Personal Systems

IBM's MAGAZINE FOR TECHNICAL COORDINATORS

MARCH/APRIL 1995

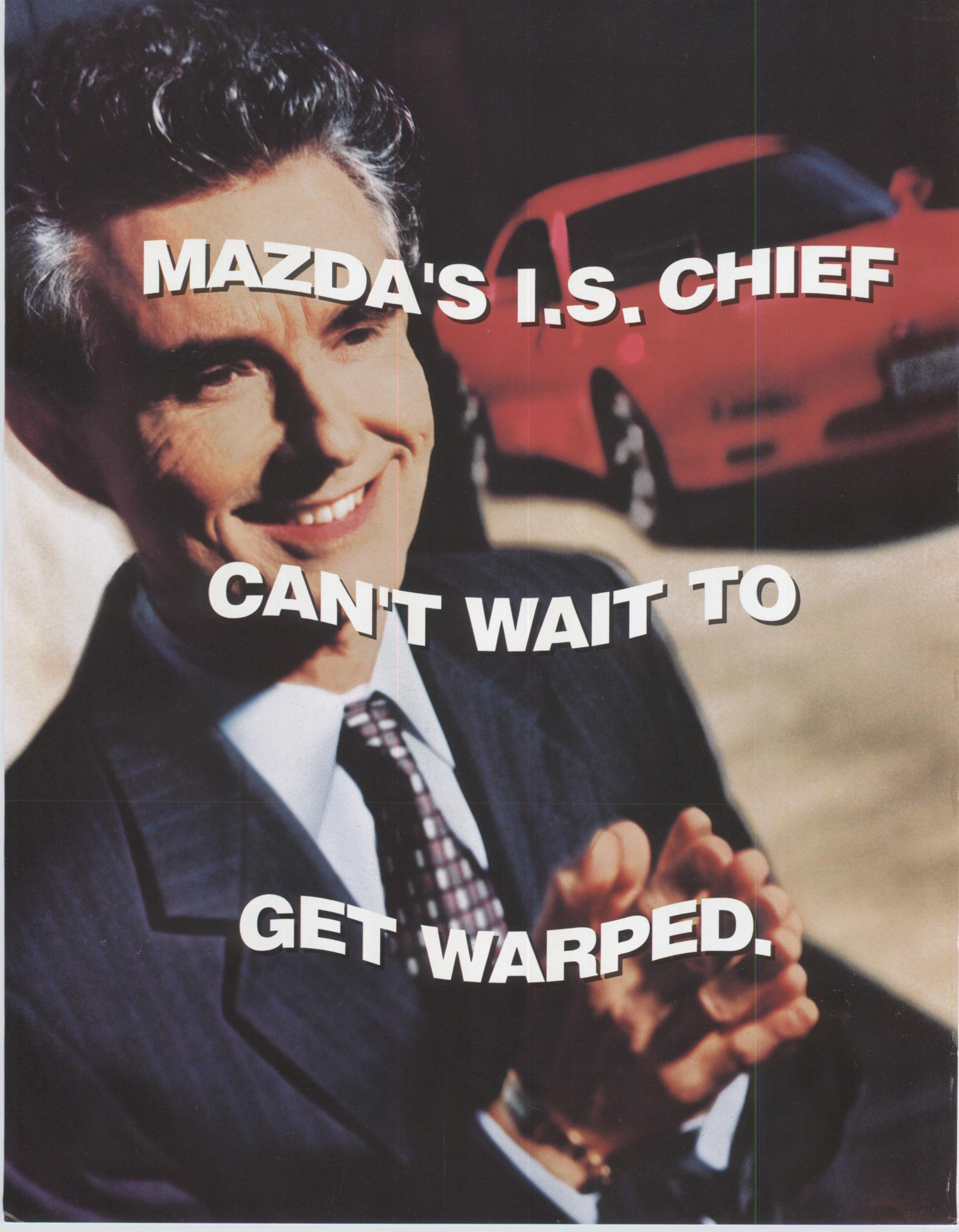
OS/2 Warp's Internet Access Kit: Door to the Universe

PC DOS 7 Enhanced Memory Performance

DeskMan/2 for OS/2 Warp

TCP/IP: How it Works

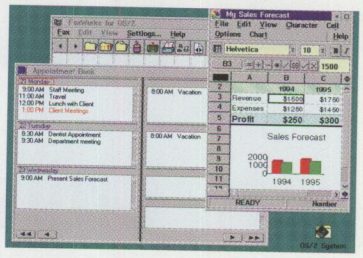
IBM


MAZDA'S I.S. CHIEF
CAN'T WAIT TO
GET WARPED.

OS/2 is in its third rev, so it's solid, stable, and mature.

Mike Anzis is the I.S. man behind the wheel of Mazda's computers. And OS/2® Warp is about to make his life easier.

From a basic 4MB laptop to a client/server system, the OS/2 family now scales an even wider variety of PC platforms.



The BonusPak gives mobile users the applications they need.

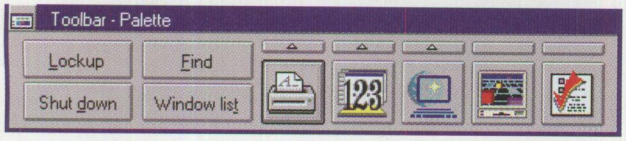
As Mike puts it: "We use OS/2 on our headquarters' client/server systems. It's also installed in our 900 dealerships. But until now, we haven't been able to get OS/2 everywhere we need it – on laptops in the field. OS/2 Warp changes all that." OS/2 Warp is the 32-bit, multitasking, Windows™-friendly way to run a computer. With simple installation and proven reliability, OS/2 Warp is a total computing solution that performs ruggedly at every level.

IT'S ONE MODEL THEY REALLY NEED ON THE ROAD.

And OS/2 Warp is a real communicator. With fax, Internet e-mail, and desktop conferencing, there isn't an easier way to keep those out on the road in the loop.

OS/2 Warp also offers Mike Anzis rock-solid reliability. "I know from years of experience with OS/2, I can trust it to keep performing. Now I can enjoy this peace of mind at every level."

OS/2 Warp is available for under $90. To get warped, stop by your local software dealer, or call 1 800 3 IBM-OS2. Ask for a free demo disk.



A toolbar gets users into their applications quickly and easily.

The new 32-bit, multitasking, multimedia, Internet-accessed, crash-protected, Windows-friendly, totally cool way to run your computer. **OS/2® WARP**

IBM®

OS/2 Warp is available from your software dealer. It's also available from IBM for $89 by calling 1 800 3 IBM-OS2.

Please circle #15 on reader service card.

Cruisin' on the Internet

I don't know about you, but all I hear these days is "Internet," "Internet," "Internet." The other night I saw that NBC Evening News now has an Internet ID—making it easier for all of us to voice our opposing opinions! Internet's e-mail capability has truly brought the world closer. Now its graphical capability is offering a whole new world to those of us in the publishing industry.

There are some of us who, even though we're surrounded by leading-edge technology, are just not going to accept a new concept until someone makes it easy for us. For instance, I wanted nothing to do with personal computers and the software that runs on them until OS/2 came along with Presentation Manager. (You see, the only version of Windows I was familiar with was 1.0, and I didn't see *anything* user-friendly about that!) Pictures! I want pictures! And menus! Don't make me remember those boring old DOS commands.

Well, that's how I felt about the Internet when I first heard about it. I envisioned a black screen with thousands of lines of white writing on it. And 75-character commands that I would have to remember. No pictures. No menus.

Then OS/2 Warp with the Internet Access Kit (IAK) came along.

Phil Lieberman, owner of Lieberman and Associates and frequent contributor to *Personal Systems*, was telling me how great the IAK was, so I said, "Show me. Better yet, show *Personal Systems*' readers!" And that's just what Phil does in this issue. Using his friendly, easy-to-understand writing style that we've come to know and love, Phil takes us through the IAK from the first mouse click to the inner depths of the World-Wide Web. He even makes some of those 75-character commands look friendly. And for you techno-addicts out there, Phil also elaborates on TCP/IP, the communications technology behind the Internet.

Then Van Landrum, *Personal Systems*' database manager, came to me and said, "Let's get *Personal Systems* on the Web!" Now, thanks to a lot of creative work by Van, you'll find selected articles from each issue on Internet's World-Wide Web, located at (Watch out! 75-character command coming up!) `http://pscc.dfw.ibm.com/psmag/index.html`. Or, just go to the IBM Home Page and click on our name under Publications. We've got articles, back issue indexes, and subscription information. You can also link to other IBM product information right out of our articles. Check it out! It's got pictures *and* menus!

DOS Revisited

Speaking of pictures and menus—PC DOS 7 brings a whole new look to this tried-and-true operating system. If DOS had looked like this back in the old days, I might not have been so hesitant to get into this personal computer stuff. DOS product planners, Kay Lee and Nestor Miranda, teamed up to tell you all about the new features in PC DOS 7. Its new memory management capabilities alone make it worth a try.

As technology continues to expand our horizons, we find our world growing smaller. More information is available to more people than ever before. The *USA Today* reports that half of all US households will have a computer by 1997. And the Internet, with millions of users already, is adding new users at a phenomenal rate (in 1994, an average of over a million new users *per month*). So if you're not already cruisin' on the Information Superhighway, you're in danger of being left behind in a cloud of cyberdust!

Betty Hawkins

Betty Hawkins, Editor

Big Blue IN THE Big Easy

Next to Mardi Gras and the cool sounds of the Jazz Festival, what's the hottest event happening in New Orleans? It's the **1995 IBM Technical Interchange**.

And **it's the place you need to be** if you answer "yes" to any of the following questions: Are you interested in learning about the **latest OS/2, AIX, AS/400, and MVS offerings**? Do you want to know more about **object-oriented technology** and how it is defining the next generation of operating systems and applications? Are you ready to build your **technical knowledge and skills**?

You'll be in **good company**—software designers, independent and corporate developers, MIS managers, technical coordinators, software integrators, LAN experts, device driver developers, consultants, dealers, value-added resellers, and training executives.

A **big time in the Big Easy**—learning, exploring, networking, and having fun—what more could you ask? From the **Ragin' Cajun Conference Kick Off** to the **Mardi Gras Celebration**, we'll provide lots of excitement from start to finish, all while offering you plenty of opportunities to network and enhance your technical skills. It's all happening soon at the **1995 IBM Technical Interchange** in New Orleans.

Save $200.00—Register Early!

Register before April 7, 1995 and save $200.00 over the regular $1095.00 registration fee.

Save even more—Register 3 or more people from the same company* before April 7, 1995 and pay only $795. After April 7, 1995 pay $995.00.

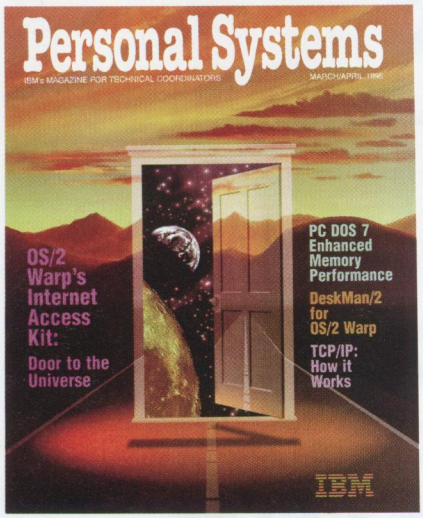
*** You must register at the same time and pay in full.**

1995 IBM Technical Interchange

New Orleans Ernest N. Morial Convention Center

May 21-25, 1995

To register or to receive more information,
call **(800) 872-7109** (in the US) or **(800) 661-2131** (in Canada)
Outside the US and Canada, call **(508) 443-4990**



ABOUT THE COVER
Noted Dallas artist Bill Carr takes us down the highway of reality, through an illusory door, and beyond to a vast universe of possibilities. Internet provides this superhighway to information, and OS/2 Warp's Internet Access Kit provides the door to the world of Internet.

Editor and Publisher
Betty Hawkins

Business Manager/ Editorial Coordinator
Donna Su

Editorial Assistant
Lia Wilson

Database Manager
Van Landrum

Subscription Coordinator
Rose McAlister

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

Manager
Dan Russell

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 address changes to *Personal Systems* Subscription Coordinator, IBM Corporation, Internal Zip 40-B3-04, One East Kirkwood Blvd., Roanoke, TX 76299, fax to (817) 961-7218, or call Rose McAlister at (817) 961-7008. The editor, Betty Hawkins, can be reached at the above mailing address, (817) 961-7525, Internet bhawkins@vnet.ibm.com, or OV/VM HAWKINSB at DALVM41B. The business manager, Donna Su, can be reached at the above mailing address, (817) 961-6268, Internet psts@vnet.ibm.com, or OV/VM DONNASU at DALVM41B.

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.

What's New For OS/2?

TAXDOLLARS 1994 for OS/2 Warp

There's still time to get your free copy of ***TAXDOLLARS 1994***, the only tax planning and preparation software program available for OS/2 users. The first 250,000 callers to take advantage of BT&T Consumer Technology's free offer will find that TAXDOLLARS 1994 makes tax planning and filing returns easier than ever before.

Taking a bold stand against tax planning software on other platforms, TAXDOLLARS 1994's friendly interface follows Form 1040 line-by-line, using an automatic link to complete other required forms or schedules. Its automatic calculation feature makes continuous updates so you know the bottom line at all times.

If you use a tax planner or CPA, TAXDOLLARS 1994 saves valuable time and reduces consulting fees dramatically. It includes the forms needed for salaried employees, homeowners, investors, small or home-based business owners, and those seeking tax credits.

BT&T Consumer Technology is dedicated to delivering simple-to-use, intuitive software for personal finance on the OS/2 operating system. To take advantage of this free offer, call (800) OS2-TAXDollars (800-672-8293). A $7.95 non-refundable shipping and handling fee is required for each order.

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

Compression Plus! for OS/2 Warp

In conjunction with IBM's latest OS/2 release, OS/2 Warp, Proportional Software announced ***Compression Plus!***–a complete on-the-fly data compression starter kit for users of the OS/2 operating system. Compression Plus! combines the newest release of Proportional Software's innovative ***Disk Compression Facility for OS/2 (DCF/2)*** bundled together with their ***Virtual Disk Samplers*** in a single package. The software is designed to get you productive quickly and easily and to allow you to begin recovering hard disk space immediately.

Redesigning this data compression utility for the multithreaded, multitasking power of OS/2 resulted in a product that is both robust and versatile. With DCF/2, you can now control which programs and data files are moved to compressed storage without totally committing to compressed files. DCF/2 does not change your hard drive in any way.

The Virtual Disk Samplers each offer a ready-to-use, compressed drive filled with a variety of publicly available objects and utilities such as icons, bitmaps, OS/2 utilities, and IBM employee-written software. The Samplers in Compression Plus! include:

- OS/2 Icons–Collection I
- OS/2 Bitmaps & Backgrounds
- OS/2 Fun Stuff
- OS/2 Must Have Utilities
- OS/2 Convenience Utilities

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

Graphical File Manager for OS/2 Warp

SofTouch Systems' ***FileStar/2 1.0 for OS/2 Warp*** is a full-featured Presentation Manager (PM) file manager that lets you gain control of your hard drive.

With FileStar/2, you can view available drives, disk directory trees, and data files while moving conveniently between directories to copy, move, rename, and delete files. Save precious disk space by archiving files in the popular ZIP format using the ***InfoZIP*** archiving products included with this release. You can display archived files in a ZipView window and select them for browsing or decompression. Current information on file sizes, drive free space, and the OS/2 swap file is constantly displayed to keep you informed of your disk status.

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

VX-REXX Client/Server Edition

Watcom's award-winning visual development environment for OS/2, *VX-REXX*, is now available in a client/server edition. The new Watcom ***VX-REXX 2.1 Client/Server Edition*** includes the powerful features of VX-REXX Standard Edition plus it lets you develop client/server applications using the powerful new connection, query, and chart objects:

- *Simplify database development*–The database connection and query objects make GUI client/server application development simple and approachable. With these objects you can quickly build windows to display and update data from a wide range of database systems. The package also includes a database administrator that lets you easily create new databases, define tables, and set the column types and relationships.
- *Support for DB2 and more*–VX-REXX Client/Server Edition makes it easy to connect to a wide range of databases: DB2/2, DB2/6000, DB2 for MVS, DB2/400 for AS/400, DB2/VSE and VM (SQL/DS) for VM and VSE, plus Watcom SQL and ODBC-enabled databases.
- *Visualize your data*–Create more than a dozen types of charts with the chart object. Customize your charts from 150 display options including chart type, 3D view, and titles. Charts can be bound directly to SQL queries, making it easy to create visual client/server applications.

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

Guidelines Desktop for OS/2 Warp

JBA International, a leading software and development tools supplier, confirmed its commitment to OS/2 Warp with the release of ***Guidelines Desktop***, a graphical, event-driven application development tool for OS/2 developers.

Guidelines Desktop is designed to protect your existing corporate investment in data, staff skills, and systems. Its range of features embraces the spectrum of the development scale, from stand-alone applications to enterprisewide corporate systems engineering. It utilizes an object-oriented high-level language called JOT with full point-and-click code prompting.

Guidelines Desktop does not require you to learn the complex syntax and rules associated with C and C++. At the same time, if you want to utilize the full power of C++ and native, platform-specific application programming interfaces (APIs), you can call on the language directly whenever required.

Guidelines Desktop also allows you to embrace recent developments in object orientation. The product can fully utilize existing system object model (SOM)-based objects written for either IBM OS/2 2.x or Microsoft Windows. The platform-independent nature of JOT, generating C++ code, means no change needs to be made to Guidelines Desktop source code, regardless of the platform in use.

The product has a comprehensive and extensible range of controls including support for IBM CUA '91 objects (containers, notebooks, sliders, value sets), status bars, tool bars, business graphics, animated pushbuttons, picture fields, and formatted entry fields. To aid screen design, Guidelines Desktop also provides a range of layout tools such as horizontal/vertical control alignment, object spacing/sizing, and snap-to-grid.

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

VisPro/C and VisPro/C++ for OS/2

HockWare Incorporated has extended their visual programming offerings with the announcement of ***VisPro/C*** and ***VisPro/C++*** for OS/2 2.x.

VisPro/C and VisPro/C++ allow you, as an application developer, to quickly develop OS/2 GUI programs. Both products have extensive drag-and-drop programming capabilities; support an abundance of user interface programming objects; are fully Workplace Shell integrated; provide a built-in visual DB2/2 database designer; generate high-quality, non-proprietary code; provide a custom object builder that is based on the IBM SOM; and provide many other features that allow you to develop 32-bit, multithreaded, client/server applications.

Some of the functions in VisPro/C and VisPro/C++ include:

- Workplace Shell-enabled drag-and-drop programming that automatically generates code statements.
- Comprehensive set of CUA '91 objects supported, from simple buttons to the robust notebook object.
- Multiple development views, including Layout, List, Event Tree, and Settings.
- Integrated environment for compiler.

- Quality code generation.

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

Swapper Plus! for OS/2

A new utility software program, ***Swapper Plus!***, is available from Proportional Software for OS/2 Warp. Swapper Plus! offers you a combination of compressed virtual disk, RAM disk, and swap file manager. With this product installed, you'll find that your OS/2 swap file requires less physical disk space, performs with the speed of a RAM disk, and can be spread over multiple physical and logical (even network) drives. Now you can exploit your system-specific hardware like PCMCIA flashcards as you allocate swap file space.

Benefits of Swapper Plus! include:

- *OS/2 system swap space caching*–Prevents disk thrashing commonly caused by swap file I/O.
- *OS/2 system swap space compressing*–Decreases the amount of physical space required for the swap file and compresses in the background–minimizing the impact of compression on system performance.
- *Swap file customizing*–Spreads the swap file over multiple physical, logical, or network disks.
- *Fail-safe mechanism for swap file overflow*–Prevents OS/2 system crashes caused by insufficient swap space.

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

Pegasus Resource Monitor for OS/2

Pegasus Resource Monitor for OS/2, from C.O.L. Systems Inc., is a kernel-level, real-time, Presentation Manager tool for monitoring system and application resource usage under OS/2 2.x. Including a configurable sample rate, performance threshold setting and alarms, working set, thread detail display, kill process options, as well as real-time graphical indicators and long term data logging for post-processing analysis, the Pegasus Resource Monitor provides the most comprehensive monitoring tool for OS/2 systems.

Unlike other monitors, the Pegasus Resource Monitor, ***Pegasus AppMon***, ***Pegasus FileMon***, and ***Pegasus CacheMon*** utilities work with information retrieved directly from the OS/2 kernel. This is a superior, less intrusive, and less resource consumptive collection method compared to other performance monitoring products.

C.O.L. Systems recently announced lower prices and new distribution methods for Pegasus. Although C.O.L. Systems does not offer direct sales, Pegasus is available from the following sources:

- Peter Norloffs OS/2 BBS at (703) 385-4325
- Indelible Blue, Inc. at (800) 776-8284

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

ATS for OS/2

ATS for OS/2 Version 3, from MHR Software and Consulting, is a very powerful task scheduling tool for OS/2 2.0 and above. With ATS, you define what type of programs you want to run, when you want your programs to run, and how they should be run. No matter how complicated your scheduling needs, ATS can handle them.

ATS for OS/2 provides you with a strong, stable, and configurable job scheduler with the power to execute native OS/2 programs based upon various criteria. Version 3 includes the ability to start DOS and Windows programs, job queues, and additional event types; enhancement of existing event types; increased user configurability; restarting and requeuing of tasks; enhanced user interface; and enhanced manual signaling.

At the heart of ATS are three entities: events, tasks, and job queues. Events must happen before a task can be started. A task definition contains all of the information required to start a program as well as the set of criteria defining when the program can be run. A job queue is used to control the use of resources and also to restrict a flood of tasks from being started at the same time.

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

Comprehensive Security Offering for OS/2

Pinnacle Technology, the maker of OS/2's premier security offering, announced its agreement with M&T Technologies, Inc. to bring additional function to ***The Desktop Observatory***. Code named ***"C2-4-OS/2,"*** the new offering combines security, network, and management function currently found in Pinnacle's The Desktop Observatory with the access security found in M&T's ***MicroSAFE*** products.

General features include:

- *Access security*–Inhibits all access to disk data when invoked.
- *Application security*–Prevents user access to specific applications, commands, and objects.
- *Reporting*–Logs, reports, and initiates specific events.
- *Managing*–Centrally maintains and manages passwords and desktop profiles in a file that is less than 10 KB.
- *Networking*–Works over any network operating system including Novell, IBM, Microsoft, Banyon, DCE, and dial-in implementations.

The Desktop Observatory is one of the few products already tested by IBM and Pinnacle for its ability to run on the forthcoming PowerPC.

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

LAN Intensive Care Utilities for LAN Server

LAN Intensive Care Utilities (LAN ICU) for IBM LAN Server 3.0 and 4.0, recently announced by Lieberman and Associates, is the first product specifically designed to administrate, diagnose, and repair problems within IBM LAN Server.

The package includes the following utilities:

- *Exporter and importer*–Allow users, groups, aliases, public applications, and access controls to be moved from domain to domain.
- *Analyzer*–Using artificial intelligence rules, checks all user accounts and repairs problem accounts. The analyzer implements a corporatewide security and resource assignment policy based on group memberships and operator privileges.
- *Conversion*–Allows Domain Definition Reports from LAN Server 1.0 and 2.0 to be used to build a new LAN Server 3.0 or 4.0 domain.

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

Mesa 2: Gaining the Competitive Edge with OS/2

By Jeri Dube and David Pollak

This article describes the technical and business reasons that made Athena Design choose the OS/2 operating system for expanding the Mesa market.

The real achievers in this world start looking for the next mountain to conquer as soon as the vista of the mountain they are currently climbing is within sight.

After achieving significant penetration in the NextStep spreadsheet market, Athena Design was ready to expand its spreadsheet application, Mesa, to other platforms. Wanting to enter a market where it could deliver value by enabling its customers to meet their goals, Athena Design focused on being customer-centric rather than being solely profit-centric. From this business philosophy derived the technical requirements that would allow Athena to create an innovative and quality spreadsheet product. The criteria for this new platform were to provide:

- 32-bit support
- Multithreading
- Preemptive multitasking
- Object support
- A robust imaging model

OS/2: The Next Mountain to Conquer

The choice for Athena was simple. OS/2 offered the technical features needed to deliver spreadsheet innovations in a market that was ripe with opportunity for a growing independent software vendor. In the NextStep world there was a clear niche for a traditional spreadsheet application. In the OS/2 world, however, which included DOS/Windows applications, the spreadsheet playing field was crowded with strong incumbents. It was obvious that Athena needed to differentiate the product from other spreadsheets and provide a smooth migration path from the incumbent spreadsheets to Mesa 2.

Platform for Attacking Competitive Deficiencies

Upon close investigation, definite areas of deficiency in the incumbent spreadsheets became apparent. These areas included:

- Simultaneous recalculation of multiple spreadsheets
- Variety and flexibility of graphing options
- Interoperability and integration of spreadsheet functions with other applications
- User productivity features

Using OS/2 as the platform for Mesa 2, Athena could attack competitors' deficiencies with innovation and energy while relying on OS/2 to provide the base level functionality.

Exploiting Multitasking and Multithreading

Many spreadsheet users need to work with more than one spreadsheet at a time. Other users require the real-time updating of spreadsheets from real-time data feeds such as stock market data or medical instrument monitoring. Therefore, Athena included these capabilities in Mesa 2.

With OS/2's preemptive multitasking and multithreading features, Athena could fulfill these requirements in a straightforward manner. Since the OS/2 operating system provides multiple threads, Athena did not need to clutter its application code by writing functionality that should be in operating systems and is in OS/2. All Athena programmers needed was to assign each open spreadsheet a thread. This multithreading feature allows Mesa 2 users to do other functions while long operations, such as real-time updates, take place. If required, Mesa 2 will alert the user if the updated spreadsheet meets some condition during the stream of updates.

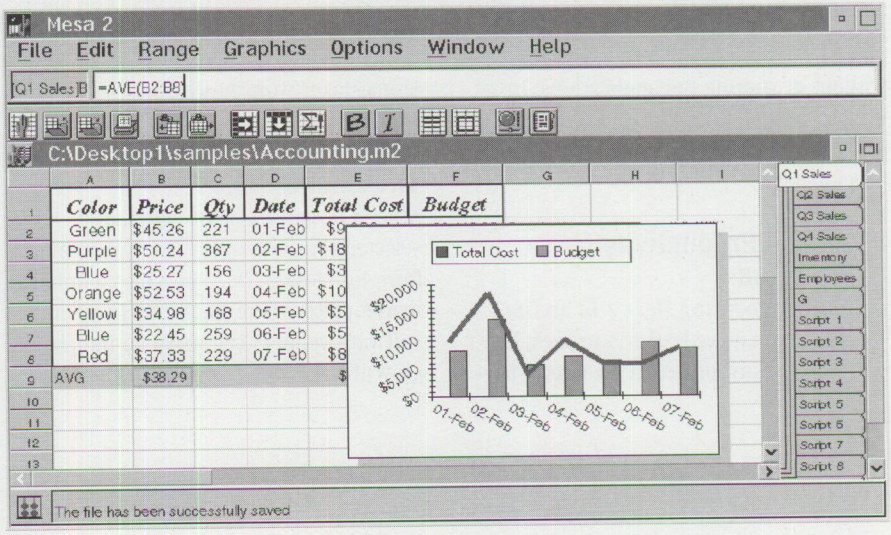


Figure 1. Text Placement Flexibility with Mesa 2

Getting More Graphic with a Richer Model

While it seems that there is an infinite variation in hardware configurations for Intel machines, independent software vendors (ISVs) have a finite number of development resources. An operating system like OS/2 that provides a rich imaging model allows an application programmer to focus on writing good code and creating great graphics. OS/2's graphics programming interfaces (GPIs) allowed Athena to concentrate on the graphics and not the graphics adapters. The GPI calls work the same on the printer, so Athena had only to write and debug one set of display routines rather than writing separate code for the display and for the printer. In this way, Athena used the OS/2 GPI to provide arbitrary scaling and rotating of fonts and objects. This means that Mesa 2 users can place text labels in any position that enhances the look and readability of their graphs (Figure 1).

Integration with Objects

From its experience with corporate customers' NextStep environments, Athena Design discovered the need to combine commercial spreadsheets with homegrown corporate applications. For example, many brokerage houses integrate a spreadsheet application with real-time data feed into their stock trading application. Athena facilitates this kind of work by shipping an object library with every Mesa 2 spreadsheet. With OS/2's superior object environment, including the system object model (SOM), Mesa 2 users can easily incorporate the same building blocks that Mesa 2 uses. Furthermore, in SOM-based applications, new, upgraded Mesa 2 objects can easily replace older versions without disrupting the applications. In this way, Athena Design has extended its customers' toolset for their SOM-based development.

Effortless Productivity Advantages

In general, producing Mesa 2 in an OS/2 environment allowed Athena to include productivity advantages in its spreadsheet product with very little effort. For example, by taking advantage of the standard OS/2 Workplace Shell features, Mesa 2 can support dragging and dropping colors and fonts into a worksheet. OS/2's strong object-oriented environment facilitated the development of Mesa 2's *formula inheritance*. That is, formulas in a spreadsheet can be dynamically linked so that when a formula is changed in a linked cell, all linked cells automatically inherit the new formula, eliminating the need to copy and paste.

Since REXX scripting is standard with OS/2, Athena incorporated REXX into Mesa's design. This feature lets Mesa 2 interact with other OS/2 REXX products. Furthermore, Mesa users who build spreadsheets and models for others can take advantage of the REXX scripting to make their spreadsheets usable and productive for less experienced users.

A Facilitating Platform

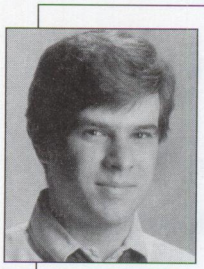
From a technical point of view, OS/2 is the consummate operating system for facilitating development. With OS/2, developers can concentrate on developing their applications rather than on overcoming the deficiencies of an operating system. The 32-bit linear addressing model eliminates the need for programmers to worry about using more than 640 KB of RAM. Also, OS/2's built-in help system makes creating help screens a simple task that helps satisfy usability requirements. Furthermore, Athena Design found a tool that converts files written in FrameMaker to the information presentation facility (IPF) files used in the OS/2 help system. This was useful to Athena because its user documentation was all written using the FrameMaker word processor.

OS/2: The Right Choice . . . The Best Choice

Athena Design feels strongly that it selected the right operating system platform for Mesa 2. OS/2 provides a unique marketplace where small independent software vendors can make an impact and grow their businesses. The field of OS/2-exploiting applications is far from being saturated, yet the OS/2 volumes are significant and are growing every day. With OS/2 Warp's availability, it is clear that IBM has made a serious commitment to the long-term success of OS/2. For the application developer, now is the time to produce an OS/2 application while the field is wide open and the potential is great.

For more information about Athena Design's Mesa 2 product, circle 11 on the Reader Service Card.

Jeri Dube is a planner responsible for worldwide communications in support of marketing and selling OS/2 Warp. Her first assignment for IBM in Boca Raton, Florida was on the original planning team for OS/2 for the PowerPC. Before moving to Boca Raton, Jeri worked as a systems test planning manager, a customer relations representative, and an internal manufacturing efficiency consultant for IBM's Large System Division in Poughkeepsie, New York. Jeri also worked as an IBM marketing representative and a programmer focusing on evaluating the performance of IBM storage products. She received a Master's degree in Operations Research from Georgia Tech.



David Pollak is currently president of Athena Design, which he incorporated in 1989 after seeing the NeXT computer. Athena's first product was CultureShock, an addictive game written for the NEXTSTEP platform. Athena began developing Mesa 2 in January, 1994. David holds a Law degree from Boston University School of Law and a Bachelor's degree in Psychology and Economics from Rhode Island College.

Managing the Workplace Shell with DeskMan/2

By Gary Kushner

For DOS there is Norton Utilities: a software utility package so useful you can't imagine working without it. For OS/2, there is now DeskMan/2: a suite of utilities designed to manage OS/2's Workplace Shell (WPS). This article reveals how DeskMan/2 lets you create customized desktops, back up and restore those desktops to the desired configuration, and manipulate windows and objects in the Workplace Shell in a virtually unlimited number of ways.

Development Technologies' (DevTech's) latest release of DeskMan/2, Version 1.51, supports IBM's latest version of OS/2, OS/2 Warp. Enhancements include support for OS/2 Warp's animated icons and launch pad, as well as increased functionality of the original elements of DeskMan/2.

DeskMan/2 is not a monolithic program; it actually comprises four components: DeskMan/2, DM2Image, VUEMan/2, and WPS extensions (see Figure 1). These components extend the existing functionality of the WPS, plus they add new features such as protection, backup, and virtual desktops. These features allow you a level of control over the WPS that used to be possible only through programming.

DeskMan/2

Before there was DeskMan/2, there was Blackhole. Sci-fi aficionados know that the black hole can and will destroy anything that ventures into it. That was the intent of Blackhole, a freeware program designed before DeskMan/2 by one of DeskMan/2's authors. Blackhole, the software utility, acted like a super shredder—drop *any* object on it, even objects with an attribute of NODELETE, and it was gone forever. Although the function is no longer called Blackhole, one of the many features of DeskMan/2 is the ability to delete any object from a system.

There are several reasons why you might want to delete an object with an attribute of NODELETE. The Shredder object can be an unforgiving tool. You may not want to give your novice end users the chance to accidentally shred an important object or folder. The Shredder object, however, is an undeletable object. DeskMan/2 will delete the Shredder object from any OS/2 desktop.

Other instances where you might want to delete undeletable objects include those objects that may have become corrupted by the system, application objects designated as undeletable that you don't want on your system, and CD-ROM files that are installed as read-only files, which can't normally be deleted.

DeskMan/2's components not only allow individual users to get the most out of OS/2, they also allow system administrators and technical coordinators to standardize, customize, and manage desktops for their end users. DeskMan/2 looks and acts like a very special folder object. You can create a backup copy of an object by dragging it into the DeskMan/2 folder, or you can back up all of the desktop objects at one time. Later you can restore any one group or all of the objects back to the desktop. Because DeskMan/2 uses documented WPS methods to back up objects, you can restore objects across versions of OS/2 or even to different computers.

With DeskMan/2, you can create a standard desktop to be used throughout your organization. One computer can be configured with the standard desktop; then, using DeskMan/2, you can *clone* that standard desktop to other computers. You can move desktops between versions of OS/2 by saving your current desktop, installing the new version of OS/2 to a freshly formatted partition, then using DeskMan/2 to restore the desktop. Installing new

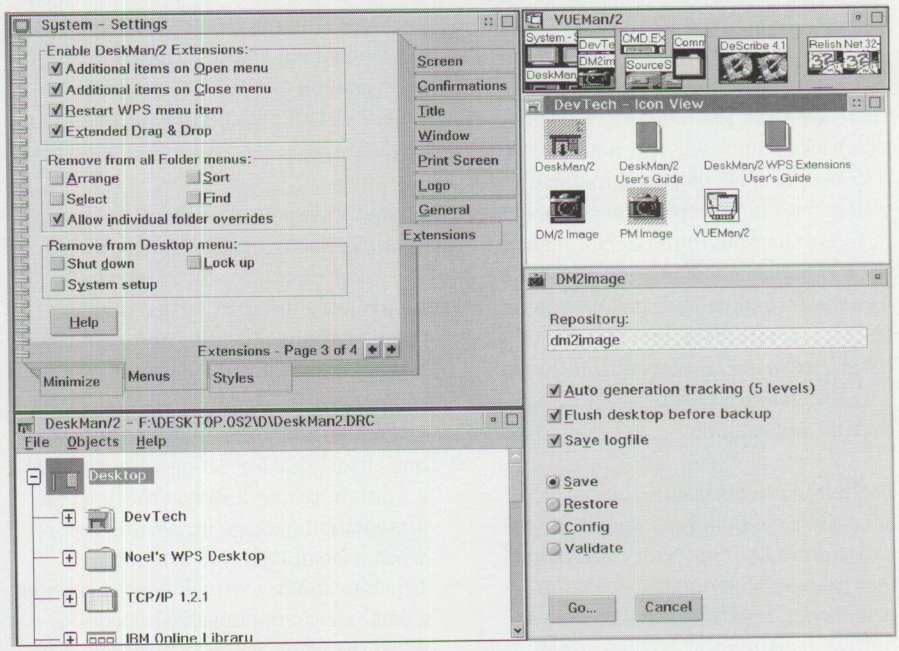


Figure 1. DeskMan/2's Four Components

versions of OS/2 to a fresh partition ensures that remnants from the previous version are not left behind to cause problems.

Although, as a system administrator, you usually want to have complete control of a system, it is frequently necessary to limit end users' control. DeskMan/2's safety features allow you to selectively turn off any of DeskMan/2's features.

Other DeskMan/2 features include:

- The ability to directly modify any object's attributes, such as `NODELETE`.
- The ability to query and modify object settings, including class type and ObjectID.
- A complete command line interface that allows sophisticated manipulation of WPS objects from the command line or REXX programs.

DM2Image

While DeskMan/2 creates an object-by-object backup of a system, DM2Image creates a snapshot backup. A *snapshot* is a collection of all the WPS objects in the system, plus the `OS2.INI`, `OS2SYS.INI`, `CONFIG.SYS`, and (if they exist) `STARTUP.CMD` and `AUTOEXEC.BAT` files, which are compressed into one file called a repository. If WIN-OS/2 is installed on the system, DM2Image also stores the WIN-OS/2 desktop in the same repository file. The repository file can then be used at any time to restore a computer's desktop back to the point when the snapshot was taken.

There are some limitations to the snapshot backup approach. It is not possible to restore only some objects; a complete system must be restored. You can only restore a snapshot to the same computer (or another computer with the *exact* same hardware) running the same version of OS/2. Even with these limitations, DM2Image is the quickest way, in case of a total system meltdown, to get a system back up and running.

DM2Image can also be used as an administrative tool. For example, if you have a training room with workstations that need to be periodically reset for new classes, DM2Image can make that process as painless as possible. Or if your organization has a standard hardware configuration for workstations, you can use DM2Image to quickly set up a standard desktop and `CONFIG.SYS` file on new machines.

There are two versions of DM2Image: a Presentation Manager (PM) version and a command line version. The PM version has a nice graphical user interface (GUI) and is good for day-to-day operations. As long as your system is running, the PM version can be used to restore a desktop to a previous snapshot state. But in case of a real emergency, you can boot your system from floppy diskettes and use the command line version to restore the desktop.

DM2Image is highly customizable. It can be configured to automatically create and track up to 99 generations of snapshots. In addition, any other files, such as LAN configuration files that should be backed up along with the system files, can be automatically compressed and added to the repository whenever you run DM2Image.

VUEMan/2

In a multitasking operating system like OS/2, desktop real estate is always at a premium. VUEMan/2 is a Presentation Manager program that greatly expands desktop real estate by providing up to 81 virtual screens. The VUEMan/2 program window is a graphical representation of each virtual desktop and all of the windows that are open in each of the virtual desktops. You can, at any time, view the contents of any of the virtual screens on your monitor by just clicking the mouse on the virtual desktop to which you want to jump.

A great way to use VUEMan/2 is to divide your daily activities into categories. As the day progresses you can switch between the different virtual desktops running all the programs pertinent to that task. It's almost like having a different computer for each task. For example, if you are an accountant, you might have one virtual desktop set up with copies of a word processor, a spread sheet, and a presentation package for writing reports. When a client calls, you can quickly switch to another virtual desktop running a database containing client portfolios. When the phone call ends, you can switch back and continue the report right from the point where the telephone rang. If you are a developer, you might have one virtual desktop set up for development and another for running the program just as if you are the end user.

VUEMan/2 has a feature called a layout to aid in setting up virtual desktops for particular tasks. A layout stores the location and size of a group of windows within one or more virtual desktops. You can set up virtual desktops, then save the setup as a layout. The next time you want to use your computer in the same way, you can start all of the applications together–by putting shadows of their program objects into a work area folder–and then start VUEMan/2. VUEMan/2, using the saved layout, arranges the programs and positions windows, placing each program in the specified virtual screen.

Another VUEMan/2 option is sticky windows. Sometimes you want to have one window (for example, the system clock) show up in every virtual desktop. VUEMan/2 allows up to six windows to be specified as *sticky*. These windows stick with you as you jump from one virtual desktop to another. If you specify the clock as sticky, it will always be present on the active virtual desktop.

Each virtual desktop is represented in the VUEMan/2 program window. You can use the VUEMan/2 program window to directly manipulate the virtual desktops. Using the mouse, you can drag a window from one virtual desktop and drop it onto another, making the job of virtual desktop management easier to do than to say!

WPS Extensions

The WPS extensions are a tightly integrated set of enhancements to many aspects of normal Workplace Shell operation, from securing objects to enhancing drag-and-drop.

Security

Many levels of security are possible with the WPS extensions. A global password can be applied to all folders or all objects. Then before a user is allowed to open a folder or run a program, the user must type the password in a dialog box. If a global password isn't desired, individual folders or objects, even the main desktop folder, can be protected with its own password. If the desktop folder is protected,

a dialog box asking for a password appears as the system boots. The password must be entered before the system can finish bringing up the desktop.

The WPS extensions can provide another level of protection by restricting normal WPS functions. Many standard object menu items (for example, Arrange, Sort, Open, or Settings) can be removed from all folders globally or on a per-object basis. Shut down, Lockup, and Open System settings can also be removed from the desktop menu. In addition to removing menu items, you can limit the type of operations the WPS allows for all objects, for example, the ability to delete or to drag objects.

By using the WPS extensions to password-protect folders, you can protect sensitive information from casual inspection. You can also protect system administration utilities by placing, for instance, the LAN adapter and protocol support (LAPS) and LAN Requester configuration utilities in a password-protected folder. And you can protect all command line session program objects as well as the system setup and template folders.

By turning off the ability to delete, move, or create objects as well as password-protecting all folders, you can create a desktop where the only operations a user can perform are those you select. Casual users can run programs only from objects you designate.

Evolution

Security is only one aspect of the WPS extensions. DeskMan/2 enhances the usability of the WPS. Two ways in which it does this include:

- The ability to create shadows or copies of objects using the mouse without holding down keys on the keyboard.
- The ability to minimize a folder when a program is run or another folder is opened from it. With DevTech's WPS extensions, minimize-on-use is a property of an object, so individual objects within a folder can be set differently. For example, you can set up an application folder so that opening the documentation file does not close the folder, but running the application does. However, OS/2 Warp's minimize-on-use option is a property of the folder, so all objects in the folder behave the same way.

Some other features of the WPS extensions include:

- The ability to save virtual input/output (VIO) window positions, so text mode programs open in the same location each time they are run.
- A menu item to open a folder's parent folder.
- An option to open or close all of the folders in a chain.
- The ability to restart the WPS if it becomes unstable.

All of the individual features of the WPS extensions can be enabled or disabled.

The Future

Right now, DeskMan/2 is one of the few software packages so useful that every OS/2 user should have a copy. The next version looms on the horizon. DevTech is keeping tight-lipped about what new features might appear, but some hints have slipped out:

- A more consistent user interface and tighter integration between the different utilities.
- A *software bus* where other DevTech or even third party modules might be able to plug in and share the user interface.

But whatever comes with the next version, if the present is any indication of the future, one thing is for sure: it's going to be fun!

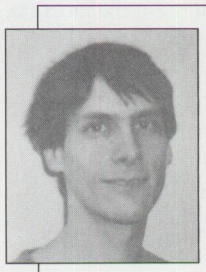
Versions

The current version of DeskMan/2 is 1.51. You should upgrade to this version, especially if you are using OS/2 Warp. DevTech has, so far, the unusual policy of not charging for upgrades downloaded electronically. Electronic support is officially provided on CompuServe, IBMLink's OS2BBS, and America OnLine. You can download the patch files from any of these sources as well as from the Internet FTP site: `hobbes.nmsu.edu`. Support is also available though phone and fax. If you do not have access to these online services, you can obtain the patch files by mail from Indelible Blue, Inc. for a minimal fee. Call (800) 776-8284 or fax your request to (919) 878-7479.

A hard copy manual is now available to current users and can be ordered through existing vendors.

For further information, contact Development Technologies, Inc., 308 Springwood Road, Forest Acres, SC 29206, or circle 12 on the Reader Service Card.

Phone: (803) 790-9230
Fax: (803) 738-0218
IBMLINK: DEV0691
CompuServe: 71333,3362
Internet: 71333.3362@compuserve.com
America OnLine: DevTech.



Gary Kushner has worked in the computer industry as a programmer, consultant, and educator for over 15 years. He has used OS/2 as a development platform since version 1.1. Currently he is involved in a new company that plans to develop applications for OS/2.

Circus du COMDEX: The Running of the Geeks

By Todd L. Watson

Todd Watson,* Personal Systems*' sometimes "lite" writer, relates the saga of his trek to the mother of all computer shows, COMDEX.

Being a journalist who writes mainly about computers, technology, and other (not necessarily related) topics, I always figured going to COMDEX would be like making a peregrination into technology's innermost soul, a heavy trip into the Lost Wages desert night, which I envisioned might be replete with glorious, life-changing visions about the cosmos, computers, and the purpose of my electron-centered existence in this universe. Barring that, I figured I'd at least take in a couple of good floor shows.

So when the opportunity arose for me to cover an IBM press conference (and do some general milling about on the show floor) at this past fall's mother of all trade shows, I jumped at the opportunity.

What the *$*%&#!* was I thinking?

First off, I feel as if I'm still waiting to go to COMDEX, and this despite the fact that, as I relate this saga, I've been back from there for three weeks now. My experience defies description, but as a caution to prospective, future desert travelers, I'll chance the possibility of PTCD (post traumatic COMDEX disorder) and make a valiant attempt at describing my experience there.

COMDEX is very similar to that Samuel Beckett play, *Waiting for Godot*, except that instead of waiting for a deity to arrive, at COMDEX one waits for everything else—bathrooms, telephones, buses from the hotel to the Lost Wages Convention Center, a table at that okay restaurant at Caesar's Palace, taxis, telephones, buses from the Lost Wages Convention Center back to the hotel, telephones, a Big Mac at (heaven forbid) the glitzy McDonald's on the strip, buses from the Lost Wages Convention Center over to the Sands Convention Center, TELEPHONES, TELEPHONES, TELEPHONES!!!

You get the picture.

My wait started as soon as I landed at the airport. It was late on a Sunday afternoon, and you would have figured everyone was at home watching football like normal patriotic Americans. Not on COMDEX Sunday Eve. The line to get a cab reminded me of those Russia-like bread queues I used to stand in for the

Shock Wave roller coaster at Six Flags Over Texas–a feat, which I must tell you, I only put up with until the age of 16, when I discovered you could get the same sensation by slamming a six-pack of Shaefer's beer.

I considered going straight back inside the airport and entering the first bar I found so that I could watch football instead of standing in line, but decided to stick it out and make my way on over to the hotel.

So began "Fear and Loathing in Lost Wages, Part Deux."

The spirit of "coopetition" that has recently loomed like a dense fog over the computer industry had definitely migrated over the desertous wasteland of Lost Wages, for people from competing companies were actually sharing cab rides to their respective hotels. With the best fiscal interest of IBM in mind (and figuring I'd be better off using my corporate "Don't Leave Home Without It" card for something more important, like cash advances at the casino), I hooked up with a guy from Intel. As we cruised out of the airport at a whopping five miles an hour, he informed me that he was working on Intel's 960 embedded RISC microprocessor. Great, I thought, what's that?

When I told him I worked for IBM, he got a worried look on his face, like I was some reverse anthropomorphic version of the PowerPC chip who might bite off his head at any moment. In the true spirit of the day, I told him not to worry about our little friendly competition, that he should just "Get Warped," and that we would give him a job when Intel eventually became the Wang of microprocessors. (If only it had been a few weeks later, I could have wheedled him into pulling out his Pentium laptop to help calculate the tip.)

As we drove out of the airport, I saw that IBM had been kind enough to rent a gigantic Diamond Vision screen so that I could catch the end of the Cowboys/49ers game. Then I realized that football was a lost cause, as the screen was informing all 190,000 computer geeks fleeing the airport that they were about to be warped beyond all warpings and to fasten their seatbelts because, five miles per hour or not, this year's COMDEX was gonna be a wild ride.

A short while later, my heart began to pitter patter as the cab pulled up to that Eighth Wonder of the World, the Great Pyramid known as the Luxor Hotel. Hey, I had studied enough high school Egyptology to remember what happened to that King Tut dude in *his* pyramid, and I had no plans for getting dug up 2,000 years from now only to discover that Bill Gates had indeed successfully taken over the universe. But this was COMDEX, and beggars couldn't be choosers–I figured a pyramid was the next best thing to sleeping on a bench along Lost Wages Boulevard, even if I did have to stay in the same room as my manager.

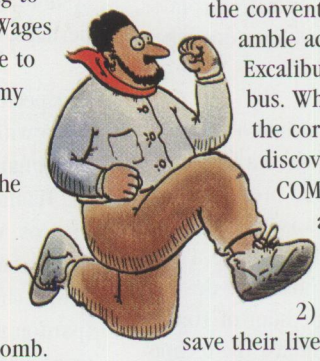
I skeptically entered the large confines of the behemoth Giza structure and approached the check-in counter to inform the hotel personnel I was ready to be escorted to my tomb. Instead of giving me a key, they handed me a small magnetic card with Warp smeared all over it. Man, I thought to myself, is there anywhere we *won't* advertise this thing?

When I discovered I was staying on the 13th floor, I almost turned tail back to the airport–I'm not suspicious, just very cautious–but decided that superstition in Lost Wages was a cheap commodity. After circumnavigating the mile long hallways of the Luxor several times, I finally came to realize that pyramids have inclinators, not elevators, and soon was making my 45 degree ascent up to lucky number 13.

Upon entering my room, I immediately checked out the bathroom and was sorely disappointed to *not* find a Warp banner wrapped across the toilet seat. I walked over to look out the slanted windows and, after bumping my head on the angled surface, attempted to calculate how many bottles of Windex the housekeeping staff went through in a year to keep the Great Pyramid so sparkly clean. I didn't have my Pentium PDA handy, though, so after a meditative silence, I instead gasped in awe at Lost Wages' million points of light.

I became so moved that I bowed down and began to pray to the gods of Mount Computus, for I knew that I had indeed arrived at the mecca of technology.

Being the nomadic warrior that I am, I promptly set up shop on the fancy Luxor desk to check my e-mail. When I turned on the boob tube for a little background noise, I discovered COMDEX had its own TV channel. That's when I knew for sure that the show had gotten too big for its britches, although I hadn't yet traipsed across the actual show floor. Even IBM got rid of its own TV channel.



The next day I discovered that to get to the convention center, I had to amble across the street to the Excalibur Castle to catch the bus. When I walked around the corner to the bus stop, I discovered about 300 other COMDEXgoers who were also 1) too cheap to take a cab over to the convention center, or 2) couldn't find a cab to save their lives. Figuring IBM would want me to get the most for their money out of this experience, I went back to the Luxor in search of a cab.

I hooked up with a couple of other COMDEXers, and we were fortunate enough to catch a ride with a cab driver who, I'm sure, works for the Mafia in his spare time. He jovially related to us the story of an unfortunate gentleman who, one night not too long before, thought he was about to engage in one of the activities which only Nevada can provide on a legal basis (if you catch my drift) and instead woke up to discover he was missing a kidney! Lost Wages apparently has a booming black market for transplantable organs.

Although I desperately wanted to know how one wakes up after a drug-induced slumber and comes to the realization that a kidney is missing (a molar or an eye I could understand), I figured that Don Cabbie was not the person to ask; and anyway, I had business to attend to at the convention center. I can assure you, though, that for the rest of the trip, I kept my eyes peeled for beautiful women trying to conceal long syringes under their evening gowns.

Upon my arrival at the convention center, I immediately felt guilty for taking a cab (even though my share only cost IBM $3). It seemed that instead of renting a huge convention hall, IBM, Apple, and Motorola had pooled their money to pitch a big tent outside the convention hall. They called it the PowerPC Pavilion, but they couldn't fool me–I know a tent when I see one.

Figuring that we would be serving Warp dogs and Budweiser, I entered the tent thinking how great it was that I would be saving loads of money on my expense account, only to discover there was no food or beer inside–just microprocessors. Since I've never been a silicon fan, I decided I would hold off on eating computer chips until I was ravenous enough to start seeing mirages.

Although I work for IBM, I had obtained a media badge in order to score a free lunch in the IBM-sponsored press room and have people think I was a very important person (which, of course, I'm not). This meant I could not enter the show floor until the magic hour of 10:00 a.m. arrived. So I stood amidst the techno-hungry masses and began to panic, figuring I was about to experience something akin to the Oklahoma Land Rush.

Boy, was I right.

The Annual Running of the Geeks in Lost Wages makes Pamplona's bull sprint seem like a flea rodeo. Even Hemingway would have sat this one out. Ten a.m. struck, and you would have thought IBM was handing out ThinkPad 755CDs (which, by the way, won The Interface Group's and *Byte* Magazine's Best of COMDEX) to the first 10,000 people who made it to the other end of the show hall a couple of miles away. To keep from getting wingtipped to death, I quickly took refuge in what turned out to be the WhenDoze 95 SWAT Team headquarters.

At first not realizing I was in enemy territory, but being the cognizant dude that I am, I immediately tuned in to the whispers about Microsoft providing its customers "early experiences" with WhenDoze 95. Virtually scratching my head in an attempt to decipher their code-speak, I wondered if having an "early experience" was something akin to learning how to tie one's shoes, then finally realized what they meant to say was *beta.* (Why use the right word when an obscure one will do?)

When I discovered they were charging people $30 a pop to help THEM get the bugs out of THEIR betaware, I knew we were in for a good battle. How could a gargantuan company like Microsoft have the gall to charge its customers 30 big bucks for such a tiny little operating system, and a beta version to boot (or, perhaps, not boot)? That is the question.

I suspected it must have had something to do with Bill Gates' recent purchase of Leonardo da Vinci's ancient sea scroll, the Hammer Codex, for a few million dineros. When I caught wind that Billy Bob Gates was hangin' around the IBM booth, checking out the totally awesome (pardon the Valleyspeak, but it was, in a word, awesome) OS/2 Warp demo, I considered telling him that looking for a solution to Microsoft's difficulties with WhenDoze 9? in a Leonardo da Vinci parchment was like using a Pentium computer to calculate *pi* out to 25 digits–try though they might, they just weren't going to come up with the right answer. (As they said on the Net, "Intel Inside, Doesn't Divide.")

But enough industry parlance. It's pointless to argue with a few hundred mental marketing reps, and Billy Bob was too busy planning his Napoleonic takeover of business as it is known in the Western Hemisphere to concern himself with minor technical details. So like a good little IBM foot soldier, I escaped the horrific confines of the Microsoft encampment and made my way into the *sanctum sanctorum* of IBM's Warp Room, where prospective Warpees were test driving the latest, and seriously greatest, version of IBM's premier desktop Wizard of OS.

No longer prisoners of time, trapped behind windows of memory-limited despair and eternally suspended hourglasses, I was almost moved to tears as all those test drivers' eyes glazed over with the possible revelations and permutations that computing in a timeless expansion of unlimited Warpdom might bring to their computer-ridden lives. Poor things. Like me, they had come to the desert in search of visions, but had gotten Warped instead.

Which, in the end, was probably just as well. Visions are highly overrated–if Lou Gerstner didn't need one, why in the world did I?

Deciding I'd had enough of COMDEX for one lifetime, I made my way out of the convention center and into the serpentine line leading to Bus Route #2 for the long trek back to the Great Pyramid. Kidneys still intact, mind mildly expanded, and nerves greatly overwrought, I reflected on the great wise words of that 19th century French poet, Paul Valery:

"The trouble of our times is that the future is not what it used to be."

Couldn't have said it better myself.

So I propose a toast: Here's to yesterday, because you never know–in the wild and wacky business of computing, tomorrow may never come.

Todd L. Watson, assistant editor for *SQ: IBM's Magazine of Software Technologies* and */AIXtra: IBM's Magazine For AIX Professionals*, has been with IBM since 1991, specializing in AIX and RISC System/6000 communications issues. He holds a BA degree in English and an MA in Mass Media Studies from the University of North Texas in Denton. He can be warped via the Internet at radar@vnet.ibm.com.

Road Trip! Touring the Side Roads of the Internet

By Van Landrum

This is the first of a series of articles by Van Landrum,* Personal Systems' *database administrator, describing the highways and side roads of the Internet. In future issues, Van will take you on road trips to the interesting places he has discovered–from the mundane to the exotic. This time Van will take you from the rooftops of Boulder, Colorado, to the coffee break room at Cambridge University in England!

Hi! My name is Van Landrum and I am the database administrator for IBM's *Personal Systems* magazine as well as */AIXtra* magazine and the Technical Coordinator Program. Some days I'll be sitting at my desk trying to figure out the right way to connect several database tables and execute an impossible SQL statement when I just want to yell *"AARGHHH!"* and give up. Well, I've found that when nothing is going right, and it feels like brain cells are dropping dead by the score, two words come to mind: *ROAD TRIP!* Back in the seventies I lived in Chicago, and when I got bored with the everyday life, I jumped into my 1971 MGB and drove to Boston or Los Angeles or down to Houston. Now I don't have time to get into my car and drive across the country. Instead, I open up the WebExplorer and take a road trip on the Internet.

A road trip on the Internet is a wonderful way to get away from it all for a few minutes. I have always enjoyed traveling. I like to go places I haven't been before and see things I haven't seen before. I guess it's just the change of scenery that revitalizes me. There are so many interesting places to visit I wish I could make a living traveling all over the world.

I have found, however, that visiting different places through the Internet also gives me a whole new outlook on life and work. Using the WebExplorer I downloaded from IBM with the OS/2 Warp Internet Access Kit (IAK), I can visit the Capitol in Washington DC, the Louvre in Paris, colleges and universities, major corporations, electronic malls, as well as visit with people all over the world–all from the computer at my desk.

In the next several issues I'll take you along as I ride the Information Superhighway. We'll look at ways to have fun, relax, and find important information. Along the way, we'll learn about other people and the work that they do as well as how they spend their free time. Hopefully you'll get ideas on how to use this new resource in your own personal way.

Riding the Internet on the WebExplorer

The WebExplorer began in June of 1994 when the World-Wide Web was beginning to gain mass appeal. IBM was developing OS/2 Warp and planning to include an Internet Access Kit. IBM looked at Mosaic, developed by the National Center for Supercomputer Applications (NCSA), but a few IBM programmers wanted to develop their own browser. To find out more about the IBM team that created the WebExplorer, I jumped into my WebExplorer and headed out down the Internet.

First stop was the IBM Home Page (`URL=http://www.ibm.com`). This colorful page is one of IBM's access ramps to the Internet (see Figure 1). This page offers news about IBM (as I write this, the lead story is "IBM Halts Shipments of Pentium-Based Personal Computers"); a message from our CEO, Lou Gerstner, welcoming all newcomers to the IBM World-Wide Web server; and other related items about IBM and the Internet. One of the listings, "Developing WebExplorer: four folks, four months, no sleep," sounded interesting. With a click of the mouse, I took the next exit and headed for `http://www.ibm.com/Features/4guys.html`.

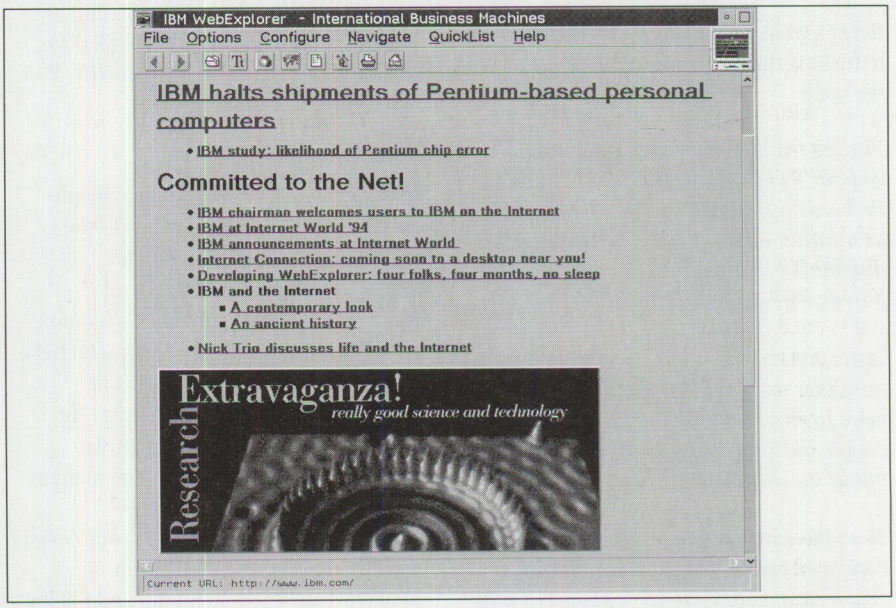


Figure 1. IBM Home Page

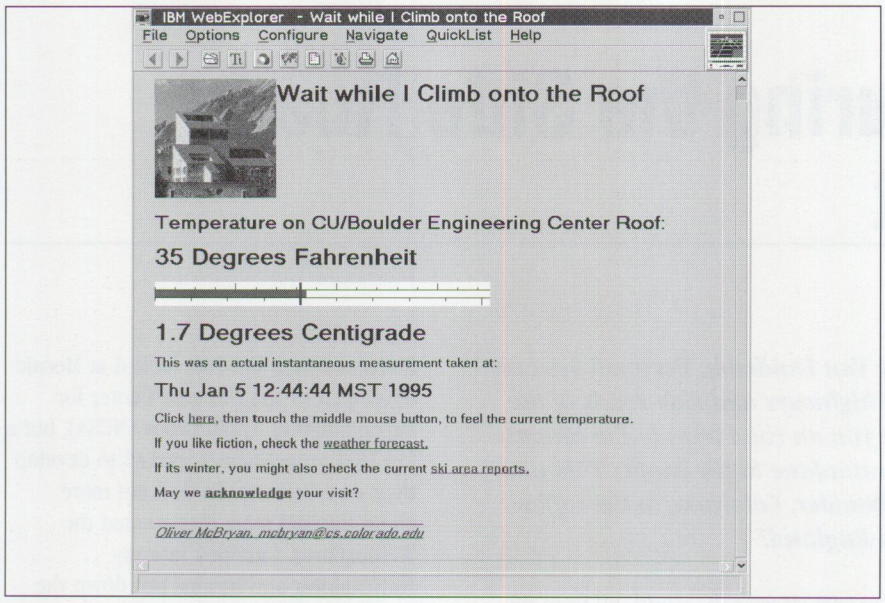


Figure 2. Thermometer at University of Colorado

Editor's Note: For a quick primer on exactly how to move around the WebExplorer, see Phil Lieberman's article "A Guide to OS/2 Warp's Internet Access Kit" later in this magazine.

The article that appeared explained how David Greenwood, Barbara Walters, Scott Penberthy, and Mike Ward pulled together and built a great interface to the World-Wide Web. Not an easy task mind you, since David and Barbara were in North Carolina and Scott and Mike were in New York. The article goes on to explain how, while working late nights, they met one deadline after another to create a web browser that would take advantage of OS/2 Warp's capabilities.

Here is what they had to say about the hardships they encountered while writing the code:

Putting together the 100,000 lines of code in WebExplorer was not without its hardships. Since the effort began as an after-hours project, quirks like automated lighting and air conditioning posed some unusual challenges.

"After 8 p.m., the lights go out in our building," explained Greenwood. "You have to run down the hall and push a couple buttons every 20 minutes to stay out of the dark."

To make matters worse, the WebExplorer code was being compiled on a 486 PC. "Each change of code took about 10 minutes to run through the compiler, so you'd get to make one or two changes; then you'd have to run down the hall and push the button to keep the lights on. That would give you enough time to run back and make one or two more changes, and then run back down the hall for the lights."

"And on top of that, our air conditioning shuts down at 7 p.m.," added Walters. "You have to remember that this was summertime in North Carolina."

Just goes to show what four people can accomplish in four months if they don't mind missing a little shut-eye.

Take a trip to `http://www.ibm.com/Features/4guys.html` and find out the special features they added to the WebExplorer to make it worthy of OS/2 Warp.

Indiana Innovations

You'll be amazed at how some people are using the Internet. Let's visit the Center for Innovative Computer Applications at the University of Indiana. We can go see Dennis Gannon, Juan Villacis, and Shelby Yang, who have set up cameras to take pictures of their offices every three minutes. By connecting to their web server, we can see inside their offices. The address for spying on the CICA folks is `http://www.cica.indiana.edu/news/spy.html`. I looked in on them on January 4, 1995 at 4:33 p.m. and Shelby was the only one working.

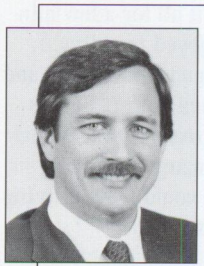
From Indiana, let's go to the University of Colorado. There they have another innovative example of using the web. They connected a thermometer atop the CU/Boulder Engineering Center roof to a web site. By connecting to `http://www.cs.colorado.edu/htbin/temp` you can get the current temperature on the building. On Thursday, January 5 at 12:44:44 p.m. MST it was 35 degrees (see Figure 2). Notice the comment tells you how to "feel" the temperature using the middle mouse button!

Cambridge Coffee

Now let's extend our road trip across the Atlantic Ocean to England where we can spy on the Trojan Room coffee machine at the Cambridge University computer laboratory. Connect to URL `http://www.cl.cam.ac.uk/coffee/coffee.html` to see how much coffee is left in the pot. (I thought the beverage of interest would have been tea!)

These examples show some of the innovative ways to use the Internet. While you may not care to see Dennis Gannon's office or know how much coffee is in the pot in England, just envision ways you could make use of such technology in your company.

In the coming issues we will visit a wide range of sites, including the Louvre Museum in Paris, the White House, weather stations, NASA, colleges, electronic malls, and many others. We will see many uses for the Internet and the ways people around the world are creatively delivering information on the Information Superhighway as we travel into the 21st century.



Van Landrum is a marketing support representative in the IBM Personal Systems Competency Center in Roanoke, Texas. He is the 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`

Visualizer–The Conversion Begins

By Bob Angell

***In this second part of Bob Angell's series on IBM's Visualizer product, he discusses importing data, converting applications, and developing applications with and without Visualizer. The first article, "Visualizer, DB2/2, and You–An End-User's Perspective," appeared in the January/February 1995 issue of* Personal Systems.**

Welcome to the next segment of our discussion on Visualizer. As most of you know by now, Visualizer is a visual front-end tool used to gather, retrieve, display, and massage database data. It can access a myriad of information from DATABASE 2 OS/2 (DB2/2) and its own Visualizer table structure to OS/400 data.

My goal with the Visualizer articles is to chronicle a project my organization, Applied Information and Management Systems (AIMS), undertook: the conversion of a DOS database program to a Visualizer- and DB2/2-based client/server application. The original July 1995 deadline to complete this conversion was moved to January 1995. We were frantically trying to learn and understand all of the Visualizer family of products, especially the development package. In the remainder of these articles, I will discuss exactly what we did to get the application ready and delivered.

Visualizer Tables

Much of the underlying structure of our project's original DOS-based application was easy to reproduce. Using the templates that came with the Visualizer package, we were able to import the data into structured query language (SQL)- and Visualizer-based tables. Importing data into a Visualizer table is fairly easy, since Visualizer supports a wide variety of import data strategies. The tables, queries, views, and other Visualizer functions all resemble the Visualizer table structure shown in Figure 1. Each function has a menu with familiar and consistent operations. The tool bar is also a consistent feature; once you learn all of the many buttons (something I am still struggling to master), interacting with Visualizer becomes a breeze.

SQL Tables

Importing your data into an SQL table is somewhat different from importing it into a Visualizer-based table. First you need to create a database either with the Visualizer SQL database template or by hand. Once the database has been created, you can create an SQL table or import the data from another source. One of the easiest ways to do this is to create a Visualizer table, populate it, and then convert it to an SQL table.

We found that this is the easiest way of getting a `.DBF` file into an SQL format, although we wish it were a bit more intuitive. Each of the tables in our project converted without a glitch, but it became apparent that with a fairly large table (>50,000 records) an ASCII import was needed to minimize file overhead. We discovered that if data is needed for an application, interacting with a Visualizer table through the development package rather than an SQL table is easier and more straightforward.

Programming the Application

The DOS-based application was easy to program and prototype. All programming was procedural (allowing only one event to occur at a time) as compared to Visualizer's event-driven (allowing more than one or the same process to run at any given time), powerful, and feature-rich high-level Application System Language (ASL). The ASL provides a way to interact with SQL sessions, link externally written code, and control user-written applications.

To begin the application conversion, we stepped through the tutorial very carefully. As a basis for our application, we created a window using a Visualizer Window object with both standard and custom menus. The nice thing about Visualizer's standard menus is that they have all the code written to open and edit

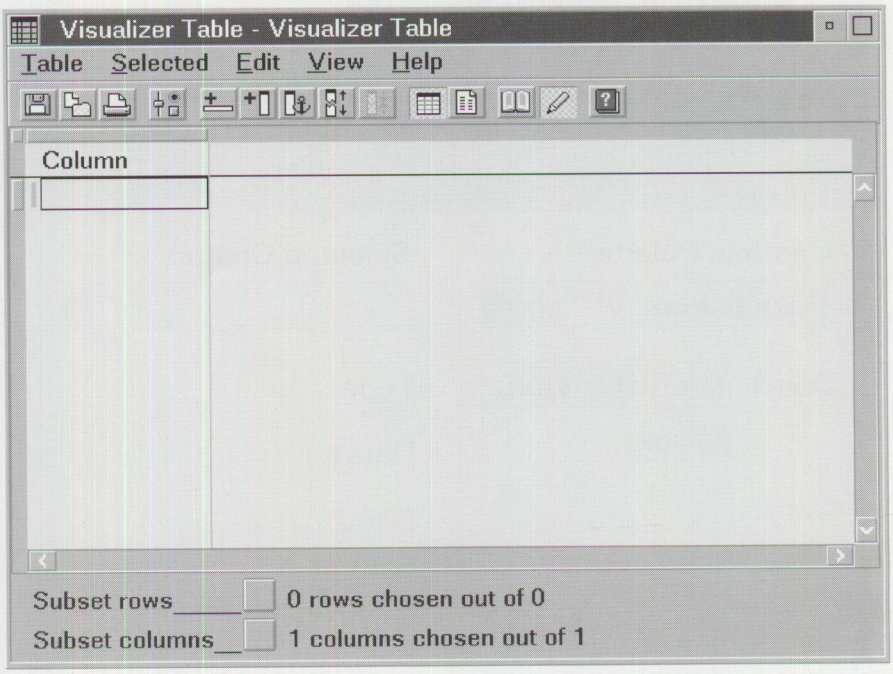


Figure 1. Visualizer Table Structure

```
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!
!! This is a procedure to take data out of a Visualizer
!! table called WinDoc.TAB and place the contents into the
!! vector (array) docs02[] for use with the DropEntry
!! list for all of the Physician names for the project
!! OrthoPEDIC.
!!
!! Written by: Bob Angell
!! Date: December 1994
!!
!! Copyright 1994-5-Applied Information & Management
!! Systems
!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!

PROCEDURE docFill
DO
  DECLARE NUMERIC i              ! This is the looping element and is
                                 ! defined here.
  !
  let tname = "WinDoc.TAB"       ! The name of the Visualizer table
  let tloc="c:\desktop\orthdev"  ! Location of the Visualizer table
  !
  OPEN TABLE XYZ,                ! Let's open up a table and call it
                                 ! XYZ
   NAME = tname,                 ! Name of the table is tname
   LOCATION = tloc,              ! Location of the table is tloc
   OBJECTCLASS = "IBMTABLE",     ! Tell Visualizer its an IBMTABLE
                                 ! types
   MODE = "READ"                 ! Just need to read from the table
  !
  let numbrows = XYZ'ROWS        ! The number of ROWS in the
                                 ! Visualizer table
  !
  DEFINE docs02[numbrows]        ! Assign vector size = ROWS in
                                 ! Visualizer table
  DO i = 1 : numbrows
   LET docs02[i]=XYZ.ABREV[i]    ! Value from table column is
                                 ! assigned to vector
  END
  SHUT XYZ                       ! Once the table is used, it needs
                                 ! to be shut
END
```

Figure 2. Procedure to Take Data Out of a Visualizer Table

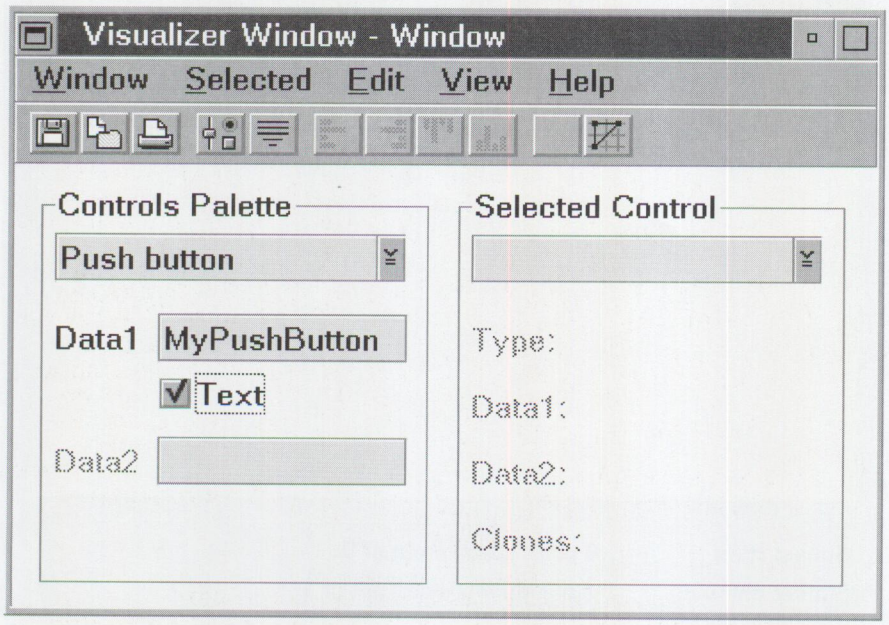


Figure 3. Visualizer Window Object

files, interact with Visualizer Charts, print, and call online help.

The window was then assembled using drag-and-drop, one of OS/2's premier, time-saving features. The Visualizer Window object can control the shape, size, and color of any window that is created. Additionally, it can control the font, window behavior, and scroll bars. Once the window and menus were assembled, we compiled it and operated it as a stand-alone program.

This simple-to-create window is the basis for our Visualizer project. After the main window and menu system are installed, you need to connect all menus and add their functionality. If you are going to use the development package, you need to learn to use the "make" program, as this is the easiest way to track and produce a large application. The "make" facility produces a program file that combines all the windows and other code to make the application a free-standing program.

The application we developed uses several DropEntry lists. The code that the Visualizer Development package produces was changed so that all the information for the lists came from Visualizer tables. This is best illustrated with the code shown in Figure 2.

Note: All comments are made using the exclamation mark. The above behavior was not easy to figure out, but with help from the product support team in Texas, we were able to move forward.

Collecting Data

Once we understood the windowing process under Visualizer, we concentrated on the main data collection input screen. Each screen can contain up to 16 different data elements ranging from a Single Line Entry to a Dropdown Entry and even a Radio Button or two. These data elements are part of the Controls Palette located on the front of the Window object (see Figure 3).

Each of these elements has a name associated with it in the "Data1" entry box. In some cases, an array or vector is also assigned and is recorded in the "Data2" area. If you are using plain text to describe the window elements, there is a "Text" check box after the "Data1" entry.

Once the data has been placed on the Visualizer window, the values show up on the right side of the development environment as shown in Figure 3.

As you visually create each item (or data element) on the screen, you can copy and manipulate it for the appropriate data that it will collect. You can even define the type of information that should be collected and place controls on what data should reside with the element. *Cloning* is another Visualizer feature process that allows you to group like items together and know that they will be acted upon one at a time.

The window we developed contained approximately 50 separate data elements and took about three hours to create. Had we programmed this in C++, it would have taken the better part of a day just to code the window behavior and several more hours to debug. Because of its window behavior, its functionality, and its ease of creation, it made sense for us to use Visualizer to develop our database applications.

Support

You can find support for Visualizer using several avenues, including fax systems. The IBM Visualizer support team receives and answers facsimiles 24 hours a day, seven days a week. This is a very good way to receive support. Even during the holidays when IBM is practically non-functional (and when this article was written), I was able to use the fax service and get my problem resolved in a reasonable amount of time.

To use this fax service, registered customers should print out the source code of the application or problem area, include a description of the problem(s), and fax it to (817) 962-4916. Faxes are answered in the order they are queued during normal business hours.

Unofficial Visualizer support can also be found electronically on the OS2BBS, CompuServe, and will probably be found on the Internet as the product is more widely used and accepted. You also may get support through the usual toll number (512) 339-5000 from 8:00 a.m. to 8:00 p.m. EST. No matter which method you use, support is always provided by very competent individuals–which is rare to find these days.

In the next article, "Visualizer–The Conversion Continues," we will further outline our project's conversion process.

Bob Angell is a principal with Applied Information and Management Systems (AIMS) in Salt Lake City, Utah. A management consulting firm, AIMS specializes in management information systems integration, OS/2 development and integration, total quality improvement engineering, and other related services. Bob's specialties include multiplatform data integration, database design and development, simulation and modeling of complex environments (neural networks), and OS/2 software development. Bob can be reached through the Internet at `bangell@cs.utah.edu`.

What's New in PC DOS 7

IBM's PC DOS 7 is the latest release of the industry standard for DOS and Windows users. PC DOS is "classic" 16-bit DOS, considered by many to be the best DOS in the industry for Intel and compatible personal computers. PC DOS 7 has many new features, plus significant usability and performance enhancements, which are detailed in this article.

PC DOS 7 establishes IBM as the leader in providing new DOS technology to the computer industry. PC DOS 7 is the most full-featured DOS in the marketplace, and its application programming interfaces (APIs) are 100% MS-DOS- and Windows-compatible, making PC DOS 7 the best choice for all DOS and DOS/Windows users.

With this new release, you can get up to 40 KB of additional free memory, Stacker 4.0 data compression, online help, docking support, file synchronization, enhanced Personal Computer Memory Card International Association (PCMCIA) support, and REXX programming capabilities. PC DOS 7 also includes significant enhancements to PC DOS's already superior line of utilities (Figure 1). In addition, PC DOS 7 is available in 18 languages. These enhancements, plus features such as improved backup, virus detection, online editor, advanced power management, undelete, program schedulers, pen support, and other performance improvements are discussed in detail below.

Kay Lee
IBM Corporation
Boca Raton, Florida

Nestor Miranda
Electronic Data Systems
Boca Raton, Florida

Memory Management

DOS users want to maximize the available conventional memory. PC DOS 7 requires less conventional memory than DOS 6.x for comparable functions, freeing more memory for user applications. The extra conventional memory available to PC DOS 7 applications typically ranges from 5 KB to 40 KB.

Running a DOS application while ensuring that enough memory is still available for other programs can be challenging as well as frustrating. PC DOS 7 addresses this challenge by employing a variety of memory reduction techniques.

These PC DOS 7 components have been optimized to use less memory:

```
COMMAND.COM
DOSKEY.COM
MOUSE.COM
MSCDEX.EXE
RAMBOOST.EXE
SMARTDRV.EXE
ANSI.SYS
DISPLAY.SYS
HIMEM.SYS
RAMDRIVE.SYS
```

The following switches or techniques are implemented in PC DOS 7 for memory savings:

- `MOVEXBDAHIGH` loads the XBDA (eXtended BIOS Data Area) into upper memory, if available.
- `FASTA20` increases the A20 handling speed and improves performance.
- `DOSDATA=UMB` loads the system tables (`FILES`, `FCBS`, `BUFFERS`, `LASTDRIVE`, `STACKS`) into upper memory if available (`DOS=UMB` must be in effect, and an upper memory manager must be installed).
- `DOSDATA=NOUMB` loads the system tables into conventional memory (this is the default).
- `IBMBIO` code has been moved to the high memory area (HMA). PC DOS 7 dynamically allocates `IBMBIO` data as needed.
- Compression can now be loaded into protected mode using DOS Protected Memory Services (DPMS). This frees up approximately 20 KB of conventional memory.

Memory Manager

PC DOS 7's memory manager is EMM386 (not to be confused with the RAMBoost optimizer). With EMM386, you no longer have to choose between expanded memory specification (EMS) and extended memory specification (XMS) when starting your system.

EMM386 uses available upper memory block (UMB) RAM. Its search algorithm finds more free space in the high memory area as well as in unallocated areas of memory. EMM386 can also allocate expanded and extended memory from shared pools.

Memory Optimizer

PC DOS 7's integrated memory optimizer, RAMBoost, automatically and dynamically optimizes your system's memory. RAMBoost looks for the best arrangement for device drivers and for terminate and stay resident (TSR) programs within UMB. Because it frees up conventional memory (under 640 KB), you can now run DOS programs that may have previously experienced memory shortage problems.

The RAMBoost optimizer in PC DOS 7 works much like other optimizers. First, it scans your configuration files and determines which TSRs and device drivers are being loaded. Next, it probes memory and orders the TSRs and device drivers in what it determines is the optimal way. Finally, it saves the optimized configuration and restarts your PC.

Then RAMBoost goes one step further: it provides "dynamic optimization," an intelligent learning and tracking feature that watches over your system configuration. If you have modified your configuration (such as by adding entries to your configuration files), and the configuration watcher detects a change in any of the tracked files, RAMBoost asks you if you would like to re-optimize. If you respond with YES, RAMBoost goes into LEARN mode and optimizes your system. Otherwise, you can select NO and optimize at a later time.

Alternately, you can set RAMBoost to AUTOMODE. In this mode, RAMBoost does not ask you whether to re-optimize. Instead, RAMBoost automatically re-optimizes whenever the configuration watcher detects changes. This technique requires no user intervention. In comparison, MS-DOS's MemMaker does not implement dynamic memory optimization. Instead, MemMaker uses the static approach–you are responsible for knowing when to run MemMaker.

PC DOS 7's RAMBoost now prompts you if it finds an EMM386 statement in your `CONFIG.SYS` file and gives you the choice of either letting RAMBoost create a new EMM386 statement or using the existing EMM386 statement. (In DOS 6.x, if an EMM386 line was found, it was used as-is, but if it was not configured correctly or optimally, you would not realize the best possible memory savings.) Although it is usually preferable to permit RAMBoost to create a new EMM386 statement, PC DOS 7 offers you a choice if you have customized the EMM386 statement and want to continue using that statement.

Another RAMBoost feature is its multi-configuration awareness. RAMBoost searches the `CONFIG.SYS` file to determine whether PC DOS 7's multi-configuration support has been implemented. If found, PC DOS 7 optimizes and tracks each environment separately, giving you dynamically optimized memory utilization in multi-configuration environments. In contrast, MS-DOS's MemMaker is not multi-configuration-aware.

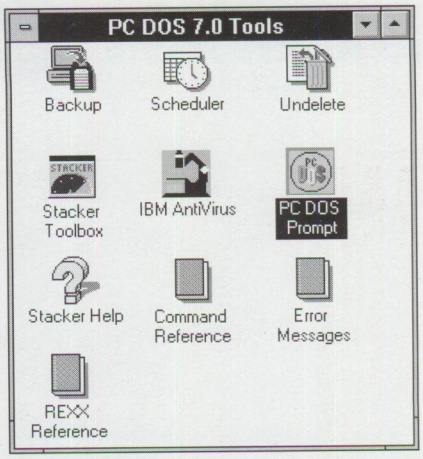


Figure 1. DOS and Windows Utilities in PC DOS 7

Additional UMB Device-Driver Support

PC DOS 7 includes support for device drivers found in 8088 and 80286 computers, thus supporting EGA, VGA, and EMS cards. With PC DOS 7, the extra memory found in these cards won't be wasted. If you have video cards or EMS boards, you can realize as much as 60 KB of additional free memory.

Advanced Upper Memory Usage Editor

PC DOS 7 comes with an easy-to-use graphical editor that allows you to directly manipulate the high memory area.

Data Compression

PC DOS 7 includes Stacker for Windows and DOS, the award-winning data compression standard (Figure 2) that safely increases your system's disk capacity.

Upgrading from prior compression technologies such as MS-DOS Double Space/DriveSpace, PC DOS SuperStor/DS, or earlier versions of Stacker is fast and easy. Even with its many features, Stacker is designed to use less conventional memory than other compression programs.

Stacker's patented LZS compression and the new Stacker SmartPack, both of which are in PC DOS 7, combine to give you a better compression ratio and more disk space than any other data compression product. Whereas other compression schemes have, at best, an average 1.7-to-1 compression ratio, Stacker enables a

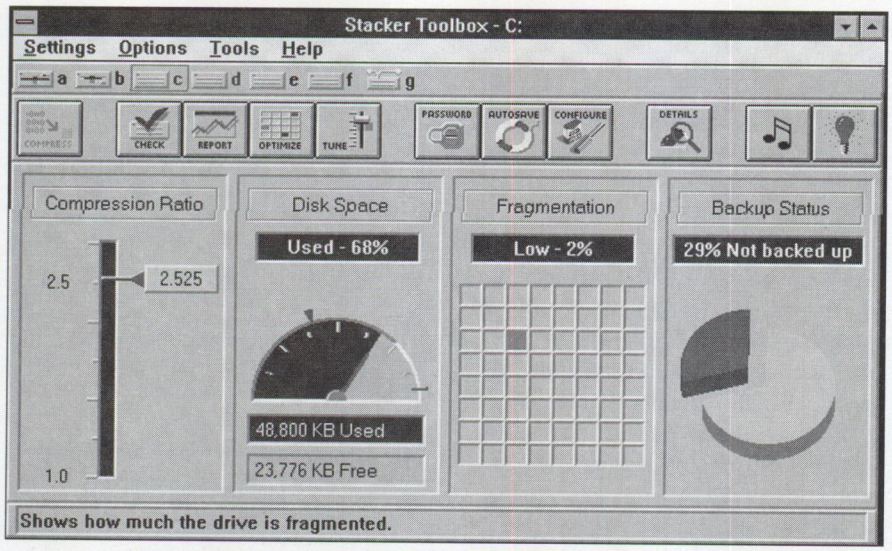


Figure 2. Stacker for Windows and DOS

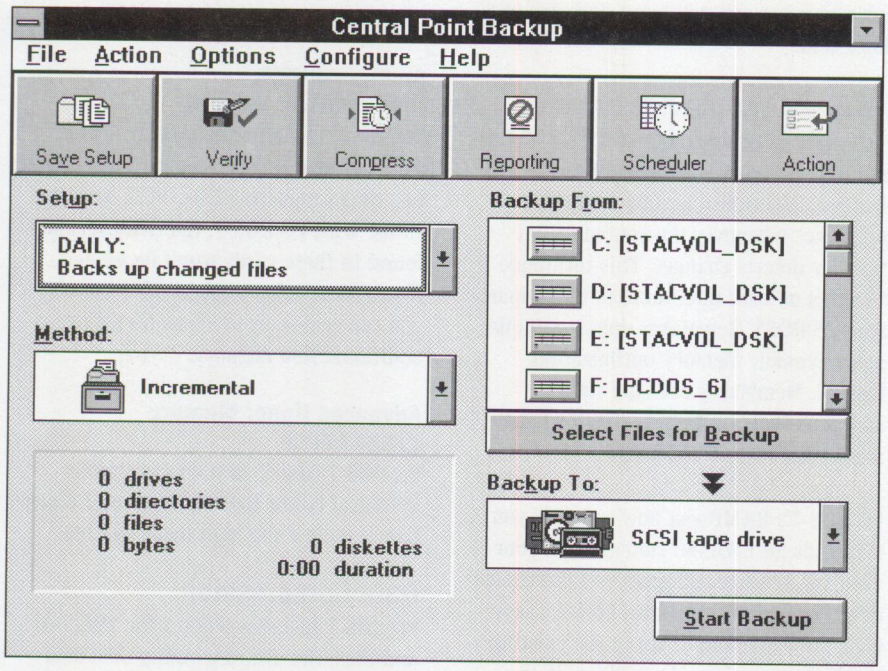


Figure 3. PC Tools Backup Program

100 MB disk to hold 250 MB of data—a 2.5-to-1 ratio. Stacker is the first real-time software compression technology that breaks through the 2-to-1 compression barrier, more than doubling your system's hard-disk capacity.

PC DOS 7's new compression features give you:

- *Easy access to your Stacker-compressed drives from either a DOS or Windows Toolbox.* You can easily see how much free space is available, how fragmented the drives are, when the data was last backed up, which compression settings are being used, the current compression ratio, and disk usage information.
- *The ability to read and write data on compressed disks on another computer, anywhere—even if that computer does not have Stacker compression installed.* Stacker Anywhere, Stacker's transportable floppy technology (called Universal Data Exchange in PC DOS 6.3), makes using compressed diskettes practical in all environments. In contrast, with MS-DOS's compression scheme, you cannot read from or write to compressed diskettes unless both computers are running the same compression scheme, e.g., DriveSpace or DoubleSpace.
- *Easy conversion from DriveSpace, DoubleSpace, SuperStor/DS, or other Stacker drives to the new Stacker 4.0 compression algorithm.* The resulting conversion yields more space and reliability.
- *More free memory than other compression schemes by loading the compression driver into protected mode using DPMS support.* The compression driver uses only 17 KB of memory in the address space below 1 MB.
- *Fine-tuning options to control the balance between how tightly and how quickly you want data compressed.* The Stacker Tuner controls the balance between how fast Stacker works (MaxSpeed) and how tightly it compresses data (MaxSpace). Stacker Setup automatically sets the Tuner to maximum space, which gives you maximum compression. You can fine-tune your compression settings with the three Toolbox settings or by editing the `STACKER.INI` file.
- *Protection for your data.* Every time you start your system, PC DOS 7 runs AutoProtect to ensure that your data is in good condition.
- *Reminders, by flashing an icon or sounding an audio tone when maintenance tasks need to be performed.* You can customize PC DOS 7's compression to let you know when the disk is getting full, it is time to back up files, or it is time to optimize the disk drive.
- *The ability to check your data and Stacker drive integrity and to fix any errors by using the Check tool option.* Checking drive integrity includes checking the file structures on the disk, repairing any errors found, and checking the disk media (surface scan).
- *An estimate, based on the kind of data on your hard disk, of how much your data will compress.* (Compression ratios vary depending on the type of data.) No longer do you have to guess which ratio to enter—PC DOS 7 automatically adjusts its compression based on the data in the system, always giving you the most possible space.
- *Password protection for your compressed drives.* By assigning passwords

to your compressed drives, you can ensure that only you or users you specify have access to information on these drives. PC DOS 7 supports read/write or read-only passwords.

Backup

PC DOS 7 includes a full-featured DOS and Windows version of Central Point's PC Tools Backup utility program (Figure 3). Both user interfaces are easy and intuitive, using a tree-view display of your file system for easy point-and-click operations. However, if you do not want to use the DOS or Windows interface, PC DOS 7 lets you run Backup from the command line.

This backup utility comes with features not found in other DOS products, including valuable file viewers (for both DOS and Windows files) that let you view the contents of files prior to backing them up (Figure 4). The Backup program also includes a very easy-to-use scheduler program that you can use to back up your system's hard disks during a time when the computer is otherwise idle.

With today's hard disks exceeding 300 MB, a full backup solution must include the ability to back up to tape. PC DOS 7 supports tape backup to many tape formats, including QIC 40/80 and small computer system interface (SCSI) devices. You can also specify to have your data compressed during backup.

PC DOS 7 also supports optical, hard-disk, diskette, high-speed diskette, and removable disk drive backup devices.

AntiVirus

Protecting PC and LAN environments must be a key component of every business recovery process. Understanding the importance of safeguarding your business is easy, but protecting against the threat of computer viruses can be difficult. Viruses are a leading cause of business disruptions, and inadequate protection against them can bring your business to a standstill. Consider:

- Three new viruses are created each day.
- Viruses are rarely isolated to a single PC; they spread via networks and diskette distribution.
- Instances of businesses infecting one another are on the rise.

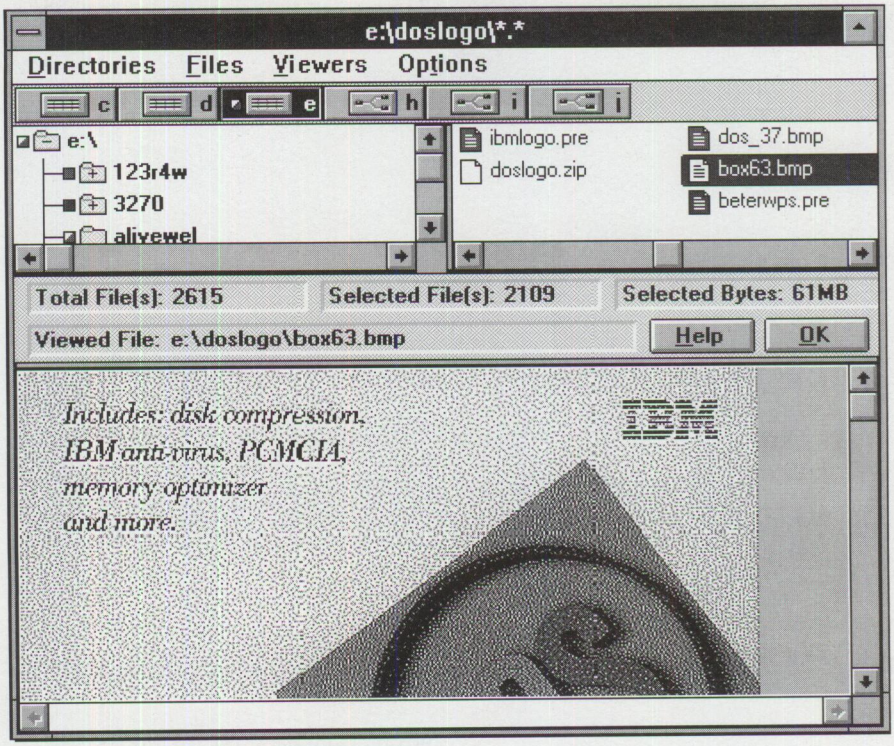


Figure 4. File Viewer in PC Tools Backup

PC DOS 7's AntiVirus feature (Figure 5) is an integral part of any recovery strategy, providing the most technologically advanced virus database and tracking service available today. Developed by IBM research scientists, PC DOS 7's AntiVirus feature benefits from their years of experience dealing with virus incidents around the world. The result is a highly reliable product that has been certified by the National Computer Security Association. With PC DOS 7, you gain access to highly advanced virus detection technologies, dedicated technical support, and the security of knowing that your business continues to be protected even as new viruses are discovered.

AntiVirus is "install-and-forget" protection for your computer. As soon as it is installed, AntiVirus begins protecting your system from being penetrated by viruses. Thereafter, it remains active all the time, preventing common viruses from becoming active in your system and spreading to other systems. In addition, AntiVirus scanning can be run automatically at pre-scheduled times to check all of your system's hard disks for viruses.

AntiVirus does not damage the programs that it is disinfecting. It disinfects programs only when it knows exactly what

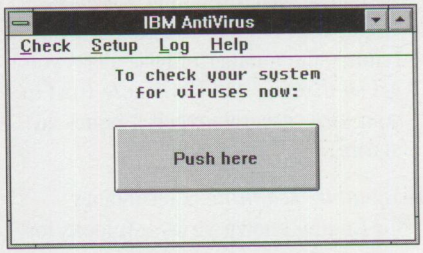


Figure 5. AntiVirus Program

the virus is and how to disinfect it. If AntiVirus does not recognize the virus, it notifies you of its suspicions.

AntiVirus provides:

- A full-screen DOS and Windows interface.
- Instant and automatic virus protection against over 2,100 known viruses as well as unknown viruses.
- Automatic checking for "boot sector" viruses on diskettes.
- Detection and disinfection of viruses with the touch of a button. Detection technology virtually eliminates false positives, and all virus incidents are logged for tracking and control.

AntiVirus Features

Major AntiVirus features include:

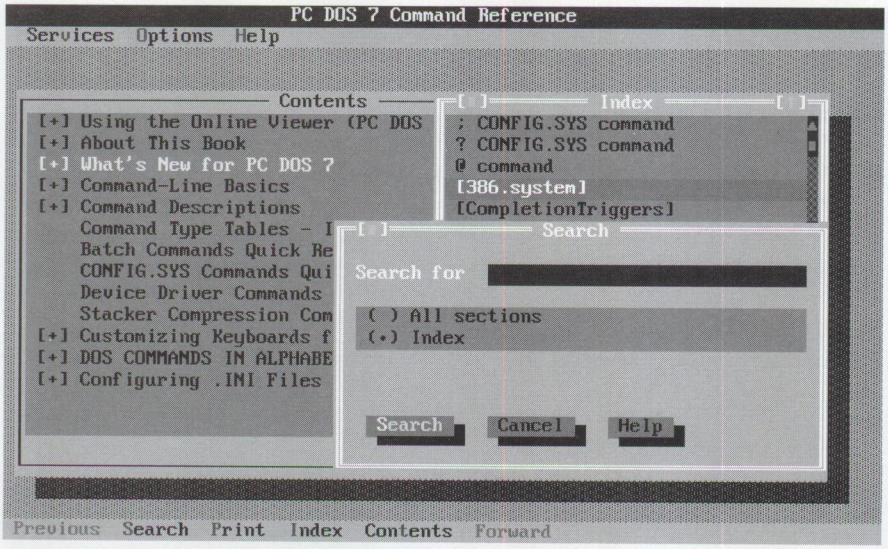


Figure 6. PC DOS 7 Help Screen

- *Integrity checking,* which tries to detect unknown viruses. It tries to find out when things have changed in the system and if the changes were made by a virus rather than something legitimate. It informs you only if things change in such a way that a virus is suspected. Integrity checking drastically reduces the number of false alerts you get so that you will have more trust in your virus program when it issues an alarm.

- *Signature scanning,* a technology for finding known viruses. It looks for a particular sequence of bytes, or more complicated things, that exist in a known virus. If a certain sequence of bytes occurs both in a virus and in a file or a boot sector on a computer, it indicates that a virus is on the computer.

- *Fuzzy logic detection,* a technology that IBM developed to complement signature scanning. Many viruses are variants of older viruses, where only a few bytes in a sequence are modified in order to circumvent antivirus programs' signature strings. PC DOS 7's AntiVirus program uses a fuzzy scanning technique that is smart enough to determine when these changes have occurred and is less prone to allow a virus to go by undetected.

- *TSR protection.* PC DOS 7's AntiVirus protection comes in the form of a TSR program. This is very important, because an effective protection scheme requires an active sentry at all times to warn you when a virus is present or when one has been introduced to your system.

- *Safe running of infected programs.* Another valuable feature in PC DOS 7's AntiVirus program is its ability to safely run infected programs as though they were not infected. The ability to run critical applications that have been infected by a virus—without any fear of spreading the infection or doing damage—is indispensable.

- *Minimal RAM requirements.* It is important for antivirus software to have as small a memory profile as possible. PC DOS 7's AntiVirus memory footprint is very small in comparison to other antivirus programs. When loaded into expanded memory, PC DOS 7's AntiVirus program takes up as little as 1 KB; when loaded into conventional memory, the AntiVirus program takes up to 6 KB. By keeping the memory footprint small, PC DOS 7 allows you to keep the AntiVirus program running at all times, ensuring maximum protection while not using memory that can be better utilized by other programs. In contrast, MS-DOS's memory footprint can take as much as 44 KB.

- *Network drives.* Antivirus program users usually have to scan network drives with one program and local drives with another. PC DOS 7's AntiVirus program scans all drives at once, no matter how they are configured. As long as DOS recognizes a drive, PC DOS 7's AntiVirus program will protect it.

- *Customized alert messages.* You can personalize alert messages to say anything you want. You can customize alert messages to give specific phone numbers and directions to follow when a virus is encountered. For example, a message might say: "Warning: Contact Virus Specialist Elaine Smith at extension 675."

- *Virus service plan.* PC DOS 7's virus update plan is a flexible service plan that includes quarterly updates and a toll-free electronic bulletin board that provides up-to-date information and protection against the latest viruses. If you happen to get a virus that has not been seen before, you can contact IBM AntiVirus Services. This service gives you information for detecting and eliminating the virus as quickly as possible.

Online Help System

The PC DOS Viewer is an online publication viewer. It lets you search, view, and print information in online books created by the OS/2 IPF Compiler. With PC DOS 7, you can now easily and quickly access online information (Figure 6).

This versatile viewer includes features like hypertext links, extended print, and search capabilities. It uses a subset of the information presentation facility (IPF) standard format and can read other books that use this format (`.INF` extension).

Features in the PC DOS online help facility are:

- *Multipane viewer,* which lets you view the table of contents along with the item.

- *Hyperlinks,* highlighted words or phrases that link you to related subjects and topics.

- *Services,* which let you search by keyword, print one or more sections, or copy information to an ASCII file.

- *Options,* which let you change the way information is displayed on your screen by selecting from Expand one level, Expand branch, Expand all, Collapse branch, Collapse all, Contents, and Index.

Included in PC DOS 7 are three online books:

- *PC DOS Command Reference* (CMDREF.INF)
- *REXX Information* (DOSREXX.INF)
- *PC DOS Error Messages* (DOSERROR.INF)

To create an .INF document that can be viewed by the DOS Viewer, you must use the OS/2 Toolkit, which contains the OS/2 IPF Compiler. The OS/2 Toolkit is included in The Developer Connection for OS/2, a subscription service available by calling (800) 6-DEVCON (800-633-8266).

PC DOS handles only a subset of the IPF tags documented for OS/2. Even though PC DOS can view OS/2 .INF files, it cannot display OS/2 books that use tags not supported by the DOS Viewer. IBM's *Information Presentation Facility Guide and Reference,* order number S10G-6262, gives detailed information for using and creating IPF-based books.

Docking Support

Mobile users who have plug-and-play-enabled hardware can take advantage of "cold," "warm," and "hot" docking support. Indeed, PC DOS 7's docking support can eliminate the need to reboot your docking station whenever you dock your mobile PC. Whether you need to reboot will depend on what is connected to your docking station.

Cold docking means that when a mobile PC is docked, it must be rebooted in order to recognize any of its associated peripherals. Cold docking support in PC DOS was enhanced with the multi-configuration support implemented in DOS 6.x.

During *warm* docking, a mobile PC is in a suspended state. In this situation, many if not all of the associated peripherals are recognized and activated. In *hot* docking, a mobile PC is fully active when docked and many if not all peripherals are recognized and activated. In PC DOS 7, support has been extended to both warm and hot docking.

DOSDOCK

Accompanying this new docking support is the ability to configure a mobile PC to execute a pre-defined command, batch file, or program when either docking or undocking. This new function, called DOSDOCK, executes at the DOS command line. It loads and remains resident, waiting to dock or undock. When the event occurs, DOSDOCK examines the environment for the parameters and (depending on the plug-and-play interrupt received) tells DOS to dock or undock. A message informs you that the support is executing. DOSDOCK greatly enhances usability and makes PC docking and undocking simple and painless.

Docking support cannot be activated while running Windows. However, PC DOS 7 notifies you, in a popup Windows message, that you must shut down Windows to allow DOSDOCK to begin.

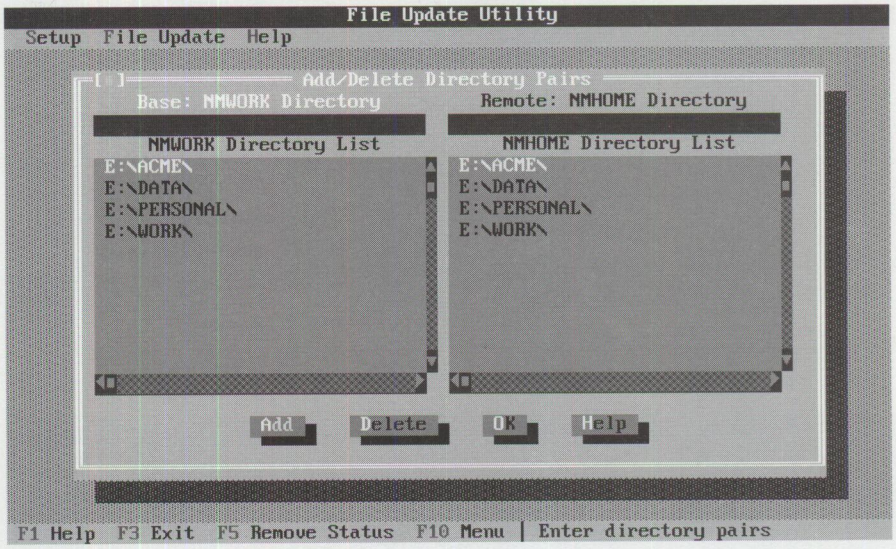


Figure 7. File Update Program

DYNALOAD

With a new command, DYNALOAD, you can dynamically load certain device drivers from the DOS command prompt without modifying your CONFIG.SYS file and restarting your computer. DYNALOAD does not run under Windows, in virtual DOS machine (VDM) mode, or while a task swapper is active.

The *PC DOS 7 Command Reference* lists the drivers shipped with DOS that are supported by DYNALOAD. However, you may find that many additional device drivers are DYNALOADable. Because there are no standards for loading DOS device drivers after the CONFIG.SYS file is processed, the rule of thumb is: Attempt to load the device driver. If any errors occur, or the driver does not behave properly, do not use DYNALOAD with that device driver. Drivers that DYNALOAD definitely cannot load include those that require DOS system initialization, execution via the CONFIG.SYS file, and Block Device Drivers. (IBM has tested only the drivers listed in the *PC DOS 7 Command Reference*.)

File Synchronization

PC DOS 7's File Update (Figure 7) keeps your work current–wherever your work resides. This new full-screen utility automatically updates files between two personal computers, enabling you to easily synchronize files between your mobile and desktop computers. You don't have to remember which files you have changed–just let PC DOS's File Update synchronize the files!

The PC DOS File Update utility provides filters for ignoring non-data files and for synchronizing files between two computers. To use File Update, you need to connect two computers, via parallel or serial port, LAN connection, or even "sneaker net" (on diskette). File Update's full-screen and command-line interface works with any connection that shows a drive letter.

PC DOS 7's INTERLNK utility lets you connect two PCs using a parallel or serial cable. To use INTERLNK:

1. Connect two systems via either a serial or parallel cable (the parallel transfer rate is much higher than serial).
2. On both PCs, include the line DEVICE=C:\DOS\INTERLNK.EXE in the CONFIG.SYS file.

Before running INTERLNK/INTERSVR:
Server has A: and C: drives
Client has A: and C: drives

After running INTERLNK/INTERSVR:
Client has A:, C:, D:, and E: drives, where
Client's D: drive is Server's A: drive, and
Client's E: drive is Server's C: drive

Figure 8. Using INTERLNK/INTERSVR

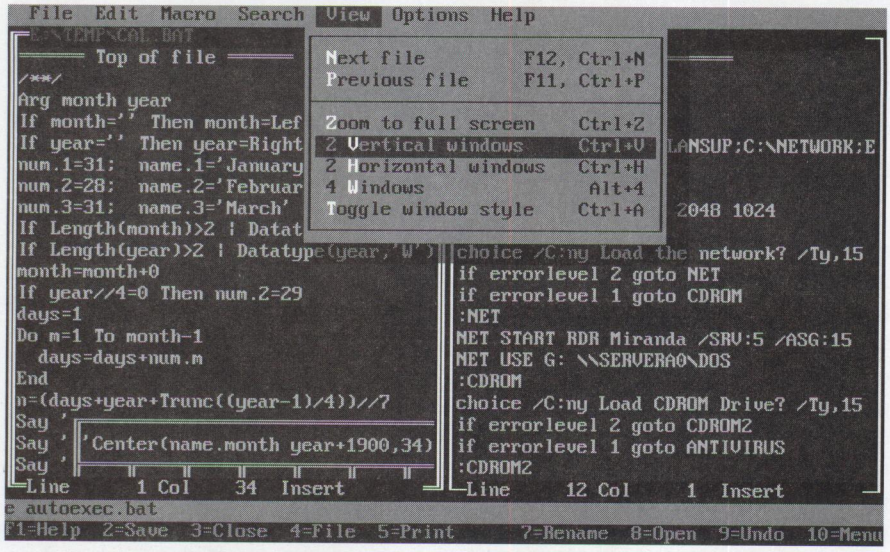


Figure 9. New Look for the E Editor

<DRAW> lets you draw freehand with the arrow keys
<SORT> lets you mark a block of data and sort it
<ADD> lets you add columns
<BOX> lets you draw boxes of arbitrary size

Figure 10. E Editor Macros

3. Reboot both PCs to activate the INTERLNK driver.
4. Establish one PC as the server, and at that computer's keyboard, type INTERSVR.
5. Establish the other PC as the client, and at that computer's keyboard, type INTERLNK.

After you complete the above steps, the server displays a screen that shows the remapped driver letters. Now you are able to access the server's drives from the client. An example is given in Figure 8.

Now you can work transparently at your client computer, accessing the server drives as though they were yours. The server cannot otherwise be used during the INTERLNK connection.

Editor

The E Editor was designed by IBM Research to be a full-function editor, giving you all the basic editing functions. Its power and flexibility enable you to perform many functions found only in high-end word processors.

New pull-down menus and mouse support make the E Editor (Figure 9) easier to use. You can edit multiple files, view them simultaneously, cut and paste between them, set margins and tab stops, reflow paragraphs, search and replace, and undo previous actions. The E Editor's autosave feature helps to ensure that you don't lose any of the data you have created.

Several new features in the E Editor include:

- *Full menu and mouse support*
- *Browse mode*
- *Deletion recovery*
- *132-column support*
- *Expanded search support* (searches multiple files, places its findings in a separate file, and allows you to toggle back and forth between the searched files and findings).
- *REXX and C auto-syntax support* (automatically provided when you use a language-specific keyword, followed by the space bar. For example, suppose you edit a REXX language file and you use the language keyword IF, followed by the space bar. When you press the space bar, auto-syntax takes place–the E Editor automatically puts the associated THEN and ELSE keywords in the file for you. This helps eliminate syntax errors.)
- *ACALC* (a function that supports integer and floating-point arithmetic as well as a rich set of logical operands. This command can also be invoked directly from a DOS command prompt.)
- *Switch capability*
- *Customizable* E.INI
- *Online help*
- *Powerful macros* (see Figure 10)

Another powerful feature in PC DOS 7's E Editor is the ALL command. This command creates the file *.ALL, which contains all instances of a search. Once this file is created, you can use the Ctrl+Q keys to toggle between the *.ALL file and the actual file where the occurrence took place. This feature is very useful when debugging code.

Advanced Power Management

To help minimize the power usage of mobile systems, PC DOS 7 supports the latest standard of Advanced Power Management (APM) 1.1. PC DOS 7 continues to support systems that contain an APM 1.0 BIOS, and supports the two-phase broadcast protocol for standby/suspend requests.

Undelete

PC DOS 7's undelete feature (Figure 11) is a full-function, full-screen program that allows you to re-access files that have been inadvertently deleted. Both DOS and Windows interfaces are provided.

PC DOS 7's undelete feature supports three levels of protection. The lowest level is *Standard*, which gives you the ability to recover files that have not been overwritten.

The middle level of protection is *Tracker*. With this level of protection, DOS leaves the files on the hard disk but marks the file's clusters as available. DOS then records the file's cluster address. As long as the clusters have not been overwritten, the chance of recovery is excellent.

The highest level of protection is *Sentry*. Files protected by Sentry are saved to a hidden directory and can be retrieved by Sentry in perfect condition. If Sentry detects that your hard disk is running out of space, it automatically removes the oldest files that have been stored in the hidden directory, freeing up space for newer ones. You can customize this feature.

With PC DOS 7's file viewers, you can examine the contents of files prior to undeleting them. Files are shown in their native format, when it can be determined; otherwise, they are shown in either text or binary format. Windows file viewers are available when using the Windows undelete interface.

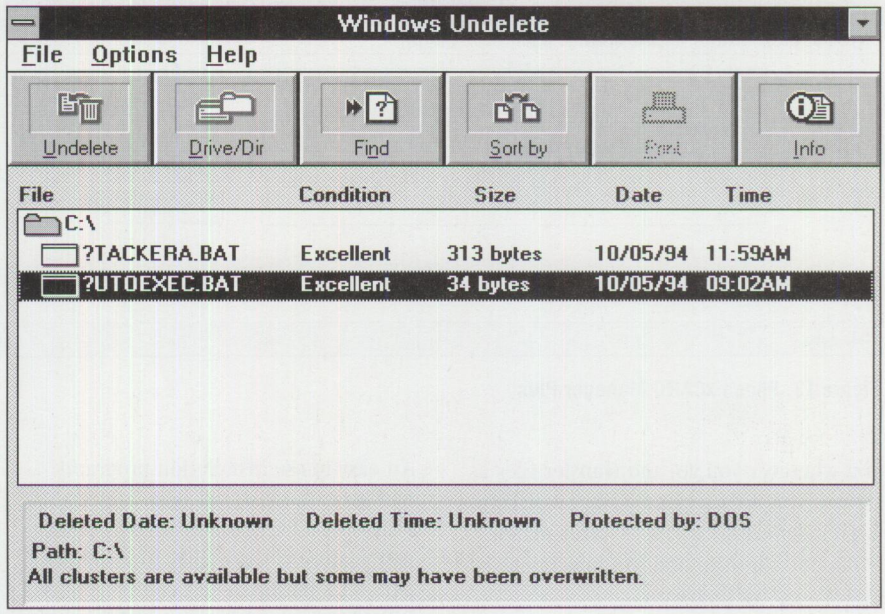


Figure 11. Undelete Program

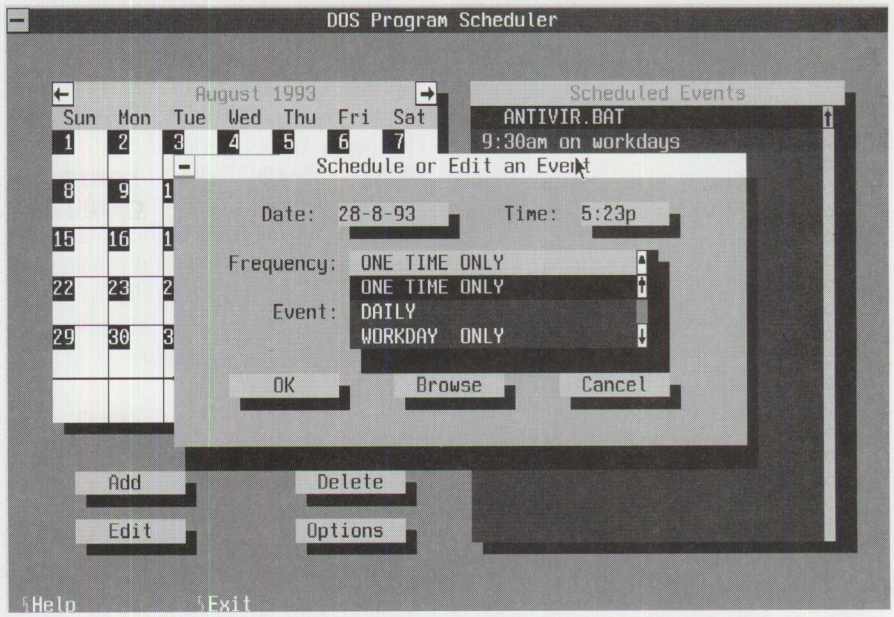


Figure 12. Program Scheduler

Program Scheduler

PC DOS 7 includes a full-functioning program scheduler (Figure 12) for running programs or DOS commands automatically.

It's great to have tools such as BACKUP, DEFRAG, and AntiVirus, but it's not so great to have to bide time waiting for them to finish running. With the easy-to-use calendar interface in PC DOS 7's program scheduler, you can schedule all these utility programs, as well as any other DOS programs or commands, to run at a certain time.

REXX Programming

REXX has been added to PC DOS 7 as the programming language of choice. REXX for PC DOS includes utilities and REXX commands for designing powerful REXX programs that can also run on other operating systems such as OS/2 Warp.

PC DOS 7 REXX is a superset of the standard REXX programming language with some new functions added. PC DOS 7's REXX support includes:

- *Standard and advanced function, .BAT file commands, and arithmetic operations.*
- *A REXX-aware kernel.* The kernel checks the syntax in the first two bytes of a .BAT file. If it finds a REXX comment, it launches the REXX interpreter and executes the program. If it does not recognize the .BAT file as a REXX program, the kernel processes it as a batch file.
- *Portability.* REXX is portable to other operating systems such as OS/2, AIX, and VM.

PCMCIA Support

PC cards of all types are expanding the definition of mobile computing. Storage cards let you take your information with

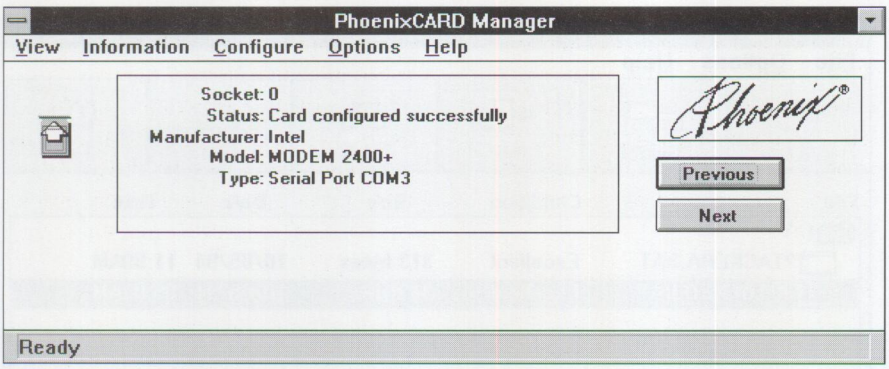


Figure 13. PhoenixCARD Manager Plus

you wherever you go, and transfer files as though you are carrying a floppy diskette. Fax/modem cards and LAN cards make communications to and from portable systems easier than ever.

As shown in Figure 13, PC DOS 7 comes with PhoenixCARD Manager Plus (PCM+), a set of drivers, utilities, and applications designed to provide maximum function for a wide range of PCMCIA cards that conform to the standards of the Personal Computer Memory Card International Association.

In the past, installing or configuring a PCMCIA card was often messy and frustrating. Each time you removed or inserted a PCMCIA card, you had to use a series of commands to alert the computer, and then the system had to be manually reconfigured to accept the new card. Despite all this, frequently the computer still could not recognize the new card.

With PC DOS 7, all this has changed. Instead of a manual full of new commands filled with arcane utility switch options, PC DOS 7 gives you simple menu selections through either a DOS or Windows interface. Most cards—whether they are flash memory, static random access memory (SRAM), network, fax/modem, or IDE hard drives (ATA)—are configured automatically by PC DOS 7. On the rare occasions when you need to set up or reconfigure a card, PC DOS 7 has an easy-to-use setup and configuration program.

Do you need to swap out a card in the middle of a program? Go ahead! PC DOS 7 can set up and reconfigure on the fly, so there is no need to back out of an application and begin again each time you want to use a new card.

An easy-to-use installation program and user interface has been added to PC DOS 7's PCMCIA support. Another new feature is the ability to recognize DOS 6.x's multiple boot configurations. PCMCIA's installation program is now fully aware of multi-configuration and makes sure that the installation is being performed with the desired boot configuration. Each PCMCIA component recognizes this support and uses this information accordingly.

Program Association and Hot Pluggability

One of the nicest features in PC DOS 7's PCMCIA support is the ability to associate cards with programs. With this feature, any time you insert a card that has been associated with a program, the system runs that associated program.

For example, suppose you associate a fax program with a given fax card. When you insert the fax card into the PCMCIA slot, PC DOS 7's hot-pluggability recognizes the card. Then PC DOS 7 determines that the card has been associated with the fax program in your computer and runs that fax program.

New Features in PCMCIA Support

New features in PC DOS 7's PCMCIA support include:

- A setup program for novice users to install PCMCIA support, while also providing custom features for the sophisticated user
- Support for Microsoft's Flash File System II (FFSII)
- Easy-to-use information/configuration utilities for managing PC cards in both DOS and Windows versions
- Hot insertion and removal of PC cards without rebooting
- Support for power management
- Support for the largest number of PCMCIA cards: Flash Memory, SRAM Memory, Network/LAN, fax/modem, ATA, hard disks, SCSI, and other I/O cards
- Card and Socket Services 2.1 compliance
- Centralized initialization file (`.INI`) support for all components
- Support for DOS 6.x's multiple configuration
- FDISK utility that supports multiple partitions
- Super Client Driver that supports definable card lists for minimal memory requirements
- Enhanced resource detection in the Resource utility
- Improved error reporting mechanism

Following are details about some of these new features.

Socket Services

Socket Services is the application programming interface (API) for all card support. Socket Services is a DOS-loadable driver that can be loaded in the `CONFIG.SYS` file or run as a TSR from DOS. Socket Services works with any OEM 2.0-level socket services in any OEM or IBM system. It also works with existing PCMCIA cards that meet the 2.1 specification. Socket Services complies with the PCMCIA Card Services 2.1 interface specification as defined by the PCMCIA Board.

Additionally, PC DOS 7 includes the socket services required to support the following PCMCIA controllers:

- Cirrus Logic CL-PD6710
- Cirrus Logic CL-PC6720
- Databook 86082
- Databook 86082A
- IBM Stinger
- Intel 82365SL (A-step, B-step, and final)
- Ricoh RF5C266/RF5C366

Card Services

Card Services is operating-system code that provides a standard API for clients.

PC DOS 7 incorporates PCMCIA Card Services at the 2.1 level.

Super Client Driver

The Super Client Driver is a collection of client drivers that perform PCMCIA card configuration. After configuration, the PCMCIA card operates exactly as though it was a permanent component when the system was started.

PC DOS 7's Super Client Driver consists of a set of client drivers that is determined at link time. Each individual client driver can either be linked separately or with a group of other client drivers.

The Super Client Driver does not support every PC card in the industry. If the Super Client Driver does not support a PCMCIA card, then a client driver must be provided by a third-party supplier.

Information Utility Program

PC DOS 7's information utility program displays the status of each PCMCIA card that has been installed into the PCMCIA card socket, advising you whether a socket is empty, or whether the socket is in the process of configuring a card. If a card has been installed, the information utility displays the name and information about the PCMCIA card. In the case of a non-configurable card, a message is displayed informing you of that status. This utility has DOS and Windows interfaces.

Super Memory Technology Driver

PC DOS 7's Super Memory Technology Driver (MTD) is a DOS driver designed to support read, write, erase, and copy functions for PCMCIA memory cards. An MTD is needed for each specific memory technology that requires read, write, erase, and copy functions. It is similar in design to PC DOS 7's Super Client Driver and consists of a collection of MTDs that operate only with Card Services.

Virtual Driver for FAT Block Devices

In PC DOS 7, the Virtual Driver for File Allocation Table (FAT) Block Devices supports ATA-compatible fixed disks and SRAM cards formatted in a FAT-structured format.

Windows VxD Driver

The Windows VxD Driver in PC DOS 7 allows fax and modem PCMCIA cards to be fully operable under all Windows sessions when inserted into a PCMCIA socket. Card configuration is always performed under Windows. However, Windows allows the fax/modem cards to be available only to the current Windows program. The VxD driver resolves this restriction and makes the fax/modem cards available to every Windows session.

SRAM Format Utility

The SRAM format utility program formats SRAM PCMCIA cards being accessed as either drive A or drive B. This utility is necessary because a DOS format program limits the capacities available for drives A and B. SRAM cards can also be made bootable by using this utility program.

FAT Diskette Emulation

A driver is provided in PC DOS 7 for FAT diskette emulation. When this driver loads, it emulates a diskette drive on the specified socket, then registers it with Card Services as a memory client.

Copy/Erase for FLASH PC Cards

PC Copy and PC Erase for FLASH PC Cards are special utilities that are not part of PC DOS 7 but are available for OEM customers from their OEM sales representatives on an as-needed basis.

Pen Support

With its PenDOS support, PC DOS 7 gives PC hardware manufacturers the ability to include pen enablement for no additional cost–it's part of the PC DOS operating system. PC DOS 7's PenDOS supports existing, unmodified, mouse-aware DOS applications as well as pen-aware applications. The pen extensions offered in PC DOS 7 are mouse emulation, gesture recognition, numeric recognition, writing window, and a popup soft keyboard.

Mouse emulation enables any current existing DOS mouse-aware application to function with a pen, just as it would with a mouse.

Gesture recognition brings common editing gestures to your current DOS applications (if those applications support the gestures "undo," "cut," "paste," and so on). For example, if you are using a word processing program for DOS that is mouse-aware, you can use the pen to pull down menus, select items, or highlight text and then make an X gesture to delete the highlighted text.

Numeric recognition enables pen-aware applications or standard DOS applications to accept numeric handwritten text. Numbers can be handwritten directly into fields. If using standard DOS applications, a writing window must be displayed. Once it is displayed, numbers can then be handwritten, recognized, and sent to the DOS application.

Another PenDOS extension in PC DOS 7 is the *popup soft keyboard*. This feature displays a keyboard on the screen, which can be used to send keystrokes to any DOS application.

The only Pen extension not shipped in PC DOS 7 is alphanumeric handwriting recognition. This extension is available in the PenDOS operating system. A coupon in the *PC DOS 7 User Guide* enables PC DOS 7 users to upgrade to the full-featured PenDOS operating system for $39.95 (a $40.00 savings).

Performance Improvements

PC DOS 7 has undergone extensive performance optimization, focusing on memory usage. Other areas that have been optimized include 386/486 performance, interrupt handling, command processor tuning, and character device handling.

Other Items in PC DOS 7

PC DOS 7 continues to feature several items originally included in PC DOS 6.3.

Clean Boot–PC DOS 7 has an option at bootup called clean boot. When you see the DOS startup screen and press F5, your system bypasses the `CONFIG.SYS` and `AUTOEXEC.BAT` files, sending you directly to the `C:>` prompt. To bypass both those files and the `DBLSPACE.BIN` (compression driver) file, press Ctrl+F5.

Interactive Boot–Interactive boot lets you step through your `CONFIG.SYS` and `AUTOEXEC.BAT` files. When you press F8 in the DOS startup screen, the system steps through both files line-by-line, prompting you to enter `Y` or `N` after each line.

Selective Boot–PC DOS 7 lets you set up multiple configurations within a single computer. This is done within your `CONFIG.SYS` and `AUTOEXEC.BAT` files by placing groups of statements into configuration blocks. When you boot, PC DOS

```
echo off
choice /c:yn Do you want to copy AUTOEXEC.BAT?
if errorlevel 2 goto exit
if errorlevel 1 goto doauto
:doauto
copy autoexec.bat autoexec.bak
:exit
cls
```

Figure 14. Sample Use of the CHOICE Command

displays a customized menu indicating the different available configurations, then asks you to choose the configuration you want to run. This feature provides flexibility when managing multiple configurations, especially in office environments where one computer might perform various functions.

Improved SmartDRV–PC DOS 7 has an improved SmartDRV disk cache that increases the performance of hard-disk and CD-ROM drives. Two major improvements in SmartDRV are:

- The addition of *lazy writing*, also called *delayed writing*, which stores information to be written to the hard disk and writes it to the disk later, when the system is less busy. This results in significant performance improvements for write operations.
- SmartDRV is now executable (`*.EXE`), allowing you to change, enable, or disable caching on the fly.

Disk Defragmentation–Defragmentation reorganizes files on your hard disk so that each file is intact rather than fragmented. Defragmentation speeds up data access and retrieval. PC DOS 7's defragger uses extended memory to defragment even large hard disks safely and efficiently.

Interactive Batch Processing–Interactive batch processing brings the step-through feature found in interactive boot to batch files. Its functions are similar to interactive boot, except that you do not have to reboot your system in order to step through a batch file. Instead, simply type `command /y /c filename.bat` where `command` starts a new copy of the PC DOS command interpreter, `/y` steps through the batch file, `/c` executes the command and returns, and `filename.bat` is a sample batch file name.

The `command` feature is handy for debugging a failing batch file. When you execute a batch file in interactive mode, the system steps through each batch command individually, asking you whether you want to execute that command (yes or no). With this feature, you are able to observe where the batch file fails.

In addition to using interactive batch processing for debugging batch files, you can also use it to step through a batch file to omit certain commands.

CHOICE–With the `CHOICE` command, you can get user input from batch files. By putting `CHOICE` in a batch file, you can specify what to prompt the user. The user types in a response that is passed to your batch file in an error-level return code. Within the batch file, you can now use the user's response to make conditional branches. Figure 14 shows an example of using the `CHOICE` command in a batch file.

DELTREE–The `DELTREE` command lets you delete a whole sub-tree or a hierarchical directory structure of files at once. If you want to delete a directory, you no longer need to delete all the files and subdirectories first. Instead, you can specify `DELTREE` followed by the name of the directory you want to remove. PC DOS 7 then deletes all subdirectories and files below and including the level of the directory you specified.

MOVE–The `MOVE` command lets you move a file or group of files from one directory to another or move a whole directory hierarchy from one directory to another directory. The files in the original directory are placed in the new location, then deleted from the original location.

Installation Enhancements–PC DOS 7 has a "smart" installation, which recognizes the multi-configuration support delivered in PC DOS 6.1 and MS-DOS 6.0 and 6.2. Until now, when upgrading or changing to DOS 6.x, the installation process did not recognize multi-configuration support, and installation files would be randomly placed within the `CONFIG.SYS` file. In PC DOS 7, installation is now aware of multi-configuration support and files are added to the `CONFIG.SYS` file correctly.

Additional installation enhancements include the following:

- PC DOS 7 accesses CD-ROM drives by using MSCDEX 2.25.
- An option enables viewing or changing any modifications made to the `CONFIG.SYS` and `AUTOEXEC.BAT` files. Once the installation is complete, the `CONFIG.SYS` and `AUTOEXEC.BAT` files are displayed side by side in the E Editor. At this point, you can edit either file. PC DOS 7 places comments in these files to show you what has been added or changed.
- The `DOSKEY` command-line statement is now automatically added to your `AUTOEXEC.BAT` file.
- Mouse navigation is supported.

QCONFIG Enhancements–The `QCONFIG` utility in PC DOS 7 provides information about your computer system, including what kind of memory it has and how much of it is available for your programs. `QCONFIG` loads hard-disk information at initialization. Support has been added for new adapters and planar boards.

No-Swap DISKCOPY–PC DOS 7's no-swap `DISKCOPY` copies the entire contents of one diskette to another by using the system's hard disk as temporary storage. This new feature eliminates the task of swapping diskettes multiple times when using `DISKCOPY`.

File Overwrite–PC DOS 7's file overwrite feature prompts you before overwriting files that have identical names. This feature keeps you from accidentally deleting files. PC DOS 7 has a switch that allows you to turn this feature off.

Default Prompt–The default prompt in PC DOS 7 displays the current drive and path followed by an ">".

FIND–The `FIND` command supports wildcards in file names and subdirectories to allow easier searches.

COMMAND.COM Switch–The `/o` switch has been added to the `COPY`, `XCOPY`, and `MOVE` commands to disable the overwrite prompt.

In PC DOS 6.x, whenever you try to overwrite an existing file, the system prompts you that the file already exists and asks if you want to overwrite it. PC DOS 7 includes the `/o` switch to disable this prompting.

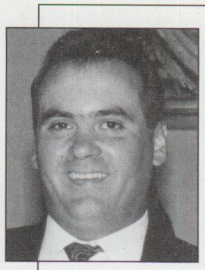
Additional Keyboard–Additional keyboard support in PC DOS 7 includes the new German keyboard standard, DIN 2137, as well as additional support for Eastern European countries.

PC DOS 7 Documents

The following information is available from IBM:

- *PC DOS 7 Technical Update Video Cassette.* To obtain, call (800) 456-1426.
- *PC/MS DOS Feature Comparison Chart.* To obtain, call (800) IBM-4FAX and specify document number 1530.
- *PC DOS 7 Specification sheet.* To obtain, call (800) IBM-4FAX and specify document number 2435, or order IBM publication number G221-4320.
- *PC DOS 7 What's New sheet.* To obtain, call (800) IBM-4FAX and specify document number 1532.
- *PC DOS 7 Supplemental Features.* To obtain, call (800) IBM-4FAX and specify document number 1543.

Kay Lee is a staff PC DOS product planner within IBM's Personal Software Products division, Boca Raton, Florida. She entered IBM as an information developer after graduating from the University of Florida with a BS in Journalism and a minor in Computer Science. Kay has an extensive background in quality assurance, project management, technical writing, and product planning.



Nestor Miranda, who works for Electronic Data Systems, is a PC DOS product planner within IBM's Personal Software Products division, Boca Raton, Florida. In addition to product planning, he has been instrumental in building product awareness for PC DOS through writing publications, providing media support, and attending business shows. He joined the IBM account in 1992 as a lead in the Distributed Application Environment development team, then joined the Mobile, Voice, and Pen organization as a planner for PenDOS 2.2. Nestor has a BS degree in Computer Science and is working toward his MS in Data Communication Management from Barry University. Nestor can be reached via the Internet at `nestor_miranda@vnet.ibm.com`.

OS/2 Warp Boot and Recovery Options

This article is about OS/2 Warp's archive and recovery feature, which not only protects users from unrecoverable situations but also offers flexible boot options.

Think back a few years ago—or maybe just last week. You're at your desk, thinking to yourself, "It's four o'clock on Friday, and I'm almost out of here!" Then it happens—the phone rings. The person at the other end is in a panic: "I accidentally powered my machine off. Now, when I power it back on, all I get is a blank screen. I have to get my report out by five." Immediately you think to yourself: "I'm sure the `.INI` files or the Desktop is corrupted. I'll have to go over there and rebuild them. They probably don't have a backup. Then I'll have to manually rebuild all of their objects. And I was almost out of here!"

If this scenario hits close to home, I have great news for you.

OS/2 Warp has a robust archive and recovery feature. Not only can you keep backup copies of critical OS/2 system files, but you can also select different `CONFIG.SYS` files during the boot process.

This article gives you the details and the know-how to protect your users from unrecoverable situations and to provide them with flexible boot options.

Edward Duhe'
IBM Corporation
Roanoke, Texas

Archiving

To enable your OS/2 Warp system for archiving, you must change the default settings in the Desktop's Settings notebook. To open the Desktop's Settings notebook, press the right mouse button while pointing to a blank area of the Desktop, then select Settings from the popup menu. Once the Settings notebook has opened, go directly to the Archive page by clicking on the Archive tab with the left mouse button (see Figure 1).

The Archive page is where you configure the archiving functions of your OS/2 Warp system. Let's look at the different options that can be set.

- The *Create archive at each system restart* option automatically archives the key files that OS/2 Warp needs in order to restart. Three generations of archive files are retained; the oldest generation is overwritten each time OS/2 Warp is restarted. OS/2 Warp archives the files during the boot process, so activating the archive process causes your boot time to increase and requires more disk space to hold the files.

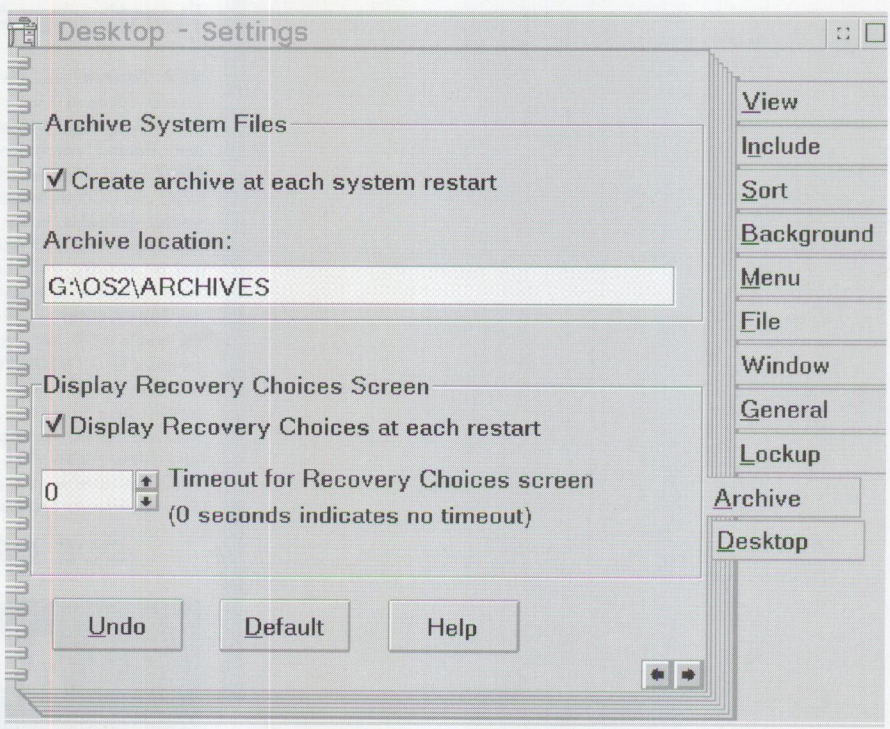


Figure 1. Desktop Settings Notebook Archive Tab

In a typical environment, you don't need to archive your system files each time you restart OS/2 Warp. The critical system files may not change very often. Once you have your OS/2 Warp system configured and customized the way you like it, you can archive the files and then turn off the archiving feature. This gives you a set of files to recover from, without constantly incurring the overhead of archiving during every restart.

- The *Archive location* option specifies the location of the archived files. The default is to store the files in the \OS2\ARCHIVES subdirectory, but you can specify any other place.

  If you change the archive location, your old archive files are not lost. Resetting the archive location to the location of your old archive files enables you to use the old files.

- The *Display Recovery Choices at each restart* option is one of two ways to access the Recovery Choices screen, which is used to restore your archived files (see Figure 2). When you check this option, the Recovery Choices screen displays every time you start your OS/2 Warp system. You can also access the Recovery Choices screen during startup of OS/2. When the white block is displayed in the upper left corner of the screen, press Alt+F1. This brings up the Recovery Choices screen, for this time only. If you want to see the Recovery Choices screen every time you boot OS/2 Warp, you must check the "Display Recovery Choices" option.

If you display the Recovery Choices screen during each startup, you can also specify how long to wait for input on that screen before continuing with a normal boot. The wait can be from zero seconds (indicating no wait) to 999 seconds.

The What and Where of Archiving

Now that you have enabled archiving, let's take a look at what gets archived and where it is stored.

Several files are critical to successfully start up OS/2 Warp. By default, OS/2 Warp copies the files OS2.INI, OS2SYS.INI, CONFIG.SYS, AUTOEXEC.BAT, STARTUP.CMD (if it exists), and OS2INIT.CMD, as well as the \DESKTOP subdirectory structure, to the specified archive subdirectory.

```
                    RECOVERY CHOICES

Select the system configuration file to be used, or enter the option
corresponding to the archive desired.

ESC - Continue the boot process using \CONFIG.SYS without changes
C   - Go to command line, (no files replaced, use original CONFIG.SYS)
V   - Reset primary video display to VGA and reboot
M   - Restart the system from the Maintenance Desktop (Selective Install)

Choosing an archive from the list below replaces your current CONFIG.SYS,
Desktop directory, and INI files with older versions.  These older versions
might be different from your current files.  Your current files are saved in
\OS2\ARCHIVES\CURRENT.

1) Archive created 12-20-94  2:03:20PM
2) Archive created 12-19-94  10:55:22PM
3) Archive created 12-19-94  8:46:52PM
X) Original archive from INSTALL created 1-1-80  11:18:50AM
```

Figure 2. Recovery Choices Screen

The files that are archived are defined in the OS2.KEY file, located in the \OS2\ARCHIVES subdirectory. The OS2.KEY file is an ASCII file that can be edited, enabling you to add more files to be archived. For example, Figure 3 shows an OS2.KEY file with PROTOCOL.INI and EXPLORE.INI added. These files are archived along with the default files.

You can add any file to the OS2.KEY file, but the more files it has, the more time it takes to boot, and the more disk space it needs.

Figure 4 is an example of the directory tree for the \OS2\ARCHIVES subdirectory.

In Figure 4, several files and directories are in the \OS2\ARCHIVES subdirectory:

- The first subdirectory, \OS2\ARCHIVES\CURRENT, is where OS/2 Warp stores your currently active system files and Desktop structure, before restoring one of your archived sets of system files and Desktop structure.
- The next subdirectory, \OS2\ARCHIVES\0X, contains the system files and Desktop that were created when you initially installed OS/2 Warp.
- The next three directories contain your three generations of archived files. The \OS2\ARCHIVES\01 subdirectory contains the oldest generation of your archived files and Desktop, while the \OS2\ARCHIVES\03 directory contains the most recent.

Figure 5 is an example of the contents of a generation subdirectory (in this case,

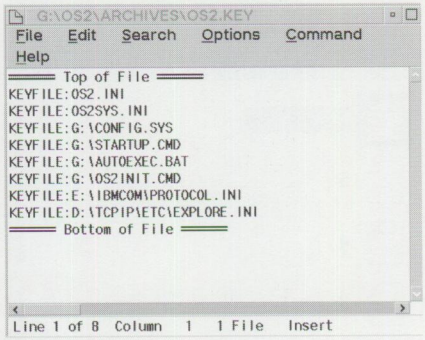


Figure 3. OS2.KEY File

generation 3). The files that are named with numbers are the archives. The KEYS.$$$ file provides the key that maps archived files to their original files. For example, the file G:\OS2\OS2.INI is archived as G:\OS2\ARCHIVES\03\0, the file G:\OS2\OS2SYS.INI is archived as G:\OS2\ARCHIVES\03\1, and so on for all the files defined in the \OS2\ARCHIVES\OS2.KEY file. The G:\OS2\ARCHIVES\03\DESKTOP subdirectory contains a copy of the \DESKTOP directory structure, providing a way to recreate your Desktop.

Restoring an Archive

Now that you have archived all of your critical files, how do you restore them when you need them?

You can restore the archived files through the Recovery Choices screen (shown in Figure 2), which presents you with several options:

- Esc continues the boot process using your current OS/2 Warp system files.

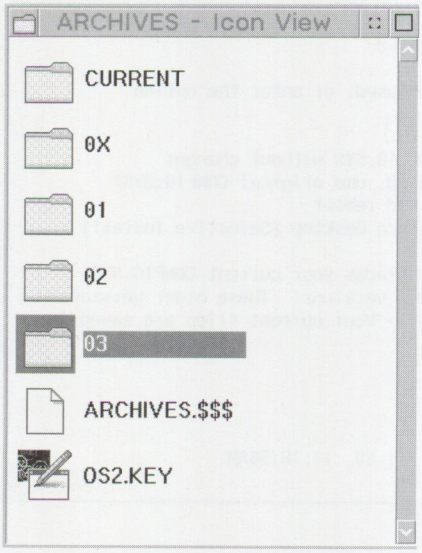


Figure 4. \OS2\ARCHIVES Subdirectory Tree

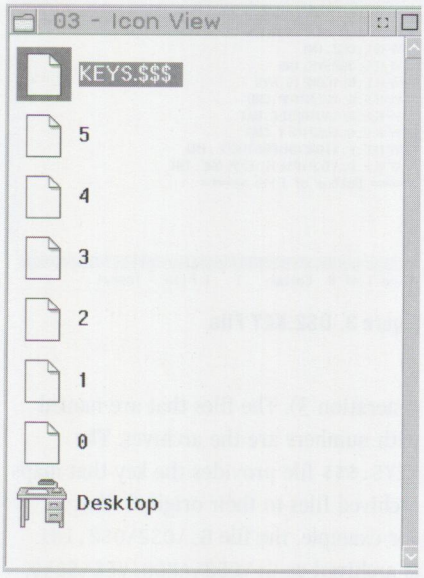


Figure 5. Generation 3 Subdirectory

- C takes you to an OS/2 command prompt. There, you can use a utility like the TEDIT editor to manually change your CONFIG.SYS file.
- V gives you a way to easily reset your system's video back to VGA. Once reset to VGA, you can then install the drivers necessary for your video subsystem.
- M boots your OS/2 Warp system to the Maintenance Desktop. From the Maintenance Desktop, you can run Selective Install. In the event you need to reinstall your OS/2 Warp system, booting the Maintenance Desktop and running Selective Install saves you from having to reinstall the first seven diskettes.
- 1 through 3 correspond to your three generations of archive files. When you select one of these numbers, OS/2 Warp restores the corresponding archived files, and then boots using the newly restored files.
- X restores the system files that were created when you first installed OS/2 Warp.

Choosing CONFIG.SYS Files When Booting

The Recovery Choices feature in OS/2 Warp not only gives you a way to recover your system but also a new way to customize your OS/2 Warp system's boot process.

OS/2 Warp enables the use of multiple CONFIG.SYS files. At boot time, you can select which CONFIG.SYS file you would like to boot with. For example, if you are using a laptop computer that needs certain device drivers when attached to a docking station, but not when it runs stand-alone, you can configure two separate CONFIG.SYS files. Then, through the Recovery Choices screen, you can select which CONFIG.SYS file you want to run.

Whenever you press a single key in the Recovery Choices screen, OS/2 looks for a CONFIG.? and/or an ALTF1?.CMD file in the \OS2\BOOT directory, where ? is the key you pressed. You may use any unique alphabetic character except the letters reserved for options: X, x, C, c, M, m, V, v.

If OS/2 Warp finds a CONFIG.? file in the \OS2\BOOT directory, it boots using that CONFIG.? file instead of the \CONFIG.SYS file.

If OS/2 Warp finds an ALTF1?.CMD file in the \OS2\BOOT directory, it executes the ALTF1?.CMD file after it processes the CONFIG.? file.

If a CONFIG.? file does not exist, but an ALTF1?.CMD file does exist, OS/2 Warp uses your current CONFIG.SYS file, then executes the commands in the ALTF1?.CMD file.

Practical Use

Now that we have spelled out all the features and options, how do you put them to practical use?

To illustrate, let's use the above example–a laptop computer used in two environments. In the first environment, the laptop is attached to a docking station. In the second environment, the laptop is used on the road as a stand-alone computer. You may have different requirements for device drivers and CONFIG.SYS parameters, depending whether your laptop is attached to the docking station or is running stand-alone.

For this scenario:

1. Configure your computer initially for the docking station.
2. Copy the CONFIG.SYS file to the \OS2\BOOT directory, renaming it to CONFIG.D (where D stands for docking).
3. Copy \OS2\BOOT\CONFIG.D to \OS2\BOOT\CONFIG.S (S for stand-alone).
4. Edit the CONFIG.S file and remove the unnecessary device drivers for the docking station. Then save the updated CONFIG.S file.
5. Create files named \OS2\BOOT\ALTF1D.CMD and \OS2\BOOT\ALTF1S.CMD. The contents of these files could be as follows:

   In ALTF1D.CMD:
   ```
   COPY \OS2\BOOT\CONFIG.D
    \CONFIG.SYS
   ```

   In ALTF1S.CMD:
   ```
   COPY \OS2\BOOT\CONFIG.S
    \CONFIG.SYS
   ```

6. Now that your alternate CONFIG.SYS files are set up, you can use the Recovery Choices screen to choose which CONFIG.SYS file your OS/2 Warp system uses at boot time.

   For example, when you select D while the Recovery Choices screen is displayed, OS/2 Warp begins booting with the CONFIG.D file. Next, OS/2 Warp runs the ALTF1D.CMD file, which copies the CONFIG.D file atop the existing default CONFIG.SYS file. Subsequent boots are now set up for the docking station.

   When you are ready to use your laptop as a stand-alone computer, you simply return to the Recovery Choices screen and press S. This reverses the process, and now OS/2 Warp boots for stand-alone use.

Customizing the Recovery Choices Screen

Now that you have created new boot options for your OS/2 Warp system, you may want to create menu references for those boot options in the Recovery Choices screen.

The Recovery Choices screen is created from three different ASCII files in the `\OS2\BOOT` directory:

- `ALTF1TOP.SCR` contains the top part of the Recovery Choices screen. You can use an ASCII editor to change this file if you wish (see Figure 6).
- `ALTF1MID.SCR` contains the screen entries for your archived files (see Figure 6). This file is updated each time your OS/2 Warp system archives your system files. Any changes you make to this file are overwritten by OS/2 Warp. It's best to leave this file as is.
- `ALTF1BOT.SCR` appends to the bottom of the Recovery Choices screen. Typically this file is empty. Edit this file to add the new boot options for selecting `CONFIG.SYS` files. In our scenario, the docking station and standalone options can be added and will display on the Recovery Choices screen (Figure 7).

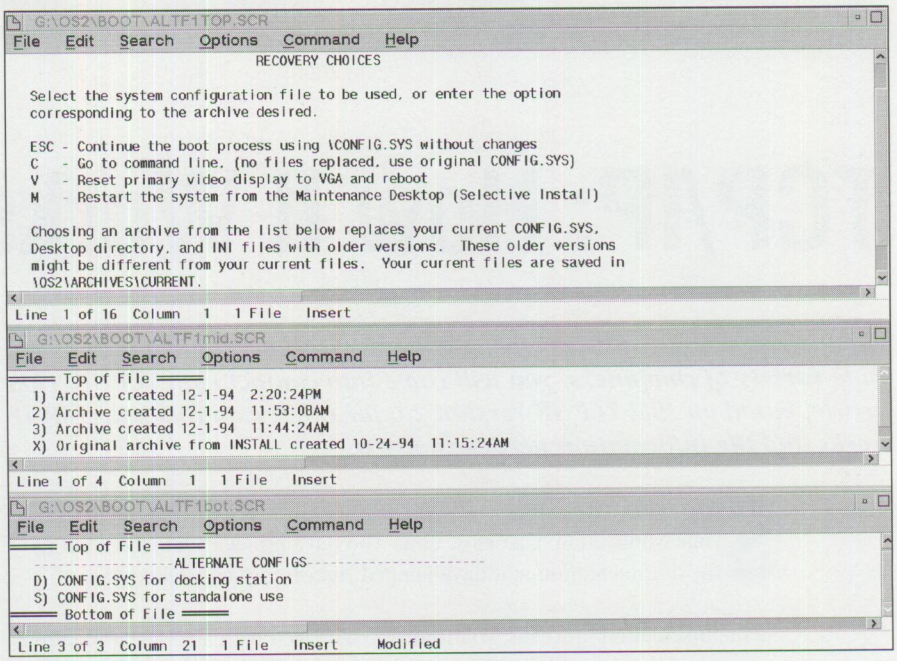


Figure 6. Files That Make Up the Recovery Choices Screen

```
                    RECOVERY CHOICES

Select the system configuration file to be used, or enter the option
corresponding to the archive desired.

ESC - Continue the boot process using \CONFIG.SYS without changes
C   - Go to command line, (no files replaced, use original CONFIG.SYS)
V   - Reset primary video display to VGA and reboot
M   - Restart the system from the Maintenance Desktop (Selective Install)

Choosing an archive from the list below replaces your current CONFIG.SYS,
Desktop directory, and INI files with older versions.  These older versions
might be different from your current files.  Your current files are saved in
\OS2\ARCHIVES\CURRENT.

1) Archive created 12-20-94  2:03:20PM
2) Archive created 12-19-94  10:55:22PM
3) Archive created 12-19-94  8:46:52PM
X) Original archive from INSTALL created 1-1-80  11:18:50AM

------------------ALTERNATE CONFIGS-----------------------
D) CONFIG.SYS for docking station
S) CONFIG.SYS for standalone use
```

Figure 7. Recovery Choices Screen with New Boot Options

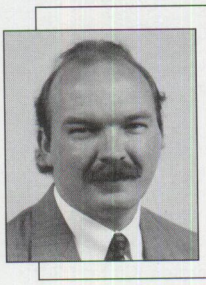
Peace of Mind

OS/2 Warp's Recovery Choices feature provides a great way to protect your OS/2 Warp system from corrupted system files. Maintaining several generations of archives gives you added peace of mind that you will have at least one set of valid files to restore if you need them. And being able to select from different versions of `CONFIG.SYS` files when booting means you can have different customizations for different scenarios such as when you are docked or not docked, network-connected or not network-connected, etc.

Remember the panicked user at the beginning of this article? With OS/2 Warp installed, you have a ready answer: "Reboot your OS/2 Warp, press Alt+F1 while the white block is on the screen, then select 1 from the menu." Problem solved, system restored, and you're out of there!

Reference

This article has presented only a few of the many great new enhancements in OS/2 Warp. For additional information about not only the new boot options of Warp but also performance enhancements, installation enhancements, usability improvements, printing enhancements, new applications, and more, refer to my previous article "OS/2 Warp" in the January/February 1995 issue of this magazine.



Edward Duhe' is a marketing support representative in the IBM Personal Systems Competency Center in Roanoke, Texas. He provides technical support for LAN Distance and OS/2. Edward joined IBM in 1987 as a systems engineer in South Louisiana. He has a BS degree in Business Administration from Louisiana State University and can be reached via Internet at `eduhe@vnet.ibm.com`.

TCP/IP: How It Works

Whether you are using the Internet or trying to communicate with a wide variety of computers, you will come into contact with TCP/IP. This article, based on IBM TCP/IP Version 2.0 for OS/2, explains how TCP/IP works and the information needed to use it.

So, you just received a copy of TCP/IP, and now you are setting it up on your workstation. Like most of us, you have probably skipped reading the documentation and have jumped right into the installation.

As the installation proceeds, you find yourself faced with a series of required parameters that look like nothing you have ever seen before. You try to guess, but your guesses are rejected mercilessly.

You finally break down and read the manual. If you have been working in the world of UNIX, you can follow the manual, but if you have only worked on mainframes and PCs, you will most likely find yourself lost.

That's why I wrote this article–a quick mainframe and PC primer for TCP/IP.

Background of TCP/IP Protocol

TCP/IP, which stands for *transmission control protocol/internet protocol,* is simply a standard for sending and receiving packets of information over some sort of connection. The connection could be Token Ring, Ethernet, dial-up (SLIP/PPP), or even a two-way radio link (such as a packet radio used by ham radio operators).

TCP/IP is a media-independent protocol–it doesn't care what type of wiring or connection you use for moving packets of information from place to place.

Philip Lieberman
Lieberman and Associates
Beverly Hills, California

In truth, TCP/IP is a series of different protocols. The IP portion of the protocol moves packets (sometimes called *datagrams*) from computer to computer. (In TCP/IP terminology, all computers are called *hosts* even if they are just workstations.) The TCP portion of the protocol makes sure that the packets arrive intact and in the proper order. TCP takes care of errors and retransmissions. The gory details about how these different parts of TCP/IP work were spelled out long ago.

TCP/IP packets can contain many different types of information. One of the more interesting uses of TCP/IP packets is as a carrier of non-TCP/IP traffic. The IBM AnyNet product, for example, can move advanced program-to-program communications (APPC), NetBIOS/NetBEUI, and systems network architecture (SNA) packets via the TCP/IP protocol. This AnyNet capability is known as the *Multiprotocol Transport Feature.* With this feature, transparently shooting data packets of different protocol formats from place to place over TCP/IP means that the network administrator only has to worry about routing TCP/IP packets. For programmers, life is also simple–they can create packets in their favorite protocol format and let AnyNet transport them to the other end using TCP/IP.

The TCP/IP Layered Model

TCP/IP is a layered protocol. At the top of the layering model are the application programs that use TCP/IP. Down through the layers are sections that provide

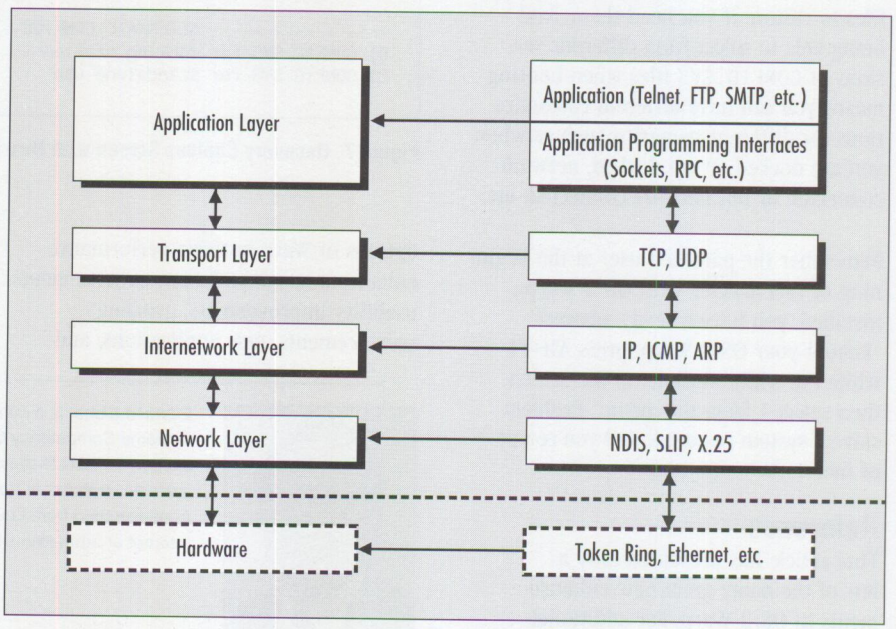


Figure 1. IBM TCP/IP Version 2.0 for OS/2

reliability features and diagnostics. Figures 1 and 2 provide an overview of TCP/IP architecture.

Figure 2 shows that TCP/IP *applications* live in the application layer.

The *transport* layer provides guaranteed error-free communications (TCP) and datagram (UDP) services. As TCP uses the IP layer to route packets, it verifies that the packets arrived in order and intact. If there is a problem with a packet, TCP resends the packet(s) that did not arrive properly. Most programs use the TCP protocol, but some programs that perform their own error connection or don't need this feature use the UDP services instead.

The *internetwork* layer provides the basic service of routing packets between computers. IP is a datagram or packet-level protocol with no error correction or retry logic; the TCP layer uses the IP layer to create a reliable communication link. ARP takes care of mapping between the physical addresses of the adapter(s) and the 32-bit IP address used by all of the upper layers. ICMP controls flow, detects unreachable destinations, redirects routes, and checks for the existence of remote hosts (`ping`).

At the bottom is the *network* or *hardware* layer. TCP/IP can use just about any kind of hardware for its transport. The details of mapping real physical addresses to IP addresses are taken care of above this level.

Layer	Protocols and Functions
Application	Telnet File Transfer Protocol (FTP) Trivial File Transfer Protocol (TFTP) Simple Mail Transfer Protocol (SMTP) Domain Name System (DNS) Simple Network Management Protocol (SNMP) Remote Printing (LPR and LPD) Talk Finger Routed X-Windows System Remote Procedure Call (RPC) Network File System (NFS) Remote Execution Protocol (REXEC) Socket Interfaces
Transport	Transmission Control Protocol (TCP) User Datagram Protocol (UDP)
Internetwork	Internet Protocol (IP) Internet Control Message Protocol (ICMP) Address Resolution Protocol (ARP)
Network (or hardware)	Network Driver Interface Specification (NDIS) Serial Line Internet Protocol (SLIP) X.25 Protocol

Figure 2. Programs and Functions at Each Layer of TCP/IP

Background of TCP/IP Programs

The term TCP/IP denotes more than just a protocol. It normally refers to a packaged suite of protocols and utilities. The number of utilities you receive can vary from the most basic troubleshooting tools (like `ping`, `ARP`, and `hostname`) to a full suite of applications including Gopher, Mosaic, Mail, FTP, Telnet, TN3270, TN5250, and more.

Many add-ons are available if you use TCP/IP. Some of these add-ons are in the public domain and can be downloaded from the Internet, while others are full commercial packages.

UNIX has spawned a series of graphical user interfaces (GUIs) designed specifically for TCP/IP network operation. These GUIs include X-Windows, Motif, and Mosaic. X-Windows and Motif provide event-driven GUI front ends for UNIX computers and for general-purpose programming. Mosaic provides a hypertext information retrieval environment using a relatively simple format called *Hypertext Markup Language* (HTML). Mosaic is used on the Internet to paint pretty pictures and give you hypertext links to data. GUI programmer toolkits are available for all of these environments.

If you want simple, cheap, fast file sharing, you can add Network File System (NFS) to your network. This client and server code, originally developed by Sun Microsystems, allows a computer (server) to share its hard-disk directories with other computers (clients) on the network.

Before you rush out and replace your copies of Novell NetWare and IBM LAN Server with NFS, you need to know that NFS has very limited security and is unaware of OS/2's Extended Attributes (forget about drag-and-drop to any NFS drive). Also, NFS has no file locking unless additional file-locking processes are running. But for quick and dirty file sharing, NFS is about as fast and cheap as you can get.

The Gory Details: Addressing

Addressing is the key to successful TCP/IP operation. When a network adapter card is manufactured, it is given a unique address that can be used with total assurance that it will be the only one in the world with that address. This is known as the *physical address* of the card.

In TCP/IP, you must also give each card an additional address, known as the *internet protocol (IP) address.* The IP address is a 32-bit number that is normally assigned by your TCP/IP network administrator to ensure that there are no accidental duplicate addresses on the network (a bad situation, should it occur).

An IP address is represented as four decimal numbers (each a single byte) separated by decimal points. Each value can range from 0 to 255 (for example: `129.37.5.6`).

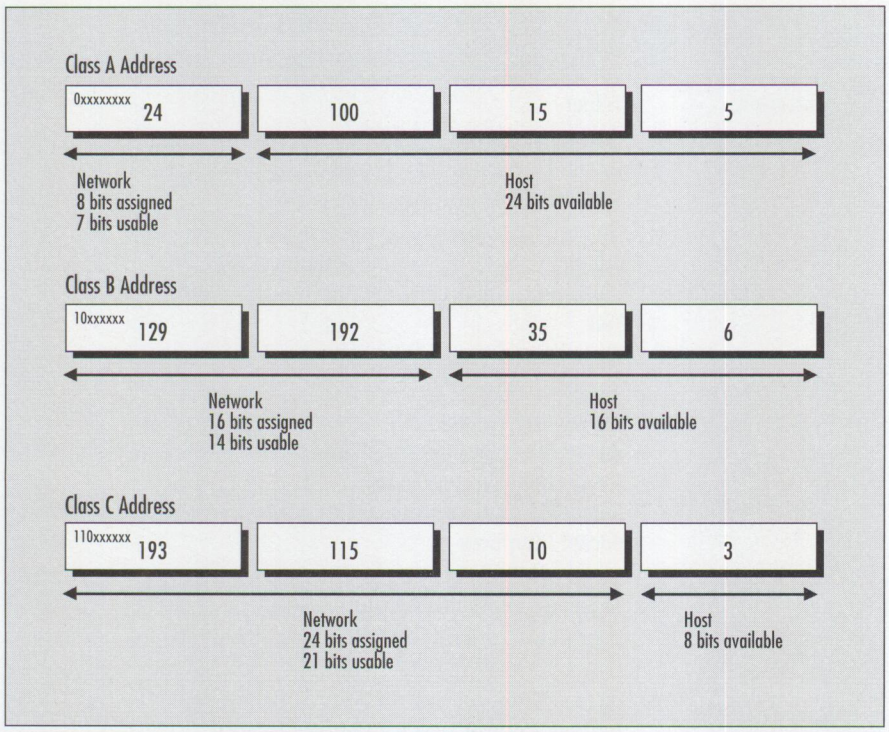


Figure 3. Three Classes of IP Addresses

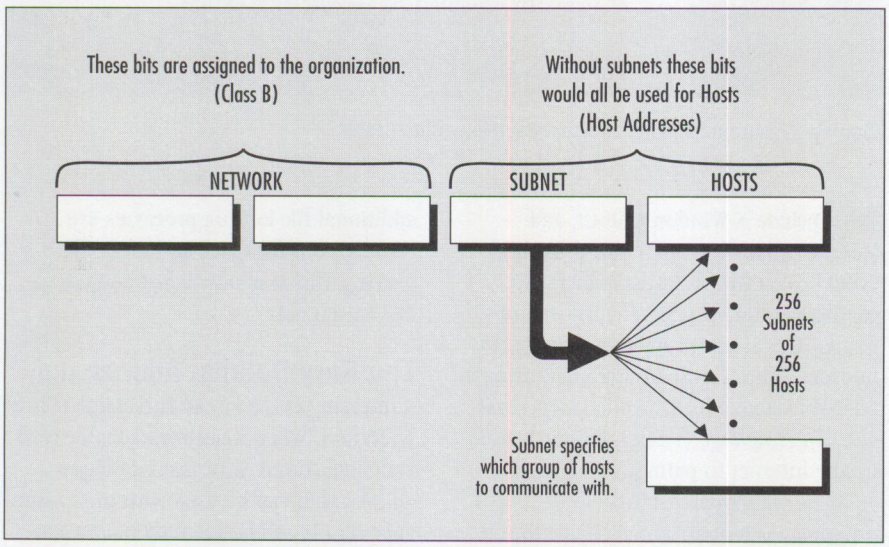


Figure 4. Subnets

If you were to calculate the number of possible addresses that exist with a 32-bit number, you would find that there are: 2^32 or 4,294,967,296 unique addresses. Certainly 4 billion addresses should be enough to address every possible computer in the world. However, things are not that simple.

When the original 32-bit address system was designed, the ranges of IP addresses were broken into ranges known as *classes*. In TCP/IP systems, there are three classes, known as A, B, and C. You can determine the class of your network by examining the first byte of the IP address:

- If the first byte is less than 128, you have a Class A address.
- If the first byte is from 128 to 191 inclusive, you have a Class B address.
- If the first byte is from 192 to 223 inclusive, you have a Class C address.

Figure 3 illustrates the three classes of IP addresses.

The class of IP address defines the number of potential host addresses you can assign. Class A has 24-bit addresses, which can accommodate 16,777,216 total host addresses; Class B has 16-bit addresses (65,536 total host addresses); and Class C has 8-bit addresses (256 total host addresses).

If your network is never going to be connected to the Internet, then the decision of class is open to the administrator. But if there is any possibility of hooking up to the Internet, you will want to get an Internet Protocol (IP) Network Number. You can do this by contacting:

> Network Solutions
> InterNIC Registration Services
> 505 Huntmar Park Drive
> Herndon, VA 22070
> Phone: (800) 444-4345 or
> (703) 472-4777
> Internet userid:
> `hostmaster@internic.net`

Forms can also be obtained via anonymous FTP from `rs.internic.net`. For a network number, fill out the form in `templates/internet-number-template.txt`, and send it to `hostmaster@internic.net`. For details about filling out this and other forms, refer to Appendix B of Craig Hunt's book, *TCP/IP Network Administration* (see the bibliography at the end of this article).

Subnet Masks

In some networks, you may decide to break up the IP address class into smaller groups of addresses. This is accomplished by using a bit-pattern mask on the IP address. This bit-pattern mask is known as a *subnet mask*. The subnet mask allows you to convert potential host bits into additional local network bits. These extra network bits are strictly for your organization's use.

Subnets allow large groups of hosts to be administered in smaller groups. For example, Figure 4 starts out with a typical Class B address that has 16 bits. By using a subnet mask that covers the upper 8 bits of the host address portion, we can create 256 subnetworks (upper 8-bit mask on host bits), each containing 256 hosts (lower 8 bits of host bits). Figure 4 illustrates the use of a subnet mask and subnets. In Figure 4, there are 8 bits in each group.

The subnet arrangement was created for organizations that want to break apart host administration into smaller groups. The use of subnet masks is totally optional. But this does not mean you can leave the subnet field blank (is anything simple?) when installing the TCP/IP software.

If you are not using subnet masks, you must set the bit pattern to 1 in bits that are part of your network, and 0 for bits that are part of your host's field. For example, for a Class B network (16 bits of networks and 16 bits of hosts), the default would be `255.255.0.0`.

Ports

The IP address gets you to a specific computer; but once there, how do you use its resources? The answer is through an additional level of addressing within that computer known as *port addresses* or *port numbers*. The port address is an additional 16-bit number that can represent up to 64 KB contact points within a computer.

You might want to think of the IP address as an address for an apartment building. Within that building, the apartments would be the port addresses (all residing in the same building).

Programs that communicate between TCP/IP-connected computers must specify the IP address and the port number of both their local computer and the remote computer to complete a connection. Depending on the type of communication traffic, the port address may not need to be specified explicitly (most common services have well-known fixed port addresses).

Figure 5 shows some of the most common port addresses and their services. A more complete list can be found in the file `TCPIP\ETC\SERVICES`. The file's location may vary by computer and operating system.

There are two general types of port addresses. First, there are addresses with fixed port numbers for services such as Telnet (terminal emulation) and FTP (file transfer protocol). Second, there are addresses that accommodate dynamic assignment of port numbers. Dynamic port assignments enable multiple conversations between two computers.

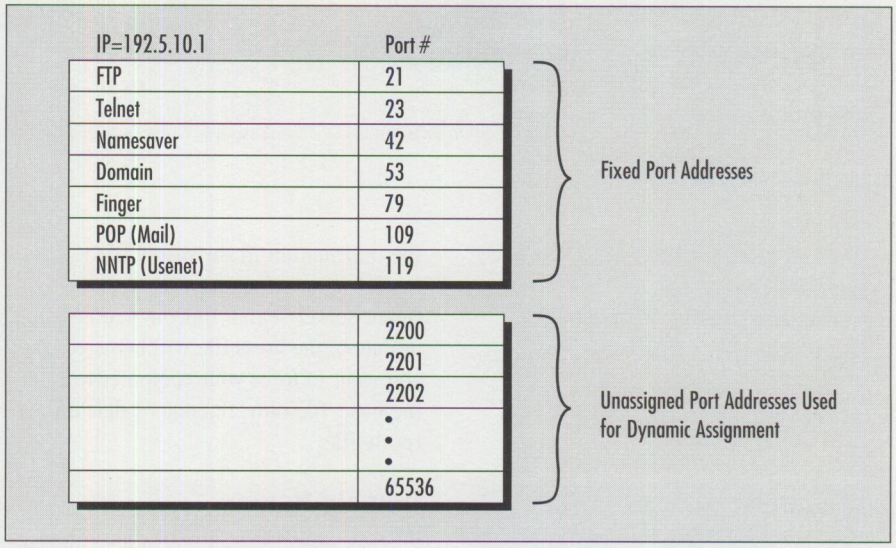


Figure 5. Common TCP/IP Port Addresses

It is common to use both fixed and dynamic addresses for a single connection. To set up a host-to-host terminal session using the Telnet terminal emulator, the originator starts at a fixed address on the target computer, and then moves on to another port. (This is like going to a California Department of Motor Vehicles office—everybody gets into one line to get an appointment to wait in another assigned line.)

At the DMV, you can make an appointment ahead of time to eliminate the first line for appointments. In TCP/IP, this case means you have a dedicated and fixed port address for your transaction. Because this fixed port may be busy, most TCP/IP software waits a while on the fixed port until the port becomes available (so even TCP/IP applications have to wait in line). If the port does not become available, then the program will give up eventually (sort of like having to get back to work after waiting an hour at the DMV during your lunch break).

As you might have guessed, it is common for a computer to carry on multiple conversations through a variety of ports simultaneously. For this reason, computers that function as servers (i.e., FTP, Telnet, NFS) normally run a priority-based, preemptive multitasking system like OS/2, UNIX, or NT. Clients of TCP/IP can run almost any operating system, including DOS and Windows (although simultaneous communication may be limited on these platforms).

Host Names and Domain Name Service (DNS)

Computers like numbers, but humans like names. To make life easier for humans, TCP/IP requires that each computer have a text name in addition to its IP address. The text name for the IP address is known as the *host name*. For example, a computer might have the name `enterprise` or `slip50-12.ca.us.ibm.net`. To find the host name of your computer, type `hostname`.

Host names and their corresponding IP addresses make up an important part of TCP/IP. We need to know the correspondence between host names and IP addresses in order to find other computers on the network from our computer. Similarly, other computers need to have our name and IP address to find us. To resolve all of these names and numbers, TCP/IP provides two solutions.

First, if the network of TCP/IP computers is stable and small, you can create a file, known as the *hosts* file, that contains the names and their IP addresses. This file is then placed on every TCP/IP computer in the directory `\TCPIP\ETC` as the file `HOSTS`. (The location of the `\TCPIP\ETC` directory may vary by operating system and machine.)

Figure 6 contains a sample `HOSTS` file. The hosts file solution has a weakness—when you add a new host, you have to distribute the updated file to all of your other computers. To eliminate the need to constantly distribute revised hosts files, a *domain name server* (DNS) is used.

```
129.5.24.1    Host1     server1
129.5.24.3    PC1       student1   # This is a comment
129.5.24.4    PC2       student2
```

Figure 6. Sample Hosts File

```
domain eng.mit.edu
nameserver 129.34.128.245
nameserver 129.34.128.246
```

Figure 7. Sample Resolv File

Extension	Organization Type
com	commercial organizations
edu	educational organizations
gov	government organizations
mil	military organizations
net	network support organizations
org	anything other than one of the above

Figure 8. Common Domain Extensions

The DNS server houses the current tables that list host names and their corresponding IP addresses. Whenever a host wants to find another host by name, it asks the DNS server for its current IP address. This centralization of information allows you to keep things up to date in a single place.

The locations of the primary and backup DNS servers are kept in a special file known as the *resolv* file. The resolv file is in the directory `\TCPIP\ETC` as file `RESOLV`. Figure 7 gives an example of a resolv file.

In Figure 7, the first line is the name of the domain for this workstation. The next two lines provide the names and addresses of the primary and additional name servers.

To summarize the name resolution: If you are using a fixed set of servers and the definitions are static, you would use a hosts file. If you want dynamic assignments of names to IP addresses, you would use the resolv file to find the DNS server and have it look up the addresses for you.

You can have both a resolv and hosts file on the same computer. If you have both, the information in the resolv file is used first; if no name server can be found (name-server search timeout can be 1 to 2 minutes), the hosts file will be used. If you want to force your computer to use the hosts file only, rename or delete the resolv file.

Domain Name

In the sample resolv file in Figure 7, you may have noticed a new term called the *domain name.* An example of a domain name is `ibm.net`. This is the default domain for users of the Internet Access Kit (IAK) in OS/2 Warp. Another domain name is `netcom.com`, which is the name of the domain provided by the Internet supplier Netcom Corporation.

The general form of a domain name is the organization name, followed by a period and the type of organization. Figure 8 summarizes the common domain extensions and organization types in the USA domain.

For organizations outside of the USA, the last letters generally indicate the country where the domain is located. For example, `.fl` denotes Finland, `.uk` is the United Kingdom, and `.fr` specifies France.

You will also encounter subdomains within subdomains, such as our previous example address of `slip50-12.ca.us.ibm.net`. (I obtained this name by entering the `hostname` command after logging into Advantis using the OS/2 Warp Internet Access Kit.) In this address, `slip50-12` refers to the Los Angeles node (`slip50`), modem dial-up port 12. The rest of the address refers to the California (`ca.`) subdomain, within the United States (`us.`) subdomain, within the `ibm.net` domain. (In case you are curious, the name server returned an IP address of `129.37.50.12` for the SLIP connection to that port.) Figure 9 shows where our company is in the domain hierarchy.

Special IP Addresses

There are still a few more items to deal with before we are ready to rock and roll on TCP/IP!

Broadcast Address

All of the hosts on a particular network are always monitoring a special IP address known as the *broadcast address.* The broadcast default address has all host bits set to 1. For example, on a Class B address of `192.146.0.0`, the broadcast address would be `192.146.255.255` if no subnet masks are being used.

Destination Address

Most TCP/IP installations provide for communications to many different IP addresses. The *destination address* allows you to limit communications to a simple point-to-point link. If you are not using this special feature, you can leave this parameter blank.

Routing Information

The existence of networks and subnets causes traffic to be isolated to only your subnet. To get out of the subnet, as well as to get to other physical networks, you must provide *routing information.* Routing information consists of the addresses of other networks as well as gateway computers that act as intermediaries for moving information from one network to another.

Fortunately, most users only have to specify the default route to a single gateway computer that takes them to the outside world. The administrator of the gateway computer is responsible for making sure that the routing tables in the gateway route users to where they need to go. These same tables are also partially responsible for network security.

The *default route* is the location of the gateway computer to be used when no other routing information is provided to get to a specific IP destination. Additional types of routing paths can be provided to get to other networks, subnets, and even specific hosts. The *maximum number of hops* to get to each type of destination can also be specified. Again, the routing information will normally be supplied by your network administrator. If you are on a single network with no gateways or routers, you can leave this information blank.

Firewalls

When you connect computers together in a wide area network, how do you restrict traffic so that a hacker cannot damage your system? The answer is to

set up a *firewall.* The goal of a firewall is to give users access to the information they seek while at the same time preventing a direct and uncontrolled path to an unprotected network.

A firewall is neither a bridge nor a router. It is a computer that can perform work on multiple network segments. If you are on segment A (on one side of the firewall computer), and you want some data from segment B (on the other side of the firewall computer), you have to ask the firewall computer to fetch the data for you. At no time do you have access to segment B, either directly or indirectly. You must use the firewall computer to do the work for you as your agent.

An example of a firewall computer is an Internet news server. The news server normally has two network adapters. One adapter is on the Internet, and the other is on your company's secure network (see Figure 10). From your secure network, you can ask the news server for the latest posts from Internet newsgroups. The news server then goes out on the Internet to pick up and deliver that information to you.

Notice that the news server has two IP addresses. One is its address on the Internet (you have no idea what this address is), and the second is the address you use to access it from your secure network. Internet users are completely isolated from the addresses in your secure network.

Now, Get Those Numbers!

In this article, I wanted to give you an overview of TCP/IP terminology and practice. You should also now be ready to ask the right questions of your TCP/IP network administrator–IP addresses, subnet masks, broadcast addresses, hostnames, hosts and resolv files, and gateways.

Bibliography

The following books are excellent sources for additional information about TCP/IP:

- Craig Hunt, *TCP/IP Network Administration*, O'Reilly and Associates, 1992, ISBN 0-937175-82-X.
- Dan Harkey and Bob Orfali, *Client/Server Programming with OS/2 2.1*, Third Edition, Van Nostrand Reinhold, 1993, ISBN 0-442-01833-9.
- Mark Miller, *Troubleshooting TCP/IP: Analyzing the Protocols of the Internet*, M&T Books, 1993, ISBN 1-55851-268-3.
- Martin Arick, *The TCP/IP Companion: A Guide for the Common User*, QED Publishing, 1993, ISBN 0-89435-466-3.

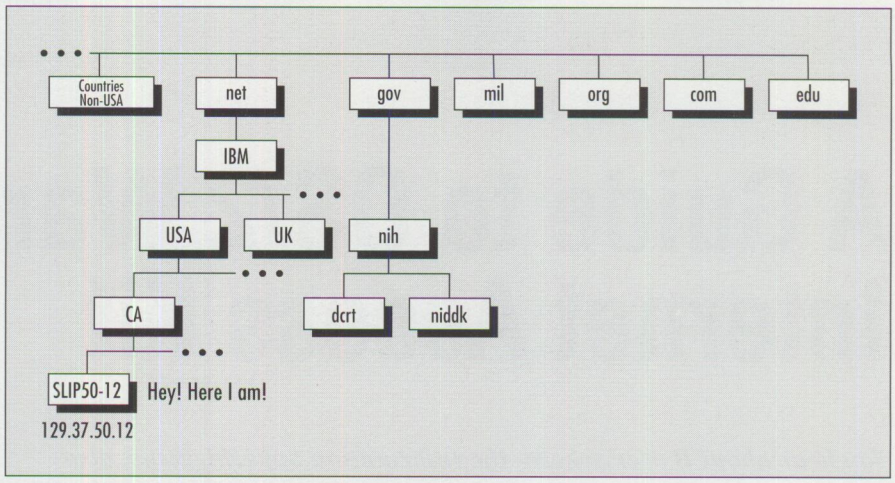


Figure 9. Domain Hierarchy Location of slip50-12

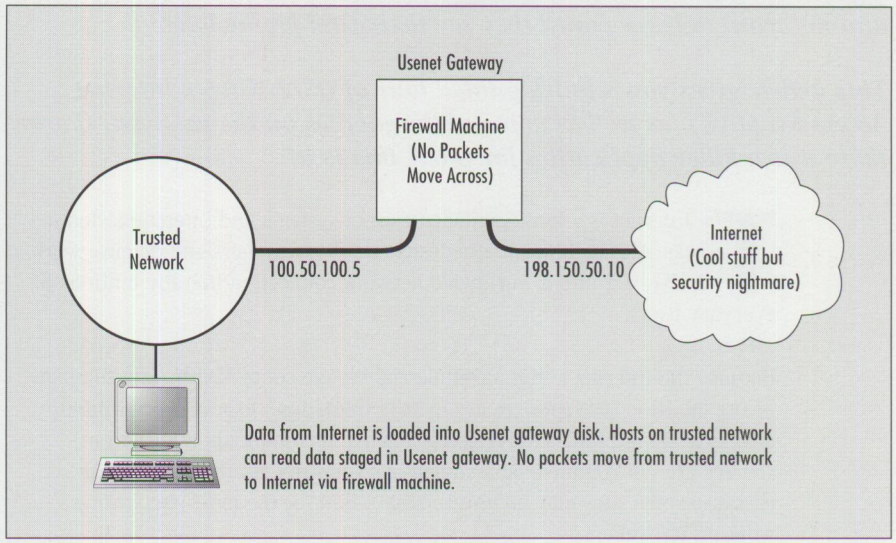


Figure 10. Firewall Computer for Internet Newsgroup Access

Philip Lieberman is a well-known author, lecturer, and consultant. Mr. Lieberman is also the owner of Lieberman and Associates, which produces the LAN Utilities for IBM LAN Server software product. His organization provides end-user support, corporate software development, and all types of training services for OS/2 and OS/2 products. Mr. Lieberman can be reached at:

Lieberman and Associates
221 N. Robertson Blvd. Suite C
Beverly Hills, CA 90211
Phone: (800) 829-6263 or
(310) 550-8575
Fax: (310) 550-1152
BBS: (310) 550-5980 CompuServe: 76426,363
IBMMAIL: USMVHLVH Internet: lanicu@ibm.net

A Guide to OS/2 Warp's Internet Access Kit

You hear about it everywhere–the Information Superhighway, also known as the Internet. With IBM's OS/2 Warp, you can get online on the Internet in just a few minutes. But be forewarned: the Internet is highly addictive–many people have been known to enter cyberspace and not want to leave (until they see their first phone bill).

This article gives you a quick guided tour of OS/2 Warp's Internet Access Kit (IAK), as well as some of the goodies on the Internet. If ever there was a killer application for OS/2, this is it!

Philip Lieberman
Lieberman and Associates
Beverly Hills, California

The Internet is a loose confederation of systems and interconnected networks. Although originally designed for researchers and defense work, it has evolved into a worldwide network connecting literally millions of everyday users.

Because the Internet is not a commercial system controlled by one company or organization (the services are usually free), don't expect everything to always work. Sites are always going down and coming up; information sources are not complete; and sometimes you can't get there from here (because some sites may be unreachable)! But, for the most part, things work pretty well.

To get onto the Internet, you must either use an Internet service provider, or your organization must connect its network to the Internet. Because there is little or no security for Internet traffic, most organizations give their users limited Internet access. This limitation is necessary so that outside hackers cannot access or damage sensitive company information.

TCP/IP, SLIP, and PPP

Connecting to the Internet is pretty simple. The Internet uses transmission control protocol/internet protocol (TCP/IP) for all of its communications. If your organization is already connected to the Internet and is using TCP/IP, you are ready to go. If not, all is not lost–you can still access the full Internet by using a modem to dial an external Internet service provider.

There are two different protocols for establishing a TCP/IP connection over a serial modem. The first, and most popular serial connection protocol, is serial line internet protocol (SLIP). The other serial connection protocol is point-to-point protocol (PPP). PPP is a more modern protocol, but its current usage is limited.

SLIP is the default protocol used by the Internet Access Kit supplied with OS/2 Warp. The initial version of the Warp Internet Access Kit did not contain PPP

support, but this protocol is now available as an update at no cost via file transfer protocol (FTP) or through the Software Update object provided with the IAK.

The Need for Speed

The dial-up connection to an Internet service provider is an amazing thing–in effect, you receive a temporary internet protocol (IP) address on the Internet that is as functional as any direct connection. But one downside to dial-up connections is their speed. Even using a 14.4 KBPS modem, you have to get used to long delays when downloading graphics or files. General response time to most requests is pretty good, however.

If speed is an issue but money is not, there are plenty of options. Digital ISDN (Integrated-Services Digital Network) terminal adapters, a step up from standard analog modems, provide 64 KBPS performance. Also, some service providers have leased-line telephone connections that offer 56 KBPS and T1 connections at much higher costs. These services provide Internet access to groups of users.

Internet Service Providers

When you first install the IAK, you can get up and running within minutes by using the built-in, one-button registration. This default registration sets up an account for you with the IBM Global Network (also known as Advantis in the US and other selected countries). There are many advantages to using IBM's own network, including:

- Access from 700 cities
- Access from 90+ countries
- Toll-free, 800-number access (at extra cost)
- 24-hour customer support

Although the IBM Global Network is the most convenient Internet service provider, it is not the least expensive. With the IAK, you can decide to use a local provider who may offer lower costs and/or higher speeds.

In the world of Internet, there are two basic types of accounts. *Shell* accounts give you terminal emulation on a UNIX machine. Many suppliers of shell accounts charge a fixed fee with unlimited connect time. The downside is the lack of a real TCP/IP connection, which means that shell accounts cannot use many Internet goodies, like Mosaic and graphics browsers, without a lot of hassles.

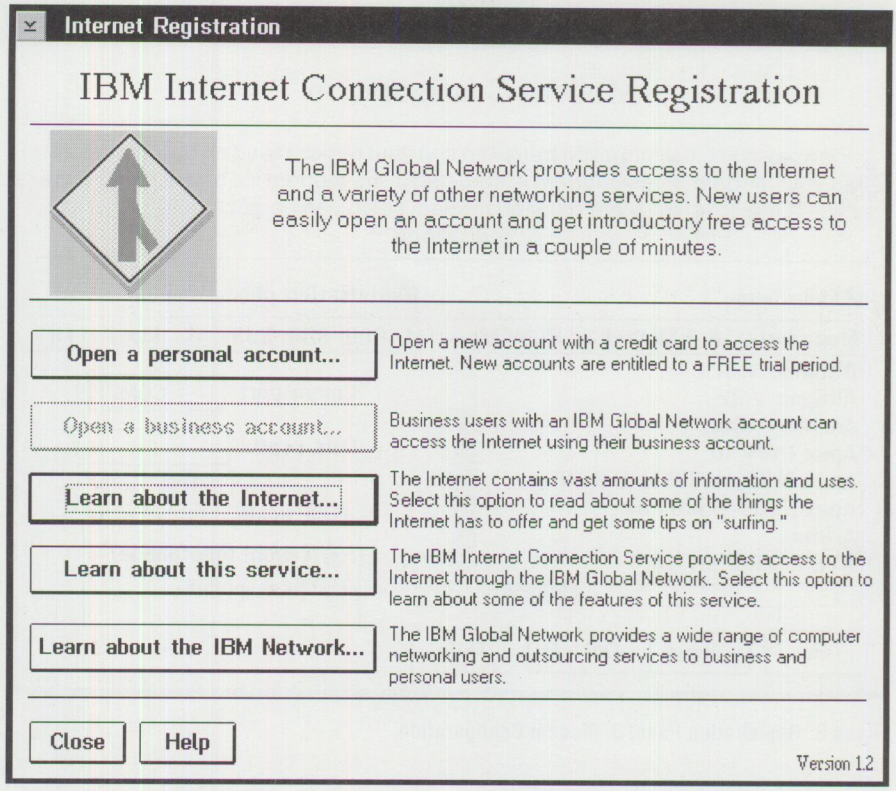


Figure 1. Start Registration

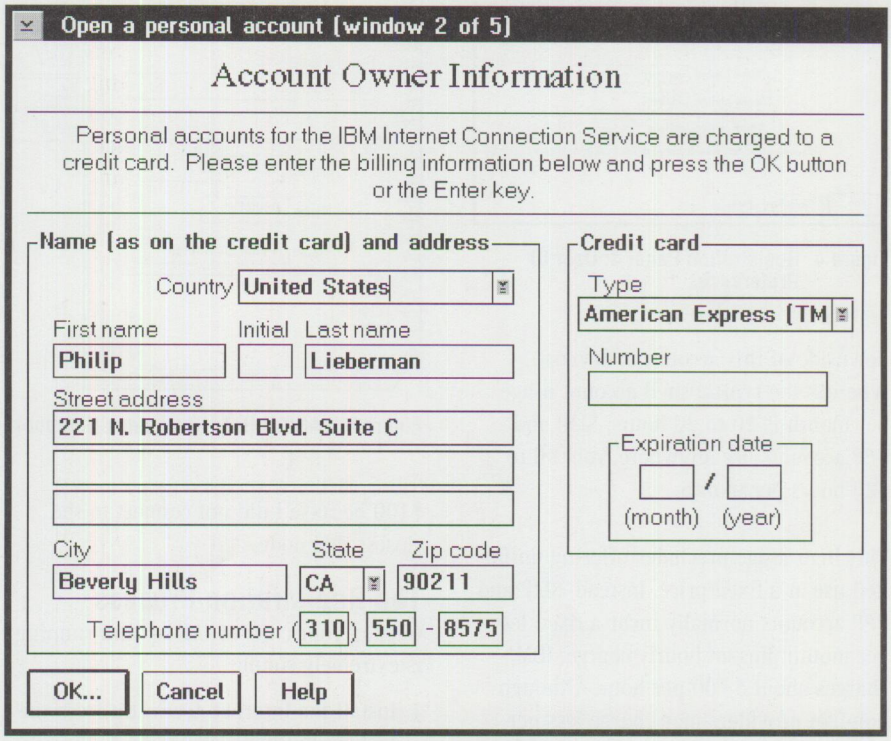


Figure 2. Registration Panel 2: Account Owner Information

The other types of account, *SLIP* and *PPP* accounts, give you full TCP/IP capabilities, where your computer is a full node on the Internet. This type can exploit the full capabilities of the Internet. The

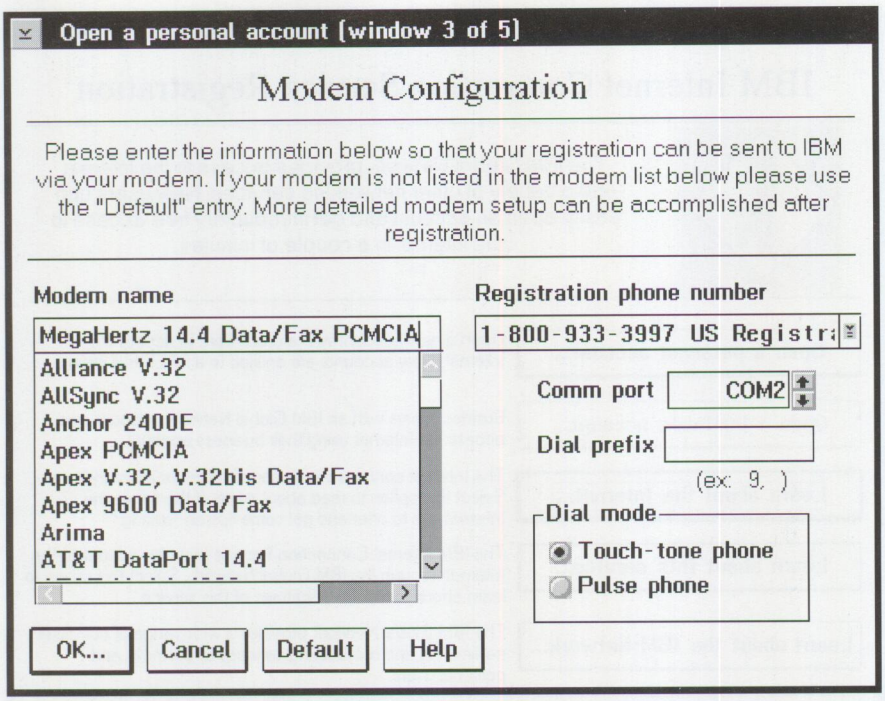


Figure 3. Registration Panel 3: Modem Configuration

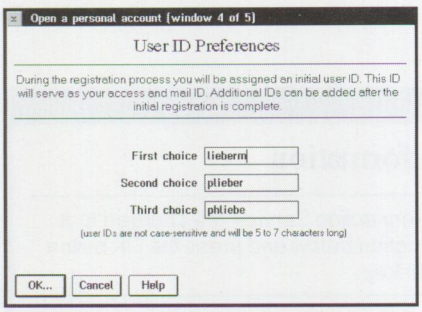


Figure 4. Registration Panel 4: User ID Preferences

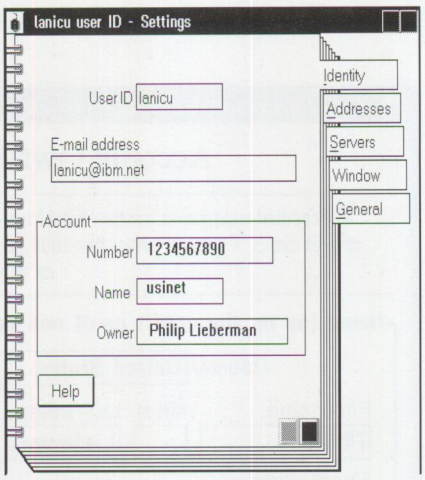


Figure 5. LANICU User ID Settings Notebook

downside of this account is the cost. Whereas the typical shell account usage per month is 10 to 20 hours, SLIP and PPP accounts use anywhere from 40 to 120 hours per month.

This high usage precludes offering unlimited use at a fixed price. Instead, SLIP and PPP accounts normally incur a fixed fee per month plus an hourly charge. IBM charges about $3.00 per hour. Although non-IBM providers may charge less per hour, they usually require account establishment fees, and they may offer only a limited number of nodes to connect to.

Don't forget that you also normally pay the telephone company for the call time. I discovered this last charge the hard way, when my monthly phone bill was over $100 because I did not connect to the closest IBM node.

The Registration Process

Getting your copy of IAK up and running is extremely simple:

1. Install the Internet Access Kit from the IBM Warp BonusPak.
2. Open the Internet Customer Services folder, and click on the Registration icon to start the Registration application.
3. You then see the Internet registration screen (Figure 1). The only registration available in the initial IAK is for a personal account. To start the registration, simply click on the "Open a personal account..." option.
4. You now see the first of five information panels. The first displays the legal agreement. (You will read it word-for-word because you know you should.) To proceed to the next panel, just click on the OK button.
5. The second panel (Figure 2) requests your account billing information and address. You must have a credit card to establish an account. The registration process even verifies that you give it a valid credit card number! After filling out this panel, click on the OK button to proceed to the next panel.

   You can add more users to your account later, after your registration is completed and your account is established. These users are added in a slightly different manner from the registration process given here. Your additional users are given unique identities, but their connect time is billed to your account.
6. To connect to the registration server, in the third panel (Figure 3) you must provide the type of modem, communication port, dial prefix (if needed), dial mode, and the proper phone number to use for your country. (Fortunately, this is a toll-free call in most countries.) To proceed to the next panel, click on the OK button.
7. You must have a unique identity when using the Internet. This user ID is needed for sending or receiving e-mail, authenticating yourself to other systems, and more. The user ID, normally five to seven characters in length, can be any name you wish, other than a name already taken by another user. To deal with this possibility, in the fourth panel (Figure 4) you are given three suggested choices that you may change. Once you are happy with your user ID selections, click on the OK button to proceed.

   Up to now, you have only filled the information panels. During the actual registration, the registration server tries to give you your first choice of user ID. If it is unavailable, it then tries your second choice, then third choice. The user ID eventually selected by the registration server is reported at the end of the actual registration process.

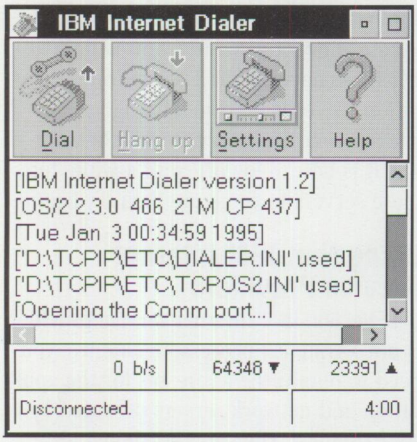


Figure 6. IBM Internet Dialer Application

The name that is finally selected becomes part of your complete Internet address. For example, my first choice was the name `lanicu`. This choice was accepted, and my full address on the Internet became `lanicu@ibm.net` when the registration was completed.

8. You are now ready to submit your Internet account request. In the fifth panel, select the "Send registration to IBM" button. Be patient–this step takes a few minutes.
9. When the registration finishes, you see a series of very important panels. The information from these panels must be transcribed to paper and kept in a secure place. The information includes your user ID and e-mail address plus the following confidential information:
   - Account number, used for billing
   - Account name, normally `usinet` for customers in the US
   - Account owner, the name used for billing
   - Password, which you should change to something you can remember

If you have problems with your account, you will need to know your user ID, account name, and possibly your account number. All of this information (except for the password) is contained in the Settings notebook of the user ID object on the desktop. Figure 5 shows my identity page after opening my Settings notebook.

Getting Online for the First Time

Your Internet connection starts with the Internet Dialer, found in the "IBM Internet

Updating Your Phone Number List

As part of your initial registration, IBM has already selected a local phone number for you. But watch out–it may not be your most economical phone number. To specify your local phone number, you will have to change the "Primary phone number" and possibly get a more up-to-date phone list, using the steps given below.

To update the phone list:

1. You must already be logged on to the Internet (using whatever phone number has been initially assigned to you) to get a new phone list.
2. Click on the Settings icon in the Internet Dialer window.
3. Click on the "Download phone list..." button.
4. In the dialog box, select the check box marked "Phone list."
5. Click on the Download button. You will get a notification when the update has been completed.
6. Close the notebook if you are done, or change the Primary or Backup phone number.

To change the phone number used for Internet access:

1. Click on the Settings icon in the Internet Dialer window.
2. Click on the "Primary phone number" drop-down list button.
3. Select a phone number.
4. Close the notebook.

Once you have the updated phone list, you will want to log off, select the nearest number from this new list, and log on again.

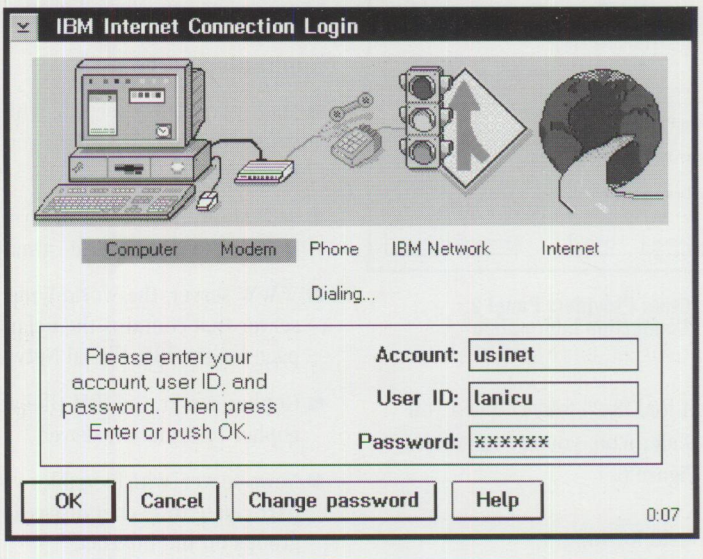


Figure 7. Connection Login Panel

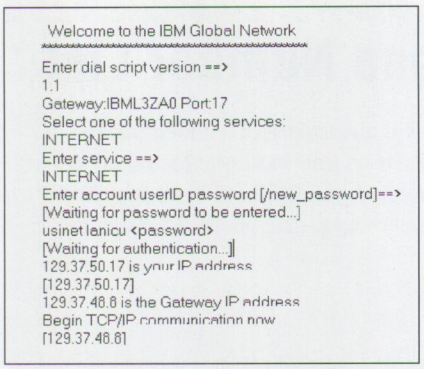


Figure 8. Login Dialog Listbox

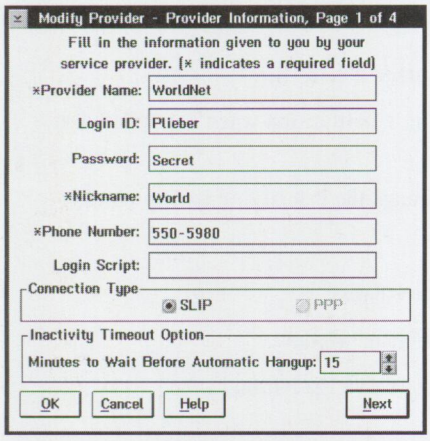


Figure 9. Other Providers Panel 1: Provider Information

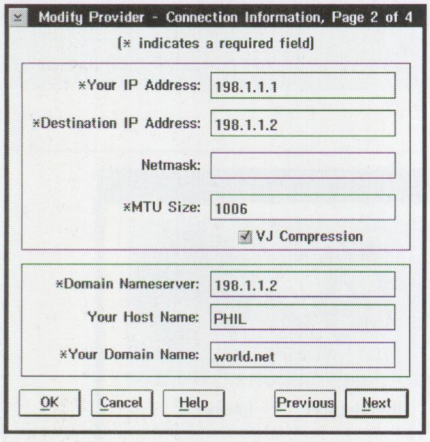


Figure 10. Other Providers Panel 2: Connection Information

Connection for OS/2" folder. When you open the Dialer icon, you see the dialer screen in Figure 6.

Here is how to log on.

1. Click on the Dial icon, found in the Internet Dialer window, to start the connection sequence.
2. You see the Connection Login panel (Figure 7). In this panel, most of the information should already be filled in for you. The only piece of information normally needed is the password. If you log into a friend's computer, modify the user ID and put in your personal password.

The Connection Login panel is fun to watch. You can see the entire login process graphically. To see the gruesome details of the login, you can look at the listbox for the dialer (which is usually obscured by the login panel). Figure 8 illustrates a typical login dialog.

The contents of Figure 8 may seem like gibberish, but there is important information here if you find that some of the services in the IAK are not working right. For example, you will have problems if any of the following servers used by the IAK are not functioning correctly:

Network Servers:

- Domain-name server (DNS), which converts names to IP addresses.
- FTP server, which is needed for code refreshes and software updates.
- Registration server, which handles initial user registration, adding and deleting user IDs, and changing passwords.
- Configuration server, which (at login) looks up the names and locations of the application servers.

There are backups for all of these servers.

Application Servers:

- Mail server (POP server), which holds your incoming mail when you are not logged on.
- SMTP server, which directs your outgoing mail to its proper destinations.
- WWW server, the World-Wide Web server that contains the default home page for the IBM Global Network.
- Gopher server, the IBM Global Network gopher information server.
- News server, which enables you to participate in Usenet discussion groups on the Internet.

By looking at the names and functions of the servers just listed, you can get a good idea of which features will not work properly if one of the network or application servers is not functional. For example, overall IAK function is limited when the domain-name server or configuration server is not functioning at login. If you see that these servers are not working, it is best to log off, wait a while for these servers to come up, then try again.

Using Non-IBM Internet Service Providers

Using a non-IBM Internet service provider is not a complex process. In essence, you need to obtain the addresses for the application and network servers provided by your supplier. The supplier should also give you a set of internet protocol (IP) addresses for both your account and the service provider. Of course, you are also given a user ID, password, e-mail address, provider name, and a telephone number to call for your connection. Setting this up is simple:

1. Open the Internet Utilities folder, and start the "Dial other Internet providers" application.
2. Next, you see a panel that lets you select the supplier you want. To do this, click on the "Add Provider" button.
3. You are then asked to fill in four panels with the information supplied by your Internet service provider. The first panel is shown in Figure 9.

   The only unusual things in this panel are the Nickname and Login Script fields. The nickname is just a name that you make up for the serial connection to this supplier. The login script is the location of either an ASCII file or a REXX procedure that your supplier has provided to handle the logon to his system.

   The ASCII file contains commands to be sent and responses that are expected. It is used when all of the logon information remains unchanged. You can find a copy of sample ASCII response files in the directory `\TCPIP\ETC` in the files `RSAS.RSP`, `SAMPLE.RSP`, `DEMON.RSP`, and `JVNCNET.RSP`. You must modify these files to use them. The name of the modified file to be used is placed in the Login Script field.

   If you need to deal with more dynamic login situations, a series of sample REXX scripts can be found in the subdirectory `\TCPIP\BIN` in the files `ANNEX.CMD`, `CYBERNET.CMD`, and `TDC.CMD`. As with the ASCII file, you

place the name of the modified REXX script plus its parameters into the Login Script field. For example, for the ANNEX.CMD script, you insert the entry

```
ANNEX.CMD atdtX-XXX-XXX-XXXX
<login id> <password>
```

into the Login Script field. Check each REXX script for its specific command-line requirements. When you complete this panel, click the Next button to go to the second panel.

4. The second panel (Figure 10) and subsequent panels request configuration information that only your Internet service provider can furnish.

   The second panel first requests the IP address of your computer and the Internet provider. Depending on whether your Internet supplier is using subnets, you may be required to enter a subnet Netmask.

   The MTU size defines the maximum size of a packet on your network. The default is 1006, but some Internet suppliers may use a different number (usually smaller for quicker error detection and correction) for their SLIP and PPP connections. VJ Compression controls whether or not your supplier is using Van Jacobson header compression (the normal setting is ON or checked).

   The domain-name server is needed to find IP addresses by a computer's name. The host name is the name of your computer for the domain-name server, and the domain name is the Internet name of your service supplier's network.

   Note: When using IBM's own Internet access network, the approach to this and the remaining panels is very simple: you don't have to worry about it. The IP addresses are picked up at login as part of the handshaking. The name of the domain-name server and its backup are already in the RESOLV file preloaded on your computer. The remaining addresses are provided by checking with the IBM configuration server. IBM did a very nice job of making the configuration management as simple as possible.

   Once you have completed this panel, click the Next button to proceed to the third panel.

5. The third panel (Figure 11) requests optional address information about your news, gopher, WWW, and mail servers. Again, all of this information is supplied by your Internet service provider.

   The only required pieces of information are the addresses of your mail servers. The rest of the addresses are needed only if you want to use the listed Internet services (more about these later).

   With this panel completed, click the Next button to move to the last panel.

6. The last panel (Figure 12) asks for information about your modem. This panel should be set for your type of modem, as well as its top baud rate for PC-to-modem communications.

   Some modems are capable of significant data compression. If your modem has data compression, you might want to move the speed all the way up to 38,400 BPS. But if your computer experiences problems when running at this speed (you must have a 16550 buffered serial chip to go above 9600 BPS), try reducing the speed.

   When everything has been filled in, click on the OK button.

Now you are ready to try your setup. To dial the just-added provider, select the provider from the list and click on the Dial icon. You see a status listbox, which shows you the progress of the login process.

If everything goes right, you should see a confirmation in the Current Connection status area. We had some problems with the user ID not being processed correctly. After deleting the user ID and entering it again, it worked. We also had to add carriage returns to some of the entries (use ^M). Once we got it all right, it worked great.

Using the Internet

This section looks at the most popular utilities on the Internet:

- FTP-PM—File Transfer Protocol-Presentation Manager, which gets files from all over the world
- Telnet, a terminal emulator over TCP/IP (VT100) for using remote hosts
- TN3270, an IBM 3270 terminal emulator over a TCP/IP connection

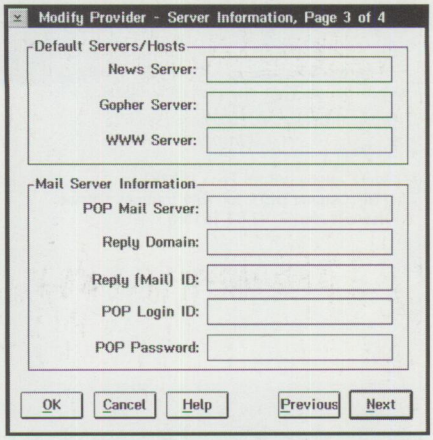


Figure 11. Other Providers Panel 3: Server Information

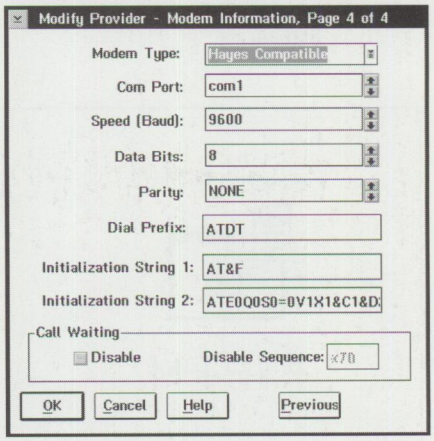


Figure 12. Other Providers Panel 4: Modem Information

- NewsReader/2, which allows you to read and write messages to over 7,000 forums
- WebExplorer, a facility (also known as Mosaic) for viewing Internet graphic hypertext pages
- Gopher, an information exploration tool
- Ultimedia Mail/2 Lite, an e-mail package
- Retrieve Software Updates, a program that enables you to get the most recent versions of IAK components

Let's start by discussing the last item, Retrieve Software Updates.

Retrieve Software Updates

If you are tired of poor product support from PC software vendors, get ready for a very pleasant surprise. IBM is constantly improving the IAK package, and you can receive free updates using the Retrieve

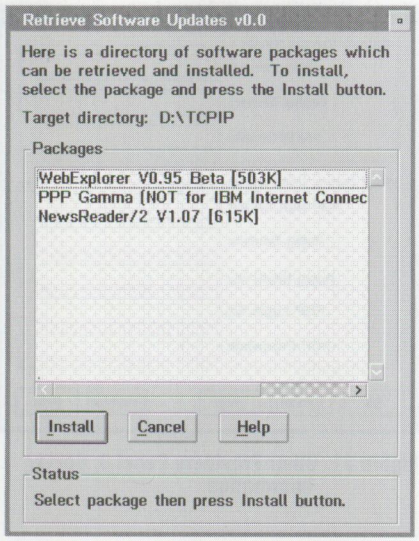


Figure 13. Retrieve Software Updates Screen

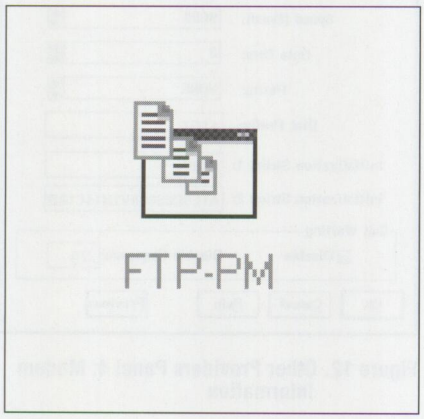


Figure 14. FTP-PM Icon

Software Updates program (`LINKUP.EXE`). This utility connects you to IBM's special update server and presents you with a list of new software that is available for immediate download.

The amazing thing about the update process is your ability to retrieve literally megabytes of data in the background while you are working with other systems on the Internet. This is accomplished by the SLIP or PPP connection and the multitasking within OS/2.

The process of obtaining updates is as simple as it can be:

1. Start the Retrieve Software Updates program from the "IBM Internet Connection for OS/2" folder.
2. Select an item to be updated (Figure 13).
3. Click on the Install button.

That's it! When the download finishes, you see a popup screen signaling completion. If you want to update another component, just select it and repeat the process. When all of the updates have been completed, you should disconnect from the Internet, then shut down and restart OS/2.

FTP-PM

If you are always on the lookout for the latest and greatest software, the Internet is a great place to find it. The most common way of sending and receiving files via the Internet is to use a special program and its protocol, both known as File Transfer Protocol (FTP).

The computers that provide file upload and download facilities are known as FTP sites. Getting into an FTP site to download files varies by the type of site from which you are requesting access. Many FTP sites are open to everybody, and they simply ask you to log on under the user name of `anonymous` with a password consisting of your e-mail ID (for example, my ID is `lanicu@ibm.net`). Sites that are open and only require a user ID of anonymous are aptly called anonymous FTP sites.

Once you log into an FTP site, you will immediately become overwhelmed with the variety of file names. To get a description of the files in a particular directory, you should first download the text file `OOINDEX.TXT` from that directory. This index file contains all of the file names and a short description of their function or contents.

There are a few "gotchas" in using FTP.

First, you must know the type of data you are moving. FTP operates in either binary or ASCII mode. If you are moving text files, you should use ASCII mode. On the other hand, if you want to move programs, you must use binary mode.

Another potential problem is the file names themselves. Most FTP sites use either a UNIX computer or an OS/2 computer running the High-Performance File System (HPFS). Next, it is common to have file names on FTP sites that don't conform to the File Allocation Table (FAT) 8.3 file naming rule. Consequently, during the transfer process you are given the option of specifying the name of the file on your end (known as the local host file name), as well as the name of the FTP site (known as the remote host file name). A word of warning: some FTP sites may refuse to take a file name with all capital letters for upload. Also, you must correctly type the case of filename(s) that you want to retrieve (i.e., `OOINDEX.TXT` is not the same file as `OOindex.txt`).

Finally, if you have worked with UNIX and OS/2, you know that path names in DOS and OS/2 use the backslash (\) to separate path entries, whereas UNIX uses the forward slash (/) to separate path entries.

Getting to Hobbes

For OS/2 users, there are many different sites you can access to retrieve software. By far, the most popular anonymous FTP site for OS/2 software is the Hobbes site at `hobbes.nmsu.edu`.

Two programs do FTP transfers in the IAK. The first is `FTP.EXE`. This program operates in either an OS/2 full-screen or OS/2 window session. To use `FTP.EXE` to get a file from the Hobbes site, do the following:

1. Open an OS/2 window session.
2. Type `FTP` <Enter> to start the FTP program.

   The remaining steps omit mention of the <Enter> key.
3. Type `OPEN hobbes.nmsu.edu` to connect to the Hobbes site.
4. Type `anonymous` (the user id).
5. For the password, type your e-mail address.
6. Type `cd /os2/32bit/network` to switch to the directory of Internet tools.
7. Type `binary` to enter binary transfer mode.
8. Type `hash` to provide the # mark for each 8 KB of data. This gives a visual cue to gauge the speed of the download.
9. Type `get` to start a file-transfer dialog.
10. Type `whois.zip` to specify the name of the file to download from the remote host.
11. Type `whois.zip` once again, to specify the name of the file to write on your computer.

12. Type `quit` to close the FTP program and connection after the transfer completes.

Note: "Whois" lets you find "who is" the administrator of an Internet site–a handy tool when you are having difficulties connecting to a remote system. Try typing `whois advantis.com`.

Creating a Graphical FTP-PM Icon

An easier way to accomplish all of this is to use the graphical version of `FTP.EXE`. Let's begin by creating an FTP-PM object (Figure 14) on your desktop, with all of the proper settings for connecting to the Hobbes FTP site.

1. In the IAK folder, open the Application Templates folder by clicking on its icon.
2. Use the right mouse button to drag a copy of the FTP-PM object into the IAK folder.
3. Fill in the Hostname field with `hobbes.nmsu.edu`.
4. Select the Authentication tab on the notebook and fill out the fields as follows:

   User: `anonymous`
   Password: (your e-mail ID)
   Account: (leave blank)
5. Select the Option tab on the notebook and change the Transfer Type to Binary.
6. Select the General tab and change the Title from FTP-PM to Hobbes FTP Site.
7. Close the notebook. You should now see the Hobbes Site icon on your desktop.

Using the Just-Created FTP-PM Icon

Aren't you glad you have to create an icon only once for each FTP site? Now comes the fun part.

To get into the Hobbes site, all you have to do is start the just-created icon. When the connection has been made (sometimes it won't complete because the Hobbes FTP site is so heavily used), you see the initial screen at the Hobbes site (Figure 15).

Let's try to download a file. A good file to download is the OS/2 Archie program (which you can use to find the location of files on the Internet).

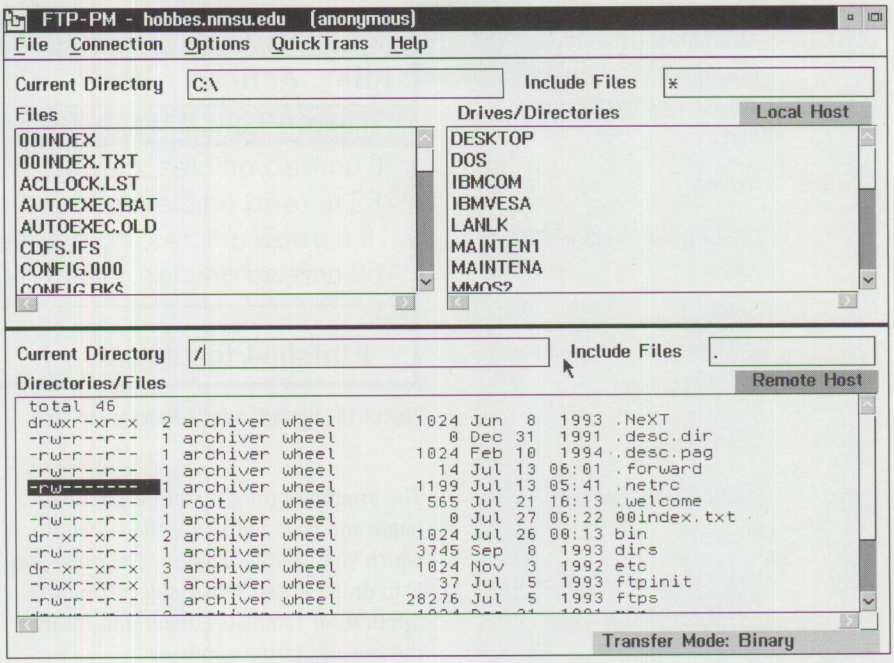


Figure 15. Initial Hobbes Site Screen

1. Change the current directory field on the remote host to `/os2/32bit/network`.
2. Scroll down the list of files on the remote host until you find the file `archie.zip`.
3. To copy the file, just put the pointer on the file name, hold down the right mouse button, and drag the file to the Files listbox. Release the button to start the transfer.
4. You get a confirmation message regarding the transfer. Just confirm the transfer.

That's all there is to doing a file transfer!

Note: Archie scans most of the anonymous FTP sites to find a file name you specify. Try typing `archie icudemo.zip` to find locations and versions of my LAN ICU package on the Internet.

You can also transfer files using the "Get remote files..." entry in the Files pull-down menu or use the Ctrl+G hot-key sequence.

Telnet

The standard terminal for most UNIX systems is the venerable VT100 manufactured by Digital Equipment Corporation (DEC). In early UNIX computers, lots of these terminals were hooked up to a single computer using serial lines. During the development of TCP/IP, a special terminal emulation protocol was developed. Known as Telnet, it allows the TCP/IP protocol to carry the VT100 terminal communications between a host computer and a terminal.

The Telnet emulator within IAK gives you this same emulation capability. The IAK Telnet emulator can emulate more than a simple VT100–it can also emulate ANSI, HFT, VT220, NVT, and a default (TCP/IP negotiates the best emulator).

You use the Telnet program to log in (as a dumb terminal) to remote computers. You cannot transfer files–that's what FTP is for–and the interface is text-only. But you can communicate with a wide variety of host systems this way.

Before I give you the details about accessing the following systems, you need to know that it is possible for some users to monitor your transactions on the Internet if they are connected to one of the Internet backbone segments. Consequently, you might want to use Telnet only for internal systems (not on Internet) or on Telnet systems that don't require a password that might be intercepted. Before you write off Telnet as an access method for the Internet, you should know that the possibility of someone actually intercepting your password is extremely low, but it is possible.

You can use the Telnet emulator two ways. First, you can start the Telnet program from the Internet Utilities folder.

alt	Alternative newsgroups (unconventional topics–anyone can start a topic here)
bionet	Biology
bit	Topics from Bitnet mailing lists
biz	Business, marketing, advertisements–commercial part of Internet
comp	Computers
ddn	Defense Data Network–military
gnu	Free Software Foundation–maker of GNU
ieee	Institute of Electrical and Electronics Engineers
k12	Education topics from grades K through 12
misc	Topics that don't fit in another category, but not weird stuff like alt
news	Discussions about Usenet newsgroups themselves
rec	Recreation, hobbies, and the arts
sci	Science
soc	Social topics
talk	Debating section

Figure 16. Popular Usenet Hierarchies

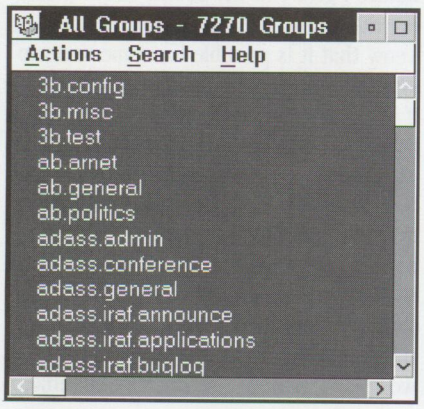


Figure 17. NewsReader/2 Group List Panel

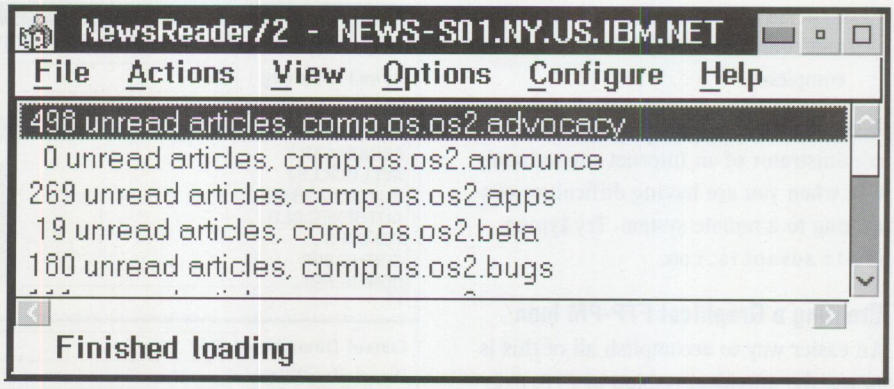


Figure 18. NewsReader/2 Newsgroup List

The program prompts you for the host name and port address of the system to which you want to connect. The other way is to drag out the Telnet object from the Application Templates folder and fill in the settings in the notebook.

Trying Out Telnet

1. Open up the Internet Utilities folder and start the Telnet program.
2. You are prompted for a host name. Enter `bbs.isca.uiowa.edu`. This is the biggest and most popular BBS on the Internet. If you cannot get through on this address, try `whip.icsa.uiowa.edu`. No port address is required.
3. When you get to the logon screen, enter as Login name `guest`.

Telnet to X-Windows: The Next Generation?

The next generation of terminal emulation is known as X-Windows. Although the IAK does not include the X-Windows emulator, you can purchase it from IBM as an extra-cost item. This terminal emulator allows you to create client/server graphical user interface (GUI) applications over TCP/IP. Because of the speed limitations of the Internet, as well as the limited speed of the modem connection, the X-Windows emulator's performance may be slow.

TN3270

Yes, you can get to SNA resources via the Internet. TN3270 is a special version of Telnet that knows how to talk EBCDIC instead of ASCII. The setup and operation of TN3270 are identical to Telnet's.

For fun, try using the TN3270 template to create a new TN3270 emulator. For the address of a TCP/IP-to-SNA host, try using `tcpgate.advantis.com` for your host name; no port address is needed. When you start this emulator, you see the "Welcome to IBMLink" screen.

If you are a member of the IBM Technical Coordinator Program, you will recognize this as the initial screen you see when logging on to IBMLink and OS2BBS1. If you have an IBMLink account, you can go ahead and log in.

NewsReader/2

This is the infinite time sink I warned you about at the beginning of this article. There are an almost infinite number of topic forums, called *newsgroups*, covering just about everything under the sun (and even things that probably should not see the light of day). For an excellent overview of Usenet (the name of the Internet network of newsgroup servers), read *The Internet Complete Reference* by Harley Hahn and Rick Stout, published in 1994 by Osborne McGraw-Hill, Berkeley, CA, ISBN 0-07-881980-6. Some of the popular hierarchies (sections with many subsections) in Usenet are listed in Figure 16.

There are also hierarchies based on geography and organizations. An example of such a newsgroup is `la.eats`, which deals with restaurants in the Los Angeles area.

The first time you start your NewsReader/2, there is no newsgroups file containing the complete list of newsgroups. As part of its initial startup, NewsReader/2 asks if you want the complete newsgroups list loaded from the newsgroup server. After answering Yes, you are presented with the list in Figure 17.

As you can see, you can read and send comments to 7,270 different newsgroups (the number as of late December) that are stored on the IBM Internet Usenet news server. Some Usenet news servers may carry more or fewer newsgroups.

Rather than scan through this entire list, you can either use the Find option in the Search pull-down menu or type Ctrl+F to select the topics in which you are interested. To find the newsgroups dealing with OS/2, specify the search parameter as OS2 (no slash). After you press Enter, the search begins and finds the first newsgroup with "OS2" in its name. If you want to add that newsgroup to the list of newsgroups that you read regularly, just double-click on it, or use the Add option on the Action pull-down menu.

Since there are many different OS/2 forums dealing with a wide variety of topics, you can search for the next "OS2" entry by using the accelerator key (Ctrl+N). To add the next newsgroup, repeat the selection/double-click operation. When you have completed adding the entries in which you are interested, you will see them listed in the main panel (Figure 18).

To select a newsgroup, simply double-click on the group in which you are interested. You will experience a delay while the NewsReader/2 program loads the current articles. When the loading has completed, you then see the Article List (Figure 19).

In the Article List, you see each message's number, topic, originator, and its number within that topic of discussion. To read a message, double-click on the entry. I want to read about the closing of a famous restaurant known as Chasen's, so I double-click on that entry. After a few moments, the article appears. If I want to reply to this article, I use the Reply option in the Actions pull-down menu.

Warning: The default behavior of the NewsReader/2 application is to copy in the original article (the one to which you are replying) and to place the cursor at the end for your comments. Although this is handy, NewsReader/2 does not send your reply unless it is longer than the text of the original article, which is not likely. Therefore, to ensure that your reply is sent, you have to delete most of the original article.

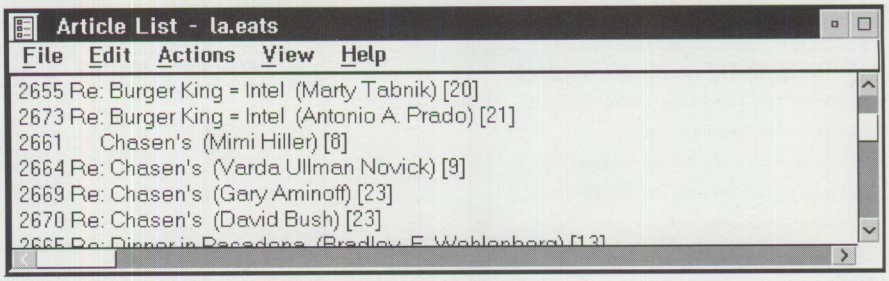


Figure 19. NewsReader/2 Article List

If you want to create a new thread of discussion, you can create it by using the Post option in the Actions pull-down menu, found in either the main window or the window of an open newsgroup.

New Groups

Almost every time you start up the NewsGroup/2 reader, there are new newsgroups. Figure 20 shows a popup window containing the list of the new newsgroups.

If you want to add one or more of these entries, just double-click on it to add it to your list. Then, every time you start NewsReader/2, you see the most recent articles that you have not yet read.

Because new newsgroups are added so frequently, every week NewsReader/2 asks you if you want to refresh the entire list of newsgroups. If you select Yes, be prepared to wait several minutes, because there are thousands of newsgroups!

WebExplorer

The most intriguing application in the Internet Access Kit is the WebExplorer. This program allows you to navigate through a series of servers known as web sites.

When you start the WebExplorer application, you are presented with the IBM home page, the first page that is downloaded after the program starts and connects to the IBM Web Server (Figure 21). From here, you can navigate within the pages of the IBM web site, or you can jump to other web sites and their pages. To switch to another site or to another page, move your mouse pointer over text or graphics on the currently visible page. If the text or graphics is a link to another

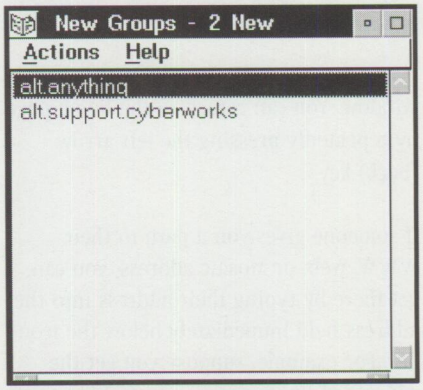


Figure 20. New Newsgroups

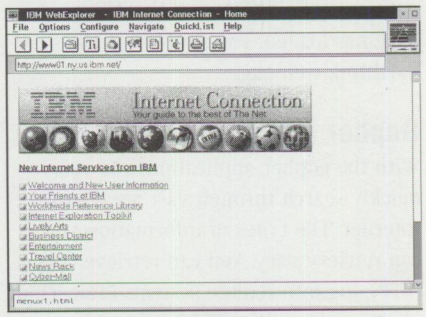


Figure 21. IBM Home Page

Figure 22. Web Path

place on the web, the pointer changes to a small page with an arrow above it.

For example, by selecting the Worldwide Reference Library from the home page and then selecting the Jet Propulsion Lab entry, you move to the JPL home page.

If you want to go back a level, click on the left arrow near the top of the

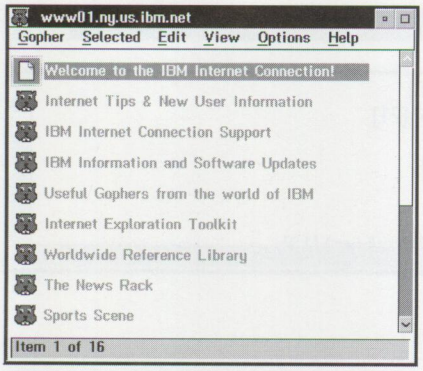


Figure 23. Gopher Opening Window

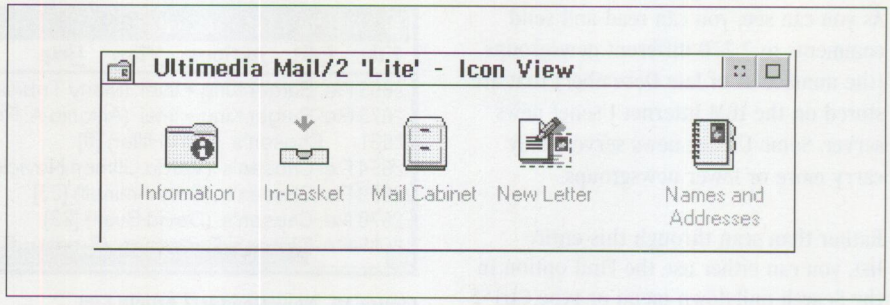


Figure 24. Ultimedia Mail/2 'Lite' Folder

window. You can return to the home page by repeatedly pressing the left arrow (back) key.

If someone gives you a path to their WWW, web, or Mosaic address, you can get there by typing their address into the address field immediately below the icon bar. For example, suppose you get the address of `http://www.earthlink.net`. To get to that page, enter the string `http://www.earthlink.net` into the path entry field (Figure 22). After the page has loaded, you should see the Earthlink web page.

Gopher

With the gopher application, you can quickly search through vast areas of the Internet. The types of information you can retrieve vary. You can retrieve files (FTP), log into remote systems (Telnet), or view graphics (like WebExplorer).

When you start the gopher application, you see a window like the one in Figure 23. By double-clicking on an entry, you can follow the trail of gopher servers.

The range of information available via gopher is mind numbing. I was able to log into San Francisco State University and find the name of a physics professor from whom I took classes many moons ago. I was even able to see the midterm grades for some of the classes in the current semester (ouch!). There are all sorts of gopher servers–the list is extremely long. The best advice is to just explore the world of gopher.

Ultimedia Mail/2 Lite

E-mail has been the central feature in the Internet since its inception. The e-mail package that comes with the IAK, Ultimail/2 Lite, is impressive, allowing you to send and receive not only text, but also binary data, as part of your e-mail messages. It has a very nice name-and-address book (with both user and groups), as well as a multilevel filing system of incoming and outgoing messages.

Sending E-Mail

Sending e-mail is very simple. After opening the Ultimedia Mail/2 'Lite' folder, you are presented with the window in Figure 24.

Next, double-click on the New Letter object. You will see a window for creating your message. You must fill out the e-mail address in the "To:" field, as well as the subject. You can customize the signature line at the end of the message using the "Settings..." option in the Letter pull-down menu. Once you are happy with your message, click on the Send icon.

After the message has been sent to the Internet mail server, you will receive a message acknowledgment popup window. (This may take a few minutes to appear.) In case the message could not be sent, it will be queued for transmission when you next log onto the Internet.

Receiving E-Mail

Receiving e-mail is as simple as double-clicking on the In-basket icon. This starts the e-mail receive program, which logs into the POP mail server run by IBM. Once the program validates your account and password with the mail server, your mail is retrieved.

An Amazing Package

The Internet Access Kit is quite a package. The software, though very powerful, is easy to use. All of the Internet components are there: FTP, Telnet, TN3270, WebExplorer, and e-mail. Because the IAK uses a SLIP or PPP connection with a preemptive, priority-based operating system–OS/2–you can literally run as many copies of these applications as you wish, at the same time. If you wonder where components like Archie or "whois?" are, the answer is just a quick FTP to Hobbes to pick them up–plus even more.

I have used the IAK daily since its release. It is extremely well-done software. If you have never thought seriously about OS/2, just load it so that you can use the IAK! As soon as you start to use the package, you'll probably say the same thing I said when I first used the IAK–"This is amazing!"

Philip Lieberman is a well-known author, lecturer, and consultant. Mr. Lieberman is also the owner of Lieberman and Associates, which produces the LAN Utilities for IBM LAN Server software product. His organization also provides end-user support, corporate software development, and all types of training services for OS/2 and OS/2 products. Mr. Lieberman can be reached at:

Lieberman and Associates
221 N. Robertson Blvd. Suite C
Beverly Hills CA 90211
Phone: (800) 829-6263 or
(310) 550-8575 Fax: (310) 550-1152
BBS: (310) 550-5980
CompuServe: 76426,363
IBMMAIL: USMVHLVH
Internet: `lanicu@ibm.net` or
`76426.363@compuserve.com`
(preferred)

CID Installation of OS/2 Warp and LAPS

This article gives a step-by-step procedure for building an OS/2 remote installation code server. The procedure includes both configuration/installation/distribution (CID) and redirected methods of installation of OS/2 Warp and LAN Adapter and Protocol Support (LAPS).

Using the procedure below, a network administrator or end user can build an OS/2 remote installation code server. This procedure allows you to incorporate previous, current, and future versions of OS/2, its ServicePaks, and the OS/2 platform applications (LAN Server, Communications Manager/2, Database Manager/2, etc.) into one structured directory tree on a single computer (assuming you have sufficient disk space).

Erik Mintz
IBM Corporation
Boca Raton, Florida

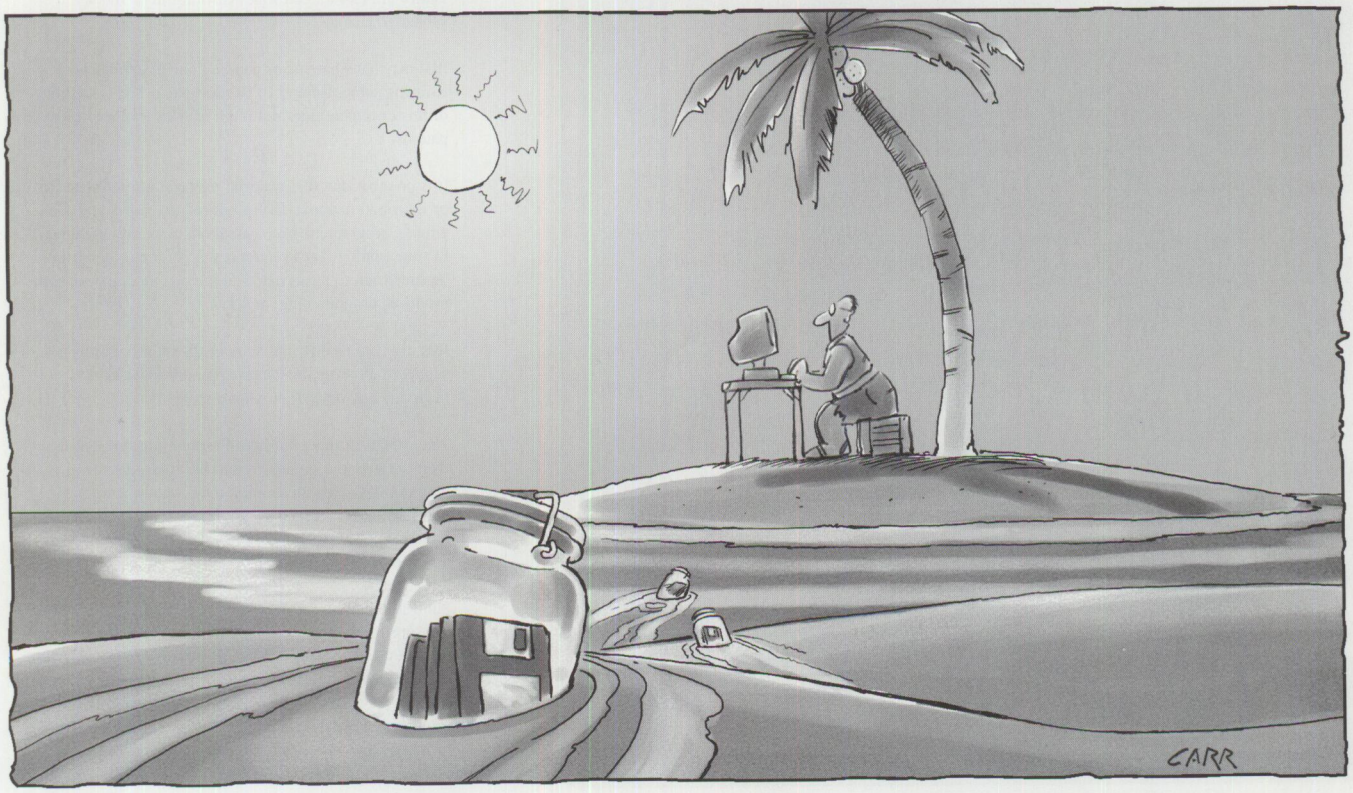
In the steps below, two types of remote installation are discussed (see Figure 1): CID and redirected. (Redirected installations are further divided into two types: response-file-driven and dialog-driven.) The procedure for building LAN Transport (LT) boot diskettes for both CID and redirected installations is described in the last step.

OS/2 2.11 and LAPS

Step 1. Install OS/2 2.11. Create a 100 MB primary partition on drive C for OS/2 2.11 and LAPS, and allocate the remainder of your drive as an extended logical partition (drive D) for your code server. I suggest using the File Allocation Table (FAT) format for drive C and the High-Performance File System (HPFS) format for drive D.

Note: OS/2 Warp and Windows together require about 50 MB of disk space when stored on the code server.

Step 2. Install LAPS on your primary partition. When you configure LAPS, choose the 16/4 Token-Ring device drivers and OS/2 NetBIOS.



XDF Support

Step 3. If your code server is not at Warp level, follow Step 3 to add the XDF Support to your server. (This information comes from Section 3.1.2.2, "Using an OS/2 2.x Code Server," in the `README.CID` file on the OS/2 Warp Installation disk [Disk_0]. Refer to `README.CID` for an explanation of the extended density format [XDF].)

To modify your OS/2 2.x system so that it can read the OS/2 Warp Version 3 XDF installation diskettes, do the following:

- For ISA computer systems:
  1. Rename `IBM1FLPY.ADD` to `IBM1FLPY.OLD` in the `\OS2` directory.
  2. Copy `XDFLOPPY.FLT` and `IBM1FLPY.ADD` from Diskette 1 to the `\OS2` directory.
  3. Add the following line to your `CONFIG.SYS` file:

     ```
     basedev=xdfloppy.flt
     ```

- For Micro Channel computer systems:
  1. Rename `IBM2FLPY.ADD` to `IBM2FLPY.OLD` in the `\OS2` directory.
  2. Copy `XDFLOPPY.FLT` and `IBM2FLPY.ADD` from Diskette 1 to the `\OS2` directory.
  3. Add the following line to your `CONFIG.SYS` file:

     ```
     basedev=xdfloppy.flt
     ```

Note: OS/2 1.x systems cannot be used to read the OS/2 Warp Version 3 XDF installation diskettes.

Step 4. Reboot your machine to load the XDF support.

Structured Directory Tree

Step 5. On drive D, create the directory structure shown in Figure 2. This directory structure is the heart of the remote installation server–it houses your OS/2 images, LCU modules, and any future version of OS/2 or its platform applications. The directory structure goes into the root directory of drive D.

NTS Utilities

Step 6. This step requires two service levels of the NTS/2 Utilities Disk: WR07000 and WR07045.

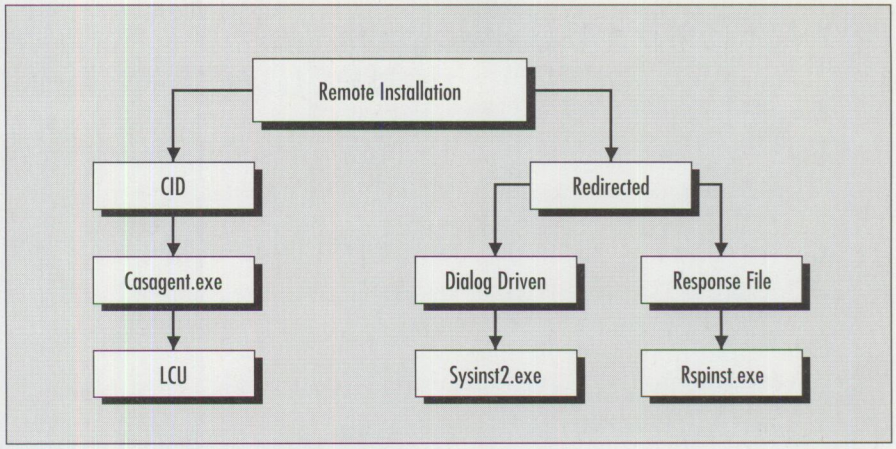


Figure 1. Two Types of Remote Installation

```
d:\cid
d:\cid\client
d:\cid\client\os2v30
d:\cid\compile
d:\cid\dll
d:\cid\dll\os2v30
d:\cid\exe
d:\cid\exe\os2v30
d:\cid\img
d:\cid\img\laps
d:\cid\img\lcu
d:\cid\img\os2v30
d:\cid\img\srvifs
d:\cid\log
d:\cid\log\lcu
d:\cid\log\os2v30
d:\cid\log\srvifs
d:\cid\rsp
d:\cid\rsp\laps
d:\cid\rsp\os2v30
d:\server
```

Figure 2. Directory Structure for Drive D on Code Server

```
copy a:\srvifs d:\cid\img\srvifs
copy a:\sample\service.*
 d:\server
copy d:\cid\img\srvifs\xi*.*
 d:\server
copy d:\cid\img\srvifs
 \service.exe d:\server
copy a:\lcu d:\cid\img\lcu
```

Figure 3. Installing NTS Utilities

Insert the NTS/2 Utilities diskette, service level WR07045, into drive A and type the commands in Figure 3.

Next, insert the NTS/2 Utilities diskette, service level WR07000, into drive A and type:

```
copy a:\applets\cas*.*
 d:\cid\compile
```

Code Server

Step 7. This step copies the LAPS disk to your code server.

1. Insert Disk 2 into drive A and type:

   ```
   unpack2 a:\bundle d:\cid\exe\os2v30 /n:setboot.exe
   ```

2. Insert Disk 3 into drive A and type:

   ```
   unpack2 a:\bundle d:\cid\exe\os2v30 /n:xcopy.exe
   ```

3. Insert Disk 4 into drive A and type:

   ```
   unpack2 a:\rexx d:\cid\dll\os2v30
   ```

4. Insert Disk 7 into drive A and type:

   ```
   unpack2 a:\cid d:\cid\exe\os2v30
   unpack2 a:\required d:\cid\exe\os2v30 /n:rspinst.exe
   unpack2 a:\required d:\cid\rsp\os2v30 /n:sample.rsp
   copy d:\cid\rsp\os2v30\sample.rsp d:\cid\rsp\os2v30\default.rsp
   ```

Figure 4. Unpack Instructions

```
# SRVIFS Configuration File
Adapter = 0
MaxClients = 5
MaxFiles = 102
Name = IMAGESRV
GroupName = No
ClientWorkers = 6
Path = d:\cid
PerClient = No
PermitWrite = No
alias=readonly,single,cid,d:\cid
alias=readonly,single,exe,d:\cid
  \exe
alias=readwrite,single,log,d:\cid
  \log
alias=readonly,single,rsp,d:\cid
  \rsp
```

Figure 5. Editing D:\SERVER\SERVICE.INI

Insert the NTS/2 LAPS diskette (WR07045) into drive A and type:

```
a:\lapsdisk a: d:\cid\img\laps
```

Step 8. This step unpacks some files necessary to set up your code server. Figure 4 contains the unpack instructions.

Response File

Step 9. In this step, you edit a response file with the minimum changes needed for a CID installation. Use an ASCII editor and change the keyword `exitonerror=0` to `exitonerror=1` in the file `d:\cid\rsp\os2v30\default.rsp`. Next, save the file. Then type:

```
copy d:\cid\rsp\os2v30
 \default.rsp
 d:\cid\rsp\os2v30
 \JOHNDOE.rsp
```

where `JOHNDOE.rsp` is an example response file for this CID installation.

OS/2 WARP Images

Step 10. This step installs your OS/2 Warp images on your code server. Type:

```
cd\cid\exe\os2v30

seimage /s:a:
 /t:d:\cid\img\os2v30
```

Here, `seimage` does an `XCOPY` of the diskettes to your server.

SRVIFS Configuration File

Step 11. This step shows how to build your SRVIFS configuration file. Descriptions of the parameters below can be found in chapter 6 of the *NTS/2 Redirected Installation and Configuration Guide* (S96F-8488). Refer to this manual to tune your SRVIFS configuration file for your environment.

In Step 6, in the statement `copy a:\sample\service.* d:\server` you copied `service.ini` to `d:\server`. Use an ASCII editor to edit `d:\server\service.ini` to reflect the statements in Figure 5.

THINSRV

Step 12. In this step, `thinsrv.exe` accomplishes three tasks:

- It creates or modifies your `STARTUP.CMD` file with statements necessary to start your code server.
- It appends path information in your `CONFIG.SYS` file.
- It adds `ifsdel.exe` to `d:\server`.

Type:

```
cd\cid\img\srvifs

thinsrv /r:d:\server\service.ini
 /s:d:\cid\img\srvifs
 /t:d:\server
 /tu:c:
```

Response File for CID Installation of LAPS

Step 13. The utility LAPSRSP can be used to create a response file for a CID installation of LAPS. Parameters for `LAPSRSP.EXE` can be found in two resources: *NTS/2 LAN Adapter and Protocol Support* (S96F-8489), section 4-2, and the IBM International Technical Support Organization's Redbook *Automated Installation for CID-Enabled Extended Services, LAN Services V3.0 and Network Transport Services/2* (GG24-3781), page 51. Follow the steps below to create a LAPS response file.

Required Hardware

The required hardware and software for the code server and its clients includes the following:

Server hardware/software requirements:

- 386 or greater processor
- 8 MB of RAM (16 MB suggested)
- 1.44 MB drive A
- Token-Ring network
- 16/4 Token-Ring Adapter (industry standard architecture [ISA] or Micro Channel)
- 160 MB of hard-disk space (minimum recommendation)
- OS/2 2.11 (recommended)
- NTS/2 LAN Adapter and Protocol Support (LAPS) service level WR07045 (to obtain LAPS service level WR07045, call [800] 992-4777). *Note:* The minimum level of LAPS must be WR07020.
- NTS/2 Utilities disk (two service levels, WR07000 and WR07045)

Client hardware/software requirements:

- 386 or greater processor
- 6 MB of RAM (minimum)
- 1.44 MB drive A
- 16/4 Token-Ring adapter (ISA or Micro Channel)
- 120 MB of hard-disk space (recommended)
- DOS and Windows or a FAT-formatted primary partition

Type:

```
cd\cid\img\laps

lapsrsp c:\ibmcom\protocol.ini
  d:\cid\rsp\laps\lapsrsp.rsp /t:c:
  /i:product
```

Here, you are creating a LAPS response file based on your code server's `PROTOCOL.INI` file. (In Step 2, when you installed LAPS, you created your `PROTOCOL.INI` file with NetBIOS support only.)

Step 14. Shut down and reboot. The message "IMAGESRV Server is active, Version 1.32" displays when your SRVIFS code server has successfully started.

LCU Command File

In steps 15 and 16, you are creating a customized LCU for each client. Along with `JOHNDOE.rsp`, steps 15 and 16 install OS/2 Warp and LAPS on your client's drive C.

Step 15. Now compile and edit your LCU command file.

Type:

```
cd\cid\compile

casprep casadv.fil
  d:\cid\client\os2v30
  \default.cmd

copy d:\cid\client\os2v30
  \default.cmd d:\cid\client
  \os2v30\JOHNDOE.cmd
```

JOHNDOE.cmd

Step 16. This step edits the file `d:\cid\client\os2v30\JOHNDOE.cmd`. Use an ASCII line editor and refer to Figures 6 and 7, which describe, by line number, the individual changes that must be made to `JOHNDOE.cmd`. Follow the instructions in double parentheses. Make the changes in Figure 6 first, then the changes in Figure 7.

LAN Transport Client Diskettes

Step 17. This step creates your LAN Transport client boot diskettes. See Figure 8 for the steps. Step 1 in Figure 8 is the only step that CID, dialog, and response-file driven methods have in common. To build a set of CID LT boot diskettes, follow steps 1 and 2. To build a set of dialog-driven diskettes, follow steps 1 and 3. To build a set of response-file-driven diskettes, follow steps 1 and 4.

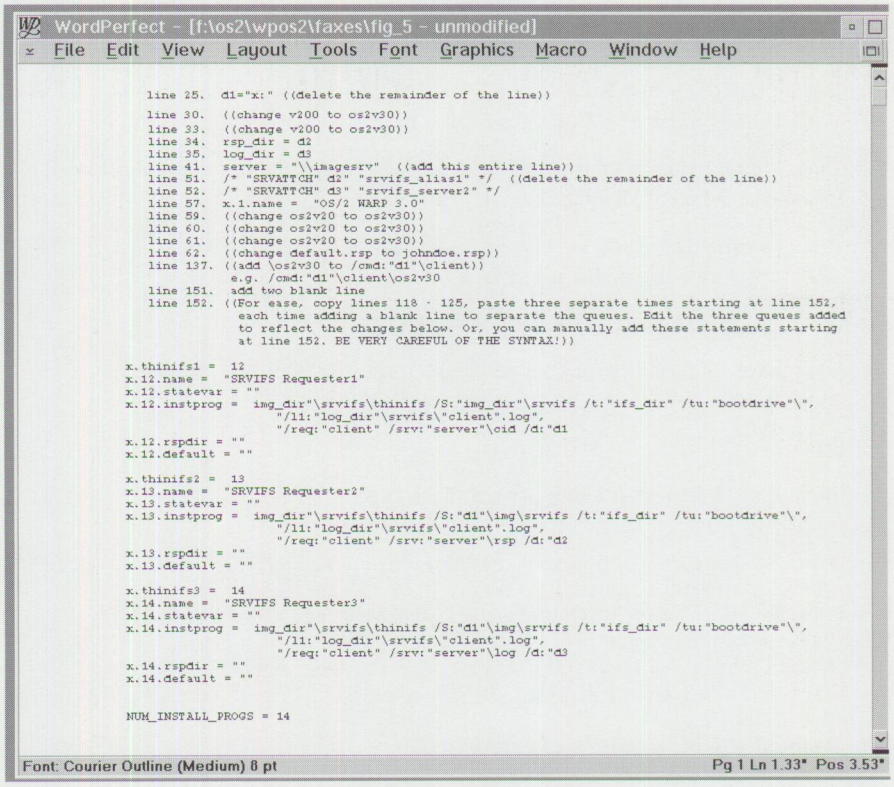


Figure 6. Editing LCU Command File JOHNDOE.cmd, Part 1 of 2

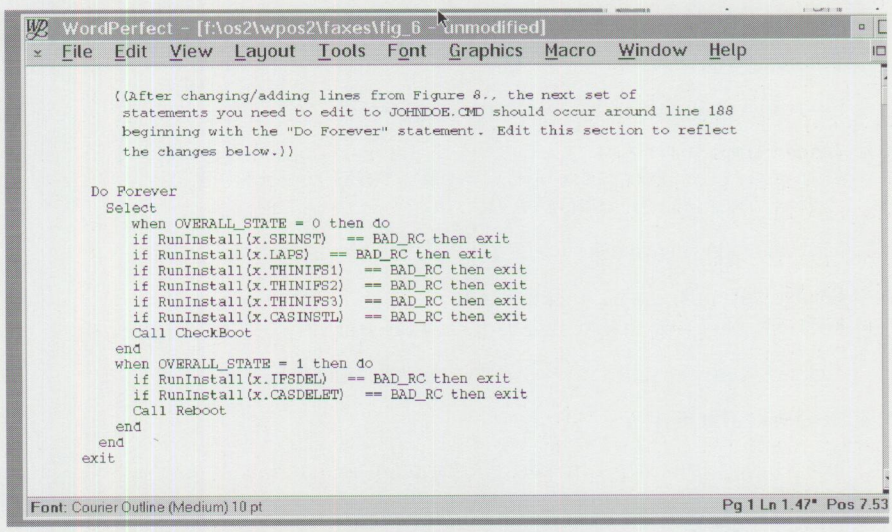


Figure 7. Editing LCU Command File JOHNDOE.cmd, Part 2 of 2

References

The following documents were used to create the procedures in this article:

- *NTS/2 Redirected Installation and Configuration Guide* (S96F-8488)
- *NTS/2 LAN Adapter and Protocol Support Configuration Guide* (S96F-8489)
- *OS/2 2.0 and 2.1 Remote Installation and Maintenance* (GG24-3780)
- *Automated Installation for CID-Enabled OS/2 V2.X* (GG24-3783)
- *Automated Installation for CID Enabled Products* (GG24-3977)
- `README.CID` on OS/2 Warp 3.0 Disk_0

Acknowledgment

The author thanks Steve Cunningham and Ahmad Ali from the CID team in the OS/2 Service and Support organization for helping to create this article.

1. Common–CID, dialog-driven, and response-file-driven

Type:
```
cd\cid\exe\os2v30
    sedisk /s:d:\cid\img\os2v30 /t:a:
```

`sedisk` prompts for twoformatted diskettes–the first is the installation disk and the second is Disk_1 of your LT boot diskettes.

Type:
```
cd\cid\img\laps
    thinlaps d:\cid\img\laps a: IBMTOK.NIF
```

Be sure to leave a space between `a:` and `IBMTOK.INF`.

If you have a Token-Ring Bus Master Adapter, use `IBMTRBM.NIF` in place of `IBMTOK.NIF`.

2. CID

Insert Disk_1 from step 1 into drive A. Then type:

```
cd\cid\img\srvifs
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:JOHNDOE /d:x:
 /srv:\\imagesrv\cid /w
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:JOHNDOE /d:y:
 /srv:\\imagesrv\rsp /w
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:JOHNDOE /d:z:
 /srv:\\imagesrv\log /w
cd\cid\img\lcu
casinstl /tu:a: /cmd:x:\client\os2v30
 /pa:x:\img\lcu /d /pl:x:\dll\os2v30;x:\img\lcu
 /l1:z:\os2v30\JOHNDOE.log
```

Add the following statement at the bottom of the `CONFIG.SYS` file on Disk_1:

```
set sourcepath=x:\img\os2v30
```

3. Dialog-driven

Insert Disk_1 from step 1 into drive A. Then type:

```
cd\cid\img\srvifs
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:*
 /d:x: /srv:\\imagesrv\cid /w
```

Make the following changes to the `CONFIG.SYS` file on Disk_1:

Change `set os2_shell = cmd.exe` to
```
set os2_shell=x:\img\os2v30\disk_1\sysinst2.exe
```

Add `set sourcepath=x: \img\os2v30` to the end of the file.

4. Response-file-driven

Insert Disk_1 from step 1 into drive A. Then type:

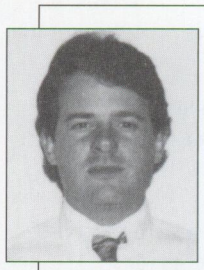
```
cd\cid\img\srvifs
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:*
 /d:x: /srv:\\imagesrv\cid /w
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:*
 /d:y: /srv:\\imagesrv\exe /w
thinifs /t:a: /s:d:\cid\img\srvifs /tu:a: /req:*
 /d:z: /srv:\\imagesrv\rsp /w
```

Make the following changes to the `CONFIG.SYS` file on Disk_1:

Change `set os2_shell = cmd.exe` to
```
set os2_shell=y:\os2v30\rspinst.exe
 z:\os2v30\JOHNDOE.rsp
```

Add `set sourcepath=x: \img\os2v30` to the end of the file.

Figure 8. Steps to Create LAN Transport Client Diskettes



Erik Mintz is an associate programmer in the OS/2 Service and Suppport Group in Boca Raton, Florida. Currently a CID analyst, he joined IBM in 1990 in environmental engineering and later worked in multimedia development. Erik has a BS in Hydrogeology from Florida Atlantic University and is completing his MS in Civil Engineering there.

Wrapping Up an OO Experience

With today's software technologies moving toward object-oriented (OO) techniques, companies must make changes to stay competitive and ensure their future market share. However, it is difficult to switch gears from procedural programming to OO programming if a company has to maintain current software products while making the transition. Companies are also reluctant to change if it means discarding existing software in which they have made a large investment and which has many years of profitable life left. IBM's goals, within the scope of the project about to be described, were to invest in future software technologies while maintaining the current software investment.

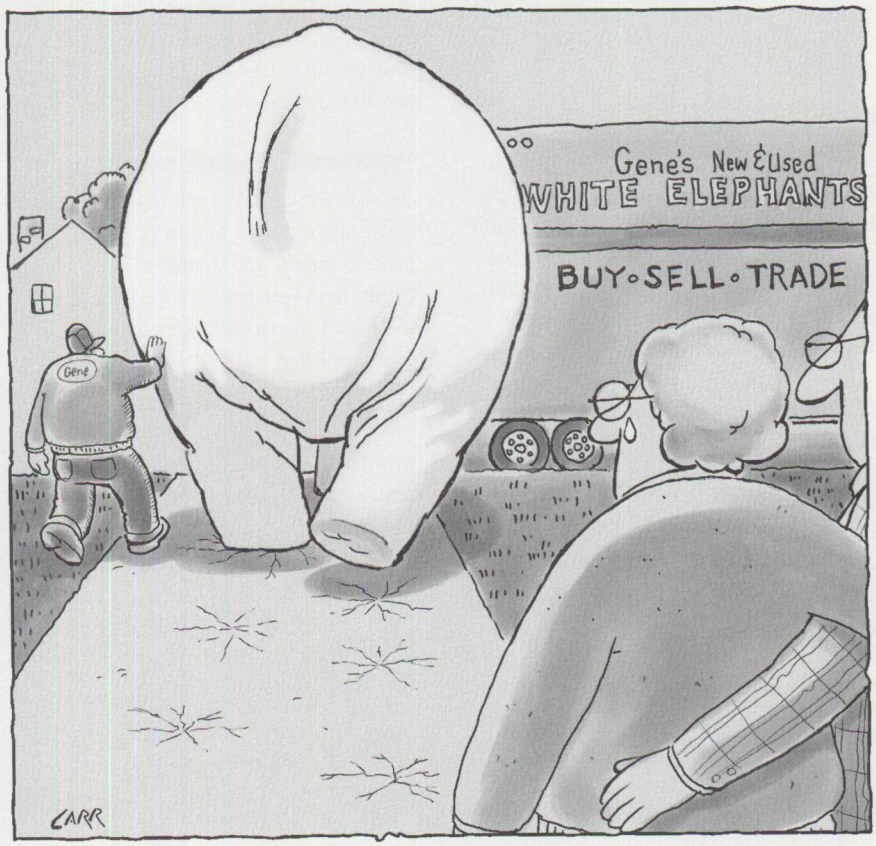
This article describes our experiences with object-oriented programming on a development project. We discuss the analysis, design, development, debugging, and testing phases of the project. Within these areas, we detail design decisions, implementations, and unforeseen pitfalls. **Note:** ***Readers should be familiar with OO terminology.***

Chris Dinallo, Dan Dorrance, Wendi Nusbickel, and Ann Robinson
IBM Corporation
Boca Raton, Florida

The goal of our development project was to incorporate certain existing but incomplete OO technology into a framework on our system. (A *framework* is a collection of classes that works together and provides a consistent programming model.) Our team consisted of four software developers with no previous experience in OO design or OO programming. However, we were experienced with a set of system services that used a procedural code base and contained functionality similar to the OO technology we inherited. It was this code base along with the OO technology that served as the point of reference for our project.

After analyzing what we wanted from the new OO technology, we realized that we could divide our project into two phases: developing user interface objects and developing subsystem objects. Of these two phases, the end-user requirements for the user interface objects were well defined and had a higher priority to get to market. Some of the more detailed design points of the underlying OO subsystem, however, were not complete. Here is how we structured the project:

- Phase 1: Implement only the OO technology's user interface objects as a layer on top of existing functions in the procedural subsystem (i.e., OO wrapping).



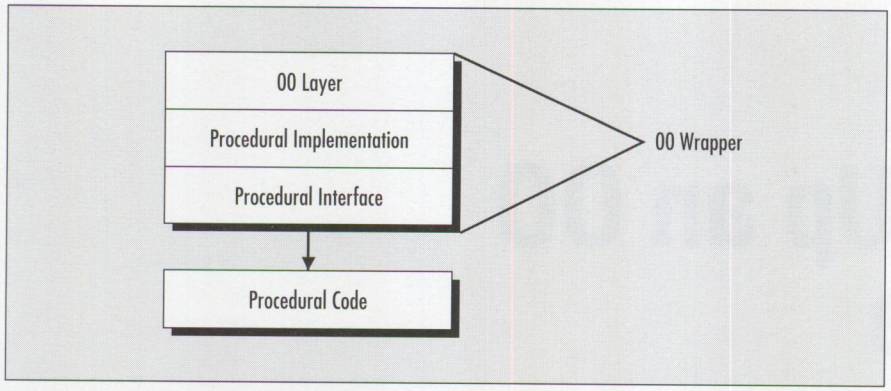


Figure 1. Incorrect, Procedural-Heavy OO Wrapper

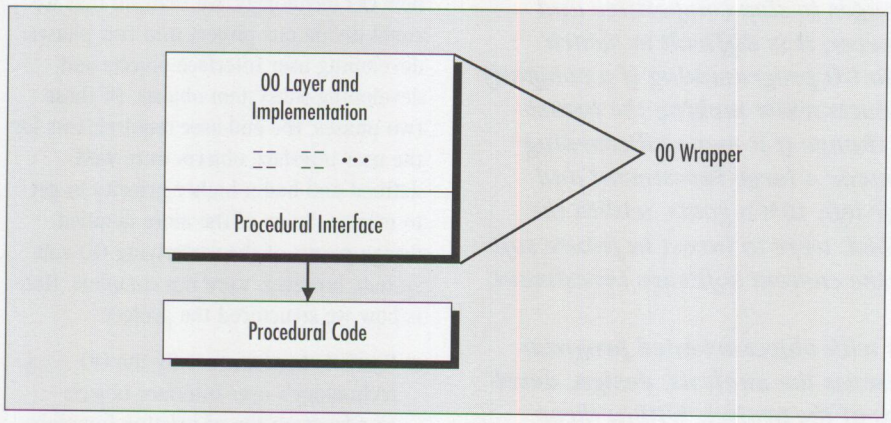


Figure 2. Correct, OO-Heavy OO Wrapper

- Phase 2: Implement the OO technology's user interface and subsystem components independent of the procedural subsystem.

Design Model

Our OO interface and subsystem had an underlying theme or design model, with different aspects of the model in the interface and in the subsystem. Together, the objects modeled a real-world counterpart such as an inventory control system or a weapons guidance system.

The user interface was at a higher (simpler) level of the model. The subsystem used the same model but at a lower (more complex) level. So the two—the user interface and the subsystem—were linked by a common overall model, although each supported a different perspective of that model.

Using an automobile as an analogy, if the design model of our subsystem were a car, then the user interface objects are things that the driver sees and interacts with (a speedometer, fuel-level indicator, seat, steering wheel, etc.), while the subsystem objects are things that a mechanic interacts with to support the user interface objects (like the speedometer cable, fuel float, steering linkage, sensors, and so on).

Iterative Design/Development

We discovered first-hand that we needed an iterative design and development process for the OO wrapper. This is something that is easy to overlook, because the job of the OO wrapper is typically oversimplified.

One of the more common pitfalls of OO wrapping is to make it more procedural-centric than OO-centric. This happened to us, partly because this was our first OO project, and interacting with the procedural subsystem put us in the procedural mindset more than the OO mindset.

Figure 1 is a diagram of the procedural-heavy OO wrapper (the *wrong* way to OO wrap), while Figure 2 is a diagram of the OO-heavy OO wrapper (the *right* way to OO wrap).

Further trial and error uncovered how easy it is to underestimate the complexity of the OO wrapper. As we experienced behavioral differences between the procedural subsystem and the OO design model, which required emulation and workarounds, the amount of code necessary for the OO wrapper methods began to increase. Moreover, we had not really applied an OO design to the OO wrapper itself, so the increasing size of the methods' code took on a procedural flavor.

Error conditions or special cases cause the OO wrapper code to become complex and large. Often, applying OO design techniques to the structure of the bridge code/wrapper code will simplify how to handle special conditions, errors, or bug workarounds. Sometimes you have to do workarounds inside of bridge code/wrapper code to handle bugs in the procedural code base, which won't be fixed because of priorities or resources. In an attempt to further separate the procedural code from the OO code, we designed supporting classes to perform tasks within the OO wrapper itself (such as threading, notification, and windowing). We also encapsulated the calls to the procedural subsystem functions.

Ideally, it would have been nice to have a complete OO design for the OO wrapper class up front. However, this was not realistic because of all the workarounds required for procedural subsystem bugs and behavior differences. Therefore, the process of designing and implementing the OO wrapper needs to be iterative. First, perform an initial design and implementation of the OO wrapper. Then, as you discover the function that must be added into the OO wrapper, re-examine the design of the OO wrapper class itself to ensure that the design goals are being met and that the wrapper is not becoming too procedural.

OO Wrapping

It's important to have a model of a real-world entity on which to base the OO design and interfaces. Otherwise, the procedural code will overly influence the OO design and will surface in the OO implementation. Because the procedural behavior is not in synch with the overall OO design there will be problems with encapsulation, inheritance, and polymorphism. Therefore, we did not start our OO wrapper implementation until we had a clear

design model for the project, including the user interface and its behavior in the context of that design model.

If we did not have a complete user interface design for the OO wrapper, we would have been designing this interface during our implementation phase. That would have required us to continually verify that the interface we were creating would fit with the design model of the project (including the underlying OO services).

In practice, it was very tempting to let the behavior of the underlying procedural "subsystem" be reflected in the behavior of the wrapper itself. For example, both the procedural and OO subsystems supported the aspects of a `DoIt()` function and `WaitForFinish()` function, which waits for `DoIt()` to finish. At first glance, this seemed to be a 1:1 mapping that should be pretty straightforward to wrap. However, as we got into the implementation, we discovered otherwise.

The OO user interface method for a `DoIt()` function, called `OODoIt()`, was asynchronous by default. If further processing of the application depended on the results of `OODoIt()`, the application then called the OO user interface method for an `OOWaitForFinish()` function. Figure 3 shows how an application uses the OO methods `OODoIt()` and `OOWaitForFinish()`.

Figure 4 shows how an application uses a procedural user interface for the `DoIt()` function, called `ProcDoIt()`, and waits for the procedural function to finish.

The procedural user interface for the `DoIt()` function did provide an asynchronous capability. However, it had no wait-for-finish concept, which the OO user interface defined. Instead, it used a notification message loop model, which resided in another thread, to receive a message upon completion of the `ProcDoIt()` function.

To provide the `OODoIt()` with a `OOWaitForFinish()` method, the OO wrapper had to do all of the following:

1. Implement an `OODoIt()` method that calls the procedural subsystem's `ProcDoIt()` function asynchronously.
2. Implement an `OOWaitForFinish()` method that waits for the `OODoIt()` function to complete. (This "wait" is accomplished using a "done" semaphore. This method blocks the main application thread while it is waiting for `OODoIt()` to finish.)
3. Create a secondary thread to receive notification (in a message loop) that the operation of the `ProcDoIt()` function has completed. Then wake up anyone waiting for `OODoIt()` to finish. (In this case, our "waiter" is the `OOWaitForFinish()` method.)

```
-Application thread-  Object Code
OOApplication() { ...
    // The object is constructed ...
    object.OODoIt();            // Call OODoIt() async
    // Do other stuff...
    object.OOWaitForFinish();   // Wait for OODoIt() to finish
}
```

Figure 3. Application Pseudo Code Depicting OO Calls

```
-Application thread- Procedural Code
ProcApplication()
{
    ...
    // Note: ApplicationMessageReceiver() already
    // running in a secondary thread.
    ProcDoIt( asynchronous );      // Call ProcDoIt() async
}
-Secondary thread in application-
ApplicationMessageReceiver()
{
    ...
    //Message processing loop in application
    while ( wait forever for a message )
    {
        ...
        // Has the operation of the ProcDoIt() function finished?
        ...
    }
}
```

Figure 4. Application Pseudo Code Depicting Procedural Calls

Thus, the behavior of the OO interface was emulated in the OO wrapper by using a combination of the base procedural subsystem's supporting function, plus new code in the OO wrapper itself, to "bridge" the behaviors of the OO design model and the procedural subsystem.

Figure 5 illustrates the OO Wrapper code, which emulates the OO interface model for the methods `OODoIt()` and `OOWaitForFinish()`.

Although both the procedural and OO implementations supported the `DoIt()` function, the behaviors of these functions in combination with the `WaitForFinish()` method were sufficiently different that additional emulation code was required. Had we not had a clear design for our OO interface, we would have been tempted to force the notification message model upon the OO interface. This was counter to the user interface design model of the project.

Applying OO Techniques

Here is a common problem we encountered: Methods that seemed to be general-purpose ended up instead being specialized, because they used derived classes when they should have used base classes.

Virtual functions defined in the base and overridden by the derived classes allow

```
—Application thread in OO wrapper—
OOWrapper::OODoIt()
{
  // Note: ProceduralMessageReceiver() already
  // running in a secondary thread.
  ProcDoIt( asynchronous );      // Call ProcDoIt() async
}
OOWrapper::OOWaitForFinish()
{
  // Sets & waits for a "done" semaphore to be woken
  // up by OOWrapper::ProceduralMessageReceiver.
  // Do not return to application until done.
}
—Secondary thread inside of OOWrapper—
OOWrapper::ProceduralMessageReceiver()
{
  //Message processing loop
  while( wait forever for a message )
  {
     ...
     // Did the operation of the ProcDoIt() function finish ?
     // Yes, wake up the OOWrapper::OOWaitForFinish()
     // method (via the "done" semaphore).
  }
}
```

Figure 5. Pseudo Code Depicting OO Wrapper Emulation of the OO Interface Model

```
Car::Drive( Corvette& c)
{
   ...
   c.GetMaxMPH();
   ...
}
```

Figure 6. Using Corvette in the Car::Drive Method

the caller of the class to execute the correct derived class code, while keeping the subsystem's method general-purpose. A method that uses the derived classes explicitly, when the base class methods could be used instead, becomes specialized and inflexible.

As an example, assume we have a base class `Car` that is the basic access to anything you want to do to a car, such as drive it. The `Car` class has a method called `Drive()`. For car models, we have a base class called `Model`. The `Model` class has a virtual method `GetMaxMPH()`, which its subclassers override. We also have a subclass of the base class `Model` called `Corvette`.

We want to use the `Car` class's `Drive()` method to drive a Corvette.

Right now, we have only one car `Model`, a `Corvette`. So we make the `Car` class's `Drive()` method use the `Corvette` subclass to get information that helps us `Drive()` it.

Figure 6 illustrates the use of `Corvette` in the `Car's Drive()` method.

Later, we add a new car model, `Ferrari`. We want to drive the Ferrari, but the `Car::Drive()` method has been specialized for the `Corvette` subclass.

We could overload the `Car::Drive()` method to accept a reference to a Ferrari instead of a Corvette. That may make sense if the operation of `Car::Drive()` has to be specialized for each model of car. However, we would need a different `Drive()` method for each `Model` of car!

A better solution would be to reference the `Model` class in the interface to the `Car::Drive()` method instead of the `Corvette` class, since `Model` is the base class for both the `Corvette` and `Ferrari` classes. This means that `Car::Drive(Corvette& c)` would be defined as `Car::Drive(Model& m)` instead.

Then we need only one `Drive` method for every `Model` of car. This is code re-use. At runtime, the Model class's virtual method `GetMaxMPH()` would get resolved to the `GetMaxMPH()` of the subclasser (either `Corvette` or `Ferrari`).

Figure 7 illustrates the code for using `Model` in the `Car` class's `Drive()` method.

For our methods, using base classes with virtual functions (used by derived classes) wherever possible gave us the best general-purpose solution.

Portability

Although the portion of code in the OO wrapper that interfaced to the procedural subsystem was not portable, the OO interfaces supported by the wrapper were portable. This is because we did not tie any operating system, C++ compiler, or hardware-specific details into the OO methods themselves.

For example, the method in Figure 8 is not portable; the parameters for the method in Figure 9 are not portable; and the method in Figure 10 is portable.

As in the case of our OO wrappers, the objects within the OO subsystem should not refer to anything that is platform-specific (hardware, operating system, C++ compiler). We strove for the highest possible level of portability in the OO subsystem, which in our case was source-code compatibility. The intent was that a development team could rebuild and run this subsystem on their platform and plug in any platform-specific objects they have.

A few of the general guidelines we tried to follow for the OO subsystem were:

- Try to keep code as portable as possible. This can be very simple, such as using a standard C++ library function versus an operating system (OS)-specific function to accomplish the same thing. For example, to open a file, the C++ runtime function open is more portable than an OS-specific system call such as OS/2 `DosOpen`.

- Write code for the general (not platform-specific) case. Some portions of our OO subsystem needed to access OS services and features that were available only on certain OS platforms. To keep this as portable as possible, we had a base class for the general sense, then subclassed it for specific OS capabilities. The subsystem itself used the base class's virtual methods, so that it did not refer to a derived class that may not exist on another OS platform. This allowed us to add the level of detail we needed for the platform, without creating an unportable OO subsystem. Figure 11 illustrates this point.
- Watch for platform service dependencies. One of the more subtle problems we saw was that an object supplied by the software vendor depends on a system service and its behavior, but sometimes that system service does not exist on other platforms. Although a system service may be wrapped in a nice class hierarchy that does not expose the system interfaces directly, it can have inherent behaviors that cannot be emulated on other platforms. Code using the system service that relies on the behavior of the service classes is not portable, even though it may appear to follow all the basic portability rules. To prevent this problem in our OO subsystem, we analyzed the underlying system services and provided code to emulate a standard behavior where necessary.
- Design configurability into the subsystem. In order to "plug in" derived classes for OS-dependent function, we needed a flexible subsystem design. It had to be able to dynamically retrieve the subclass of interest, based on a user selection—as opposed to always using a specific derived class that assumed a certain OS platform.

Different Addressing Models

The OO subsystem was intended to be used on multiple hardware platforms as well as multiple or different operating systems. We encountered an Endian issue with the OO subsystem on our platform. (The term *Endian* describes an addressing model that affects the byte ordering of data and instructions in memory.)

If code only assumed a specific Endian, or addressing model, it would be unportable to other platforms that support different Endian types. To make code Endian-portable, you should provide a runtime capability to determine the Endian-ness of the current hardware platform.

```
Car::Drive( Model& m)
{
   m.GetMaxMPH();  // Virtual, resolved at runtime
                   // to Ferrari or Corvette, or
                   // whatever the model...
}
```

Figure 7. Using Model in the Car::Drive Method

```
object.IsIBMXGAResolution()
```

Figure 8. Non-Portable Method

```
object.Show( HWND window )
object.Show( CCompilerSpecificClassShowInfo& )
```

Figure 9. Non-Portable Parameters

```
object.Show( PictureInfo& )    // where this parm contains no
                               // OS, HW, C++ compiler info
```

Figure 10. Portable Method

We have a base class `Building` with subclasses `Skyscraper`, `House`, and `Barn`. Our platform or environment for the `Building` object is the `city`.

In our subsystem, we need to get the number of floors in a current `Building`.

At runtime, the user has plugged in the fact that `Skyscraper` and `House` exist in the environment `city` (not `Barn`). If we use the base virtual function `Building.GetFloors()`, then our subsystem does not depend on the existence of any of the subclasses.

If our subsystem was written to use `Barn.GetFloors()` instead of `Building.GetFloors()`, then our subsystem would be broken for the `city` environment.

Figure 11. Platform Independence with Generic Objects

Debugging and Testing

Although the procedural code base was mature ship-level code, we experienced many growing pains while wrapping this code to behave like OO code. These pains revealed that the OO wrapper code was imposing a behavior not intended or anticipated by the procedural subsystem.

For example, wrapping allowed the objects to become so accessible that the underlying procedural subsystem had trouble managing each object, to the point where it became more of a stress test of the procedural subsystem than a test of object accessibility.

This situation was further compounded when multiple applications were running, each creating multiple objects. The procedural subsystem was designed more with the intent that an application creates only

one instance to get to the subsystem. Therefore, to get more than one instance in the system, the user creates multiple applications.

The OO wrapper also brought out problems in timing, counterproductivity in resource management, and methods that contradicted each other when used together. With each of the problems, we spent time debugging the procedural subsystem to find the cause, then reported these problems to the appropriate development team. Due to inherent design differences, some problems could not be fixed. Others that could be fixed required further testing, while still others required workarounds in the OO wrapper code. The result was that the OO wrapper code was unable to support all of the methods defined in the OO user interface layer, and project schedules lengthened due to these unforeseen problems.

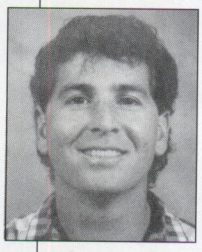
OO Goals Met

We chose to implement OO wrappers in the first phase of our project because of our project's priorities, tight schedule, and our company's large investment in a procedural system that is still maturing. Of course, there is a price to pay for OO wrapping–more time is spent mapping behaviors, debugging, and testing the procedural base than performing OO programming. However, since our top priority was to enable OO applications to use our subsystem with an OO interface, the quickest way to accomplish this was to provide the OO wrappers.

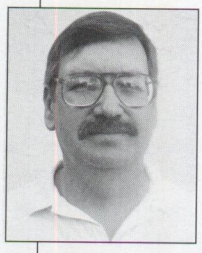
In spite of the pitfalls and our relative inexperience, we met the goals of the project on schedule. We can be confident that future software technologies are being exploited today while still maintaining a strong hold on current investments.

In retrospect, given a different set of priorities and longer time line, we would not have chosen the OO wrapper route. Instead, we would have proceeded directly into the OO subsystem implementation. For example, if our priority were to enable subclassers (subsystem extension developers) to provide new components for the OO subsystem, we would choose the OO implementation over the procedural.

The experiences drawn from this project are applicable to other projects that have goals of exploiting object-oriented technologies. In doing so, the priorities must be carefully determined and a realistic schedule put in place to manage unexpected problems. Given these considerations, you can move to object-oriented technologies in an evolutionary, as opposed to revolutionary, manner with success and return on investment.



Chris Dinallo is an advisory programmer in Natural Computing Technologies and Solutions, IBM Boca Raton, Florida, where he is the lead audio architect. He has been in multimedia related development since it began in March 1990. Within multimedia, Chris has several patent disclosures and technical publishings. His prior experience includes firmware and DOS kernel programming. He received a Bachelor's degree in Computer Science from the University of Florida.



Dan Dorrance is an advisory programmer in Natural Computing Technologies and Solutions, IBM Boca Raton, Florida. He is responsible for software development of object-oriented multimedia frameworks. Since joining IBM in 1983, he has worked on the IBM System/88, ROLM design, Advanced Servers, and OS/2 multimedia development. Before joining IBM, Dan worked in various software development capacities for Bell Laboratories, Eastern Airlines, and Siemens Communication. He has a BS degree in Mathematics from King's College and an MS in Computer Science from Northwestern University. His Internet userid is `ddorrance@vnet.ibm.com`.

Wendi Nusbickel is an advisory programmer in Natural Computing Technologies and Solutions, IBM Boca Raton, Florida, where she is currently developing object-oriented subsystems. She joined IBM Boca Raton in 1983, where she has worked on IBM PC XENIX, OS/2 development, OS/2 Special Projects and Advanced Servers, and Multimedia Software Development. Wendi was a lead developer in OS/2 utilities and is recognized in the industry for her accomplishments in networked full-motion video technology. She has a BS degree in Computer Science from North Dakota State University.

Ann Mizell Robinson is an advisory programmer in Natural Computing Technologies and Solutions, IBM Boca Raton, Florida, where she is responsible for object-oriented subsystems for multimedia. She joined IBM in 1982, developing communications software for the PC. She was one of the first members of the OS/2 development team and became one of its lead designers. Ann co-authored the book *OS/2: Features, Functions, and Applications*, published in 1988 by John Wiley & Sons. In 1989, she joined a startup company, Citrix Systems, Inc., where she worked on development of a multiuser OS/2. She returned to IBM in 1992 to work on OS/2 multimedia subsystems. Ann has a BS degree in Systems Engineering from Case Western Reserve University.

OS/2 Tips and Techniques

Running CHKDSK

You should run CHKDSK regularly as a maintenance process for your computer systems. Because the SWAPPER.DAT file is in a constant state of change, CHKDSK may show misleading results if it is issued when OS/2 has been booted from the hard drive. For best results, run CHKDSK from the floppy diskette.

CHKDSK first cleans up primary structures, then secondary structures; therefore, the CHKDSK procedure may require several passes. Use the following steps to run CHKDSK:

1. Properly shut down OS/2 by placing the mouse cursor on the desktop and clicking the right mouse button. Choose "Shut down" from the pop-up menu. Follow the instructions until the system has told you it is safe to turn off the system or reboot.
2. Insert the OS/2 Installation Disk in drive A:.
3. Press Ctrl+Alt+Del to warm boot your system, or turn your system off for 30 seconds, then turn it on to restart.
4. When prompted, remove the Installation Disk, insert Disk 1, and press Enter.
5. When the Welcome Screen displays, press Esc to display the command prompt A:.
6. Remove Disk 1 and insert Disk 2 (which contains the CHKDSK command).
7. To correct any errors on the hard drive, type the appropriate CHKDSK command followed by the drive letter on which OS/2 resides.
   - If your drive is formatted for a File Allocation Table (FAT) file system, type CHKDSK X: /F (where X: is the drive letter on which OS/2 resides).
   - If your drive is formatted for a High-Performance File System (HPFS), type CHKDSK X: /F:2 (where X: is the drive letter on which OS/2 resides).
   - If you installed OS/2 from a CD-ROM drive, type the following to change to the DISK_2 subdirectory: X:\OS2SE21\DISK_2 (where X: is the CD-ROM drive letter).
8. Press Enter; the CHKDSK procedure will start.

If you have any errors, you will need to run the CHKDSK command repeatedly until all errors are cleared.

Note: If you are getting the same errors after you repeatedly run CHKDSK, remove the diskette, then press and hold Ctrl+Alt+Del to restart your system.

Error After Successful Installation

If you have a Micropolis Small Computer System Interface (SCSI) 1 GB hard drive, you might get the following error message after you have successfully installed OS/2:

```
System cannot find the file KBD01.SYS. This driver program or data file is not located in the default path or the path specified for it is not in the CONFIG.SYS file. Install this file in the correct directory or the appropriate CONFIG.SYS file statement.
```

However, the correct statements are in the CONFIG.SYS file, and the KBD01.SYS file is located in the correct place on the hard drive. To eliminate this error message, disable the "Read ahead cache" on the Micropolis drive. A utility on the Micropolis BBS called UTIL.EXE allows you to do this.

System Hangs When Using Dual Boot to Go from OS/2 to DOS

Follow the steps below if your system hangs when you use Dual Boot to go from OS/2 to DOS:

1. Boot to OS/2 from the hard drive.
2. Shut down the system.
3. Insert a DOS bootable diskette in drive A: and reboot the machine.
4. Type the commands found in Figure 1.
5. Take the diskette out of drive A: and reboot (cold or warm).

Note: If SYS.COM cannot be found in drive A:, copy it from your \DOS subdirectory on the hard drive.

System Hangs When Using Dual Boot to Go from DOS to OS/2

Follow the steps below if your system hangs when you use Dual Boot to go from DOS to OS/2:

1. Boot DOS from the hard drive.
2. Reboot your machine using the OS/2 Installation Disk and Disk 1.

```
C: <Enter>
COPY C:\CONFIG.SYS C:\OS2\SYSTEM\CONFIG.OS2 <Enter>
COPY C:\AUTOEXEC.BAT C:\OS2\SYSTEM\AUTOEXEC.OS2 <Enter>
COPY C:\OS2\SYSTEM\AUTOEXEC.DOS C:\AUTOEXEC.BAT <Enter>
COPY C:\OS2\SYSTEM\CONFIG.DOS C:\CONFIG.SYS <Enter>
DEL C:\OS2\SYSTEM\CONFIG.DOS <Enter>
DEL C:\OS2\SYSTEM\AUTOEXEC.DOS <Enter>
RENAME C:\OS2\SYSTEM\BOOT.DOS C:\OS2\SYSTEM\TEMP <Enter>
A: <Enter>
SYS C: <Enter>
```

Figure 1. Dual Boot from OS/2 to DOS

```
C: <Enter>
COPY C:\CONFIG.SYS C:\OS2\SYSTEM\CONFIG.DOS <Enter>
COPY C:\AUTOEXEC.BAT C:\OS2\SYSTEM\AUTOEXEC.DOS <Enter>
COPY C:\OS2\SYSTEM\AUTOEXEC.OS2 C:\AUTOEXEC.BAT <Enter>
COPY C:\OS2\SYSTEM\CONFIG.OS2 C:\CONFIG.SYS <Enter>
DEL C:\OS2\SYSTEM\CONFIG.OS2 <Enter>
DEL C:\OS2\SYSTEM\AUTOEXEC.OS2 <Enter>
RENAME C:\OS2\SYSTEM\BOOT.OS2 C:\OS2\SYSTEM\TEMP <Enter>
A: <Enter>
SYSINSTX C: <Enter>
```

Figure 2. Dual Boot from DOS to OS/2

3. Press Esc at the Welcome panel to get a command prompt.
4. Type the commands found in Figure 2.
5. Take the diskette out of drive A: and reboot (cold or warm).

Installing Microsoft MS-DOS 6.2 Upgrade on an OS/2 Dual Boot System

You must install the MS-DOS 6.2 upgrade with DOS as the current operating system. If you are running OS/2 as the current operating system, follow the steps below:

1. To boot to DOS, open an OS/2 window and type:

   `BOOT /DOS`

2. Once the boot procedure is complete, place the MS-DOS 6.2 upgrade diskette in the appropriate drive and type:

   `A:SETUP` (if the diskette is in the A: drive) or

   `B:SETUP` (if the diskette is in the B: drive)

3. After a proper upgrade has been completed, review the `CONFIG.SYS` and `AUTOEXEC.BAT` files for the proper OS/2 statements. If they are not there, refer to the *MS-DOS 6.2 Installation Guide* and place the statements as necessary.
4. After these statements have been verified, dual boot should function as before. To return to OS/2, type:

   `C:\OS2\BOOT /OS2`

Installing Microsoft MS-DOS 6.2 Upgrade on an OS/2 Boot Manager System

You must install the MS-DOS 6.2 upgrade with DOS as the current operating system. If you are running OS/2 as the current operating system, follow the steps below:

1. Properly shut down OS/2 by placing the mouse cursor on the desktop and clicking the right mouse button. Choose "Shut down" from the pop-up menu. Follow the instructions until the system has told you it is safe to turn off the system or reboot. Rather than turn the machine off, execute a warm boot by pressing Ctrl+Alt+Del.
2. When the Boot Manager menu appears, choose to boot to the DOS partition.
3. Once the boot procedure is complete, place the MS-DOS 6.2 upgrade diskette in the appropriate drive and type:

   `A:SETUP` (if the diskette is in the A: drive) or

   `B:SETUP` (if the diskette is in the B: drive)

4. Once the upgrade has completed, use the DOS FDISK program to set the Boot Manager partition as ACTIVE, or refer to page 132 of the *OS/2 2.1 Installation Guide* on "making boot manager startable."

If any problems occur while trying to upgrade to MS-DOS 6.2, please refer to Microsoft Technical Support.

Full Partition Messages with OS/2 2.1

You may encounter one of the following error conditions if you have insufficient hard drive space on your OS/2 partition:

- `SYS1477: Warning! The partition containing the SWAPPER.DAT file is full. You may lose data. Do not ignore this message!` (This message appears when you try to open an application.)
- `Disk Error: The INI file cannot be written to disk. The updates are being held to automatically retry the operation but will be lost if the system is shut down before correcting the problem.` (This message appears continuously on the desktop. You may not be doing anything when it appears.)
- A message dialog box containing garbage such as binary characters continuously appears on your desktop.

To resolve these errors, it is necessary to free up disk space by deleting or moving old, unused, or unnecessary files from the OS/2 partition or by relocating the `SWAPPER.DAT` file to another partition.

Review the files you have on the OS/2 partition to determine if you can move or delete any of them. There might be OS/2 files or programs that you do not want, need, or use. A comprehensive list of these files and their locations in OS/2 2.1 is contained in Appendices E and F of the *OS/2 2.1 Using the Operating System* manual that comes with the operating system.

Another option is to move the `SWAPPER.DAT` file to another partition (if one exists) with more space. To move the `SWAPPER.DAT` file:

1. Edit the `CONFIG.SYS` file in an editor such as the OS/2 System Editor. At an OS/2 command prompt, type `E CONFIG.SYS` and press Enter.
2. Create a `D:\SWAPPER` subdirectory.
3. Change the `SET SWAPPATH` statement to point to the new drive letter. For example, change `SET SWAPPATH=C:\OS2\SYSTEM 2048 2048` to read `SET SWAPPATH=D:\SWAPPER 2048 2048`.
4. Save the changes to the `CONFIG.SYS` file and exit the editor.
5. Change to the newly designated partition (`D:` in this case), type `MD SWAPPER` to create the new directory, and press Enter.
6. Perform a proper shut down of OS/2 and reboot. You will now have a new `SWAPPER.DAT` file.
7. To delete the old `SWAPPER.DAT` file, type `CD OS2\SYSTEM` and press Enter. Then type `DEL SWAPPER.DAT` and press Enter.

Note: No data will be lost. The data contained in the `SWAPPER.DAT` file was swapped-out memory that was no longer necessary after the system was shut down.

Corrective Service Information

Figure 1 shows maintenance release levels for the listed products. This information is effective as of February 7, 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	IP07045	IP07045	4-28-94	Includes IP07001, IP07003 (DLR), and I07005. This package has a co-requisite for WR07045 NTS/2 ServicePak, which ships with it. If you already have a refresh level of 3.00.1, you don't need this package.
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	2/7/95	
DDCS/2 ServicePak	2.0	WR07031	WR07031	2/7/95	
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	WR07037	WR07037	2/7/95	
Software Developers Kit/2 (SDK/2)	1.2	WR07038	WR07038	2/7/95	
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	

Figure 1. Maintenance Release Levels **(continued on next page)**

Product/Component	Release	CSD Level	PTF Number	Change Date	Comments
LAN Distance ServicePak	1.1	IP07050	IP07050	10/18/94	
OS/2 Network Transport Services/2 SelectPak	2.00	WR07045	WR07045	4-27-94	
OS/2 LAN Adapter and Protocol Support SelectPak	2.20.2	WR07045	WR07045	4-27-94	
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

IBM Personal Systems

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.

___ 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

___ April 1993

XGA-2: Improving on a Good Thing
IBM Personal Software Products: Product Line Update
Using REXX to Customize the Workplace Shell
OS/2 Distributed Systems Management with LAN NetView
Priming and Querying Your Start Network
Multimedia Applications on IBM Token-Ring LANs
OS/2 2.0 Print Tips
Testing OS/2 PM Applications
Accessing a Remote AS/400 Using OS/2 Extended Services
Virus Information and Protection
Migrating from APPC/PC to Networking Services/DOS
OS/2 2.0 Resources
OS/2 32-Bit Application Migration Workshops
IBM OS/2 Products Available on CD-ROM

___ January 1993

PS/2 Desktop Security
IBM 486SLC2: System Performance Implications
Micro Channel Developers Association
Trackpoint II: The In-Keyboard Pointing Device
Why OS/2 2.0?
OS/2 Distributed Systems Management
CID: Remote OS/2 Configuration, Installation, and Distribution of PC Software
Start/2: Putting the Configuration into CID
LAN Server 3.0: New Thresholds in High-Performance Network Software
The Future of IBM LAN Network Management
Understanding and Using the Workplace Shell
Distributed Processing: A Case Study
Parallel Port Protocols
Developing OS/2 PM Applications with Micro Focus COBOL
OS/2: How About Notebooks?
Loadable ABIOS

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 _____

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.

Take OS/2 to the next level of performance!

Classes cover version 2.X to Warp!

OS/2 Accredited

OS/2 World
Conference & Exhibition
July 17-21, 1995

Salem Krieger

OS/2 World Conference & Exhibition is owned and produced by:

Miller Freeman, Inc.

RIVETING TECHNICAL PROGRAM

Choose from 120 classes that address your toughest OS/2 issues.

Our focused technical program is designed by the editors of *OS/2 Magazine* and in collaboration with IBM's Personal Software Products Division, which guarantees top-quality information.

You'll take home a copy of the Official 1995 Conference Proceedings—topics include: ■ **Client/server** ■ **Management** ■ **Local area networking** ■ **System administration** ■ **Programming** ■ **Multimedia** ■ **Installation and Maintenance**

COMPLETE PRODUCT EXHIBITION

Test the hottest OS/2 products. Representatives from leading companies will answer questions about your specific OS/2 challenges. You'll learn first-hand about industry trends that will affect your use of OS/2 in the future.

ENGAGING SPECIAL EVENTS

OS/2 World emphasizes hands-on learning in every aspect of the conference. The wide variety of special activities offered will guarantee a good time--and a wise investment. You'll network with peers and make valuable business contacts throughout the OS/2 community. Special events include:

■ **OS/2 and LAN Certification Testing** ■ **IBM Product Test Drive Area** ■ **Three Insightful Keynote Addresses** ■ **OS/2 User Group Meetings** ■ **National Team OS/2 Day** ■ **Vendor-Sponsored Evening Receptions** ■ **Vendor Presentation Arena** ■ **OS/2 World Bookstore**

Hynes Convention Center, Boston, MA
Tutorials: July 17
Conference & Exhibition: July 18-20
Certification Testing: July 18-21

For more info fax, mail or call today!

Phone: (415) 905-2354
Fax: (415) 905-2220
Mail:
OS/2 World Conference & Exhibition
600 Harrison Street, 4th fl., San Francisco, CA 94107
E-mail: os/2world@mfi.com

OS/2 World Conference & Exhibition

YES! **I'm interested in:** ❒ **Attending** ❒ **Exhibiting**

Name

Title Company

Address

City State/Province Zip/Postal Code

Phone Fax E-mail PSO3

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

Available and Ready for IBM LAN

Available and Ready for OS/2

Please circle #10 on reader service card.