IBM Internet Connection
- Retrieving lost programs and files
- Creating integrated documents

OS/2 Warp for Dummies: ISBN 1-56884-205-8 or IBM order number SR28-5675. For more information about IDG Books, circle 13 on the Reader Service Card.

OS/2 Warp Videos



Getting Started with OS/2 Warp, produced by ViaGrafix, a supplier of computer training products about OS/2 and related products, is one of a series of four training videos for new users of OS/2 Warp. All installation and setup tasks are demonstrated with on-screen tutorials, and a learning disk is provided for hands-on experience.

Some of the topics covered on the video include:

- Opening a folder, using pop-up menus, viewing applications
- Using the LaunchPad, changing the Lockup Image
- Changing sound options, object titles, program names
- Creating, changing, and deleting objects
- Copying and moving files and folders and using the Drives folder
- Installing DOS, Windows, and BonusPak applications
- Using Dual Boot and Shutdown
- Maintaining floppy and hard disks

Other videos in the series are *Learning OS/2 Warp, Advanced; Using Internet with OS/2 Warp*; and *Using OS/2 Warp for Multimedia.*

For more information, circle 14 on the Reader Service Card.

The Soap Box Derby

By Todd Watson

Todd Watson's extensive forays into the wilds of cyberspace over the past year were taken very seriously by IBM's Software Solutions Division in Somers, N.Y.–so much so that it recently hired him on as its "electronic media communications specialist." Although he's not yet sure what dangerous undertakings his new job will entail, his virtually adventurous life is certain to become more cybersobering by the light year. In fact, it sounded so serious that Todd decided he'd better step up on his soapbox and poke some fun at advertising campaigns–everyone's–before he himself becomes a monument to Rodin's pensive thinker. Nothing is sacred–Todd leaves no stone unturned in his pursuit of promotional puffery.

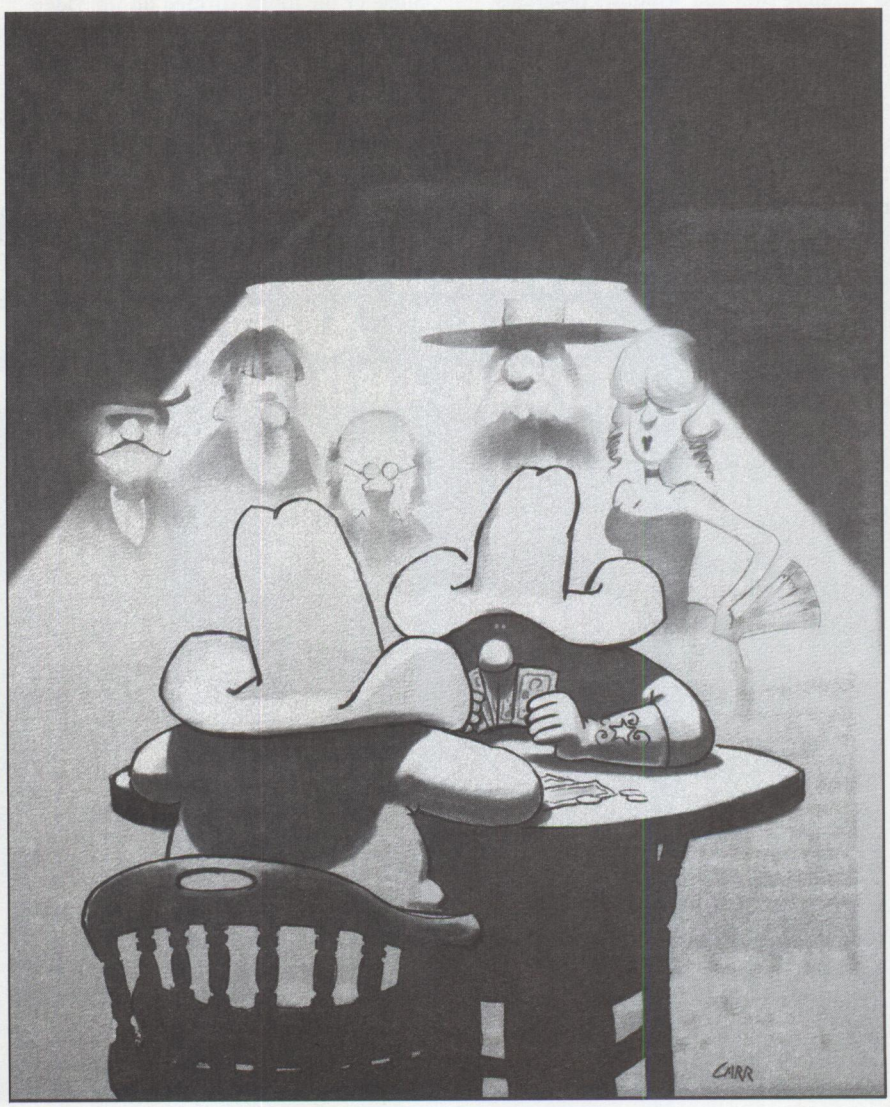
Just last year IBM shucked its forty some-odd advertising agencies in favor of consolidating its ad dollars with one behemoth agency, the celebrated Ogilvy and Mather, which, before the transition, was in charge of Microsoft's campaign to billboard the universe. Co-founded by advertising maestro David Ogilvy, who once said that you can't bore people into buying your product, O&M has proven that IBM is like a Timex watch–it takes a. . . well, you know the rest.

The change couldn't have come at a more crucial moment. The continuing contest for guiding the future of desktop, client/server, and enterprise systems has never been more zealous. Strategies are being molded around their evolution; careers will be made and broken because of their consequences.

That said, everybody likes a worthy opponent, including those of us at Big Blue. Healthy competition can go a long way towards helping manufacture success–for our own technologies as well as those of our competitors. None of us lives and works in a vacuum, and competition can help pull our respective sides together as a team during times of turbulence, during periods when rational thought would suggest we should instead be tearing one another limb from limb.

In the truest tradition of an open and competitive free market, rivalries between giants such as Microsoft "Where Do You Want to Go Today?" Corporation, Intel "Do You Have the Power?" Corporation, Digital "I Don't Know Their Slogan But I Do Remember Their New TV Commercials Because They're Like a Cross Between MTV and *NYPD Blue*" Equipment Corporation, and the like are good, clean, spite-ridden, bloody-knuckled American fun. Despite the fact that I was never a scrapper in grade school, instead firmly adopting that age-old maxim that the pen (okay, the word processor) is mightier than the sword, I've become quite impassioned of late about IBM's hard-target marketing and am ecstatic that we seem to be getting roused up for the significant, down-in-the-trench skirmishes ahead.

To kick things off, there's been the recent reappearance of the computer company branding campaign. Establishing a brand identity is a Sisyphean chore, I'm sure.



More than a few uptown Manhattan advertising executives have pondered such an assignment over too many martinis at Elaine's and have come up dry, not to mention soused. Instead of selling a product, they must sell an image, a concept, an *idea* of who a company is and what it represents. There's no there *there*, really. But unlike peddling snake oil out of a covered wagon, there should be some substance behind the slippery veneer of hyperbole.

The word on the street (Madison Avenue, actually) is that the IBM Solutions for a Small Planet campaign has become a big hit, both in the advertising realm and in the computer industry at large. What's not to like? They're sophisticated, worldly, *accessible* ads. The ads show ordinary people talking amongst themselves in a vacuum of high-tech *patois*, discussing technological solutions that work beyond physical, geographical, and cultural boundaries. Since the IBM solutions target specific environments and accompanying problems, they invite the audience along for an exciting ride into today's computing, rather than into the ostensibly distant future. While doing so, they keep the wistfulness in check and instead focus on the tangible. And it appears IBM is laughing all the way to the Arbitrons (especially after the oh-so hip AS/400 surfing commercial coup).

Then Microsoft, in its own attempt at establishing a firmer brand identity in the minds of computer users worldwide, began asking that rather open-ended question–"Where do you want to go today?"–with its latest broadcast and print brand campaign. I wonder what Redmond's Windows95 programmers' answer to that question might be. Considering their latest vaporware schedule, it looks as if they've been going anywhere else but to work at the labs in Bellevue. August, they say? Don't hold your breath, if Buggy Beta M8's recent hostile press reception was any indication. And then to have Mr. Gates stand before the computer press corps and announce that 95 is essentially nothing more than another hopscotch square on its way to a full-blown Windows NT!

I may just be a sore underdog, but the Microsoft brand campaign seems to be very similar to the company's software strategy–it sounds like a good idea because it's just chaotic enough that nobody has time to stop and figure out exactly what direction they're going, or when they're going to arrive (with the exception, perhaps, of Judge Sporkin). It's sort of like catching a subway on your first trip to New York City. You have absolutely no idea what you're doing or where you're likely to end up, but you'd better jump on fast or you'll get left behind–or worse yet, end up staring at the cowcatcher of a runaway train.

Where *do* you want to go today?

Hey, Paris would be nice, seeing as I've only experienced the European continent through my extensive virtual travels. I know, I know, I'm just another ignorant, unilingual, ethnocentric, chauvinistic, paleolithic American male whose one trip outside the U.S. landed him in southern Mexico just in time for last year's uprising in Chiapas. That experience convinced me that had I been in Iran in '79, I would have been one of the first hostages taken.

Of course, Venice wouldn't be so bad either. Ever since I saw a young Diane Lane in *A Little Romance,* I've been a hopeless romantic, waiting impatiently for the day to arrive when I could kiss my sweetheart in one of those gondolas under the Bridge of Sighs just as the sun is setting. Just thinking about it, I get that warm, mushy feeling poets can describe so well.

Then again, a trip to Seattle wouldn't be half bad, either. There seem to be a lot of "way hip" young people doing a lot of "way hip" things in that particularly "way hip" city at the moment, and I feel "way left out." Of course, it's so "way close" to Redmond that I might break out in competitive hives, so maybe I'll just stay home after all–there's always Virtual Seattle: `http://www.seattle.is.a.way.hip.place.to.be.hip`.

But like my daddy always said, what I want and what I get are two completely different things. Today, like any other, where I *am* going is to the office to slave away in front of my PC. Another day, another *dinero*. American Express won't let me leave home without paying their bill, corporate card or not, so that means there are articles to write, people to call, products to promote, magazines to produce, articles to write–you get the picture. So where I want to go today has nothing to do with the fact of where I'm likely to end up.

It was probably very similar with the Edsel. Remember the Edsel? Ford Motor Company does, even though they've spent the past 37 years trying to forget about it. Ford (where "Quality is Job One") introduced the 1958 Edsel and tried to pass it off as the car of the future. Named after Henry Ford's son, the Edsel had it all: double headlights, gull-winged rear deck, horse-collar grill, button-operated automatic gears–the works. Everything you could possibly hope for in a '58 car, yet hardly anyone bought it. This was supposed to be the vehicle that would change driving as we knew it. So what happened?

Many observers argued that people just weren't ready for the future. Others pointed out that the car was fabulously ugly. Whatever the reason, the fact still remains that the Edsel was one of the biggest flops of the century. It certainly didn't help matters when, on a live Bing Crosby TV special (which, oddly enough, marked that crooner's transition from radio to TV), Rosemary Clooney's Edsel wouldn't crank!

They say history has a way of repeating itself.

Despite the massive campaign about to be launched by our contemporaries in Redmond, all the activity on the press front suggests that this time the proof had better be in the pudding. Running media interference will be a moot point because people have an alternative–and quite an efficient one, I might add.

See, the movie marketeers in Hollywood have known for decades what mass media-centric advertising agencies seem to so often forget: that empirical, word of mouth promotion is the single most powerful form of persuasion. A personal endorsement from a neighbor down the street or a colleague down the hall carries much more weight than an overstated headline from a print ad or a sound bite from a TV commercial. Likewise, one bad word from a few million good friends can send what was supposed to be the next *Jurassic Park* into video store purgatory.

So here's my proposal. If you use OS/2 Warp, if you believe in it as a product and a philosophy (Zen and the Art of Warping), as do I, if you use it every day and feel as though it makes you a more productive individual, then go and tell somebody. It's that simple.

Better yet, approach everybody you know who uses a personal computer–whether they run Windows 3.1, DOS 3.0, Linux, or even C/PM–and tell them all about OS/2 Warp. Tell them about its multitasking and multithreading, about its BonusPak, about its easy-to-use connection to the Internet, about how they can run their native Windows applications, no problemo. And all for the price of less than three megs of RAM.

Or even better yet, invite all your friends over to your house. You know, the ones who are absolutely terrified of getting another general protection fault after they've spent days doing an itemized tax return on their PC. The ones who call you in the middle of the night to ask if there's anything they can do to save the letter they were writing to Aunt Edna, which disappeared into thin air. The ones who send you angry e-mail missives about how they're never going to use their "stupid" computer again.

Make it a *soireé*. Invite them over for a popcorn and soda pop TupperWarp party and show them firsthand what crash protection actually means. Open four or five or even six programs, call up the Internet Connection's WebExplorer and go surfing, print your Great American Novel, and copy a file or two. Make sure you have all the windows open so your audience can watch all this actually going on at once.

Seeing is believing. Turn the concept of multitasking into a tangible demonstration. If you do it right, you're likely to receive a reaction similar to the one received by members of the Silicon Valley Homebrew Computer Club who, way back in 1975, stared in amazement at the Altair 8800. Yes, they may just have been a bunch of blinking lights, but those lights were illuminating the path that led into the future of computing technology.

Finally, tell them that while they wait for the construction of the Great Ark that's supposed to save them from the rising flood of 16-bit misery, they're missing the getaway speedboat.

Believe you me, there's going to be advertising galore. TV commercials will air on every channel from HBO to MTV to Bubba's TV Trash Talkathon. Fancy print spreads will run in major papers around the globe. The World-Wide Web will be riddled with newfangled, fancy promotions designed exclusively for the purpose of sucking your brain into the inner abyss of cyberspace.

But don't buy it. Like New Coke back in 1985, it's all a bunch of hype.

If I were the creative director on this campaign, here would be my scenario for IBM's media riposte:

It's a dark and stormy night. We're in an old house way out by the edge of town. A three-day poker game is in the final stretch. The other players got out hours ago. It has all come down to these two contenders. Two cowboys.

The players sport 72-hour shadows. Their eyes are weary and have huge, black circles underneath their lids. The pack of curious spectators standing around the table watches hungrily. They're dying to see who's going to come out on top.

As the dealer shuffles the cards for the final hand, the two adversaries look deep into one another's eyes–cautiously, but with great respect.

All the money is on the table.

The cards are dealt. The two cowboys concentrate, trying to figure out whether or not the other side is bluffing.

The cowboys look over their hands, studying them carefully, then glance suspiciously back up at their opponent. The betting begins and, after raises from either side, just as quickly concludes.

"Whatta ya got?" the one cowboy asks.

The other smiles a knowing smile and lays down his cards.

"Four fours," he says, savoring the moment. "Whatta *you* got?"

A long pause, then the other cowboy fans out his cards. "Four eights."

But this is old news. IBM laid its hand down a long time ago.

Todd Watson, formerly the assistant editor for *SQ: IBM's Magazine of Software Technologies* and */AIXtra: IBM's Magazine For AIX Professionals*, has been with IBM since 1991. Now an electronic media communications specialist for IBM's Software Solutions Division, Todd holds a BA degree in English and an MA in mass media studies from the University of North Texas in Denton. He can be interactively a) manifested b) reached c) haunted d) obliterated at radar@vnet.ibm.com.

New! The Desktop Observatory 4.0

Easily Load and Lock Desktops

By Charles Dircks

Centrally controlling OS/2 is a major issue for administrators of large local area networks (LANs). Many LANs need OS/2's flexibility and power but can do without the risks brought on by allowing users complete access to the system resources. Associating particular desktops with particular users (or groups) solves many of these issues. More security, however, is desirable.

Needing to administer and regulate desktops, a south Texas insurance company evaluated several options. The critical need to "lock down" desktops prompted this company to choose The Desktop Observatory from Pinnacle Technology. This article discusses the insurance company's requirements and decision process.

The south Texas insurance company's basic requirement is to centrally create and distribute an OS/2 desktop. The nature of the project, however, requires urgency and, of course, protection of information assets.

Finder

The Desktop Observatory Version 4.0 allows administrators to create a desktop for a given group of users by simply snapping a picture of the desired desktop and saving it to a central file. This is performed by a powerful system object model (SOM) interface that Pinnacle Technology calls the "Finder" (Figure 1). After "taking the picture" of the required desktop, the administrator can easily apply restriction settings, hide objects entirely, or manage "child" objects within folders.

The Finder performs several critical functions. It finds all the objects on the desktop (and within folders). It can filter (hide) applications or objects that are not appropriate for various groups of users. Does an accountant, for example, really need access to the OS/2 System folder? If not, simply click on Hide before you save your ideal desktop for all accountants. The file is then saved centrally to be associated with user IDs and delivered over a LAN.

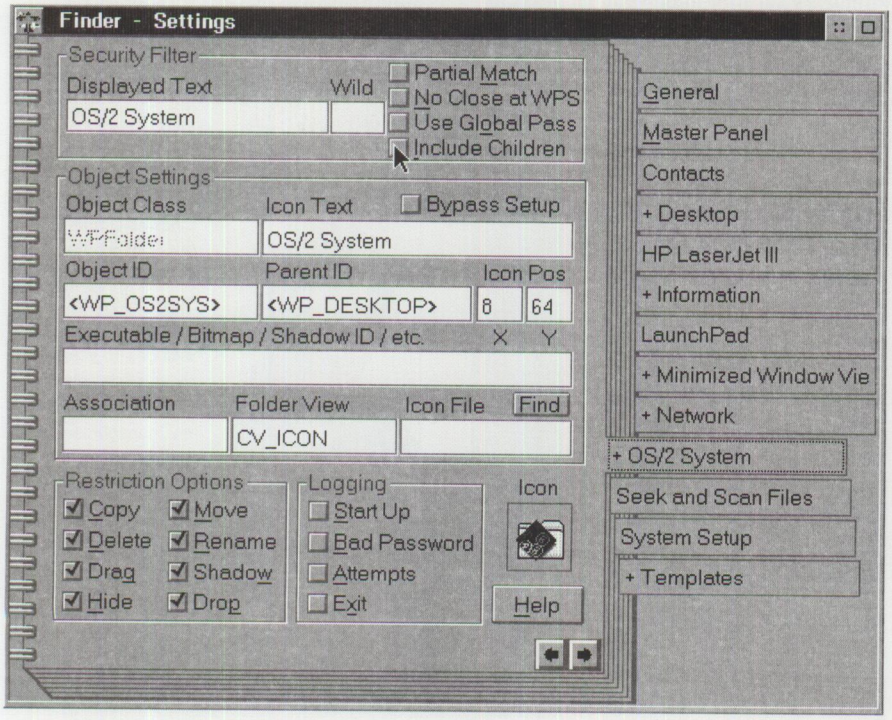


Figure 1. Pinnacle's Finder Object

On client machines, the Finder object filters objects that don't meet the administrator's criteria. The insurance company wants all of its clients to come up with a LAN logon icon. Pinnacle's Finder identifies all the objects on the client, including the OS/2 predefined objects, and hides them. The logon icon, of course, makes it through the filter.

Desktops on the Fly

With The Desktop Observatory, this Texas firm can build an appropriate desktop on any OS/2 workstation. The Desktop Observatory files are small (about 20 KB) so they don't impact network bandwidth, and, because the insurance company wants to secure laptops, The Desktop Observatory can even deliver desktops over a phone line. This becomes important as the work force becomes more mobile.

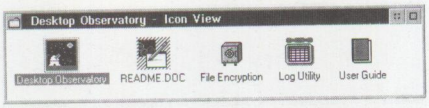


Figure 2. The Desktop Observatory Security Features

Many utilities can back up `.INI` files or use REXX utilities combined with other programs. Requirements for true central control and speed, however, are problematic when using `.INI` files or REXX strategies to deliver desktops. Most of us don't want to become REXX programmers for life! An off-the-shelf offering such as The Desktop Observatory reduces REXX maintenance and speeds delivery.

Real Security Options

Another important feature is security. If the requirements of this south Texas insurance company stop with creating and distributing desktops from a file server, Pinnacle probably would suggest The Desktop Commander, a low-cost, off-the-shelf product that includes Pinnacle's Finder technology.

Like many companies, the insurance company not only wants to exclude users from some applications, it also needs to exclude users from specific files and user directories. For example, few users need to access the `CONFIG.SYS` file. By restricting access to certain system files, administrators can increase system security. The latest release of The Desktop Observatory includes the powerful security daemon found in earlier versions.

In version 4.0 of the Observatory, the filters are simply set when creating a desktop to be associated with a given group. For example, users from Personnel will have Personnel information on their OS/2 clients. If someone not cleared to access Personnel information attempts entry, the attempt can be knocked down and logged based on user ID and machine. A warning message can also be sent to the unauthorized user.

Pinnacle also offers power-on/boot protection that can be centrally administered, if necessary (Figure 2). This feature provides password access to all data on the computer, eliminating the possibility of unauthorized access to mission-critical data.

Pinnacle Technology

Pinnacle has been delivering OS/2 centralized security products for over three years. These products include:

- *Kid Proof/2*–Desktop security for stand-alone machines (kids get their own desktop when they sign on–not yours!). Figure 3 shows the Kid Proof/2 administration controls.
- *The Desktop Commander*–Networked desktops, no security daemon.
- *The Desktop Observatory*–Networked desktops, sophisticated security daemon.

Products are upgradable and volume licenses are available.

Managed desktops require OS/2 2.1 or higher. Administrative desktops using Finder require OS/2 Warp.

For more information, circle 5 on the Reader Service Card, call Pinnacle at (317) 581-6262, or write to Internet ID `ehenning@pinnacletech.com`.

Flexibility

Pinnacle's products are very flexible. What if you simply don't know your security requirements? Then use The Desktop Commander and upgrade to The Observatory as requirements change.

What if your organization needs desktop delivery and reliability without completely limiting a user's individuality, such as choosing bitmap backgrounds or icon placement? In this case, simply use Pinnacle's Machine Independent Desktop (MID) capabilities built into The Desktop Observatory. Regardless of where your users log on, they get their customized desktops.

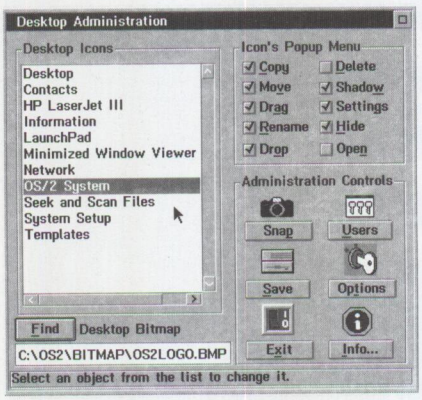


Figure 3. Kid Proof/2 Administration Window

Summary

The south Texas insurance company centralized their LAN administration with The Desktop Observatory. Now they save time by centrally customizing all workstations, thus protecting mission-critical data with The Desktop Observatory's security features.

The Desktop Observatory is used by companies worldwide to address these same critical desktop needs: central desktop administration, security, reduced REXX creation and maintenance, and speed. Over the long term these companies can upgrade as security needs and capabilities evolve.

Charles Dircks is the chief security architect for Pinnacle Technology, Inc. His background includes 15 years focusing on security, bandwidth, and central management issues within the airline industry. He was a featured speaker at the 1995 OS/2 Technical Interchange. He has a degree in Computer Science from Indiana University.

Road Trip! Cruisin' to the Olympics

By Van Landrum

This issue's road trip takes you to the Olympics home page. Drop in from time to time during the next year or so of preparation, then hang out for the results while the games are going on.

Even though the Olympics are 464 days away as I write this, we can still monitor progress towards the opening day. Let's take a road trip to Atlanta, the home of the 1996 Olympic Games.

After hearing about the new Olympic home page, I jumped onto my WebExplorer and headed for `http://www.atlanta.olympic.org`. The Olympic home page features a colorful image map announcing the event as the 1996 Centennial Olympic Games, the torch symbol for the Atlanta Olympics, the IBM logo (IBM is the Official Internet Information Systems Provider for The Atlanta Committee for the Olympic Games), and a countdown of days until the 1996 Olympic Games (Figure 1).

When I select Welcome, a picture of Billy Payne (President and CEO of The Atlanta Committee for the Olympic Games) displays along with a video and sound files. One of the sound files contains almost a minute of welcome from Mr. Payne. It's nice to see Internet's video and sound capabilities put to good use.

Further down the page is a link to an article on how the web server for the Olympic Games was started. The article tells how millions of people from all over the world will be able to take a ride on the information superhighway and stop at the 1996 Centennial Olympic Games. The web server staff plans to provide "a wealth of continuously updated facts, figures, photos, illustrations, video, and audio content–all aimed at providing the latest news possible on the what, when, and where of all the sports as well as how to buy tickets and be a part of the excitement." When you think about it, what better information to put on the World-Wide Web than information about world-wide interest in the Olympic Games?

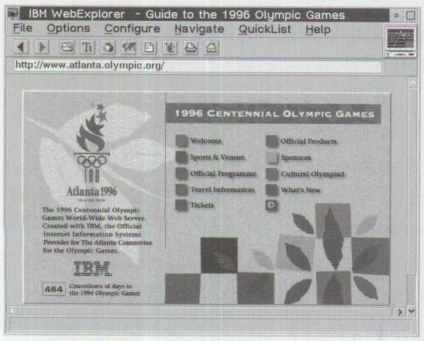


Figure 1. Olympic Home Page

Delivering this information are powerful, high performance, parallel computers from IBM. "The 1996 Olympic Games Server is a great example of how organizations and companies can benefit from the power of the Internet," said John Patrick, IBM vice president of Internet applications, in his keynote speech at Internet World. "Information on the 1996 Olympic Games is in high demand around the world. By putting it on the Internet, the data will be available to more people than ever before . . ."

1996 Olympic Games Server

The 1996 Olympic Games Server presents an array of information under nine major headings:

Welcome–both video and audio greetings from Billy Payne, the 1996 Olympic Games at a Glance, answers to frequently asked questions, and current news items.

Sports & Venues–explanations of the 1996 Olympic Games' sports plus descriptions and photos of the facilities hosting the events, including seating capacity and dates of events held in each facility.

Official Programme–schedule of Olympic events by sport, day, or location.

Travel Information–information on the city of Atlanta, its accommodations, and its transportation system.

Tickets–information on the more than 11 million Olympic Games tickets available, including prices, ticket brochure, ticket sales times, and customer services.

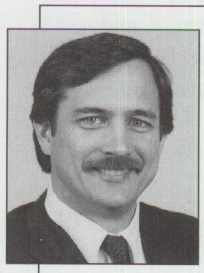
Official Products–information on products such as commemorative bricks, coins, countdown T-shirts, and official merchandise catalogs.

Sponsors–list of 1996 Olympic Games sponsors and a web page for each. Hyperlinks are provided to the corporate web pages when available.

Cultural Olympiad–information on the 1996 Olympic Arts Festival and other Cultural Olympiad programs.

What's New–the latest press releases from Atlanta Committee of the Olympic Games (ACOG)–with new releases added to the 1996 Olympic Games Server the moment they are issued–and a look at key dates ahead in the countdown to the summer of 1996.

This year, take an occasional trip to the 1996 Olympics home page and monitor the pre-game activities. Next year it will be a great place to get up-to-the-minute results of the actual games.



Van Landrum is a marketing support representative in the IBM Personal Systems Competency Center in Roanoke, Texas. Database administrator for *Personal Systems* magazine, */AIXtra* magazine, and the Technical Coordinator Program, he is also responsible for the *Personal Systems* magazine Home Page on the World-Wide Web. Van has a BBA degree in Business Computer Information Systems from the University of North Texas in Denton. His Internet ID is `vlandrum@vnet.ibm.com`.

DB2 for OS/2 V2.1: The Next Generation

By Bill Wong

This issue of* Personal Systems *focuses on the latest features of DB2 for OS/2. In the following article, Bill Wong summarizes these features.

Today, DB2 for OS/2 enjoys the position of market leader for OS/2 databases. With hundreds of thousands of licenses distributed worldwide, no other OS/2 database comes close to DB2 for OS/2's acceptance as a foundation for delivering mission-critical applications.

Some of the reasons so many organizations trust DB2 are the quality and reliability features IBM designs into all of its products, regardless of the platform. Performance also plays a vital role in DB2 for OS/2's success–the DB2 database program continues to be optimized for the OS/2 environment. This strategy allows thousands of organizations to continue to use PCs as servers rather than migrate to much more expensive UNIX-based solutions.

DB2 for OS/2 provides the best performance on OS/2 in the industry today. The new DB2 for OS/2 V2.1 database continues to extend its performance leadership by introducing the industry's best optimizer (Figure 1). An article in this issue, "Performance Enhancements in DB2 for OS/2 V2.1," reviews the new optimizer, the new storage architecture, and new performance enhancements available for distributed environments.

This version introduces database features that provide a foundation to build object-oriented applications. See "Getting Object-Oriented" for information on how to manipulate large objects such as binary large objects (BLOBs), character-based large objects, user-defined functions, and user-defined data types (Figure 2). IBM's strategy is to push more function into the database domain so that the database can manage both the application and data logic, should the developer require it.

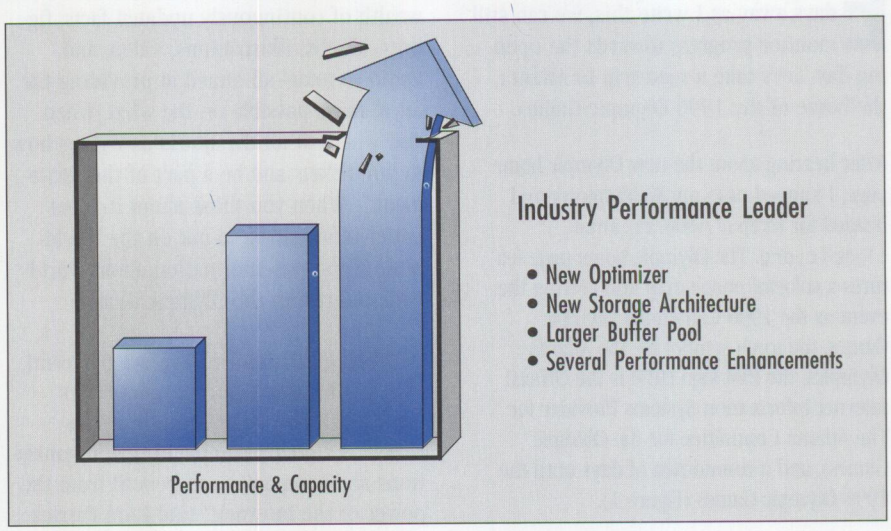


Figure 1. Extending Performance Leadership

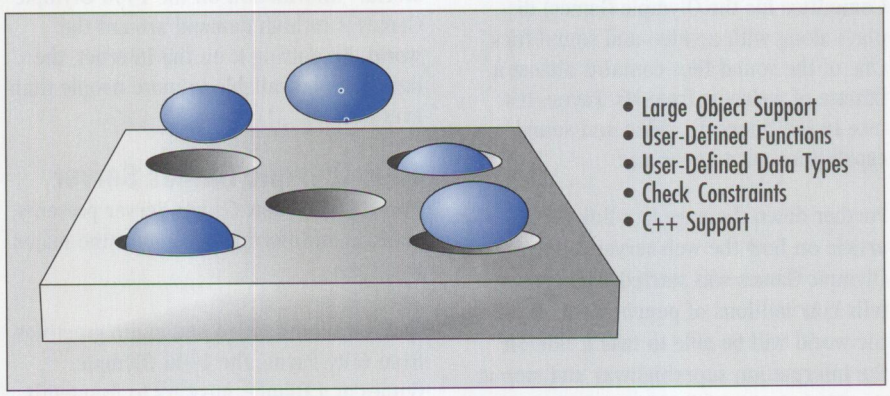


Figure 2. Object-Oriented Features

Several new SQL enhancements have been included to conform to the ANSI SQL3 standards. These are topics of another featured article, "Enhanced SQL," that reviews some of the advanced SQL features including triggers and recursive SQL support that models hierarchical or network relationships. The recursive SQL (Figure 3) is an example of DB2-exclusive technology that addresses business problems that previously remained outside the domain of relational technology, such as bill of materials (BOM) applications.

An article focusing on connectivity, "Enterprisewide Connectivity Using DB2," reviews the new distributed features, new stored procedures, and new packaging options (Figure 4). Distributed Data Connection Services (DDCS) V2.3 continues to provide the best and lowest cost alternative for distributing data.

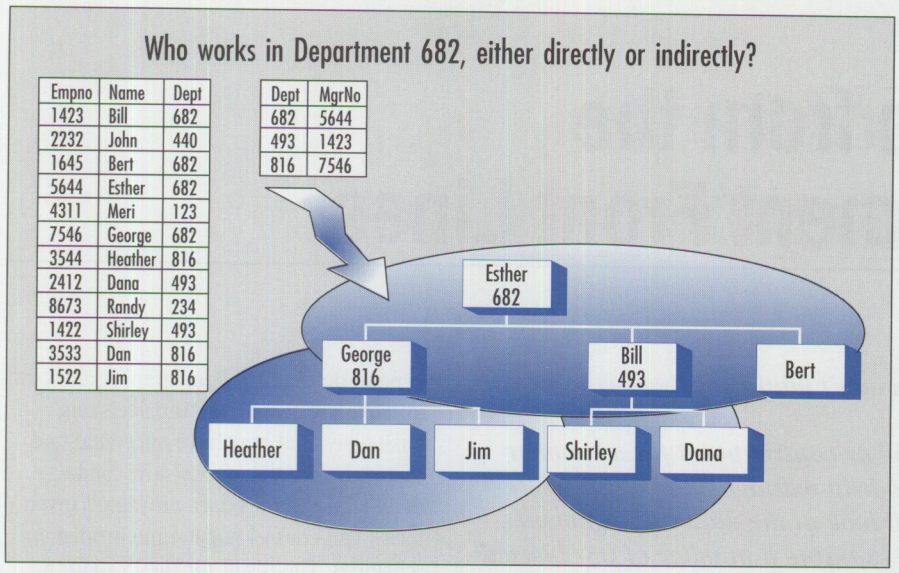


Figure 3. Recursive SQL

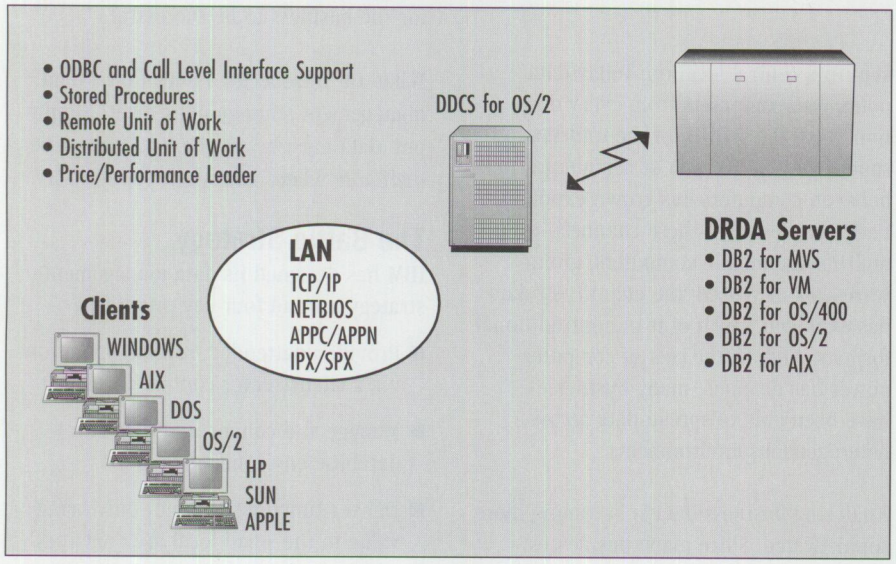


Figure 4. Connectivity

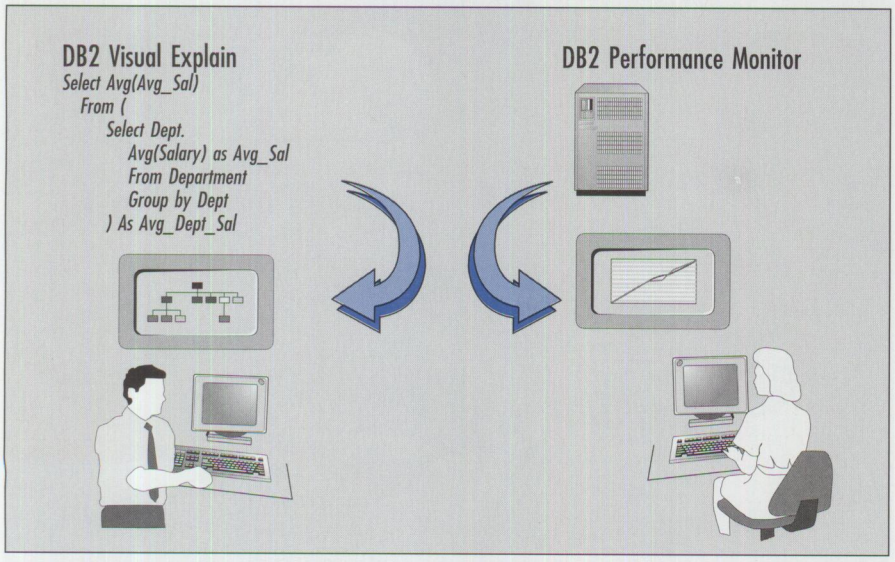


Figure 5. DBA Tools

Tools

Several new tools and utilities can assist with managing the database either locally or in a distributed environment. Enhancements to existing utilities have also been made to improve usability and performance. These new tools for DB2 for OS/2 are shown in Figure 5:

- Visual Explain
- Performance Monitor
- DataPropagator Relational (replication services)
- DataHub (systems management)

We've highlighted these new tools in this database-dedicated issue. Expect to read more about database tools in future issues of *Personal Systems*.

Summary

DB2 for OS/2 V2.1 and DDCS V2.3 represent a significant advance in database technology. Users and developers will find performance enhancements, the capacity to manage larger databases, new distributed features, several new tools, advanced SQL features, and many object-oriented features to address the accelerated business demands of today's organizations.

We hope you enjoy this issue dedicated to the many advanced features of DB2 for OS/2 V2.1 and DDCS V2.3, and we look forward to your comments. Use *Personal Systems*' reply card between pages 56 and 57 or our Internet IDs, `psts@vnet.ibm.com` or `bhawkins@vnet.ibm.com`, for your comments.

Bill Wong is a DB2 technology specialist in Software Solutions, Toronto Laboratory, IBM Canada Ltd. He provides worldwide technical support for DB2 and DDCS on workstations and helps customers position DB2 technology within their organizations. Bill frequently participates in trade shows and speaks to DB2 user groups. His previous work included DB2 and CICS systems programming on MVS and serving as a DB2 DBA. Bill has taught courses on database management and strategic IT planning at the University of Toronto.

OS/2 Victories from the Data Management Front Lines

By Nancy Miller

All that data . . . what good is it if you can't manage it?

OS/2 plays a crucial role as a platform or control point for a number of data management tools designed to help manage data residing on all IBM platforms. In this article, we'll look at the data management challenge, glance at the strategy, and examine a number of OS/2-based tools that have become formidable weapons in the battle to manage data effectively.

Large and small companies are battling to control ever-increasing quantities of data–data that is crucial to their businesses' operations and profits in a time of increasing global competitiveness. For many companies, there is not only more data than they ever imagined, it is also distributed across enterprises and even spread around the world. The information systems (IS) vision for many companies is of thousands of widely distributed clients–many with palmtop or laptop computers–all needing access to multi-terabytes of data. In this world, effective data management is not just a matter of convenience, it is a matter of survival.

For someone with database administration responsibilities, this can be a frightening vision. All the database administrators (DBAs) I know tell war stories about spending late nights and weekends trying to solve tough management issues, about impossibly slow data access, about lost data or data they know is there somewhere if they can only find it! Fortunately, the weapons available for these data management wars are improving all the time. And, like a good general, IBM has a strategy for data management success.

The Data Management Challenge

Until recently data management has meant dealing with data in fairly structured forms such as databases or reports that can easily be stored or queried and kept in a central location. Today data pours into computers from every direction. With the growth of the information superhighway, the web of communications between computers has grown exponentially. As the use of these channels–e-mail, multimedia, mobile computers, groupware–has increased, the quantity of data has exploded, much of it in nontraditional formats. And, as the cost of computer power has dropped, many businesses have been able to spread data across heterogeneous environments.

Analyzing business data is becoming more sophisticated. Since many businesses depend on detailed demographic analyses to make product marketing decisions, businesses that may have once analyzed their sales data quarterly now conduct daily analyses to reduce unwanted inventories and control just-in-time production schedules. The result for the IS shop? A widening torrent of data that must be rapidly understood and readily available for the business to be successful.

What are the specific issues that a data management strategy must meet? Simply put, data must be *secure*, and it must be *available* where and when you need it.

The Battle Strategy

IBM has designed its data management strategy around four key principles:

- Provide unattended, remote operations via a client/server control point
- Manage and connect multi-vendor database environments
- Deliver functions that not only bring value to the small local area network

(LAN) environment but also scale to the large distributed enterprise

- Provide an integration platform for the suite of database tools

This strategy is implemented through DB2 for OS/2's DBA tools, advisor tools, utilities, and middleware.

DataHub for OS/2—A DBA Tool

DataHub for OS/2 gives you a single OS/2 control point–the DataHub/2 workstation–to manage complex client/server relational database environments. The managed databases–either local or remote–can include DB2 for OS/2, DB2 for AIX/6000, DB2 for MVS, DB2 for VM, and DB2 for OS/400.

Following are some of the administrative tasks that DataHub can perform:

- Display database objects that exist within the system and display relationships between those objects
- Copy objects such as tables (including their authorizations) between databases
- Display the status of relational units of work
- Create, drop, and alter database objects
- Manage database authorizations
- Invoke and schedule database utilities and commands such as `BACKUP` and `RECOVER` for remote or local databases
- Automate database management tasks, either through the DataHub command scheduler or by interoperating with FlowMark, IBM's workflow management product

With DataHub for OS/2 V2 planned for availability in third quarter 1995, data management capabilities will be increased for large distributed environments. These new capabilities include:

- Support for concurrent task operation
- Interoperability with other systems management tools

  DataHub for OS/2 V2 can generate simple network management protocol (SNMP) traps that can be processed by NetView for OS/2. Also, NetView for OS/2 alerts can be processed as DataHub for OS/2 alerts.

- Support for issuing OS/2 commands to remote OS/2 workstations

  With this capability, you can manage remote OS/2 directories and files, determine which processes are running, perform OS/2 tasks, even re-IPL the OS/2 workstation.

- Discovery of other databases on the LAN
- Expanded network protocol support

  DataHub for OS/2 V2 will support NetBIOS and transmission control protocol/internet protocol (TCP/IP) for DataHub tool conversations as well as advanced program-to-program communication (APPC) and advanced peer-to-peer networking (APPN).

DataHub is an integration platform for other management tools. Many of the tools featured in this article can be integrated into DataHub, giving you a single graphical user interface (GUI) to perform many administrative tasks.

Advisor Tools

Advisor tools give you functions for managing performance, predicting capacity, monitoring events, managing alerts, and integrating network management. IBM has devised a number of advisor tools as part of its strategy for managing data from the OS/2 platform.

New DB2 for OS/2 V2.1 Tools

DB2 for OS/2, IBM's relational database product for the OS/2 platform, has always provided users with several system management tools–a recovery tool for backup and restore, a directory tool for managing local and remote database locations, and a configuration tool for managing all the parameters available for tuning the databases as well as the database manager component.

Now, DB2 for OS/2 V2.1 features robust new tools for data management: database systems monitor, snapshot manager, performance monitor, the database director, a new version of Explain, and a SNMP agent. Performance Monitor, Visual Explain, and Database Director are DataHub GUI tools that can provide an integrated management platform. You can find details on these tools in this issue's "DB2 for OS/2 Administrative Tools" article.

DB2 Estimator for Windows

The DB2 Estimator tool estimates the performance of DB2 for MVS applications, transactions, database utilities, and SQL statement performance. During the application design phase, it can help evaluate alternative designs. During production, it can determine the impact of either hardware or workload changes. It can be run from either OS/2 or Windows and can be optionally invoked from DataHub.

Middleware

In the introduction, I talked about the often bewildering complexity of the distributed data environment. Middleware technology is designed to help you deal with that complexity by providing a consistent interface and letting you access data transparently, regardless of location.

IBM's data replication products address the requirement to maintain multiple copies of data at different sites in a distributed environment. Data replication may be established to provide local access to data and to allow systems to function autonomously at times when there may be failure points in the network. Or a user might require data to be replicated to a local workstation to analyze market trends.

For example, a business might need to analyze inventory information each day to make ordering decisions. Point-of-sale information can be replicated to the host at night where the data can be consolidated, then replicated to a workstation for decision support analysis the next morning. The decision support process is thus off-loaded from the host and can be performed locally by the decision maker.

The following products use OS/2 as a control point but may support data replication across a range of operating environments.

DataPropagator Relational

DataPropagator Relational (DPropR) supports replication between the IBM DB2 family of databases: DB2 for MVS, DB2 for OS/2, DB2 for OS/400, and DB2 for AIX/6000. Data may be propagated either as a full refresh or as a "changes-only" update from one database to another. Additionally, Sybase, Oracle, and other databases can be targets of full refresh propagation through interoperation with DataJoiner.

DPropR is implemented through two separate components: Capture collects the

source data and Apply copies the data at the target. Communication between the platforms uses Distributed Relational Database Architecture (DRDA). The DPropR Control Point uses the facilities of a DataHub OS/2 machine with its graphical user interface.

DataRefresher and DataPropagator NonRelational

DataRefresher and DataPropagator NonRelational tools leverage legacy host data sources.

DataRefresher can run from a DataHub/2 control point to provide full refresh copying from any MVS source (IMS, VSAM, DB2, flat files, and others) to targets in MVS, VM, VSE, AS/400, AIX/6000, and OS/2. Data can be gathered from up to 16 different sources using heterogeneous joins and can then be directed to up to 99 different targets in a single pass of the source data. In addition, DataRefresher can be used to bring data into the DPropR staging area for further propagation.

DataPropagator NonRelational allows IMS data to be replicated across the IBM DB2 family of databases. It uses synchronous replication to copy from IMS to DB2 and from DB2 to IMS. The Data Refresher graphical user interface may be used to set up the mappings between hierarchical segments and the relational tables. Once data has been brought into DB2, it can be accessed by any of the DB2 family relational database systems using DRDA.

Utilities

Utilities are tools designed to address routine operating requirements. Most often, utilities are provided with database engines. They may be invoked remotely and scheduled to execute automatically through integration with DataHub.

DB2 for OS/2 Engine Utilities

Engine utilities, which include `REORG`, `RUNSTATS`, `BACKUP`, and `RECOVER`, are already a part of the DB2 for OS/2 product and provide important systems management capabilities. DB2 for OS/2 V2.1 has `LOAD` and `UNLOAD` utilities. For information on these new utilities, see this issue's "Performance Enhancements in DB2 for OS/2 V2.1" article.

ADSTAR Distributed Storage Manager

Beginning with 2.1, DB2 for OS/2 databases can be backed up using ADSTAR Distributed Storage Manager (ADSM). This product will back up databases remotely to MVS, VM, AIX/6000, and OS/2 systems.

Decision Support Tools

Decision support tools help you easily access the data you require. IBM's decision support tools support your data requirements at every skill level and bring data together from disparate data sources.

DataGuide for OS/2

You know that data is out there somewhere . . . but where is it? DataGuide addresses the problem of keeping track of data in a complex distributed environment. It provides client/server information catalogs on a LAN so that users can share and reuse business information.

DataGuide features include:

- A graphical user interface
- The capability to directly launch OS/2, Windows, DOS, and client/server applications such as Visualizer
- The ability to run multiple programs concurrently

DataGuide provides a workgroup solution for information sharing of objects including spreadsheets, charts, reports, tables, queries, and multimedia data.

Visualizer Family

IBM Visualizer is IBM's tool set for querying, analyzing, and presenting data from IBM databases and has been featured in a series of articles appearing in *Personal Systems* magazine. (See the January/February, March/April, and May/June 1995 issues of *Personal Systems*.) For details on developing Visualizer applications, see the "Visualizer Development" article in this issue.

Visual Warehouse

A data warehouse is an informational data store specifically designed for analyzing business data. A well-designed data warehouse lets you build a complete view of the important aspects of your business. IBM's Information Warehouse framework is a set of database management systems, interfaces, tools, and facilities for managing and delivering business information. Many of the products discussed in this article participate in the Information Warehouse.

Visual Warehouse is the newest member of the IBM Information Warehouse family of products. It is designed to pull together data from corporate data sources or a centralized warehouse into the workgroup environment. The Visual Warehouse Solution includes Visual Warehouse Manager, DataGuide/2, Visualizer Query and Charts, DB/2 for OS/2, DDCS (Distributed Data Connection Service) for OS/2, a Visual Warehouse Non-Relational Adapter, and consulting services. The Non-Relational Adapter allows Visual Warehouse to extract data from VM, IMS, VSAM, and MVS flat files.

Visual Warehouse aggregates data from multiple sources and combines it to form a *business view*, a single view of data that decision makers need. The Visual Warehouse engine automatically retrieves data and creates the data warehouse. Administrative functions such as logs, statistics, and security are built in.

Summary

Data is dynamic, and what meets your needs today may not tomorrow. The solutions listed in this article are scalable. If a business finds it has outgrown one platform, it can easily move its data to another and still manage it from OS/2.

In short, data management doesn't have to be a battle. Many solutions are available for the OS/2 platform that can turn a data management battle into a peaceful information domain so that data can truly be an asset.

Nancy Miller is a marketing support representative in the OS/2 Systems Support and Services Group in IBM's Personal Systems Competency Center in Roanoke, Texas. Nancy has provided support for DB2 for OS/2 and its predecessor product, Database Manager, for six years and has published several articles about these two products. She is a graduate of Oklahoma Baptist University and studied Computer Science at Texas Woman's University. Nancy's Internet ID is `nmiller@vnet.ibm.com`.

Voting Kiosks: The Future of Electronic Elections

By Dan Britton and Meri Ortali

This article describes an award-winning joint venture between IBM and the University of Alabama to design a convenient, functional, and highly secure voting process located in kiosks around the University's campuses.

On December 9, 1994, IBM and the University of Alabama received the DB/EXPO 1994 RealWare Award for Desktop Database and Workgroup Computing in the Office. This prestigious award recognized the University's pioneering work in using on-campus kiosks for its advanced electronic voting booth and IBM's DB2 for OS/2 as the backend server for this application.

RealWare Award winners are selected for giving users outstanding solutions in real-world applications. The entries are judged on the innovative use of technology, the delivery of measurable results to the user, and the efficient use of resources. This joint IBM/University of Alabama application succeeded in all three areas.

The Concept

The development of the student election application began a year and a half ago by the University of Alabama's leading-edge research group, Advanced Technology Group (ATG), led by Dr. Rod Riley. The ATG uses DB2 for OS/2 as the backend database for all its multimedia client/server projects. DB2 for OS/2 was chosen for the kiosk application because of its performance characteristics and its support for referential integrity, which is used heavily within the award-winning application.

The kiosk application addresses the University's need for fast, accurate, and secure elections. In addition, because the new process is highly automated, fewer people are needed to administer the election process. This application demonstrates that electronic voting, for all types of elections, is not too far in the future.

"The Election Board of the University approached our team about providing a better way to conduct student elections," Dr. Riley explained. "The electronic kiosk concept was work we had already begun to experiment with, so making the election process part of it was a simple task. We started this work to satisfy the information needs of our students, students who have grown up watching MTV and listening to 30-second sound bites. Interactive computing is the future. Students expect to interact with the world through computers, and that's what we give them."

The Process

Prior to an election, students can walk up to kiosks located at several places throughout the campus. By using a touch screen, they can select, for example, a student government office. Pictures of the candidates running for that office are displayed. After touching a candidate's picture, the students can see and hear a full motion 20-second video clip of the candidate explaining his or her platform.

Also before an election, the Records office provides a list of eligible voters. This is downloaded from a VSAM structure into DB2 for OS/2. They will consider DataRefresher in the future to automate routine data replication.

On election day, the same kiosks become electronic voting booths. Students walk up and enter their student ID. The security part of the voting program validates the voter as a student at the University. Pictures of the candidates running for office are displayed. Students vote for the candidates of their choice. Voter response can be determined at any point during the election cycle. At the end of the election period, a simple query to the database will tell who has won.

For security reasons, many of the database details cannot be revealed. However, we can tell you that the DB2 for OS/2 database server contains three primary tables that are used in the election process:

- An election table listing the student government offices and candidates
- A table listing eligible student voters
- A table listing the IDs of students who have attempted to vote more than once

As students are verified as eligible to vote, they select their choices from the touch screen. After verifying their selections, they press OK to process their votes. The election tables are updated and the fact they have voted is recorded. With the security mechanisms built into the application and the database, fraud has successfully been avoided.

Business rules are established that govern the application's required characteristics. Only eligible voters can vote. For example, the election for student government allows all registered students to vote. However, an election for officers in a graduate program will allow only students in that graduate program to vote. Students can vote from any kiosk, but they can vote only once. If a student attempts to vote more than once, the information is written to a separate table in the database, which can then be used to generate a report that is given to the Election Board. For demographic purposes, the kiosk location, date, and time are recorded for each vote. The actual vote by the student remains anonymous.

In addition to the electronic elections, the kiosks support student queries regarding campus telephone numbers and campus

events including sports activities, class schedules, and course offerings.

The ATG is currently working on its latest version of the electronic election system. Version 3 of the application will use a new multimedia application development tool, IBM's VisualAge object-oriented development environment, because it has the multimedia and database parts needed by the application. Enhancements to the next version of this application include the use of binary large objects (BLOBs) in Version 2 of DB2 for OS/2 or Version 2 of DB2 for AIX for video segment storage. With the ability to turn off logging for BLOBs, performance can be maintained in this real-time environment.

The Components

The current election application consists of currently available components. Each kiosk contains an IBM PS/2 Ultimedia Model 77 equipped with a touch screen display.

According to Bryan Spencer of the ATG, "We chose the Ultimedia Model 77 because it is one of the most reliable machines on the market today. These machines run 24 hours a day, seven days a week without stopping. In over 18 months of use, none of the machines experienced a single hardware failure."

Each machine is equipped with an ActionMedia/2 playback adapter to display the digitized video segment that students select from the touch screen. OS/2's operating system provides the function needed to support multimedia applications. Recently, the ATG moved from Version 2.11 to OS/2 Warp. This migration has provided noticeably better performance on both the clients and the server.

NetBIOS on a local area network (LAN) connects each computerized kiosk to the DB2 for OS/2 server, also an Ultimedia Model 77. This connection provides excellent performance during the elections.

Future enhancements to the kiosks include migration to DB2 for OS/2 and DB2 for AIX Version 2 plus support for transmission control protocol/internet protocol (TCP/IP) instead of NetBIOS.

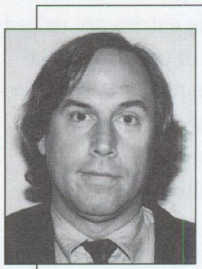
The Result

"We love it!" Dr. Riley exclaimed about the application overall and the VisualAge environment in particular. "I don't know how we ever did application development without [VisualAge] and it certainly is a powerful way to build new applications quickly. The combination of VisualAge and the DB2 family delivers a one-two punch that is untouchable, to say nothing of the support from the Cary lab. Actually, it's not so much support as it is an extension of our own team. They are always there for us!"

Ms. Kelley Swinney, advanced technology specialist at ATG, said, "IBM has been exceptional. They are very willing to help us in whatever we need to do. They respond to us immediately. In the time frames in which we worked during the election, we had critical deadlines we had to meet. IBM has been there to support us all along."

Dr. Riley remarked that "IBM has gotten its share of knocks in the past few years, but it is still a technology leader in software and hardware. We continue to rely on IBM to deliver products that allow the Advanced Technology Group of the University of Alabama to fulfill our visions. Time and time again, IBM has delivered those products. Our winning of the RealWare award would not have been possible without our partnership with IBM."

Although Dr. Riley was impressed with the other RealWare award finalists, he believes the University of Alabama and IBM won because "[The ATG] project was associated with something that you could explain to average people and they would understand. We're taking technology to the real world and solving real problems."



Dan Britton has worked with database products since 1987 when he joined the OS/2 Database Manager development group in Austin, Texas. He has recently finished a two-year assignment in the Database Technologies group at the IBM Toronto lab and will be joining IBM's Santa Teresa lab to continue his work supporting the DB2 family of products.

Meri Ortali has spent the last seven years working with IBM relational database products. She has held a variety of positions in development, product and strategic planning, and technical support. Meri has experience in helping customers implement client/server solutions. Recently she has been involved in developing and providing technical education for DB2 Common Server Version 2 and DB2 Parallel Edition.

Performance Enhancements in DB2 for OS/2 V2.1

DB2 for OS/2 V2.1 has several enhancements allowing it to support larger databases and more complex applications. In addition to these functional enhancements, DB2 for OS/2 has undergone a number of changes to ensure high performance of both the new and existing functionality. This article describes the performance enhancements in DB2 for OS/2 V2.1.

More and more these days, business applications must process higher volumes of increasingly complex data. To meet the growing needs of today's more complex environments, IBM has enhanced its popular 32-bit relational database management system, DATABASE 2 (DB2) for OS/2. So that you can support larger databases with better performance than ever before, IBM developers significantly improved the Structured Query Language (SQL) compiler, the database engine internals, data storage, and data accessing components in DB2 for OS/2.

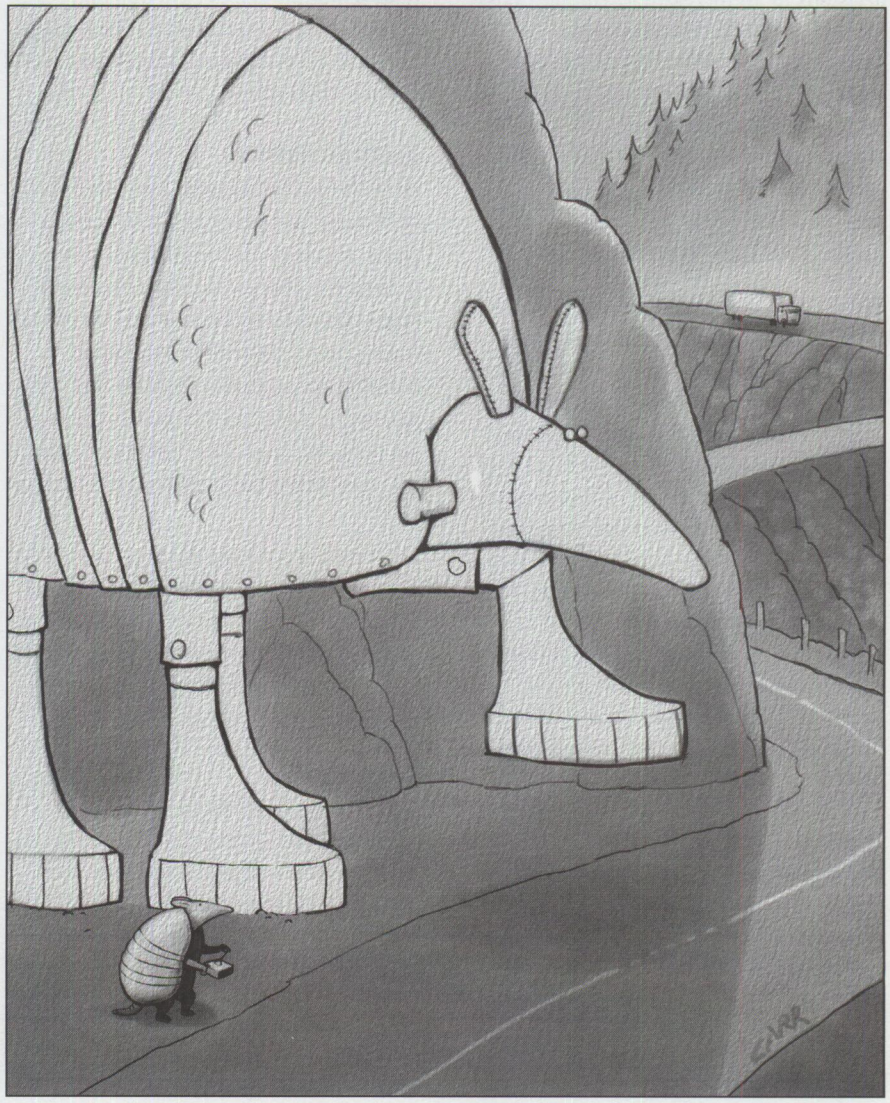
Mike Logan
IBM Corporation
Roanoke, Texas

Businesses are using more tools or applications that automatically generate queries. To remain flexible, these query generators seldom focus on optimizing queries and can generate very complex queries for what may seem, to the user, a simple request. The SQL compiler in version 2.1 has been specifically enhanced to improve this environment.

SQL Compiler Enhancements

Perhaps the most important part of the database engine is the SQL compiler. The compiler processes each SQL request to determine what must be done to satisfy the request and the fastest way to complete the work.

The SQL compiler in DB2 for OS/2 V2.1 has been greatly enhanced to make better



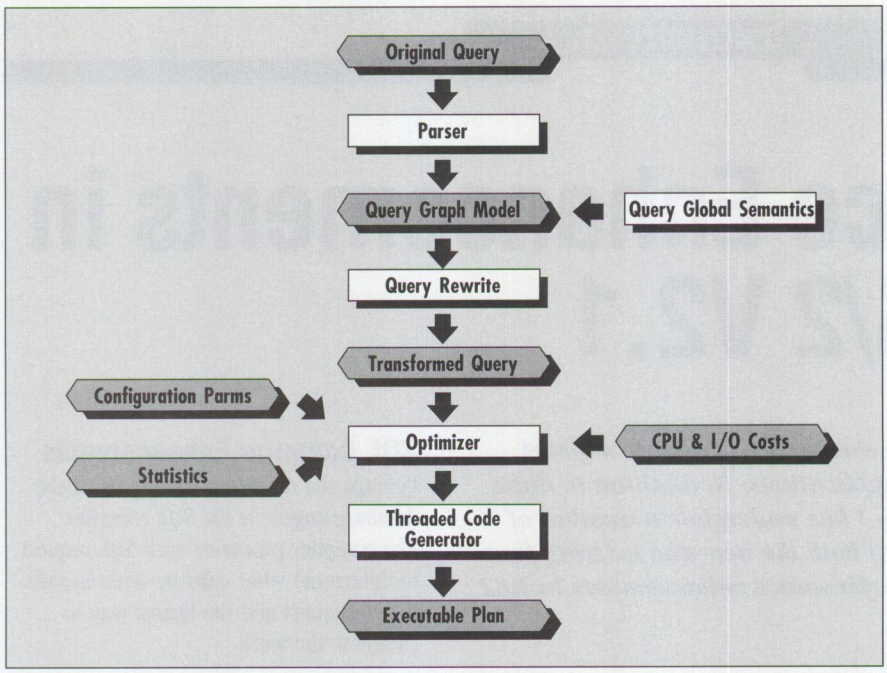


Figure 1. SQL Query Compilation Process

decisions in an increasingly complex environment. These enhancements are the foundation for many of the other enhancements in this release, such as the extended SQL standards support and the object-oriented extensions. The compiler's new design also helps to make it more scalable and extendable to keep up with today's users' rapidly growing demands.

What Does the Compiler Do?

To better understand the enhancements to the SQL compiler, you must first understand exactly what the compiler does. The compiler goes through five main phases when converting an SQL query to an executable statement: *parsing, query semantics, query rewrite, query optimization,* and *threaded code generation.* The stages themselves are not new—the version 1.x compiler also performed them. But in version 2.1, each stage has been enhanced or improved to be more efficient and thorough. Figure 1 shows an overview of the stages in the SQL query compilation process.

Parsing

The first phase in query compilation is the *parsing* phase. During this phase, the compiler examines the statement for syntax errors. If the statement is valid, the compiler converts it into an internal representation called the Query Graph Model (QGM) by parsing and applying its semantic routines. The QGM is a flexible representation that is easy to modify and enhance, thus simplifying the more complex task of optimizing the statement to execute efficiently.

Query Semantics

Once the query has been converted into QGM form, the *query global semantics* phase augments the original query to include the query's complete execution scope. This means that the actions of executing check constraints, referential integrity constraints, recursion, and triggers are included at this point. The compiler also replaces view references with the view definition's QGM representation.

Having these changes made to the QGM prior to any optimization is a significant enhancement. This allows the entire execution scope of the query to be optimized. These extensions were not optimized by the version 1.x compiler.

Query Rewrite

Next, the compiler begins its *query rewrite* phase. During this phase, the sophisticated compiler transforms the query into a form easily handled by the optimizer. The rewrite phase is rule-driven, easily allowing future enhancements. While the DB2 SQL compiler has always done some query rewrite, the new compiler offers an extensive set of transformation options.

In general, the compiler attempts to transform queries from one-row-at-a-time execution to a more set-oriented execution. These transformations help the optimizer choose the most efficient path for even the most complex queries.

The new query rewrite transformations include:

- Addition of implied predicates
- General predicate pushdown
- Existential subquery to join transformation
- Magic set transformations
- Redundant join elimination
- Quantified predicates to scalar subqueries conversion
- INTERSECT to EXISTS subquery conversion
- OR to IN conversion
- IN to join conversion
- View merging
- DISTINCT elimination

Query Optimization

Once the query has been converted to its simplest form, the compiler enters its *query optimization* phase. In this phase, the cost-based optimizer determines the most efficient way to execute the query. The new compiler has a much more extensive set of basic access paths to choose from, and it investigates more options to determine the best join order.

Because the compiler's ability to choose the most efficient path depends on how well it models each possible path's associated costs, significant work has been done to improve the compiler's ability to more accurately measure each path's costs. Two of the most important costs associated with any plan are central processing unit (CPU) and input/output (I/O) costs. In the past, these values were hard-coded constants based on average values for common devices. In DB2 for OS/2 V2.1, the CPU speed is calculated based on the average elapsed time in milliseconds required to execute a CPU instruction. The I/O costs are based on the overhead associated with the disk controller, including seek and latency times plus the time in milliseconds needed to read a 4 KB page into the buffer. The values for these times

depend on the storage management method chosen for the table space; a default is still provided. The optimizer also uses different weights to model the costs of random versus sequential I/O. In some situations, this can greatly affect the selected path.

Another important enhancement is how DB2 for OS/2's optimizer uses the statistics gathered by the RUNSTATS utility. After the RUNSTATS utility gathers statistics about the physical characteristics of the data and indexes, the optimizer uses this information to calculate the number of I/Os that each possible path requires.

The RUNSTATS utility in version 2.1 has been enhanced to optionally gather distribution statistics for the column values within a table. Distribution statistics help DB2 for OS/2 to more accurately estimate the number of rows that will satisfy a query and thus choose a more accurate access method. In version 1.x, the compiler assumed that all data values in a column were uniformly distributed, that is, each value occurs the same number of times. This is usually true for only a few columns in the table.

Version 2.1 also details more statistics to help estimate the number of page accesses you need to scan an index. To help you to make better decisions, you can then use these statistics to model your environment with different buffer sizes, more data, or better clustering.

Threaded Code Generation

Once the optimizer has selected the most efficient path, it passes this information to the threaded code generator. Here, in the compiler's *threaded code generation* phase, the access path is converted to the actual executable plan. The executable code improves performance with new execution techniques.

What All This Really Means

By now you may be thinking that this sounds like a lot more overhead, and you might be right in some cases. The new optimization techniques can potentially take far more time and resources than the old methods. However, for very complex queries, your savings in execution time far outweigh the additional compile time. The penalty will be felt more by applications that issue dynamic SQL, because those statements are compiled at runtime. Static SQL statements will pay the penalty only at bind time, not at runtime.

Optimization Level	Typical Usage
Minimal (0)	Only when low overhead is priority, very simple queries on well-indexed tables
DB2 V1 (1)	General purpose use in a CPU or memory constrained system
DB2/MVS (3)	Suitable for broad range of applications, has fewer options than default
DB2 V2 (5) (default)	Mixed environment of both simple and complex transactions
Maximum (9)	Only for very complex or long-running queries

Figure 2. Query Optimization Levels

Fortunately, if you don't need all of this fancy stuff, you don't have to use it. DB2 for OS/2 allows you to specify the level of optimization needed for your application. There are five levels that specify the type of joins that will be considered, the query rewrite rules that may be applied, and the number of plan options that will be considered.

Figure 2 summarizes the five levels of query optimization. The first two levels (Minimal and DB2 V1) use only limited query rewrites, do not use distribution statistics, and do not use list prefetching. The last three (DB2/MVS, DB2 V2, and Maximum) offer increasing levels of query rewrites, a wide range of access methods, and may use both distribution statistics and list prefetching.

The default level uses all the query rewrite and access methods needed in most cases. Typically, you will use another level only to reduce compile time or resource usage or for very complex queries.

To find which optimization level best meets your needs, you will have to weigh the benefits versus the overhead for each of your applications.

I/O Enhancements

I/O is, by far, the most time-consuming task that a database must perform. As databases become increasingly larger, I/O becomes more of an issue.

DB2 for OS/2 V2.1, using several enhancements to help reduce the time spent waiting for I/O, improves overall system performance. The improvements involve making I/O more asynchronous and parallel at both the hardware and software levels.

Database Partitioning

To support larger databases, give the user more flexibility to place data, plus improve performance and administration, DB2 for OS/2 V2.1 uses a new storage model based on the table space. A *table space* consists of one or more storage areas called *containers*. In OS/2, a container can be either a directory or a file. The containers that make up a table space may reside on multiple devices, thus spanning media without needing disk arrays.

DB2 provides two classes of table spaces: *System-Managed Space* (SMS) and *Database-Managed Space* (DMS). In an SMS table space, the file system manages and allocates space in the storage area. An SMS table space consists of one or more directories under which files are created and expanded to hold database data. Each file contains a single database object and is extended one page at a time as the object grows. DB2 for OS/2 V1.x was basically a single SMS table space with a single directory.

A DMS table space consists of one or more pre-allocated files into which the database manager places data. You can

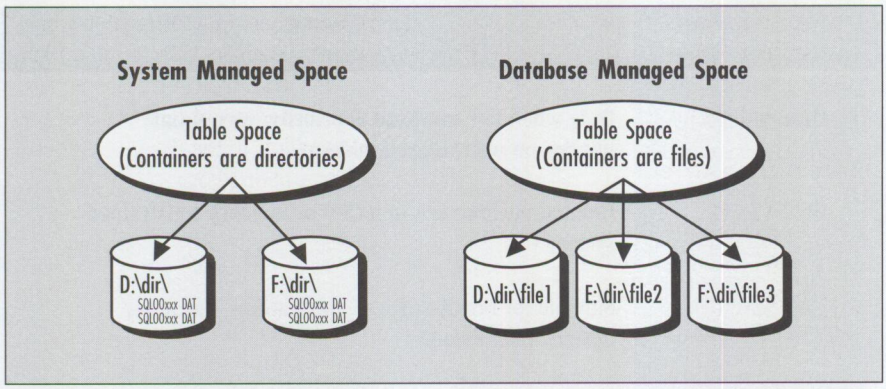


Figure 3. Table Space Storage Types

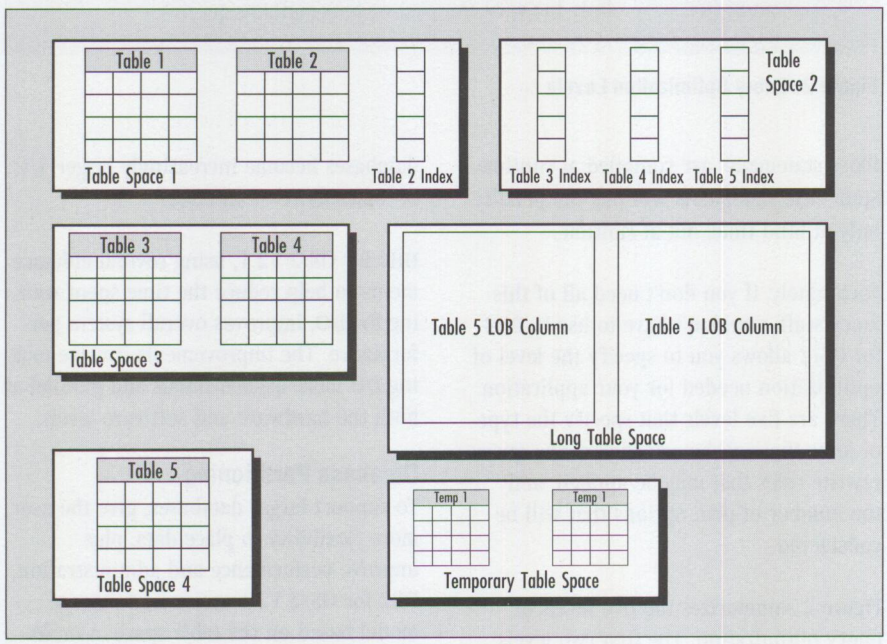


Figure 4. Data Placement Within Table Spaces

store multiple database objects in a single file, and the file is not extended as the objects grow. You can add more files to the table space to increase its storage capacity, and the data is then automatically rebalanced across all containers. When you create a table in a DMS table space, the table's rows are distributed across the containers in the table space. In addition, you can place tables, their long or large object (LOB) columns, and indexes in separate table spaces. Figure 3 illustrates the table space storage types.

When you define a table space, you must specify the type of data that may be placed in the table space. There are three options: regular, long, and temporary. A *regular* table space can store any type of data except temporary tables. Regular is the default. A *long* table space stores long or LOB table columns and has been optimized specifically for this type of data. A *temporary* table space stores only temporary tables such as those used to perform sorts or joins. A temporary table space also has internal optimizations specifically for short-lived temporary files. Figure 4 shows data placement within table spaces.

This new storage model offers several performance benefits. The most obvious is its media-spanning capabilities, which support larger databases, help reduce I/O contention and wait periods due to disk head movement, and give administrators more control over the data's physical placement on the system. For example, you can now place frequently used tables or indexes on faster drives to help reduce I/O time. LOBs and less frequently used data can be placed on slower devices where the performance impact is not as noticeable.

Another major performance benefit comes during backup and recovery. Each table space can be backed up or restored independently. During this process, all table spaces can remain online. Version 2.1 also supports backing up to and restoring from multiple concurrent devices. This allows multiple device read/write on both the source and destination during a backup/recovery process, greatly reducing the elapsed time necessary to complete the task.

Prefetching Data Pages

DB2 for OS/2 enhances I/O performance when prefetching data pages into the buffer pool. *Prefetching* pages means retrieving pages of data from the disk in anticipation of their use. This helps promote CPU and I/O parallelism and reduce idle time.

DB2 implements two types of prefetching: sequential prefetch and list prefetch. *Sequential* prefetch is used to access data in a near linear fashion. This technique may be used for a table scan or a highly clustered index scan. While executing the query, DB2 can determine whether this technique can be used for a particular query. This process is known as *sequential detection.* Because there is some overhead associated with sequential detection, it may be optionally turned off.

List prefetching is a technique that DB2 can use to retrieve data by first creating a sorted row identifier (RID) list. By first sorting the RIDs of the rows that satisfy a query and then retrieving the data, a single page access accesses more rows, thus reducing disk thrashing. This technique is most likely to be used by multiple-index access paths. The SQL compiler determines at compile time whether or not to use list prefetch.

Big-Block Reads

To further speed up prefetching operations, DB2 also implements big-block reads. A *big-block read* is a single I/O operation that reads in multiple 4 KB pages. In the past, all disk I/O was done one page at a time. When reading in consecutive pages, as is done during prefetch, it makes sense to read larger chunks at a time. It takes roughly the same amount of time to read two pages separately as it

would 32 consecutive pages simultaneously. This can significantly reduce I/O time.

Asynchronous Page Cleaners

In DB2 for OS/2 V1.x, when an agent process needed to read a page of data from disk into memory, it first had to find a slot in the buffer pool. If this slot contained a page that had been changed since it was read in, the agent had to wait while the page was written out to disk before the new page could be read in.

Version 2.1 alleviates this problem by introducing asynchronous page cleaners. An *asynchronous page cleaner* is a background thread that, when triggered by the engine, traverses the buffer pool, writing out modified pages to disk. This process is asynchronous to the engine's regular workload. As a result, the agents are less likely to get a slot that has not yet been written to disk, thus eliminating wait time and improving the overall response of the application. As a side benefit, asynchronous page cleaning reduces the time necessary to recover from a soft failure, such as a power outage, since more of the transactions will already be written to disk. You can configure the number of page cleaners and the frequency at which they run.

High-Speed Load Utilities

To further reduce the time necessary to load large amounts of data into DB2, version 2.1 introduces a high-speed load utility. The load utility can bulk load data into a database or append large amounts of data to an existing table. The load utility can build indexes and gather statistics while loading all supported data types, including LOBs. On average, the load should function approximately 10 times faster than the current import utility.

The load utility takes advantage of parallel I/O in two ways. First, it supports multiple input/output files, which can be located on separate devices to reduce overhead. Second, by enabling the table space in which the destination table resides to span multiple devices, you can achieve parallel I/O at both the source and destination of the load.

Client/Server Enhancements

Several enhancements in version 2.1 directly affect the performance of remote clients. These improvements show up mainly as reductions in elapsed time to establish a connection. Distributed Database Connection Services/2 (DDCS/2) environments are most likely to benefit from these enhancements.

Directory Caching

In version 1.x, any time a connection was established, DB2 had to read the directory information (system, node directory, local, and/or Database Connection Services [DCS] directories) from disk in order to locate the database. Version 2.1 gives the option of having the directory information loaded into memory at DB2 startup time. This eliminates the I/O and reduces elapsed time for the connect.

Connection Reuse

Another enhancement that helps client connect time is connection reuse. In most situations where a connection had been previously established with a server, the communication link between the two will still be in place and can be reused. This eliminates the communication startup time as well as the time it takes to start an agent on the server. Agents on the server do not end when a connection ends; they remain on the server until they are needed for another transaction.

DB2 for OS/2 V2.1 implements a multithreaded database engine.

Compound SQL

In much the same way that record blocking improves performance in a client/server environment, so will compound SQL.

Compound SQL allows multiple SQL statements to be grouped into a single executable block. They are sent to the server as a group instead of individually. This helps eliminate some of the communication overhead required to execute the statements. An example of an application that will benefit from compound SQL is one that does bulk inserts. By grouping them, you get one send/receive transmission instead of multiple ones.

There are two types of compound SQL: `ATOMIC` and `NOT ATOMIC`. The difference between the two types is how success or failure is handled. In `ATOMIC`, if any one statement in the block fails, all fail. With `NOT ATOMIC`, each statement fails or succeeds independently of the results of the other statements in the block. DB2 for OS/2 V2.1 supports both `ATOMIC` and `NOT ATOMIC` compound SQL, but only `NOT ATOMIC` may be passed through DDCS/2.

SQLCODE Mapping

An enhancement that benefits some DDCS/2 users is the ability to turn off `SQLCODE` mapping. DDCS/2 uses a map file for matching `SQLCODE`s from one platform to another. This results in extra overhead at connect time, because the map file must be read in from disk. DDCS/2 now allows this process to be turned off. If the connection is to one of IBM's Distributed Relational Database Architecture (DRDA) servers, a default will be used from a single copy in memory. If the host is a non-IBM platform, no mapping is done and codes are returned as-is.

Other Performance Enhancements

DB2 for OS/2 V2.1 also has some general performance enhancements. While none of these has a significant internal or external change, each one contributes to the overall performance improvement available in this release.

Multithreaded Database Engine

In DB2 for OS/2 V1.x, the database engine and all application agents ran as separate processes under OS/2. DB2 for OS/2 V2.1 implements a multithreaded database engine. This means that all database activities run within a single process and address space. Each application agent runs as a dedicated thread. Most other asynchronous activities run as separate threads. These include:

- Deadlock detectors
- I/O servers (data prefetchers)
- Asynchronous page cleaners
- Event monitors
- Communication and interprocess communication listeners

This helps reduce the memory requirements, allows for more efficient switching between tasks, and reduces other overhead on the system. However, it does not imply a change in the way that client applications work. There is still a single

commit-scope for all threads in a process, and all database access within a process on a client will be serialized.

Blocking for Local Fetches

One of the most important performance options for DB2 client/server requests has been its ability to do record blocking. *Record blocking* refers to the process of returning multiple records that satisfy a query in one transmission to the requesting client.

DB2 for OS/2 V2.1 now allows record blocking for local applications as well. DB2 can return a block of rows when the initial `OPEN CURSOR` is executed. Subsequent `FETCH` operations retrieve data from this block until all rows in the block have been processed. Only after all the rows in a block have been processed will DB2 cross the firewall into the engine for the next block. This primarily benefits queries that return a large number of rows, like a table scan with thousands of rows. The size of the block used for local blocking is based on the Application Support Layer Heap Size.

Updateable Statistics

Despite the SQL compiler's sophisticated query optimization, it is sometimes desirable to force it to choose a specific path. Previous versions of DB2 did not provide many ways to influence the optimizer. DB2 for OS/2 V2.1 lets you update selected system catalog statistics. Not only does this allow you to greatly influence the decision of the optimizer, it also allows you to model production performance in a test environment and perform what-if analyses. You can also modify statistics on table spaces, tables, columns, indexes, and functions.

Pre-Sorting Keys for Create Index

Creating an index on a large table can often be time-consuming. DB2 for OS/2 V2.1 includes a new time-saving option for creating indexes. DB2 will now optionally sort index keys prior to inserting them into the index. Indexes with low cluster ratios or cluster factors will benefit the most from this option. Queries can also perform better using an index created with a sort. Note that creating an index with an initial sort requires about twice the disk space of not performing the sort first.

General Overhead Reductions

In addition to all the other performance enhancements in version 2.1, an effort was made to reduce overhead whenever possible. Examples are:

- Reducing the number of key comparisons necessary during an index scan
- Reducing sort overhead
- Minimizing the accesses to each page of temporary tables
- Efficiently allocating and freeing temporary tables

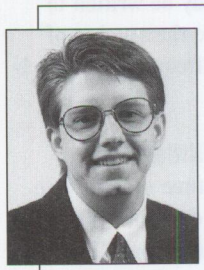
Although the results of these improvements may not be directly noticeable to the end user, they will improve performance overall.

Many Major Performance Benefits

To summarize, here are DB2 for OS/2 V2.1's major performance benefits grouped by usage:

- Queries
  - Enhanced SQL compiler
  - Blocking for local fetches
  - Index manager, sorting, and other overhead reductions
  - I/O prefetching and big block reads
  - Distribution statistics
- Online Transaction Processing (OLTP)
  - Blocking for local fetches
  - Index manager, sorting, and other overhead reductions
- Utilities
  - High-speed load
  - Backup/restore enhancements
  - I/O prefetching (export and create index)
  - Pre-sorting keys in create index
  - Database-managed storage
- General
  - DDCS/2 performance
  - CONNECT performance
  - Multithreading
  - Updateable statistics

DB2 for OS/2 V2.1's performance enhancements benefit every user of a relational database!



Mike Logan is a marketing support representative within the IBM Personal Systems Competency Center, Roanoke, Texas, where he has provided technical support and consulting on DB2 for OS/2 and on DDCS since joining IBM in 1993. Mike has a BS degree in Computer Science from the University of North Texas. He can be reached via Internet at `mlogan@vnet.ibm.com`.

DB2 for OS/2 Administrative Tools

With all the new functionality in DB2 for OS/2 V2.1, understanding what happens inside DB2 and the performance costs of each transaction becomes more critical. Easily managing and administering the database are also critical needs in today's environment. As explained in this article, DB2 will provide a set of new and enhanced tools to meet those needs.

Mike Logan
IBM Corporation
Roanoke, Texas

Monitoring the activities of the database engine can provide a wealth of information to optimize the DB2 configuration parameters, to tune DB2 performance, application performance, and Structured Query Language (SQL) performance, and to understand activity occurring in the database.

Monitoring DB2

DB2 provides two monitoring tools: a system monitor (which includes a snapshot monitor and an event monitor) and a performance monitor. All monitoring can be done either locally at the server or from a client workstation.

Database System Monitor

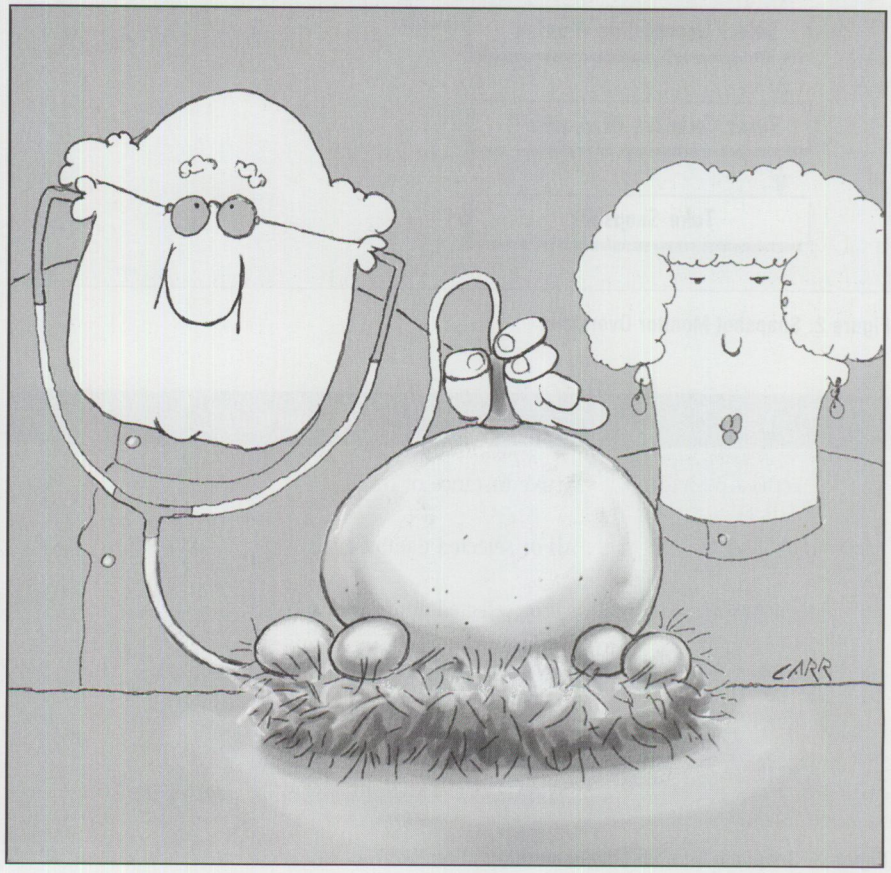
The DB2 engine now lets you monitor its activity through the database system monitor. Access to the data provided by the database system monitor is available through commands and application programming interfaces (APIs). The data may be collected as either a snapshot (point-in-time) or by event (over a period of time). The types of information that may be gathered include those listed in Figure 1.

The kinds of information reported by the system monitor are counters, gauges (current value), high-water marks, information (names, etc.), and timestamps.

Snapshot Monitor

The snapshot monitor provides information for a specific point in time. It provides cumulative counters and current information. You can reset many of these counters to allow you to get information over a specific period of time. The snapshot monitor collects information based on the level and groups requested (Figure 2). The available levels of information are shown in Figure 3.

The monitor always collects some basic snapshot information. More detailed information, as shown in Figure 4, can be optionally collected.



Configuration switches enable the collection of these optional groups. Defaults are set in the Database Manager configuration, but can be overridden by any application that requests monitor data.

General Information	SQL Statement Details	Database Activity
Database Connections	Sort Work	Agents and Applications
Locks and Deadlocks	SQL Cursors	CPU Usage
Buffer Pool Activity	Communication Activity	Logging
I/O Activity	Table Activity	Caching
SQL Statement Activity	Unit of Work Status	Programming Elements
		Overflow Elements

Figure 1. Data Monitoring Information

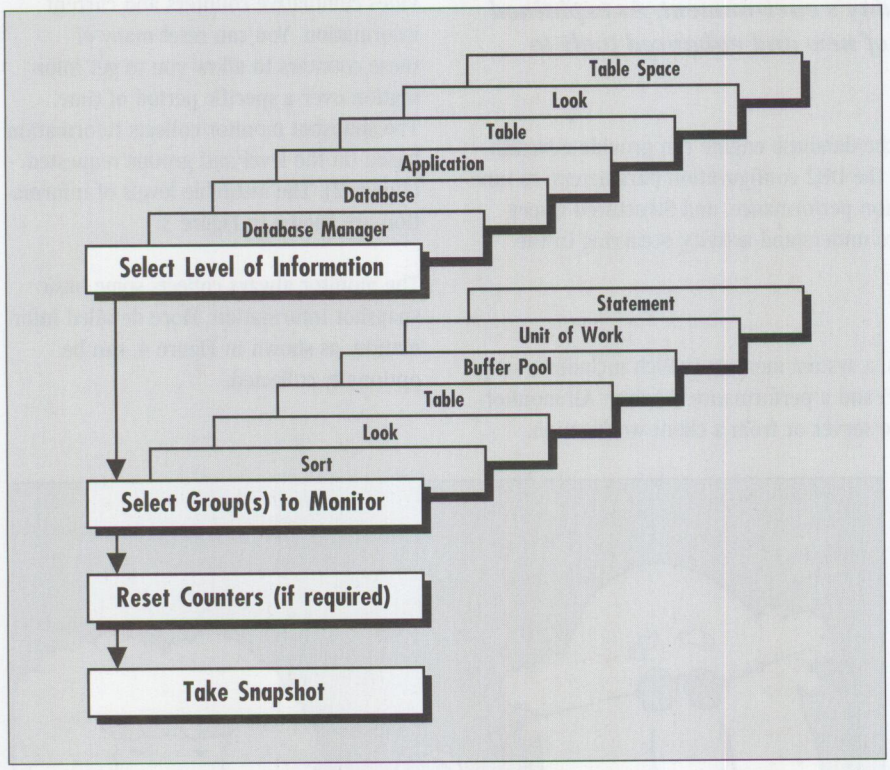


Figure 2. Snapshot Monitor Overview

Level	Applies to
Database Manager	Active instance of DB2
Database	All or selected databases
Application	All or selected applications
Tables	Tables accessed while table switch is on
Locks	Connected applications
Table Spaces	Table spaces accessed while table switch is on

Figure 3. Snapshot Monitor Levels of Information

Event Monitor

The event monitor summarizes the activity that occurred throughout a particular event. It is particularly useful for capturing information about transient events such as connections, transactions, and deadlocks (Figure 5). Information gathered by an event monitor will be sent to either a file or a pipe and will only be reported when the event completes. Figure 6 lists the event types, the information they provide, and when the information is recorded.

Event monitors are database objects that are created via SQL. The `CREATE EVENT MONITOR` statement may be issued interactively or imbedded in an application. The connections, statements, and transactions monitored by an event monitor can be limited by a `WHERE` clause to reduce the information gathered. Unlike the snapshot monitor, the event monitor does not have any data collection switches. All data that it reports is collected as long as the monitor is active.

Performance Monitor

The performance monitor is a graphical user interface (GUI) tool that uses system monitor information to provide graphical viewing, analyzing, alerting, and reporting capabilities (Figure 7). Collected data may be viewed graphically or textually. The collected data can be viewed real time and/or be recorded for later playback and analysis. Recorded data can also be loaded into a performance database for further querying and reporting.

The performance monitor does not report only the absolute values obtained from the system monitor. You can also create formulas for calculating your own performance variables based on the monitored information. A variety of common calculations is provided. For any performance measurement, built-in or user-defined, you can define exception conditions by specifying a threshold value. When this value is reached, the performance monitor will take any of the following actions you specify:

- Notification through a window or audible alarm
- Logging a record in a log file
- Executing a command or program
- Notifying a management product such as the IBM NetView family of products

Explaining DB2

Administrators and application developers have long depended on DB2's Explain capabilities to help understand how DB2

will execute a specific SQL statement. Previous versions of DB2 for OS/2 had an Explain utility that only read access plans after the fact and reported the chosen path options. Version 2.1 introduces a much improved Explain facility that provides much greater detail about the access plan, the conditions that affected the optimizer's decisions, and the costs of each step in the query's execution. Explain information is optionally gathered at statement compile time and is stored in a set of relational tables in the database. Explain information may be gathered on either static or dynamic SQL.

Visual Explain

Visual Explain is a GUI tool that graphically displays Explain information in an easy-to-understand format (Figure 8). Relationships between database objects (tables, etc.) and the operations required for a statement are shown. From this graph, you can obtain detailed information about both the objects and the operations. For objects, the information includes all relevant catalog statistics that the optimizer looks at when determining a plan. For operations, the information includes CPU and I/O cost estimates. This information allows an administrator or developer to quickly locate trouble SQL, and the most expensive operations in that SQL, to help focus the tuning effort.

Visual Explain lets you model the impact of statistical changes in the database environment. For example, you are able to see any changes that more rows of data make to the access plan without actually adding the data.

Managing DB2

Ease of management is a very important criterion in today's environment. DB2 V2.1 improves administration capabilities over previous versions by allowing remote administration. This capability can now be exploited by system management tools like IBM's DataHub. DB2 V2.1 also comes with its own administration tools: the Database Director and a simple network management protocol (SNMP) subagent.

Remote Administration

In previous versions of DB2 for OS/2, the administrative tasks you could perform from a client were limited. Many tasks such as backup, restore, updating the database manager configuration, etc. all had to be performed at the server. In

Group	Information Available
Sorts	Number of heaps used, overflows, sorts performed, time required
Locks	Number held, number of deadlocks, locked object, application ID
Tables	Measure of activity (rows read, rows written)
Buffer pools	Reads/writes, time taken, asynchronous reads, direct reads/writes
Units of work	Start times, end times, wait time, completion status
SQL statements	Number of commits, rollbacks, selects, failures, time

Figure 4. Snapshot Monitor Groups

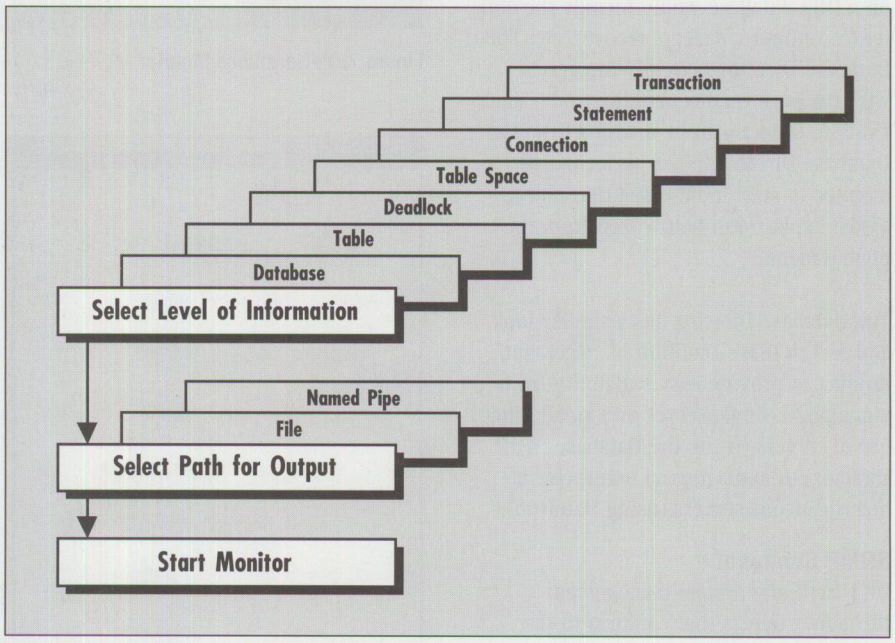


Figure 5. Event Monitor Overview

Event Type	Information Available	Reported When
Database	Statistics on database operation, tables, and table spaces	Last application disconnects
Table	Overflow information	Last application disconnects
Deadlock	Resources and applications involved	Deadlock occurs
Table Space	Details on buffer pool usage	Last application disconnects
Connection	Sorts and SQL statement counts	Application disconnects
Statement	Statement identification	SQL statement completes information
Transaction	Start and stop times	Transaction completes (commit or rollback)

Figure 6. Event Monitor Event Types

version 2.1, this restriction is eliminated. With the exception of the directory services tasks (catalog, uncatalog), all administrative functions are supported from a remote client.

Database Director

The Database Director deletion is a new GUI tool to manage DB2 and its databases. It provides the functionality of the previous DBA tools combined into a single tool. It has an object-oriented user interface that makes performing these tasks very easy. Database Director lets you simply click on any DB2 object and get a menu of all options available for that object (Figure 9). For example, operations on a DB2 database would include the ability to configure, backup, recover, etc. The Database Director is tightly integrated with the performance monitor and Visual Explain. Objects can be dragged from the Database Director to the performance monitor to start monitoring that object. Visual Explain can be invoked from the pop-up menu.

The Database Director has a similar look and feel to IBM's DataHub management product to provide easy scalability from managing a single server or a small number of servers (using the Database Director) to managing an enterprise of heterogeneous servers (using DataHub).

SNMP Subagent

DB2 facilitates system management through products that conform to the industry-standard SNMP, like NetView for OS/2. SNMP is a widely available and well-accepted protocol for managing distributed systems.

The DB2 SNMP subagent is based on the Internet Engineering Task Force (IETF) Relational Database Management System (RDBMS) Management Information Base (MIB). The IETF RDBMS MIB is an emerging industry standard for managing relational databases (RFC 1697).

The DB2 SNMP subagent is a resident program that implements the following functions:

- Generate alerts to an SNMP manager in the case of a severe DB2 error
- Support the attributes of the IETF RDBMS MIB

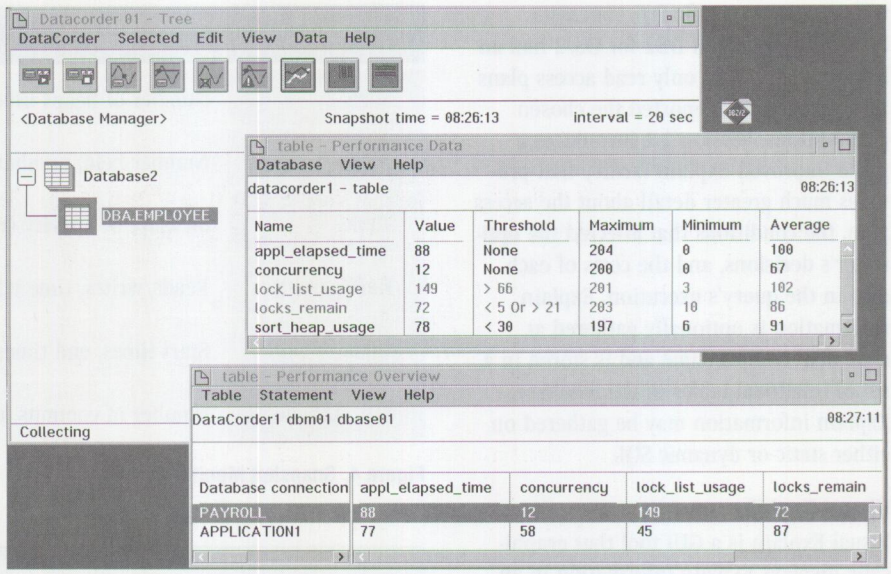


Figure 7. Performance Monitor

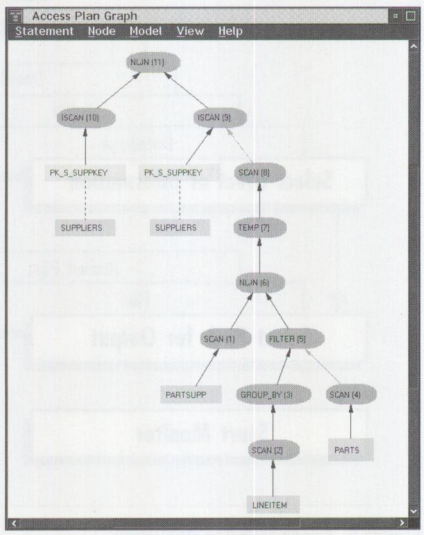


Figure 8. Visual Explain

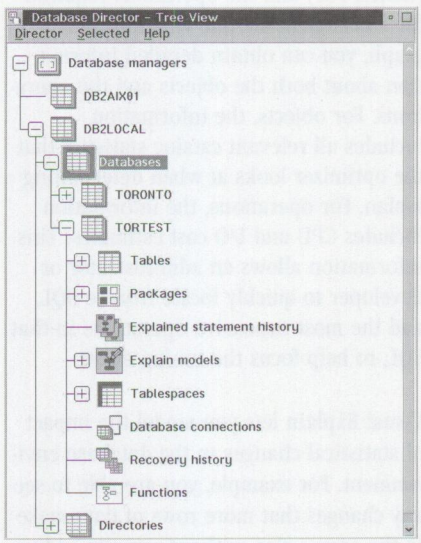


Figure 9. Database Director

- Comply with the Internet Engineering Task Force standard (RFC 1697)

Tools to Make Jobs Easier

DB2 for OS/2 V2.1 has enhanced performance and administrative tools. These tools give administrators and developers access to valuable information to make their jobs easier, plus they provide the insight and information that DB2 users have come to expect, in a very flexible and easy-to-use manner.

Mike Logan is a marketing support representative in the IBM Personal Systems Competency Center, Roanoke, Texas, where he has provided technical support and consulting on DB2 for OS/2 and DDCS for OS/2 since joining IBM in 1993. Mike has a BS degree in Computer Science from the University of North Texas. He can be reached via Internet at mlogan@vnet.ibm.com.

Database Recovery with DB2 for OS/2

As databases grow larger and move closer to 7x24 (7 days a week, 24 hours a day) availability, database administrators increasingly face challenges to implement database recovery strategies within smaller windows of opportunity. To ensure success for these database administrators, DB2 for OS/2 V2.1 provides significant enhancements in database recovery. This article discusses those enhancements.

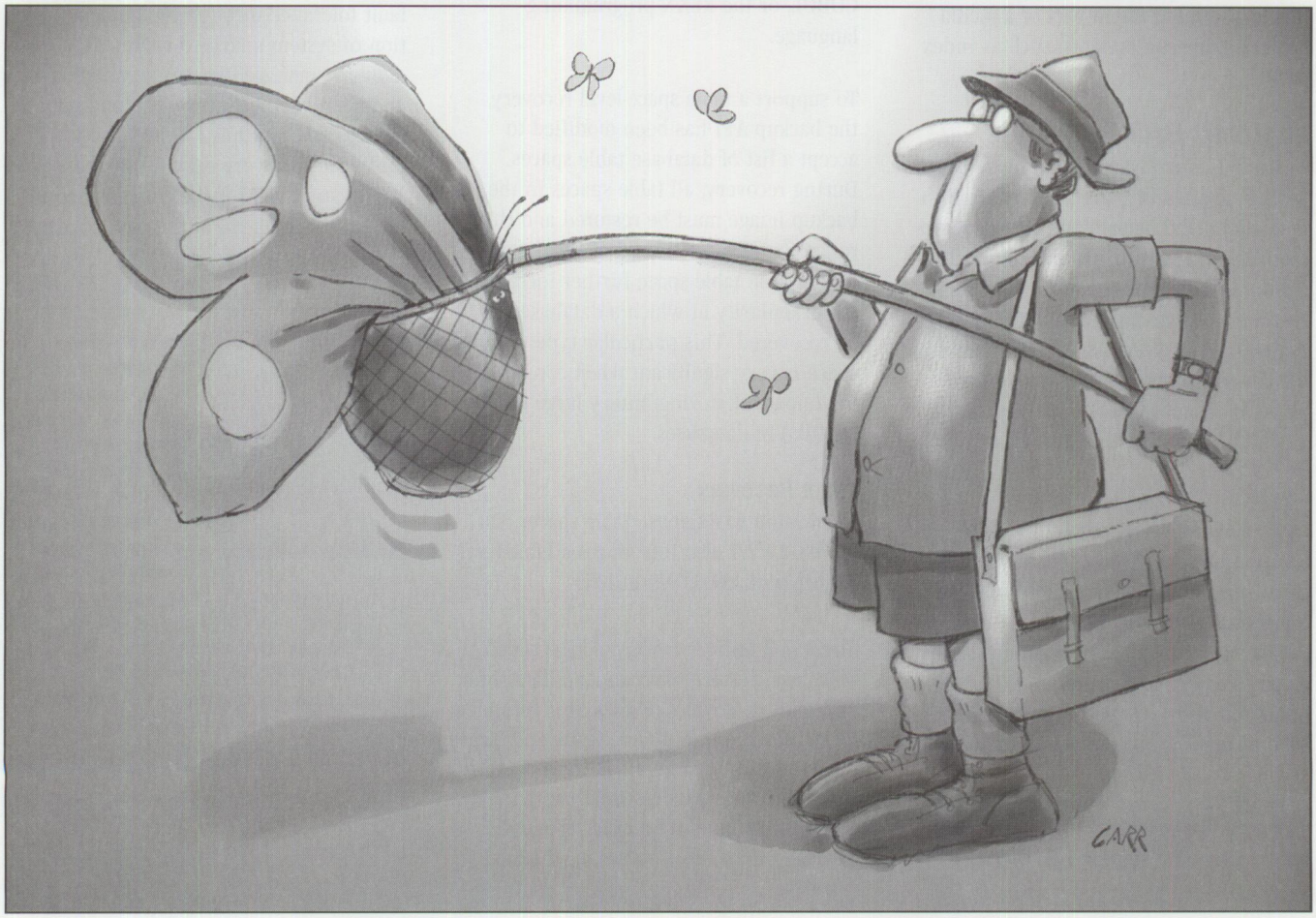
Chris Fierros
IBM Corporation
Kansas City, Missouri

IBM's DB2 for OS/2 V2.1 continues to meet database administrators' expectations for database recovery. This new version includes improvements in fault tolerance, backup and restore processes, and recovery management.

Improved Fault Tolerance

DB2 for OS/2 V2.1 introduces the concept of a database table space. A *database table space* can be viewed as a logical database partition, much like a logical partition on a disk drive.

A table space can be allocated as either a *system-managed space* (SMS) or a *database-managed space* (DMS). An SMS table space is simply a table space managed by the operating system's file system, in this case OS/2. A DMS table space, on the other hand, is managed by DB2 for OS/2 and has some recovery advantages over an SMS table space.



One advantage of a database table space is that it increases the database's fault tolerance to media failures. In previous releases of DB2 for OS/2, a database that failed to restart after a media error would have to be completely restored from a backup copy. In DB2 for OS/2 V2.1, the damage is isolated at the table space level, and the table space is placed in a "pending roll forward" state. This provides two important functions. First, it gives the database administrator the opportunity to recover at the table space level. Second, it allows users to continue to access all other table spaces in the database. In the event of a total disk failure, only those table spaces located on the failed disk are marked as damaged. Once the disk has been replaced, all affected table spaces will have to be restored from a backup copy and will require roll forward recovery.

A DMS table space provides even more fault tolerance by allowing table data, index data, and long field (LONG) data or large object (LOB) data to be separated into different table spaces. Using a DMS table space in this manner provides fault tolerance at the table component level by further isolating the impact of a media error to table components, such as index data.

Improved Backup/Restore

As databases grow larger and move closer to 7x24 availability, the time required to perform critical backup and recovery procedures continues to grow while database maintenance windows continue to shrink. To support large databases with high availability, DB2 for OS/2 V2.1 introduces the concept of parallel backup/restore procedures and increases the granularity at which backup/restore procedures can be performed.

In previous versions of DB2 for OS/2, the backup and restore procedures were executed serially. This meant that only one input/output (I/O) device could be used to back up or restore the database at any given time. It therefore required several hours to back up a very large database, especially if the backup was made to a slow media device such as a tape unit.

A parallel backup/restore procedure will reduce this time significantly, because multiple I/O buffers and multiple I/O devices can be used to back up or restore a single database. To support multiple I/O buffers and multiple I/O devices, a new backup application programming interface (API) (`SQLUBKUP`) and a new restore API (`SQLURSTO`) have been introduced.

Improved Database Recovery

In previous versions of DB2/2, the backup procedure supports only two levels of database backup, either the "entire database" or "changes only." The backup procedure is invoked via the database backup API. The backup API can be invoked with the "changes only" option only after an "entire database" backup has been performed. The restore process is invoked via the database restore API. The restore API can be invoked with the option to either replace the existing database or create a new database. This API also supports the option to place the database in a "pending roll forward" state after the restore has completed.

Both the backup and restore database APIs can be accessed from the recovery tool or the command line interface (CLI). They can also be called directly from C, COBOL, or the REXX programming language.

To support a table space-level recovery, the backup API has been modified to accept a list of database table spaces. During recovery, all table spaces in the backup image must be restored and then rolled forward to the end of the database logs. A DMS table space further increases the granularity at which a database can be recovered. This particular type of table space is very significant when considering the impact of storing binary large objects (BLOBs) in databases.

Table Recovery

In addition to database table spaces, DB2 for OS/2 V2.1 also introduces a table-level restore with the LOAD utility.

The LOAD utility provides a high-speed, table-level restore that can load data from multiple I/O devices or files. It is intended for processing initial table loads and large table appends. Unlike database and table space recovery, data restored using the LOAD utility cannot be rolled forward. However, if an error occurs during load processing, the load process itself can be restarted and/or recovered. The LOAD utility supports delimited ASCII, non-delimited ASCII, and PC/IXF file formats.

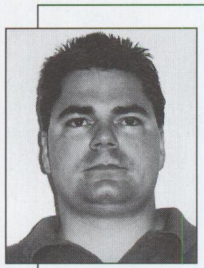
Improved Recovery Management

As managing database recovery becomes more complex, the database recovery features of DB2 for OS/2 continue to be enhanced. To help database administrators, DB2 for OS/2 V2.1 introduces a recovery history file that includes database backup and restore procedures.

Stored with each database, the *recovery history file* provides summary information at the database level. It is automatically updated by DB2 for OS/2 when a database backup, restore, or table load is performed. Using the recovery history file, database administrators identify which portions of the database have been backed up, when they were backed up, and where the backups are located.

Significant Enhancements

IBM's DB2 for OS/2 provides database administrators with significant recovery enhancements. Features such as improved fault tolerance through the implementation of system managed spaces or database managed spaces, parallel backup and restore procedures with multiple I/O devices and multiple I/O buffers, and table-level recovery provide database administrators with the edge needed to stay one step ahead of the next disaster!



Chris Fierros is an information technology specialist in the IBM Consulting Group in Kansas City, Missouri. He provides consulting services to customers developing Executive Information Systems and Decision Support Systems with OS/2 and the DB2 for OS/2 family of database products. Chris joined IBM in 1991 as an OS/2 applications specialist in IBM's Personal Systems Competency Center in Roanoke, Texas. He has a BBA degree in Business Computer Information Systems from the University of North Texas.

Getting Object-Oriented with DB2 for OS/2 V2.1

With DB2 for OS/2 V2.1, IBM has greatly enhanced its premier 32-bit database system. Along with many functional and performance enhancements to the database engine, the new version of DB2 is poised to better support and promote object-oriented concepts and object-oriented development.

What are some of these object-oriented extensions? How can they be used? This article shows how these powerful extensions can be implemented in your environment.

Rick Weaver
IBM Corporation
Roanoke, Texas

DB2 for OS/2 V2.1 comes with several new features that enhance and support object orientation and object-oriented development. These features include large object support, user-defined functions, user-defined data types, and C++ support. Each of these features is discussed in detail in this article.

CHECK Example

Throughout the article, a simple example illustrates these object-oriented concepts. The example requires a table that contains information about a checking account. The CHECK table must store the following data elements:

- Account number
- Check number
- Check payee
- Check date
- Cleared status
- Scanned image of the check (75 KB bitmap)

The examples in this article use the CHECK table as a model.

Large Object Support

Before DB2 for OS/2 V2.1, the maximum length of a row without a LONG VARCHAR was limited to 4096 bytes. If a LONG VARCHAR was used, the maximum size was 32768 bytes. This was limiting if you wanted to store large data elements such as images, sounds, binary executables, or other data elements larger than 32768 bytes.

In this check example, the check's scanned image is larger than 32 KB, so if you were to create the check table with DB2/2 V1.x, you would probably create it using the data definition language (DDL) shown in Figure 1.

In CHECK_IMAGE_PATH, you can then store a pointer (a fully qualified path) to the bitmap image, but not the bitmap itself, in the database.

Although this design is functional, it does open up some interesting considerations:

- When performing a database backup, the backup does not store the bitmap with the database.
- What if the bitmap were moved to another location?
- What process synchronizes the location of the bitmap with the CHECK_IMAGE_PATH column?

In DB2 for OS/2 V2.1, all the above considerations no longer exist, because the new version of DB2 now stores objects up to 2 GB in the database. Three types of large objects (LOBs) can be stored: binary large objects (BLOBs), character large objects (CLOBs), and double-byte character large objects (DBCLOBs).

If you want to implement the same check example in DB2 for OS/2 V2.1, you would use DDL to create the table as shown in Figure 2.

You can then store all the data elements inside the database and eliminate any extra files needed to support large objects. This reduces the database administration as well as the programming effort to maintain the database.

To LOG or NOT LOG

Figure 2 shows the use of the LOGGED option in the DDL to create a table with a BLOB column type. The LOGGED option specifies that any changes to the CHECK_IMAGE column will be logged in the database transaction log. If you use very large LOB elements, then logging to the transaction log may be extremely expensive and create additional input/output (I/O). If you choose the NOT LOGGED option, place the log in a separate table space for recovery (and performance). The

```
CREATE TABLE CHECK (ACCT_NUM CHAR(16) NOT NULL,
                    CHECK_# INTEGER NOT NULL,
                    PAID_TO VARCHAR(50) NOT NULL,
                    CHK_DATE DATE NOT NULL,
                    CLEARED INTEGER NOT NULL,
                    CHECK_IMAGE_PATH VARCHAR(254) NOT NULL)
```

Figure 1. DDL for CHECK Example Using DB2/2 V1.2

```
CREATE TABLE CHECK (ACCT_NUM CHAR(16) NOT NULL,
                    CHECK_# INTEGER NOT NULL,
                    PAID_TO VARCHAR(50) NOT NULL,
                    CHK_DATE DATE NOT NULL,
                    CLEARED INTEGER NOT NULL,
                    CHECK_IMAGE BLOB(75K) LOGGED COMPACT)
```

Figure 2. DDL for CHECK Example Using DB2 for OS/2 V2.1

```
CREATE FUNCTION EMAIL (INT) RETURNS INT
  EXTERNAL NAME 'C:\C\UDF\UDF!email'
  LANGUAGE C
  PARAMETER STYLE DB2SQL
  VARIANT NOT FENCED
  NO SQL EXTERNAL ACTION
```

Figure 3. Registering a UDF with the Database

COMPACT option specifies that minimal disk space will be used for storing LOBs.

DB2 frees any extra disk pages in the last group used by a LOB value. However, storing data this way may penalize performance if the LOB is updated with a larger size.

Making Your Own Functions

DB2 for OS/2 V2.1 introduces a powerful way to expand Structured Query Language (SQL) functionality with user-defined functions (UDFs). *User-defined functions* let you write scalar functions in an external 3GL programming language (the programming language must conform to the ANSI C standard for the calling and linkage conventions) and invoke the UDF via SQL, a trigger, or a view definition. This opens the door to many possibilities:

- UDFs can send an e-mail message when a new row has been added to a queue table.
- UDFs can upload a text file to a mainframe when an SQL statement is issued.
- UDFs can perform any function that the standard SQL function repository does not support.

The only limitations to UDFs are that they must not contain any SQL, and they must return a scalar (single) return value.

Creating a UDF

Back to the check example. Create a UDF that sends an e-mail message and updates the CLEARED column with the UDF's return code. The first step is to register the function to the database by issuing the CREATE FUNCTION DDL shown in Figure 3.

UDF Parameters

In Figure 3, the first line of the CREATE FUNCTION DDL defines the UDF function name known by DB2, the parameter(s) passed to the function, and the datatype returned by the UDF. The EXTERNAL NAME is the name of the DLL (in this case, UDF.DLL) and !email refers to the entry point (function name) in the UDF.DLL. The entry point must be exported in the .DEF file when building the UDF DLL. Additionally, the entry point name and the UDF external name must both have the same case.

The LANGUAGE C line is mandatory for any UDF declaration; it denotes the calling convention and linkage option that the user-defined function must use. Any UDF must conform to the ANSI C prototype for it to function properly as a UDF.

The PARAMETER STYLE DB2SQL, another mandatory clause, specifies the conventions for passing parameters to and from external functions. The DB2SQL value reflects the ISO/ANSI draft standard as of September, 1994.

VARIANT specifies that the UDF may or may not return the same value if the same parameter values are passed to the UDF. NOT VARIANT functions will return the same value if the same parameter values are passed to the UDF.

After completely debugging a UDF, you can configure the UDF to run within the DB2 engine. If a function is NOT FENCED, then it runs within the DB2 engine and is potentially much faster than a FENCED UDF. A FENCED function is safer, because it runs separately outside the DB2 engine, which reduces the risk of database engine failure due to a UDF.

The NO SQL clause is also mandatory, because a UDF cannot execute SQL statements internally. The EXTERNAL ACTION clause specifies that the !email function is going to perform an action outside the domain of DB2 for OS/2 (e.g., sending an e-mail message).

Writing and Calling a UDF

The second step is to write the UDF in a 3GL language such as C. Figure 4 illustrates sample source code for an e-mail function.

Once compiled into a DLL, the UDF can be invoked through the SQL statement shown in Figure 5.

In Figure 5, the SQL statement sets the CLEARED flag in the CHECK table to the return code of the e-mail application programming interface (API) call, as passed by the output variable shown in Figure 4.

UDFs can also be overloaded. Overloaded UDFs simply mean that you can have multiple UDF functions with the same name. If the UDF is overloaded, the arguments or data types must be different so that DB2 can determine which UDF to call.

UDFs can also be registered to other source functions. If you want to use the

built-in function avg with a new User-Defined Type (UDT) (discussed later), then a UDF can be created that will use the source average function. The DDL to create a source function would look like this (all on one line):

```
CREATE FUNCTION
     avg(newdatatype)
     RETURNS newdatatype
     SOURCE SYSIBM.AVG(INT)
```

Unlimited Typ(ing)

User-Defined Types (UDTs) support the creation of non-standard data types that are derived from existing built-in DB2 data types. Once created, the UDT can then be used in DDL to create tables, triggers, and so on. UDTs provide data abstraction from the base data type, which promotes encapsulation and provides a foundation for future object-oriented extensions.

UDTs can only be created over the built-in DB2 data types. If the UDT is not created upon a BLOB, CLOB, DBCLOB, LONG VARCHAR, or LONG VARGRAPHIC, then DB2 can create comparison functions on the new data type if the WITH COMPARISONS clause is specified in the CREATE DISTINCT TYPE statement. These comparison functions include: =, >=, >, <, and <=. Additionally, casting functions will be created to cast the UDT back to its base data type, and the base data type back to the UDT.

For the BLOB, CLOB, LONG VARCHAR, and LONG VARGRAPHIC data types, you must create UDFs to support comparison with those data types.

In the check example, if you want to create a UDT BITMAP in place of the BLOB data type, issue the following DDL statement:

```
CREATE DISTINCT TYPE
     BITMAP AS BLOB(75K)
```

Once the data type is created, the DDL to create the new check table is shown in Figure 6.

An error will occur if you create a UDT of Boolean (BOOL) to represent the CLEARED column name (CREATE DISTINCT TYPE BOOL AS INTEGER WITH COMPARISONS), and then want to issue an SQL statement:

```
SELECT * FROM CHECK
     WHERE CLEARED = 1
```

```
/*UDF to send an email message*/
#include <stdlib.h>
#include <stdio.h>
#include <sqlsystm.h>
void SQL_API_FN email
 (
   short *input,          /*ptr to input argument*/
   short *output,         /*ptr to output argument*/
   short *input_ni,       /*ptr to input null indicator*/
   short *output_ni,      /*ptr to output null indicator*/
   char sqlstate[6],      /*sqlstate*/
   char fname[28],        /*fully qualified function name*/
   char finst[19],        /*function specific name*/
   char msg[71])          /*msg text buffer*/
 {
   int rc=0;              /*return code*/
   char emailstr[80];
   char user[9]="XXXXXXXX";

                          /*Send e-mail message, return 0 if
                          successful, return 1 if unsuccessful*/
   rc=emailstd(user,emailstr);
   *output=rc;            /*returns rc back to calling SQL
                          statement.*/

   return;
 }
```

Figure 4. Sample Source Code for an E-Mail Function

```
UPDATE CHECK
SET CLEARED=email(CHECK_#)
WHERE CHECK_#=:check_num;
```

Figure 5. SQL Statement to Invoke the UDF

```
CREATE TABLE CHECK (ACCT_NUM CHAR(16) NOT NULL,
                    CHECK_# INTEGER NOT NULL,
                    PAID_TO VARCHAR(50) NOT NULL,
                    CHK_DATE DATE NOT NULL,
                    CLEARED INTEGER NOT NULL,
                    CHECK_IMAGE BITMAP LOGGED COMPACT)
```

Figure 6. DDL for Creating a UDT BITMAP

Since the comparison is between a BOOL and an INT (integer), the numeric 1 must be cast into a BOOL data type for the SQL statement to work. (DB2 cannot compare two different data types without a cast.) To fix the SQL statement, do the following:

```
SELECT * FROM CHECK
     WHERE CLEARED = BOOL(1)
```

C++ Support

If you were developing in C++ with prior versions of DB2/2, it was difficult to embed SQL in an application, because the DB2/2 V1.x precompiler did not support or allow SQL to be embedded within a C++ module. The solution for Version 1.x C++ applications was to build function wrappers in C, embed the SQL in C code, and invoke those functions from C++.

In DB2 for OS/2 V2.1, the precompiler can now preprocess a C++ program. Within a C++ program, you can embed host variables in a class called data members and

```
CHECK.HPP
//Check class definition
class CHECK
 {
   EXEC SQL BEGIN DECLARE SECTION;
   char acct_num[17];
   long check_num;
   char paid_to[51];
   char chk_date[12];
   long cleared;
   sql type is blob image(75k) *check_image;
   EXEC SQL END DECLARE SECTION;
   public:
   ...
   check(char *, long, char *, char *, long, PBYTE, long);
//Constructor
 ~check(void); //Destructor
 };
************************************************************
CHECK.SQC

...
EXEC SQL INCLUDE 'CHECK.HPP';

void check::check(char *acct, long ch_num, char *pay, char
*chk_dt, long cl, PBYTE image, long imagelen) :
check_num(ch_num), cleared(cl), check_image(new check_image_t)
 {
   strcpy(acct_num,acct);
   strcpy(paid_to,pay);
   strcpy(chk_date,b);
   check_image->length=imagelen;
   memcpy(check_image->data,image,imagelen);

   EXEC SQL INSERT INTO CHECK VALUES(:acct_num, :check_num,
:paid_to, :check_date, :cleared, :*check_image);
...
 }

check::~check(void)
 {
   delete check_image;
 }
```

Figure 7. Using C++ in SQL

can embed SQL statements within member functions.

A sample of using C++ with SQL is shown in Figure 7. For any LOB object data member, you should make sure that the data member is declared as a pointer to the LOB (as shown by the new operator in the constructor). If the object data member is declared as a normal variable, the memory to store the object is allocated on the stack and can cause stack overflow errors. Allocating the memory for the LOB on the heap through the new operator eliminates this problem and is much more efficient.

DB2 will use this pointer *only* to resolve references to host variables. A host variable cannot be referenced by explicitly qualifying an object instance (i.e., `SELECT CLEARED FROM CHECK WHERE CHECK_# = :my_obj.check_num`).

As with C programs, all host variables must be unique within a C++ module, even if the variables are local to a class or function.

Get Object-Oriented!

With its large object support, user-defined functions, user-defined types, and C++ support, the DB2 for OS/2 V2.1 product is poised to greatly enhance your object-oriented development effort. Exploitation of each of these extensions, along with with the addition of triggers in the new DB2, opens a new world of increased functionality and productivity for DB2 for OS/2 developers, administrators, and end users.

Rick Weaver is an advisory marketing support representative in the IBM Personal Systems Competency Center in Roanoke, Texas. He provides services and on-site consulting on DB2 for OS/2, DDCS for OS/2, SOM, and object technology on OS/2 to customers throughout the U.S. He previously provided Q&A support for OS/2 Database Manager. Rick joined IBM in 1990 after earning a BS degree in Computer Science from Southwestern Adventist College. His Internet ID is `weaveros2@vnet.ibm.com`.

Enhanced SQL in DB2 for OS/2 V2.1

This article discusses and provides examples of the new SQL functions in DB2 for OS/2 Version 2.1.

For the new user in a relational database environment, Structured Query Language's (SQL's) power and ease of use quickly becomes apparent. Programs that in the past required many lines of code now become just a single SQL statement. Programs that required days to code and debug can now be written in minutes using an SQL statement.

To meet user requirements for additional function and increased stability, DB2 for OS/2 V2.1 delivers the following significant SQL enhancements:

- Application Server (AS) clause in select list
- IN list expressions
- Scalar fullselect expressions
- Self-referencing subqueries for DELETE, INSERT, and UPDATE
- Nested table expressions
- Common table expressions
- Recursive query support
- Triggers

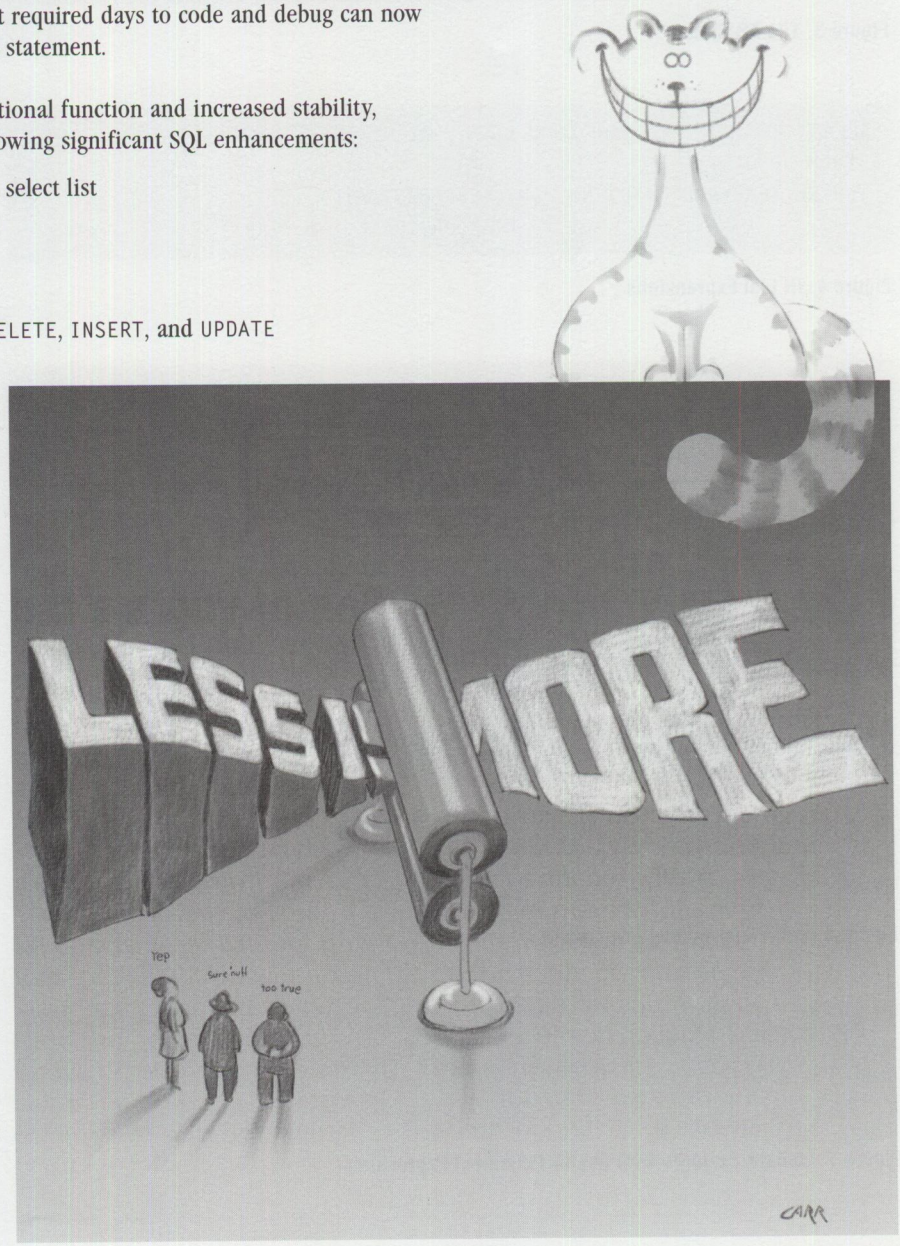
John Casey
IBM Corporation
Roanoke, Texas

Following are details and examples of each of these enhancements. Several of the examples are based on the sample tables shipped with DB2 for OS/2.

AS Clause in Select List

The AS clause lets you name an expression in the select list of an SQL statement. Without this capability, you would have to use the sample SQL statement in Figure 1 to order an answer set based on an expression in the select list.

This solution becomes error-prone when you modify the content of the select list. Using the AS clause, this request can be coded as shown in Figure 2.



TECHNICAL

```
SELECT EMPNO, LASTNAME, SALARY + COMM
  FROM EMPLOYEE
  ORDER BY 3
```

Figure 1. Sample SQL Statement Without AS Clause

```
SELECT EMPNO, LASTNAME, SALARY + COMM AS PAY
  FROM EMPLOYEE
  ORDER BY PAY
```

Figure 2. Sample SQL Statement with AS Clause

```
CREATE VIEW PRJ_LEADER
  AS SELECT PROJNO, PROJNAME, DEPTNO, RESPEMP, LASTNAME,
    SALARY+BONUS+COMM AS TOTAL_PAY
  FROM PROJECT, EMPLOYEE
  WHERE RESPEMP = EMPNO AND PRSTAFF > 1
```

Figure 3. The AS Clause

```
SELECT PROJNO, PROJNAME, PRSTDATE, PRENDATE
 FROM PROJECT
 WHERE YEAR(PRENDATE) NOT IN (YEAR(PRSTDATE),
                              YEAR(PRSTDATE + 1 YEAR))
```

Figure 4. IN List Expressions

```
UPDATE T1
  SET C1 = (SELECT D1 FROM T2 WHERE D2 = :hv1)
  WHERE (SELECT F1 FROM T3 WHERE F2 = :hv2) IS NOT NULL
```

Figure 5. Scalar Fullselect Expressions

```
DELETE FROM TABLE1
   WHERE COL2 IN (SELECT COL1 FROM TABLE1)
```

```
INSERT INTO TABLE1 (SELECT * FROM TABLE1
```

```
UPDATE TABLE1 SET COL1 = NULL
   WHERE COL2 IN (SELECT COL1 FROM TABLE1)
```

Figure 6. Self-Referencing Subqueries

```
UPDATE EMPLOYEE A SET SALARY =  SALARY * 1.10
  WHERE SALARY < (SELECT AVG(SALARY) - 10000 FROM EMPLOYEE B
                  WHERE A.WORKDEPT = B.WORKDEPT)
```

Figure 7. Update Example with Scalar Fullselect Expression

The AS clause simplifies CREATE VIEW by eliminating the need to code a select list just because of derived columns, as shown in Figure 3.

In addition, when the AS clause is used, the AS value will be returned in the SQLNAME field in the SQLDA when an SQL DESCRIBE statement is executed.

IN List Expressions

The IN list can now contain an expression. The SQL statement shown in Figure 4 uses expressions in an IN list and returns information about projects where the year of the ending date is neither the same year as the starting date nor the year following the starting date.

Scalar Fullselect Expressions

A scalar fullselect returns a single value: one row of data consisting of one column. A scalar fullselect in an expression allows you to retrieve data values from the database for use in an expression. You can use an expression in such places as predicates and select lists. For example, in Figure 5, a scalar subquery is used in multiple places in the UPDATE statement.

The UPDATE statement in Figure 5 contains a scalar fullselect in the SET clause and in the WHERE clause, and is only valid if the host variables provided results in either one row or no rows. When no rows are returned, the value is NULL.

Self-Referencing Subqueries for DELETE, INSERT, and UPDATE

Within the statement itself, the DELETE, INSERT, and UPDATE statements can reference the same table that is the object of the statement. As an example, the SQL statements in Figure 6 will no longer return an error message but will execute successfully.

The subqueries are evaluated before any changes are applied to the table, so that the result of the statement is clearly defined.

The UPDATE example in Figure 7 gives everyone a 10 percent raise if their salary is $10,000 below the average salary of their department, and it uses a scalar fullselect expression that is correlated and self-referencing.

Nested Table Expressions

A nested table expression is a result table obtained from one or more other tables through evaluating a fullselect specified in the FROM clause. The example in Figure 8 uses a nested table expression and gives the average total pay, by education level and year of hire, for employees with an education level greater than a user-specified education level.

In this query, a nested table expression first extracts the year of hire from the HIREDATE column for subsequent use in the GROUP BY clause. This capability allows the indirect use of a name from the select list's AS clause in the GROUP BY clause. This is especially useful when you need to group by the result of an expression.

The example in Figure 9 uses a nested table expression and gives information about sales representatives and the average salaries and headcounts of their departments.

Common Table Expressions

A common table expression is a named result table defined in a WITH clause prior to a fullselect. The common table expression's identifier in the WITH clause can be used as a table name in any FROM clause throughout the remainder of the fullselect that follows. Repeated uses of the common table expression identifier as a table name will use the same result table each time. This differs from a view that may determine the result table for each reference with possibly different results.

The example in Figure 10 uses a common table expression as part of a query listing employees whose average total pay is less than the average for other employees who were hired in the same year and have the same education level.

In Figure 10, the first common table expression, called PAYLEVEL, includes the year of hire and the total pay for each employee, based on a selected minimum education level specified in a host variable. This common table expression's column names are determined from the names used within the select list of the subselect following the AS keyword.

The second common table expression in the WITH clause, called PAYBYED, uses the first common table expression to determine the average total pay by education level and year of hire. In this case, the columns were given names in the column list following the common table expression's identifier.

Following the WITH clause, the fullselect performs the query that joins the two common table expressions by education level and year of hire, then selects the employees whose total pay is less than the average total pay.

Recursive Query Support

You can create recursive queries by using common table expressions. The recursive query example in Figure 11 lists all of the managers to whom employee 000340 reports either directly or indirectly.

```
SELECT EDLEVEL, HIREYEAR, AVG(TOTAL_PAY) AS AVG_TOTAL_PAY
FROM (
      SELECT EMPNO, YEAR(HIREDATE) AS HIREYEAR, EDLEVEL,
             SALARY + BONUS + COMM AS TOTAL_PAY
        FROM EMPLOYEE
        WHERE EDLEVEL > :edlvl
      ) AS PAYLEVEL
GROUP BY EDLEVEL, HIREYEAR
```

Figure 8. Nested Table Expressions—Example A

```
SELECT THIS_EMP.EMPNO, THIS_EMP.SALARY,
       DINFO.AVGSALARY, DINFO.EMPCOUNT
FROM EMPLOYEE THIS_EMP,
   (SELECT OTHERS.WORKDEPT AS DEPTNO,
        AVG(OTHERS.SALARY) AS AVGSALARY,
        COUNT(*) AS EMPCOUNT
   FROM EMPLOYEE OTHERS
        GROUP BY OTHERS.WORKDEPT
   ) AS DINFO
WHERE THIS_EMP.JOB = 'SALESREP'
   AND THIS_EMP.WORKDEPT = DINFO.DEPTNO
```

Figure 9. Nested Table Expressions—Example B

```
WITH
  PAYLEVEL AS
    (
    SELECT EMPNO, YEAR(HIREDATE) AS HIREYEAR, EDLEVEL,
      SALARY+BONUS+COMM AS TOTAL_PAY
    FROM EMPLOYEE
    WHERE EDLEVEL > :edlvl
    ),
PAYBYED (EDUC_LEVEL, YEAR_OF_HIRE, AVG_TOTAL_PAY) AS
    (
    SELECT EDLEVEL, HIREYEAR, AVG(TOTAL_PAY)
      FROM PAYLEVEL
    GROUP BY EDLEVEL, HIREYEAR
    )
SELECT EMPNO, EDLEVEL, YEAR_OF_HIRE, TOTAL_PAY,AVG_TOTAL_PAY
  FROM PAYLEVEL, PAYBYED
  WHERE EDLEVEL = EDUC_LEVEL
    AND HIREYEAR = YEAR_OF_HIRE
    AND TOTAL_PAY < AVG_TOTAL_PAY
```

Figure 10. Common Table Expressions

```
WITH RECMGR (EMPNO,LASTNAME,WORKDEPT,
  MGR_EMPNO,MGR_LASTNAME,MGR_WORKDEPT,MGR_ADMRDEPT)
AS (SELECT E.EMPNO, E.LASTNAME, E.WORKDEPT,
    MGR.EMPNO, MGR.LASTNAME, MGR.WORKDEPT, D.ADMRDEPT
  FROM EMPLOYEE MGR, DEPARTMENT D, EMPLOYEE E
    WHERE E.EMPNO = '000340'
      AND E.WORKDEPT = D.DEPTNO
      AND D.MGRNO = MGR.EMPNO
  UNION ALL
  SELECT MGR.MGR_EMPNO, MGR.MGR_LASTNAME, MGR.MGR_WORKDEPT,
      E.EMPNO, E.LASTNAME, E.WORKDEPT, D.ADMRDEPT
    FROM RECMGR MGR, DEPARTMENT D, EMPLOYEE E
      WHERE MGR.MGR_ADMRDEPT = D.DEPTNO
        AND D.MGRNO = E.EMPNO
        AND MGR.MGR_EMPNO <> 000010'
  )
SELECT * FROM RECMGR
  ORDER BY MGR_WORKDEPT DESC
```

Figure 11. Recursive Queries

```
CREATE TRIGGER new_hire
              AFTER INSERT ON employee
              FOR EACH ROW
              UPDATE company_stats SET  nbemp = nbemp + 1
CREATE TRIGGER former_emp
              AFTER DELETE ON employee
              FOR EACH ROW
              UPDATE company_stats SET nbemp = nbemp - 1
```

Figure 12. Triggers

This is a *recursive* query because the fullselect creating the common table expression RECMGR is a union, with the subselect following the union referring to RECMGR in its FROM clause.

To understand this query, it may be helpful to consider the subselect prior to the union as initializing the query. The result of this subselect is the employee and immediate manager.

The subselect following the union is the iterating or recursive part of the query. To find the next manager, the predicate of this subselect relates already-selected values from RECMGR to values from the DEPARTMENT and EMPLOYEE tables. Recursion continues until this subselect returns a null answer set. In the example, recursion terminates when the query tries to find a manager for the president (MGR.MGR_EMPNO <> '000010').

Triggers

A trigger defines a set of actions that are executed, or "triggered," by an update operation on a specified base table. An update operation includes INSERT, UPDATE, and DELETE SQL statements. When such an SQL update operation executes, the trigger is said to be fired. You can use triggers to:

- Automatically generate a value for a newly inserted row
- Read from other tables for cross referencing purposes
- Write to other tables for audit trail purposes
- Support alerts (for example, electronic mail messages)

You can also use triggers to support general forms of integrity such as business rules. For example, a business may want to refuse orders that exceed the customer's credit limit. Triggers can perform tasks such as automatically updating summary data. Triggers are coded once, consistently enforced at all times, and significantly simplify the task of ensuring database integrity.

For example, to automatically track the number of employees in a company statistics table, you could define the two triggers in Figure 12.

The first trigger adds one to the employee count in the COMPANY_STATS table when a row is inserted into the EMPLOYEE table. The second trigger subtracts one from the employee count in the COMPANY_STATS table when a row is deleted from the EMPLOYEE table.

Triggers can be fired before or after the update operation executes. A trigger can contain multiple SQL statements of the following types:

- INSERT
- searched UPDATE
- searched DELETE
- SELECT

Triggers cannot be defined on tables that may be updated by enforced referential constraints. Triggers can, however, coexist with the referential constraints RESTRICT or NO ACTION.

With these SQL enhancements, DB2 for OS/2 significantly broadens the power available to you. These enhancements to SQL will make difficult tasks easier and impossible tasks feasible.

John Casey is a consulting market support representative in the Relational Systems Support department, IBM Dallas Systems Center. He provides assistance and services to customers implementing distributed database solutions that involve DRDA, DataPropagator Relational, and Visual Warehouse. John has been a key member of the IBM national relational database technical support team for the past 14 years, holds a BS in Accounting from Penn State, and has worked for IBM for 29 years.

Enterprisewide Connectivity Using DB2

DB2 for OS/2 V2.1 and Distributed Database Connection Services (DDCS) V2.3 offer a rich set of new distributed functions and features that enables you to design enterprisewide client/server systems. This article gives an overview of these distributed features on a LAN and in an enterprise environment.

In a local area network (LAN) environment, the DB2 for OS/2 database supports the NetBIOS, internet packet exchange/sequenced packet exchange (IPX/SPX), transmission control protocol/internet protocol (TCP/IP), and advanced program-to-program communication/advanced peer-to-peer networking (APPC/APPN) protocols.

DOS, Windows, OS/2, AIX, Hewlett-Packard, Sun, and Apple workstations can use the call level interface (CLI) call from X/Open to access the DB2 database, while Windows applications can use open database connectivity (ODBC). All that is required for connectivity at the client workstations is the Client Application Enabler, which now comes packaged with the server. With the installation of a Software Developer's Toolkit, these client workstations can also be used as development platforms.

Figure 1 shows the local database connectivity features in DB2 for OS/2 V2.1.

Distributed Unit of Work (DUW) is now possible with the latest DB2 for OS/2 release, allowing applications to connect to multiple databases within a unit of work. For example, an application can read data from one database, then update another database, and either commit or roll back the unit of work.

Bill Wong
IBM Canada Ltd.
North York, Ontario, Canada

Stored procedures are a useful way to integrate data and programming logic. From a connectivity perspective, they are also useful in minimizing network traffic on the LAN.

DB2 for OS/2 V2.1 still supports the old stored procedure–Database Application Remote Interface (DARI). In addition, this latest database release introduces two new kinds of stored procedures:

- SQL Call statement
- Compound SQL

The SQL Call statement is supported through static SQL, but the procedure name can be specified as a runtime value through a host variable. In many cases, the old DARI call can be replaced by the SQL Call statement, and we recommend that the SQL Call statement be used in the future.

Figure 2 illustrates the new stored procedures.

The Compound SQL statement lets workstation applications batch several SQL statements and send them to the server as one executable block. This can be very useful for transaction-based programs since prior releases required each insert, update, or delete to access the server.

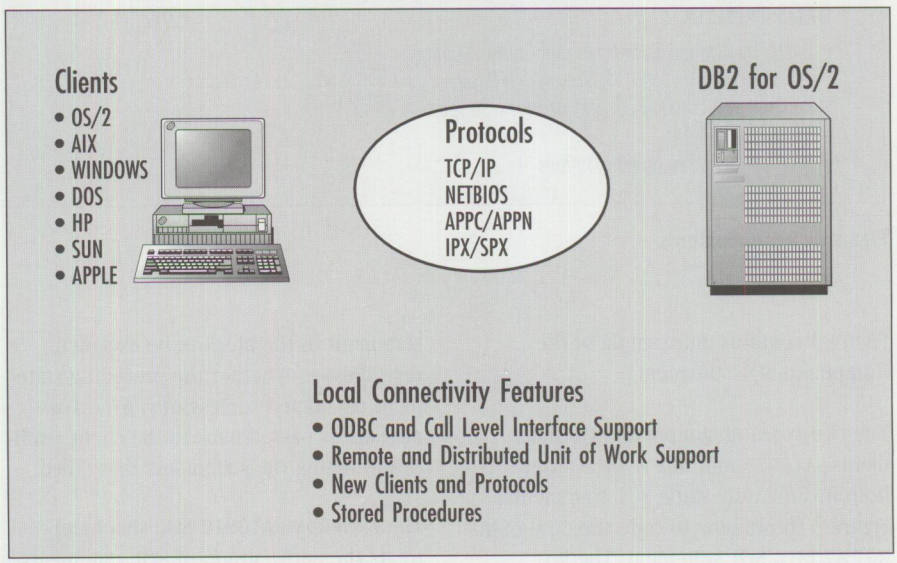


Figure 1. Local Database Connectivity

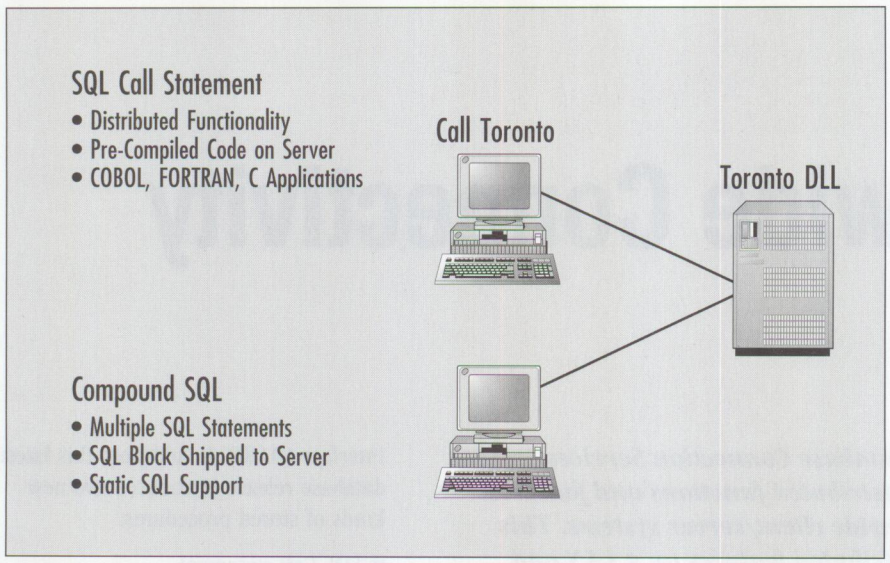


Figure 2. New Stored Procedures

```
EXEC SQL BEGIN COMPOUND NOT ATOMIC
   INSERT INTO BILL.STAFF (ID, NAME) VALUES ('10', 'DANA');
   INSERT INTO BILL.STAFF (ID, NAME) VALUES ('11', 'SHIRLEY');
   INSERT INTO BILL.STAFF (ID, NAME) VALUES ('12', 'TAYLOR');
END COMPOUND;
```

Figure 3. Compound SQL Statement

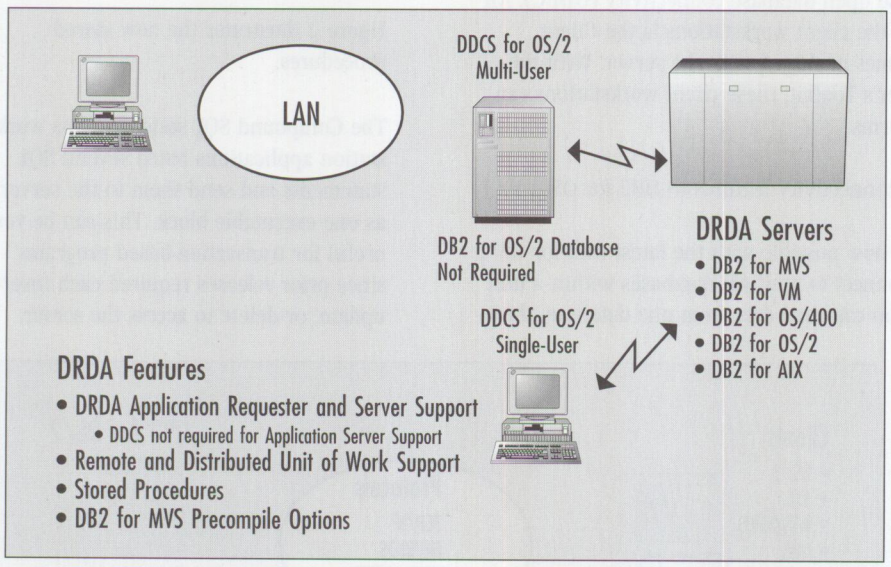


Figure 4. DRDA Features

Figure 3 contains an example of the Compound SQL statement.

The two types of compound SQL statements—ATOMIC and NOT ATOMIC—can both include only static SQL statements. Figure 3 shows how to code the COMPOUND NOT ATOMIC SQL statement. The NOT ATOMIC statement allows each SQL statement in the block to be executed, regardless of whether the preceding statement completed successfully. An error message is passed back to the client application listing the statements that failed.

The COMPOUND ATOMIC SQL statement treats the entire group of SQL statements as a unit of work. If one statement in the group fails, the entire block is considered to have ended in error, and any changes made to the database within the block will be rolled back.

Enterprisewide Client/Server Database Access

Distributed Relational Database Architecture (DRDA) is a client/server architecture built into the DB2 for OS/2 database that allows the local DB2 for OS/2 clients to access any remote database from the DB2 family of databases. Host code is not required at the remote database. DRDA gives organizations a much more cost-effective way to implement enterprise-wide client/server database applications. Figure 4 illustrates DRDA's features.

DDCS, a separate product, is required on the server to enable the clients to access data throughout the enterprise. The newest release of DDCS, DDCS for OS/2, allows organizations to install just DDCS—not purchase a database as a prerequisite. Since many organizations' corporate data resides on a mainframe, DDCS enables these organizations to distribute the application code to the client workstations while keeping the data centralized.

Developers requiring both local and remote data can still install DB2 for OS/2 and the DDCS code at the server. However, with the new packaging options, one server can be dedicated for remote database activity while another is dedicated to meeting local database requests. All of this is transparent to the client application.

With the introduction of DB2 for MVS V4.1, DB2 positions itself ideally as the enterprise database server. DB2 for MVS V4.1 has increased the number of remote connections from 10,000 to 25,000. A site that runs DB2 as a data-sharing sysplex can support 800,000 clients on a 32-way sysplex.

DB2 for OS/2 can act as a DRDA Application Requester (AR) by passing SQL requests from local workstation clients to a remote database server. The latest release now allows DB2 for OS/2 to be an Application Server (AS) and accept SQL requests from other remote database servers. Thus, DB2 for MVS can be a client to DB2 for OS/2, the server. The DRDA AS function is built into the database. No additional software is required to enable the AS capability, not even DDCS 2.3.

Both the Compound SQL statement and the SQL Call statement are supported by DRDA (DARI is not supported using DRDA). The Compound SQL statement can flow from the client all the way to a remote DRDA Application Server. Only the NOT ATOMIC Compound SQL statement is supported in this environment.

Figure 5 shows how to use stored procedures with DRDA.

Support for the SQL Call statement across DRDA enables the stored procedure at the DRDA Application Server site to embed not only SQL calls, but also non-relational database requests. Using DB2 for MVS as an example, the stored procedure can be invoked to access IMS, VSAM, or a third-party database. The procedures can be written in COBOL, PL/I, C, Assembler, or using an application generator like VisualGen.

Building Enterprise Client/Server Systems

To access multiple databases, the DUW precompile options in Figure 6 are available.

The first option, CONNECT, specifies whether an SQL CONNECT statement should be processed as a remote unit of work request, TYPE 1 CONNECT, where only a single database is accessed in a unit of work. TYPE 2 specifies that DUW is required or multiple database connections are required.

The SQLRULES option allows you to specify if you want to have the TYPE 2 CONNECT statements, DUW, follow either DB2 rules of switching connections or the ISO/ANS SQL92 rules.

The SYNCPOINT option specifies how commits or rollbacks will be coordinated among multiple database connections. ONEPHASE allows multiple database connections, but only one site can be updated within a unit of work, while all other database connections must be read-only. TWOPHASE specifies that a two-phase commit is required across multiple databases. Finally, the NONE option specifies that the application is responsible for recovery and for updating multiple databases.

In a DRDA environment with a DB2 for MVS and OS/2 database, a syncpoint manager is required for a multi-site update.

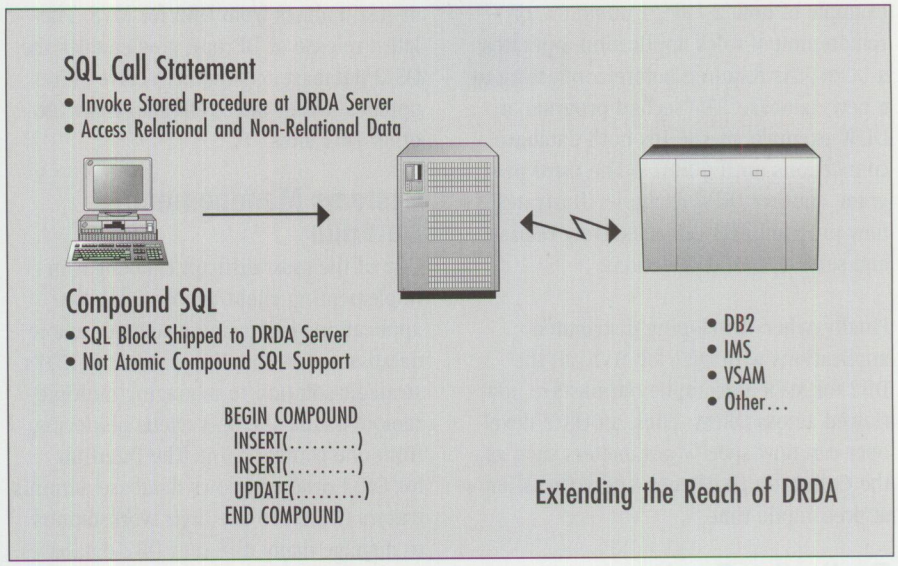


Figure 5. Stored Procedures Using DRDA

- CONNECT (1] 2)
- SQLRULES (DB2] STD)
- SYNCPOINT (ONEPHASE] TWOPHASE] NONE)
- DISCONNECT (EXPLICIT] AUTOMATIC] CONDITIONAL)

Figure 6. DUW Precompile Options

TYPE 1 CONNECT	TYPE 2 CONNECT	TYPE 2 CONNECT
CONNECT TO D1 SELECT UPDATE COMMIT	CONNECT TO D1 SELECT UPDATE	CONNECT TO D1 SELECT UPDATE
CONNECT TO D2 INSERT COMMIT	CONNECT TO D2 INSERT	CONNECT TO D2 INSERT RELEASE CURRENT
CONNECT TO D1 SELECT COMMIT CONNECT RESET	CONNECT TO D1 SELECT RELEASE ALL COMMIT	CONNECT TO D1 SELECT RELEASE D1 COMMIT

Figure 7. Distributed Applications Examples

This feature will be part of an upcoming release of Communications Manager.

The DISCONNECT option specifies which database connections are released at the COMMIT. EXPLICIT releases database connections marked for release by the SQL RELEASE statement; AUTOMATIC releases all the connections; and CONDITIONAL releases connections that have no open cursors.

Figure 7 lists examples of distributed applications.

The three programs referenced in Figure 7 illustrate some of the new precompile options. The first program is an

example of how a TYPE 1 CONNECT, or remote unit-of-work application, operates; a COMMIT is required before connecting to a new database. The second program, a DUW example, maintains both database connections until COMMIT. The third program, another DUW example, illustrates how programmers can selectively release and set database connections.

Finally, when developing distributed applications with DB2 for MVS, all the DB2 for MVS precompile options are now flowed across DRDA. Thus, an OS/2 developer can now specify parameters such as the Collection ID, Owner, and a Qualifier at precompile time.

Replication Services: DataPropagator Relational

Many times organizations must provide a copy of corporate data to another business unit at a remote site. In its first release, DataPropagator Relational (DPropR) was able to copy a table from one DB2 for OS/2 site to another site, as well as to receive either table refreshes or just updates from DB2 for MVS. The latest release of DPropR now enables the OS/2 database to send or receive either updates or full table refreshes from the other DB2 sites.

Systems Management: DataHub

One of the most difficult challenges in implementing client/server database applications is managing several remote databases. The DataHub product is IBM's strategic solution to managing multiple remote databases, on various platforms, from one control point. The DataHub for OS/2 product allows database administrators (DBAs) to use their workstations to manage many different DB2 databases, regardless of whether the DB2 data resides on OS/2, AIX, AS/400, VM, or MVS.

Meeting the Challenge

The new DB2 and DDCS for the OS/2 environment offer unprecedented options in data placement and access. Many of the new connectivity features, such as the ability to access non-relational data through stored procedures, are exclusive to the DB2 family of databases. DB2 for OS/2 V2.1 and DDCS V2.3 are designed to meet the challenge of distributing data throughout the enterprise.

Bill Wong is a DB2 technology specialist in Software Solutions, Toronto Laboratory, IBM Canada Ltd. He provides worldwide technical support for the DB2 and DDCS products on workstations, and he helps customers position the use of DB2 technology within their organizations. Bill frequently participates in trade shows and speaks to DB2 user groups worldwide. His previous work includes DB2 and CICS systems programming on MVS and serving as a DB2 DBA. Bill has taught courses on database management and strategic IT planning at the University of Toronto.

Visualizer Development

Object-based development for OS/2 has ease of use, powerful function, and access to data, enabling you to develop decision support applications at warp speed. This article gives you a test drive of this powerhouse–IBM's Visualizer Development.

Jerris L. Riffel
IBM Corporation
Roanoke, Texas

Visualizer Development, a development tool, enables you to build powerful, complex, customized decision support applications quickly and easily. But describing it, particularly in a magazine article, is not easy–you have to see it to appreciate it. So let's take a look at what it would take to build a simple temperature conversion application with Visualizer Development.

This application has (as shown in Figure 1) a window with a menu, so you can choose whether you want to convert Celsius to Fahrenheit or Fahrenheit to Celsius; a single line edit (SLE) field to enter the source temperature; a text field to display the converted temperature; and a pushbutton to execute the conversion.

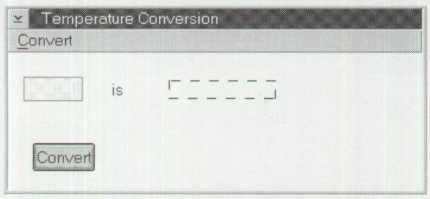


Figure 1. Temperature Conversion Application

As you build this application, you will see that Visualizer Development, developed with the OS/2 environment in mind, can capitalize on OS/2's power. You build new objects (menus, windows, programs, etc.) by copying templates from the OS/2 Templates folder.

Visualizer's development process starts by constructing the interface, then building a program to react to it. Each menu item (each word on the menu bar) is a separate menu object; in this example, you have only one. The menus are placed on windows along with any controls (pushbuttons, text, drop-down lists, etc.). Then a program is built from the resulting window(s). Finally, the menus, windows, and programs are compiled into an application.

So let's get started!

Build the Menu

You will build the menu first. From the OS/2 Templates folder, drag a Visualizer Menu template to a folder you've created to hold this application. Then rename it to `ConvMnu` by using Alt+mouse button 1 (Figure 2).

After opening `ConvMnu`, you will see the menu editor (Figure 3). This editor allows you to create the menu bar entry (the text on the menu bar) and the menu entries under it. Like each of the Visualizer Development objects, all of the menu editor functions are available via the drop-down menus from the menu bar. Some of the more frequently used functions are also on the Toolbar (the icon buttons below the menu bar) for easy access. And some of the items on the window can be directly accessed by double-clicking on them.

Set the text for the menu bar entry by double-clicking on the "Menu bar entry: (N)ew" area on the menu editor window.

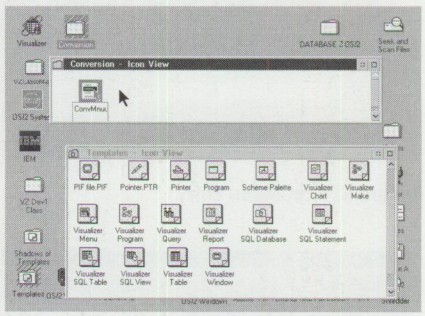


Figure 2. Dragging a Menu

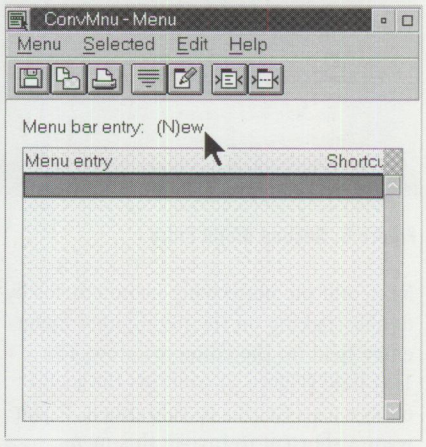


Figure 3. Menu Editor

In the Menu Bar Entry window, you define the text on the menu bar (Figure 4). The name in the Text field is the label on the menu. The Mnemonic field is a short-cut character for selecting the menu from the keyboard. The Reference field is the object name. If you want to associate help with this menu, you can enter a Res id number.

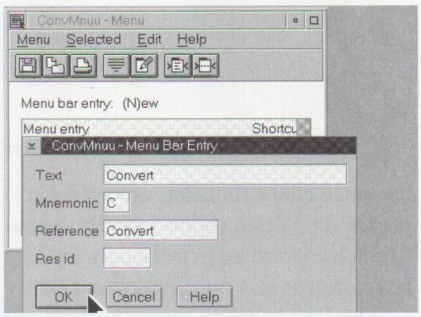


Figure 4. Menu Text

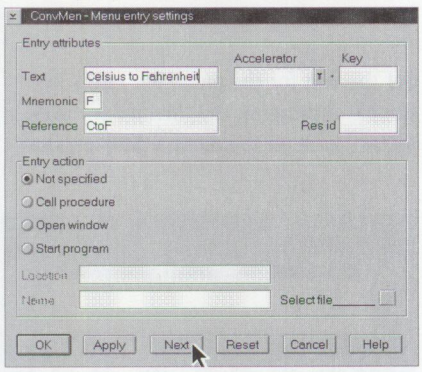


Figure 5. First Menu Entry

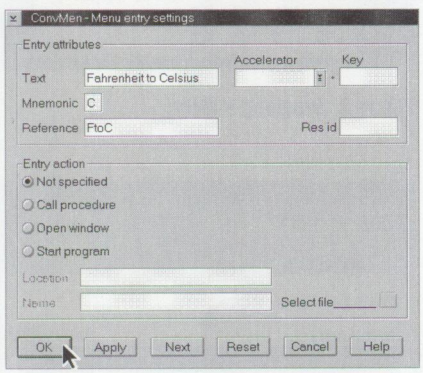


Figure 6. Second Menu Entry

Next, add the underlying menu entries by double-clicking on the highlighted menu entry field on the editor window. The Text field (Figure 5) contains the menu entry's text: Celsius to Fahrenheit. If you want a shortcut to invoke this menu entry, then you can specify a two-key combination in the Accelerator and Key fields. The Mnemonic, Reference, and Res id fields have the same meanings as those on the Menu Bar Entry window.

Notice that you can identify procedures, windows, or programs to be invoked when this menu entry is selected; however, select the default this time.

After pressing the Next button, enter the second menu entry (Figure 6) to convert Fahrenheit to Celsius.

After clicking on the OK button to return to the editor, click on the Save toolbar icon and close the editor. You've just created the menu.

Build the Window

Next, you will complete your user interface by building the window. Just as you did for your menu, drag the Visualizer Window template from the OS/2 Templates folder to your folder. Name the window `ConvWin` and open the object.

As you can see in Figure 7, the window editor has two windows: the main window and a template window. The main window is used to select controls and

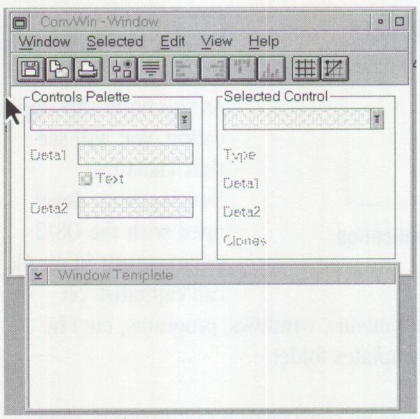


Figure 7. Window Editor

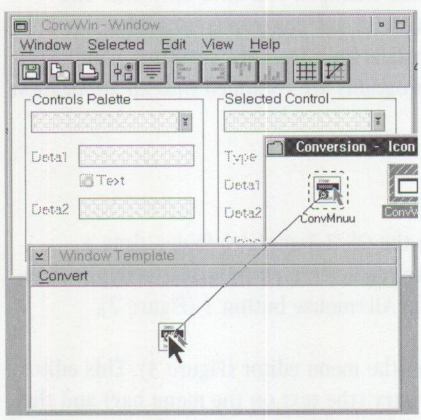


Figure 8. Adding a Menu to the Window

their attributes, while the results are shown on the template window.

Place the menu on the window first by dragging `ConvMnu` from the folder onto either editor window (Figure 8).

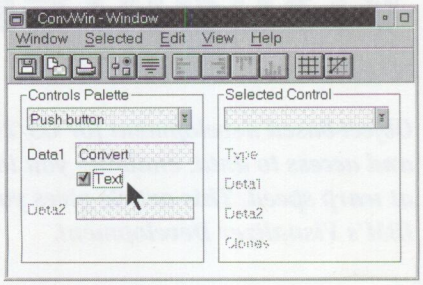


Figure 9. Adding a Pushbutton

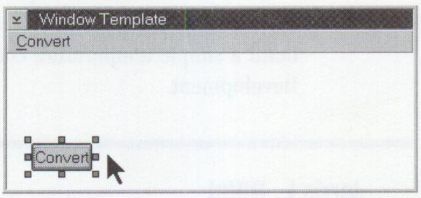


Figure 10. Pushbutton Handles

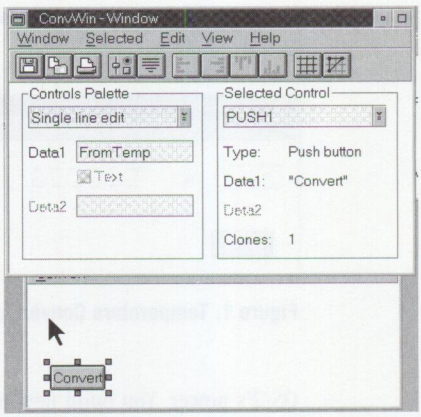


Figure 11. Adding an SLE

Begin adding the controls by selecting a pushbutton control from the Controls Palette on the main window (Figure 9). By entering "Convert" (without the quote marks) in the Data1 field and checking the Text box, you indicate that you want to place the text "Convert" on the button. To place it on the window, use the mouse to select the position on the template window. To reposition it, grab it with mouse button 2 and drag it around the window. To resize it, grab one of the eight handles that surround it and drag the border (Figure 10).

Create a field to input the source temperature by selecting "Single line edit" from the Controls Palette and entering

"FromTemp" in the Data1 field (Figure 11). Now, values entered in the SLE can be found in the variable `FromTemp`. Place the SLE on the template with the mouse, just as you placed the pushbutton.

Place the text "is" on the window by using a Text control from the Controls Palette, entering "is" in the Data1 field, checking the Text box, and placing it on the template with the mouse (Figure 12).

To display the converted temperature, select a Text control from the Controls Palette again (Figure 13). In the Data1 field, enter "ToTemp." By not checking the Text box, you tell the editor that `ToTemp` is a variable and you want its contents displayed. Then place it on the template window with the mouse.

To give your user a visual cue of where the output temperature will appear, add an outline attribute to the field by selecting "Selected" from the main window menu bar, then "Outline." After selecting the Line Style (Figure 14), click on OK.

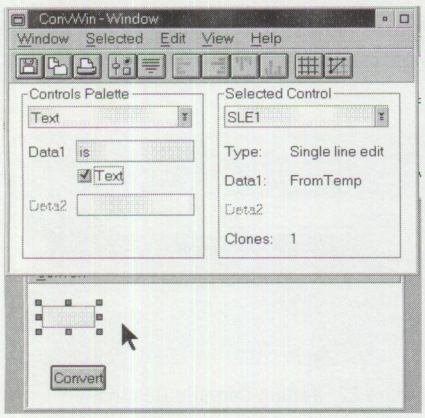


Figure 12. Adding Static Text

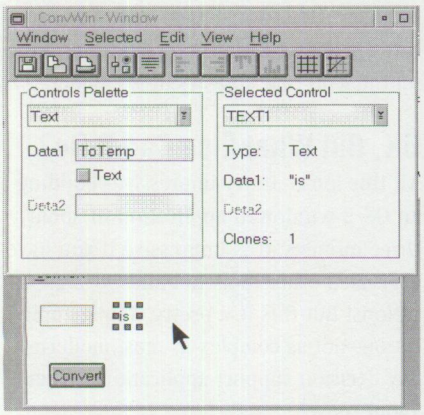


Figure 13. Adding Variable Text

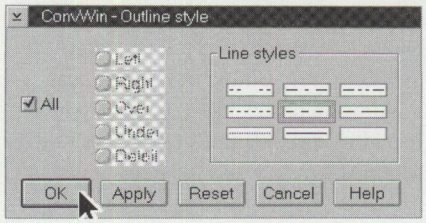


Figure 14. Outlining Variable Text

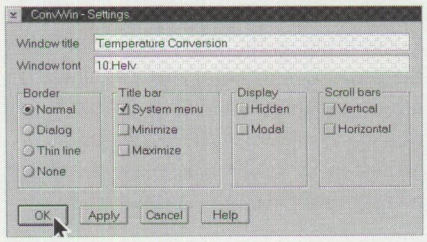


Figure 15. Window Settings

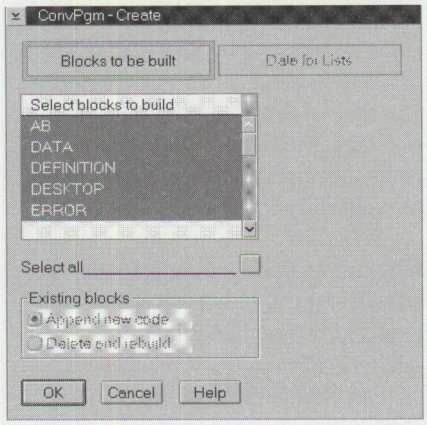


Figure 16. Event Selection

To change the window title (Figure 15), select the Settings toolbar button (fourth button from the left in Figure 13), change the window title to "Temperature Conversion," then click on OK.

The window is complete. Save it with the Save icon on the toolbar and close the editor.

Build the Program

Next, you need a program to drive the user interface. Like you did for the menu and window, drag a Visualizer Program template into your folder, rename it to `ConvPgm`, and open it. You could start building the logic to run the interface, but it is simpler to let the editor create much of the code. Drag `ConvWin` from your folder and drop it on the open program editor. After the editor examines the window, you see the window in Figure 16 that lets you select the event blocks you want built.

Visualizer programs are block-structured and event-driven. This means that for each event you want to handle, you create a block of code. If you do not have a block for an event that is invoked, then nothing happens. *Block structuring* means that each event is coded with the following structure:

```
ON eventname
DO
        ...
END
```

In the `START` event, you will typically initialize any variables and open a window so the user can interact with the application. In the `SELECT` event, you will determine which window control the user selects, then react to that selection.

In this example, you click on OK to accept the default event blocks that the program editor has selected. Now you see that the editor has listed the event blocks on the left side and listed the procedure blocks on the right (Figure 17). You can open each block by double-clicking on its name.

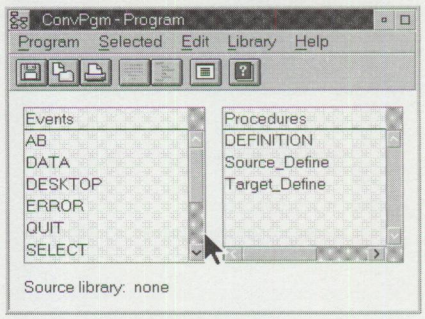


Figure 17. Program Editor

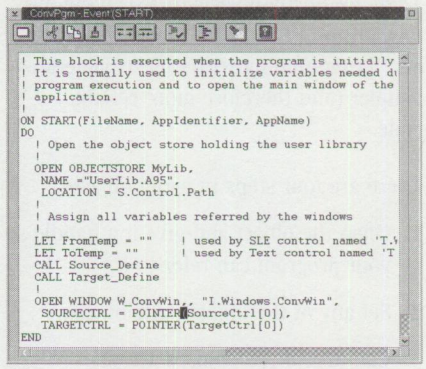


Figure 18. START Event Block

In the `START` block (Figure 18), notice that the editor has written skeleton code and comments that tell you what is being done. Also notice that it has initialized

the variables identified on the window: (FromTemp and ToTemp).

Object-Based Development Language

The statement in Figure 19 demonstrates the object-based nature of Visualizer's Development language, ASL.

In Figure 19, you open an object with a class of WINDOW, giving it a handle of My_Window for the program to use and setting some of its attributes (width = 300, height = 200) and the colors of the foreground and background (red and white).

```
OPEN WINDOW MY_Window,
     SizeX=300, SizeY=200,
     FGColor="Red",
     BGColor="White"
```

Figure 19. Code with an Object-Based Nature

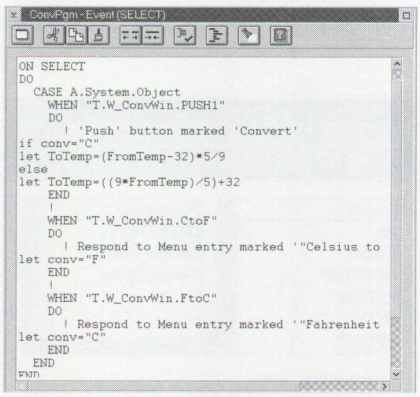


Figure 20. SELECT Event Block

There are many objects in ASL: windows, pushbuttons, tables, databases, printers, etc. Programming with objects generates simpler (and therefore more reliable) code.

There are four steps in using objects:

1. Open the object and give it a handle so your program can refer to it.
2. Set any of the object's attributes.
3. Call one of the object's actions.
4. Shut the object when you're finished.

Each object has its own attributes and actions. For example, a window has attributes including width, height, color, border type, title, and others, for a total of 42 attributes. The window also has eight actions including refresh, minimize, and maximize. On the other hand, the multimedia object, MMFILE, has attributes including VOLUME and VISIBLE as well as actions like PLAY and RECORD.

One of the real treasures in the editor is its link to the online reference manual. If you forget some syntax or want to search for a function, you simply press the F1 key to bring up the reference manual. If the cursor is on a word that can be recognized, you are taken directly to that section. So if you forget the attributes or actions for a window, just place the cursor on the word "window" and press F1.

Back to the example. You really do not have to make any changes to the START block. However, you do need to include logic to handle selection of the menu entries or the pushbutton. That will be handled in the SELECT block (Figure 20). Close the START block with the system icon (upper left of the START block window), then open the SELECT block from the editor window.

In Figure 20, you again see that the overall code structure has been created and commented. Also notice that each selectable item on the window has a DO-END block created with a comment identifying which control the block handles. In this application, you add code to each menu selection, setting a variable Conv to the type of conversion selected. You also add an IF-THEN-ELSE set of statements to the pushbutton block to perform the correct calculation and load the results into ToTemp.

You can save time by entering each new line at the left margin. You can then use the Check Syntax and Format buttons at the top of the window to check the syntax and to indent and stylize the code.

That's it! The program is complete. After closing the SELECT block, save the program with the Save button on the toolbar and close the program editor.

Complete the Application

Only one more step completes this application. Drag a Visualizer Make template from the OS/2 Templates folder to your program folder, rename it to ConvMak, and open it.

The Make object is much like a "bill of materials" for the application. It contains the list of objects to be compiled and identifies where the compiled application will be placed. To build this list, drag ConvWin from the folder to the open ConvMak object. Notice that ConvMak recognized that the menu, ConvMnu, was included on the window, and therefore included it in the list. When you drag your program onto Make, the list is complete (Figure 21).

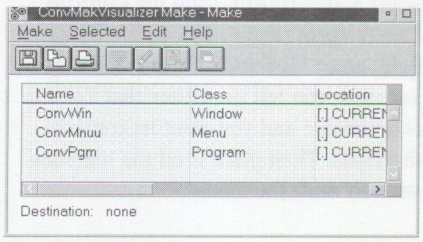


Figure 21. Make Facility

Now, identify the location of the compiled application by first selecting the Make menu, then the Set Compile Destination entry. The location defaults to the folder.

Enter "ConvApl" as the name, and click on OK (Figure 22).

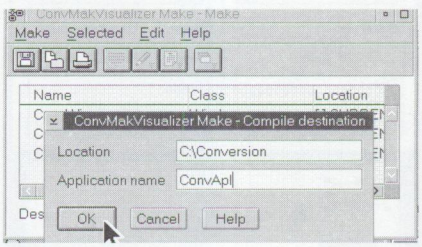


Figure 22. Setting Compile Destination

The last step is to press the Compile button (fourth button from the left) on the toolbar. After compilation completes, you can run the application by double-clicking on the icon that appears in the folder.

OK, But What Else Can It Do?

All this looks pretty impressive–building an OS/2 windowed application in about three minutes. It is impressive if you have ever used other tools to build OS/2 applications! But this is a pretty simple application–not as complex as most modern-day decision support applications. What about those?

Visualizer Development is more than able to handle complex applications. With its strong data access capability and broad range of function, this product should be in most enterprises' toolboxes.

Let's take a look at some of Visualizer Development's other functions and features.

Access to Data

As part of the Visualizer family of products, Visualizer Development was designed with data access in mind. The strongest interface is with IBM's premier relational database family, DB2. When DB2 for OS/2 or Client Application Enabler for OS/2 (CAE/2) is teamed with Visualizer Development, the DB2 databases throughout your enterprise are easily accessible.

How easily? The example in Figure 23 demonstrates opening a database called BUDGET and selecting all rows from the SALES table. The first statement opens a Visualizer Table to receive the results from DB2. The attributes define where you want the table created. The second statement opens the connection to DB2. The attributes identify the database you want and where it is cataloged (the system catalog). Because the DB2 family has Distributed Relational Database Architecture (DRDA), the physical database can be located anywhere in the network–it can be on your workstation, on a LAN server, or on a remote host under MVS, VM, OS/400, or AIX–even on another continent!

The third statement opens a data transmission object that uses a DB2 connection object, "MySess." The attributes identify that you will be reading from the database and placing the results in the table you opened earlier (MyTab).

The fourth statement provides the SQL Select statement to the transmission object. The final statement executes the data retrieval.

The data access options are too numerous to discuss here. But you can read and write to the IBM DB2 family and many PC interchange formats. You can read from other relational databases via the Q+E interface library. In fact, the Q+E drivers for Sybase and Oracle are included with the product!

```
OPEN TABLE MyTab,                !Table for results
  name="mytab.tab",              !OS/2 filename
  location="c:\",                !OS/2 location
  mode="write"                   !Write vs Read

OPEN SQLSESSION MySess,          !SQL Object
  DBNAME="BUDGET",               !Database Name
  DRIVE="0"                      !Use DB2 catalog

OPEN SQLTRANS MyTrans, MySess,!SQL Data Object
  TABLE=MyTab,                   !Use MyTab for results
  MODE="R"                       !Read vs Load

!Set the SQL Select Expression
CALL Trans'Setexpr("SELECT * FROM SALES")
CALL Trans'Snapshot()            !Retrieve results
```

Figure 23. Retrieving Relational Data

More Features

Some of Visualizer's other features that add real power include the ability to:

- Perform arithmetic, statistical, trigonometric, precision, character string manipulation, and date/time/duration functions.
- Display bitmaps, metafiles, graphics.
- Support multimedia audio and video.
- Exchange data with other applications via:
  - PC interchange formatted files (read and write).
  - OS/2 clipboard (cut, copy, paste).
  - Dynamic data exchange (DDE) so your application can be a data client or server to other applications and products.
  - Direct manipulation (dragging/dropping icons) so you can react to icons dropped on your application or provide an icon that can be dropped on other objects (products and applications).
- Use high level language application programming interface (HLLAPI) to interact with a host using IBM's Communication Manager. You can upload/download data, send keystrokes, and examine host screens for information.
- Call and be called by C programs.
- Call REXX programs.
- Send objects/files to other applications using a mail system like cc:Mail.
- Set "timers" to automate actions.
- Call various OS/2 services, including starting other OS/2, DOS, or Windows programs.

Visualizer Family of Products Integration

One of Visualizer's more powerful features is integration with other Visualizer products. There are application programming interfaces (APIs) to the Visualizer Chart, Report, SQLStatement, SQLTable, and Table objects delivered by other Visualizer family products. You can include a chart that someone else customized, or your program can dynamically create one by using the APIs. This integration can be a real time saver if your application must deliver a chart, report, etc. Obviously it is easier to use an API rather than code one yourself.

Technical Information

Visualizer Development, like Visualizer Query for OS/2, can be installed either stand-alone or in a LAN server/requester environment. The information below assumes a stand-alone environment.

Visualizer Development hardware requirements are:

- 80386 processor (80486 or better is recommended)
- VGA display (SVGA or XGA is recommended)
- 16 MB of memory, minimum
- Mouse or pointing device

Disk space recommendations are:

- Query for OS/2: 15 MB to hold the product. At least 20 MB should be available for OS/2 swapper expansion. Installation requires 60 KB on the boot drive.
- Development: 3 MB to hold the product. At least 20 MB should be available for OS/2 swapper expansion. Installation requires 60 KB on the boot drive.

Software requirements are:

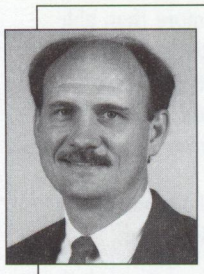
- OS/2 2.1 or higher
- Visualizer Query for OS/2 (program number 5622-118)

Optional:

- IBM DATABASE 2 for OS/2 or IBM Client Application Enabler for OS/2 (program numbers 5622-044 and 5622-129, respectively)

Easy Decision Support Applications

As you have seen, Visualizer Development is an object-based, event-driven, block-structured tool that allows you to build customized decision support applications for the OS/2 environment. It was developed with data access as a central theme and contains the power to get the job done. It has ease of use built in—and the IBM Corporation to back it up!



Jerris (Jerry) L. Riffel educates and consults with IBM customers on decision support topics. He is in IBM's National Systems Center in Dallas, Texas. He is IBM's representative for decision support to the IBM user group GUIDE. Jerry has been involved with the introductions of Visualizer, Personal AS, and Data-Guide/2. In his 20+ years with IBM, he has been a programmer and team leader in Lexington, Kentucky; leader of the development team that brought the IBM Application System (AS) to the IBM Information Network (IN); and manager of an IN business systems analysis group in Tampa, Florida. Jerry joined IBM in 1974, has a BS degree in Computer Science from Kansas State University, and has completed some graduate work at the University of Kentucky.

Performance: DCE RPC as a DB2 for OS/2 and DB2 for AIX Transport

Behind many business applications is a good Relational Database Manager (RDBM). The question I am most often asked is "How does DCE perform as an RDBM transport?"

This article looks at three methods of integrating DB2 for OS/2 and DB2 for AIX into a Distributed Computing Environment (DCE) application and discusses the strengths, weaknesses, and the DB2 for OS/2 performance of each implementation.

Bob Russell
IBM Corporation
Austin, Texas

Many business applications are inherently data-driven. Being able to secure and rapidly access this data can be critical. Developing applications on DCE assumes the choice of DCE security, location transparency, and transport features. The performance of DCE Remote Procedure Call (RPC) as a Relational Database Manager (RDBM) vehicle is therefore interesting in this context.

RDBMs employ several security and transport methodologies. Some of these services are duplicated when both DCE and an RDBM are installed on the clients. Most RDBMs require database manager software to be installed and active on all clients; DB2 for OS/2 requires about 7 MB of client disk space and 0.5 MB (idle) to 1.5 MB (active) of client memory.

DB2 for OS/2 database clients currently share the common User Profile Managment (UPM) security logon facility with LAN Server. This requires clients to log on to both DB2 for OS/2 UPM and DCE Security Services. The IBM LAN Server Enterprise (LSE) product being developed will combine LAN Server 4.0 with DCE and will share the DCE Security and Cell Directory Services. Although LSE will use DCE security, DB2 for OS/2 will initially require the additional UPM security service.

DCE provides five levels of RPC authentication to protect data across the network: Connect (login), Call, Packet, Packet Integrity (CRC), and Packet Privacy (encryption). The RPC authentication level for these tests is set at Connect level; the user's DCE security credentials are verified by DCE Security when the client binds (connects) to the DCE RPC application server. Connect is consistent with the security provided by DB2 for OS/2-LAN Server User Profile Management.

When developing a DCE application, it is desirable to protect the RDBM's data access and integrity with DCE Security, as well as provide the location transparency of DCE Cell Directory Services (CDS). Advantages to using DCE RPC rather than the native RDBM transport include:

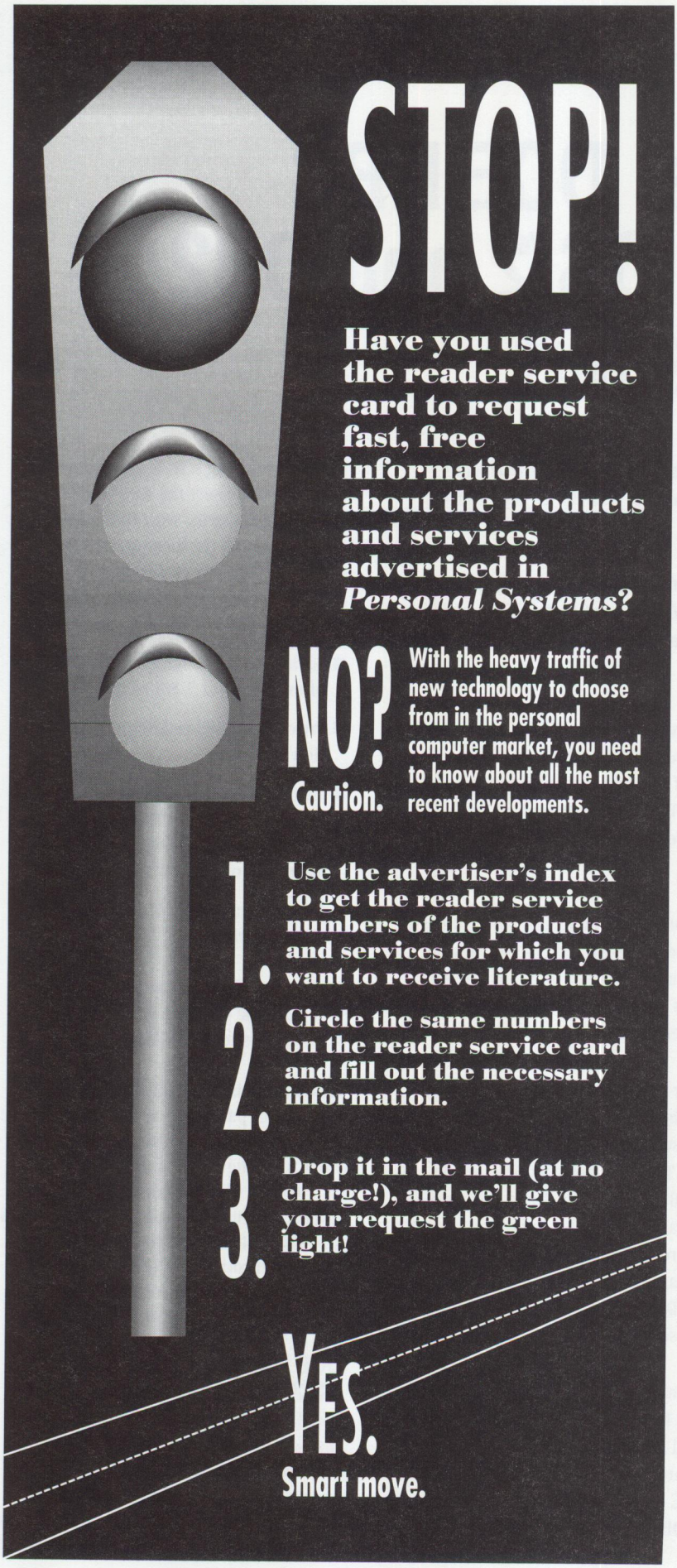
- Saving client resources: 0.5 to 1.5 MB of memory and 7 MB of disk.
- Providing a higher level of communication security.
- Using the location transparency of CDS rather than the machine-specific mapping of database server network addresses of DB2 for OS/2 servers.
- Providing a more consistent programming interface.

DCE's cross-platform features can also eliminate the need for DB2 for OS/2 gateway services to access DB2 for MVS.

Most industrial-strength RDBMs provide a transaction packaging or stored procedure mechanism to process an atomic unit of work in a single call to the database server. The DCE RPC provides a DCE-based transaction model that is consistent across all DCE platforms. Accessing the database manager through DCE RPC allows the application server to communicate with the database manager as a local process.

DCE Threads Versus DB2 for OS/2 Processes

DCE is based on a multi-thread rather than a multi-process design. Clients bind (connect) to the DCE RPC application server process. Each RPC call connects temporarily to one of the RPC call threads allocated by the RPC runtime on behalf of



Personal Systems

FREE

Product Information Index

Reader Service Number	Company	Page #
10	AccuSoft	9
16	Cawthon Software	3
7	Clear & Simple, Inc.	8
6	Client Server Networking	7
19	Hilbert Computing	9
13	IDG Books	11
20	Indelible Blue, Inc.	11
1	International Software Solutions	6
12	John Wiley & Sons	11
17	Kovsky Conference Production	7
15	Lieberman and Associates	1
9	Lotus Development Corporation	9
8	microMEDIA Imaging Systems, Inc.	8
4	Microway	6
2	MultiQuest Corporation	6
18	On-Line Data	9
22	Onspec Automation Solutions	Cover 3
5	Pinnacle Technology	15
21	Piper Research, Inc.	24
24	Prominaire	Cover 2
11	SAMS Publishing	10
23	Softouch Systems	Cover 4
14	ViaGrafix	11
3	4th Dimension Software Ltd.	6

the application server. RPC runtime creates a single queue for the application. The queue size is eight requests multiplied by the number of RPC call threads defined by the application.

RDBMs for OS/2 come in two varieties. Some are thread-based, while others, like DB2 for OS/2 V1.2, are process-based; the client connects to a unique DB2 for OS/2 V1.2 server agent process.

Since implementing a thread-based RDBM is somewhat straightforward as a DCE RPC application, I do not dwell on thread-based RDBMs in this article. The focus of this study is on DB2 for OS/2 and DB2 for AIX, which are both process-based RDBMs. The performance results shown in Figure 2 reflect only DB2 for OS/2 V1.2.

The OLTP Test Application

This study's test application is loosely based on an online transaction processing (OLTP) banking transaction benchmark. Each transaction processes three Structured Query Language (SQL) updates and one SQL insert within the scope of an SQL commit/rollback atomic transaction. The focus of this study is on the performance of DCE RPC rather than on DB2 for OS/2; therefore, the size of the OLTP database is scaled to be rather small.

Disclaimer: This OLTP benchmark differs from the database industry-standard TPC Benchmark A in three significant ways: (1) the ratio of the throughput to the database size and number of clients is not consistent with the TPC Benchmark A rules; (2) the DB2 for OS/2 forward recovery logging is not enabled, and (3) the think time between transactions is set to zero. The results shown in Figure 2 are not, nor do we claim that they are, representative of DB2 for OS/2 behavior in a properly structured and approved TPC Benchmark A test environment.

Implementation Models and Performance Studies

I have chosen three models of DB2 for OS/2 as a database manager for the DCE OLTP application (Figure 1, lines B, C, and D). The throughput shown in Figure 2 for B, C, and D reflects the connection-oriented transmission control protocol/internet protocol (TCP/IP) transmission control protocol (TCP) as the transport protocol. There is zero think time between client

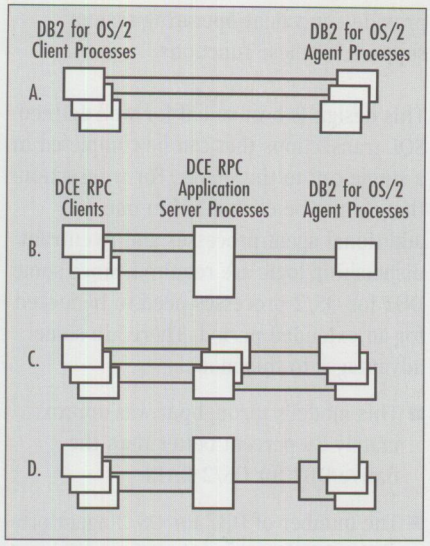


Figure 1. Models

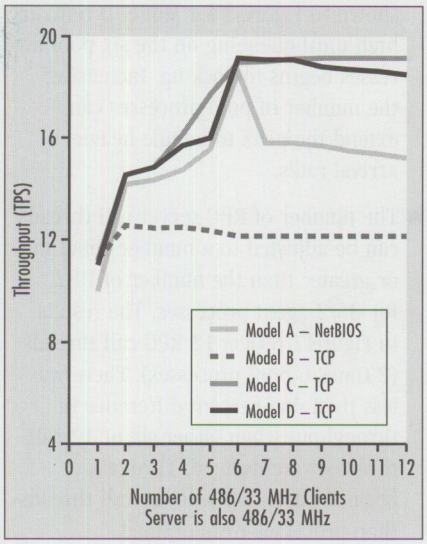


Figure 2. OLTP Performance

RPC calls. Therefore, the actual arrival rate at the server is much higher than the number of clients indicated in Figure 2.

Model A

Model A, shown on the first line in Figure 1, is the native Database Application Remote Interface (DARI). In Figure 2, Model A's performance is used as the baseline for performance comparisons for Models B, C, and D.

DB2 for OS/2 V1.2 creates a new database agent process on the server when the DB2 for OS/2 client opens a database session.

Model B

Model B, on the second line in Figure 1, is a single DCE RPC server process plus single DB2 for OS/2 database process model. Although DB2 for OS/2 V1.2 is a multi-process RDBM, it can also be accessed as a single process by a multi-thread DCE RPC application server. DB2 for OS/2 will serialize individual SQL statements. With some additional synchronization using OS/2 semaphores or DCE thread locks, atomic transactions can be serialized.

Two variations of Model B were tried:

- The first variation, and poorest performer (not shown in Figure 2), was implemented with multiple RPC call threads. This required using semaphores to enforce the integrity of atomic transactions. The OLTP throughput for this variation of Model B was less than half (45 percent) of the native DB2 for OS/2 DARI (Model A) throughput.
- The second variation–with only one RPC call thread–performed much better. This pushes the serialization back to the RPC queueing level. There were two clear advantages over the first variation of Model B: (1) the throughput was much better than the first variation and was only 20 percent slower than the native DB2 for OS/2 DARI, and (2) there was no need to use semaphores to protect atomic transactions.

The second variation of Model B was much easier to implement than any of the other methods discussed in this article. If absolute database performance is not critical, you might consider the second variation.

There are some disadvantages to both single-process variations of Model B:

- There is a lack of concurrency and overlap in both the DB2 for OS/2 database process and in the RPC server application by single-threading the server side of the application. The loss of efficiency can be minimized by not loading the RPC server manager routine (stub) with non-essential work.
- In large, busy systems, there will be a high incidence of RPC retry activity. RPC runtime provides queueing for only eight requests per RPC call thread, since only one RPC call thread was defined by the application server. More than eight queued requests will fail and will require a retry by the client side of the application.

Model C

Model C, on the third line in Figure 1, is a multiple RPC server process plus multiple DB2 for OS/2 process model. This model uses an additional instance of the DCE RPC server application spawned in an OS/2 child process for every client connected to the application. Each instance has only one RPC call thread, and each client has a dedicated application server process and a re-entrant DB2 for OS/2 process.

Model C might be best suited for *ad hoc* database activity and long-running atomic transactions. This model closely mimics the native DB2 for OS/2 model for multiple remote clients.

The peak throughput for this model is about 25 percent better than the native DB2 for OS/2 DARI. This model can provide the highest level of sustained performance.

Model C may eventually cause an out-of-memory condition in a heavily loaded system. The memory working set for each DB2 for OS/2 database process is about 400 KB. For 100 clients, this comes to 40 MB.

Model D

Model D, on the fourth line in Figure 1, is a single RPC server plus DB2 for OS/2 process pool model. This model of the DCE RPC server application spawns a fixed number of database agent processes when the application server is started. The RPC server application manager routine (stub) dispatches the requests to the first available database pool process.

Model D's maximum throughput with six pool processes was approximately 25 percent better than the native DB2 for OS/2 DARI. The optimum number of pool processes was between six and ten.

A very simple dispatching algorithm was used for this model's OLTP implementation. A simple Boolean array, a couple of semaphores, and some named-shared memory worked very well. OS/2 and AIX provide equivalent operating system support for these functions.

This design is best suited for well-defined SQL transactions that can be completed in a single call to the server. For transactions that cannot be committed in one call, additional agent processes and re-entrant dispatching logic are required, since some DB2 for OS/2 processes need to be locked for an extended period. There are some advantages to this model:

- This model's throughput was approximately 25 percent better than the native DB2 for OS/2 DARI.
- The number of DB2 for OS/2 agent processes can be adjusted to the peak performance level to minimize DB2 for OS/2 queuing. The peak throughput shown in Figure 2 for Model D remains high until queueing on the six pool processes begins to back up. Increasing the number of pool processes can extend the peak to handle heavier arrival rates.
- The number of RPC server call threads can be adjusted to a number equal to or greater than the number of DB2 for OS/2 agent processes. The results in Figure 2 reflect 12 RPC call threads (2 times 6 pool processes). There was less than one percent difference in throughput when either six or 12 call threads were defined. Throughput began to drop when fewer call threads than processes were defined.

  A higher number of call threads could increase the request queue size allocation by DCE runtime. We saw in Model B that queueing in the RPC runtime request queue was well tolerated.

  Having too many call threads may degrade performance because of excessive thread switching. A good starting point for a number of call threads is between one and two times the number of pool processes.
- DB2 for OS/2's memory working-set is fixed at a low level regardless of the number of DCE clients on the system.

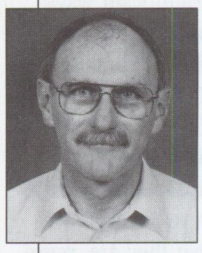
Models for Different Needs

Each of these models provides acceptable application performance. Each is best suited to a different application design and SQL transaction profile.

Model B uses a single-thread design and is suited for short, single-statement SQL requests. Performance may not be acceptable for complex, multi-statement transactions. This model's performance is the least impressive of the three models but requires the fewest memory resources and is the easiest to implement. This design is adequate for casual database access.

Model C uses multiple RPC server processes and is suitable for *ad hoc* queries and transactions requiring user intervention before committing the transaction. This model provides good sustained performance but may eventually pose a memory concern. With a little clever programming to manage RPC server child processes, this design can handle any SQL transaction profile.

Model D uses a process pool and is suitable for any transaction that can be completed in a single call to the application server. This model might be well suited to large systems where both performance and server memory resources are important.



Bob Russell is an advisory programmer and team leader of the Heterogeneous Environment Team in PSP Systems Performance, Austin, Texas. Since joining IBM in 1963, Bob's assignments have included Electric Typewriter Division customer engineer, Office Products Division administrator, systems analyst for the Installed Machine Inventory System, and performance analyst for DisplayWrite, OS/2 Database Manager, DCE for OS/2 and AIX, and LAN-based client/server products.

Remote Program Load of OS/2 Warp from NetWare 3.12

As more users demand the capabilities and performance of OS/2 Warp on their computers, LAN administrators will face new challenges in meeting those demands. This article explains how to solve one of those challenges: providing OS/2 Warp to users of diskless workstations, specifically in a NetWare environment.

Many of today's DOS/Windows environments are ready to move to the next level of computing: OS/2 Warp! However, making the move is not always as simple as booting the installation diskette and installing OS/2 Warp on top of your hard disk's existing DOS/Windows.

Many computing environments use diskless workstations because of their advanced security and easy support. If you have diskless workstations, then you are probably accessing your data, applications, and printers with Novell's NetWare. If this sounds like your environment, then hope is on the way—you, too, can get Warped!

Steven H. Bumgarner
IBM Corporation
Roanoke, Texas

Jay L. Curtiss
IBM Corporation
Milwaukee, Wisconsin

In this two-section article, we explain the procedures necessary to set up Remote Program Load (RPL) OS/2 Warp from a NetWare 3.12 file server. The first section discusses design issues, restrictions, and implementation recommendations. The second section defines the steps for setting up the entire RPL environment. This section is based on the text from the NetWare Client for OS/2 2.1 online book and makes changes, corrections, and additions to that text. The result is a document that you can use to set up a pilot of the OS/2 RPL environment at your location. Since you are probably familiar with normal administrative tasks (such as how to add users), we have abbreviated some of the instructions to include only the relevant information. If you are uncomfortable performing routine administration on your server, then we suggest you enlist the services of a Certified NetWare Engineer (CNE).

In our investigations, we installed Windows-based and OS/2-based applications and created a test environment consisting of a NetWare 3.12 file server and 10 IBM PS/2 systems RPLing OS/2 Warp from the file server. We used the NetWare Requester for OS/2 2.11, which also supports OS/2 Warp. We followed the RPL process outlined in the NetWare Client for OS/2 2.1 online book and discovered several additions and corrections that make the OS/2 RPL process easier and more manageable.

Design Considerations and Recommendations

We, as well as Novell, recommend that you first pilot OS/2 Warp RPL on a separate file server dedicated to the RPL process. In addition, you should place future sets of RPL clients and their associated RPL server on separate physical networks. There are several reasons for this:

- *Performance.* We tested a simultaneous boot of ten 486 machines via a 16 Mbps token-ring network, and the server was able to handle the load and boot all the machines in less than two minutes.

  We did not have the facilities to test the effect of large swap files on server performance. Once you start running communication programs such as Communications Manager/2 or transmission control protocol/internet protocol (TCP/IP) for OS/2, your swap file will increase, as it will with memory-intensive programs such as Lotus 1-2-3 and WordPerfect for Windows.

  If you were to combine the RPL server functions with your existing production data and application server, the overall performance would probably not be acceptable to your user community.

- *RPL Workstation Management.* With an initial pilot of about 20 RPL clients, one RPL server should be adequate. You will probably not want to move beyond 50 RPL clients per RPL server, and through experience, you may find that the optimum number is less. Eventually, you will need additional RPL servers to handle the RPL workload.

- *RPL Server Management.* Once you have more than one RPL server on the same physical network, Novell recommends that you replicate all of the RPL files on all of the RPL servers. The reason for this is the way that the RPL servers handle the clients' requests. If a client usually connects to one RPL server, and it happens to be busy, the

RPL request will find its way to the next server that is handling RPL requests. If that server doesn't have the client workstation's address defined, the connection is made but the request is not serviced, and the client has to reboot and try again. *Note:* RPL requests are ignored by non-RPL servers not running `RPL.NLM`.

This may seem like a simple solution, but it introduces a new problem: If all of the RPL files, including the OS/2 desktop files, are on all of the RPL servers, then the desktop files may get out of synchronization, depending on which server you connect to. For example, on Monday you connect as usual to RPL1, but today (Tuesday) it is busy, so you connect to RPL2. While connected to RPL2, you decide to change the background of your desktop to the `PINES.BMP`. Tomorrow (Wednesday), you connect to RPL1 again, and your desktop background reverts to the one you had on Monday—no pine trees! Although this example is trivial, you can see the multitude of problems that can arise from this situation.

We think a better solution is to put each set of RPL clients and their server on a separate physical network. Many Ethernet hubs used today allow the network to be segmented simply through its configuration software. If you have a Token-Ring environment, you can accomplish segmentation by using bridges. If you use a bridge, then by using a bridge filter, you can filter out frames with the RPL Service Access Point (SAP) and keep each RPL group isolated and manageable. FILTER4, available with the IBM Token-Ring Bridge Program, allows you to filter specific SAPs. The SAP for RPL frames is x'F8'.

- *Security and Data Management.* OS/2 RPL's design requires that all necessary files be placed on the `SYS:` volume. This includes not only the OS/2 system files, but all of the files required for each individual workstation. Each user requires at least read, write, create, erase, modify, and file-scan rights to his or her respective RPL home directories. This can lead to integrity problems on the `SYS:` volume caused, for example, by someone filling up the available space. Additionally, if you assume a requirement of 10 MB per RPL workstation for swap, spool, and control files, then 50 users will boost the capacity requirements for just RPL clients to over 500 MB. On a dedicated RPL server, that size is acceptable, but on a data or application server, it is best to have a smaller `SYS:` volume dedicated to system files and public utilities.

Adding an RPL server to your network introduces a single point of failure that can affect a number of users at once. By splitting up users among multiple servers, you can lessen the effect of an RPL server outage.

Because the RPL server is a single point of failure, you may want to consider adding redundancy or backup capabilities. This can be accomplished by disk mirroring or duplexing or by having a backup server that can easily be swapped with a server that is down.

From a management perspective, you will want to consider standardizing your RPL workstation hardware as much as possible. Having identical video and network card configurations will greatly simplify RPL administration. When you introduce additional video boards and drivers, you must reinstall OS/2 on a workstation with a hard drive and either save a new image (not recommended) or determine which files and `CONFIG.SYS` changes are required, then add them to the image on the RPL server. This can cause an administrative nightmare if you use a lot of different video drivers.

Another point to consider is the physical network structure. You should have a superior network infrastructure in place, including all cabling and network hardware. Since the network is the workstation's only link to its `C:` drive, it is vital that the network link be as reliable as possible.

One last point to consider is which version of OS/2 Warp will be installed: OS2 Warp or OS/2 Warp with WIN-OS2. With OS/2 Warp, certain Windows programs that are installed on the RPL image `C:` drive will not work properly. These include Write, Cardfile, and other programs that expect to create temporary files on the drive from which they run. Also, because the RPL process in the current NetWare Requester for OS/2 was designed for OS/2 2.1 (which came with WIN-OS2 support), there were no provisions made for sharing programs in a `WINDOWS` subdirectory. However, with the release of OS/2 Warp with WIN-OS2, the sharing problem has been corrected.

General Prerequisites

You will need three computers to install and enable an RPL workstation. First, you need a computer with a hard disk, a network card, OS/2 Warp, and NetWare Requester for OS/2 installed. This machine, referred to as the *client image workstation*, must be able to access the server. The second computer, the *RPL workstation*, needs only a network card that can access the server. The third computer, the *RPL server*, is where the RPL workstation will attach and load the OS/2 RPL image (referred to as the RPL server). Each of these three computers has its own set of prerequisites, detailed below.

Prerequisites for the Client Image Workstation

The following steps create an OS/2 client image that you can copy to the file server and use as the RPL image:

1. Start with a clean primary partition of at least 75 MB for drive `C:`.
2. Install DOS, formatting drive `C:` as FAT. We used PC DOS 6.3.
3. Install Windows 3.1 (or 3.11 or Windows for Workgroups if you prefer).
4. Install OS/2 Warp (we used express install).
5. In the `CONFIG.SYS` file, remark the following line: `IFS=C:\OS2\HPFS.IFS /CACHE:64 /CRECL:4`
6. Install the OS/2 NetWare Requester for OS/2 2.11. We recommend you get the latest diskette images from Novell. (Diskettes can be downloaded from CompuServe or from the Novell FTP server at `ftp.novell.com`.) For installation step 1, choose the TOKEN ODI LAN driver. In step 2, set IPX support for DOS and Windows to On, and select Global NetWare Shell Support. Choose the default settings for the `AUTOEXEC.BAT` file, and make sure that both `TBMI2.COM` and `NETX.EXE` are selected. For step 3, select SPX Support for OS/2 Sessions. While this may not be necessary immediately, it is used by NPRINTER and by cc:Mail, just to give two examples. Save the `CONFIG.SYS` file when prompted. Don't reboot yet.

7. Set up the NET.CFG file to point to the server where the user will log in, which should be different from the RPL server. You can use the Configuration option in the installation program or use an editor to make the changes. See the sample NET.CFG file in Figure 1.
8. Make sure the workstation loads only the programs that the RPL workstations need. Programs can be loaded from the CONFIG.SYS or .CMD files or from the Startup folder. To help shorten boot time, we disabled programs and connections autostart in the CONFIG.SYS file: SET AUTOSTART=TASKLIST,FOLDERS, LAUNCHPAD.
9. Shut down, reboot, and test the interface to NetWare. Make sure you can log in to your existing NetWare file server.
10. Install and test other Windows applications that will run "locally" (i.e., that run from the shared C: drive instead of as a network application).
11. Make sure all programs the RPL workstations need at startup are located on drive C:. For example, make sure your e-mail program on C:\EMAIL does not store mail files in the D:\MYMAIL directory or use E:\TEMP for initialization files. The RPL workstations will not have drives D: or E: available.
12. Install and test Windows applications that will run from the network. By "network" we mean a file server other than the RPL server.
13. Install any of the OS/2 Warp BonusPak applications that you want to use from the shared C: drive. These may be installed on the network.
14. Arrange the Windows and OS/2 desktops the way you want the RPL workstation to appear. The RPL workstation's initial desktop will look like the client image workstation's desktop. These desktops can be altered by individual users later.
15. Shut down and reboot. Save the desktop (via shutdown and reboot of OS/2) before you copy the OS/2 files to the server.

Prerequisites for the RPL Workstation

RPL workstations must have network boards with remote boot programmable read only memory (PROM) attached.

Remote boot PROMs allow workstations to connect to the network to access boot information.

Determine the type of remote boot PROM you have. The RPL workstation installation will ask if you have a "New Enhanced Boot PROM" or an "Older Style Boot PROM." Older style boot PROMs were made before 1992. If you don't know what type of PROM you have, find the part number and call the PROM manufacturer. Our testing has shown that OS/2 RPL works only with the New Enhanced Boot PROM.

Make sure the workstation is cabled to the network and that you know the network and node address of the RPL workstation.

Prerequisites for the RPL Server

The RPL server must be running either NetWare 3 or 4. All of our tests used NetWare 3.12, thus all of the instructions below are for that version.

You must have supervisor object rights to the RPL server.

You must have enough disk space on the RPL server's SYS: volume to contain the OS/2 image and other files necessary for RPL. Novell recommends a minimum of 30 MB, but realistically you need 50 MB for OS/2 Warp plus enough space for the additional user files. In our lab setup running OS/2 Warp, we required a 2 MB SWAPPER.DAT file, plus about 0.5 MB for various system files. To provide a safety margin in case the SWAPPER.DAT grows large, or for large print jobs that take up extra spool space, it is best to have at least an extra 10 MB of disk space per user. After a period of use, you can reevaluate this requirement based on your usage patterns.

Software fixes to be applied: Acquire 312PT6.EXE (or the most current) from Novell. To STARTUP.NCF, add PM312.NLM, DIRSPCFX.NLM, EAWRITFX.NLM, EAACCFIX.NLM, EADATFIX.NLM, EAFLTFIX.NLM, EASUBFIX.NLM, EAWRNFX.NLM, and EAREPLFX.NLM (see the sample STARTUP.NCF file in Figure 2). These patches fix problems reading and writing OS/2's extended attributes on the file server.

```
link driver token
   frame token-ring
link support
   buffers 14 4202
netware requester
   preferred server FS1
```

Figure 1. Sample NET.CFG File

```
load ps2opt, slot=1
load pm312
load dirspcfx
load eaaccfix
load eafltfix
load eareplfx
load easubfix
load eawritfx
load eadatfix
load eawrnfx
```

Figure 2. Sample STARTUP.NCF File

```
file server name RPL1
ipx internal net 2CAAC634
load TOKEN slot=3 frame=TOKEN-RING
bind IPX to TOKEN net=9
load route
load rpl
bind rpl to token ps=rpl1 nodefault
```

Figure 3. Sample AUTOEXEC.NCF File

RPL Server Administration

When adding RPL support to your server, first load RPL.NLM for IBM boards on the RPL server. If you use IBM boards or boards that support New Enhanced Boot PROMs, use the following procedure.

To set up the server to use RPL with this type of workstation, load and bind the RPL loadable module on each server. Add the statements in Figure 3 to the AUTOEXEC.NCF file on the file server so that they run every time the server is started.

Replace token with the name of your server's LAN driver. To bind RPL to more than one driver, use additional BIND statements. Replace RPL1 with the name of the server that has the RPL files.

Creating Users and Granting Rights

To complete the RPL installation, you must have supervisor privileges on the server to which the RPL workstation will attach. Set up accounts for each of the RPL workstations with the following procedure:

1. Log in to the RPL server as supervisor or supervisor-equivalent.
2. Using SYSCON, create the username RPL. At this point, do not automatically create the home directory. RPL is the username that all the RPL workstations will use to make the initial connection to the network. Do not require a password for the RPL user, or the RPL workstations can't make the connection. Protect the account by granting limited access (done in step 4).
3. Create a user for each person who will use an RPL workstation. In our example, our user is named JOHN. Again, do not automatically create the home directories. Although the RPL workstation installation process creates the user directories for you, the installation will work more smoothly if you open an OS/2 window and create the directories now. For each user who will RPL, create a directory in the USER directory, naming it the same as the user name. For example, if you have a user JOHN, create the directory SYS:RPL2\USER\JOHN. In this article, we call this the user's home directory.
4. Grant access to the user name RPL. The RPL user needs at least read and file-scan access to the SYS:RPL2 and the SYS:RPL2\COMPUTER subdirectories.
5. Grant minimal access rights to each RPL workstation user (JOHN). All RPL workstation users need read and file-scan access to the SYS:RPL2, SYS:RPL2\OS2, and SYS:RPL2\NETWARE subdirectories. Grant each user supervisory access rights to his or her home directories under SYS:RPL2\USER (SYS:RPL2\USER\JOHN). This can probably be satisfied by granting read, write, create, erase, modify, and file-scan rights to the user's home directory. However, Novell recommends supervisory access rights.

   The easiest, most manageable way to grant access to the SYS:RPL2 directory is to create a group (for example, RPLOS2); grant the group read and file-scan rights to the SYS:RPL2 directory; and add the RPL users (JOHN) to the group (including the user RPL). You do not have to give users access to other users' home directories under SYS:RPL2\USER. (The NetWare manuals explain more about NetWare security as well as granting and prohibiting access.)

   Do not put any time restrictions on any of the RPL users. Any user with time restrictions will automatically be logged off soon after his or her authorized time interval expires. Since the RPL users must always be logged in to have their C: drive available, this automatic logoff could cause data integrity problems if the C: drive were suddenly removed without performing a shutdown. Also note that, since the SYS:RPL2 directory becomes the root of drive C: for all RPL workstations, all RPL users require access to this area. Do not store confidential files in this directory. Instead, store them in another network directory where access can be restricted.

Adding RPL Workstations

When you add RPL workstations to a network, the following actions occur: 1) directories are created on the RPL server, 2) files are created on the RPL server, and 3) files are copied to the RPL server and the CONFIG.SYS file is copied to the RPL server and modified. Each of these actions is detailed below.

Directories are Created

If you have not installed RPL workstation support, the SYS:RPL2 and the SYS:RPL2\USER and SYS:RPL2\COMPUTER directories will be created on each RPL Server.

Directories will also be created for each workstation when you add a user's home directory under the SYS:RPL2\USER directory. This directory has the same name as the username you specify. For example, a user named JADAMS would have a home directory of SYS:RPL2\USER\JADAMS. If you specify a logical name (for example, LAB), then the home directory will be SYS:RPL2\USER\JADAMS\LAB.

A user's node address directory is created as SYS:RPL2\COMPUTER. This directory has the same name as the last 11 characters of the workstation node address. For example, if the node address or universally administered address (UAA) is 1223345BBCCC, the node address directory is SYS:RPL2\COMPUTER\223345BB.CCC.

A subdirectory containing the OS/2 desktop is created in each user's home directory. This directory is called DESKTOP in OS/2 Warp, for example, SYS:RPL2\OS2\USER\JADAMS\DESKTOP.

Finally, the SPOOL subdirectory is created in each user's home directory. This directory contains all desktop printing files, for example, SYS:RPL2\OS2\USER\JADAMS\SPOOL.

Files are Created

A file called CONFIG.WSS is created in each user's node address directory on the server. This file tells OS/2 to search the user's home directory on the network for the OS2.INI, OS2SYS.INI, CONFIG.SYS, and other listed files instead of trying to find these files in their typical locations on a local hard drive. It also tells OS/2 to use the desktop files and \SPOOL subdirectory from the user's home directory on the network rather than from a local hard disk. After the CONFIG.WSS file is created, it will need to be updated to include the files necessary to have individual Windows .INI and .GRP files. A sample CONFIG.WSS file is shown in Figure 4.

A BOOTCONF.SYS file is created in the SYS:LOGIN directory. If a BOOTCONF.SYS file exists (from other RPL workstation installations), the information for the new workstations is added to the beginning of the file. This file tells the server which boot image file to use for each workstation, and it contains lines identifying which RPL workstation uses which type of network board. Each entry consists of 0x followed by the 12-digit address of the Token-Ring card, an equal sign, and finally the name of the remote boot image file. For example, a line for a token workstation might be: 0x10005ae83ec2 = RPLOS2.200.

Files Are Copied

All files in the \DESKTOP and \SPOOL subdirectories are copied from the local hard drive of the OS/2 client image workstation to the RPL server. Also, all files in CONFIG.WSS are copied. CONFIG.WSS must be customized later for Windows

files. See the sample CONFIG.WSS file in Figure 4.

These files allow RPL workstations to load customized versions of OS/2 and the OS/2 desktop. Your users will be able to use any color scheme, desktop bitmap, or object arrangement they like.

CONFIG.SYS is Copied and Modified

A CONFIG.SYS file is copied from the local hard drive of the OS/2 client image workstation from which you are installing to each user's subdirectory on each RPL server. Figure 5 shows a sample of the CONFIG.SYS file from the client image workstation.

This CONFIG.SYS file is modified slightly for each user. The line that loads the High-Performance File System (HPFS) is commented out, and the directory in the SWAPPATH line is changed. All lines beginning with IFS= are commented out before the IFS=NWIFS line. The NWIFS line must be the first installable file system loaded. If the user wants to run a CD-ROM drive from the RPL workstation, the IFS=CDFS line must be moved to after the NetWare Requester statements in the CONFIG.SYS file. Also, disk caching is turned off.

Figure 6, "Directory Structure for RPL Workstation Support," gives a complete network directory structure for remote workstations.

Installing OS/2 Files for RPL Workstations

When you install OS/2 files for RPL workstations, several actions occur.

A SYS:RPL2 directory is created on each RPL server you select. All RPL workstations use this directory as though it is their local boot drive. Once RPL users log in, they can access all information in this directory. Since the SYS:RPL2 directory is on the server and not on a local disk, NetWare security still applies. For example, you can limit RPL users' access to specific directories and files. See the "Creating Users and Granting Rights" section earlier in this article for more about securing the SYS:RPL2 directory and its subdirectories.

Every file from workstation drive C: is copied to the SYS:RPL2 directory on the server. If you set up PS/2 computers as

```
USERNAME jadams
DIRECTORIES
"C:\DESKTOP" "C:\USER\jadams\DESKTOP"
"C:\NOWHERE" "C:\USER\jadams\NOWHERE"
"C:\SPOOL" "C:\USER\jadams\SPOOL"
FILES
"C:\NETWARE\NWREQOS2.MSG" "C:\NETWARE\NLS\ENGLISH\NWREQOS2.MSG"
"C:\CONFIG.SYS" "C:\USER\jadams\CONFIG.SYS"
"C:\WP000000.$$$" "C:\USER\jadams\WP000000.$$$"
"C:\STARTUP.CMD" "C:\USER\jadams\STARTUP.CMD"
"C:\OS2INIT.CMD" "C:\USER\jadams\OS2INIT.CMD"
"C:\AUTOEXEC.BAT" "C:\USER\jadams\AUTOEXEC.BAT"
"C:\NET.CFG" "C:\USER\jadams\NET.CFG"
"C:\OS2\MDOS\WINOS2\SYSTEM\SETUP.INF" "C:\USER\jadams\SETUP.INF"
"C:\OS2\OS2.INI" "C:\USER\jadams\OS2.INI"
"C:\OS2\OS2.###" "C:\USER\jadams\OS2.###"
"C:\OS2\OS2.!!!" "C:\USER\jadams\OS2.!!!"
"C:\OS2\OS2SYS.INI" "C:\USER\jadams\OS2SYS.INI"
"C:\OS2\OS2SYS.###" "C:\USER\jadams\OS2SYS.###"
"C:\OS2\OS2SYS.!!!" "C:\USER\jadams\OS2SYS.!!!"
"C:\OS2\NWTOOLS.INI" "C:\USER\jadams\NWTOOLS.INI"
"C:\OS2\NWTOOLS.!!!" "C:\USER\jadams\NWTOOLS.!!!"
"C:\OS2\NWTOOLS.###" "C:\USER\jadams\NWTOOLS.###"
"C:\OS2\SYSTEM\SWAPPER.DAT" "C:\USER\jadams\SWAPPER.DAT"
"C:\WINDOWS\WIN.INI" "C:\USER\jadams\WIN.INI"
"C:\WINDOWS\MOUSE.INI" "C:\USER\jadams\MOUSE.INI"
"C:\WINDOWS\CONTROL.INI" "C:\USER\jadams\CONTROL.INI"
"C:\WINDOWS\MSD.INI" "C:\USER\jadams\MSD.INI"
"C:\WINDOWS\PROGMAN.INI" "C:\USER\jadams\PROGMAN.INI"
"C:\WINDOWS\ATM.GRP" "C:\USER\jadams\ATM.GRP"
"C:\WINDOWS\MAIN.GRP" "C:\USER\u1\MAIN.GRP"
"C:\WINDOWS\ACCESSOR.GRP" "C:\USER\jadams\ACCESSOR.GRP"
"C:\WINDOWS\GAMES.GRP" "C:\USER\jadams\GAMES.GRP"
"C:\WINDOWS\STARTUP.GRP" "C:\USER\jadams\STARTUP.GRP"
"C:\WINDOWS\ATM.GRP" "C:\USER\jadams\ATM.GRP"
"C:\WINDOWS\5250ELIT.GRP" "C:\USER\jadams\LOTUS.GRP"
```

Figure 4. Sample CONFIG.WSS File

RPL workstations, you must obtain an OS/2 Warp installation diskette, because the installation program copies some files from this diskette. Complete the steps below only if you are setting up RPL workstations for the first time, upgrading OS/2 on the RPL server, or adding a new RPL server to the network where RPL workstations will attach (only for the server you add).

1. On the client image workstation, log in as supervisor to the RPL server where you want to put the OS/2 files for RPL workstations.
2. Open the Novell group icon on the desktop.
3. Choose the Install icon in the Novell-Icon View window. The NetWare Workstation for OS/2 Installation Utility window appears.
4. From the Installation menu, choose Remote workstations . . . The Remote Workstation Installation window appears. *Note:* The Copy All Files and Setup Workstation . . . option has three separate components: copying OS/2 files, copying RPL files, and setting up RPL workstations. Each of these procedures is explained individually.
5. Select Only Copy OS/2 Files . . . and click on OK. You can choose other actions from the Remote Workstation Installation screen.
6. Choose the servers onto which you want the files copied. You are attached to the servers you select.
7. Click on OK. All OS/2 and NetWare Client for OS/2 files are copied to the servers.
8. The program will ask if you will be using PS/2 systems. If so, this is where you will use the OS/2 Warp installation diskette. Make sure the source drive is A: (you will have to change it).

Installing RPL Files for RPL Workstations

When you use the installation program to install RPL workstation support, several actions occur.

Directories are created on the RPL server. SYS:RPL2 is created if it doesn't already

```
PROTSHELL=C:\OS2\PMSHELL.EXE
SET USER_INI=C:\OS2\OS2.INI
SET SYSTEM_INI=C:\OS2\OS2SYS.INI
SET OS2_SHELL=C:\OS2\CMD.EXE
SET AUTOSTART=TASKLIST,FOLDERS,LAUNCHPAD
SET RUNWORKPLACE=C:\OS2\PMSHELL.EXE
SET COMSPEC=C:\OS2\CMD.EXE
LIBPATH=C:\IBMCOM\DLL;.;C:\OS2\DLL;C:\OS2\MDOS;C:\;C:\OS2\APPS\DLL;
      C:\MMOS2\DLL;C:\NETWARE;C:\NETWARE\NLS\ENGLISH;L:\OS2;P:\OS2;
SET PATH=C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;C:\;C:\OS2\MDOS;
      C:\OS2\APPS;C:\WINDOWS;C:\MMOS2;C:\NETWARE;L:\OS2;P:\OS2;
SET DPATH=C:\IBMCOM;C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;C:\;
      C:\OS2\BITMAP;C:\OS2\MDOS;C:\OS2\APPS;C:\WINDOWS;C:\MMOS2;
      C:\MMOS2\INSTALL;C:\NETWARE;C:\NETWARE\NLS\ENGLISH;L:\NLS;P:\NLS;
SET PROMPT=$I[$P]
SET HELP=C:\OS2\HELP;C:\OS2\HELP\TUTORIAL;C:\MMOS2\HELP;
      C:\NETWARE\NLS\ENGLISH;
SET GLOSSARY=C:\OS2\HELP\GLOSS;
SET IPF_KEYS=SBCS
PRIORITY_DISK_IO=YES
FILES=20
BASEDEV=IBMKBD.SYS
DEVICE=C:\OS2\BOOT\TESTCFG.SYS
DEVICE=C:\OS2\BOOT\DOS.SYS
DEVICE=C:\OS2\BOOT\PMDD.SYS
BUFFERS=90
IOPL=YES
REM DISKCACHE=D,LW,AC:C
MAXWAIT=3
MEMMAN=SWAP,PROTECT
SWAPPATH=C:\USER\JADAMS 512
BREAK=OFF
THREADS=256
PRINTMONBUFSIZE=134,134,134
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
SET KEYS=ON
SET BOOKSHELF=C:\OS2\BOOK;C:\MMOS2;
SET SOMIR=C:\OS2\ETC\SOM.IR;C:\OS2\ETC\WPSH.IR;C:\OS2\ETC\WPDSERV.IR
SET SOMDDIR=C:\OS2\ETC\DSOM
REM SET DELDIR=C:\DELETE,512;
BASEDEV=PRINT02.SYS
BASEDEV=IBM2FLPY.ADD
BASEDEV=IBM1FLPY.ADD
BASEDEV=IBM2SCSI.ADD /LED
BASEDEV=XDFLOPPY.FLT
BASEDEV=OS2DASD.DMD
SET EPMPATH=C:\OS2\APPS;
PROTECTONLY=NO
SHELL=C:\OS2\MDOS\COMMAND.COM C:\OS2\MDOS
FCBS=16,8
RMSIZE=640
DEVICE=C:\NETWARE\VIPX.SYS
DEVICE=C:\NETWARE\VSHELL.SYS GLOBAL
DEVICE=C:\OS2\MDOS\VEMM.SYS
DOS=LOW,NOUMB
DEVICE=C:\OS2\MDOS\VXMS.SYS /UMB
DEVICE=C:\OS2\MDOS\VDPMI.SYS
DEVICE=C:\OS2\MDOS\VDPX.SYS
DEVICE=C:\OS2\MDOS\VWIN.SYS
DEVICE=C:\OS2\MDOS\VW32S.SYS
BASEDEV=OS2SCSI.DMD
DEVICE=C:\OS2\MDOS\VMOUSE.SYS
DEVICE=C:\OS2\BOOT\POINTDD.SYS
DEVICE=C:\OS2\BOOT\MOUSE.SYS
DEVICE=C:\OS2\BOOT\COM.SYS
DEVICE=C:\OS2\MDOS\VCOM.SYS
CODEPAGE=437,850
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
BASEDEV=XGA.SYS
DEVICE=C:\OS2\BOOT\XGARING0.SYS
DEVINFO=SCR,VGA,C:\OS2\BOOT\VIOTBL.DCP
SET VIDEO_DEVICES=VIO_XGA
SET VIO_XGA=DEVICE(BVHVGA,BVHXGA)
```

Figure 5. Sample CONFIG.SYS File (continued on next page)

exist. The \USER and \COMPUTER subdirectories are created under SYS:RPL2 on each server. The following RPL files (found on OS/2 NetWare Requester diskette WSOS2_2) are copied to the SYS:LOGIN directory on the RPL server: NE2.200, NE2000.200, NE1000.200, RPLOS2.200, RBOOT.RPL, ETHER.RPL, and TOKEN.RPL.

The MINI.IFS and MICRO.IFS files are copied to the SYS:RPL2 directory.

Complete the steps in this section only if you are setting up RPL workstations for the first time, upgrading existing RPL workstations to a new version of OS/2, installing a new version of NetWare Client for OS/2, or adding a new RPL server to the network where RPL workstations will attach (only for the server you add). Don't do the steps in this section every time you want to add an RPL workstation to the network. (See "Setting Up RPL Workstations.") If the installation program is already running, go to Step 4.

1. On the client image workstation, log in as supervisor to the RPL server where you want to put the OS/2 files for RPL workstations.
2. Open the Novell group icon on the desktop.
3. Choose the Install icon in the Novell-Icon View window. The NetWare Workstation for OS/2 Installation Utility window appears.
4. From the Installation menu, choose Remote workstations . . . The Remote Workstation Installation window appears. *Note:* The Copy All Files and Setup Workstation . . . option has three separate components: copying OS/2 files, copying RPL files, and setting up RPL workstations. Each of these procedures is explained individually.
5. Choose Only Copy RPL Files. . . You can also choose other actions from the Remote Workstation Installation screen.
6. Choose the servers onto which you want the RPL files copied. You will be attached to the servers you select.
7. Click on OK. All RPL boot files are copied to the servers.
8. Select the source drive for the RPL files. The default is the drive where

INSTALL.EXE was loaded. If you loaded INSTALL.EXE from the hard drive, change this to your floppy drive and insert the first installation diskette.

9. Click on OK.

Setting Up RPL Workstations

Each RPL workstation that you boot from the network must be added to the RPL setup on the RPL server. You identify to the server the user for that workstation, the node address (UAA), and the network board.

The UAA is the universally administered address, which is permanently set on the LAN card by the card manufacturer. RPL workstations are added by running the NetWare Client for OS/2 installation program from the client image workstation.

Complete the steps in this section on each server if you are setting up RPL workstations for the first time, upgrading existing RPL workstations from OS/2 1.3 to OS/2 Warp, or adding RPL workstations to the network. If the installation program is already running, go to Step 4.

1. On the client image workstation, log in as supervisor to the RPL server where you want to put the OS/2 files for RPL workstations.
2. Open the Novell group icon on the desktop.
3. Choose the Install icon in the Novell-Icon View window. The NetWare Workstation for OS/2 Installation Utility window appears.
4. From the Installation menu, choose Remote workstations . . . The Remote Workstation Installation window appears. *Note:* The Copy All Files and Setup Workstation . . . option has three separate components: copying OS/2 files, copying RPL files, and setting up RPL workstations. Each of these procedures is explained individually.
5. Choose Only Set Up Workstations . . .
6. Click on OK. The Select 1 or More File Servers window appears.
7. Choose servers on which you want to define RPL workstations. For a list of available servers, choose Attach. Then choose the arrow at the end of the Server name field. For more information, use the online help.

```
DEVICE=C:\OS2\MDOS\VVGA.SYS
DEVICE=C:\OS2\MDOS\VXGA.SYS
REM - NETWARE REQUESTER STATEMENTS BEGIN -
SET NWLANGUAGE=ENGLISH
DEVICE=C:\NETWARE\LSL.SYS
RUN=C:\NETWARE\DDAEMON.EXE
REM - ODI-DRIVER FILES BEGIN -
DEVICE=C:\NETWARE\TOKEN.SYS
REM - ODI-DRIVER FILES END -
DEVICE=C:\NETWARE\ROUTE.SYS
DEVICE=C:\NETWARE\ODINSUP.SYS
DEVICE=C:\NETWARE\IPX.SYS
DEVICE=C:\NETWARE\SPX.SYS
RUN=C:\NETWARE\SPDAEMON.EXE
REM DEVICE=C:\NETWARE\NMPIPE.SYS
REM DEVICE=C:\NETWARE\NPSERVER.SYS
REM RUN=C:\NETWARE\NPDAEMON.EXE
DEVICE=C:\NETWARE\NWREQ.SYS
IFS=C:\NETWARE\NWIFS.IFS
RUN=C:\NETWARE\NWDAEMON.EXE
REM DEVICE=C:\NETWARE\NETBIOS.SYS
REM RUN=C:\NETWARE\NBDAEMON.EXE
DEVICE=C:\OS2\MDOS\LPTDD.SYS
REM - NETWARE REQUESTER STATEMENTS END -
```

Figure 5. Sample CONFIG.SYS File

8. When you've selected all the servers you want, click on OK. The Select the Type of Boot PROM in the target workstation window appears.
9. Remember, you should be using the new enhanced boot PROMs. Select New Enhanced Boot PROM.
10. Click on OK. The Add Remote Boot Workstation window appears.
11. Complete all necessary fields in the window. These fields will usually be the Node Address, ODI LAN Driver, and User Name. Choose TOKEN.SYS for the LAN driver. User Name should be the actual name or login name of the person for whom you are setting up the RPL workstation. Logical Name becomes the name of the subdirectory under that user. For example, if you set up the user name as SUSAN and the logical name is LAB, the path would read RPL2\USER\SUSAN\LAB. You must use a logical name any time you have a user who will need to be defined on more than one physical workstation.
12. Choose Add.
13. Exit the installation program.
14. Copy all of the Windows .INI and .GRP files to each RPL user's home directory. Also, edit each user's PROGMAN.INI file to point to the .GRP files on the user's home directory. See the sample PROGMAN.INI file in Figure 7.

This is a diagram of the network directory structure for remote workstations.

```
SYS:
\RPL2 (contains OS/2 System files
       found in the root of C: drive)
   \OS2 (OS/2 files)
   \NETWARE (Requester files)
   \USER (User's home directories)
   \USER1
      CONFIG.SYS
      OS2SYS.INI
      OS2.INI
   \DESKTOP
   \SPOOL
Other system files, including Windows
.INI and .GRP files
   \USERx
   \LOGICAL
      CONFIG.SYS
      OS2SYS.INI
      OS2.INI
   \DESKTOP
   \SPOOL
Other system files, including Windows
.INI and .GRP files
   etc...
   \COMPUTER
   \NodeAddr.001
      CONFIG.WSS
   \NodeAddr.002
      CONFIG.WSS
   etc...
```

Figure 6. Directory Structure for RPL Workstation Support

```
[Settings]
Window=70 10 1017 750 1
Order= 3 4 5 1 2 6
display.drv=vga.drv

[Groups]
Group1=C:\USER\U1\MAIN.GRP
Group2=C:\USER\U1\ACCESSOR.GRP
Group3=C:\USER\U1\GAMES.GRP
Group4=C:\USER\U1\STARTUP.GRP
Group5=C:\USER\U1\ATM.GRP
Group6=C:\USER\U1\LOTUS.GRP
```

Figure 7. Sample PROGMAN.INI File

15. The CONFIG.WSS file must be updated for each workstation to include the Windows .INI and .GRP files. This file tells the NetWare redirector where to look for particular files. (See the sample CONFIG.WSS in Figure 4.)

    The CONFIG.WSS is found in the user's node address directory under the RPL2\COMPUTER directory. This directory has the same name as the last 11 characters of the workstation node address. For example, if the node address is 1223345BBCCC, the node address directory would be SYS:RPL2\COMPUTER\223345BB.CCC. In the future, you may install other applications that expect to find files on a certain directory, thus requiring additional updates to this file.

Configuring NetWare Client for OS/2 for RPL Workstations

You can create a NET.CFG file for RPL workstations two ways: by running the NetWare Client for OS/2 installation program and choosing Remote workstations. . . from the Configuration menu or by creating a NET.CFG file on the client image OS/2 machine, which then gets copied to the server when you copy all of the OS/2 files. We recommend the second option, since the majority of users will have the same NET.CFG.

When you configure RPL workstations, the NetWare Client for OS/2 installation program puts a NET.CFG configuration file in the home directory of the users you specify. The home directory is located under SYS:RPL2\USER on each server on the local network.

The installation program puts a line in each user's CONFIG.WSS file in the workstation subdirectory, telling NetWare Client for OS/2 where to find the NET.CFG file. For example, for JOHN's NET.CFG file, the installation program puts in the following (all on one line): "C:\NET.CFG" "C:\USER\JOHN\NET.CFG"

Miscellaneous Tips

To use RPL on a workstation with a hard disk, label your hard drive to be other than C:. Assign drive C: to the SYS:RPL2 network directory. OS/2 thinks this network directory is the local drive C:.

The local hard drive should have enough space to accommodate the expected size of the OS/2 SWAPPER.DAT file. It should be formatted as File Allocation Table (FAT). The main benefit of a local swap file is to decrease network traffic.

Locating SWAPPER.DAT Locally

To set up a local SWAPPER.DAT file:

1. Go to the home directory on the network for the user who will be using the RPL workstation. For example, if JOHN will be using the RPL workstation with a hard drive, go to the directory SYS:RPL2\USER\JOHN. Each user's home directory on the network contains that user's CONFIG.SYS file and several other OS/2 files. Figure 6, "Directory Structure for RPL Workstation Support," shows a complete directory structure.
2. Using a text editor, open the CONFIG.SYS file.
3. In the CONFIG.SYS file, find the line that designates the SWAPPATH. You can search for SWAPPATH. The line looks like the following: SWAPPATH=C:\USER\JOHN 2048 2048. Remember that to OS/2, the SYS:RPL2 directory is drive C:.
4. Change the SWAPPATH line to point to the local hard drive. For example, to point to the root of local drive D: SWAPPATH=D:\ 2048 2048.
5. Save and exit the CONFIG.SYS file. When you boot the RPL workstation, SWAPPER.DAT will be stored locally.

Using a RPL Workstation with Virtual DOS

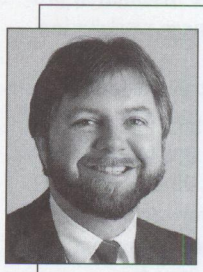
RPL workstations are unable to remap drive C:. Even if you are using global DOS, RPL workstations treat global DOS the same as private DOS. For example, if you were in global DOS and remapped drive C:, the remapped drive C: will not show up when you go to a private DOS session.

Question Marks on the Launchpad

After each RPL workstation boots for the first time (and only the first time), all icons on the Launchpad will be question marks. This is a RPL restriction for which we have not found a permanent fix. However, it is easy to fix temporarily by simply dragging each question-mark object to the shredder and deleting it. As soon as the last one is deleted, the Launchpad automatically rebuilds with the correct icons.

Get Warped!

Now you, too, can get Warped, even if you don't have a hard drive in your workstation!



Steven H. Bumgarner is an advisory marketing support representative in IBM's Personal Systems Competency Center in Roanoke, Texas. He is a certified Novell Enterprise CNE and a certified OS/2 engineer. He provides technical support and consulting services for NetWare and OS/2. Steven has worked with and taught classes on Novell products and OS/2 for seven years and has been involved with local area networks at IBM since he was hired in 1979. Steven's Internet ID is bummer@vnet.ibm.com.

Jay L. Curtiss is an information technology specialist with the IBM Wisconsin Availability Center in Milwaukee, providing customers with software support services for local area networks and OS/2. He has held a variety of positions during 12 years with IBM, including CMS development in Endicott, New York; systems engineer in Lincoln, Nebraska; and most recently, technical support at the Personal Systems Competency Center in Roanoke, Texas. He is a certified Novell Enterprise CNE and IBM OS/2 engineer. Jay has BS degrees in Agricultural Economics and Computer Science from the University of Nebraska, home of the national champion Cornhuskers. His Internet address is jcurtiss@vnet.ibm.com.

Questions & Answers

What is an "outer join" process, and how do I implement it using DB2 for OS/2?

A join is the process of combining data from two or more tables into a result table based on data that is the same in both tables. A join may be an inner join or an outer join. With the *inner join,* the result table will contain information only from rows that exist in both tables. An *outer join* is designed to preserve unmatched rows in the result table.

As an example, assume you have the tables in Figure 1 to track parts and suppliers. Using an inner join, you get only parts that have suppliers and suppliers that have parts. What if you have parts with no suppliers and suppliers with no parts? If you want a result table that includes them all, you must implement an outer join process. The outer join technique used in Figure 2 uses the CAST specification, a new capability in DB2 for OS/2 V2.1.

Table	Columns
PARTS	PART#, PARTNAME
SUPPLIERS	SUPP#, SUPPNAME
PART_SUPP	PART#, SUPP#

Figure 1. Parts and Suppliers Tables

The SQL example in Figures 2 and 3 uses UNION and EXISTS clauses to bring into the result table all suppliers for all parts and also all those parts that have no suppliers.

The example in Figure 2 is called a *left outer join* since it includes the inner join and the rows from the left table (in the FROM clause in the inner join) that were not matched when performing the inner join. If you want a *full outer join,* where the suppliers with no parts are also listed, you need the UNION statement in Figure 3 added to the SQL statement in Figure 2.

```
/*---------------------------------------*/
/* inner join                            */
/*---------------------------------------*/

SELECT P.PARTNAME, S.SUPPNAME
   FROM PARTS P, SUPPLIERS S, PART_SUPP X
     WHERE X.PART# = P.PART # AND X.SUPP# = S.SUPP#
UNION

/*---------------------------------------*/
/* rows missing from the left table      */
/*---------------------------------------*/

SELECT P.PARTNAME, CAST (NULL AS VARCHAR(20)) AS SUPPNAME
   FROM PARTS P
     WHERE NOT EXISTS (SELECT PART#
                       FROM PART_SUPP X
                       WHERE X.PART# = P.PART#)
```

Figure 2. Left Outer Join

```
/*---------------------------------------*/
/* rows missing from the right table     */
/*---------------------------------------*/

UNION

SELECT CAST (NULL AS VARCHAR(20)), S.SUPPNAME
   FROM SUPPLIERS S
     WHERE NOT EXISTS (SELECT SUPP#
                       FROM PART_SUPP X
                       WHERE X.SUPP# = S.SUPP#)
```

Figure 3. Full Outer Join

I have a database created under code page 850. Now I am running under code page 437 and need to access the database. Is there any way to change the code page of the database?

Once a database has been created in a particular code page, you cannot change the code page of the database. Your alternatives for accessing the database will depend on the version of DB2 for OS/2 you are running.

The alternatives for Version 1.x are:

1. Change your code page to 850. If you have the statement CODEPAGE 437,850 in your CONFIG.SYS file, you can enter the command CHCP 850 to change your currently running code page from 437 to 850.
2. If changing your code page is not an alternative, you must export all the tables in IXF format, using an application running code page 850. You will then need to create a new database using code page 437 and import all the tables, using the FORCEIN option. The FORCEIN option directs the import

Code Pages	Countries
437, 819, 850, 863 1004, 1051, 1252	Austria, Australia, Belgium, Canada, Denmark, Finland, France, Germany, Italy, Latin America, Netherlands, Norway, Portugal, Spain, Sweden, Switzerland, UK, USA
813, 869, 1253	Greece
852, 912, 1250	Croatia, Czech Republic, Hungary, Poland, Romania, Serbia/Montenegro (Latin), Slovakia, Slovenia
855, 915, 1251	Bulgaria, FYR Macedonia, Serbia/Montenegro (Cyrillic)
857, 920, 1254	Turkey
862, 916, 1255	Israel
864, 1046, 1089, 1256	Arabic countries
866, 915, 1251	Russia
932, 942	Japan
938, 948	Taiwan
949, 970	Korea
1381, 1383	People's Republic of China

Figure 4. Supported Code Page Conversions

utility to not check for code page incompatibilities.

DB2 for OS/2 V2.1 provides character conversion support that allows your application and database to use different code pages. If your application is running code page 437, character conversion will take place when you access the database created under code page 850. However, be aware that there is a performance overhead for this conversion. If you want to avoid the character conversion overhead, use one of the options listed for Version 1.x, with this exception: if you import *without* the `FORCEIN` option, the system will perform the code page conversion for you.

Figure 4 shows the code page conversions that are supported by Version 2.1. Any code page can be converted to any other code page that is listed in the same row of the table.

Are there any Internet ftp sites where I can get information and fixes for DB2 for OS/2?

Yes. The following site is available: anonymous ftp to `ps.boulder.ibm.com`.

For DB2 newsletters, go to directory `ps/products/db2/info`.

For the DB2 ServicePak, go to directory `ps/products/db2/fixes/us/db22v12`.

– Nancy Miller, IBM Corporation
Roanoke, Texas

Corrective Service Information

Figure 1 shows maintenance release levels for the listed products. This information is effective as of June 8, 1995. CSDs may have been updated since press time.

To order all service packages–except for the OS/2 2.0, OS/2 2.1, OS/2 2.1 for Windows, and OS/2 2.0 Toolkit ServicePaks–call IBM Software Solution Services at (800) 992-4777. For the OS/2 2.0 ServicePak (XR06100), OS/2 2.1 ServicePak (XR06200), OS/2 2.1 for Windows ServicePak (XR06300), or the IBM Developer's Toolkit for OS/2 2.0 ServicePak (XR06110) on diskettes or CD-ROM, call (800) 494-3044. Most OS/2 service packages are also available electronically from the following sources:

- **OS/2 Bulletin Board Service (BBS)**: Once connected, select Option 2. (Corrective services are also listed under the General category on the IBMLink BBS.) To subscribe to the OS/2 BBS, call (800) 547-1283.
- **IBM Personal Computer Company (PCC) BBS**: Call (919) 517-0001. Service packages are located in Directory 4.
- **CompuServe**: Download service packages from the IBM OS2 FORUM library (GO IBMSERV).
- **Internet**: Do an anonymous FTP from `software.watson.ibm.com`. Most packages are located in the `\PUB\OS2` directory. TCP/IP packages are located in the `\PUB\TCPIP\OS2` directory.

–Arnie Johnson, IBM Corporation, Austin, Texas

Product/Component	Release	CSD Level	PTF Number	Change Date	Comments
OS/2 Standard Edition	1.3	XR05150	XR05150	2-10-93	
OS/2 Extended Edition	1.3	WR05200	WR05200	5-12-93	WR05200 replaces WR05050, which can no longer be ordered on diskette
OS/2	2.0	XR06100	XR06100	9-1-93	XR06100 replaces XR06055.
OS/2 2.10 ServicePak	2.1	XR06200	XR06200	3-1-94	This package is not for OS/2 2.1 for Windows.
OS/2 2.11 for Windows ServicePak	2.11	XR06300	XR06300	5-24-94	
OS/2 Toolkit	2.0	XR06110	XR06110	9-1-93	
	1.3	XR05053	XR05053	3-23-92	
OS/2 LAN Server/Requester ServicePak	2.0	IP06030	IP06030	4-25-93	
OS/2 LAN Server/Requester ServicePak	3.0	IP07060	IP07060	5-10-95	Supersedes IP07045.
OS/2 Extended Services Database Manager ServicePak	1.0	WR06035	WR06035	11-18-93	Supersedes WR06001, WR06002, WR06003, WR06004, WR06014, and WR06015.
DB2/2 SelectPak	1.0	WR07030	WR07030	2-6-95	
DDCS/2 ServicePak	2.0	WR07031	WR07031	2-6-95	
Database Manager DB2/2	1.2	WR07035	WR07035	1-16-95	
DDCS/2	2.0	WR07036	WR07034	1-30-95	Order WR07036 from Boulder on diskettes. Order WR07034 electronically.
Client Application Enabler/2 (CAE/2)	1.2	WR07043	WR07043	6-6-95	
Software Developers Kit/2 (SDK/2)	1.2	WR07048	WR07048	6-6-95	
Extended Services Comm Mgr ServicePak	1.0	WR06025	WR06025	11-29-93	
System Performance Monitor (SPM/2) ServicePak	2.0	WR06075	WR06075	12/10/93	
LAN Distance ServicePak	1.1	IP07050	IP07050	10-18-94	
OS/2 Network Transport Services/2 SelectPak	2.00	WR07060	WR07060	5-10-95	Must be LAPS 2.11 or above. If not, order WR07045 first.

Figure 1. Maintenance Release Levels **(continued on next page)**

Product/Component	Release	CSD Level	PTF Number	Change Date	Comments
OS/2 LAN Adapter and Protocol Support SelectPak	2.20.2	WR07060	WR07060	5-10-95	Must be LAPS 2.11 or above. If not, order WR07045 first.
Communications Manager/2 Version 1.01 ServicePak	1.01	WR06050	WR06050	6-11-93	Available only on diskette.
CM/2 Version 1.11 ServicePak	1.11	WR06150	WR06150	5-31-94	Available on diskette and CD-ROM.
DOS	4.0, 4.01	UR35284	UR35284	9-26-91	
	5.0	UR37387	UR37387	9-22-92	
C Set/2 Compiler	1.0	CS00050	XR06150	6-29-93	
C Set C++ Compiler	2.0/2.01	CTC0002	XR06102	12-15-93	
C Set C++ Compiler	2.0/2.01	CTC0010	XR06190	9-15-94	
C Set C++ Utilities	2.01	CTM0006	XR06196	9-15-94	
C Set C++ Utilities	2.00	CTL0007	XR06197	9-15-94	
TCP/IP for OS/2 Base and Application Kit	2.0	UN64092	UN64092	8-24-94	
TCP/IP for OS/2 DOS Box Kit	2.0	UN57546	UN57546	8-24-94	
TCP/IP for OS/2 Extended Networking	2.0	UN60005	UN60005	6-21-94	
TCP/IP for OS/2 Programmer's Toolkit	2.0	UN57887	UN57887	6-21-94	
TCP/IP for OS/2 Domain Name Server	2.0	UN60004	UN60004	8-24-94	
TCP/IP for OS/2 Network File System	2.0	UN57064	UN57064	6-21-94	
TCP/IP for OS/2 X-Windows Server	2.0	UN68122	UN68122	1-20-95	
TCP/IP for OS/2 X-Windows Client	2.0	UN59347	UN59347	8-24-94	

Figure 1. Maintenance Release Levels

BACK ISSUE INDEX

These back issues of *Personal Systems* are available to provide valuable information. Indicate the desired quantity for the issues you want to order and complete the information on the following page.

___ May/June 1995

What's New for OS/2?
Thanks for the Memory
Road Trip! Disney on the Internet
Apache Students Use the Power of the Pen (Light Pen)
Visualizer: The Conversion Continues
The Internet: A New Dimension?
IBM LAN Doctor Services
Borland C++ 2.0 Brings OWL to the OS/2 Presentation Manager
LAN Server Logon Internals
LAN Server 4.0 Performance, Capacity Enhancements, and Tuning Tips
OS/2 Warp for Developing PC Games
Controlling the OS/2 Desktop From a File Server
Jump-Start Your PC with Component Upgrades

___ March/April 1995

What's New for OS/2?
Mesa 2: Gaining the Competitive Edge with OS/2
Managing the Workplace Shell with DeskMan/2
Circus du COMDEX: The Running of the Geeks
Road Trip! Touring the Side Roads of the Internet
What's New in PC DOS 7
OS/2 Boot and Recovery Options
TCP/IP: How It Works
A Guide to OS/2 Warp's Internet Access Kit
CID Installation of OS/2 Warp and LAPS
Wrapping Up an OO Experience

___ January/February 1995

Technical Connection Personal Software Is the Answer!
Visualizer, DB2, and You–An End-User's Perspective
Insiders' Software Unveiled
Need a Specialist for Your LAN Server 4.0?
One-Stop Shopping
OS/2 Warp
OS/2 for SMP
Multimedia File I/O Services
Need a Fix?
IBM LAN Server 4.0: New Features and Comparisons with NetWare
IBM DCE Hetrogeneous Enterprise Performance

___ November/December 1994

Evolution, Not Revolution–Pen Computing Comes of Age
Handwriting Recognition: The State of the Art
Pen Digitizing Hardware
It's HapPENing!
Bill Carr: Fastest Draw in the West
Work Management in the Field
Communicating Without Wires: IBM's Mobile Communications Module
Tomorrow's Networking Today–from IBM's Personal Systems Competency Center
Customers Speak Out About Consult Line
New DeScribe 5.0–Leader of the Pack
Super-Fast PenDOS
Pen for OS/2
A Development Environment for Pen-Centric Applications
Writing DOS Installation Programs for Selective Boot Systems
OS/2 for PowerPC: Transforming Architecture into Implementation

___ September/October 1994

"Sneaker Net" or Systems Management?
Like Father, Like Son
The Book Shelf
Cajun Electric Cooks Up OS/2 GUI with VisPro/REXX!
Application Development by Program Integration
IBM REXX for NetWare
GammaTech REXX SuperSet/2–Give Your REXX Programs the Power of C
BranchCard: A Viable Option to Stand-Alone Hubs
A Hands-On Primer for REXX
Visual REXX Development Environments
CID Installation of OS/2 2.11 and LAPS
Upgrading from Microsoft LAN Manager to IBM LAN Server 3.0
Stretching Your LAN with LAN Distance
DB2/2–More Than Ever Before!
NetBIOS, SNA, and NetWare IPX Coexistence Under OS/2

___ July/August 1994

IBM's Personal Systems Support Family–Customer-Influenced Design
OS/2 Times and Scores the 1994 Indianapolis 500
Software Compatibility: Good Relationship or One Night Stand?
Migrating Windows Applications to OS/2: Easing the Migration Path
OS/2 Conference Draws Praise
DCE: An Application Primer
Distributed Performance Characteristics of IBM DCE for OS/2
Architecture Soup: Understanding Modern IBM PC Architecture
TSHELL: A Text-Based Alternate Shell for OS/2
Extended Attributes for Files
Developing Lotus Notes Applications
Conserving Power with Personal System Power Management
Superstor/DS Data Compression in PC DOS 6.x
LAN NetView Object Registration Services

___ May/June 1994

"Wrightsizing" at USAir
Getting the Word Out at Chemical Banking Corporation
Back Up for the Future
Lost in Cyberspace
The Book Shelf
Threads
Redirected Installation of OS/2 2.x
LAN Server Ultimedia 1.0 Performance and Tuning

___ March/April 1994

If I Only Had a Brain
Speech Recognition Products Untie Your Hands
Telecommuting in the '90s
Point of View: Not Just Another Database Article
Professional Certification Program from IBM
Celebrate the Past on Your Trip Back to the Future!
OS/2 2.1 Performance Tuning Tips–Part II
PC File Systems
NetWare 4.01 for OS/2: Features and Installation
NetWare Requester for OS/2, V2.01: Features and Installation
What's New in Novell NetWare 3.12?
LAN Analysis Using IBM's Data*gLANce* Network Analyzer
NetWare Questions and Answers

___ January/February 1994

Plan, Plan, Plan Your NetWare 4.01 Network
LAD/2 in the LCU and NetView DM/2 Environments
Easy Setup of CID Code Servers
Managing Token-Ring Bridges with IBM's LAN Network Manager
IBM DCE for OS/2 Multiuser Application Performance
Performance of Key Functions in DCE for OS/2
VisualAge: Its Features and Virtues

___ November/December 1993

IBM PC-DOS 6.1: More Features than MS-DOS 6.0
SystemView Information Warehouse DataHub
Developing DataHub Tools
Using MMPM/2 to Create Multimedia Applications
Advanced Client/Server Computing Using the IBM ThinkPad
Communications Manager/2: A New Look
Overview of IBM NetWare 4.01
OS/2 2.1 Performance Tuning Tips

___ September/October 1993

IBM PSP's LAN Systems Solutions
An Introduction to PCMCIA
PCMCIA Software: The Key to Compatibility
OS/2 Support for PCMCIA Memory Cards
Improving Remote Initial Program Load Performance
Installing and Configuring CM/2 1.0
Writing CID-Enabled Applications
Integrating LAD/2, CM/2, and DB2/2 with IBM LAN NetView Start
DB2/2–DB2 Comes to the Desktop

___ July/August 1993

OS/2 2.1–Everything You Wanted It to Be and More
Using REXX to Customize the Workplace Shell–Part II
Client/Server Application Development with OS/2 and CICS/ESA
Upgrading to OS/2 LAN Server 3.0–Advanced
Developing OS/2 LAN Server Services
PCMCIA PC Cards Provide Expandability and Network Interfacing
Using the IBM ThinkPad with OS/2 and CM/2

Send this form with a check or money order, payable to **NCM Enterprise**, to: NCM Enterprise, P.O. Box 165447, Irving, TX 75016-9939. You can also fax both pages of this form to **(214) 518-2507** (please include VISA / MasterCard /AmEx/Diners number and expiration date), or call **(800) 678-8014**.
All orders must be prepaid. Checks must be in U.S. dollars.

BACK ISSUE ORDER FORM

NAME _____

COMPANY _____

ADDRESS _____

_____

CITY _____ STATE _____ ZIP _____

TELEPHONE (_____) _____

Price is $12.00 per issue, plus $3.95 shipping & handling per copy. Overseas orders add $9.95 shipping & handling per copy.

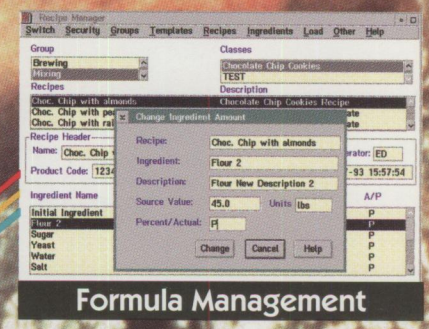
Texas residents add applicable sales tax.

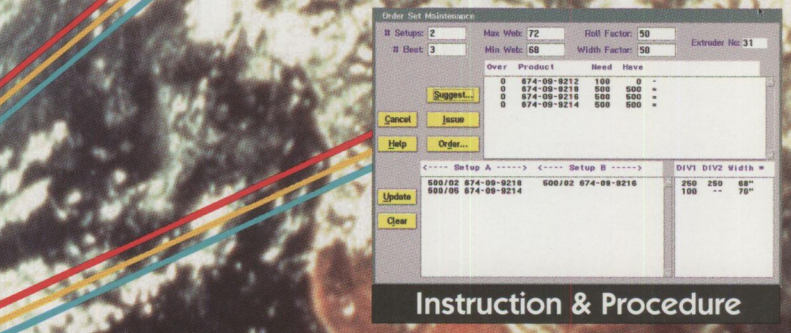
I have enclosed a: ❑ Check ❑ Money order
Charge to: ❑ VISA ❑ MasterCard ❑ AmEx ❑ Diners

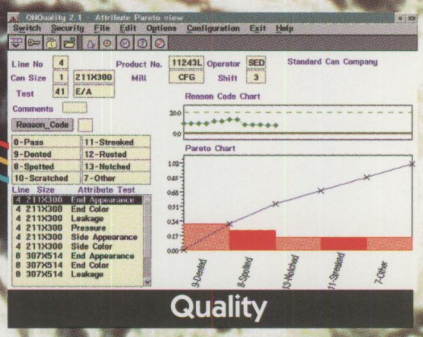
CREDIT CARD NUMBER _____

SIGNATURE _____ EXPIRES _____

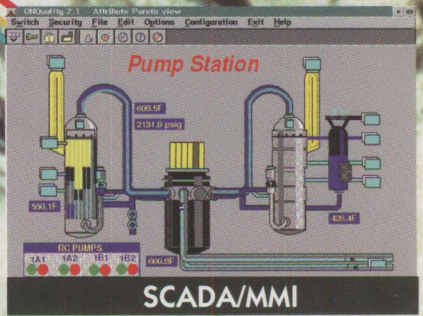
How To Build *Point-to-Plantwide* SOLUTIONS







ONSPEC® Easy



Ready! for OS/2™

Ready! for IBM LAN™

OS/2 Warp

Lotus BUSINESS PARTNER

1-800-939-0439

(916)939-0262

Please circle #22 on reader service card.

Do you cringe everytime you see those strange error messages on your desktop? Wouldn't it be great to boot OS/2 much faster? Or how about cleaning up precious hard disk space by totally uninstalling OS/2 programs you never use?

Special Introductory Price!!
$79.95!!

UniMaint™

System Maintenance and Uninstaller for OS/2

UNINSTALLER

You have this program, you might have booted it up once. It's taking up disk space and cluttering up your desktop! You could delete the program from your disk drive but various system entries and associated files from the program are scattered EVERYWHERE! You sit back and breath easier now. You know you have UniMaint™. UniMaint will completely remove that program and ALL of it's associated files including system entries! All you have to do is simply drag the program icon to the UniMaint Uninstall icon and -- it is history.

SYSTEM PERFORMANCE

UniMaint improves system performance and desktop reliability by helping you maintain key system files. You now can easily maintain all those pesky INI files, including the OS2.INI and OS2SYS.INI files. You can also purge obsolete Workplace Shell entries and fully edit all INI files! What this means to you is FASTER AND MORE RELIABLE OS/2 PERFORMANCE!

SYS

INI

DESKTOP BACKUP AND PORTABILITY

Your pride and joy... The ultimate desktop layout. What a devastating blow it would be if you lost it! You can sleep easier now because UniMaint will backup and restore your desktop, and it's FULLY customizable!
Your desktop at home is the desktop of all desktops but that one at work really isn't up to par. You don't have the time to set down at the office and arrange that desktop to meet your standards. With Unimaint all you have to do is put your home desktop on a disk and take it to the office and WOW!! The ultimate desktop layout is now in the office. You have total desktop portability that is flexible and easy to use.

MULTI

[800] 944-3028
Outside U.S. & Canada [405] 947-8085

STS

fax [405] 632-6537

SofTouch Systems, Inc.
Workstation Division
1300 South Meridian, Suite 600, Oklahoma City, OK 73108-1751

Please circle #23 on reader service card.