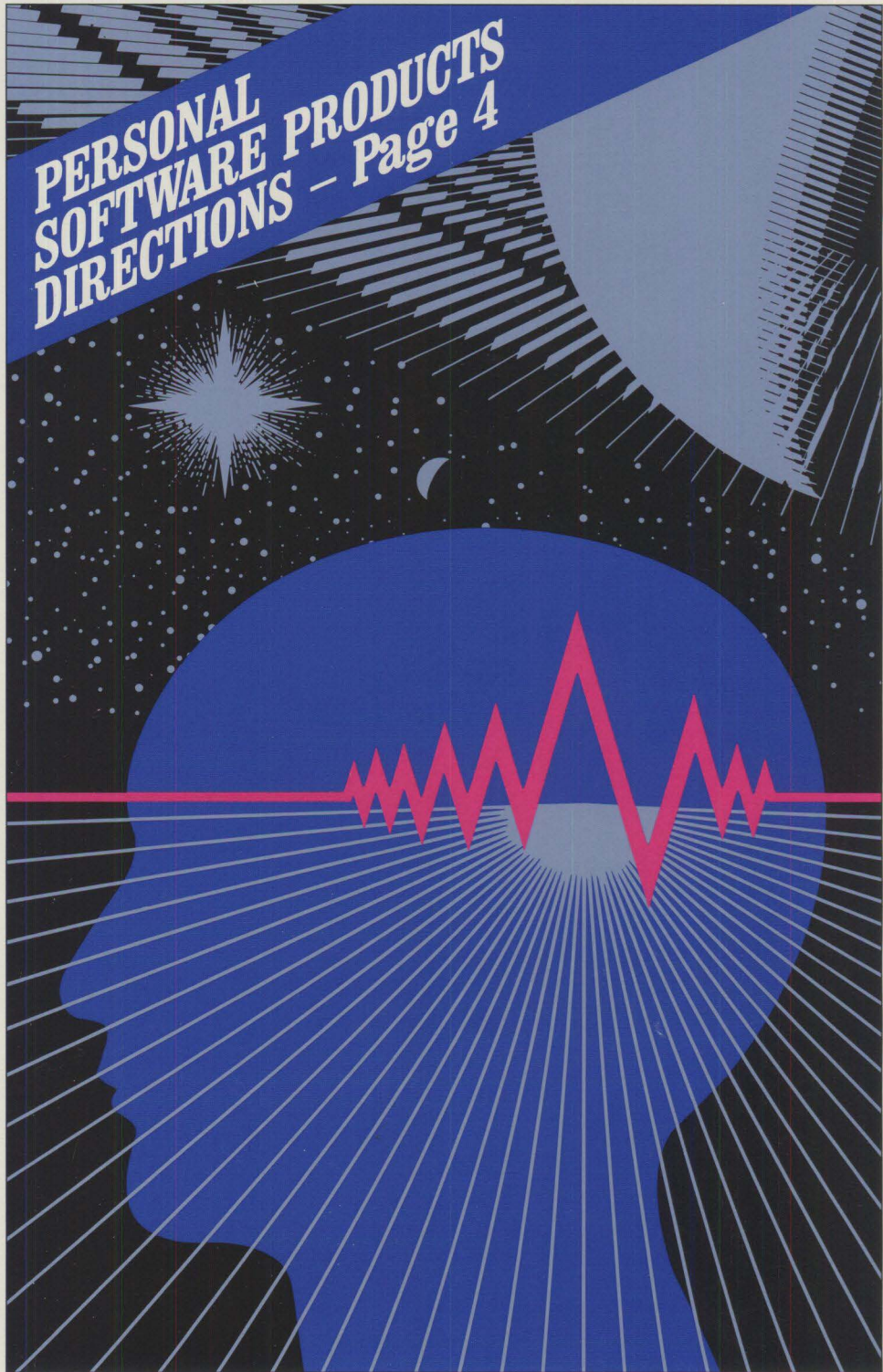


April 1993

Personal Systems



IBM Personal Systems Technical Solutions



\$12.00

IBM Personal Systems Technical Solutions is published quarterly by the IBM National Software Systems Support Center, International Business Machines Corporation, Roanoke, Texas, U.S.A.

Acting Editor	Elisa Davis
Cover Design	Corporate Graphics
Illustrator	Bill Carr
Publication Services	The TDA Group
Typesetting & Diagrams	Genesis Publications
Manager	Bill Hawkins

To correspond with *IBM Personal Systems Technical Solutions*, please write the editor at:

IBM Corporation
Internal Zip 40-A2-04
One East Kirkwood Blvd.
Roanoke, TX 76299-0015

To subscribe to this publication, call 1-800-551-2832. IBM employees should order through SLSS using order form number GBOF-1229.

IBM is now accepting paid advertising in the *IBM Personal Systems Technical Solutions* Magazine. For more information, contact Bill Hawkins, IBM, Internal Zip 40-A2-04. Phone: (817) 961-7376. Fax: (817) 961-5666.

Copying or reprinting material from this magazine is strictly prohibited without the written permission of the editor. Titles and abstracts, but no other portions, of information in this publication may be copied and distributed by computer-based and other information-service systems.

IBM believes the statements contained herein are accurate as of the date of publication of this document. However, IBM hereby disclaims all warranties as to materials and workmanship, either expressed or implied, including without limitation any implied warranty of merchantability or fitness for a particular purpose. In no event will IBM be liable to you for any damages, including any lost profits, lost savings or other incidental or consequential damage arising out of the use or inability to use any information provided through this service even if IBM has been advised of the possibility of such damages, or for any claim by any other party.

Some states do not allow the limitation or exclusion of liability for incidental or consequential damages so the above limitation or exclusion may not apply to you.

This publication could contain technical inaccuracies or typographical errors. Also, illustrations contained herein may show prototype equipment. Your system configuration may differ slightly.

IBM has tested the programs contained in this publication. However, IBM does not guarantee that the programs contain no errors.

This information is not intended to be a statement of direction or an assertion of future action. IBM expressly reserves the right to change or withdraw current products that may or may not have the same characteristics or codes listed in this publication. Should IBM modify its products in a way that may affect the information contained in this publication,

IBM assumes no obligation whatever to inform any user of the modifications.

Some of the information in this magazine concerns future products, or future releases of products currently commercially available. The description and discussion of IBM's future products, performance, functions, and availability are based upon IBM's current intent and are subject to change.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not imply giving license to these patents.

It is possible that this material may contain reference to, or information about, IBM products (machines and programs), programming or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such products, programming or services in your country.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation whatever.

All specifications are subject to change without notice.

© Copyright 1993 by International Business Machines Corporation



Printed on recycled paper

Contents

Hardware

- 1** XGA-2: Improving on a Good Thing

Software

- 4** IBM Personal Software Products: Product Line Update
- 22** Using REXX to Customize the Workplace Shell
- 33** OS/2 Distributed Systems Management with LAN NetView
- 41** Priming and Querying Your Start Network
- 47** Multimedia Applications on IBM Token-Ring LANs
- 55** OS/2 2.0 Print Tips
- 62** Testing OS/2 PM Applications
- 67** Accessing a Remote AS/400 Using OS/2 Extended Services
- 72** Virus Information and Protection
- 77** Migrating from APPC/PC to Networking Services/DOS
- 82** OS/2 2.0 Resources

Random Data

- 90** Little Solutions
- 95** New Products
- 99** OS/2 32-Bit Application Migration Workshops
- 101** IBM OS/2 Products Available on CD-ROM

Trademarks

IBM, PS/2, OS/2, Micro Channel, NetView, Systems Application Architecture, AS/400, Presentation Manager, Workplace Shell, AIX, PROFS, OS/400, DB2, Audio Visual Connection, SQL/400, AT, and RISC System/6000 are registered trademarks of International Business Machines Corporation.

SAA, AD/Cycle, XGA, Ultimedia, OfficeVision, 3090, SQL/DS, DRDA, CUA, Common User Access, PowerPC, Information Warehouse, Distributed Relational Database Architecture, S/390, System/370, PS/ValuePoint, VM/ESA, AIX/ESA, VSE/ESA, CICS/VSE, and AIX/6000 are trademarks of International Business Machines Corporation.

3Com and EtherLink II are registered trademarks of 3Com Corporation.

PostScript is a registered trademark of Adobe Systems, Inc.

TrueType, Apple, Mac, and Macintosh are registered trademarks of Apple Computer, Inc.

ATI is a registered trademark of ATI Technologies, Inc.

AutoCAD is a registered trademark of Autodesk, Inc.

Borland is a trademark of Borland International.

Carnegie-Mellon is a registered trademark of Carnegie-Mellon University.

CASE:PM is a trademark of CASEWORKS, Inc.

CompuServe is a registered trademark of CompuServe Incorporated.

DECnet is a trademark and DEC is a registered trademark of Digital Equipment Corporation.

Headland is a registered trademark of Headland Technology, Inc.

HP, LaserJet, Hewlett-Packard, and DeskJet are registered trademarks of the Hewlett-Packard Company.

Hitachi is a registered trademark of Hitachi Ltd.

INFORMIX is a registered trademark of Informix Software, Inc.

POSIX is a trademark and IEEE is a registered trademark of the Institute of Electrical and Electronics Engineers.

Indeo and i386 are trademarks and ActionMedia, Intel, and DVI are registered trademarks of Intel Corporation.

EASEL is a registered trademark of Interactive Images, Inc.

COMDEX is a registered trademark of the Interface Group, Inc.

ISO is a registered trademark of the International Organization for Standardization.

LANQuest Labs is a registered trademark of LANQuest Group.

Lexmark is a trademark of Lexmark International.

Lotus and Lotus Notes are registered trademarks of Lotus Corporation.

Microsoft is a registered trademark and Windows is a trademark of Microsoft Corporation.

Motorola is a trademark of Motorola, Inc.

NEC is a registered trademark of NEC Corporation.

Novell and NetWare are registered trademarks of Novell, Inc.

Motif, Open Software Foundation, OSF/1, and OSF are registered trademarks of the Open Software Foundation.

ORACLE is a registered trademark of Oracle Corporation.

Radius is a trademark of Radius, Inc.

Sony is a registered trademark of Sony Corporation.

NFS is a registered trademark of Sun Microsystems, Inc.

Encina and Transarc are registered trademarks of Transarc Corporation.

Toshiba is a trademark of Toshiba Corporation.

UNIX is a registered trademark of UNIX System Laboratories.

Western Digital is a registered trademark of Western Digital Corporation.

WordPerfect is a registered trademark of WordPerfect Corporation.

Ethernet is a trademark of Xerox Corporation.

X/Open is a trademark of X/Open Company, Limited.

XVT is a trademark of XVT Software, Inc.

XGA-2: Improving on a Good Thing

Jim Paolantonio
IBM Corporation
Boca Raton, Florida

IBM introduced Extended Graphics Array-2 (XGA-2) in September 1992. XGA-2 significantly enhances the original XGA design. XGA-2 provides stable, flicker-free viewing through high refresh non-interlaced video.

To appreciate Extended Graphics Array (XGA™), it is necessary to review some video history. Video Graphics Array (VGA) was the original video integrated into IBM PS/2® personal computer systems in September 1987. VGA combined all the predecessor modes of the Monochrome Display Adapter (MDA), the Color Graphics Adapter (CGA), and the Enhanced Graphics Adapter (EGA), while offering six new VGA modes. VGA had a video buffer consisting of 256 KB of Dynamic RAM (DRAM), which enabled graphic resolutions of up to 640 x 480 with 16 colors.

For users requiring higher resolution and performance, IBM introduced the 8514/A Display Adapter at the same time as VGA. The 8514/A offered resolutions up to 1024 x 768 with 256 colors and hardware-assisted drawing functions.

XGA in Review

IBM first offered XGA in October 1990 as the standard base video in the PS/2 Models 90 and 95 XP 486 systems. The XGA video subsystem consists of two main chips: the Display Controller Chip (DCC) and the Serializer/Palette/DAC (SPD). XGA

video memory consists of Video RAMs (VRAMs). VRAMs are dual-ported, eliminating the bandwidth bottleneck suffered in the past by VGA video using single-port DRAMs.

XGA video has three mutually exclusive operating modes. XGA has built-in VGA-compatible hardware that supports all VGA operating modes. Existing VGA applications run unchanged on XGA hardware. Dual-ported VRAMs on XGA increase VGA mode performance. Up to 1 MB of VRAM is supported on XGA. A 132-column text mode (25 rows of 8-pel-wide characters) is incorporated into XGA mainly to support Mainframe Interactive (MFI) 3270 terminal emulation applications.

The essence of XGA is the Extended Graphics Function mode. In this third mode, a graphics coprocessor performs hardware-assisted drawing functions including line drawing, Bit Block Transfers (BitBLT), and area fill operations. These hardware functions, optimized for graphical windowing environments such as OS/2® Presentation Manager® (PM) and Microsoft® Windows™, increase performance by relieving the host processor from per-

forming comparable operations in software. XGA is a 32-bit busmaster, enabling it to transfer bitmaps between system memory and the local video buffer.

XGA's SPD contains 6-bit red, green, and blue Digital-to-Analog Converters (DACs), enabling a total range (palette) of 256 K colors. With 512 KB of VRAM installed in the Extended Graphics mode, XGA provides a graphics resolution of 640 x 480 with 256 K colors or 1024 x 768 with 16 colors (from a palette of 256 K colors). With 1 MB of VRAM, a 16-bit "direct color" mode is available at 640 x 480 resolution. This supports the display of 65,536 colors simultaneously, permitting the portrayal of photo-realistic images. At 1024 x 768 resolution, 256 colors can be displayed with 1 MB of VRAM.

In the Extended Graphics mode, XGA has a 64 x 64 pixel (four color) hardware sprite (cursor). The sprite is stored in a separate memory buffer and overlays the video background. This increases system performance by eliminating host processor save/restore operations normally required to manipulate conventional software cursors.

XGA uses non-interlaced scanning with a frame refresh rate of 60 or 70 Hz, at the 640 x 480 resolution. At the 1024 x 768 resolution, XGA uses interlaced scanning with a frame refresh rate of 44 Hz to conserve video bandwidth.

XGA's Software Device Drivers

The original XGA hardware was shipped with several software device drivers. These device drivers provide an interface between the application

programs and the XGA hardware for different operating systems. The DOS Adapter Interface device driver enables applications written for the 8514/A Adapter to run unchanged on XGA hardware. XGA device drivers for Microsoft Windows 3.0 and OS/2 PM 1.2 graphical windowing environments and AutoCAD® (Releases 10 and 11) also are provided.

Introducing XGA-2

XGA-2 video is now integrated into the system boards of PS/2 Models 56 and 57 486 SLC2. XGA-2 is available as a Micro Channel® adapter and is standard with PS/2 Models 76, 77 486 SX, 77 486DX2, 95 XP 486, and the Model 95 Server. IBM Ultimedia™ systems also come with XGA-2 video. Radius™, Inc. (an OEM partner with the IBM Visual Subsystems group) has developed an XGA-2 adapter for the Industry Standard Architecture (ISA) bus.

XGA-2 provides more of everything that users want in XGA. XGA-2 supports more colors, more operating modes, and higher non-interlaced frame refresh rates. In addition, design enhancements have been made to the SPD, graphics coprocessor, and memory controller. Device drivers have been updated to support improvements in XGA-2 and to add new features.

More Colors

The SPD in XGA-2 contains 8-bit DACs as compared to 6-bit DACs on the initial XGA. This gives a palette of 24 bits or 16.7 million colors. The trade press has coined 24-bit color as "true color" since 16.7 million colors is the maximum that the human eye can distinguish.

More Modes and Attributes

Since the arrival of VGA and before XGA, vendors have been trying to increase screen resolutions beyond 640 x 480 while maintaining the same 256 KB of DRAM. A flurry of pro-

Resolution	Colors	Scanning	Refresh Rate
640 x 480	256 ¹	Non-interlaced	Up to 75 Hz
640 x 480	64 K	Non-interlaced	Up to 75 Hz
800 x 600	256 ¹	Non-interlaced	Up to 75 Hz
1024 x 768	256 ¹	Non-interlaced	Up to 75 Hz
1280 x 1024	16 ¹	Interlaced ²	45 Hz
1360 x 1024	16 ¹	Interlaced ²	45 Hz

¹ Out of a color palette of 16.7 million colors

² Because of hardware design constraints, interlaced scanning is required.

Figure 1. XGA-2 Display Characteristics

prietary VGA modes have evolved. The most prevalent "Super-VGA" mode is the 800 x 600 resolution. XGA-2 supports the Super-VGA 800 x 600 resolution mode with 64 K colors.

*XGA-2 offers
non-interlaced scanning
at a high frame refresh
rate of 75 Hz at
1024 x 768.*

The new SPD in XGA-2 supports MFI text-mode character attributes, such as character underline. The SPD features a new programmable pixel clock—the original XGA had fixed oscillator frequencies—that allows for 9-pel-wide characters in the 132-column text mode. This creates more legible text by allowing more space between characters.

"Flicker-Free": It's the Law

Interlaced scanning alternately scans the odd (one field) and even lines (the other field) of an image. Two fields compose one frame. Interlaced scanning, along with the high brightness on today's CRTs, accentuates flicker. The most direct method to eliminate flicker is to use non-interlaced scan-

ning with higher frame refresh rates. An impetus to this has been a set of International Standards Organization (ISO®) health and safety standards adopted by the European Community (EC). ISO Standard 9241, Part 3 specifically states that computer displays must provide a stable, flicker-free image. The higher speed (90 MHz) and programmable pixel clock in the new SPD enable XGA-2 to run at high non-interlaced frame refresh rates.

XGA-2 offers non-interlaced scanning at a high frame refresh rate of 75 Hz at 1024 x 768—previously, interlaced scanning was used. XGA-2 also supports the new non-interlaced IBM displays: Models 9515, 9517, and 9518. See Figure 1 for display characteristics.

Graphics Coprocessor Enhancement

In the original XGA design, the graphics coprocessor did not support hardware-assisted drawing operations in the 16-bit direct color mode because of chip size constraints. So the original XGA device drivers were limited to 256-color support because of performance considerations.

The coprocessor in XGA-2 now supports the 16-bit direct color mode, and the device drivers have been revised to take advantage of it. The performance of the graphics coproc-

essor also has been enhanced by the addition of an auxiliary coprocessor *busy bit*. This bit enables the host processor to interrogate the status of the XGA coprocessor without interrupting the coprocessor operation in progress.

DMQS: A New Feature

The new Display Mode Query and Set (DMQS) feature provides information to the device drivers about all XGA video subsystems in the system unit and the type of display attached. The XGA information includes the system-unit slot locations of the XGA display adapters, I/O base addresses, memory-mapped coprocessor addresses, and video buffer addresses. The display information identifies the display ID, which reveals the resolution and horizontal/vertical resolution characteristics of the display. In the original XGA design, host software had to interrogate individual XGA registers to obtain this information. DMQS is implemented in the Basic Input/Output System (BIOS) and in the XGA device drivers. DMQS provides the information to take maximum advantage of the attached display's features, such as horizontal and vertical resolution and refresh rates.

Enhanced Device Drivers

Device drivers provided with XGA-2 support DOS Version 5.0, OS/2 Version 2.0, Windows Version 3.1, AIX® PS/2 Version 1.3, and AutoCAD Releases 11 and 12. These device drivers include support for DMQS, which enables the use of both IBM and non-IBM monitors. Windows, OS/2, and AutoCAD device drivers support the 64 K direct color mode.

Summary of XGA-2 Features

The following list summarizes the XGA-2 features.

Graphics coprocessor: The coprocessor performs hardware-assisted

drawing, including line drawing, BitBLT, and area fill operations. This accelerates performance by relieving the host processor from performing these operations in software. These operations are performed for 1, 2, 4, 8, and 16 bits per pel. The original XGA did not support 16-bit-per-pel operations.

Flicker-free image: Non-interlaced scanning up to a 75 Hz refresh rate eliminates flickering. This meets the requirements of ISO Standard 9241, Part 3. Original XGA used interlaced scanning at 1024 x 768 resolution.

*Device drivers support
DOS, OS/2 2.0,
Windows 3.1, AIX PS/2,
and AutoCAD.*

MCA busmaster: The graphics coprocessor can access bitmaps located in both system memory and the local video buffer. This increases performance by offloading the host processor.

Video RAM: Dual-ported Video RAMs increase performance over single-port DRAMs. One megabyte of VRAM is standard on XGA-2.

Hardware sprite: The hardware cursor overlays the video background. It increases performance by eliminating host save/restore operations normally required for software cursors.

16.7 million colors: XGA-2 has 8-bit red, green, and blue DACs providing a color palette of 16.7 million colors. The original XGA has 6-bit DACs with a palette of 256 K colors.

64 K direct color: XGA-2 displays up to 65,536 colors simultaneously,

enabling the portrayal of photo-realistic images.

Display Mode Query and Set: DMQS identifies key XGA subsystem and display parameters, which enables XGA to work with IBM and non-IBM displays.

Super-VGA mode: XGA-2 supports the popular 800 x 600 Super-VGA mode.

Device driver support: XGA software device drivers are provided for DOS, OS/2, AIX PS/2, and Microsoft Windows. XGA device drivers for AutoCAD Releases 11 and 12 also are provided. On XGA-2, this includes the support of the 16-bit direct color mode.

132-column text mode: XGA-2 supports 132 characters (9 pel wide) per row with 25 rows. This is primarily for MFI 3270 terminal emulation. XGA-2 supports MFI character attributes, such as character underline. The original XGA had 8-pel-wide characters with no MFI character attributes.

In conclusion, XGA-2 is positioned to take us into the future. With the explosion of multimedia technology over the next few years, it will be in the middle of the foray.

Jim Paolantonio is an advisory engineer in the IBM Visual Subsystems Group of Entry Systems Technology in Boca Raton, Florida. Jim's current responsibilities include hardware simulation of XGA video. Jim was part of the original development team for XGA video on the PS/2 Model 90 XP 486.

IBM Personal Software Products: Product Line Update

Dee Anne Safford
IBM Corporation
Boca Raton, Florida

This article discusses IBM's product line and planned future directions for personal computing software. It is important to understand the direction IBM is planning so you can make decisions about the right platform to invest in today.

IBM's vision for personal computing in the 1990s is to increase user productivity by providing easy-to-use solutions that take advantage of personal computer capabilities. Fundamental to IBM's vision is making client/server environments, whatever their size or scope, easy to integrate into a business, and easy to use and manage. By providing a wide range of options, establishing strong partnerships, and supporting industry standards, we provide solutions that help users minimize the risks involved in technology investments.

IBM's goal for the 1990s is to bring information and people together. Our approach is simple. We begin with powerful Graphical User Interfaces (GUIs) that "feel" the same regardless of the system being used. Then we add powerful extensions that make access to networked information and resources as simple as if they reside on the desktop. Finally, IBM provides a road to the future—an evolutionary path that helps protect current investments while positioning users to exploit revolutionary new technologies—in both hardware and software.

The IBM Personal Software Products Division

As organizations reshape themselves to become more competitive, they need flexible systems that provide easy access to information and services, regardless of the underlying computing hardware or operating systems, or the location of the information.

Because of the dynamics of today's marketplace, IBM has created a separate division, Personal Software Products (PSP), to focus on delivering flexible, scalable software platforms that can solve diverse business problems. Figure 1 shows the objectives of the PSP division. IBM PSP is building products with a commitment to quality and reliability—a key component to supporting complex business environments.

IBM PSP is investing significantly in the following areas:

- Enhancing DOS, OS/2, and AIX systems with new technology, such as objects, multimedia, and pen support. IBM PSP also supports a broad range of industry-

standard hardware and software, while maintaining application compatibility and coexistence with today's platforms.

- Connecting and supplementing these systems with common components, such as LANs, Extended Services, and LAN administration tools. These components provide scalable and consistent functions and interfaces for client/server environments.
- Delivering distributed information systems built on open industry standards to support client/server environments. These environments can enable transparent access to data regardless of its location.
- Enabling the evolution of distributed information systems by advancing object-oriented technology. Object-oriented technology will make distributed systems more flexible and easier to use, while boosting productivity for software developers.

For users, the focus of IBM PSP will be the following:

- Easy access to information and resources—anytime, anywhere
- Systems that leverage current investments in software, hardware, applications, data, and training—bridging today's islands of information
- Solutions that adapt easily to new technologies, positioning users to embrace these enhancements as they need them

Customer Requirements

Today's marketplace is breeding a new class of business—one that is leaner, flatter, and more competitive.

Increasing competition, growing economic pressures, and an accelerating rate of change are part of these dynamics. As companies grapple with the issues of growth and success—even survival—several common issues emerge.

Cross-department information sharing: While an engineering department may need to run complex modeling programs, it also needs to share information with other departments, such as product planning, finance, personnel, and the market support organization.

Real-time monitoring of business results: Many executives want real-time sales results to help forecast demand, project cash flow, and plan advertising and public relations activities.

Easy access to information regardless of location: Many organizations have mobile sales forces that need online access to product information and the ability to develop sales proposals without being in their offices.

Successful organizations recognize that the ability to respond, even to the slightest change in customer preference or buying patterns, can set them apart. Leading-edge organizations are committed to the effective use of technology—maximizing their current systems while investing in new capabilities that support the information systems.

Historically, these requirements meant different architectures and different hardware and software. Solutions maximized for a single environment have bred islands of information that are difficult to bridge; multiple envi-

ronments are difficult to manage as a single network resource.

Any discussion of the future of operating systems software must take three factors into account: software and hardware innovation will continue, making change a way of life; continued, successful innovation must protect existing investments; and operating systems must integrate a broad range of platforms, from \$300 palmtops to \$3 million workstations.

No one product or solution meets everyone's needs. Selecting solutions that will grow with the company, protect its investments, and enable the work force to share information should not mean giving up the power associated with specialized products or customized applications. They also should not limit the organization's flexibility. These sophisticated requirements, diverse computing environments, and the need to share information make the choice of an operating system platform critical. It

determines the choices available for applications and networking.

Products to Meet These Requirements

IBM's offerings begin with a powerful suite of desktop operating systems and operating system extensions, such as multimedia and pen support. Exciting enhancements also are planned.

DOS



IBM DOS is the entry-level operating system designed for desktop systems, as well as new smaller systems

such as portable computers and handheld devices. Today, DOS runs on Intel® microprocessor-based hardware from IBM or virtually any other PC vendor. IBM continues to optimize DOS for environments that have lower memory and disk space capabilities. This will enable users to

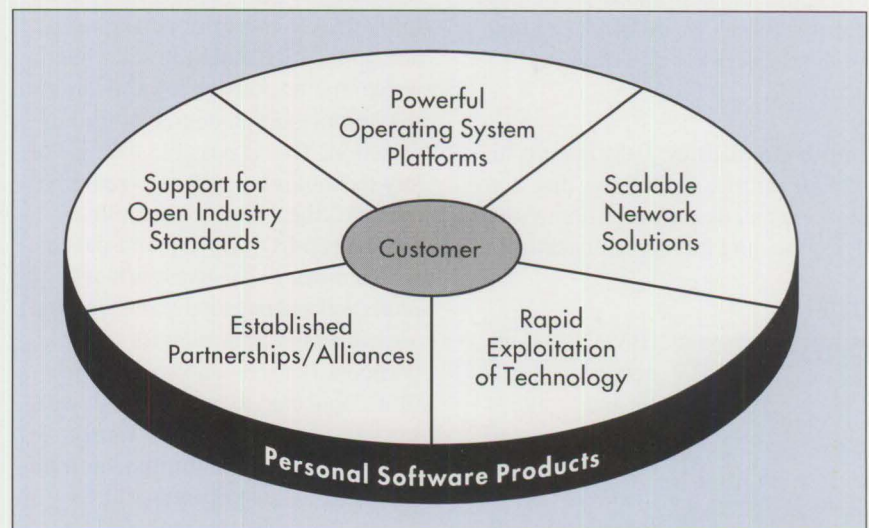


Figure 1. Personal Software Products Division

upgrade their current systems, and supports the emerging handheld devices and portable systems.

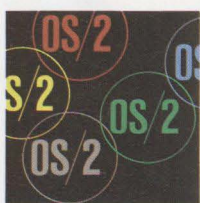
For many customers, DOS is essential to protect their investment in installed hardware. DOS is important for portables, handheld devices, pen-enabled hardware, desktop systems, and even home computing environments. Therefore, we will continue to add capabilities that simplify use, while supporting the following advanced capabilities:

Pen support: IBM recently announced pen support for DOS with PenDOS 2.0. PenDOS 2.0 is an advanced operating system extension available to pen hardware manufacturers worldwide. It supports more than 100 existing mouse-aware DOS applications without modification, and provides support for the development of new pen applications. PenDOS 2.0 opens new possibilities for applications that require a natural interface, such as signature verification, graphics design, and annotation of documents.

One new feature of PenDOS is a set of gestures common across IBM pen systems. Instead of developing new pen applications from the ground up, developers can easily adapt existing DOS programs for pen-based hardware.

Improved utilities: IBM intends to include utilities in DOS for disk compression and defragmentation, virus detection, and full-screen backup.

OS/2



IBM OS/2 is an operating system for Intel micro-processor-based PCs that exploits the power of today's advanced 32-bit PC hardware. Over 400 personal computer types are currently

supported. For users, exploiting 32-bit technology means increased performance and reliability. Developers get an advanced development environment that supports 32-bit graphics, object-oriented technology via the System Object Model (SOM), and multimedia extensions—capabilities that will soon become integral to the desktop. Both users and developers can take advantage of the intuitive GUI, the Workplace Shell®.

The Workplace Shell is the next-generation user interface available today with OS/2 2.0. It was developed as a user interface that works the way people work. It shields users from the complexity of the operating system; they work with familiar objects. For example, users can simply drop a picture of a file onto a picture of a printer to print. It also is flexible, so users can tailor the appearance of their desktops.

OS/2 is the only PC operating system that runs DOS, Microsoft Windows, and OS/2 applications concurrently and seamlessly. Users just bring up the application; OS/2 handles which environment is needed. Unlike DOS and Windows, OS/2 is designed to keep ill-behaved applications from crashing the entire operating system. OS/2 delivers superior performance and application concurrency while remaining responsive to applications that require a high degree of user interaction. This is possible because of key technologies such as preemptive multitasking, priority scheduling, overlapped I/O, and demand-paged virtual memory. Features such as crash protection and preemptive multitasking are especially important for users—particularly users who spend much time connected to a network, downloading files from bulletin boards, or interacting with a mainframe or mini-computer at headquarters.

OS/2 2.0 has benefited from a tremendous amount of user feedback,

and user involvement continues. Based on this feedback, IBM plans to add several key enhancements to OS/2 Release 2.1 for delivery during early 1993.

- **Performance enhancements** have been added to both the Workplace Shell and the base system by reducing resource requirements and providing improved paging algorithms.
- **A 32-bit graphics engine** has dramatically improved the handling of graphic images and state-of-the-art palette management. Users can now create complex graphics that were impossible in a 16-bit environment. Palette management offers more flexibility with colors and fonts than ever before, enabling users to customize individual desktops or to create more realistic looking images.

The 32-bit graphics engine supports 32-bit video display drivers: XGA, Super VGA, and VGA. The Super VGA device drivers are based on the five most popular chip sets—Tseng, ATI®, Western Digital®, Headland®, and Trident—which provides significant coverage of the graphics adapters and displays currently installed.

- **Win-OS/2** has been enhanced to support Microsoft Windows 3.1 applications, including support for the printer device drivers provided with Windows 3.1, TrueType® fonts, multimedia extensions, Object Linking and Embedding (OLE), and most of the applets in Windows 3.1. Dynamic Data Exchange (DDE), OLE, and Clipboard functions are faster. The configuration capability has been improved beyond Win-OS/2 Version 3.0. Win-OS/2 Version 3.1 performance has improved up to 35% over Win-OS/2 Version 3.0.
- **Advanced Power Management (APM) and PCMCIA Level 1**

support has been added. These functions are critical to portable computer users. The APM support will conserve power, improving battery life. The PCMCIA support recognizes the credit card-sized memory and peripheral adapters.

- **Support for additional printers and CD-ROMs** will be included in this release. Printers will include the new Lexmark™ Series 42xx, HP® DeskJet®, and HP LaserJet® IV printers. CD-ROM support will be provided for market leaders such as Sony®, Hitachi®, NEC®, and Toshiba™.

OS/2 Release 2.1 will be shipped on CD-ROM as well as diskettes.

IBM plans to continue to reduce the resource requirements of OS/2, enabling computers with limited disk and memory resources to run more efficiently. Both IBM and PC-compatible hardware vendors will be making additional device drivers available for graphics adapters, Small Computer Systems Interface (SCSI) devices, printers, and various multimedia devices.

PenPM



OS/2 will be enhanced to support pen and speech recognition technologies. With the new pen support (PenPM), users can replace the keyboard or mouse with a pen in DOS, Windows, and OS/2 applications that are not pen-enabled. For developers, an additional set of Application Programming Interfaces (APIs) has been created. They will be shipped in a toolkit so applications that recognize the pen can be developed. We also will provide a set of gestures that will be common across all applications; for example, a delete mark will be mapped to current methods for deletion. Handwriting recognition and

training for using handwriting should be available later in 1993.

Voice recognition technology also is making exciting advances. No longer will users be forced to “train” the software to the sound of their voices. High recognition rates—approaching 100%—will make it more practical to use this advanced capability in general-purpose applications.

Multimedia



OS/2's Multi-Media Presentation Manager/2 (MMPM/2) provides capabilities beyond anything imagined in the design of original PC operating systems. A multimedia operating system must provide superior and dedicated throughput while simultaneously remaining responsive to users. This can be done only by an operating system that coordinates its resources in

this complex environment where graphics, sound, and full-motion video are being integrated in real time. OS/2's underlying technologies are critical to supporting this environment. MMPM/2 Release 1.0 with additional enhancements is planned for inclusion in OS/2 Release 2.1.

MMPM/2 provides a common programming interface to control both multimedia devices and the data streams that flow from these devices. Media Control Interface (MCI) APIs include Windows 3.1 and the Multimedia Extension to Windows 3.0. However, by exploiting the multitasking capabilities of OS/2, IBM has introduced stream management and stream handling to the MMPM/2 extensions to OS/2. As a result, OS/2 application programs can control multimedia devices (through Play, Record, and similar commands), and can control and synchronize the data streams as well. This is important, for example, in ensuring lip synchroniza-



tion with software motion video, as well as for higher function in editing and playback. MMPM/2 and OS/2 is the only software environment where a multimedia audio and video stream can be played while a spreadsheet is calculating or a word processing document is loading in the background.

MMPM/2 includes support for multimedia logical devices (such as audio adapters, CD-ROM drives, and video-disc players) and other hardware devices, such as logical media devices that are managed by the Media Device Manager (MDM). The actual implementation is not relevant to an application because MMPM/2 provides device independence with the command message and command string interfaces. Following are some logical devices currently supported by MMPM/2:

- Amplifier-Mixer
- Waveform Audio
- Sequencer
- CD Audio
- CD XA
- Videodisc

MMPM Toolkit/2 is available to assist product developers. A simple installation from CD-ROM, using the multimedia installation program, is all that is required. Once installed, the toolkit sample programs illustrate the use of the comprehensive multimedia device and data handling capabilities of MMPM/2.

MMPM/2 Release 1.0, with enhancements in audio device driver support, is planned for inclusion in OS/2 Release 2.1.

Advances in CPU power, data storage, and compression technology have provided key technologies for creating and playing digital video data on personal computers. The high capacity disk drives and CD-ROMs

satisfy the large storage needs for digital video data. Additionally, i386™ and equivalent processors give sufficient processing power to handle digital video data in real time. When these advances are combined with image compression techniques, the result is a powerful integration of video and the personal computer.

Several compression algorithms are currently in use throughout the industry. Some of these algorithms, such as RTV, MPEG, and P*64, use additional video hardware to compress and decompress the digitized video. Others like Road Pizza, Compact, Video 1, and Indeo™ are less numerically intensive and can be handled by software running on the main CPU and still maintain sufficient frame rates to effect motion. These are referred to as "software-only" algorithms or "Software Motion Video (SMV)."

While the software-only techniques are attractive due to their low cost, the video quality of these algorithms is typically lower than the hardware-based algorithms. Consequently, producers of digital video data constantly struggle with the issues of trading off between lower quality/cost of SMV versus the higher quality/cost of hardware-assisted video. The result has been an outcry from the industry for a single compression technique capable of providing a spectrum of levels of quality from a single copy of the digital video data. To satisfy these requirements, IBM PSP has developed a state-of-the-art compression technique called Ultimotion.

Ultimotion is a playback time-scalable video data stream. Specifically, the frame rate, output resolution, and color depth characteristics of the video can be set when the video is played, but the resulting characteristics during playback are determined by the capabilities of the playback

platform. The playback platform capabilities affecting the video quality are CPU, data bandwidth, and video adapter. Based on the characteristics of these components, the output quality of a single Ultimotion video stream can be "scaled" to the playback platform.

MMPM/2 Release 1.1 is available today in beta form on a CD-ROM. A copy can be obtained by calling (800) 3IBM-OS2.

Ultimedia Tools Series



It does not matter if you are a newcomer, an intermediate user, or a long-time professional with multimedia applications—using IBM's Ultimedia Tools Series (UTS) will help you create exciting, innovative multimedia applications and presentations. UTS is a series of products from IBM and multiple vendors that adhere to a common architecture. These programs help bring your multimedia applications to life.

Creating a Standard Multimedia Architecture

IBM has done more than just put all the UTS tools in one place and make them simple to order. Together, the creators of UTS products have developed a UTS architecture that will make it easier to use your development tools for years to come.

IBM announced in November 1992 the Ultimedia Builder/2, Workplace/2, and Perfect Image/2 products. These OS/2 tools are follow-on products to the Audio Visual Connection® (AVC). Ultimedia Builder/2 is an authoring tool; Ultimedia Workplace/2 is a SOM-based data management tool; and Ultimedia Perfect Image/2 is an image enhancement tool. These products will be available in mid-1993. Future IBM offerings are planned

that will use advanced information management query technology to enable database searches based on image content characteristics, such as color, texture, and shape. A multimedia server can be connected using OS/2 2.0 and LAN Server 3.0. This server is capable of supporting up to 40 separate full-motion video streams to workstations on 16 Mbits-per-second Token-Ring local area networks.

Also planned is synchronized delivery of sound and motion by a RISC System/6000[®] file server. This offering will run under AIX/6000[™] and has client capabilities for OS/2 and AIX. This means multimedia applications can be created for OS/2 and use data streams from the server.

The UTS architecture means all UTS tools share common file formats and a similar user interface, so it is easy to share files or work with multiple packages. Depending on the choice of programs, you can manipulate multimedia objects across programs, use a clipboard to copy and paste items from one application to another, and use DDE techniques across Windows applications.

In the future, there will be greater interoperability among existing packages, including drag-and-drop capabilities, and regular upgrades to today's UTS product line. So when choosing a development tool from the Ultimedia Tools Series, you can be certain that it will continue to be among the best in the industry.

UTS packages are available for DOS, Windows, and OS/2. UTS product literature is available, as well as a comprehensive CD-ROM disc that includes a tutorial, product information and demonstration, a glossary of multimedia terms, and product selection and ordering information. Call (800) 228-ULTI (8584) for assistance.

AIX

AIX is IBM's version of the UNIX[®] operating system for IBM mainframes (AIX/ESA[™]), RISC-based workstations (AIX/6000), and Intel-based PCs (AIX PS/2). AIX PS/2, developed by the PSP division, is a robust operating system with superb multi-user, multitasking, security, and connectivity features. In mid-1992, IBM released AIX PS/2 Version 1.3, which offers enhanced application performance, improved memory management, and enriched graphics support in a UNIX-based operating environment for Intel-based systems.

Demonstrating IBM's strong commitment to open systems, AIX PS/2 supports interoperability with other AIX, UNIX, IBM, and non-IBM systems. AIX PS/2 Release 1.3 provides the following features:

- Full POSIX[™] IEEE[®] 1003.1 1988 standard compliance, as specified in Section 2.1.2.2 of the IEEE standard
- Enhanced windowing and a GUI through X-Windows Version 11R5 and Motif[®] 1.1.3
- Multiple concurrent DOS 5.0 sessions
- EZ Utilities for enhanced systems management tools to support IBM Business Partners and customers with large installations
- Mini-cartridge tape for ease of installation
- Full hardware support and exploitation for models of IBM PS/2 systems based on 32-bit IBM and Intel technology using both Micro Channel and IBM AT[®]-bus architectures

AIX is an excellent choice for users who want an open systems solution and the ability to run an impressive array of engineering/scientific and commercial applications.

Scalable Networking Solutions

The 1990s will likely be known as the decade of client/server computing. The demand for network access and information sharing across organizations of all sizes is growing at a phenomenal rate. Therefore, IBM PSP has developed a comprehensive and scalable LAN product line that enables users to realize the advantages of client/server computing today (see Figure 2). In addition to the LAN Server, IBM offers NetWare[®] from IBM, a wide range of LAN administration tools including electronic mail and workgroup computing solutions, and powerful productivity tools, such as communications and database managers. These solutions enable users to focus on running their businesses instead of running their systems.

OS/2 LAN Server 3.0



OS/2 LAN Server 3.0 is the fastest IBM PC-compatible network server on the market today, as verified by LANQuest

Labs[®] in October 1992. It supports DOS, DOS/Windows, Macintosh[®], and OS/2, as well as Transmission Control Protocol/Internet Protocol (TCP/IP) connections.

Both entry and advanced levels are available to address the needs of small companies and departments as well as large international enterprises. The entry level provides an economical solution for both 16-bit and 32-bit systems. The advanced level adds features such as the High-Performance File System (HPFS) and additional error recovery, giving it industrial-strength availability. OS/2 LAN Server is supported on a range of personal computer types as well as PS/2s. LAN Server 3.0 makes life easier for users, LAN administrators, and systems managers, as follows.

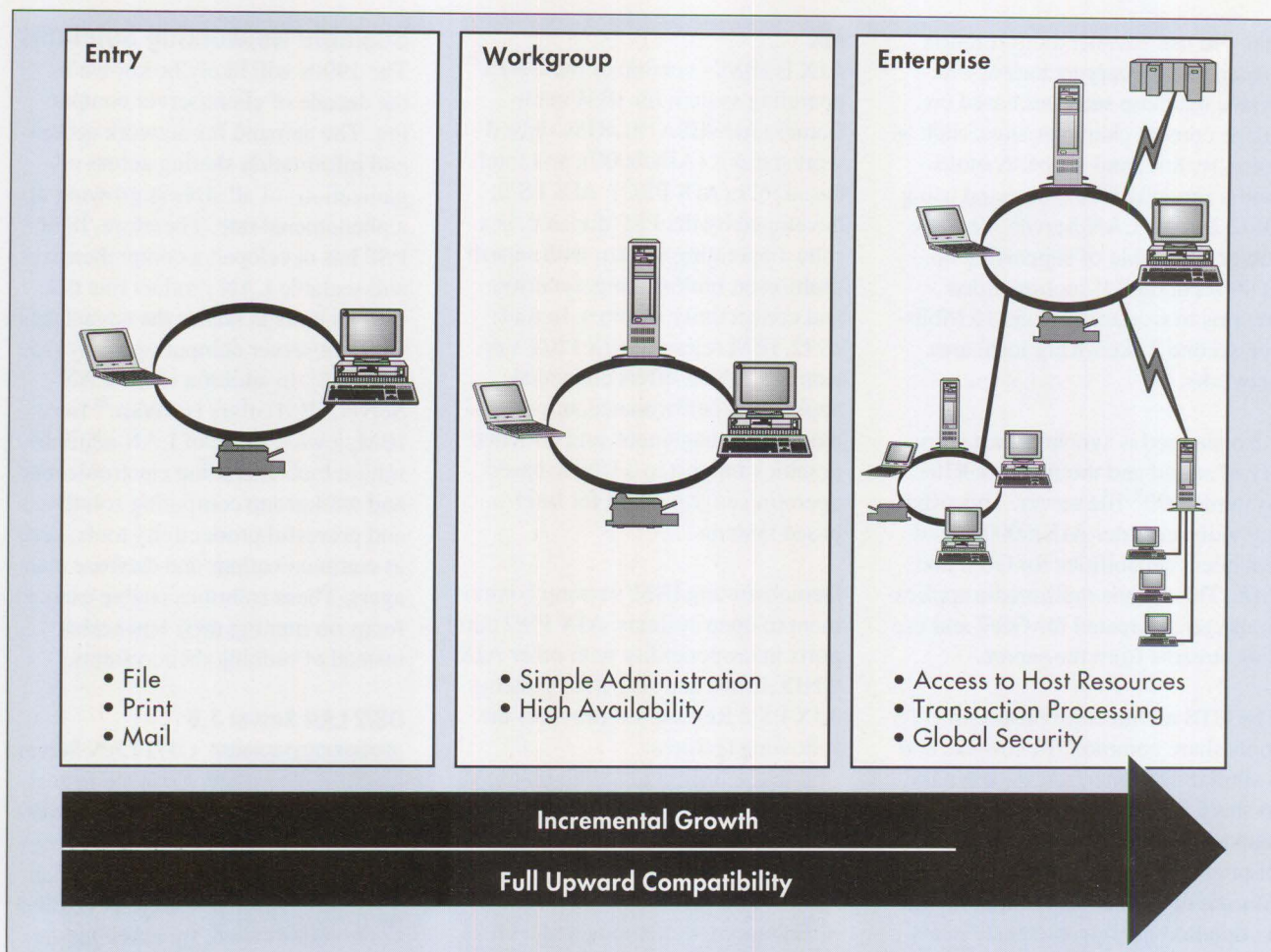


Figure 2. IBM's LAN Systems Family

Users: To provide improved productivity and usability, the OS/2 LAN Server running OS/2 provides a single view of all available network resources and automatically allocates these resources. Combined with the Workplace Shell, these once complicated technologies become easy and intuitive.

LAN administrators: Managing large LANs from a central location is essential in the ever increasing complexity of business computing. LAN Server 3.0 provides an architecture that enables location-independent resource management, making system-wide changes easier to implement and control.

Systems managers: Identifying bottlenecks and fixing them, resolving errors quickly, predicting capacity requirements, distributing application and system code, and providing fixes are important tasks for the long-term success of LAN solutions. These facilities, either LAN-based or integrated with IBM's host-based NetView®, provide a powerful set of facilities for LAN management.

Just as we are enhancing OS/2 for portable computer users, we also will be enhancing LAN Server 3.0 to support portable computer users. We plan to add remote client support to all LAN services. Then portable clients will be able to dial into their LAN Server and have full access to

data and facilities on the LAN. Remote users are supported as if they are locally attached to a LAN. They can communicate with other clients on the LAN Server and use other specialized servers, such as communications gateways or Lotus Notes® servers.

To provide application transparency, IBM is developing and supporting programming interfaces (including the industry-standard Sockets and X/Open's™ transport interface) as well as protocol bindings that facilitate communication across multiple protocol types from a single network interface card. By implementing these interfaces, users and their existing applications operate consistently across multiple network environ-

ments. For example, programs written to the Sockets interface will operate transparently across TCP/IP, NetBIOS, IPX, and Systems Network Architecture (SNA) networks.

As simple file and print sharing on LANs gives way to distributed computing and specialized servers, our LAN products will continue to treat the entire LAN as one system. Central to distributed LAN systems are common services that support application interoperability and workgroup application development. IBM and others, such as Hewlett-Packard® and Digital Equipment Corporation, have selected the Open Software Foundation's® (OSF's®) Distributed Computing Environment (DCE) as the fundamental technology to provide these services. DCE's facilities incorporate key functions such as Remote Procedure Call (RPC), network time management, security, and distributed directory services. When implemented, a single application can be distributed across multiple systems composed of both IBM and non-IBM hardware.

Since these new applications will be used in systems where information is passed to users in real time, IBM offers transaction management services—Encina® from Transarc® Corporation and Customer Information Control System (CICS). These services will allow distributed application transactions across a network of heterogeneous systems, providing data integrity if a system or network fails at any point during a transaction.

We plan to extend the benefits of object technology to the LAN environment, bringing the productivity benefits of objects to distributed applications. Video servers will bring the technology of multimedia to the LAN.

We plan to introduce these enhancements as a family of LAN systems

products. Applications will run across the family, and users can add and upgrade family members with no disruption to their clients or applications.

NetWare from IBM

Today, IBM PSP markets two network offerings: OS/2 LAN Server running with OS/2 and NetWare from IBM. Although there are many similarities between the two products, each is best suited for different situations. For example, many users want resource sharing in a distributed application environment. Since both clients and OS/2 LAN Servers run on OS/2, they share a common set of APIs, making development of distributed applications significantly easier. If a user needs simple resource sharing and does not plan to implement distributed applications, either product is an excellent solution. Sometimes specific application requirements will dictate the selection.

Both products play a significant role in IBM's distributed LAN systems strategy, but they remain separate product lines with unique system services, APIs, and communications transports. Users should choose their network operating system after carefully considering their future requirements. Whatever product is chosen, users of NetWare from IBM and OS/2 LAN Server benefit from IBM service and support.

OS/2 Interoperability with UNIX

IBM's strategy is to make OS/2 and UNIX interoperate as widely as possible. TCP/IP for OS/2 provides interoperability with UNIX-based systems (including AIX) that support open standards such as TCP/IP, Network File System (NFS®), File Transfer Protocol (FTP), TELNET, and X-Windows.

Lotus Notes and cc:Mail

Major elements of any networked personal computer environment must include electronic mail and work-

group computing capabilities—turning LAN systems from print and file serving into true client/server systems. As the result of a strong partnership with Lotus®, an integral part of IBM's LAN System and Office-Vision™ product offerings are Lotus cc:Mail and Lotus Notes.

Lotus cc:Mail provides high-end electronic mail capabilities. The cc:Mail clients include DOS, Windows, Macintosh, OS/2 Workplace Shell, and UNIX, plus gateways to PROFS®, OS/400®, DEC®, NetWare's MHS, and many more.

With the vast number of gateways available for cc:Mail, LAN-based users have one of the most powerful user interfaces available, while supporting the exchange of notes and files between users on the LAN and those on other host E-mail systems. This makes cc:Mail the most comprehensive electronic mail and messaging system in the industry.

Lotus is a pioneer in the development of the Vendor Independent Messaging (VIM) standard. This standard, endorsed and supported by Apple®, Borland™, IBM, Novell®, and WordPerfect®, enables mail to be integrated with other software vendors' applications as well as customer-written systems. For example, this capability, supported by cc:Mail, allows reports to be generated automatically and status notes to be sent to interested parties directly from applications.

For users needing the power of a groupware solution, Lotus Notes is the industry-leading workgroup tool. With Lotus Notes—a true client/server application—everyone in a workgroup, whether across the hall or across the continent, has access to all the information needed to make informed decisions, whether it is text, graphics, or scanned images. Lotus Notes also contains a powerful appli-

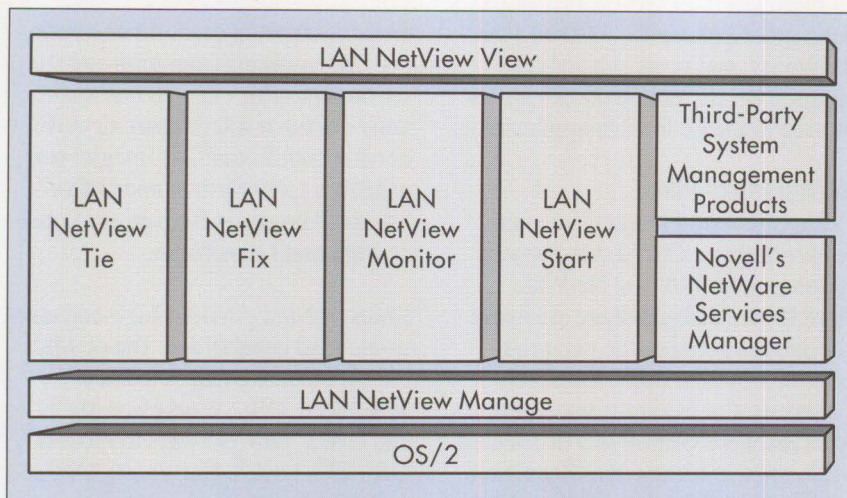


Figure 3. IBM's LAN NetView Family of Products

cation development platform that can be used to create custom systems that specifically serve individual business needs.

LAN Administration Tools

Making a network as easy to use as your desktop requires a new generation of LAN administration tools. Installation, maintenance, performance monitoring, and problem solving are just a few issues that grow exponentially as the size of the LAN grows. IBM PSP treats the LAN as a single system to ensure the reliability, manageability, capability, and security of traditional host systems.

Configuration, Installation, and Distribution (CID) and the LAN NetView product family are tools and processes that simplify configuration, installation, and distribution tasks and management of LAN systems. CID goes beyond current software distribution techniques; cloning multiple sets of shared software is a solution only when every workstation and end-user requirement is identical. CID allows each workstation to be unique. Distribution can be managed from a decentralized LAN, a centralized host, or in combination in a large network. The benefit is dramati-

cally reduced end-user involvement with software installation, version control, and support.

OS/2 and LAN Server are CID-enabled. More than 160 companies, including WordPerfect, Lotus, and Novell, have indicated that they plan to CID-enable many of their products.

LAN NetView Family

The LAN NetView family, shown in Figure 3, makes using and managing a network easier and more transparent for LAN administrators—from monitoring the performance of the system to automating problem determination. A description of each component follows.

LAN NetView View: Provides a graphical interface for managing the LAN based on the X/Open Management Protocol (XMP) architecture. The View program is planned for availability later in 1993.

LAN NetView Start: Enables administrators to configure and validate different types of connections, as well as to automate and simplify tracking and configure workstation software. For example, this simplifies the process of adding a new department

or reconfiguring a LAN to improve performance, because it supports customized, unattended installation for OS/2, LAN Server, Extended Services, and other CID-enabled applications. The Start program is available today.

LAN NetView Monitor: Evaluates the entire system's performance, enabling administrators to identify problems and to analyze performance trends across a LAN. For example, it enables administrators to anticipate the need for a new server and alerts them when critical resources are not operating efficiently. This provides important capacity planning data and performance tuning information. This product evolved from the current System Performance Monitor/2 product. LAN NetView Monitor is planned for availability later in 1993.

LAN NetView Fix: Enables automated problem determination. It receives both hardware and software fault notifications across the LAN, and can help determine the probable cause of a problem and recommend actions. LAN administrators can isolate and resolve problems faster, thereby increasing network availability. The Fix program is planned for availability later in 1993.

LAN NetView Tie: Allows NetView administrators to manage workstations from a host—an important option for users who want to administer their LANs from a central location. The Tie program is planned for availability later in 1993.

LAN NetView Manage: Provides systems management services, such as inventory queues, configuration management, and industry-standard protocols for OS/2, DOS, Windows, and NetWare. LAN NetView Manage is planned for availability later in 1993.

LAN Network Manager: Manages LAN media and will become a LAN NetView application. This tool can identify media-related problems, such as a sliced cable, and application problems by looking at the data stream being sent across the LAN. LAN Network Manager is available today.

Extended Services

Extended Services for OS/2 is separate from OS/2 and provides communications and relational database functions. Extended Services includes many connectivity support and connection protocols, all of which can be active simultaneously. Whether information resides on a 3090™, an AS/400®, a DEC system, or a remote NetWare server, Extended Services provides the needed connectivity.

As relational database applications are moved from a host to LANs or desktops, investments in application development, education, and expertise must be protected. The powerful client/server Database Manager—part of IBM's family of relational databases that includes DB2® and SQL/DS™—capitalizes on experience with host relational databases. Extended Services works with both the 16-bit OS/2 1.3 and OS/2 2.0—an advantage in mixed 286 and 386 environments.

Extended Services is a key component of OS/2's family for communications protocols (such as Advanced Program-to-Program Communication or APPC, 3270, 5250, Async/ASCII, and Advanced Peer-to-Peer Networking or APPN) and relational database access (SQL and DRDA™). These are key building blocks for future cooperative processing applications, whether they are developed by third-party applications vendors or by MIS departments.

Communications Manager/2

For users who want to get more from their desktop PCs, IBM has announced an enhanced product designed to bring the application and communication resources of large networks to the end user's fingertips. Communications Manager/2 enables the desktop to access large networks, allowing them to share in the richest set of networking services in the industry today.

This can increase productivity of end-user professionals who must access corporate applications, whether in finance, banking, insurance, or any industry where users in large and small networks need to share information.

Communications Manager/2, now packaged separately from OS/2 Extended Services, offers new functions, enhanced ease-of-use, and improved cost-effectiveness. Some new capabilities for Communications Manager/2 and Communications Manager Client Server/2 include the following:

- The client/server option allows customers to concentrate all Communications Manager functions in a server. For end users to share distributed applications across many parts of the network, individual packaging is ideal. For end users with less intense usage, such as those primarily emulating host applications, the client/server packaging will save both money and memory.
- Support for the Integrated Services Digital Network (ISDN) basic rate interface enables customers to save money by using digital, switched telephone lines over short or long distances. It also provides higher bandwidth options for high-productivity and distributed applications supporting such features as image and high-resolution graphics.

- Multiple servers can be used with multiple clients, providing automatic load balancing as information is given and received. This helps to maximize the efficiency of the network and ensure a fast end-user response time.
- Installation and configuration of Communications Manager/2 can be done quickly and easily. Communications Manager/2 also supports CID capability for remote installation, with no end-user involvement.
- Keyboard customization and color remapping is easily done for both 3270 and 5250 emulation environments.
- For network administrators, Communications Manager/2 offers much new power. This includes better alert routing to NetView, IBM's flagship network management product, and better alert support for APPN distributed environments. First Failure Support Technology (FFST) helps LAN administrators isolate and identify networking problems more effectively.
- Communications Manager/2 provides broader access to applications and other resources in many more hosts. Now a single desktop user can access up to three hosts concurrently and utilize as many as 26 different host sessions, 16 of which can be active at any one time.

Database2 OS/2

Database2 OS/2 (DB2/2) is a powerful 32-bit OS/2 2.0-based relational database management system for users in a client/server LAN environment who want to take full advantage of IBM's industry-leading database technology.

DB2/2 runs on a stand-alone PC or functions as a database server on a

LAN for PCs running OS/2, DOS, and DOS/Windows.

DB2/2 is the 32-bit follow-on product to the database management components of the 16-bit version of OS/2 Extended Services 1.0 Database Manager and OS/2 Extended Edition 1.3. It will support applications written for those environments.

DB2/2 exploits OS/2's 32-bit operating system and its flat memory model for improved database performance. It also provides enhanced DB2 compatibility, remote systems administration, and several other features designed to improve reliability and systems management. In addition, DB2/2 is an open database platform supporting industry standards.

Distributed Database Connection Services/2

IBM has developed the Distributed Database Connection Services/2 (DDCS/2) to provide access to data

anywhere. As a complementary product to Extended Services, it offers host database connectivity for OS/2 clients. Working with Database Manager, DDCS/2 allows DOS, Windows, and OS/2 clients to access host databases conforming to the DRDA. This includes not only IBM's DB2, SQL/DS, and OS/400, but several third-party database products, such as ORACLE® and INFORMIX®.

Requirements for Future Directions

Today's users are making trade-offs when determining the appropriate operating system and platform to develop and run the software that solves their business problems. DOS and OS/2 are Intel-based, while UNIX is often RISC-based. Advanced Intel-based 32-bit applications are available on OS/2. Even if they could run on the same platform, user interfaces are so dissimilar that end-user education and productivity, as well as end-

user support, can be adversely impacted.

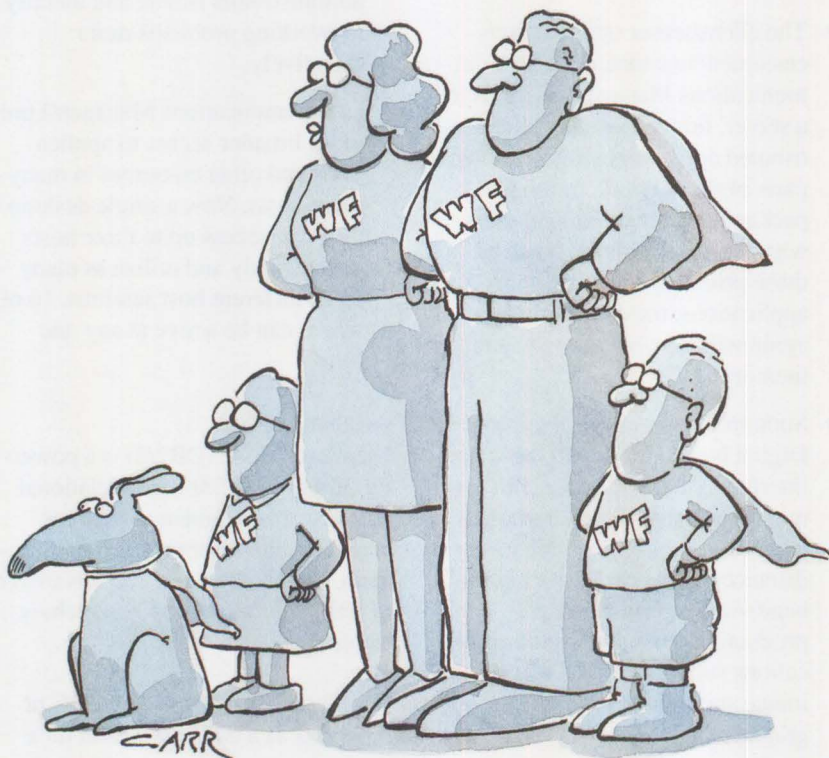
Today, users are compromising with an inferior application, or they "swallow" the expense of multiple systems on the same desktop. Could there be a way to run existing UNIX applications on Intel processors, and existing DOS, DOS/Windows, and OS/2 applications on RISC processors? Currently, a vast amount of processing power on desktops is mostly unused. Could we start using those idle MIPS? Is there a way to ensure that applications written today will still be of value as technology changes? Additionally, can we build and maintain a library of code (objects) that could be used and reused by OS/2, UNIX, and application developers? Could it be possible to make DOS, OS/2, and UNIX look and feel the same, using the most advanced user interface in the industry?

Introducing the Workplace Family

To satisfy these requirements and more, IBM is introducing the Workplace family. As shown in Figure 4, the Workplace family includes some current offerings and a new platform, the Workplace Operating System (OS). We are designing the Workplace family to do the following:

- **Improve usability** for users, LAN administrators, and application developers
- **Help protect customer investment** in hardware and software
- **Address business requirements** ranging from stand-alone users to connected enterprises
- **Increase productivity** for users, LAN administrators, and application developers

We plan to continue enhancing OS/2 and AIX, building on their unique strengths and attributes while taking advantage of similar architectural



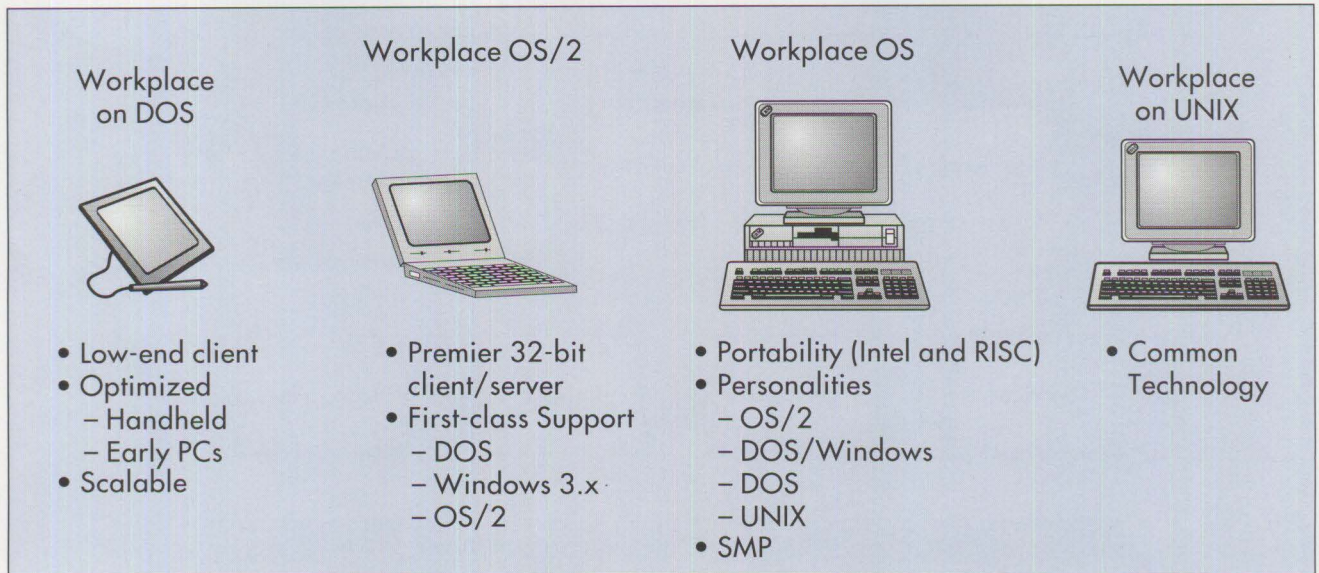


Figure 4. IBM's Workplace Family

components. The architectural components that members of the Workplace family will share are the Workplace Shell, objects, and distributed computing. While using this technology across product lines, we plan to make it available to other software and hardware companies.

Workplace Shell

Improved usability is key to personal computing in the 1990s. Based on user feedback, IBM will be using the Workplace Shell across its product line. We are working toward developing a powerful, easy-to-use interface for DOS that complements the new generation of portables—Workplace on DOS. This will improve the usability of DOS by eliminating complex syntax and command strings. It will provide a GUI consistent with OS/2, simplifying the use of DOS and migration to OS/2. Workplace OS/2 will continue to be enhanced to make it an even more powerful, yet easy-to-use user interface. The Workplace Shell GUI is also being developed for the UNIX environment.

Providing users with similar interfaces across operating system platforms can achieve the benefits of

reduced education costs, lower support costs, and higher productivity. A common user interface across DOS, OS/2, Workplace OS, and UNIX is the first step in resolving the platform dilemma facing users today.

Workplace OS

The Workplace OS platform has been designed to complement and extend the current OS/2, DOS, and AIX operating system family. Workplace OS is a flexible, modular platform that allows “pluggable” support for device drivers, application environments, and operating system services (from advanced file systems and communications systems to multimedia, pen, and speech systems).

With the Workplace OS platform, users will be able to take advantage of multiple operating system environments called *personalities* on a single desktop. A personality takes on the characteristics and support of an operating system platform, such as OS/2 or DOS. Personalities can include DOS, Windows, OS/2, and UNIX. The UNIX personality is being designed to support different UNIX application environments. Initially, the UNIX personality is planned to

be based on OSF/1[®] 1.2 Application Environment Specification (AES) and will support existing AIX PS/2 Version 1.3 applications. Other UNIX environments under consideration are AIX for the RISC System/6000 and SVR4.x.

The Workplace OS platform will take advantage of the common technology (Workplace Shell, objects, application frameworks, and distributed computing) that will be utilized across the IBM PSP family.

The Workplace OS platform will use microkernel technology. Microkernel technology supports multiple hardware platforms, such as Intel and RISC. It also will exploit a variety of multiprocessor-based computer technologies, such as symmetric multiprocessing, parallel processing, and loosely coupled clusters. Figure 5 shows the Workplace OS architecture.

Hardware Independence with the IBM Microkernel: Today's applications are usually written for specific operating systems. This has created a problem for both MIS and developers. They must select the platforms and then write the application for

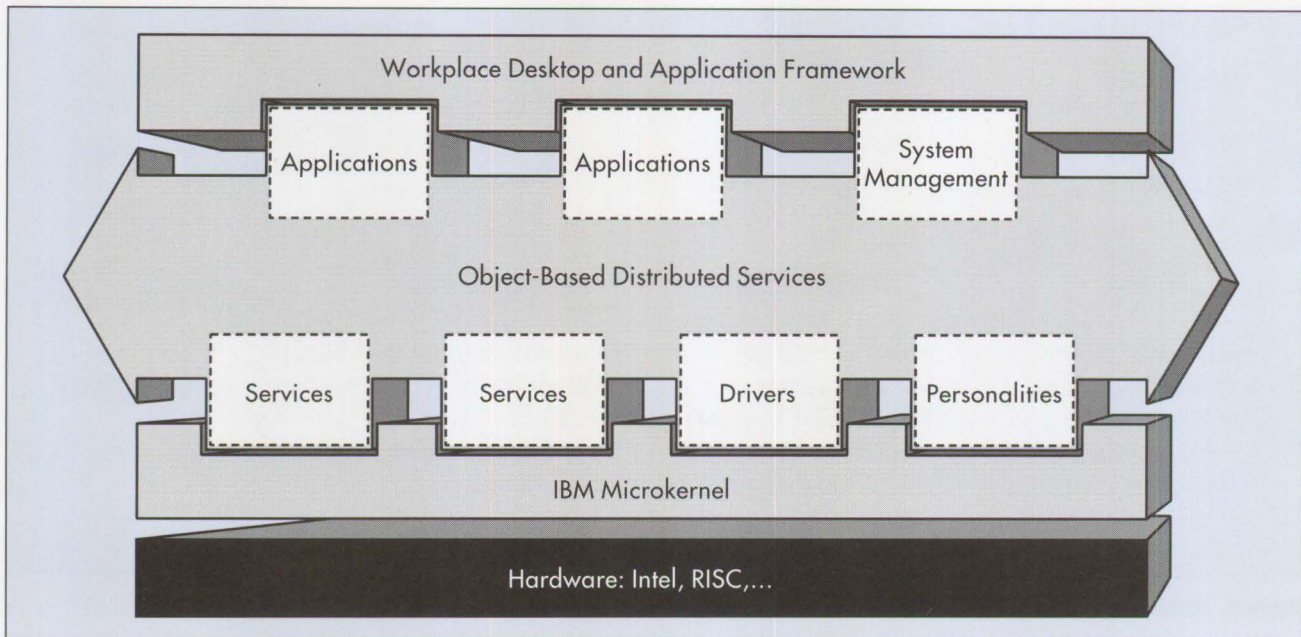


Figure 5. Workplace OS Architecture

each system's native operating system. To resolve the problem of developing multiple versions of the same application, IBM is developing the IBM microkernel with operating system personalities.

IBM's approach takes a single microkernel and makes it run on many processor types—whether Intel or RISC—such as IBM's PowerPC™ RISC processor under development by IBM, Apple, and Motorola™. However, instead of developing an IBM proprietary microkernel technology, we chose to work from an industry-standard kernel. Our microkernel architecture is based on Carnegie-Mellon® University's Mach 3 microkernel. Developing from an industry-standard kernel allows IBM to offer this kernel to other hardware and software companies in the industry, underscoring IBM's strong commitment to open systems and industry standards.

IBM has extended the Mach microkernel to provide additional capabilities and enhanced performance. Among the enhancements is the ability to have user-level device drivers

which run outside of the microkernel, reducing its size and allowing dynamic loading/unloading of device drivers. Also, the Interprocess Communication (IPC) mechanisms have been enhanced and performance greatly improved. Real-time support has been added and the thread scheduling algorithms in the microkernel have been improved.

To maximize performance, the microkernel supports threads that can execute simultaneously across multiple microprocessors, providing scalable system performance.

Support for Multiple Processors: Microkernel symmetrical multiprocessing support enables applications to run different threads of execution simultaneously on different microprocessors within the same system. Users can scale the performance of the system as application and system workload demands increase. When the symmetrical multiprocessing microkernel is used in combination with our distributed computing environment, a system's workload can be allocated efficiently

across multiple systems in a single network, delivering substantially better performance.

Support for Wide Range of Applications: To ensure existing and new applications will continue to run, we are building personalities that work with the microkernel. The microkernel contains the microprocessor-specific code (containing only the code that controls fundamental CPU access and processes), but the personalities provide the application environment. Therefore, DOS, Windows, OS/2, and UNIX applications run unmodified within their specific personality on the microkernel. The result is an optimized microkernel and personality that can accomplish more, and with greater flexibility and reliability than a monolithic operating system.

Object-Oriented Technology

IBM PSP is developing object-oriented technologies with a distributed system and applications focus, pursuing a strategy to significantly improve the next generation of computing. Our focus is on creating an

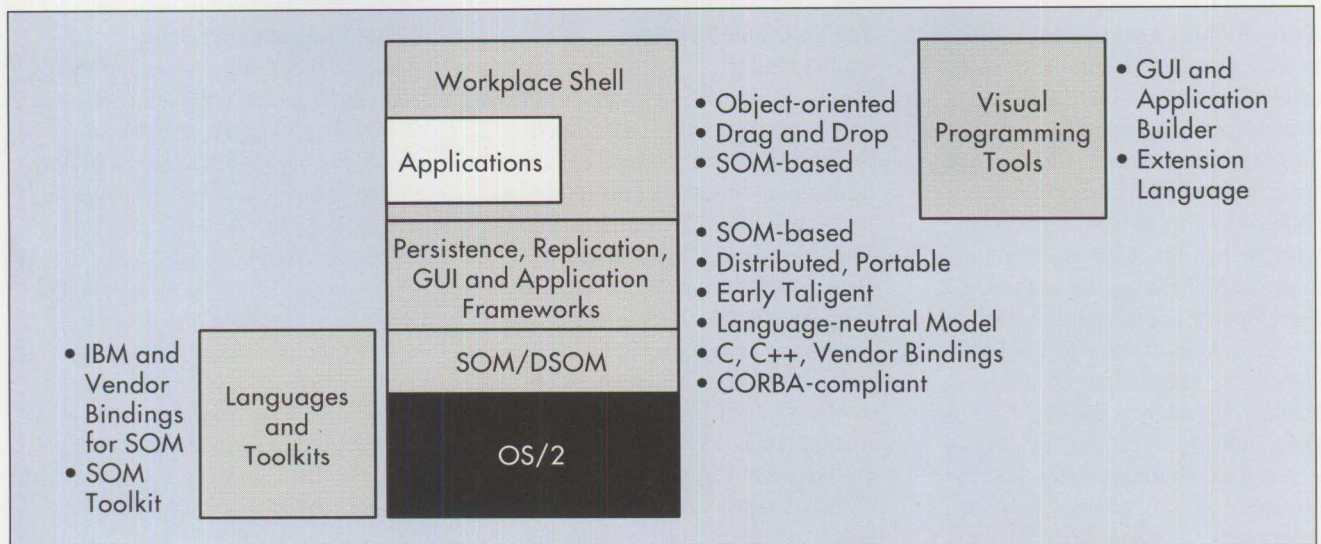


Figure 6. Object Enabling and Visual Programming Tools

open environment for developing distributed applications.

An End User's View

From the user's perspective, object-oriented technology will make using personal computers more intuitive and easier to use. For example, in OS/2's Workplace Shell, a user can simply "drag and drop" the picture of a file onto a picture of a printer to get a document printed. The user does not have to remember complex operating system instructions. Object-oriented technology will bring several benefits to users:

- **Increased ease of use and productivity** that includes more intuitive applications and consistent interaction among applications
- **Greater application availability** since applications can be developed faster because of reusable code
- **Higher quality application reliability** due to reusable code

An Application Developer's View

Object-oriented programming has established itself as an important methodology in developing high quality, reusable code. Operating systems

and tools vendors are beginning to offer class libraries and frameworks. Visual programming tools are emerging to assist end users with software application assembly.

Our strategy is to develop products that provide a basis of developing object-oriented distributed applications and enable domain experts and end users to create distributed line-of-business applications.

The paths to achieve this strategy are the development of system-level enabling technologies and frameworks, and creating application development shells and protocols. At the system level, efforts are focused on the SOM. At the applications development level, a set of application frameworks and visual programming tools are under development, as shown in Figure 6.

The System Object Model

In OS/2 2.0, IBM introduced a new system for developing class libraries and object-oriented programs. This system is called System Object Model. SOM is a technology for packaging object-oriented class libraries. It provides a language-neutral model

for defining libraries of objects that operate across many computer languages. Because a single version of an application will operate on many computer systems, software developers should be able to spend their time building new applications. They should no longer waste time building multiple versions of each application for every supported computing platform. Class libraries built with SOM can be defined and implemented in one language and be usable in another language. Today's SOM supports the C language. Additional language support, including C++ and COBOL, will become available in 1993 and 1994.

Because this approach allows developers to build applications by tailoring and linking pretested objects from different developers' libraries, two fundamental changes can occur in software development. First, software developers become more productive; they can create sophisticated applications in a fraction of the time of traditional programming methods. Second, the work invested in creating an object has multiple dividends since the object is reused in future applications.

IBM's SOM is a clean-sheet approach to unlocking the benefits of the object-oriented systems model. It is not dependent on any specific computer language, architecture, CPU, or operating system. SOM is currently shipped as part of OS/2 (runtimes with the operating system, and bindings and SOM compiler in the OS/2 Developer's Toolkit); it has been a funded development effort at IBM since early 1989. Since then it has been reviewed by some of the computer industry's leading developers of programming languages, and by commercial and corporate software developers. These reviewers anticipate SOM's role as a universal translator of applications and objects developed in diverse programming languages.

Through the SOM language's Object Interface Definitions and Object Interface Compiler, applications written in other programming languages can operate within SOM. SOM is designed to support many common platforms including OS/2, AIX, Windows, DOS, MVS, CICS, AS/400, other versions of UNIX, and future operating systems derived from our relationships with third-party operating systems providers. Support for OS/2 is available today.

SOM II

SOM will be extended to support multiple inheritance and will use the Object Management Group Common Object Request Broker Architecture (OMG CORBA) interface definition language and runtime APIs. Support for additional language bindings and/or native support will appear in compiler products from IBM and other vendors. SOM will be exploited as the foundation for other frameworks and tools that will be developed, such as object storage and distribution, and visual programming.

The Distributed System Object Model

Today, SOM provides non-distributed local object support. Distributed SOM (DSOM) is planned to support transparent remote access to objects in a distributed environment. DSOM functions as an Object Request Broker (ORB), similar to RPC. It supports the full object-oriented programming paradigm via distributed computing facilities like Sockets and the Distributed Computing Environment (DCE). DSOM will allow application portability through OMG's CORBA-compliant ORB. DSOM will extend all the advantages of SOM to a full range of distributed environments.

Support for homogeneous environments (OS/2 and AIX) will come first providing intra-machine (multiple processes, single machine) and inter-machine support. Intra-machine support is useful when objects are implemented by programs (versus libraries) which cannot be linked directly into an application or when objects must be in a different address space than the application, for example, for integrity. Support for heterogeneous environments will follow that exploit available distributed computing services such as DCE.

Frameworks

Frameworks will simplify software development by offering built-in functionality that can be easily extended. Frameworks are a collection of objects with established relationships that serve as a foundation for specific implementations. We will offer system- and application-level frameworks to include persistence and replication for saving and sharing objects, as well as system and application frameworks to accelerate development productivity. These will include a subset of Taligent's frameworks and services.

Visual Programming Tools

We are developing a visual application development environment, a set of initial application objects, an object-oriented application framework, and an extension language. These will include an extensible palette of objects as components for building applications. These applications will be built by dragging and dropping objects into an application layout, selecting properties for the objects through a dialog box, and then visually establishing connections between objects. Through these actions, actual code is generated and compiled.

The visual programming tools will include GUI objects that are a superset of IBM's Common User Access™ (CUA™ '91) and platform-specific controls. Over time, the application objects will include text objects, 2-D and 3-D chart objects, forms objects, table objects, and links to spreadsheets and databases.

Taligent

Taligent is an independent joint-venture established by IBM and Apple. A key objective in forming Taligent was to bring the benefits of object-oriented technology to customers more quickly. Taligent's native environment will coexist with IBM's current products. Integrating Taligent-derived technology into IBM's Workplace family will provide a stable path to Taligent's new generation of systems.

Taligent will play an important role in optimizing the 32-bit object environment by introducing revolutionary object technologies. As shown in Figure 7, IBM PSP will integrate technology from Taligent in future versions of IBM's products. Eventually, Taligent will introduce an object-oriented environment, re-engineered from the ground up, and will build compatibility around it for today's 32-bit OS/2 applications.

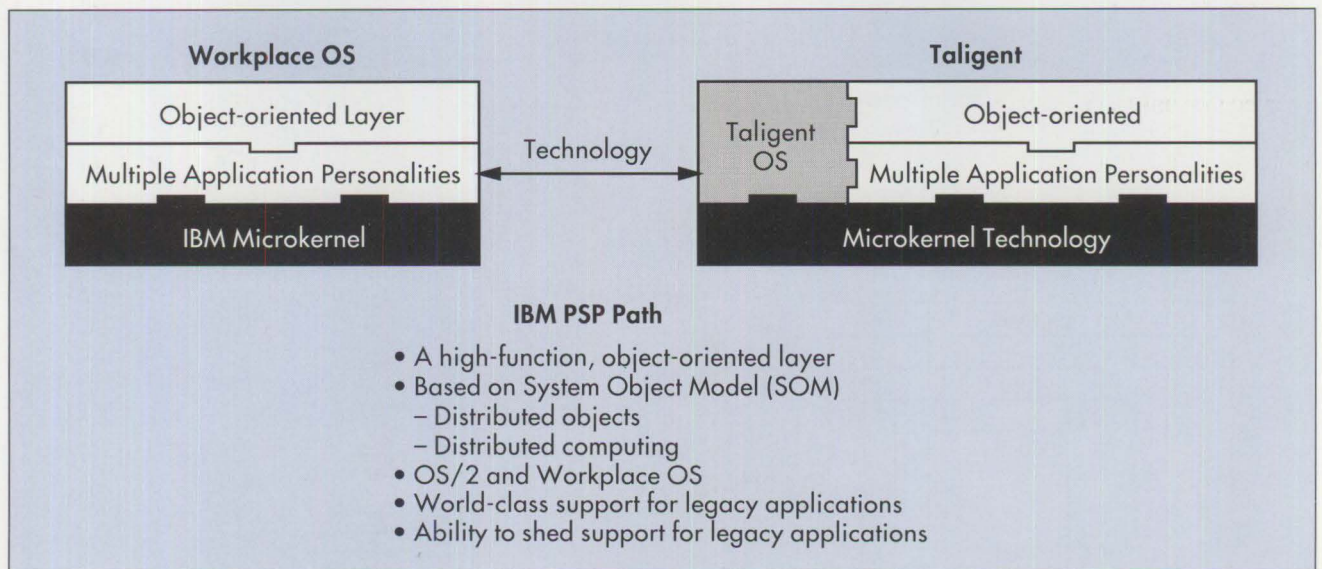


Figure 7. IBM's Object-Oriented Direction

The Future of Distributed Computing

IBM plans to provide a distributed system infrastructure that can be used to access resources from any desktop or application, whether it is running on a client, a server, or a mainframe. This includes client operating systems such as DOS, DOS/Windows, Macintosh, AIX, OS/2, and Taligent. Our goal is to help protect customers' existing investments in hardware, software, applications, and data, while enabling the new capabilities offered by network systems.

Today's information needs have stretched far beyond dedicated personal computers acting within their islands of information. Simple interconnectivity must be pushed forward so that not only can any personal computer connect to any system, but applications on each connected computer have a way to jointly process information. We are advancing these connectivity services on the client, server, and host systems to provide the next logical progression of networked systems: distributed computing. Figure 8 shows how we are simplifying distributed computing.

IBM is approaching this by using open industry definitions for these application and operating system interfaces. For example, DCE is being driven by the Open Software Foundation, an organization in which IBM and other major computer vendors have been active for many years. OSF's goal is to enable users to implement computing environments where systems and applications work together, locally and globally, regardless of what vendor hardware and software is chosen.

From the variety of architectures available to OSF members, we have selected the industry standard DCE. We have committed to incorporating DCE into OS/2, OS/400, AIX, MVS, and VM. Beyond IBM-supported operating systems, DCE is the common selection among other system suppliers. Through this industry-wide support of DCE, businesses can finally achieve the goal of distributed computing throughout an enterprise network. Our distributed SOM technology will support DCE.

IBM has begun to deliver these advanced functions to users. Distrib-

uted System Services (DSS), our core advanced network architecture, is as adept at supporting client/server computing on a LAN as it is at supporting distributed computing across an enterprise. DSS will allow applications and data to reside any place that an individual or an organization chooses.

Some components of DSS will come from IBM, such as DRDA—the way relational databases “talk” to each other. Even here the solution will not be IBM-unique, since Oracle, Informix, and several other vendors already have adopted or committed support for this database-to-database communication language. Other elements of DSS will include components created through industry consortia, such as the Vendor Independent Messaging standard developed by Lotus for E-mail.

Consistent technologies for key functions, such as remote program execution, network time, security, and global directory services, should enable true, cross-platform interoperability. For example, it will be possible to develop and run a single application across multiple systems,

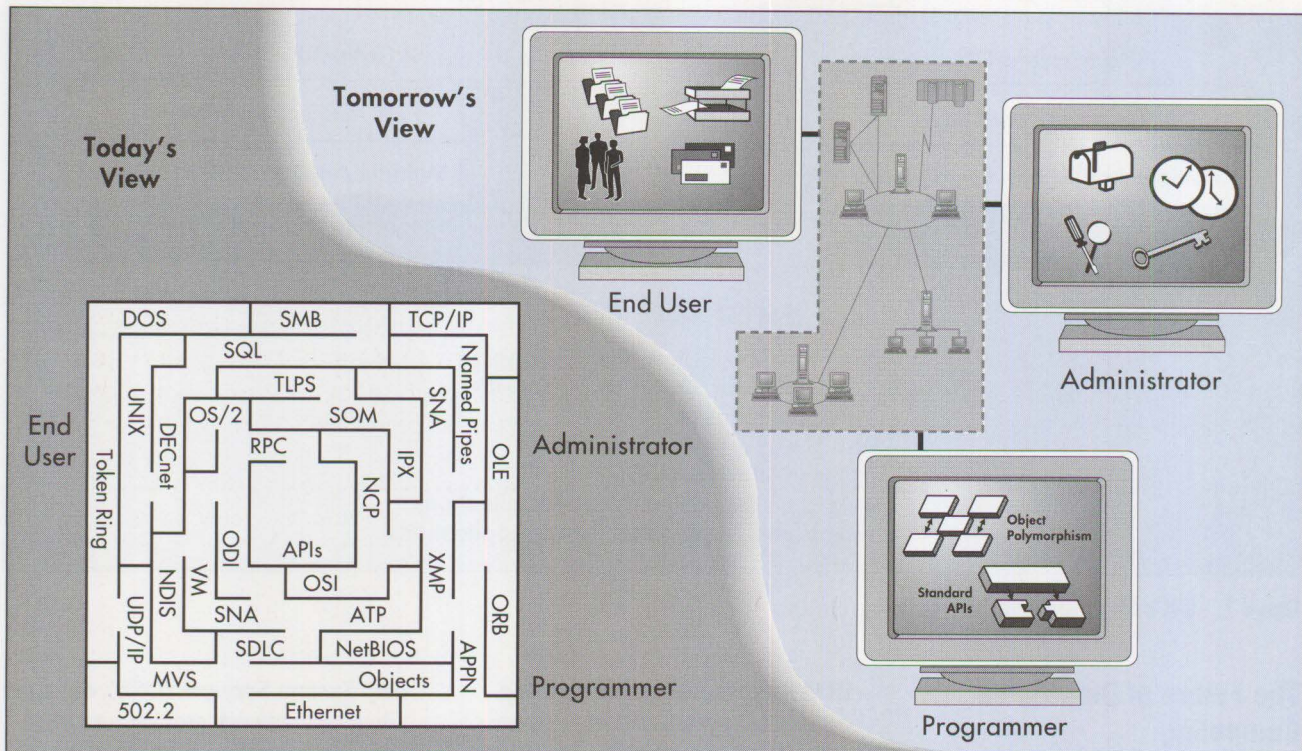


Figure 8. A Simpler View of the World with Distributed Computing

while still providing the ability to manage and support it with full problem-determination capabilities.

Getting Started Today

You can start with our offerings today, giving you the right products and support to move easily and safely into the future. DOS, OS/2, AIX, LAN Server, and NetWare from IBM are a base from which your system can evolve.

We have discussed how our advanced technologies will be integrated into our products—increasing the power and usability of today's products while developing our future product lines to be even more flexible, consistent, easy to use, networked, and open. This results in investment protection for users.

When considering ways to solve business problems, you face several key challenges. First, users have solutions in place, and many feel strongly about the solutions they have chosen. Therefore, we offer products today that support their individual choices. Second, our products offer the opportunity to improve the way you do business today. Manage it better, react quicker, and move information closer to the users.

Finally, we understand that it is critical to build solutions that are open, supporting standards that are prevalent in the industry from applications to networking protocols. The implementation, interconnection, modification, support, and growth of your systems can be simplified by choosing solutions built on open standards.

IBM PSP will continue its leading role in creating, driving, and implementing industry standards.

Acknowledgment

I wish to thank Neeraj Srivastava, Mac McCarter, D'Ann Ostrom, and Karl LaWall for their time and energy in helping to write this article. They are presently involved with developing market strategies for Personal Software Products.

DeeAnne Safford is a program manager in IBM's Personal Software Products Marketing Strategy Group. She is presently involved with developing the market strategies for Personal Software Products. She holds an MBA from Nova University in Fort Lauderdale, Florida.

Subscription Form for /AIXtra: IBM's Magazine For AIX Professionals

The IBM RISC System/6000 with AIX platform is one of the fastest growing advanced UNIX workstations in the industry. You can find out why and get the inside story on IBM's AIX systems straight from the source with a subscription to /AIXtra: IBM's Magazine For AIX Professionals. Detailed technical articles are written by the experts — people who design, develop, and support AIX and related products. /AIXtra magazine covers AIX systems, software, networking, implementation, and more.

A single issue costs \$12.95; you can subscribe now and receive a one-year subscription for only \$55.00 (Canada/Mexico \$80; other countries \$100). Just complete this form and either fax it to (415) 948-4280 (please include VISA/Mastercard number and expiration date), or mail your check or money order to The TDA Group, Box 1360, Los Altos, CA 94023-1360, or call (415) 948-3140. All orders must be pre-paid. Checks must be in U.S. dollars drawn on a U.S. bank.

1. Do you currently have AIX installed?
Yes _____ No _____
How many years experience have you had with AIX or UNIX? _____
2. If not, do you plan to install AIX within the next 24 months?
Yes _____ No _____
3. Other operating systems in use:
 UNIX Sys. V
 SCO Unix/Xenix
 Ultrix
 A/UX
 HP/UX
 Sun/OS/Solaris
 PC/MS-DOS
 OS/2
 VM or MVS
 Windows
 Mac
4. The number of employees at your location:
 1-99 100-199 200-399
 400-999 1,000-1,999 Over 2,000
5. The number of employees company wide:
 1-99 100-199 200-399
 400-999 1,000-1,999 Over 2,000
6. Applications in use:
 CAD/CAE
 Education
 Science/Engineering/Research
 Graphics
 Point of sale
 CASE/Software Development
 Financial Services
 Data Base
 AI
 Numeric Intense Computing
 Manufacturing/CIM
 Office/Professional
 Communications/Networking
 Mapping (GIS)
 Other
7. Are you involved in evaluating, recommending, specifying, or approving any of the following?
(Check all that apply.)
 Workstations
 Terminals
 X stations
 Software development tools
 Storage/memory
 Printers
 Training
 Maintenance/service
 Business applications
 Graphic software
 I/O boards
 Modems
 Comm./Networking
 Servers
 DBMS
 Tape drives
 DOS to UNIX tools
 UPS
8. How much will your organization spend on computer related products/services in the next year?
 Over \$10 million
 \$1-\$10 million
 \$250,000-\$999,000
 \$100,000-\$249,000
 \$50,000-\$99,000
 \$10,000-\$49,000
 Under \$10,000
9. Your primary job title:
(Check the one best descriptor.)
 Corporate and Financial Management (Pres., Owner, CEO, VP, Marketing Dir., Gen. Mgr., Financial VP, CFO, Controller, Treasurer)
 Computer Systems Management (DP/MIS Dir., Network Dir., Communications Mgr., Systems Analyst, Software Developer, CIO, Systems Administrator)
 Engineering Management and Staff (VP Engineering, Chief Engr., Technical Director, Systems Integration Mgr.)
 Consultant and Educator (Computer/Network Consultant, Computer Technology Educator)
 Other (please specify) _____
10. Your company's primary business activity:
 Manufacturing (computer hardware)
 Manufacturing (non-computer products)
 Systems Integrator, VAR, OEM
 Software Development
 Financial: Banking, Insurance, Real Estate
 Retail, Wholesale, Distribution
 Utilities, Communications Services, Transportation
 Government and/or Military
 Computer Services, Data Processing, Service Bureau
 Health or Legal Services
 Education
 Consulting
 Agriculture, Mining, Construction, Petroleum, Forestry, Chemical
 Architecture/Engineering
 R&D, Testing, Evaluation Labs
 Other qualified business including Hotels, Publishing, Amusements, and Non-Profit Organizations
 Other (please specify) _____

IMPORTANT: PLEASE COMPLETE ALL INFORMATION REQUESTED.

(Please print or type.)

Name _____

Company _____

Address _____

City _____ State _____

Country _____ Zip Code _____

Business Telephone _____

Yes, please enter my paid subscription:

Signature _____

Charge to VISA Mastercard
Number _____

Expiration Date _____

Using REXX to Customize the Workplace Shell

Bret Curran
IBM Corporation
Roanoke, Texas

This is the first of two articles about using REXX to customize the Workplace Shell (WPS). This first article provides the foundation for building objects in the WPS, and shows how to incorporate REXX programs into a Configuration, Installation, and Distribution (CID) installation. The second article (to appear in July 1993) will describe not only how to create WPS objects, but also how to manipulate existing objects and the WPS in general. The documentation and sample programs work with OS/2 2.0 with the ServicePak applied.

Installing OS/2 2.0 takes 21 diskettes, plus an additional 14 diskettes just for the ServicePak. There must be a better way to manage OS/2 installations—and there is! Using IBM's Configuration, Installation, and Distribution (CID) to install and maintain OS/2 and its subsystems helps tremendously. What could be easier than inserting two diskettes to install OS/2 2.0, Extended Services, LAN Server, and other CID-enabled applications, and then leaving to do other work instead of doing the diskette shuffle for hours?

Although CID has greatly simplified the process of installing on multiple machines, it has shortcomings in certain areas. However, these can usually be handled with a little SMOP (simple matter of programming). For example, if a hundred computers needed to be installed at company C&C, this would typically require a person to walk to each station and swap diskettes. After that, the same person would likely customize each Desktop. This can be as simple as placing a program object on the Desktop that starts the company's "bread and butter" application. It also can be as complex as completely redesign-

ing the look and feel of the Workplace Shell (WPS).

While the CID process can save the C&C administrator from swapping diskettes, each machine still would have only the default Desktop, as shown in Figure 1. To finish the process, the administrator could sit at each of the hundred workstations customizing the Desktops individually, or create a REXX program that automatically performs the customization.

This article shows how easy it is to use REXX to customize the WPS as part of a CID installation. It covers the following topics:

- Using REXX and REXXUTIL
- Creating a folder object
- Creating a program object
- Creating a shadow object
- Creating a LAN CID Utility (LCU) command file that installs OS/2 2.0 and the ServicePak, and customizes the WPS
- Putting it all together to customize all one hundred Desktops to look like Figure 2

Using REXX and REXXUTIL

REXX is included with OS/2 2.0. Since it is an installable option, it may not be loaded on a system. Determine if it is installed by looking in the \OS2\DLL subdirectory. There will be several REXX*.DLL files if REXX is installed.

REXX is an extensive language, but the heart of working with the WPS lies in using the functions included with the REXXUTIL Dynamic Link Library (DLL). This article focuses on just one of the functions from that DLL. To learn more about REXX and the REXX Utility (REXXUTIL) functions, open the Information folder and then open the REXX Information object.

Of the many functions included with REXXUTIL, we will concentrate on the SysCreateObject function. A brief description of the call is included in the online REXX information; however, much more must be known before you can work effectively with the WPS. For further details, refer to *Using the REXXUTIL SysCreateObject Function* (the reference section at the end of this article explains how to obtain the document).

A REXX program is simply a text file created by any ASCII editor. The first line of any REXX program must start with a comment to indicate that it is a REXX program and not an OS/2 command file. Note that both OS/2 command files and REXX programs have the same file name extension of .CMD. Additionally, REXXUTIL.DLL must be loaded to use the SysCreateObject function. This is done by including the lines shown in Figure 3 in the program.

Now we are ready to program using the SysCreateObject function. Figures 4, 5, 6, and 7 show details about the many parameters and values of the call. You may want to

just skim through the figures; they will be more useful as references than as reading material. These figures are extracted directly from *Using the REXXUTIL SysCreateObject Function*.

To change the values in Figure 7, use SET keyname=____, as shown in the following example:

```
SET DOS_FILES=45;SET
DOS_HIGH=1
```

Some DOS settings use values of 1 for on, 0 for off (the default), as shown in the following example:

```
SET COM_HOLD=1;
```

To add more than one DOS_DEVICE statement, you need to separate them with hex 0A (line feed). Following is a REXX example:

```
h0A='0A*X
```

```
SET DOS_DEVICE=C:\OS2\MDOS\
ANSI.SYS'h0A'C:\OS2\MDOS\
EGA.SYS...'
```

Creating Workplace Shell Objects

After reviewing the figures, let's put the documentation to good use. Company C&C is getting ready to perform a CID installation of a hundred computers and wishes to customize the Desktop at the same time. They have asked us to write a REXX program that will create the following:

- C&C Company folder on the Desktop
- C&C application in the C&C Company folder
- Shadow of an OS/2 command prompt on the Desktop for easy access

As stepping stones, let's take each item and create an individual program to perform the operation. Afterward, we will see how to combine all

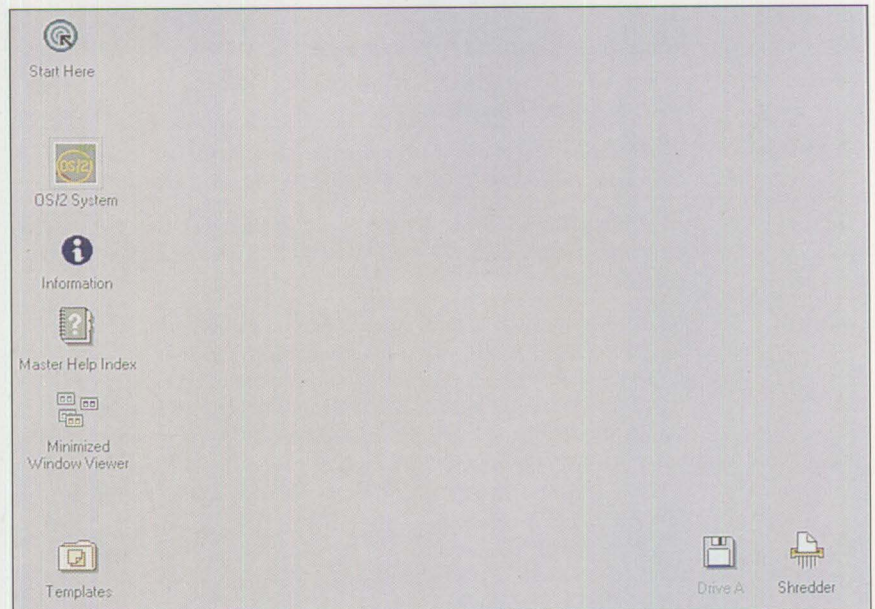


Figure 1. Default OS/2 Workplace Shell Desktop



Figure 2. Customized OS/2 Workplace Shell Desktop

```
/* REXX program to create ... */
/* Load REXXUTIL DLL */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs
```

Figure 3. Sample REXX Portion to Load the Functions of REXXUTIL.DLL

Parameter	Description
syntax	result = SysCreateObject(classname, title, location <,setup>,<,duplicateflag>)
classname	A registered object class defined to the system. Of particular interest are the WPFolder/WPProgram/WPSshadow classes.
title	The title you want to see for the object. If you wish to break the title line, use the caret “^” symbol in the title; for example, ‘First line^Second Line’.
location	The object location is most often specified by an object ID (such as <WP_DESKTOP>), but a file system path (such as C:\MYSTUFF) also can be used. You can look at the \OS2\SYSTEM\INI.RC file to see the predefined objects and object IDs. Here are some predefined object IDs of system folders: <WP_DESKTOP>: Desktop <WP_START>: Startup <WP_OS2SYS>: System <WP_TEMPS>: Templates <WP_CONFIG>: System Setup <WP_INFO>: Information <WP_DRIVES>: Drives <WP_NOWHERE>: Hidden
setup	This field needs the most explanation. The setup field contains a series of keyname=value pairs separated by semicolons that change the behavior of an object. Each object class documents its keynames and the parameters it expects to see. The WPFolder and WPProgram and classes are shown in Figures 5 and 6.
duplicateflag	There are three possible values for this parameter. Only the first character is examined: F, R, or U. FailIfExists: No object should be created if an object with the given object ID already exists. This is the default. ReplaceIfExists: If an object with the given object ID already exists, the existing object should be replaced. UpdateIfExists: If an object with the given object ID already exists, the existing object should be updated with the new information.

Figure 4. Parameters for the SysCreateObject Function

the programs into one, making it easier to run from the LCU command file.

Using REXX to create new objects is easy if you use a fill-in-the-blank approach. Whether creating a folder object, a program object, or one of many other objects, certain blanks must be filled in, and then it is just a matter of typing. The hard part is figuring out what needs to go in the blanks. This will become much clearer as we go through a few ex-

amples. Following are the blanks that must be filled in for the SysCreateObject function:

- classname
- title
- location
- setup
- duplicateflag

Creating a Folder Object

As requested by the C&C Company, we must create the C&C Company

folder on the Desktop. Let’s walk through this first example, remembering the blanks to fill in. If necessary, refer to Figure 4, which shows the parameters for the SysCreateObject function. Also, refer to Figure 5 for the values for the WPFolder class.

The first blank is `classname`. Since we are trying to create a folder, fill in this blank with `WPFolder`. The second blank is the `title`, which has already been given: `C&C Company^Folder` (the caret separates this onto two lines). The third blank, `location`, also has been given to us: they requested it to be on the Desktop. Therefore, put the object ID of the Desktop, `<WP_DESKTOP>`, into the location blank. An object ID is any string that begins with “<” and ends with “>”.

The fourth blank, `setup` (which is optional), is usually simple when creating a folder. Give the folder the object ID `<C&C_COMPANY_FOLDER>`. Note that the underscores could have been spaces, but underscores are usually used for consistency with the existing object IDs. The fifth and final blank, `duplicateflag` (also optional), is usually left at the default `FailIfExists`. If an object already exists with the identical object ID, the `SysCreateObject` call will not create the second object.

That’s all it takes to create a new folder on your Desktop. Figure 8 shows the entire program. Type it and then run it. If everything goes well, you should see the “Successful...” statement plus the new folder object on your Desktop.

Creating a Program Object

Now that the C&C Company folder is on the Desktop, the next step is to create the C&C application inside that folder. Again, refer to Figure 6, which shows the values for the `WPProgram` class.

Keyname	Value	Description
OPEN	ICON	Opens icon view when object is created.
	TREE	Opens tree view when object is created.
	DETAILS	Opens details view when object is created.
ICONVIEW	s1[,s2,...sn]	Sets icon view to specified style(s).
TREEVIEW	s1[,s2,...sn]	Sets tree view to specified style(s).
DETAILSVIEW (styles)	s1[,s2,...sn]	Sets details view to specified style(s).
	FLOWED	Flowed list items.
	NONFLOWED	Non-flowed list items.
	NONGRID	Non-gridded icon view.
	NORMAL	Normal size icons.
	MINI	Small icons.
	INVISIBLE	No icons.
	LINES	Lines in tree view.
	NOLINES	No lines in tree view.
BACKGROUND	filename	Sets the folder background; filename is the name of a file in the \OS2\BITMAP directory of the boot drive.
WORKAREA	YES	Makes the folder a work area folder.
MINWIN	HIDE	Hides views of this object when their minimize button is selected.
	VIEWER	Minimizes views of this object to the minimized window viewer when their minimize button is selected.
	DESKTOP	Minimizes views of this object to the Desktop when their minimize button is selected.
VIEWBUTTON	HIDE	Views of this object will have a hide button as opposed to a minimize button.
	MINIMIZE	Views of this object will have a minimize button as opposed to a hide button.
CCVIEW	YES	Creates new views of this object every time the user selects Open.
	NO	Open views of this object will resurface when the user selects Open.
ICONFILE	filename	Sets the object's icon.
ICONRESOURCE	id,module	Sets the object's icon; id is the identity of an icon resource in the module Dynamic Link Library (DLL).
ICONPOS	x,y	Sets the object's initial icon position. The x and y values represent the position in the object's folder in percentage coordinates.
TEMPLATE	YES	Creates an object as a template.
	NO	Resets object's template property.

Figure 5. WPFolder Setup String Parameters

We need to decide how to fill in the blanks to create this program object, just as we did when creating the folder. First, classname needs to be WPProgram since we will be creating a program object. The title is easy: use C&C Application. The location is different from when the folder was created. Since we wanted the folder on the Desktop, we used its object ID <WP_DESKTOP>. Now since we want to put the program object inside the folder that we created, use that folder's object ID, <C&C_COMPANY_FOLDER>.

The setup field for a program object has a few more options than for a folder object. First, let's give the program object the object ID <C&C_APPLICATION>. Next, tell the program object the path and name of the program that we want it to run. Instead of writing a real C&C application, use the OS/2 command processor. For the EXENAME field, use C:\OS2\CMD.EXE (assuming that OS/2 is installed on the C: drive). The last field, PROGTYP, tells what kind of program it is. Set this up as an OS/2 window; use WINDOWABLEVIO as the value.

That is all for creating a program object. Try typing the program, as shown in Figure 9, and then running it.

Creating a Shadow Object

To finish the request from C&C Company, one more task must be completed, and that is creating a shadow of the OS/2 window command prompt and placing it on the Desktop for easy access. The SysCreateObject function will be used again, but with WPSshadow as the classname.

A shadow is probably the easiest object to create. Let's go through the program and fill in the blanks. First, the classname needs to be WPSshadow since we are creating a shadow. Second, the title can be anything, so call it OS/2 Window^Shadow. Again, the caret sym-

Keyname	Value	Description
EXENAME	filename	Sets the name of the program.
PARAMETERS	params	Sets the parameters list, which may include substitution characters.
STARTUPDIR	pathname	Sets the working directory.
PROGTYPE	FULLSCREEN	Sets the session type to OS/2 full screen.
	PM	Sets the session type to PM.
	SEPARATEWIN	Sets the session type to WIN-OS2 window running in a separate VDM.
	VDM	Sets the session type to DOS full screen.
	WIN	Sets the session type to WIN-OS2 full screen.
	WINDOWABLEVIO	Sets the session type to OS/2 windowed.
	WINDOWEDVDM	Sets the session type to DOS windowed.
	WINDOWEDWIN	Sets the session type to WIN-OS2 windowed.
MINIMIZED	YES	Minimizes the Start program.
MAXIMIZED	YES	Maximizes the Start program.
NOAUTOCLOSE	YES	Leaves the window open when the program terminates.
	NO	Closes the window when the program terminates.
ASSOCFILTER	filters	Sets the file name filter for files associated with this program; multiple filters are separated by commas.
ASSOCTYPE	type	Sets the type of files associated to this program; multiple filters are separated by commas.
MINWIN	HIDE	Views of this object will hide when their minimize button is selected.
	VIEWER	Views of this object will minimize to the minimized window viewer when their minimize button is selected.
	DESKTOP	Views of this object will minimize to the Desktop when their minimize button is selected.
VIEWBUTTON	HIDE	Views of this object will have a hide button as opposed to a minimize button.
	MINIMIZE	Views of this object will have a minimize button as opposed to a hide button.
CCVIEW	YES	New views of this object will be created every time the user selects open.
	NO	Open views of this object will resurface when the user selects open.
ICONFILE	filename	Sets the object's icon.
ICONRESOURCE	id,module	Sets the object's icon. The id is the identity of an icon resource in the module DLL.
ICONPOS	x,y	Sets the object's initial icon position. The x and y values represent the position in the object's folder in percentage coordinates.
TEMPLATE	YES	Creates an object as a template.
	NO	Resets an object's template property.
OBJECTID	<name>	Sets the object's identity. The object ID will stay with the object even if it is moved or renamed. An object ID is any unique string enclosed within <>. This also may be a real name specified as a fully qualified path name.
HELPPANEL	id	Sets the object's default help panel.
HELPLIBRARY	filename	Sets the help library.

Figure 6. WPPProgram Setup String Parameters (Continued)

Keyname	Value	Description
OPEN	SETTINGS	Opens the settings view when object is created.
	DEFAULT	Opens the default view when an object is created.
NODELETE	YES	Will not allow you to delete the object.
NOCOPY	YES	Will not allow you to make a copy.
NOMOVE	YES	Will not allow you to move the object to another folder; will create shadow on a move.
NODRAG	YES	Will not allow you to drag the object.
NOLINK	YES	Will not allow you to create a shadow link.
NOSHADOW	YES	Will not allow you to create a shadow link.
NORENAME	YES	Will not allow you to rename the object.
NOPRINT	YES	Will not allow you to print the object.
NOTVISIBLE	YES	Will not display the object.

Figure 6. WProgram Setup String Parameters

COM_HOLD	HW_NOSOUND
DOS_BACKGROUND_EXECUTION	HW_ROM_TO_RAM
DOS_BREAK	HW_TIMER
DOS_DEVICE	IDLE_SECONDS
DOS_FCBS	IDLE_SENSITIVITY
DOS_FCBS_KEEP	KBD_ALTHOME_BYPASS
DOS_FILES	KBD_BUFFER_EXTEND
DOS_HIGH	KBD_RATE_LOCK
DOS_LASTDRIVE	MEM_INCLUDE_REGIONS
DOS_RMSIZE	MEM_EXCLUDE_REGIONS
DOS_SHELL	MOUSE_EXCLUSIVE_ACCESS
DOS_STARTUP_DRIVE	PRINT_TIMEOUT
DOS_UMB	VIDEO_FASTPASTERVIDEO_MODE_RESTRICTION
DOS_VERSION	VIDEO_ONDEMAND_MEMORY
DPMI_DOS_API	VIDEO_RETRACE_EMULATION
DPMI_MEMORY_LIMIT	VIDEO_ROM_EMULATION
DPMI_NETWORK_BUFF_SIZE	VIDEO_SWITCH_NOTIFICATION
DPMI_DOS_API	VIDEO_WINDOW_REFRESH
EMS_FRAME_LOCATION	VIDEO_8514A_XGA_IOTRAP
EMS_HIGH_OS_MAP_REGION	XMS_HANDLES
EMS_LOW_OS_MAP_REGION	XMS_MEMORY_LIMIT
EMS_MEMORY_LIMIT	XMS_MINIMUM_HMA

Figure 7. DOS Setting Parameters for WProgram Setup String

bol separates the first line from the second. Third, the location field shows where to place the object, so use <WP_DESKTOP> since we want the shadow to be placed on the Desktop. Finally, the SHADOWID field is asking what object we want to have shadowed. In this case, we are creating a shadow of the OS/2 window

command prompt, for which the object ID is <WP_OS2WIN>. Once the blanks are filled in, type in the program and then run it, as shown in Figure 10.

Before creating the LCU command file, let's put all the programs into one. We could simply merge the files

and do some editing, but there is an easier way when programming multiple objects to be created. Let's take a look.

If you have programmed before, this is easy. If all these individual programs were merged into one, you would quickly notice the amount of

```

/* FOLDER.CMD  REXX program to create a folder          */
/* Load REXXUTIL DLL                                   */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

/* Fill in the "blanks" as parameters to pass to SysCreateObject */

classname='WPFolder'
title=    'C&C Company^Folder'
location= '<WP_DESKTOP>'
setup=    'OBJECTID=<C&C_COMPANY_FOLDER>;'

result = SysCreateObject(classname, title, location, setup, 'f')

/* Once the call has been made, test to see if it was successful */
/* by examining the variable result.                               */

if result = 1 then
  say 'Successful!!  'title 'has been created!'
else
  say 'NOT!  Failed to create 'title'. Return code is 'result'

exit

```

Figure 8. REXX Program (FOLDER.CMD) to Create a Folder Object

```

/* PROGRAM.CMD  REXX program to add a program object    */
/* Load REXXUTIL DLL                                   */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

/* Fill in the "blanks" as parameters to pass to SysCreateObject. */
/* In the setup field, the three items are concatenated with the ||, */
/* characters. It is done this way to ensure there are no spaces  */
/* when concatenating.                                           */

classname='WPProgram'
title=    'C&C^Application'
location= '<C&C_COMPANY_FOLDER>'
setup=    'OBJECTID=<C&C_APPLICATION>;'||,
          'EXENAME=C:\OS2\CMD.EXE;'||,
          'PROGTYPE=WINDOWABLEVIO;'

result = SysCreateObject(classname, title, location, setup, 'f')

/* Once the call has been made, test to see if it was successful */
/* by examining the variable result.                               */

if result = 1 then
  say 'Successful!!  'title 'has been created!'
else
  say 'NOT!  Failed to create 'title'. Return code is 'result'

exit

```

Figure 9. REXX Program (PROGRAM.CMD) for Creating a Program Object


```

/* SHADOW.COMD  REXX program to create a shadow                                */
/* Load REXXUTIL DLL                                                            */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

/* Fill in the "blanks" as parameters to pass to SysCreateObject                */

classname='WPSHADOW'
title=    'OS/2 Window^Shadow'
location= '<WP_DESKTOP>'
setup=    'SHADOWID=<WP_OS2WIN>'
result = SysCreateObject(classname, title, location, setup, 'f')

/* Once the call has been made, test to see if it was successful                */
/* by examining the variable result.                                           */

if result = 1 then
  say 'Successful!!  'title 'has been created!'
else
  say 'NOT!  Failed to create 'title'.  Return code is 'result'
exit

```

Figure 10. REXX Program (SHADOW.COMD) for Creating a Shadow

```

/* SETUPWPS.COMD REXX program to create a folder, program, & shadow */
/* Load REXXUTIL DLL                                                            */
call RxFuncAdd 'SysLoadFuncs', 'RexxUtil', 'SysLoadFuncs'
call SysLoadFuncs

/* First, create the C&C Company folder on the Desktop, using the */
/* lighthouse bitmap as the background for the folder, */
/* automatically opening the folder, and using one of the icons */
/* that is included with OS/2 2.0. */
/* */
/* Assumes installation of optional bitmaps. */
/* See ICONRES.COMD sample program from Mike Lamb's document1 for */
/* more information on the ICONRESOURCE values. */

classname='WPFolder'
title=    'C&C Company^Folder'
location= '<WP_DESKTOP>'
setup=    'OBJECTID=<C&C_COMPANY_FOLDER>;',
          'BACKGROUND=C:\OS2\BITMAP\LIGHTHOU.VGA;',
          'OPEN=DEFAULT;',
          'ICONRESOURCE=03, WPCONFIG;'
call BuildObject

/* Next, create the C&C Application program object, which just */
/* calls an OS/2 window.  Assumes OS/2 installed on drive C: */

classname='WPProgram'
title=    'C&C^Application'
location= '<C&C_COMPANY_FOLDER>'
setup=    'OBJECTID=<C&C_APPLICATION>;',
          'EXENAME=C:\OS2\CMD.EXE;',
          'PROGTYPE=WINDOWABLEVIO;',
          'ICONRESOURCE=19, PMWP;'

```

¹ See the References section at the end of this article.

Figure 11. SETUPWPS.COMD (Continued)

```

call BuildObject

/* Next, create a shadow of the Klondike program in the C&C      */
/* folder. Assumes Klondike game was installed                  */
/*                                                              */

classname='WPSshadow'
title=   'Solitaire Shadow'
location= '<C&C_COMPANY_FOLDER>'
setup=   'SHADOWID=<WP_KLDK>'
call BuildObject

/* Finally, create a shadow of the OS/2 window on the Desktop */
/*                                                              */

classname='WPSshadow'
title=   'OS/2 Window^Shadow'
location= '<WP_DESKTOP>'
setup=   'SHADOWID=<WP_OS2WIN>'
call BuildObject
exit

/* Build Object procedure                                     */
/*                                                              */

BuildObject:
  result = SysCreateObject(classname, title, location, setup, 'f')
  if result = 1 then
    say 'Successful!!  'title 'has been created!'
  else
    say 'NOT! Failed to create 'title'. Return code is 'result
Return

```

Figure 11. SETUPWPS.COMD

repetition. For every object to be created, there would be the same call to SysCreateObject and the same test to see if the call was successful.

If you are creating multiple objects, the easiest way to program it is to set up a procedure to handle the repetitive tasks, as shown in Figure 11. Review the program and see if you can understand what happens when it is run. Note that a few more parameters have been added to some calls to spice up the program. Also note that a shadow of the Klondike program has been added to the C&C Company folder, just in case they get bored while running their C&C application.

Creating the LCU Command File

This section assumes that you clearly understand the CID process, how it works, and how to modify the LCU

command file to achieve the desired results. To learn more about the CID process, see the reference section for further reading.

If you are planning to modify the WPS using REXX as part of a CID installation, always apply the ServicePak to the client machines before modifying the WPS (this assumes that you are CID-installing OS/2 2.0 onto the client machines). The ServicePak contains many fixes to correct bugs that were in the OS/2 2.0 WPS DLLs. If you do not apply the ServicePak, you may not be able to create objects.

Since the LCU command file is actually a REXX program, it is easy to modify so that the final task of the installation procedure calls the SETUPWPS.COMD program to customize each Desktop. When adding programs to the LCU command file,

be sure to modify two sections: the definition and the installation sections. Figure 12 shows examples of sections from an LCU command file that is set up to install OS/2 2.0 and the ServicePak, and then to customize the Desktop using the SETUPWPS.COMD that was just created. For simple REXX procedures such as our customization program, the definition section is easy. All that is needed is a title for the procedure, which will be placed into the product name field, and the name of the command file to place into the instprog field. Since the clients will be running the customization program from the code server, the best place to put the program on the code server is in the \CID\EXE subdirectory.

The installation section also is easy, as shown in Figure 12. Why was the customization program put into a separate queue (OVERALLSTATE=4)

Definition Section

```

Call RxFuncAdd 'SysSleep', 'REXXUTIL', 'SysSleep' /* Load SysSleep */
...
...
...

x.setupwps = 8 /* structure index */
x.8.name='SETUPWPS Procedure' /* product name */
x.8.statevar = '' /* state variable name */
x.8.instprog = 'x:\exe\setupwps.cmd' /* fully qualified install prgm name*/
x.8.rspdir = '' /* no auto selection */
x.8.default = '' /* */

```

Installation Section

```

Do Forever
  Select
    when OVERALL_STATE = 0 then do
      if BootDriveIsDiskette() == YES then iterate /* Check if booted from disk*/
                                                    /* if so, then goto state 1 */
      if RunInstall(x.semaint) == BAD_RC then exit /* Install maint system */
      if RunInstall(x.laps_prep) == BAD_RC then exit /* Install LAPS prep system*/
      if RunInstall(x.thinifs) == BAD_RC then exit /* Install SRVIFS requester*/
      if RunInstall(x.casinst1) == BAD_RC then exit /* Install LCU */
      Call CheckBoot /* Reboot if requested */
    end
    when OVERALL_STATE = 1 then do
      if RunInstall(x.seinst) == BAD_RC then exit /* Install operating system*/
      if RunInstall(x.laps) == BAD_RC then exit /* Install LAPS */
      if RunInstall(x.thinifs) == BAD_RC then exit /* Install SRVIFS requester*/
      if RunInstall(x.casinst1) == BAD_RC then exit /* Install LCU */
      Call CheckBoot /* Reboot if requested */
    end
    when OVERALL_STATE = 2 then do
      if RunInstall(x.semaint) == BAD_RC then exit /* Install maint system */
      if RunInstall(x.laps_prep) == BAD_RC then exit /* Install LAPS prep system*/
      if RunInstall(x.thinifs) == BAD_RC then exit /* Install SRVIFS requester*/
      if RunInstall(x.casinst1) == BAD_RC then exit /* Install LCU */
      Call CheckBoot /* Reboot if requested */
    end
    when OVERALL_STATE = 3 then do
      if RunInstall(x.fservice) == BAD_RC then exit /* Install OS/2 ServicePak */
      Call CheckBoot /* Reboot if requested */
    end
    when OVERALL_STATE = 4 then do
      Call SysSleep 45 /* Wait for WPS to start */
      if RunInstall(x.setupwps) == BAD_RC then exit /* Customize WPS Desktop */
      if RunInstall(x.ifsdel) == BAD_RC then exit /* Delete SRVIFS requester */
      if RunInstall(x.casdelet) == BAD_RC then exit /* Delete LCU */
      Call Reboot /* Reboot */
    end
  end
end
end

```

Figure 12. LCU Command File Sections

from the ServicePak? The reason is that Presentation Manager (PM) and the WPS must be up and running so that the customization program can work.

A SysSleep is added immediately before SETUPWPS.COM is called, where OVERALL_STATE=4. This is because the client has just rebooted, and the WPS usually takes some time

to come up. If the WPS is not up when you try to modify the WPS with a REXX program, the program will obviously fail. Therefore, the SysSleep is coded to put the client

on hold for 45 seconds, which is enough time for the WPS to come up even on slow computers.

Because SysSleep is a function that is included with REXXUTIL, the LCU command file must load that function before calling it. The RxFuncAdd line in Figure 12 shows how to load the SysSleep function. This line is at the beginning of the definition section within the LCU command file.

The following steps summarize how to enable the customization program to work from a code server as part of a CID installation:

1. Put the program on the server, preferably in the \CID\EXE subdirectory.
2. Modify the definition section of the LCU command file to define where the customization program is located and to load the SysSleep function, if needed.
3. Modify the installation section of the LCU command file to define the order of products to be installed, including our Desktop customization program.

Using REXX to Customize the Workplace Shell—Part 2

This article focuses on creating WPS objects using REXX and the REXXUTIL SysCreateObject function. It also shows how to incorporate the program into an LCU command file as part of a CID installation. Here we have only discussed how to *add* objects to the WPS. The other side of customizing the WPS is how to

modify the existing objects, and how to change the system installation defaults, such as the default gray color, for the WPS Desktop. That will be discussed in the second article to appear in the July issue.

We will look at using other functions that are included with the REXXUTIL DLL, such as SysIni, SysSetObjectData, and SysDestroyObject. Note that some of these functions were not part of the OS/2 2.0 release of REXXUTIL; you will need REXXUTIL from the ServicePak or a later release of OS/2. We also will discuss more about object IDs and how to use them to manipulate the WPS. Many examples will be included to get you started.

References

REXX

- Cowlshaw, Mike. *The REXX Language: A Practical Approach to Programming*. Englewood, NJ: Prentice Hall, 1990 (B35-5100). ISBN 0-13-780651-5.
- Lamb, Michael A. *Using the REXXUTIL SysCreateObject Function*. This document is part of the REXX20 package and can be found on various Bulletin Board Services (BBSs), such as the OS/2 BBS, CompuServe®, and the IBM National Support Center BBS. The package contains an updated version of REXX (updated from the OS/2 2.0 level), plus the document in text format. The document was originally an append on the OS/2 BBS.

- Lamb, Mike and Kurtz, Dick. *REXXUTIL Information and Samples*. Dick Kurtz has taken Mike Lamb's document and converted it to INF format (for use with OS/2's View utility), SCRIPT (for printing on an IBM 3820 printer), and BOOK format (for use with the READ/2 program). Ask your IBM representative to request MLAMB DLL from RRRKURTZ at PKEDVM9. Note that this is the same document as listed above; it is just formatted.

CID

- *Automated Installation for CID-Enabled OS/2 Version 2.0* (GG24-3783, ITSC Red Book)
- *Automated Installation for CID-Enabled Extended Services, LAN Server Version 3.0, and Network Transport Services/2* (GG24-3781, ITSC Red Book)
- *Network Transport Services/2: Redirected Installation and Configuration Guide* (S96F-8488)

Bret Curran is a member of the OS/2 Systems Support and Services team in IBM's Personal Systems Competency Center in Roanoke, Texas. As a member of their OS/2 Systems Management team, Bret's key focus is on Configuration, Installation, and Distribution (CID). He received a bachelor's and a master's degree in business computer information systems from the University of North Texas.

OS/2 Distributed Systems Management with LAN NetView

Bart Jacob
IBM Corporation
Austin, Texas

Marie-Christine Laffont
IBM Corporation
Paris, France

This article discusses IBM LAN NetView Agents, Agents Extended products, and the Management Information Base (MIB) they represent. It is intended for those who desire a general understanding of systems management capabilities provided by IBM LAN NetView Management. It focuses on LAN NetView agents and the potential for using them to develop systems management applications. At the time this article was written, IBM had not yet announced availability dates for the IBM LAN NetView Management family. For availability and additional technical information, contact your IBM representative.

The IBM LAN NetView Management family of products provides a framework and applications to implement OS/2-based distributed systems management solutions. The Management framework uses industry-standard interfaces and protocols that enable an OS/2 system to manage heterogeneous systems in a LAN environment. An OS/2 system also can be managed by other systems that conform to the same standards. The LAN NetView Management family of products is based on systems management standards developed by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) as part of their work on Open Systems Interconnection (OSI).

Agents are applications that perform operations on managed objects at the request of managing applications and emit notifications on behalf of managed objects. LAN NetView Management agents are a key component of IBM's LAN NetView Management family of products. They provide sys-

tems management application developers with access to many objects associated with OS/2, its major subsystems (LAN Server 3.0, Database Manager, and Communications Man-

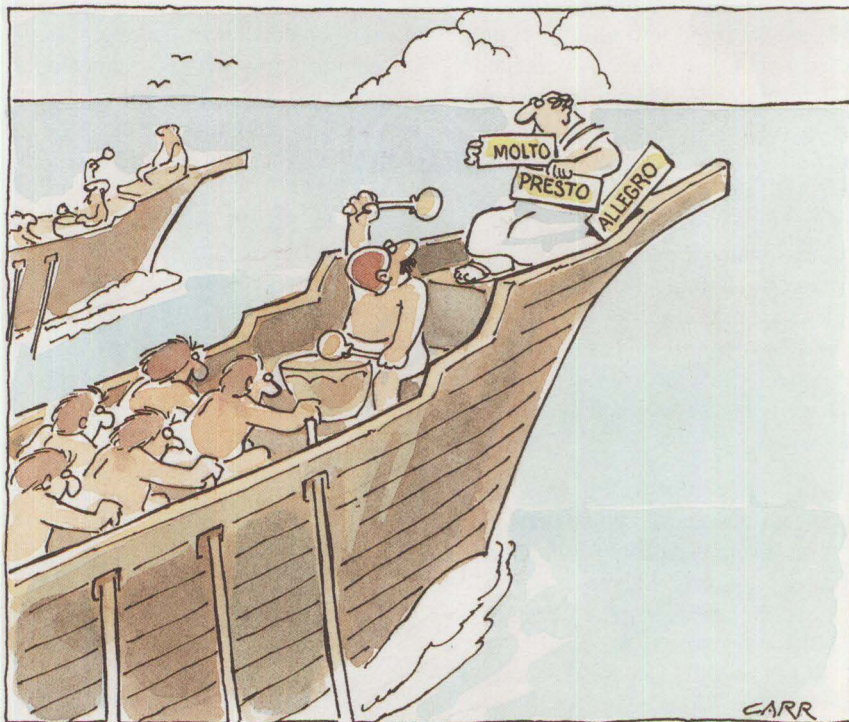
ager), DOS 5.0, Microsoft Windows 3.1, and the hardware on which they are running. IBM LAN NetView Management applications will use these agents, just as systems management applications written by other vendors or user organizations will use them. The Management Information Base (MIB) defined by these agents contains a rich set of object classes that can be used by applications spanning all systems management disciplines.

The LAN NetView Management Framework

The LAN NetView Management family of products includes a systems management framework and applications. The framework enables vendors or user organizations to write management applications.

The Object-Oriented Model

Communications protocols alone are not enough for distributed systems management. Distributed management programs need standard descriptions of resources to be managed. Therefore, the LAN NetView Man-



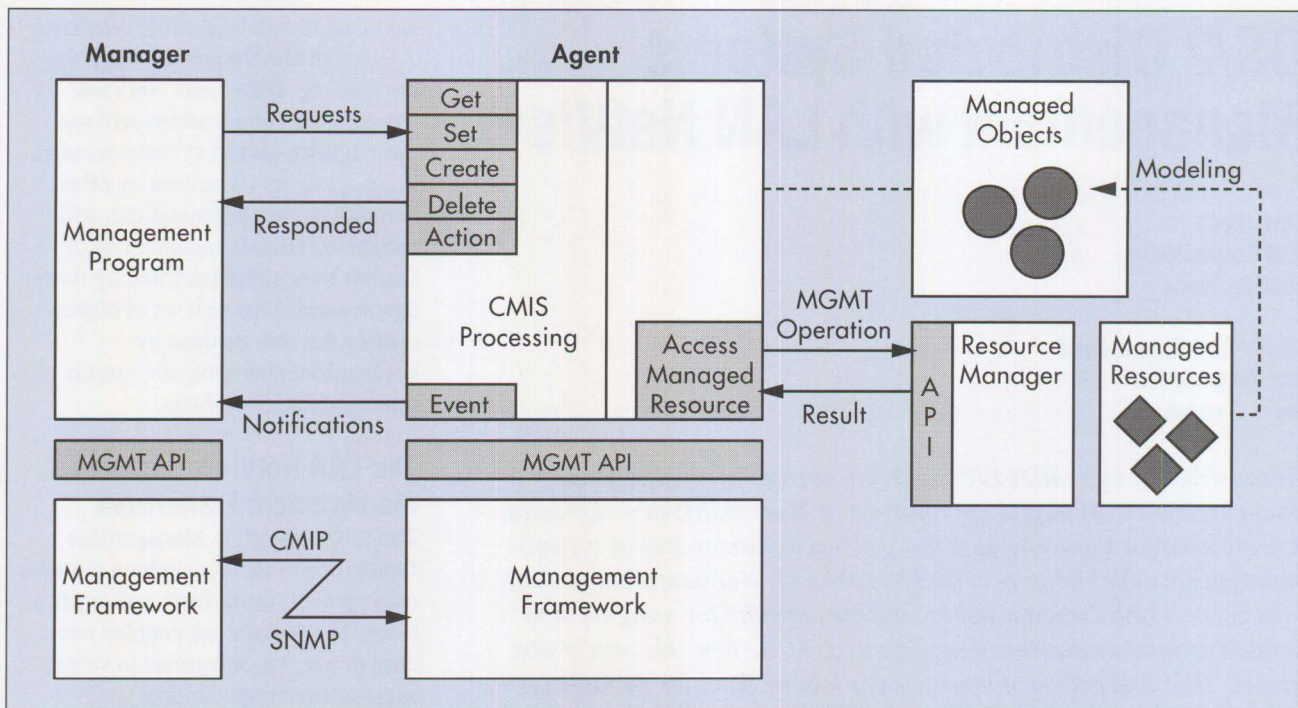


Figure 1. Systems Management Interactions

agement family uses an OSI object-oriented model for encapsulating these resources and standardizing the interfaces they present to the network.

In an object-oriented model, code and data are fused into a single entity called a *managed object*. A managed object is composed of a set of characteristics, actions that it can perform, and notifications that it can emit. In the LAN NetView Management environment, a managed object is an abstraction that models a physical resource (such as a workstation) or a logical resource (such as a file system). Resources are managed through the managed-object abstraction. A managed object is defined by the following:

- **Attributes:** Information about the object (Operations can be performed on objects or specific attributes of objects. Examples of these operations include GET and SET for attribute values, and CREATE and DELETE for objects.)

- **Actions:** Functions that an object can perform
- **Notifications:** Unsolicited messages that objects send asynchronously to agents and managers

A set of object classes and their associated attributes, actions, and notifications compose the MIB. Management applications are written to interface with objects defined by the MIB.

Managers, Agents, and Managed-Object Interaction

A distributed systems management environment is based on managers and agents. An agent is the part of a distributed management program that supervises one or more managed objects. The agent receives requests for operations to be performed on managed objects or requests for objects to perform certain actions. The agent is responsible for passing these requests to a resource manager that provides interfaces to implement the specific request in each subsystem that controls a set of resources.

Resource managers in an OS/2 environment are the LAN Server, Communications Manager, Database Manager, and even the operating system. The agent also is responsible for emitting notifications (events/traps) when it detects special conditions in the managed object.

A *manager* is the part of the distributed management program that issues requests for actions and receives notifications. A manager uses the services of one or more agents. Managers do not manage resources directly; they issue requests to objects that are represented by agents. Figure 1 shows the interactions between managers, agents, and managed resources.

Interfaces used between agents and the resources they represent are not standardized. Agents can use any interfaces available in the system to implement operations or actions. Although the framework is object-oriented, there can be non-object-oriented interfaces between agents

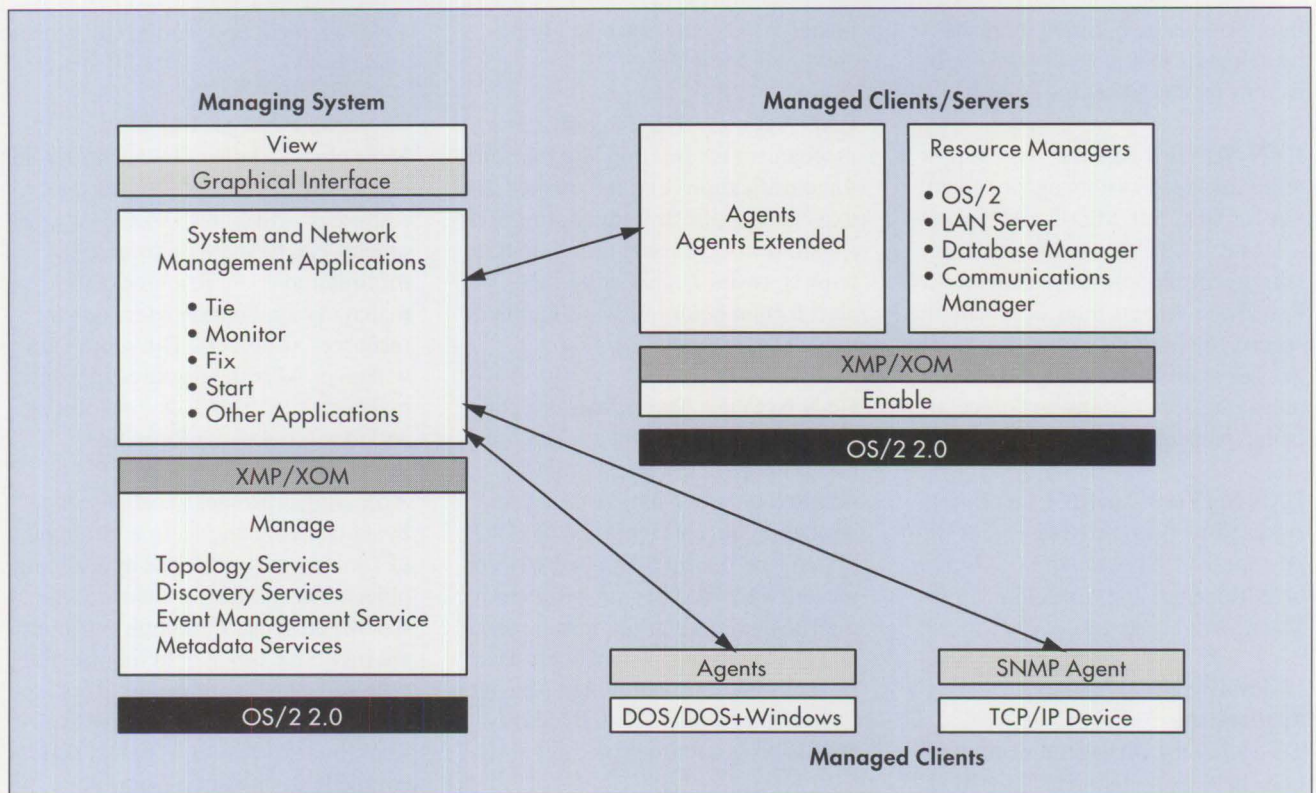


Figure 2. LAN NetView Management Environment

and resource managers. This interaction is hidden completely from the managing system.

LAN NetView Management architecture currently supports two different sets of protocols and services to be used between managing and managed systems:

- **Common Management Information Protocol (CMIP) over Link Level Control (CMOL) and CMIP over TCP (CMOT).** The services defined for CMIP are called Common Management Information Services (CMIS).
- **Simple Network Management Protocol (SNMP)** for managing Transmission Control Protocol/Internet Protocol (TCP/IP) networks and devices.

The LAN NetView Management framework provides three key Application Programming Interfaces (APIs) for building applications:

- **X/Open OSI-Abstract Data Manipulation (XOM) API.** Used to manipulate data structures associated with objects.
- **X/Open Management Protocol (XMP) API.** Used for standards-based process-to-process communications between a managing and a managed system.
- **Graphical User Interface (GUI) API.** Used by applications on the managing system to provide a consistent user interface to all management applications in a LAN NetView environment; also makes it easy for users to navigate between systems management applications.

The LAN NetView Management Family of Products

The LAN NetView Management family of products supports both managing and managed systems, as shown in Figure 2.

LAN NetView Manage: Manage provides core functions required by a managing system, including event management, metadata, topology, and discovery services. Manage provides the industry-standard XMP API and the XOM API for developing management applications. It requires OS/2 2.0.

LAN NetView View: View provides a GUI to the LAN NetView Management environment. Developers can use View programming interfaces to deliver a consistent look and feel to their management applications. View is used for easy navigation through Management applications. It also provides services to display management data in progressive layers of detail.

LAN NetView Enable: Enable provides the managed system platform on OS/2 2.0-based systems. It also provides XMP/XOM programming

interfaces for developing management agents that interact with applications on the managing system.

LAN NetView Agents: Agents provides managed system agent capabilities for DOS 5.0, Microsoft Windows 3.1, and OS/2 2.0 systems and contains agent support for the IBM LAN Requester. Agents interfaces with the specific operating system or LAN Requester to provide a bridge between managing applications and objects being managed.

LAN NetView Agents Extended: Agents Extended provides agent support for the LAN Server 3.0, Database Manager, and Communications Manager subsystems.

LAN NetView Management Applications

The following management applications have been announced by IBM. Contact your IBM representative for the latest information about availability. These applications run on a managing system (requiring Manage and View) and use the Agents and Agents Extended software on managed systems.

LAN NetView Fix: Fix is a problem management application that helps to track problems identified by notifications from agents. Information such as the following can be displayed graphically or textually:

- Notifications related to the problem
- Resources affected by the problem
- Diagnostic conclusions
- Recommended actions

LAN NetView Monitor: Monitor is a performance management application. It enables system administrators and capacity planners to monitor system performance (including the setting of thresholds), analyze perfor-

mance trends, and perform tuning and load balancing.

LAN NetView Tie: Tie provides a mechanism for filtering and transmitting notifications emitted on the LAN to a NetView host. It transforms notifications in OSI alarm format to Systems Network Architecture (SNA) alert format before forwarding them to a NetView host.

LAN NetView Start: Start is a configuration management application that provides a graphical, object-oriented interface for managing and validating the configuration of OS/2 systems on the LAN. It also provides the software files that are required by the IBM Configuration, Installation, and Distribution (CID) process used by the NetView Distribution Manager (NVDM) and the LAN CID Utility (LCU) program products.

LAN NetView Agents

This section provides additional detail about agents and introduces new concepts that are key to understanding agents and the MIB.

Resource Manager Agents

LAN NetView Agents and Agents Extended products define objects for the following resource managers:

- The operating system (OS/2, DOS 5.0, and Microsoft Windows 3.1)
- LAN Server 3.0
- Communications Manager
- Database Manager

Operating system agents are designed with as much commonality as possible, enabling management applications to be written with a minimum of operating system-specific code. For example, several object classes relating to hardware are defined by operating system agents. It is possible to reference the same object class, such as a hard disk drive, from

a management application that is managing both DOS and OS/2 systems.

Resource Manager Objects

Most objects defined by agents relate to specific physical or logical resources. Attributes associated with objects can be mapped to specific information about resources. The managing application addresses the resources as objects. The agent uses whatever APIs the resource manager makes available to access resources and the associated information.

Accessing attributes is usually done by a managing application issuing a GET or SET request to the appropriate object. This returns or modifies the specific attribute. If multiple attributes are involved, this would require multiple GET or SET operations. This is fine for most management applications, but sometimes a slightly different mode of operation is required. In these cases, special objects called management support objects are defined.

Management Support Objects (Monitor-Scanner)

LAN NetView Agents and Agents Extended have special *management support objects* to support the LAN NetView Monitor or other applications that need access to system performance data. OS/2 2.0 provides performance instrumentation for methods associated with these objects. This instrumentation provides access to low-level counters, timers, and control blocks necessary for performance management. The following several reasons show why performance data may need to be handled differently from other attributes:

- Access to different attributes must be synchronized. A series of GET functions to take a snapshot of multiple attributes would not be adequate because each would return attribute values from a different time.

- Certain counters and timers should be monitored or scanned so a notification can be generated when certain thresholds are reached.
- Certain data should be logged or captured over time and at specific intervals.

OS/2's performance instrumentation provides system-level functions required to achieve these objectives. However, implementing this through standard GET/SET interfaces to resource manager objects would be difficult, if not impossible. A special Monitor-Scanner object class provides interfaces for a managing application. The monitor-scanner object interfaces directly with OS/2 to provide the functions.

Performance information can be mapped to attributes associated with resource manager objects. MIBs associated with OS/2 and LAN Server agents define attributes associated with the monitor-scanner object. However, a managing application should not issue a GET or a SET to the resource manager object agent to retrieve or modify these attributes. Instead, the managing application should CREATE an instance of the monitor-scanner object. In creating this instance, it defines parameters, such as the specific attributes to monitor, the granularity period (how often it scans the information), thresholds, schedules, and so on.

Once the monitor-scanner object is instantiated, the performance data is gathered and monitored. Notifications are generated when thresholds are exceeded. If a managing application needs to retrieve the monitored information, it issues a specific ACTION, which returns the data.

Other management support objects are related to the monitor-scanner and the log files that it generates. Information related to these objects and

how to access the information they represent will be contained in the LAN NetView MIB documentation.

Solutions Based on LAN NetView Agents

The MIB, defined by LAN NetView Agents and LAN NetView Agents Extended products, enables applications to be written that take advantage of IBM-supplied agents. This eliminates the need for application developers to write their own agents and increases consistency across applications developed independently.

The primary objective of IBM's agent design team was to provide support for monitoring and controlling operating systems and subsystems. The secondary objective was to monitor and control specific resources owned by each operating system and subsystem. Each objective is addressed considering the five primary functional areas of systems management: operations, configuration, problem, performance, and inventory management.

Most of these areas are addressed in part by IBM LAN NetView Management applications. However, users may want additional management applications for their specific needs.

Operations Management

LAN NetView agents provide functions for operations management programs to monitor and alter the operational state of systems on a network. Functions include the following:

- Monitoring the status of resources, such as LAN servers, database servers, peripheral devices (such as printers), and critical processes or threads
- Querying and changing operational characteristics of resources, such as holding or releasing print queues
- Performing actions on resources, such as system shutdown and restart

Monitoring the Status of Resources

LAN NetView agents monitor both hardware and software resources. They provide changes in status through unsolicited notifications and enable administrators to explicitly query the status of a particular resource.

Figure 3 shows the MIB state attributes defined by ISO/IEC 10165.2 (X.721). LAN NetView agents use these states; one or more can apply to a particular resource at any time. IBM has defined additional states that are resource-specific and are not covered by those in Figure 3. Figure 4 lists some objects represented by LAN NetView agents that provide status information.

Querying/Changing Operational Characteristics of Resources

Besides querying a resource's operational status (such as operational, online, and offline), an operations management program can issue a GET operation to retrieve object attributes. These attributes contain information about operational characteristics of managed resources such as size of disk and percent utilized. Figure 5 shows examples of object attributes. Many attributes can be altered through a SET command issued to the agent.

Performing Actions on Resources

An operational management application can initiate actions to be performed by an agent on an object. The following list provides examples of some actions that are supported by LAN NetView agents:

- Shutdown/Restart the operating system
- Hold/Release a queue or a job in a queue
- Pause/Continue printing
- Pause/Continue LAN Server and Requester services
- Clear LAN Server statistics

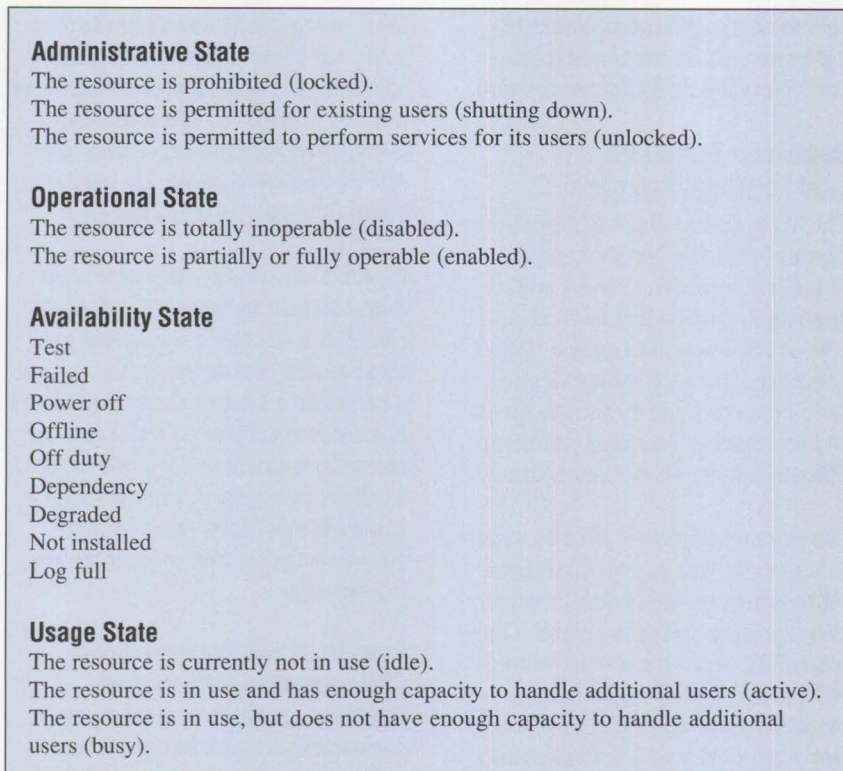


Figure 3. MIB State Attributes

- Activate/Deactivate Communications Manager, Advanced Peer-to-Peer Networking (APPN) node
- Activate/Deactivate adapter
- Activate/Deactivate logical link
- Deactivate LU 6.2 session
- Deactivate LU 6.2 conversation
- Activate/Deactivate Database Manager
- Activate/Deactivate a database
- Create/Drop a database
- Receiving notifications of configuration changes to selected systems

LAN NetView Management agents provide access to both hardware and software configuration information. The LAN NetView MIB provides object class definitions to develop a powerful configuration management

Configuration Management

Configuration management involves determining, altering, and tracking the configuration of systems within a network. These tasks can be summarized as follows:

- Retrieving the configuration of selected systems
- Changing configuration parameters on selected systems

Hardware Objects

Machine
 Display
 CPU
 Logical ports

OS/2 Programming Objects

Processes
 Threads

Printing Objects

Printers
 Queues
 Jobs

Subsystem Objects

LAN Server Objects

- LAN Server
- LAN Requester
- IBMLAN.INI Configuration

Communications Manager Objects

- Configuration File
- Advanced Peer-to-Peer Networking (APPN) Node
- Logical Link
- Transaction Program
- SNA Session
- Physical Port

Database Manager Objects

- Database Manager
- Database

Database Gateway

Figure 4. Objects Represented by LAN NetView Agents that Provide Status Information

Attribute	Description
Logical volumes	Volume size, allocation information, utilization
Monitored files	Content (CRC value), size, last modification date/time
Printer	Current print job, printer status
Print queue	List of printers, priority, queue status, number of jobs in the queue
Spooled job	Printer name, position in queue, name of user
LAN Server	Service statistics, current status
Communications Manager	Status and information for APPN nodes, logical links, SNA sessions, transaction programs, ports, and LU 6.2 conversations
Database Manager	Database state (consistent, requires backup, roll-forward in progress)

Figure 5. Examples of Object Attributes

application that can access most configuration parameters within a managed system. Being alerted about changes to monitored files provides a key function for identifying changes to configuration files on key systems within the network.

Retrieving Configuration Information

The following list summarizes some configuration information that can be retrieved from a managed system running LAN NetView agents:

- **Operating System:** Version/level
- **Memory:** Installed memory, memory allocation (OS/2), expanded/extended memory allocation (DOS), memory configuration, virtual memory (Windows)
- **Current system:** Boot drive, time/date
- **Code page, country code**
- **System parameters:** CONFIG.SYS
- **LAN Server:** Version/level, initialization parameters (IBMLAN.INI), runtime parameters
- **Communications Manager:** Version/level, list of configuration files, active configuration file
- **Database Manager:** Version/level, configuration information

Changing the Configuration

LAN NetView agents enable a management program to alter various configuration parameters through a SET operation:

- **Parameters in the CONFIG.SYS file**
- **LAN Server:** All start-up (IBMLAN.INI) parameters can be set. Selected runtime parameters also can be set or modified by a management program.
- **Database Manager:** Catalog/uncatalog databases, create/delete database directory information,

create/delete database connection services information, create/delete node directory information. Database and Database Manager configuration values can be reset to defaults.

Tracking Changes to Configurations

Agents generate notifications whenever certain resources are created or deleted and certain attributes are changed, such as configuration parameters. In addition, a Monitored File object class enables agents to monitor critical files, such as configuration files, and emit notifications if they are changed.

Problem Management

Problem management involves monitoring resources, analyzing notifications emitted from objects, and performing actions to correct, avoid, or circumvent error conditions.

Monitoring Resources

Agents send heartbeat notifications on behalf of themselves, which enables the management program to be sure that an agent is operable. Agents also send notifications when objects are created or deleted. For example, a notification is generated when APPN becomes active on a node.

Analyzing Received Notifications

Management programs analyze notifications from agents to help determine the cause of failures. Following are examples of these types of notifications:

- **LAN Server**
 - **Quality of service alarms:** Network I/O error threshold reached, disk drive nearing capacity, audit log full, and so on.
 - **Equipment alarms:** Fault tolerance system fixed a bad sector; fault tolerance system detected a difference between the con-

tents of the primary and the secondary partitions of a mirrored hard disk drive; and so on.

- **Environmental alarms:** LAN Server detected the entry of multiple failed passwords, multiple unauthorized resource-access attempts, and so on.
- **LAN Requester**
 - **Quality of service alarms:** Error log has reached its maximum size, redirector has reached the configured threshold for a specified resource.
 - **Processing alarms:** Internal processing error, resources not available, and so on.

Detailed information about each error condition is transmitted in the notification.

- **Communications Manager**
 - **APPN node:** Insufficient storage for intermediary session setup (only by APPN network node), SNA protocol error, and so on.
 - **LAN adapter used for SNA:** Open failure detected by Token-Ring lobe, CSMACD bus inoperative.
 - **Synchronous Data Link Control (SDLC) adapter used for SNA:** Link error due to the remote link station address, link error due to bad line, and so on.

Error conditions that normally generate SNA alerts in the Communications Manager cause the Communications Manager agent to generate a notification to a managing system. The notification carries information about the event (probable cause, specific problem, severity, and proposed repair actions) and the problem data (product IDs, alert type, and failure cause).

- **Database Manager**

- Processing error alarm (internal processing errors)
- State changes
- Creation/deletion of objects

Notifications include the SQL return code if applicable.

Corrective Actions

Many capabilities discussed under other functional areas, such as operations management and configuration management, also apply within problem management. Once a problem is detected and the cause determined, operations or actions typically are performed to resolve the problem. For example, a configuration change may be required with a subsequent system restart. These capabilities are addressed in the sections relating to configuration and operations management.

Performance Management

Resources can be monitored to identify performance problems, isolate the causes, and correct the situations through load balancing and reconfiguration. The MIB provides access to many counters, timers, control blocks, and other information that supports powerful performance management applications. The following list shows some information that can be monitored through LAN NetView agents:

- **OS/2 CPU:** Number of interrupts, CPU idle time
- **OS/2 memory:** Swapping statistics, number of page faults
- **OS/2 disk:** Cache utilization
- **OS/2 file I/O:** Number of open files
- **Disk I/O:** Number of bytes read/written from disk
- **Logical serial port:** Time spent reading/writing
- **OS/2 thread:** CPU used by thread, time spent waiting

- **Logical parallel port:** Number of write operations
- **LAN Server:** Server activity in read and write operations, use of buffers

As key performance indicators are tracked, management programs can be alerted as thresholds are approached.

Inventory Management

Tracking inventory in a large network can be difficult. Workstations constantly are being added, removed, upgraded, and reconfigured. Software inventory can be even more dynamic because each program can be at a different revision level.

An inventory management program can retrieve vital product data for all subsystems (hardware and software) supported by LAN NetView agents. The following list provides examples of the type of inventory management information available through LAN NetView agents.

- **Software**

- Product name, version, and the Corrective Service Disk (CSD) of the operating system
- Product name, version, and the CSD of the OS/2 LAN Server, the computer name, and the domain name
- Product name, version, and the CSD of the Database Manager and Communications Manager

- **Hardware**

- Machine identifier specified by the manufacturer
- Machine type, location, owner, contact information (specified during system installation)
- Display type
- CPU type, coprocessor type
- Hard disk and diskette drive size and capacity

- Keyboard identifier and type
- Micro Channel Architecture adapter identifier

LAN NetView Manage provides topology and discovery services. If new nodes that enter the network contain LAN NetView agents, they are identified to the managing system. Manage also discovers other nodes, such as those running TCP/IP. An inventory management application residing on the managing system can query the system and add the information to an inventory database.

References

For more information about the building blocks for systems management applications using LAN NetView agents, see the LAN NetView Management documentation. IBM had not announced documentation for the IBM LAN NetView Management family at the time this article was written. For the latest technical information, contact your IBM representative.

Bart Jacob is a senior systems engineer with the IBM International Technical Support Center (ITSC) in Austin, Texas. He joined IBM in 1980 and held several support positions until joining ITSC in 1989. Bart is currently responsible for providing worldwide technical support for OS/2 systems management. He holds an MS degree in mathematics from Syracuse University in New York.

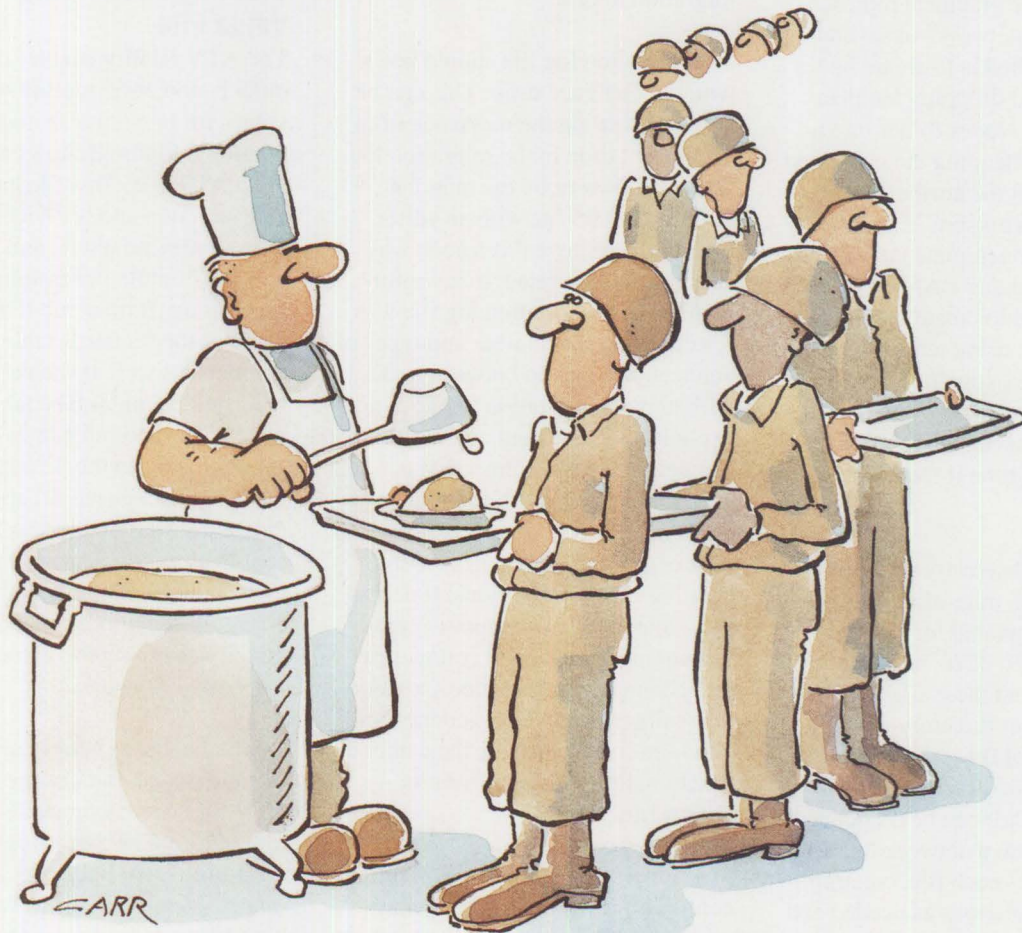
Marie-Christine Laffont is a systems engineer in Paris, France. She joined IBM in 1988 and is a specialist in OS/2. Marie-Christine is currently responsible for providing support for aerospace customers. She holds a bachelor's degree in industrial data processing from the University of Lyon in France.

Priming and Querying Your Start Network

Theodore Shrader
IBM Corporation
Austin, Texas

The IBM LAN NetView Start program gives LAN administrators the power to manage their networks by representing nodes and connections in an object-oriented graphical environment. It also provides the means to automatically create configuration files for applications on each node, and preserves this information for future use. Network representations can be created and modified easily through a variety of techniques, including Attribute Value (ATV) files. Users also can consult their stored network information by running queries against the data within their ASCII or SQL network databases. This article describes how to create a representation of your network with Start and how to run queries against your stored network.

The IBM LAN NetView Start program (Start) represents networks graphically. It gives users the ability to create response and LAN Configuration Utility (LCU) REXX command files for products installed on network nodes. The Start program is shown in Figure 1. Among other enhancements, Start Version 1.1 also enables users to create NetView DM/2 change files and introduces the concept of an application union viewer. The *application union viewer* lets users change application attributes across a collection of nodes at one time. Users can create networks from scratch or use Start's migration tools to represent existing networks through nodes and connections with the networks' topology objects.



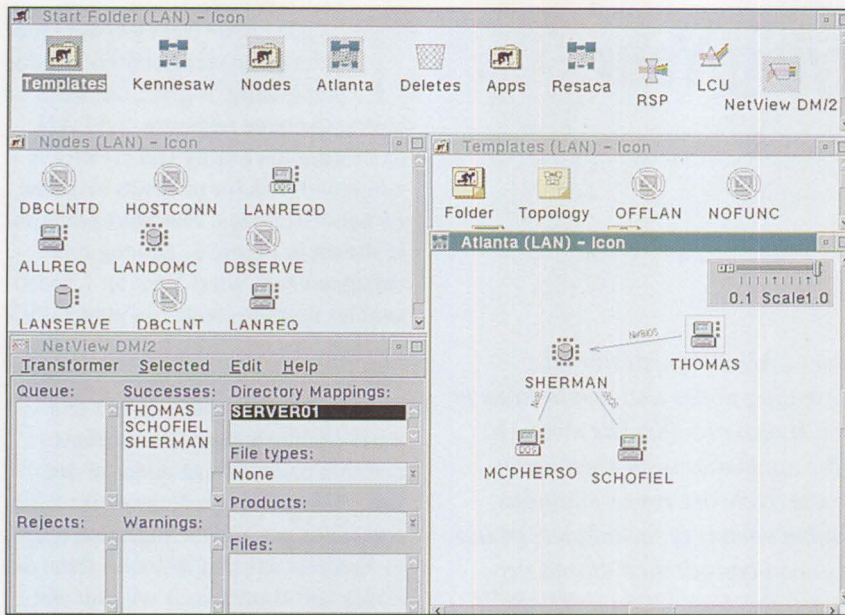


Figure 1. Example Start Topology

Creating Networks

Users can create graphical representations of their networks in several ways. One method is to create nodes by dragging and dropping template nodes from the Nodes folder into a topology, and changing the generated nodes' values in the attribute notebook for each workstation. Start contains predefined template nodes, such as a LAN server and LAN requester, that can be used to create specialized ones. However, using template nodes or duplicating existing nodes and changing their attributes (such as the adapter address) can take a considerable amount of time if the network is large.

With the Start discovery program (OCUNODE.EXE), migration import files can be generated for existing nodes with LAN, 3270, or database functionality, and these files can be stored in a server directory on a LAN. Individual files, a collection of files, or an entire directory containing migration files can be dragged and dropped onto a network topology. Start parses each file, creating nodes and connections as needed and

posting information messages into a migration log file.

The migration log file should be examined after each drop. This can be done by viewing the messages from the current drop in the migration log status window or by opening the OCUMERGE.LOG file with an editor. The messages reveal if a node was successfully migrated, if any information in the migration log file was rejected, and the number and types of connections created between nodes. Although the migration file does not explicitly include connections, Start can determine if a connection is needed by checking the adapter type and emulator function type. Start also checks the LAN function type for creating LAN connections. If necessary, Start creates host-node representations to create and complete a 3270 connection. Start does not automatically make database connections, although it does migrate the database functionality of a node from its import file.

The import file contains much of the configuration information already present and defined on the workstation.

This includes its Systems Network Architecture (SNA) workstation and LAN attributes, Distributed Function Terminal (DFT) and non-DFT emulator sessions, and adapters. These migration (import) files give users a convenient way to migrate their existing nodes into the Start network.

In addition to the other techniques, users can create an Attribute Value (ATV) file to quickly change information on nodes without opening the node settings notebook or rerunning the discovery program. This is done by creating a base set of nodes for a network or by working with an existing network. Although not all node attributes can be modified with an ATV file, this feature provides an easy way to change many common values used by nodes.

Creating an Attribute Value File

The ATV facility enables users to quickly and easily define selected values for one or more nodes in a topology. Figure 2 shows an example of an ATV file. This file has a malleable structure and is divided into sections: nodes, adapters, and DFT and non-DFT emulator sessions. These sections are further subdivided by the name of the section's table and its columns, as well as the values for the specified columns. The table and most of the column names correspond to ones in the Structured Query Language (SQL) and ASCII databases used by Start. ATV files enable users to selectively modify node values without opening and changing each node notebook or directly changing the values in the database.

Direct database changes are not recommended, but changes to the ATV files are recommended as shown in Figure 2. The Start program has a rigid format for both data and their ranges. Modifications to ASCII database files can lead easily to acciden-

```

.COMMENT
Attribute Value File for database NETWORK created by the AUTOATV.CMD Program.

.TABLE
NODES
.COLUMNS
NODENAME NODEID COMPNAME DBWORKSTATIONNAME DEFLOCALCP CPALIAS USERNAME PHONENUM MACHINELOC MACHTYPE MACHINENUM
MFGPLANT SERIALNUM WKSTCOMMENT
.VALUES
"MCPHERSO", "", "NODE0002", "NODE0002", "NET1.NODE0002", "NODE0002", "", "", "", "8570", "121", "00", "0000000", ""
"SCHOFIEL", "", "NODE0003", "NODE0003", "NET1.NODE0003", "NODE0003", "", "", "", "8570", "121", "00", "0000000", ""
"SHERMAN", "", "NODE0001", "NODE0001", "NET1.NODE0001", "NODE0001", "", "", "", "8580", "311", "00", "0000000", ""
"THOMAS", "000000", "NODE0004", "NODE0004", "NET1.NODE0004", "NODE0004", "", "", "", "8570", "121", "00", "0000000", ""

.TABLE
ADAPTERPARMS
.COLUMNS
NODENAME ADAPTERNUM ADAPTERTYPE LOCALSTATIONADD LOCALADMINADD CSMLANID
.VALUES
"MCPHERSO", 0, 5, "", "4000000000001", "CASPID"
"SCHOFIEL", 0, 5, "", "4000000000002", "CASPID"
"SHERMAN", 0, 5, "", "4000000000000", "CASPID"
"THOMAS", 0, 5, "", "4000000000003", "CASPID"

.TABLE
EMDFTPARMS
.COLUMNS
NODENAME SESSIONNUM SESSIONTYPE SESSIONNAME SESSSHORTID AUTOSTART
.VALUES
"THOMAS", 1, 1, "A", "A", 1
"THOMAS", 2, 2, "B", "B", 1

.TABLE
EMNONDFTPARMS
.COLUMNS
NODENAME SESSIONNUM SESSIONTYPE LUNAME SESSSHORTID LULOCALADD AUTOSTART
.VALUES
"THOMAS", 1, 1, "F", "F", "02", 1
"THOMAS", 2, 1, "G", "G", "03", 1

```

Figure 2. Sample ATV File

tal data loss, since the file structure is easier and perhaps more tempting to change than the SQL tables. Making any direct changes should be done with caution. Always keep a backup copy of the database.

The AUTOATV.CMD REXX program, shown in Figure 3, creates a master ATV file for all non-host nodes, adapters, and emulator sessions in the network SQL database. An enhanced version of this program is included as an applet with Start Version 1.1. The program in this article creates an ATV file by using the Database Manager command-line interface to export data rows from the NODES, ADAPTERPARMS, EMDFTPARMS, and EMNONDFTPARMS tables into separate files, and then reading those

files into the master ATV file. Every column, except one type, that can be modified by the ATV file is represented along with its true values in the database. Since the AUTOSTART column in the EMDFTPARMS and EMNONDFTPARMS sections of the ATV file modifies the STATUSBIT column in those respective tables, its values do not have a one-to-one match to a data value. Start determines if an emulator session can be automatically started by performing a bit-and operation on the contents of the STATUSBIT table with 512. If the result is not zero, the emulator session, whether DFT or non-DFT, can be automatically started.

Since a bit-and operation cannot be performed easily without further

manipulation of the export files, AUTOATV checks to see if the auto-start command-line parameter was specified when the program was invoked. If it is ON, all AUTOSTART values in the ATV file are set to on; if it is OFF, the values are set to off. If this parameter is not supplied, the AUTOSTART column is not included in the generated ATV file. In the ATV file in Figure 2, the AUTO-START values for the DFT and non-DFT sessions are set to ON (1).

It would not make sense to immediately drop the master ATV file generated by the program on the network topology, since only the AUTOSTART value for the emulator sessions could change. Instead, use the master ATV file as a base to

```

/* AUTOATV-Create an ATV file for an IBM LAN NetView Start 1.1 SQL database */
/* (c) Copyright IBM Corp., 1993 */
/* Licensed Materials-Property of IBM */
/* All Rights Reserved */
parse upper arg dbname autostart rest

if ((dbname = '?') | (dbname = '')) then
do
  say 'AUTOATV.CMD'
  say '- Creates a master Attribute Value File (ATV) for a Start 1.1 SQL database.'
  say ''
  say 'AUTOATV sql-database autostart'
  say 'AUTOATV NETWORK [OFF | ON | NONE ]'
  say ''
  say 'The sql-database parameter is required. The autostart parameter is optional.'
  say 'If OFF or ON, the AUTOSTART columns in the EMDFTPARMS and EMNONDFTPARMS'
  say 'tables will default to those column values respectively. If NONE, which is the'
  say 'default if this parameter is not specified, the AUTOSTART column will not'
  say 'be included in the ATV file for either table.'
  exit
end

autostart = translate(autostart)
if \((autostart = 'OFF') | (autostart = 'ON')) then
  autostart = 'NONE'
say 'Autostart parameter is ' || autostart || '.'

do i = 1 to 4
  delfilename.i = 'ATV' || i || '.DEL'
  msgfilename.i = 'ATV' || i || '.MSG'
end /* do */

table.1 = 'NODES'
table.2 = 'ADAPTERPARMS'
table.3 = 'EMDFTPARMS'
table.4 = 'EMNONDFTPARMS'
columns.1 = 'NODENAME NODEID COMPNAME DBWORKSTATIONNAME DEFLOCALCP CPALIAS USERNAME PHONENUM
MACHINELOC MACHTYPE MACHINENUM MFGPLAT SERIAL NUM WKSTCOMMENT'
columns.2 = 'NODENAME ADAPTERNUM ADAPTERTYPE LOCALSTATIONADD LOCALADMINADD CSMLANID'
columns.3 = 'NODENAME SESSIONNUM SESSIONTYPE SESSIONNAME SESSSHORTID'
columns.4 = 'NODENAME SESSIONNUM SESSIONTYPE LUNAME SESSSHORTID LULOCALADD'
emdftselect = "select nodename, sessionnum, sessiontype, sessionname, sessshortid "
emnondftselect = "select nodename, sessionnum, sessiontype, luname, sessshortid, lulocaladd "
orderby = "order by nodename"

/* Add autostart value, if any, to queries. */
if ((autostart = 'OFF') | (autostart = 'ON')) then
do
  columns.3 = columns.3 || ' AUTOSTART'
  columns.4 = columns.4 || ' AUTOSTART'
  if (autostart = 'OFF') then
    autovalue = 0
  else
    autovalue = 1
  emdftselect = emdftselect || ', ' || autovalue || ' '
  emnondftselect = emnondftselect || ', ' || autovalue || ' '
end

say 'Deleting old ATV*.DEL and ATV*.MSG files.'
do i = 1 to 4
  'erase ' || delfilename.i
  'erase ' || msgfilename.i
end

```

Figure 3. The AUTOATV.CMD Program (Continued)


```

call dbm 'STARTDBM'
say 'Exporting tables.'
call dbm "EXPORT FROM " || dbname || " TO ATV1.DEL OF DEL MESSAGES ATV1.MSG "||,
"select nodename, nodeid, compname, dbworkstationname, deflocalcp, cpalias, "|| ,
"username, phonenum, machineloc, machtype, machinenum, mfgplant, " || ,
"serialnum, wkstcomment from nodes where nodefunction = 1 " || orderby

call dbm "EXPORT FROM " || dbname || " TO ATV2.DEL OF DEL MESSAGES ATV2.MSG "||,
"select nodename, adapternum, adaptertype, localstationadd, localadminadd, csmlanid " || ,
"from nodes, adapterparms where nodes.objectid = adapterparms.ownerobjectid "|| orderby

call dbm "EXPORT FROM " || dbname || " TO ATV3.DEL OF DEL MESSAGES ATV3.MSG " ||,
emdftselect || "from nodes, emdftparms where nodes.objectid = emdftparms.ownerobjectid " ||,
orderby

call dbm "EXPORT FROM " || dbname || " TO ATV4.DEL OF DEL MESSAGES ATV4.MSG " ||,
emnondftselect || ,
"from nodes, emnondftparms where nodes.objectid = emnondftparms.ownerobjectid" || orderby

outfile = 'AUTOATV.ATV'
'erase ' || outfile
say 'Creating AUTOATV.ATV file.'
rc = lineout(outfile, '.COMMENT')
rc = lineout(outfile, 'Attribute Value File for database ' || dbname ||
'created by the AUTOATV.CMD Program.')
rc = lineout(outfile, ' ')

do i = 1 to 4
rc = lineout(outfile, '.TABLE')
rc = lineout(outfile, table.i)
rc = lineout(outfile, '.COLUMNS')
rc = lineout(outfile, columns.i)
rc = lineout(outfile, '.VALUES')
do until lines(delfilename.i) = 0
inline = linein(delfilename.i)
rc = lineout(outfile, inline)
end
rc = lineout(outfile, ' ')
end

say 'AUTOATV.ATV file generated.'
say 'Modify and drag and drop on your desired Start topology.'

```

Figure 3. The AUTOATV.CMD Program

customize the network. With a regular editor, users can selectively change adapter and other values and rename nodes. Then the modified ATV file can be dropped on the topology to update the network. Any information and warning messages will appear in the ATV status window.

Although AUTOATV only works on a network stored in an SQL database, it can be rewritten to include parsing routines to create a master ATV file for ASCII databases. Alternatively, a separate utility can be used to trans-

fer the ASCII database files into an SQL database before AUTOATV is run (one such utility also is included as an applet with Start Version 1.1). The AUTOATV program can be extended to include more error checking. It also can be rewritten to include bit-and functions to determine directly if a defined DFT or non-DFT session can be automatically started. However, one advantage of writing it in REXX is the language's ease of maintenance and modification, and its integration into the base OS/2 system.

AUTOATV can be refined further to create separate ATV files for each topology. The master ATV file, generated by AUTOATV, includes nodes from all topologies in the network. If this file is changed and then dropped onto a specific topology, values in the ATV file that relate to nodes outside the target topology are ignored. This program also can create a single ATV file per node by adding another parameter to the command line and appending another Where clause to each Select statement in the export; for example, and nodename='SHERMAN'.

NODES Table	2 = OS/2 1.3
EMTYPE Column	3 = DOS
0 = Uninitialized (such as no connection established)	4 = DOS with Windows
1 = DFT	SRVFILESYSYSTEM Column
2 = LAN	1 = 386 Exclusive
3 = DFT and LAN	2 = Other
4 = SDLC	3 = 386 Mix
5 = DFT and SDLC	ADAPTERPARMS Table
6 = LAN and SDLC	ADAPTERTYPE Column
7 = DFT, LAN, and SDLC	1 = Token Ring I
16 = Off-LAN or Host	2 = Token Ring II
32 = No function	3 = Token Ring /A
DBTYPE Column	4 = Token Ring 16/4
1 = Server	5 = Token Ring 16/4 /A
2 = Requester	6 = Token Ring Busmaster
3 = Requester/Server	7 = IBM PS/2 Ethernet™ Adapter
4 = DOS Requester	8 = 3Com® 3C503 EtherLink II®
16 = Local	9 = 3Com 3C523 EtherLink MC
LANTYPE Column	10 = Ungermann-Bass NIUpc
1 = Server	11 = Ungermann-Bass NIUps
2 = Domain	12 = Western Digital Ethercard PLUS AT
4 = Requester	13 = Western Digital Ethercard PLUS MC
8 = DOS	15 = SDLC
16 = None	24 = DFT
OSVERSION Column	LINETYPE Column
0 = None or Host	1 = Switched
1 = OS/2 2.0	2 = Non-Switched

Figure 4. NODES and ADAPTERPARMS Tables

```
select nodename from nodes, adapterparms
where nodes.objectid = adapterparms.ownerobjectid
and osversion = 1 and lantype < 16 and
adaptertype >= 1 and adaptertype <= 7
```

Figure 5. Query to Find OS/2 2.0 Nodes with Token-Ring Adapters and LAN Functions

The Select statement for the NODES table already has one Where clause to distinguish between regular nodes with nodefunction=1 and host nodes with nodefunction=0. The Where clause also can include the nodes.ageflag=0 statement to exclude aged nodes from the ATV file. (Aged objects are generated for a node and its objects, such as adapters and emulator sessions, when a node is dropped on a transformer.

Aged objects contain the "old" values of an object, and by comparing the old and new values, Start can generate accurate, updated response files.)

Running Queries

Although not as fast as using the ASCII database manager, invoking Start with the Extended Services Database Manager provides the bonus of being able to run SQL queries

against the database to generate reports of values within the network. Many tables have columns that can easily be related to their corresponding values in the Start user interface, such as nodename in the NODES table and the node name under the node icon in a topology container. Other values, however, such as the adapter type, are not apparent without careful examination of the database and some experimentation. The NODES and ADAPTERPARMS tables in Figure 4 provide information on some popular columns to query. Not all these values may be used, and they are subject to change in future releases of the program. However, they can be used in queries against SQL databases that are used in Start releases 1.0 and 1.1.

For example, the query in Figure 5 can be run to find all the OS/2 2.0 nodes with Token-Ring adapters and some type of LAN function.

Specialized queries also can be integrated into the AUTOATV program to further refine the ATV files generated by the program.

References

For an overview of the IBM LAN NetView Start program, see "Start/2: Putting the Configuration into CID" in the January 1993 issue of *IBM Personal Systems Technical Solutions*.

Theodore Shrader is a senior associate programmer working on the LAN NetView Start program in the IBM Personal Systems Line of Business in Austin, Texas. He joined IBM in 1989 and has worked on OS/2 Database Manager and LAN graphical application development. Theodore has a BS in computer science from the University of Texas at El Paso.

Multimedia Applications on IBM Token-Ring LANs

Khaled Amer, Ken Christensen, and Tom Toher
IBM Corporation
Research Triangle Park, North Carolina

This article describes the capabilities of IBM 16 Mbits-per-second Token-Ring products to support multimedia applications—simultaneously carrying data and multimedia traffic. IBM Token-Ring's priority mechanism makes it possible to implement bandwidth management in multimedia servers, making client/server multimedia applications on a single LAN segment a reality.

Sharing, distributing, and collaborating through multimedia significantly contributes to the improved efficiency and quality of communications. Multimedia-enabled solutions use a more natural way of communicating, which increases the effectiveness of communications. This is done by having many media types—text, image, sound, and video—combined and presented simultaneously. Future systems will search, edit, and store audio and video recordings, interactively control media presentations, and share multimedia-based information with others.

Multimedia Applications

This article focuses on client/server multimedia applications where client workstations interact with multimedia information on a server over a single IBM 16 Mbits-per-second Token-Ring segment. Other types of multimedia applications include the following:

- **Local applications:** A workstation interacting with multimedia information that is locally stored and processed
- **Person-to-person communications:** Real-time displays of people and their visual and audio materials

- **Video conferencing:** An extension of person-to-person communications to include groups of geographically dispersed participants

Data and Multimedia Traffic on the LAN

Data and multimedia traffic place different requirements on transmission capabilities of networks. Typically, data traffic does not have to be delivered within strict time bounds to preserve its meaning. However, multimedia traffic depends on the simultaneous delivery of its constituent parts. For example, in full-motion video, voice and image must be delivered at a natural speed and synchronized with each other. Such information is typically classified as *real-time* or *time-constrained traffic*. Figure 1 shows a Token-Ring LAN carrying both data and multimedia traffic originating from data and multimedia servers.

Each active multimedia session requires the delivery of multimedia objects (audio and video) at a specified bounded rate for proper perception. Therefore, multimedia objects are time-based, continuous data streams. At the high end of bandwidth requirements for Digital Video Interactive (DVI[®]) applications, full-motion video with Production Level

Video (PLV) quality uses the CD-ROM rate of 150 KB per second (1.2 Mbits per second). This enables the viewing of 30 video frames per second at a resolution of 512 pels x 480 pels full color, with 24 bits per pel. Other applications can use Real-Time Video (RTV), which has a bandwidth range from 288 Kbits per second up to 1.2 Mbits per second. This gives 15 frames per second at lower resolution (128 pels x 120 pels, 9 bits per pel). If the LAN or another component in the system cannot sustain that bandwidth, the viewer experiences pauses in the audio and video.

Data traffic has flexible bandwidth requirements. As available bandwidth changes because of other traffic in the network, it simply changes the time for transferring data. Data traffic on LANs also is bursty; that is, occasional peaks on the LAN occur as users perform functions such as transferring files between a workstation and a server.

The Token-Ring access method, described in the IEEE 802.5 standard, is based on using a token that circulates around the ring. When all stations are idle, the token circulates unused on the ring. A station wishing to transmit must wait until it detects a token passing by. The station changes the token to a frame by transmitting the frame in place of the token. Other stations wishing to transmit must wait. The frame on the ring makes a round trip and is purged by the transmitting station. The transmitting station inserts a new token on the ring immediately following the frame it transmitted, if the Early Token Release function is on. Otherwise, a new token is released only when the transmitting station has received the header of its own frame.

Token-Ring architecture contains many features well suited to multimedia traffic. Its principal strength is the predictable nature of a station's

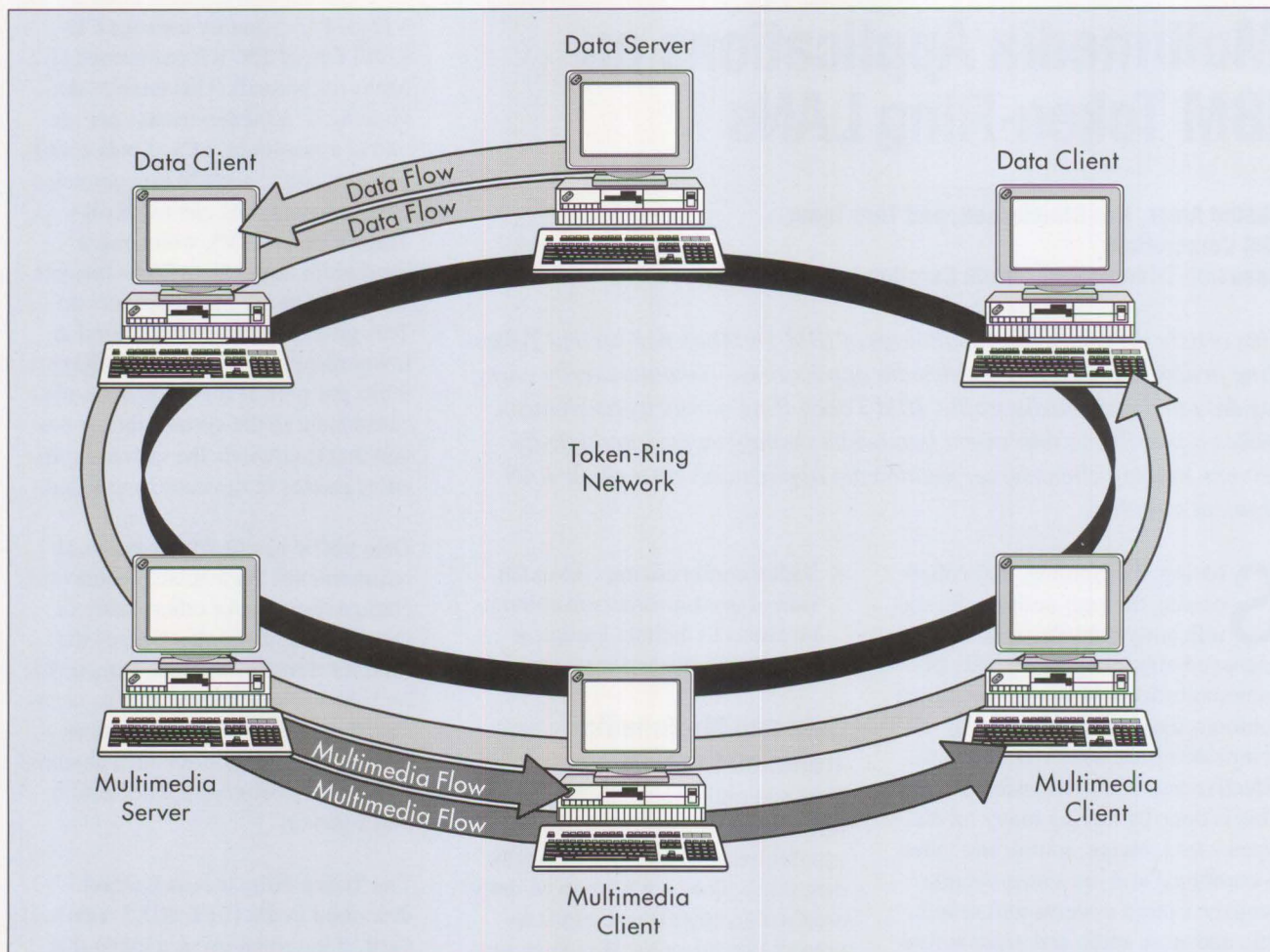


Figure 1. Simultaneous Video and Data Transmission on a Single-Segment IBM Token-Ring Network

opportunity to transmit. The worst case or longest time that a station must wait for a token is deterministic and can be calculated. This is unlike non-token-passing LANs, such as Ethernet, which cannot guarantee an upper bound on the delay encountered by a station before it can transmit.

Another advantage is that the traffic on the ring can be regulated with priority. Higher priority stations can use a lower priority token immediately or reserve a priority token and get first chance to transmit data ahead of lower priority stations. This is especially useful in LANs with a combination of multimedia and data traffic. Since multimedia traffic can tolerate only a small delay, it is

desirable to give it higher priority than data traffic.

Two capabilities are needed to transport bandwidth-constrained traffic on a LAN segment:

- **A means for guaranteeing bandwidth.** This can be done by limiting the number of active multimedia sessions on a Token Ring. This should enable the LAN segment to transport both data and bandwidth-constrained multimedia traffic.
- **An automated means for requesting and reserving bandwidth.** If a multimedia session cannot obtain guaranteed bandwidth, the session request will be refused;

otherwise, the quality of the video and audio suffers.

A multimedia session is active when a client workstation is displaying a video clip sent from a multimedia server. This article presents some general recommendations for determining how many multimedia sessions can be active simultaneously on a single-segment Token Ring. Although the article focuses on PLV full-motion video playback applications, a single-segment Token Ring can support other types of applications such as RTV, Joint Photographic Experts Group (JPEG), and Motion Picture Experts Group (MPEG) that make lesser bandwidth (needed by RTV) or greater bandwidth (for

JPEG and MPEG) demands on the network.

Multimedia Traffic on the Token Ring

A client playing full-motion audio and video with CD-ROM quality from a server on the Token Ring uses an average delivery rate of 150 KB per second or 1.2 Mbits per second, giving 30 video frames per second. Therefore, each client on the LAN must receive its data at this average rate to maintain full motion. Lab experiments show that up to thirteen active CD-ROM video playback sessions can run simultaneously on a single, dedicated, 16 Mbits-per-second Token-Ring LAN with files optimized for network playback. Generally, twelve sessions can be viewed as a practical maximum. Many 16 Mbits-per-second Token-Ring LANs with data applications run at a low average utilization; 10% to 20% is typical. Of course, there are peak periods when data traffic utilizations can reach 30% or higher for short periods of time. Still, many data LANs can handle added multimedia applications.

Users versus the Number of Active Sessions

The total number of active sessions is not necessarily the same as the total number of multimedia users that can be supported on a LAN. Typically, not all users want to have an active multimedia session simultaneously. This is analogous to the telephone system. There are far fewer telephone lines than telephones, yet telephone users rarely have to wait to make a call. A statistical analysis can be used to determine how many telephone lines are needed for a given grade of service. Grade of service is measured by the wait time to get a dial tone to make a telephone call. For client/server multimedia, grade of service can be measured similarly: the length of time a user must wait to get an active

IBM PS/2 ActionMedia II

IBM PS/2 ActionMedia® II is based on a revolutionary digital compression and decompression technology called Digital Video Interactive (DVI). Developed by IBM and Intel Corporation, ActionMedia II is the result of work dating to the early 1980s. DVI was made available to the general market in January 1992 with IBM PS/2 ActionMedia II.

IBM PS/2 ActionMedia II display adapters can provide up to 30 frames per second of full-motion video, high-quality audio, and images from several sources. These sources include local PS/2 digital storage devices, attached digital networks, or analog video (from VCRs, video disk players, broadcast TV, and so on) digitized in real-time by the Capture option. Multimedia content can be combined on-screen with text and graphics generated by a PS/2 computer. This capability enables multimedia to augment today's applications if they are extended to support ActionMedia II.

Audio, video, and still image content must first be compressed by

the display adapter when data streams are managed over a network or stored on disk. The Capture option digitizes analog input signals conforming to National Television Systems Committee (NTSC) or Phase Alternating Line (PAL) standards. This enables the signals to be monitored directly on the PS/2 display screen, captured as still frames, or compressed in real-time using the DVI technology Real-Time Video (RTV) capability. A digital signal processor handles audio compression.

Although RTV compression provides adequate display quality for many applications, commercial and critical custom applications can be produced through Production Level Video (PLV) compression. This is handled off-site by Intel-licensed vendors. PLV compression provides video images that are VCR quality or better.

DVI provides adequate video resolution using modest communications bandwidth. The trend in the industry toward improved compression algorithms should further reduce bandwidth demand in the future.

session to see a video after requesting it from a server.

Figure 2 shows how many users can be supported on a LAN for a given session wait time when bandwidth reservation techniques are used. The *session wait time* is the average time between a user requesting to view a video clip and the moment a guaranteed bandwidth multimedia session is established allowing the video clip to begin to play on the client workstation. This depends on the number of available, or unused, active sessions at any time. Each curve corresponds

to a certain frequency of video clip requests for each user. This is expressed by the mean time between requests ranging from 20 to 50 minutes. The following assumptions are used in Figure 2:

- Requests arrive randomly and can wait to be assigned a session. For example, a client application repeatedly requests active session establishment when a user requests to view a video clip.
- The duration of the video clip also is random with a mean of three minutes.

Network Transport System/2

IBM is developing a multimedia client/server option for the IBM LAN Server 3.0 running on IBM 16 Mbits-per-second Token Rings. Network Transport System/2 will provide reliable and predictable playback of multimedia files on a single-segment Token-Ring network by reserving resources, such as bandwidth, to ensure high-quality service.

The server will run a fixed number of active sessions of a particular application on a LAN segment. The server also will keep track of the number of concurrent active sessions. If a new session is requested that will exceed the maximum amount of available bandwidth, it will jeopardize the quality of service for other active sessions. Therefore, the server will refuse to establish this session. This will guarantee that network overloading is avoided, ensuring the quality of all multimedia sessions.

LAN Server 3.0 implements quality-of-service parameters for multimedia files in extended file attributes. When a multimedia file is opened for playback, the redirector and the transport can query the extended file attributes to determine bandwidth requirements for the file. The Token-Ring segment is configured to reserve an arbitrary portion of its bandwidth for multimedia traffic (up to the maximum available bandwidth for Token Ring). The bandwidth manager portion of the transport ensures that each multimedia session stays within the bounds specified in the extended file attributes. It also ensures that the aggregate bandwidth of all multimedia sessions stays within the configured limits. To guarantee the timely delivery of multimedia data to clients without interference from data traffic, all bandwidth-managed multimedia sessions have their data sent at high priority to the clients. Network Transport System/2 was beta tested during the first quarter of 1993.

The chart in Figure 2 shows that a user on a LAN with 50 others, who each request a video clip every 30 minutes, must wait an average of less than two seconds to establish a guaranteed bandwidth multimedia session.

Mixing Multimedia and Data Traffic

Installed LANs already are carrying data traffic, and the next sections show how this data traffic would be affected by adding multimedia sessions.

A lab test was run to show the effect of adding multimedia sessions on the data throughput with and without multimedia running at higher priority. It also showed the effect of the priority mechanism on preserving the integrity of multimedia sessions on a LAN carrying both multimedia and data traffic. The following setup was used:

- Two OS/2 LAN servers (33 MHz PS/2 Model M95-0KF) for multimedia traffic with IBM busmaster 10/7 Token-Ring adapters
- One OS/2 LAN server (25 MHz PS/2 Model M70-B21) for data traffic with IBM busmaster 10/7 Token-Ring adapters
- Up to 13 multimedia clients (various models of PS/2s) with IBM 16/4 Token-Ring adapters

Benchmark Results

Figure 3 shows lab measurements of the time to transfer a file from an OS/2 LAN server across the Token-Ring to a client. This represents a typical client/server data application. Figure 3 also shows the effect on the data throughput of adding multimedia video playback sessions across the LAN without increasing the multimedia priority. Later we show the effect of running multimedia at a higher priority.

First, a file was transferred from a data server to a data client on a dedi-

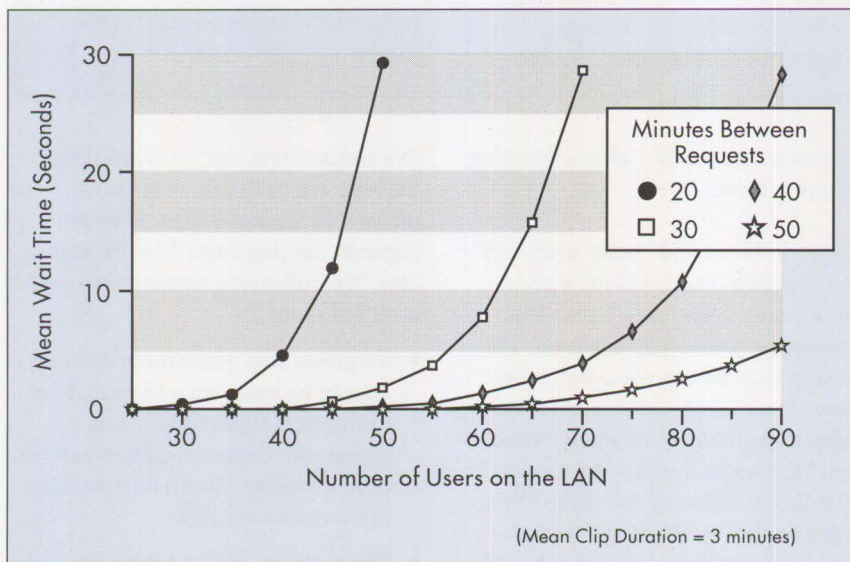


Figure 2. Mean Session Wait Time

cated Token Ring, and the data transfer time was recorded. This was performed for three file sizes:

- 2 KB, representing a small "mail" file. It used about 0.48 Mbits per

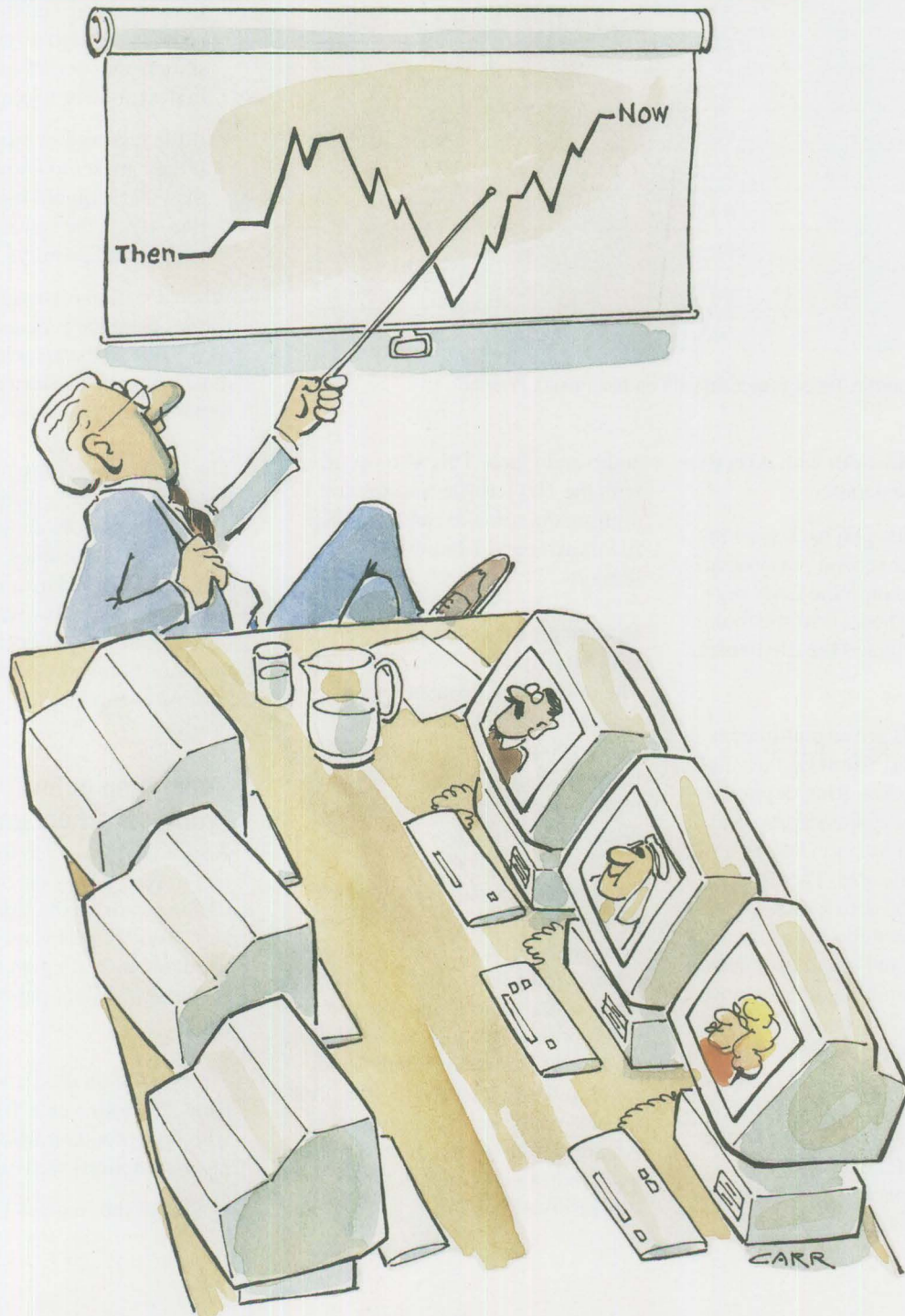
second of the available bandwidth with no active multimedia sessions.

- 100 KB, representing a spreadsheet or word processing file. It used about 3.2 Mbits per second

of the available bandwidth with no active multimedia sessions.

- 1 MB, representing a binary executable or graphics file. It used about 5.3 Mbits per second of the

INTERNATIONAL SALES CONFERENCE



Number of Multimedia Sessions	2 KB File	100 KB File	1 MB File
0	0.11	0.27	1.73
1	0.11	0.27	1.74
2	0.11	0.28	1.77
3	0.11	0.28	1.80
4	0.11	0.28	1.81
5	0.11	0.29	1.85
6	0.11	0.29	1.89
7	0.12	0.30	1.95
8	0.12	0.31	2.02
9	0.12	0.31	2.11
10	0.12	0.32	2.36
11	0.12	0.34	N/A
12	0.13	N/A	N/A
13	N/A	N/A	N/A

Figure 3. File Transfer Times (seconds) with no Multimedia Priority

available bandwidth with no active multimedia sessions.

Multimedia video playback sessions (at 1.2 Mbits per second per session) were introduced onto the LAN segment. Figure 3 shows how the multimedia traffic affected the file transfer time.

Adding up to 10 active multimedia video playback sessions on the small experimental Token Ring degraded the throughput experienced by the data application by only 20% to 35%, depending on file size. The difference between the data transfer rate of a 1 MB file with no active multimedia sessions and the transfer rate with 10 active multimedia sessions was less than 0.7 seconds. This would be barely perceptible to users. As the multimedia sessions increased to 11, the bandwidth used for both multimedia and data transfer for the 1 MB file transfer exceeded the Token-Ring capacity of 16 Mbits per second, resulting in pauses in the

audio and video. This also occurred with the 100 KB file transfer and 12 multimedia sessions, and the 2 KB file transfer and 13 multimedia sessions.

Multimedia on Token Ring with Priority

The priority mechanism feature makes the IBM Token Ring suitable for carrying multimedia traffic. This is especially useful for LANs with high mean utilization or frequent peaks of data traffic.

The priority mechanism provides equal access to the ring for all stations within a priority level. Placing priority indicators within the token's access control field achieves the token priority-passing approach. The priority of a token or frame is indicated in the first three bits (the priority bits) of the access control field. Any reservation for a different priority is indicated in the last three bits (the reservation bits).

The priority operation works as follows: a ring can transmit a frame at a given priority, using any available token with a priority less than or equal to that of the frame. If an appropriate token is not available, the ring station can reserve a token of the required priority in a passing token or frame as follows.

- If another ring station has reserved an equal or higher priority, the ring station cannot make a reservation in the frame or token.
- If the reservation bits are not set or if they are set to a lower priority than that required by the ring station, it sets the reservation bits to its required priority.

When a station removes one of its frames from the ring and finds a non-zero value in the reservation bits, it must originate a priority token equal to the reservation bits.

The IBM Token Ring provides "fairness" within each priority to prevent a ring station from continually transmitting priority frames. A priority can be preempted by a request for a higher priority. However, once the highest priority has been satisfied, the priority reverts to the lower priority, and eventually to the normal priority.

Simulation of the Priority Mechanism

A simulation was run to show the effect of the priority mechanism when running multimedia sessions. The simulation results were subsequently validated by testing on a single-segment 16 Mbits-per-second Token Ring.

The simulation used a 4 KB Token-Ring frame size on a Token Ring with background traffic. The following assumptions were used:

- Size of the ring = 4 kilometers

New High-Performance Adapter Technology

IBM intends to announce a high-performance 32-bit busmaster 16/4 Token-Ring adapter for Micro Channel that supports streaming data mode. This adapter will provide outstanding performance, especially in data servers and multimedia servers. The features listed below are implemented using new IBM Token-Ring technology that can provide performance approaching the theoretical maximum.

- **High-performance architecture:** IBM's new adapter technology provides the high performance needed for multimedia servers. This includes high throughput capacity (over 30,000 frames per second) and low latency.
- **Busmaster interface with 32-bit data width and 32-bit addressing and streaming data support:** This makes efficient use of the Micro Channel with a burst capacity of 40 MB per second. The 32-bit addressing enables the adapter to access 4 GB of memory directly.
- **Prioritization of frames:** The new adapter architecture provides mechanisms for prioritizing frames. This feature is especially valuable for multimedia servers running at high priority.
- **On-card support for shielded and unshielded twisted-pair cabling:** The card directly supports both shielded and unshielded twisted-pair cabling.

- Number of stations = 40
- Token-Ring data rate = 16 Mbits per second
- Background traffic (30% of ring capacity) consisting of an equal number of large frames (2 KB) and small frames (67 bytes)

Figures 4 and 5 show the delay encountered by the multimedia frames on the ring (token wait time plus transmission time on the ring). This is shown for various loads on the ring (representing data traffic), both with and without the priority mechanism. Figure 4 shows the mean delay, and Figure 5 shows the 99th percentile delay. The simulation results show that although the Token-Ring delay is under control in rings with low utilization, priority must be invoked to control the delay in rings with fairly high utilization.

Effects of the Priority Mechanism

With the data and multimedia traffic at the same priority (0), the multimedia sessions experienced pauses

and resulted in pauses in the audio and video.

Benchmarks of the Priority Mechanism

The experiment described in the Benchmark Results section was repeated several times with the multimedia servers running at a higher priority (3) than both the data server and the client stations. As the bandwidth of both the data and multimedia exceeded the Token-Ring capacity of 16 Mbits per second, the priority mechanism narrowed the bandwidth available to data traffic. This enabled the multimedia sessions to run unimpaired. Predictably, the data traffic rate was reduced.

This experiment shows how data traffic bursts exceeding the Token-Ring capacity do not affect multimedia sessions running at high priority. This verifies the benefit of using the priority mechanism to control the delay experienced by multimedia frames. Figure 6 shows the results.

Additional Information

Token-Ring Material

- *IBM Local Area Network Technical Reference (SK2T-0300)*

in the audio and video as the number of active multimedia playback sessions increased. When the bandwidth demand exceeded the network's capacity of 16 Mbits per second, frames queued up for LAN access. This slowed the multimedia traffic to a lower rate than that required for timely delivery of the video frames

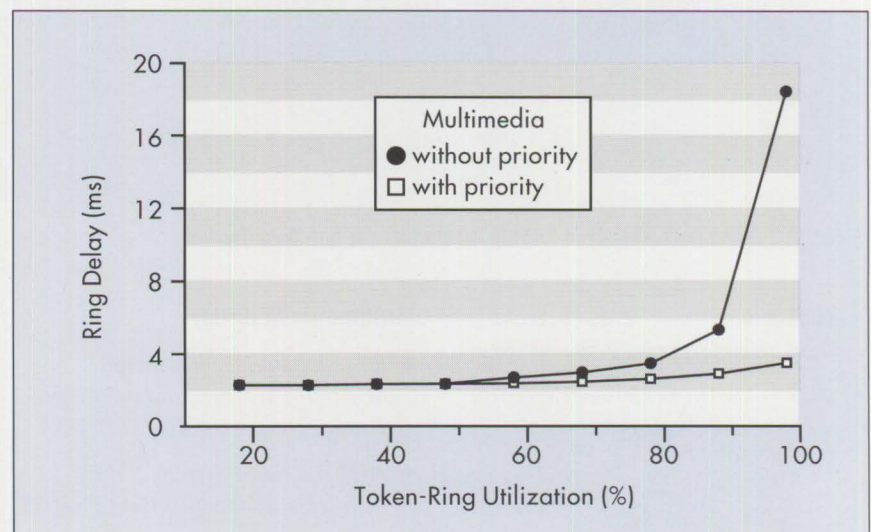


Figure 4. Simulation Results for Multimedia Mean Delays

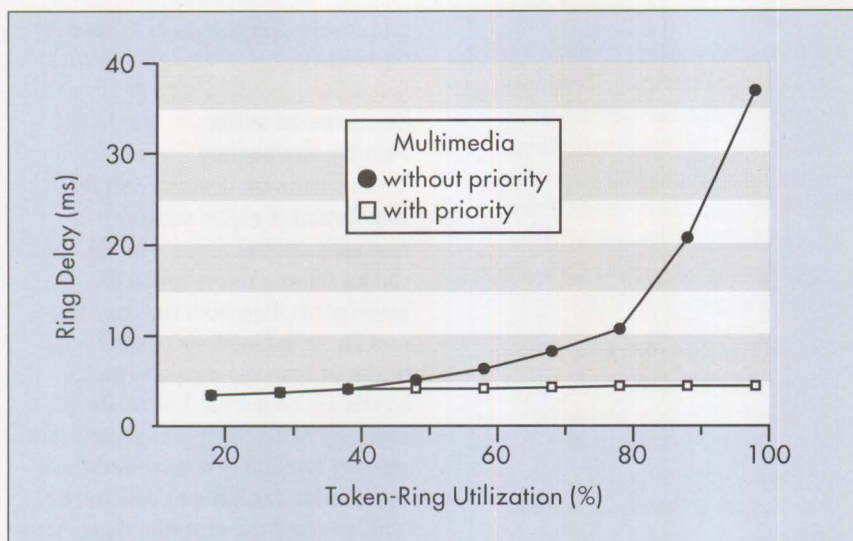


Figure 5. Simulation Results for Multimedia 99th Percentile Delays

Number of Multimedia Sessions	2 KB File	100 KB File	1 MB File
0	0.11	0.27	1.73
1	0.11	0.27	1.76
2	0.11	0.28	1.77
3	0.11	0.28	1.80
4	0.11	0.28	1.85
5	0.11	0.29	1.89
6	0.12	0.30	1.98
7	0.12	0.31	2.06
8	0.12	0.32	2.20
9	0.12	0.35	2.56
10	0.13	0.37	2.85
11	0.13	0.42	3.32
12	0.15	0.52	4.47
13	0.17	0.54	4.54

Figure 6. Effect (in seconds) of Priority on Multimedia and Data Traffic

- *IBM Token-Ring Network Architectural Reference* (SC30-3383)
- *IBM Token-Ring Network Introduction and Planning Guide* (GA27-3677)

Multimedia Material

- *Digital Audio Fundamentals, M-Audio Adapter, and the Audio Device Driver* (GG24-3795)
- *IBM Personal System/2 Multimedia Fundamentals* (GG24-3653)

- Luther, Arch C. *Digital Video in the PC Environment*. Intel/McGraw-Hill, 1991.
- *Multimedia Application Enablers and the IBM PS/2 Ultimedia Systems* (GG24-3749)
- Nusbickel, Wendi and Albee, John. "Networked Full-Motion Digital Video," *OS/2 Developer Magazine*, Fall 1992
- *Ultimedia Family of PS/2 Systems* (G520-6963)
- *Ultimedia Overview Brochure* (G229-7324)

For further information, call the IBM Multimedia Information Center at (800) 426-9402.

Khaled Amer is currently working on LAN systems for IBM in Research Triangle Park, North Carolina. He joined IBM in 1986 as a semiconductor process engineer at the East Fishkill site. He transferred to the Kingston Numerically Intensive Computing Center in 1988 to work on distributed applications and LAN performance. Khaled received an MS in electrical engineering from The California Institute of Technology and is currently working on a PhD in network performance analysis at North Carolina State University.

Ken Christensen is an engineer in IBM's LAN Advanced Development department at Research Triangle Park in North Carolina. Ken has been involved with Token-Ring development and testing since joining IBM in 1983.

Tom Toher is an information developer at IBM's Research Triangle Park Laboratory in North Carolina. Tom has been involved in local area network and cabling projects since joining IBM in 1984.

OS/2 2.0 Print Tips

Richard R. Miller
IBM Corporation
Raleigh, North Carolina

This article provides step-by-step instructions for using printers with OS/2 2.0. It also includes a procedure that eliminates the need for all users to have access to the five OS/2 2.0 Printer Driver diskettes. Sample printer configurations are suggested to maximize the number of users that can access a printer.

The information in this article was compiled to assist novice users in printing with OS/2 2.0. It describes one of many procedures that can be used to access or configure a printer on an OS/2 2.0 workstation or LAN print server. Since procedures may change over time, refer to the OS/2 2.0 documentation to resolve any conflicts or changes in operation. Corrective Service Disks (CSDs) also may change the procedures or infor-

mation needed to configure and use a printer.

Before attempting to use these procedures, you should be familiar with the basic operation of OS/2 2.0 and OS/2 2.0 objects.

Adding a Printer

This article provides information to help you add, delete, or modify printer objects on an OS/2 2.0 work-

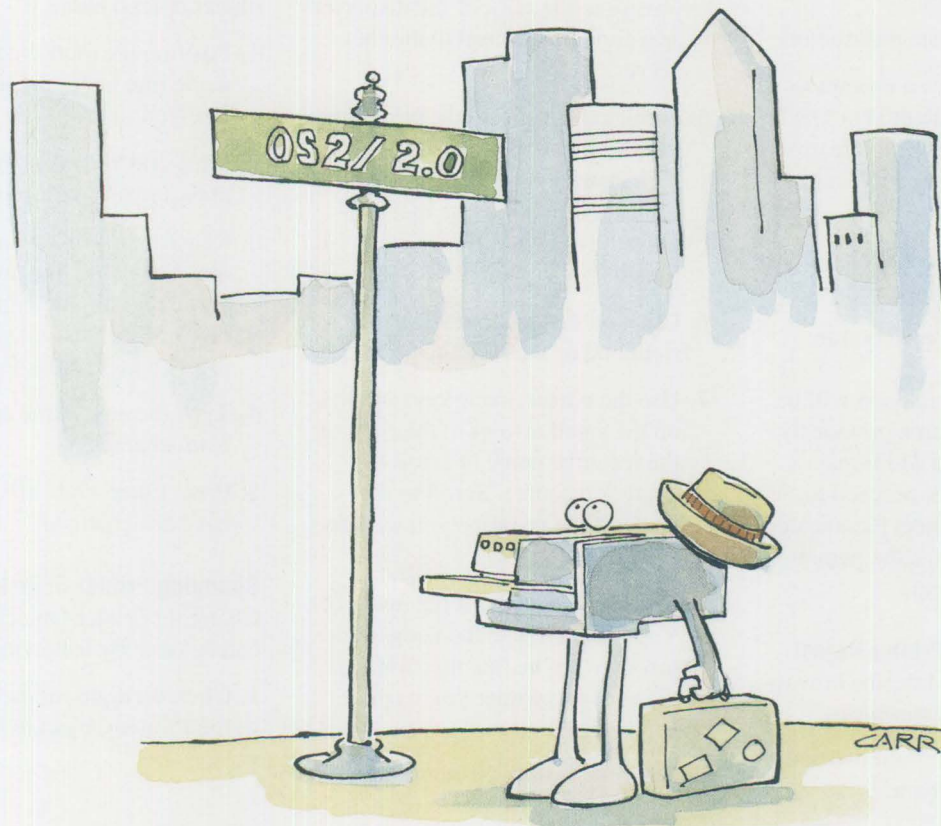
station. It was accurate as of December 1992; however, future changes to the OS/2 2.0 operating system or local procedures may invalidate these suggestions. If a problem arises, contact your local support personnel to obtain the correct procedure for your location.

In this article, a *printer object* defines a printer attached directly to a workstation, and a *network printer object* defines a printer on a LAN.

OS/2 2.0 Printer Driver diskettes are required when a new printer driver is to be installed. Alternatively, if your local LAN coordinator has created the PDRIVERx alias on an OS/2 file server, you can install printer drivers using the alias in place of the diskettes when asked for the path (default is A:\).

Creating a New Printer Object

Follow the steps below to create a new printer object.



1. Double-click the left mouse button on the Templates icon.
2. When the Templates window is displayed, position the mouse pointer on the Printer icon. Be sure to select Printer, not Network Printer.
3. Press and hold the right mouse button. Drag the Printer icon to an open area of the Desktop or an open folder, and release the mouse button.
4. When the Create A Printer window appears, over-type the word Printer with a meaningful name, such as a printer location or user group name.
5. Select the printer driver for the printer type you are creating. Refer to "Installing OS/2 2.0 Printer Drivers" in this article if the new printer type or the printer driver needed is not shown.
6. Select the Output Port where the printer is attached.
7. Click on the Create pushbutton.
8. When you receive a prompt to install the equivalent Win-OS/2 printer, click on Yes to create or No if you do not need Windows-OS/2 printers.
9. If yes, insert the correct OS/2 2.0 Printer Driver diskette.
10. Click on the OK pushbutton.

The new Printer Object icon will be displayed with the name previously entered. You can add additional printers at this time or proceed to "Changing Printer Object Parameters" in this article to change the parameters of a printer object.

Creating a Network Printer Object

The following steps describe how to create a network printer object.

1. Double-click the left mouse button on the Templates icon.

2. When the Templates window is displayed, position the mouse pointer on the Network Printer icon.
3. Press and hold the right mouse button. Drag the Network Printer icon to an open area of the Desktop or an open folder, and release the mouse button.

Before attempting to use these procedures, you should be familiar with the basic operation of OS/2 2.0 and OS/2 2.0 objects.

4. When the Access Another Network Printer window appears, click on the down arrow at the right end of the Server line. Use the cursor arrow keys or click on the small arrows to the right of the displayed server name to scroll to another server.
5. Scroll to the name of the print server you wish to access and click on it. You also can type the name of the print server on the line that contains the network printer you wish to add.
6. Click on the down arrow at the right end of the Resource line.
7. Use the cursor arrow keys or click on the small arrows to the right of the resource name to scroll to another resource. Scroll to the name of the resource you wish to access.
8. Click on the name of the resource. You also can type the name of the resource on the line that contains the network printer you wish to add to a requester.
9. Click on the OK pushbutton.

When the Install Printer Driver screen appears, it indicates that you do not have the correct OS/2 Printer Driver installed on your workstation for using a network printer. You must install the missing printer driver on the workstation. It must be the same printer driver as used on the network print server. If they do not match, you will receive no output. Some locations have the printer drivers available as a LAN resource, and other locations require the OS/2 2.0 Printer Driver diskettes. Use the "Installing OS/2 2.0 Printer Drivers" procedure to install the needed printer drivers.

When the printer driver is installed, the new Printer Object icon will be displayed with the name specified in the General parameter of the Printer object.

Changing a Printer Object or Icon Name

The procedures described below should be used for changing a printer object or icon name.

1. Position the mouse pointer on the name line under the icon to be changed.
2. Press and hold the Alt key, then press the left mouse button.
3. Release the Alt key and mouse button. The name line under the icon will be highlighted and can now be changed. To exit without making a change, press Esc.
4. Type the new name and delete any extra characters.
5. Press Enter when all information has been changed.

Changing Printer Object Parameters

Changing Printer Object parameters can be done by following these steps:

1. Click the right mouse button on the Printer Object to be changed.

2. Click on the right arrow next to the Open line to display the Printer Object action list.
3. Click on Settings. An index-tabbed folder is displayed. Click on the tab for the parameter you wish to display or change, using the arrows in the bottom left corner of this window to scroll through all the parameters.

The following paragraphs describe how to change each of the parameters that can be modified after creating the Printer Object.

View: The physical name displayed when the View tab is selected cannot be changed at this time. To change this name, delete the Printer Object and create a new object with the desired name. Do not use the Desktop Shredder to delete a printer—refer to “Deleting a Printer Object” in this article.

Refresh interval: On a network printer, you can change the refresh interval if necessary. This parameter is not displayed for local printers. Some printers require a longer timeout when large amounts of data are sent to the printer. The IBM 4029 printer has worked well at the 45-second default. Refer to the documentation supplied with your printer if you receive incomplete output or no output.

Printer driver: The following procedure can be used to add a new printer driver to a workstation. The printer driver cannot be changed on a network printer because it is controlled by the print server. When using a LAN-attached printer, you must have the same driver installed on the workstation that is installed on the print server.

1. Move the mouse pointer to any driver in the Printer Driver area.
2. Click the right mouse button.

3. Click either mouse button on Install.
4. Use your local procedure to install the printer driver needed. (One method is discussed in “Installing OS/2 2.0 Printer Drivers” in this article.)
5. Click on Refresh.
6. Select the driver to install by clicking on it with the left mouse button. To install more than one driver, hold Ctrl and click the left mouse button or select and install each driver individually.

The printer driver cannot be changed on a network printer because it is controlled by the print server.

7. Click on Install.
8. Click on the OK pushbutton when installed. You can install additional drivers if you plan to add different printers.
9. Click on the Cancel pushbutton when complete.

After the printer driver has been selected or added, you should verify the Printer Job Properties. Use the following procedure to change or verify the Job Properties:

1. Click on the Job_Properties pushbutton.
2. Change any Job Properties to your printer's specifications.
3. Click on the OK pushbutton to save Job Properties.

Output: The Output parameter does not appear in the network printer settings and is only applicable to a local printer. The Output section deter-

mines the port (LPTx or COMx) to which the printer is to be connected. On a server, this could be directly attached to the LPT1 or COM1 port or through a 4033 Network Printer Adapter. On a workstation, this would be physically attached to an available LPT or COM port.

The LPTx connections that are shaded are in use, and the highlighted LPTx is the one selected for the printer with which you are working. Pressing Default returns this screen to the settings active at entry.

Queue options: PMPRINT should be the active Queue Driver for the IBM 4029 LaserPrinter or the Queue Driver recommended by your specific printer manufacturer.

Print options: This parameter controls the selection of a separator page if desired. Specify the full path, file name, and extension for the separator page to be used. You also can disable this printer object for specific times if needed; for example, when you have an alternate printer object defined to use a different paper that could be loaded at specific times for this destination. Enter the following data on the separator line of the Print Options parameter:

C:\SP00L\4029PPDS.SEP

where C:\SP00L is the path of separator file 4029ppds.sep, as loaded.

Window parameter: There currently are no recommended changes to the Window parameter. If you have added a new printer type, refer to the documentation provided with the printer for any parameter changes.

General parameter: Do not change the comments specified in the Printer Object General parameter on a print server after users have accessed the printer. Changing comments also changes the icon name line on the

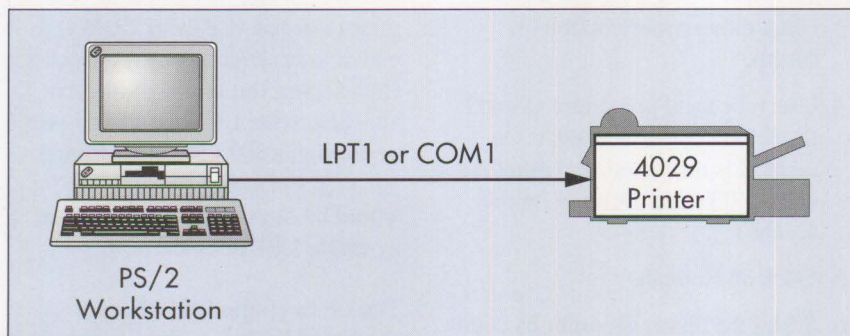


Figure 1. Configuration of One Printer and One User

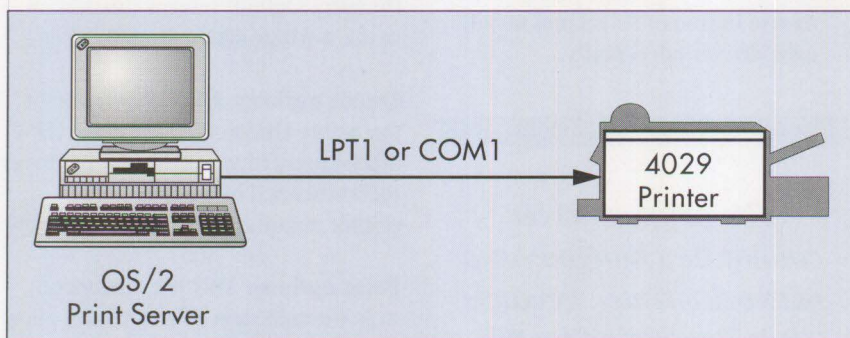


Figure 2. Configuration of One Printer on a Print Server

printer object displayed on the Print Server Desktop.

Network status: The Network Status appearing for network printers is for information only. A status of Available is normal. A status of Unavailable will appear when the Print Server is not operating correctly.

To save parameter changes for this printer object, use the following procedure:

- Double-click in the upper left corner of the Printer Settings window using the left mouse button, or click in the upper left corner of the Printer Settings.
- Click on Close.

Deleting a Printer Object

This procedure deletes an unused Printer Object from a print server or workstation. Do not use the Shredder to delete a printer object.

1. Move the mouse pointer to the icon for the Printer Object icon to be deleted.
2. Click the right mouse button on the Printer icon.
3. Click the left mouse button on Delete Action.
4. Click the left mouse button on Delete to verify deletion of the Printer Object.

Installing OS/2 2.0 Printer Drivers

The following procedure can be used to install a printer driver for a local printer on a workstation. If the printer driver file resource (Alias) is available, it can be used to install the printer drivers. Check to determine if the Alias has been created and how to use it. If the Alias is not available, use the OS/2 2.0 Printer Driver diskettes supplied with OS/2 2.0.

If Using the Printer Driver Alias

1. Determine the driver you wish to install and the disk needed.
2. Log on to the LAN.
3. From an OS/2 prompt, enter the following:

```
NET USE T: PDRIVERX
```

You can use any free disk letter for T:; x is the printer driver disk containing the driver to be installed.

If Using the Printer Driver Diskettes

1. Press Ctrl and Esc to return to the Desktop. Double-click on the icon for the print object for which you want to add the printer driver.
2. Click the right mouse button on the Printer icon.
3. Click on the arrow to the right of Open.
4. Click on Settings.
5. Click on the Printer Driver tab.
6. Click the right mouse button on any printer driver.
7. Click on Install.
8. Change the A:\ prompt in the Install New Printer Driver window to the drive letter you specified in the NET USE command.
9. Click on Refresh.
10. When the printer drivers appear in the window, select the printer driver to install. To install more than one driver, hold Ctrl and click the left mouse button or select until all required drivers are highlighted.
11. Click on Install. Press Cancel when all printer drivers have been installed, or return to step 1 to install additional printer drivers.
12. Select New Printer Driver for this printer.

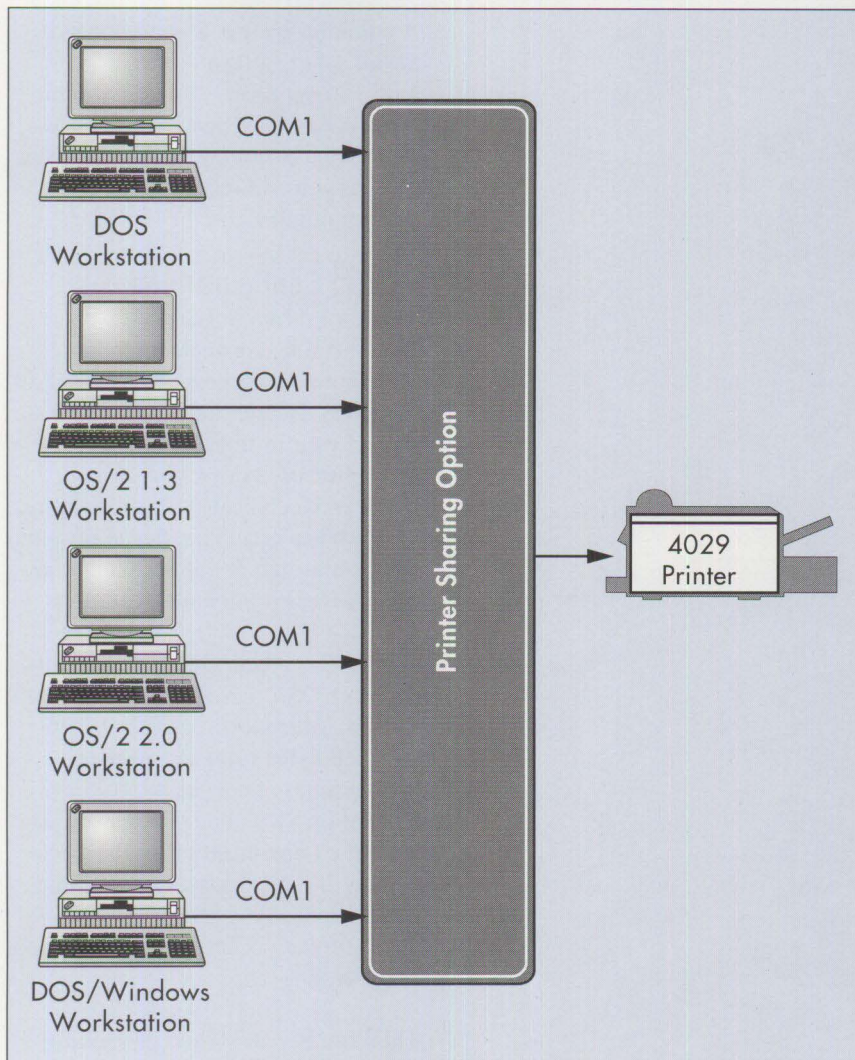


Figure 3. Configuration of One Printer on a Printer Sharing Option

13. Double-click on the System icon (upper left) to close Settings.

OS/2 2.0 Printer Driver Alias

A LAN coordinator can use this procedure to help end users install OS/2 2.0 printer drivers. This procedure allows any user on the LAN to access the Printer Drivers and eliminates the need for multiple copies of the OS/2 Printer Driver diskettes. It provides one place to be updated when maintenance is required. This section describes how to create the five aliases needed for each diskette on a LAN file server. You must create five

separate files and aliases because the OS/2 and some of the Windows-OS/2 drivers have the same names but exist on different diskettes.

The following procedure will build the files needed to create a printer driver alias on your file server.

1. Obtain the five OS/2 2.0 Printer Driver diskettes.
2. Create a PD directory on the file server.
3. Create subdirectories for Disk1 through Disk5.

4. Copy the contents of each printer driver diskette to its respective subdirectory.

Next, ensure that the Master file server is running on your domain.

1. Log on to the domain with an ADMIN ID.
2. Type NET at an OS/2 full-screen prompt.
3. Select Definitions.
4. Select Aliases.
5. Select Files.
6. Select New by pressing the spacebar, then press Enter.
7. Select Actions.
8. Select Create, then enter the Alias name (such as PDRIVER1).
9. Enter Description (OS/2 2.0 Printer Drivers).
10. Enter the server name.
11. Enter the path (C:\PD\DISK1). Use C:\PD\DISK1 if you have created your own files.
12. Set When Shared to At Server Startup, then press Enter. This allows the alias to be accessed by other users on the LAN when the file server containing the Printer Driver Alias is started (booted).
13. When the screen is repainted, scroll to the alias just created.
14. Select the alias by pressing the spacebar.
15. Select Access_Profile.
16. Select Create.
17. Set Universal access permission to R (Read) and press Enter. This action is necessary to allow DOS or LAN users on a different domain to access the Printer Driver Alias just created.
18. Scroll again to the alias just created.

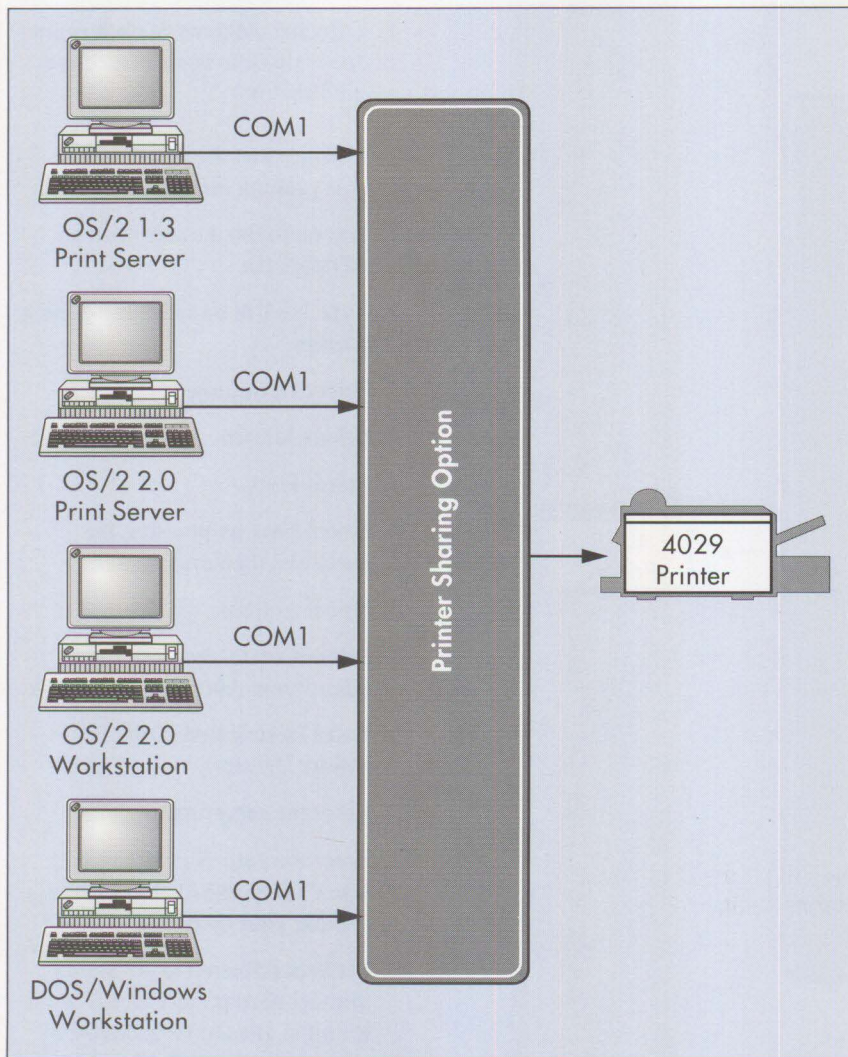


Figure 4. One Printer on a Printer Sharing Option and Print Server

19. Select the alias by pressing the spacebar.
20. Select Access_Profile.
21. Select User_List.
22. Scroll to Guest.
23. Enter the same Permissions as used in Create (such as R for Read), and press Enter.
24. Scroll again to the alias you have just created.
25. Select the alias by pressing the spacebar.
26. Select Apply, and press Enter twice.

Repeat this procedure for the four remaining driver files (Disk2, Disk3, Disk4, and Disk5).

Users can now access the Printer Driver Alias and update or install OS/2 2.0 printer drivers on their requester workstations.

Some drivers have many model types, so it is necessary to choose the driver and model that match your hardware or LAN Print Server configuration. If an incorrect driver is chosen, your data may be sent to the printer and appear to be printing, but you may not receive any output.

A common error is to send non-PostScript data to a PostScript® printer. You will not receive any output; the job will just disappear. When PostScript data is sent to a non-PostScript printer, you will receive page after page of numbers on the left edge.

Printer Configurations

Figures 1 through 5 show ways a printer can be connected to allow maximum user access. This is not a complete list and may not be applicable in your installation due to print requirements, availability of hardware, and technical skills availability. You need an accurate estimate of your print needs before committing to any multi-access printer configuration.

The PS/2 can be a DOS, DOS/Windows, or OS/2 1.3 or OS/2 2.0 workstation. If the workstation is directly attached to the printer or uses the Printer Sharing option, the Automatic Emulation Switching (AES) software should also be installed on the workstation. If the workstation is using a LAN-connected print server, the AES software is not required on the workstation.

The Print Server should be running OS/2 1.3 and LAN Server or OS/2 2.0 and Entry Server. The AES software also can be installed on the Print Server and allow mode switching between text and PostScript modes by the user's selection of different network printer queues. Separator pages that perform the mode switch can be used in place of the AES software.

One Printer, One User

This connection is the least desirable since the printer use is limited to one user at a time, as shown in Figure 1. It can be used when confidentiality of data is required or a large amount of print is generated.

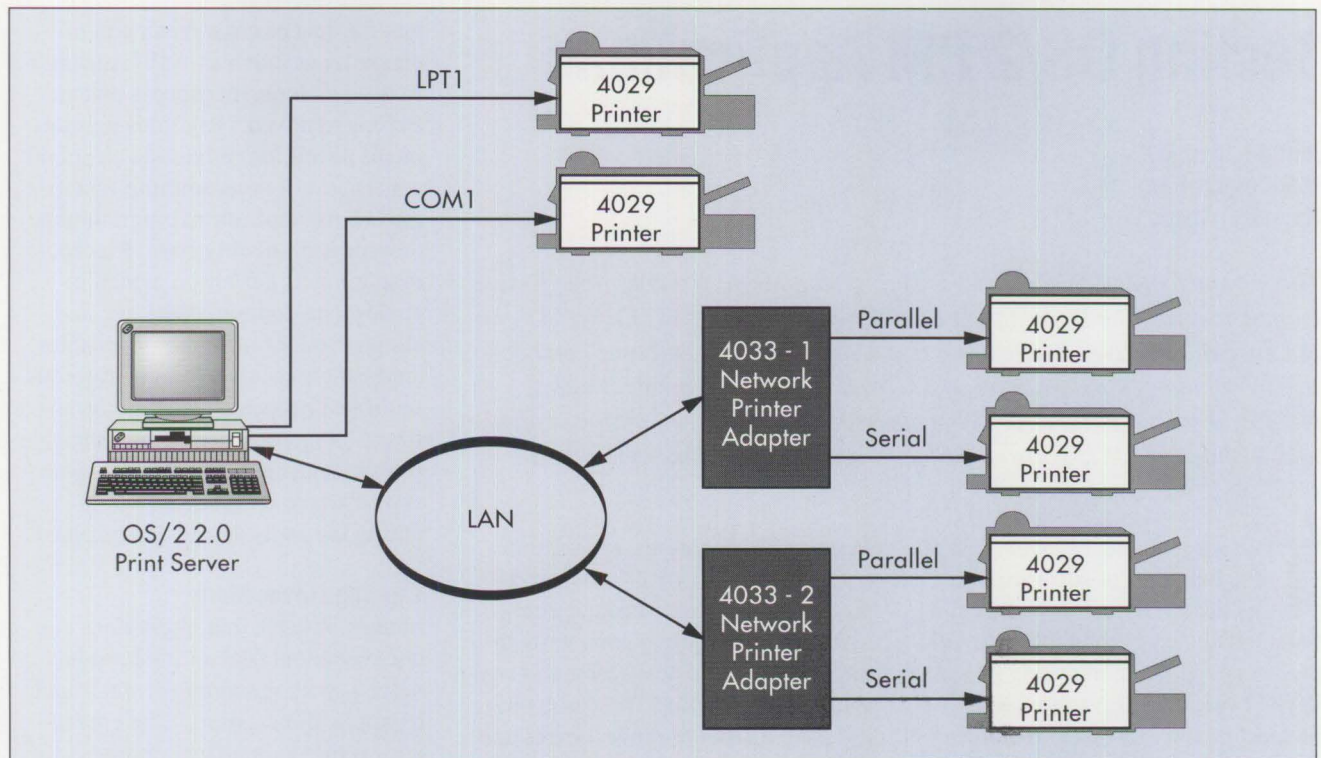


Figure 5. Configuration of Many Printers on One Print Server

One Printer on a Print Server

This connection allows the LAN-enabled user access to a high-quality printer. The disadvantage of this connection is that the printer must be physically connected to the print server, which may be located in a limited access area. Figure 2 shows this configuration.

One Printer on a Printer Sharing Option

This configuration allows up to six workstations to access one physical printer when the LAN connection is not an option. The disadvantage is that all workstations must have a way to switch modes on the printer to ensure that they can print in the mode they need. The Print Sharing Option (PSO) box must be located within 50 feet of the workstation and 6 feet or less from the printer. Figure 3 shows this option.

One Printer on a Printer Sharing Option # 2

This configuration allows a combination of print servers and/or workstations to access one physical printer when not all workstations are LAN-capable. This configuration also can be used in a LAN migration and Print Server update or test environment plan, as shown in Figure 4.

Many Printers on One Print Server

This configuration allows multiple printers to be connected to one print server. By using the 4033 or Network Printer Adapter, the printers can be located wherever a LAN connection is available. This allows one printer in a limited access area next to the print server and additional printers closer to the end user's workstation. The disadvantage is that it requires all users to be LAN-enabled. Figure 5 shows this configuration.

Richard R. Miller is a staff systems analyst in IBM U.S. Quality Research and Development in Raleigh, North Carolina. For the last six years, Richard has been working with VM and MVS Advanced Function Printing, DOS and OS/2 workstation printing, and IBM LAN printer hardware and software installation, usability, and support. His previous experience includes positions as customer engineer, program support representative, Level 2 support representative, PC application development, and VM system programmer.

Testing OS/2 PM Applications

Philip L. Arthur
IBM Corporation
Roanoke, Texas

This article focuses on the process changes and testing theory for effectively implementing the Workstation Interactive Test Tool (WITT). WITT is an automated test driver that tests an application from an end user perspective by simulating end user keystrokes and mouse movements. Testing OS/2 Presentation Manager (PM) applications is the primary focus of this article; however, WITT also tests host-based applications running on MVS, VM, VSE, and AS/400.

Testing accounts for over 50% of the total cost of software over its life cycle.¹ At first glance, that number seems too high until you dissect the application testing process itself. Typical testing activities include creating test scenarios, identifying and creating input data, establishing test databases, running test cases, evaluating test results, debugging/resolving code deficiencies, recompiling the program, and repeating one or more of these activities until the code or application is ready for production.

Several consulting studies show that the average application program goes through some or all of these activities over 25 times before development is completed. That does not include production fixes, enhancements, or maintenance. Sophisticated applications involve test planning, design, validation, completeness, and testing the entire application by separate test groups. These activities often are repeated for enhancements or code fixes with limited reuse. Therefore, test case reuse and maintainability are critical to improving testing productivity and cutting development costs.

In many organizations, application code quality is controlled by application programmers. Programmers often receive requirements, write the code, create a few test cases, and run some unit test cases. They probably use a debug tool to run test cases and identify defects, and ultimately put the code into production if the results look good. Test cases for ensuring that code meets requirements must be developed from the requirements. Detailed test plans should be defined before the coding phase.

Application Development Process

An effective development process has quality checkpoints at each development phase. The software development life cycle becomes a set of work products and releases, permitting continuous assessment of product quality and allowing for changes in the development process.

Rigorous Requirements Definition

Rigorous requirements definition enables programmers to focus on error prevention instead of error removal. B. W. Boehm states in *Software Engineering Economics*

that the cost of error repair can increase by as much as 100:1, the later in the development process that an error is removed.² Rigorous requirements definition provides a clear and concise set of programming specifications for application programmers to write high-quality code. Rigorous requirements definition, which includes data domain, data type, and allowed values information, enables application testers to build a rigorous set of test cases and validate the test results. Rigorous requirements enable programmers to build their test plan and software acceptance criteria during the analysis and design phase.

Thorough Inspections

Studies indicate that inspections can uncover about 60% of total product defects, with remaining errors found by application testing.³ The cost of error removal increases dramatically the longer the error remains in the product. Therefore, requirements and design reviews are critical to developing quality software.

Automated Testing

Considering data domain, allowed values, and application process descriptions, English-like Workstation Interactive Test Tool (WITT) test scripts can be created before the code is completed.

Root Cause Analysis

A defect database containing information about errors identified during inspection and testing plays a critical role for improving the development process, in addition to monitoring and controlling development progress. The entries in the database require root cause analysis to help prevent defects in the future.

¹ Beizer, Boris. *Software Testing Techniques*. New York: Van Nostrand Reinhold, 1991.

² Boehm, B. W. *Software Engineering Economics*. Englewood Cliffs, NJ: Prentice-Hall, 1980.

³ Dyer, M. "A Formal Approach to Software Error Removal." *The Journal of Systems and Software* (July, 1987).

Workstation Interactive Test Tool

WITT is an automated test driver tool that can record and play back key-strokes and mouse movements. It also can be substituted for much of the manual testing performed today. WITT supports Systems Application Architecture® (SAA™) interactive application environments, including 3270, 5250, and OS/2 Presentation Manager screens. It is easy to use and intuitive, and can run interactively or as an OS/2 background job.

WITT can capture and compare screens, and it supports non-compare areas. Date and time are typical non-

compare areas because they are constantly changing. Also, WITT can capture and compare strings on the screens. Captured screens can be printed and/or embedded in user documentation.

WITT is a black box test tool. The black box approach tests an application from a user perspective, focusing on user input and the resulting output. The programmer or tester builds test cases and expected results from requirements documents, design documents, and user documentation without regard to program implementation techniques. These test scenarios are executed using WITT.

WITT test case management can be as simple as storing test cases in an OS/2 directory or subdirectory, or as sophisticated as test case drivers that invoke test scripts to test a single module, a component, or the entire application.

WITT supports time-initiated background jobs, keyboard remapping, copy/paste/delete, edit using an editor, hot keys, and the High-Performance File System (HPFS). Test files can reside on the user's workstation, be shared on a LAN, or be shared on a host.

Since WITT is fully programmable, it can be used as a simple record-and-



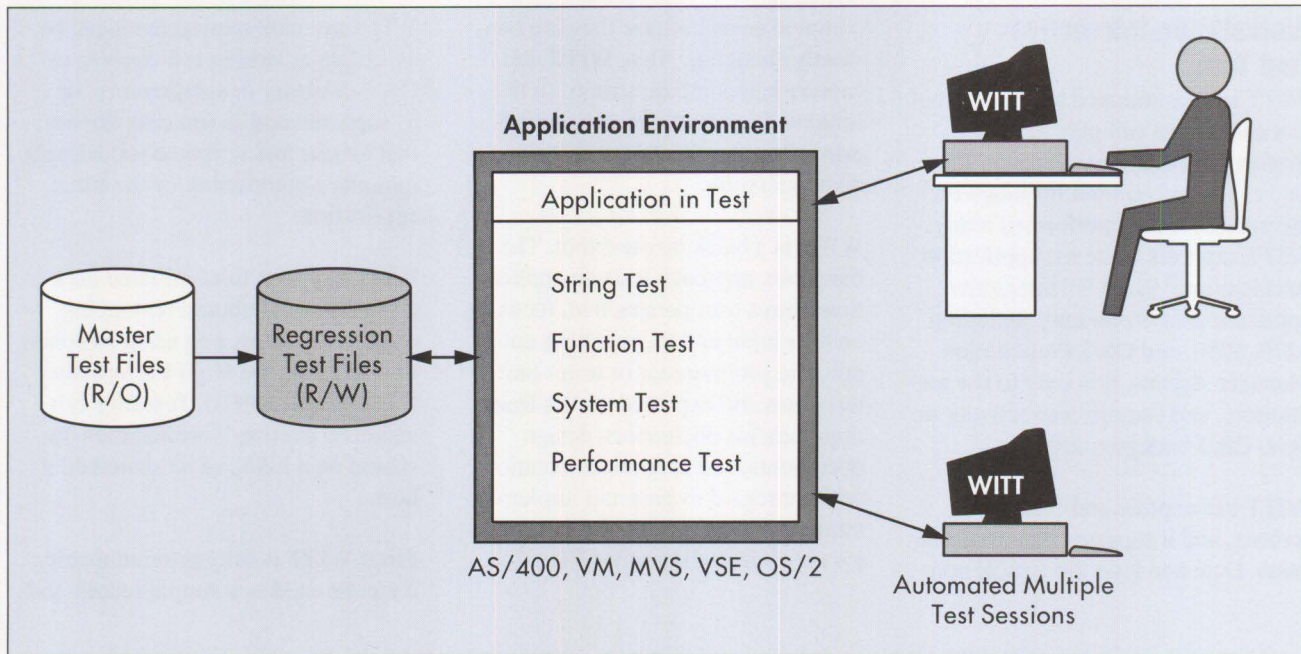


Figure 1. A Definable and Repeatable Test Environment

playback tool or as sophisticated "smart testware." WITT supports its own powerful commands and OS/2's 2/REXX language. Also, WITT scripts can call other test case scripts, 2/REXX programs, and C language programs.

The role of 2/REXX is valuable because 2/REXX provides conditional logic, exception handling and recovery, and I/O. More importantly, 2/REXX facilitates the methodology for test case reuse and maintainability, which requires identifying and changing a test script for a corresponding application change. This is sometimes viewed as a function of test management.

WITT uses OS/2 PM Application Programming Interfaces (APIs) to find icons, folders, windows, pushbuttons, pull-downs, and so on. Therefore, WITT is not sensitive to x,y screen coordinates like many test tools.

WITT is used in several areas. Application programmers use it to perform unit and string testing of applications.

System test groups use WITT to test the function of the entire application within the system environment. WITT can monitor end user response times. Performance groups use WITT to do multi-user performance testing. However, WITT's main use is regression testing.

A Development Scenario with WITT

WITT can be used to initialize application databases or files. WITT test scripts can be created by recording and playing back application test scripts or by extending test scripts with 2/REXX.

Definable and Repeatable Test Environment

Application testers must be able to refresh the application data, such as OS/2 SQL tables, to rerun test cases that modify that data. A *definable test environment* uses known inputs where programmers have defined the expected results from application transactions. A *repeatable test environment* uses databases refreshed with the same definable test data to

establish a regression test environment. Establishing a definable and repeatable test environment is critical to maximize the value of automated testing. Figure 1 shows an approach to refreshing application test data files and tables before running test cases. This test environment initialization activity is implemented as a callable WITT script, which can be used by multiple test case scenarios.

WITT Test Case Scripts

Many application programmers use WITT to record and play back test cases. Others create sophisticated testware or do performance monitoring. Following are examples of these applications.

WITT script syntax: A WITT test case is a test script consisting of keystrokes, mouse movements, WITT commands, 2/REXX statements, and test case comments. WITT scripts can be created with an editor or by recording a test session using WITT. In either case, data strings typed at the terminal are placed inside quotes and arrows; for

example, '<C370.PRJ>'. Function keys (such as F12), Clear, and Enter are placed inside quotes; for example, 'ENTER'. WITT commands also are placed in quotes; for example, 'SAVE-SCREEN' causes WITT to capture a copy of the screen for validating test (transaction) results, making future comparisons, or including in user documentation.

Creating a WITT script by recording:

To build a base script, start WITT in record mode to capture a test scenario of the application transaction. Starting WITT in record mode is easy—a few mouse clicks and typing a new test case name. WITT records the keystrokes and creates a test script, such as the one shown in Figure 2. This recording selected the IBM Software Analysis Test Tool (SATT) window and opened a file using Open on the File pull-down menu. WITT waits until the SATT application is up and ready for input. It records the Home key and the file name entered.

WITT script extended with

2/REXX statements: Sophisticated testware can be created by parameterizing a recorded script with 2/REXX. Figure 3 shows an example test case driver script that contains multiple test scenarios, which are calls to other WITT scripts that represent the functions of an application window. The driver script can call an error recovery routine, a data or file initialization routine, and the calls to the

```
/*Workstation Interactive Test Tool: Test Case Procedure. */
'WINDOW-SELECT "Software Analysis Test Tool"'
'MOUSE-BUTTON1-DOWN "WC_TITLEBAR",1,10,7'
'MOUSE-BUTTON1-UP 0,0'
'MENU-SELECT "File", "Open..."'
'ENTER'
'WAIT-FOR-WINDOW 'Open''
'HOME 1,1'
'<C370.PRJ>'
```

Figure 2. Example of a WITT Script

necessary window scripts to perform the test scenario.

The variables in Figure 3 can be parsed into 2/REXX variables for use by the WITT script representing the main window, as shown in Figure 4. This sample of a WITT script performs all the functions of the action bar. Additional calls within the script would perform functions on the subsequent pop-up windows.

WITT performance monitoring:

The 2/REXX Time function enables WITT to capture response times and store them in an ASCII file or insert them into an OS/2 database. The 2/REXX Time function has small execution time overhead (approximately 0.03 seconds) and does not significantly impact performance measurements. Figure 5 shows an example of capturing the elapsed time for the Open window to pop-up, then writing that response time information into an ASCII data set.

Test case execution and results

validation: Ideally, an independent

test person reviews the testing results instead of the application programmer who wrote the code. Once the results are verified, the WITT-captured screens should be archived for future use. Often, future changes to the application can still use all or most of the test scenarios with minimal changes. The archived screens serve as a baseline for comparison to the results from the newly changed code. WITT performs this comparison function without a tedious human review process by highlighting any screen mismatches. The tester reviews any mismatches between the archived results (screens) and the current results. The WITT Screen Printing function keeps the documentation for the application current with each application change by printing the screens to a file that can be embedded within the application documentation.

Unit and string test reruns: During new code development, programmers often make corrections to the code, which must be fully regression tested. This means that all test scenarios are rerun to ensure previously tested

```
/*Workstation Interactive Test Tool: Test Case Procedure. */
call on error name backout /* Common error recovery routine */
/* Test scenario #1: testing print function */
'CALL ACTINIT' /* Refresh/Initialize application data */
'CALL ACTBAR(File Print)' /* Select Print pull-down choice */
'CALL.....' /* Process Print window functions */
/* Test scenario #2 - testing file open function */
'CALL ACTINIT' /* Refresh/Initialize application data */
'CALL ACTBAR(File Open...)' /* Select File Open pull-down function */
'CALL.....' /* Process Open window functions */
```

Figure 3. WITT Test Driver Script

```

/*Workstation Interactive Test Tool: Test Case Procedure. */
call on error name bailout /* Common error recovery routine */
PARSE ARG file_rexx choice_rexx
'WINDOW-SELECT "Software Analysis Test Tool"'
'MOUSE-BUTTON1-DOWN "WC_TITLEBAR",1,10,7'
'MOUSE-BUTTON1-UP 0,0'
'QUERY-ACTIVE main_window_rexx'
'MENU-SELECT "'file_rexx"', "'choice_rexx'"
'ENTER'
'QUERY-ACTIVE current_window_rexx'
'WAIT-FOR-WINDOW 'current_window_rexx''
if current_window_rexx <> main_window_rexx then
do
say "Closing pull-down choice window " current_window_rexx
'SYSMENU-SELECT "Close Alt+F4"'
'ENTER'
'WAIT-FOR-WINDOW "'main_window_rexx'"
end
else
say choice_rexx "is not a selectable choice."
exit

```

Figure 4. Callable Parameterized WITT Script (ACTBAR.CMD)

```

/*Workstation Interactive Test Tool: Test Case Procedure. */
/* Measure window pop-up response time and write to file.*/
outfile='c:\witt12\response.dat'
'WINDOW-SELECT "Software Analysis Test Tool"'
'MOUSE-BUTTON1-DOWN "WC_TITLEBAR",1,10,7'
'MOUSE-BUTTON1-UP 0,0'
'MENU-SELECT "File", "Open..."
'ENTER'
startclock=time('R')
'WAIT-FOR-WINDOW 'Open''
ewindow=time('R')
call lineout outfile,'Editor Window pop-up time:', ewindow

```

Figure 5. Sample WITT Performance Monitor Script

scenarios still operate properly. The WITT examples shown are simple examples of WITT's power and flexibility when combined with 2/REXX. More elaborate and comprehensive testware can be created to correspond with the complexity of the application function. Parameterized WITT scripts serve as complete testware for functional verification of the code. WITT's copy/paste/edit capability enables the creation and replication of WITT testware during the analysis and design phases. The inputs, expected outputs, and WITT script represent the test plan documentation. This test plan documentation (test-

ware) should be reviewed and approved during the review of the application design, before writing the application code.

Enhancements/Maintenance

Many of today's large applications are poorly documented. The biggest productivity gains from WITT are reusing, enhancing, and maintaining the WITT testware (test case drivers and test scripts) along with the code.

Success Factors

The following are the critical success factors for test automation.

- Creating a definable and repeatable test environment
- Developing test case scenarios during the analysis and design phase
- Designing WITT test scripts that are easy to create, edit, and maintain
- Using a testing methodology that supports these factors

Philip L. Arthur consults on and supports AD/Cycle™ test/maintain/redevelop tools, products, and methodology for IBM. His responsibilities bring him into frequent contact with clients having sophisticated application development environments, as well as with IBM development organizations. Phil has consulted on testing methodologies, tools, and techniques from unit test to system test to user acceptance test, dramatically improving testing productivity and software quality. He has a BS in engineering from Purdue University and two years post-graduate work in engineering and business administration.

Accessing a Remote AS/400 Using OS/2 Extended Services

Timothy S. Huntley
IBM Corporation
Research Triangle Park, North Carolina

This article provides a step-by-step procedure for using OS/2 Extended Services to connect workstations on a LAN to a remote AS/400. The benefit of this approach is reduced costs compared to connecting directly to the AS/400.

Connectivity between AS/400s and intelligent workstations can be as modest as terminal emulation or as powerful as cooperative resource sharing. Workstations can be attached to AS/400s by either 5250 terminal emulation or an enhanced connectivity product called IBM PC Support/400. PC Support/400 enables personal computers to use AS/400 resources such as disk storage, files, and printers transparently. Both 5250 terminal emulation and PC Support/400 use the Advanced Program-to-Program Communications (APPC) protocol to communicate with AS/400s.

Most workstations connected to AS/400s are attached directly through LANs; however, many workstations on LANs must access remote AS/400s. One way to do this is through direct connections, as shown in Figure 1. Methods for direct connections include Synchronous Data Link Control (SDLC), X.25, or a remote twinaxial controller.

OS/2 Extended Services eliminates the need for physical connections between each workstation and the remote AS/400. With Extended Services, a single OS/2 workstation can act as a gateway for other workstations on a LAN, using a single SDLC or X.25 connection to a remote AS/400 (see Figure 2). The workstation running Extended Services uses

Advanced Peer-to-Peer Networking (APPN) to create intermediate APPC sessions between other workstations and the AS/400. APPN, a set of functional enhancements to Systems Network Architecture (SNA), provides an underlying network with intelligence. These enhancements, including dynamic directory and routing, make it easier to set up and manage computers on the edge of a network. In addition, APPN enables sessions between application end points to be routed through multiple computers.

Configuring Extended Services

The following steps are necessary to set up a single OS/2 workstation to run Extended Services and connect to an AS/400:

1. Evaluate hardware and software requirements for OS/2 and Extended Services.
2. Install OS/2 and Extended Services.
3. Configure the Communications Manager to run 5250 emulation.
4. Modify the Communications Manager and AS/400 configuration.

Evaluating Hardware and Software Requirements

Hardware requirements for Extended Services are described in *IBM Extended Services for OS/2 Hardware and Software Reference* (S04G-1014).

In addition, the gateway computer requires at least one LAN card and at least one SDLC or X.25 card.

Extended Services requires OS/2 1.3 or later. To order OS/2 directly from IBM, call (800) 3IBM-OS2 (342-6672). When placing the order, request a copy of *IBM Extended Services for OS/2 Communications Manager Configuration Guide* (S04G-1002). This is an invaluable reference when using Extended Services.

Installing OS/2 and Extended Services

If installing OS/2 1.3, read *IBM Operating System/2 Standard Edition: Getting Started*. For OS/2 2.0, start with the *Installation Guide*. OS/2 also includes tutorial information that will speed you along the OS/2 learning curve. After installing OS/2, you will need to install Extended Services. Before doing so, read *IBM Extended Services for OS/2: Start Here* (S04G-1000).

Configuring 5250 Emulation

Although it is not required to configure 5250 emulation on the Extended Services computer, there are significant advantages to doing so. Before APPN can be configured, the Communications Manager must be configured to support APPC. Since 5250 emulation uses APPC, the Communications Manager will then be configured for APPC.

To configure 5250 emulation on the OS/2 machine and the AS/400, refer to *IBM Extended Services for OS/2 Communications Manager Configuration Guide*. You also must configure LAN support in the Communications Manager. This enables other workstations on the LAN to connect to the Extended Services computer and "talk" to the AS/400. Be sure that you can use 5250 emulation, as described in the *Configuration Guide*, before continuing to the next section.

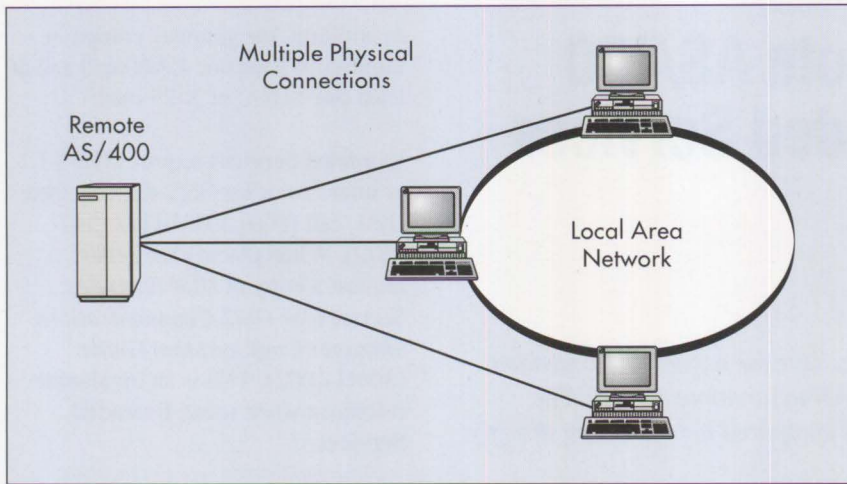


Figure 1. Direct Access to a Remote AS/400

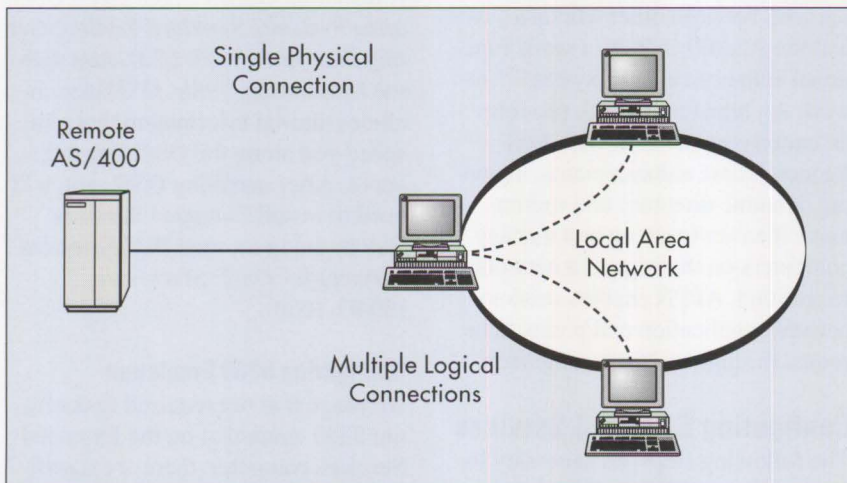


Figure 2. Access to a Remote AS/400 through Extended Services

Modifying the Configuration

To enable the new functions that APPN adds to the Communications Manager, make a few simple changes to the new Communications Manager configuration file and to the AS/400. This Communications Manager configuration file, called the Node Definitions File (NDF), is created during the installation of Extended Services. The NDF has the same file name as the binary Communications Manager configuration file, with a file extension of .NDF instead of .CFG. The NDF should look similar to the one shown in Figure 3.

Using an ASCII text editor, make the following changes to the NDF file, as shown with arrows in Figure 4:

1. On the `DEFINE_LOCAL_CP` verb, change the `NODE_TYPE` parameter from `EN` to `NN`. This change switches the OS/2 computer from an APPN end node to an APPN network node that can route APPC traffic.
2. On the `DEFINE_LOGICAL_LINK` verb, do the following:
 - Change the `ADJACENT_NODE_TYPE` parameter from `LEN` to `LEARN`.

- Delete the `FQ_ADJACENT_CP_NAME` parameter.
- Change the `CP_CP_SESSION_SUPPORT` parameter from `NO` to `YES`.
- Change the `ACTIVATE_AT_STARTUP` parameter from `NO` to `YES`.

These changes let the AS/400 and Extended Services computer establish a link, learn about each other's characteristics, and activate control point (CP-CP) sessions. Over a pair of control point (CP-CP) sessions, two computers manage APPN routing and directory functions.

3. Delete the entire `DEFINE_PARTNER_LU_LOCATION` verb and its parameters. Since APPN support is configured, it is not necessary to tell the OS/2 computer where the partner AS/400 is located.
 4. On the `DEFINE_DEFAULTS` verb, change the `IMPLICIT_INBOUND_PLU_SUPPORT` parameter from `NO` to `YES`. This change enables other workstations to connect into the OS/2 computer without having explicit definitions.
 5. Remove the entire `CNOS` verb and its parameters. Because Extended Services issues this verb as needed, you need not do it explicitly.
 6. Save this file and exit the text editor. From an OS/2 command line, type:


```
APPNV filename
```

where `filename` is the name of the NDF file.
- Three parameters need to be updated on the AS/400. Make the following changes in the controller description, as shown in Figure 5:
- Change `APPN CAPABLE` to `*YES`.


```

DEFINE_LOCAL_CP  FQ_CP_NAME(USIBM.TIM)
                  CP_ALIAS(TIM)
                  NAU_ADDRESS(INDEPENDENT_LU)
                  NODE_TYPE(EN)
                  NODE_ID(X'513EB')
                  HOST_FP_SUPPORT(NO);

DEFINE_LOGICAL_LINK  LINK_NAME(LINK0001)
                     FQ_ADJACENT_CP_NAME(USIBM.CP000002)
                     ADJACENT_NODE_TYPE(LEN)
                     DLC_NAME(SDLC)
                     ADAPTER_NUMBER(0)
                     CP_CP_SESSION_SUPPORT(NO)
                     ACTIVATE_AT_STARTUP(NO)
                     LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
                     LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                     SOLICIT_SSCP_SESSION(NO)
                     EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                     COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                     COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                     SECURITY(USE_ADAPTER_DEFINITION)
                     PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_3(USE_ADAPTER_DEFINITION);

DEFINE_LOCAL_LU  LU_NAME(LU5250 )
                  LU_ALIAS(LU5250 )
                  NAU_ADDRESS(INDEPENDENT_LU);

DEFINE_PARTNER_LU  FQ_PARTNER_LU_NAME(USIBM.AS400)
                   PARTNER_LU_ALIAS(PLU5250)
                   MAX_MC_LL_SEND_SIZE(32767)
                   CONV_SECURITY_VERIFICATION(NO)
                   PARALLEL_SESSION_SUPPORT(YES);

DEFINE_PARTNER_LU_LOCATION  FQ_PARTNER_LU_NAME(USIBM.AS400)
                             WILDCARD_ENTRY(NO)
                             FQ_OWNING_CP_NAME(USIBMZP1.CP000002)
                             LOCAL_NODE_NN_SERVER(YES);

DEFINE_MODE  MODE_NAME(QPCSUPP )
              COS_NAME(#CONNECT)
              DEFAULT_RU_SIZE(NO)
              MAX_RU_SIZE_UPPER_BOUND(2048)
              RECEIVE_PACING_WINDOW(7)
              MAX_NEGOTIABLE_SESSION_LIMIT(32767)
              PLU_MODE_SESSION_LIMIT(7)
              MIN_CONWINNERS_SOURCE(3);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(NO)
                  DEFAULT_MODE_NAME(BLANK)
                  MAX_MC_LL_SEND_SIZE(32767)
                  DIRECTORY_FOR_INBOUND_ATTACHES(*)
                  DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                  DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                  DEFAULT_TP_CONV_SECURITY_RQD(NO)
                  MAX_HELD_ALERTS(10);

START_ATTACH_MANAGER;

CNOS  LOCAL_LU_ALIAS(LU5250 )
        FQ_PARTNER_LU_NAME(USIBM.AS400)
        MODE_NAME(QPCSUPP )

```

Figure 3. Sample Node Definitions File after Installation of Extended Services (Continued)

```

SET_NEGOTIABLE(NO)
PLU_MODE_SESSION_LIMIT(7)
MIN_CONWINNERS_SOURCE(4)
MIN_CONWINNERS_TARGET(0)
AUTO_ACTIVATE(0);

```

Figure 3. Sample Node Definitions File after Installation of Extended Services

```

DEFINE_LOCAL_CP  FQ_CP_NAME(USIBM.TIM)
                  CP_ALIAS(TIM)
                  NAU_ADDRESS(INDEPENDENT_LU)
                  NODE_TYPE(NN)
                  NODE_ID(X'513EB')
                  HOST_FP_SUPPORT(NO);

DEFINE_LOGICAL_LINK  LINK_NAME(LINK0001)
                     ADJACENT_NODE_TYPE(LEARN)
                     DLC_NAME(SDLC)
                     ADAPTER_NUMBER(0)
                     CP_CP_SESSION_SUPPORT(YES)
                     ACTIVATE_AT_STARTUP(YES)
                     LIMITED_RESOURCE(USE_ADAPTER_DEFINITION)
                     LINK_STATION_ROLE(USE_ADAPTER_DEFINITION)
                     SOLICIT_SSCP_SESSION(NO)
                     EFFECTIVE_CAPACITY(USE_ADAPTER_DEFINITION)
                     COST_PER_CONNECT_TIME(USE_ADAPTER_DEFINITION)
                     COST_PER_BYTE(USE_ADAPTER_DEFINITION)
                     SECURITY(USE_ADAPTER_DEFINITION)
                     PROPAGATION_DELAY(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_1(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_2(USE_ADAPTER_DEFINITION)
                     USER_DEFINED_3(USE_ADAPTER_DEFINITION);

DEFINE_LOCAL_LU  LU_NAME(LU5250 )
                 LU_ALIAS(LU5250 )
                 NAU_ADDRESS(INDEPENDENT_LU);

DEFINE_PARTNER_LU  FQ_PARTNER_LU_NAME(USIBM.AS400)
                  PARTNER_LU_ALIAS(PLU5250)
                  MAX_MC_LL_SEND_SIZE(32767)
                  CONV_SECURITY_VERIFICATION(NO)
                  PARALLEL_SESSION_SUPPORT(YES);

DEFINE_MODE  MODE_NAME(QPCSUPP )
             COS_NAME(#CONNECT)
             DEFAULT_RU_SIZE(NO)
             MAX_RU_SIZE_UPPER_BOUND(2048)
             RECEIVE_PACING_WINDOW(7)
             MAX_NEGOTIABLE_SESSION_LIMIT(32767)
             PLU_MODE_SESSION_LIMIT(7)
             MIN_CONWINNERS_SOURCE(3);

DEFINE_DEFAULTS  IMPLICIT_INBOUND_PLU_SUPPORT(YES)
                  DEFAULT_MODE_NAME(BLANK)
                  MAX_MC_LL_SEND_SIZE(32767)
                  DIRECTORY_FOR_INBOUND_ATTACHES(*)
                  DEFAULT_TP_OPERATION(NONQUEUED_AM_STARTED)
                  DEFAULT_TP_PROGRAM_TYPE(BACKGROUND)
                  DEFAULT_TP_CONV_SECURITY_RQD(NO)
                  MAX_HELD_ALERTS(10);

START_ATTACH_MANAGER;

```

Figure 4. Modified Node Definitions File

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

Controller description	CTLD	TIM
Link Type	LINKTYPE	*SDLC
Online at IPL	ONLINE	*YES
Switched Line	SWITCHED	*YES
Short hold mode	SHM	*NO
APPN capable	APPN	*YES
Switched line list	SWTLINLST	SDLCLIN1
Maximum frame size	MAXFRAME	*LINKTYPE
Remote network identifier	RMTNETID	*NETATR
Remote control point name	RMTCPNAME	TIM
Exchange identifier	EXCHID	05D513EB
Initial connection	INLCNN	*ANS
Connection number	CNNNBR	
Data link role	SWITCHED	*NEG Bottom

F3=Exit F4=Prompt F5=Refresh F10=Additional Parameters F12=Cancel
F13=How to use this display F24=More keys

Create Ctl Desc (APPC) (CRTCTLAPPC)

Type choices, press Enter.

Station address	STNADR	03
APPN CP session support	CPSSN	*YES
APPN node type	NODETYPE	*CALC

Bottom

F3=Exit F4=Prompt F5=Refresh F10=Additional Parameters F12=Cancel
F13=How to use this display F24=More keys

Figure 5. AS/400 Controller Description

Installing AS/400 PC Support
(Local Area Network)

Select options, press Enter.

PCS directory drive	[C]
Drive the personal computer starts from	[C]
PC location name	[USIBM.PC01]
System name	[AS400]
System LAN address	[400000000001]
Which functions Press the space bar to select will you want started?	Organizer Workstation Function Message Function Virtual Printer Session Manager

Enter Esc=Cancel F1=Help F3=Exit Spacebar

Figure 6. PC Support Installation

- Change APPN CP SESSION SUPPORT to *YES.
- Change APPN NODE TYPE to *CALC.

After stopping and restarting the Communications Manager, verify that you still can connect to the AS/400 using 5250 emulation.

Configuring Individual Workstations

To install 5250 emulation or PC Support/400 on DOS and OS/2 workstations, follow the procedures used to connect them directly to an AS/400 on a LAN. However, when prompted for the location of the AS/400, specify the LAN address of the Extended Services computer. For example, when installing PC Support/400 for DOS, the System LAN Address in Figure 6 is the Token-Ring address of the Extended Services computer.

There is no need to create unique controller descriptions on the AS/400 for each workstation. Each workstation is a device attached to the Extended Services computer.

Reference

- Glowacki, Janice R.; Knight, Stephen A.; and Strom, Desiree S. *Cooperative Processing with AS/400 PC Support*. New York: Van Nostrand Reinhold, 1991.

Tim Huntley is a senior associate programmer in Networking Systems Architecture. He was a member of the IBM SAA Networking Services/2 development team and is currently assisting vendors who are writing APPC applications. He joined IBM in 1989 in Research Triangle Park where he was involved with APPN architecture. He holds a BS in computer science from North Carolina State University.

Virus Information and Protection

Darrel Knaus
D&M Consulting
Colleyville, Texas

This article describes computer viruses, the extent of the problem, ways to prevent and manage virus infections, types of viruses, and antivirus tools and services. It also provides information about IBM antivirus programs and services available to PC users.

Viruses have recently received much attention. They spread themselves (or modified versions of themselves) without the knowledge or permission of users. They can cause damage or annoy users, which makes viruses a significant concern. Because they are programs, they can do anything a program can do. Unlike other programs, they are specifically designed to spread themselves. Viruses often spread without any visible signs to the user. This is called the *infection stage*. The other stage, the *trigger stage*, can occur randomly or because of a specific event, such as number of executions, date, and so on. The virus executes any instructions included by its author. These two stages can occur frequently, often during the same execution of the virus. Depending on what the writer desired, the virus code will not do any intentional harm (although some viruses have bugs that inadvertently cause damage), or it will damage other programs and data so that without backup, all programs and data will be lost.

The Extent of the Problem

The number of known viruses has grown from a handful in 1988 to over 1,000 today, with no indication that the growth will slow down soon. Although there are no known viruses outside the PC and Mac[®] environments, the ability to create virus-type code exists in virtually all computer platforms from the PC to large main-

frames. Viruses exist and have grown on desktop systems primarily for two reasons: the openness of the PC architecture, and an operating system that is controlled by the end user, with little or no security. LANs and interconnected systems also help spread viruses. Because an infected program can be available to many users, each access by a user can spread the virus to other programs on the user's system. Without proper control, this spread can become an epidemic that can virtually cripple a business if it does not have an organized approach to virus incidents or has no antivirus programs or service.

Dealing with the Virus

Viruses present a unique security problem. These programs have grown in sophistication and in the methods they use to avoid detection. While news media may publish a dramatic incident implying doom and gloom from these programs, it is very unlikely. However, users must take steps to deal with computer viruses every day.

Today, the best way to prevent viruses is to manage the computers and staff so that it becomes difficult for viruses to enter and spread among computers. If there were no sharing of programs and information, it would be relatively easy to prevent a virus infection. However, this is not practical. Business systems run on shared data, and there will be more sharing in the future.

Several steps should be taken to provide protection from viruses while also allowing for the flexibility needed by your organization.

- **Check all media entering the business.** This is designed to detect viruses as they enter and to remove them before they spread.
- **Limit the number of people who have write access to LAN servers.** Keep a log of who has access, and keep the log current. Where possible, prevent executable files on LAN file servers from being modified by client PCs.
- **Keep diskettes write-protected.** Most virus infections spread from one system to another via diskettes that have become infected.
- **Make frequent backups.** This is the most important step to protect the assets of your company. Although you cannot completely prevent viruses, the restoration of programs and data is critical. Check the system before making the backup to ensure that it is virus-free.
- **Organize a Computer Emergency Response Team (CERT) to take action when a virus incident occurs.** This team should know what viruses are, the virus types you are dealing with, how to prevent further spread, and how to isolate and clean up the virus. IBM provides several manuals and education courses to help with this.

The ideal goal is to prevent viruses from entering the system. However, because of the nature of viruses and the rapid expansion in the number of viruses, this is not completely feasible. So a second objective is to detect new viruses that have entered, contain their spread, and remove them with minimal damage and disruption. This requires several steps:

1. **Use antivirus software regularly and keep it current.** This should include an active monitor that runs

periodically to ensure there are no viruses in memory, on hard disks, or on any diskettes that the users have in their possession. The tools also should include a passive monitor that detects viruses in programs before they are run and prevents programs from spreading viruses.

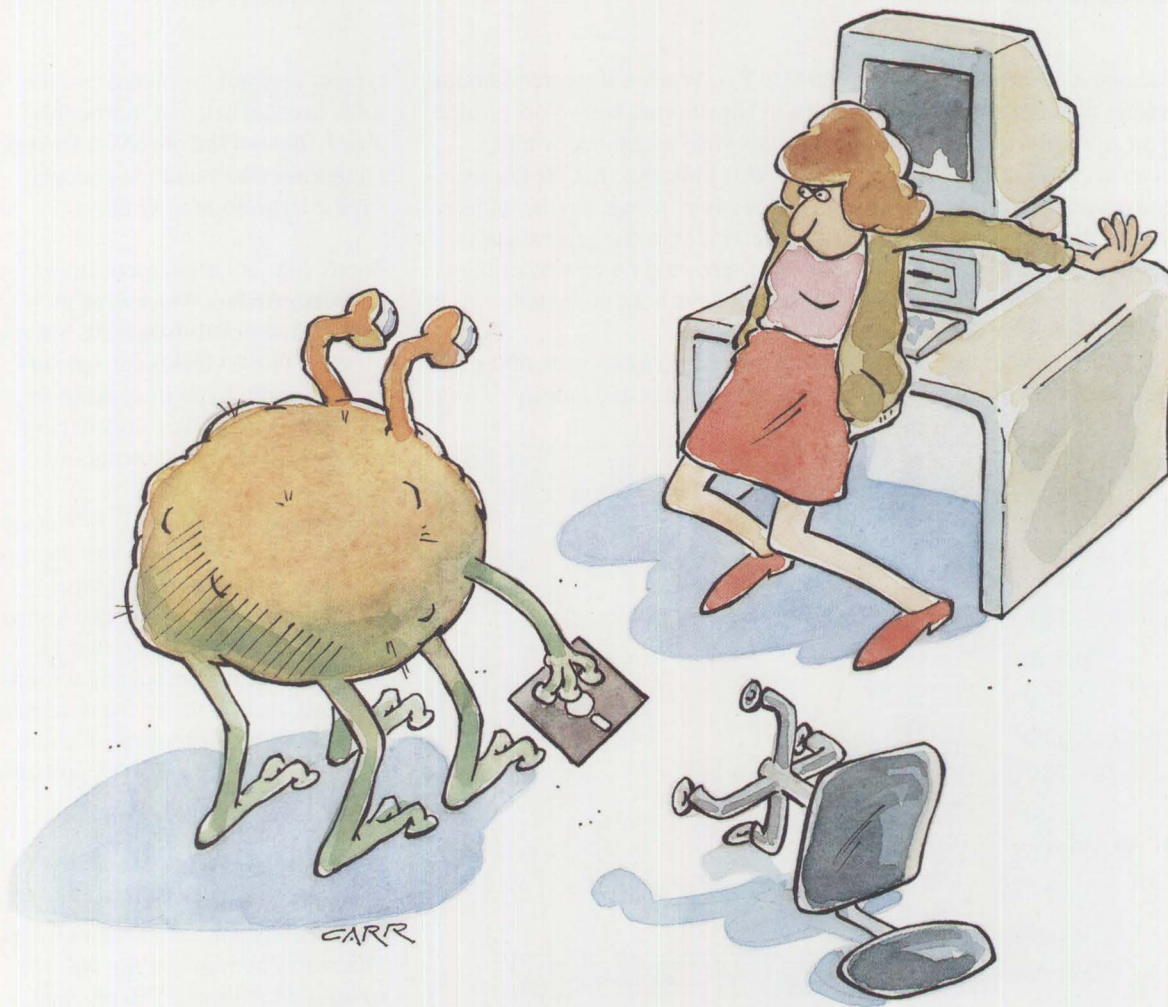
2. **Educate users about viruses, how they spread, and how to use antivirus software.** Users also should be trained on how to report an incident; they may fear reprisal

for having a virus. Maintain a penalty-free reporting environment to keep a minor virus incident from becoming a major infection.

3. **Provide a place for centrally reporting all virus incidents.** This group should be responsible for determining the size of the incident, actions to take, and when the incident can be closed.
4. **Get help!** Because of the growth in the number of viruses and the

sophistication of new viruses, it is difficult to be aware of the latest viruses and those that will be a threat. Several organizations provide assistance in dealing with viruses and updating your preventative methods.

Although the number of viruses is growing, it is not necessary to worry about all of them because the number observed in the wild is small. The remaining viruses are found only in various virus collections and have



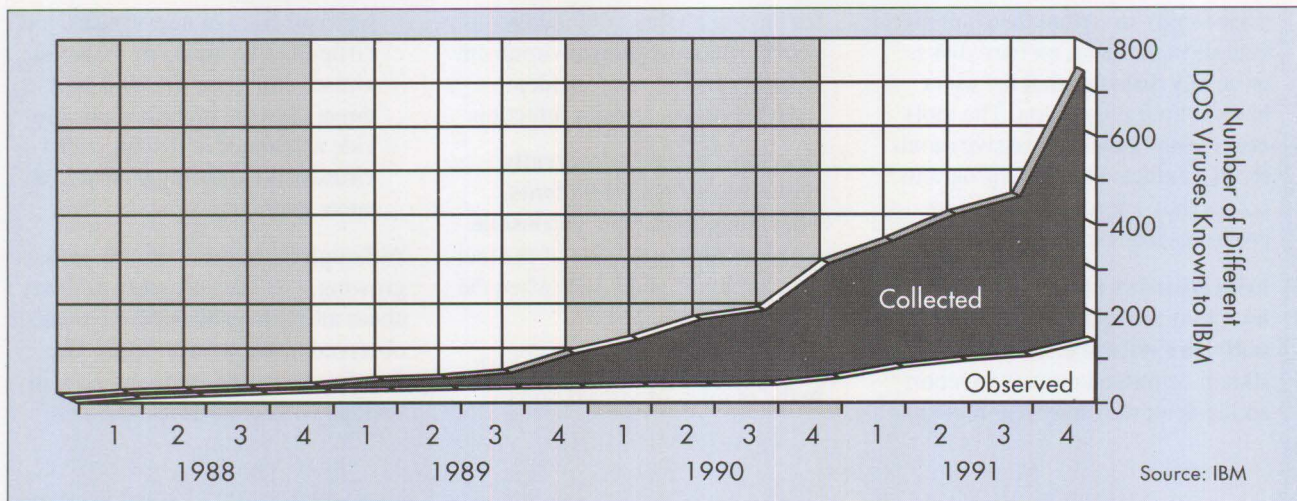


Figure 1. Extent of the Virus Problem

never been seen outside these collections. Therefore, the number of viruses that an antivirus program can detect is not an accurate measure of how good the program is if it cannot detect all viruses that are known to be spreading in the wild.

Figure 1 shows the relative size and growth of the virus problem in North America. This data was collected by

IBM's T. J. Watson Research Laboratory in Hawthorne, New York, which provides virus assistance within IBM. This indicates that the number of viruses is growing, but the number observed in the wild (general circulation) is growing much slower than those that have been collected.

When you begin looking at antivirus tools, consider the availability of this

type of research to ensure that the tools execute fast and thoroughly check files for the observed viruses. Do not spend much time checking for the less common viruses.

Focus your attention on the most common viruses. Most virus incidents consist of less than 20 different viruses. These viruses can spread quickly and remain in systems for some time before they do anything except spread to other programs.

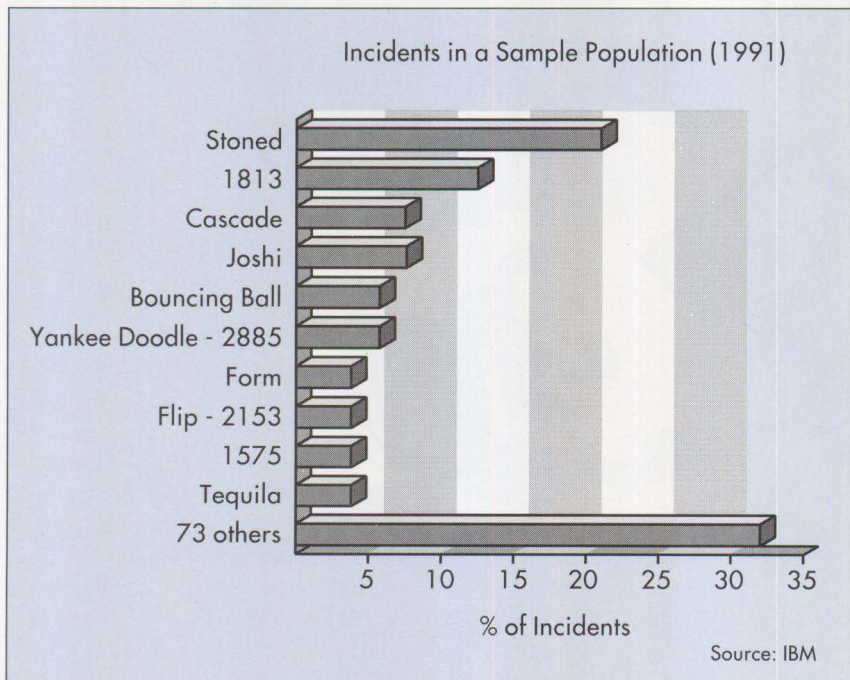


Figure 2. Frequency of Most Common Viruses

Figure 2, also from the T. J. Watson Research Laboratory, shows the most common viruses. Only 10 viruses account for about 70% of the known incidents. The bad news is that 73 viruses, while not accounting for many incidents each, make up the remaining 30% of incidents. Remember, even rare viruses can cause major problems if they infect your system.

Types of Viruses

There are two basic types of viruses: file infectors and boot sector viruses. The most common are *file infector viruses*, which spread by placing themselves in memory once an infected program is run and then infecting any new program that is loaded. This creates the potential for the system to continue to infect programs

until the system is rebooted. The user does not know the virus is resident, and the programs usually have no characteristics to suggest the presence of a virus. Usually these viruses infect either .EXE or .COM files.

The other type of virus—the *boot sector virus*—copies itself into the boot sector so that when the disk is booted, the virus is loaded first. Then it loads the original boot sector, which it moved to another location on the disk. Boot sector viruses are especially persistent and sometimes difficult to eradicate. This is because many of these viruses modify either the system boot or the master boot record and, in the process, corrupt data unintentionally. Disinfecting boot sector viruses is complex when the infected system is running security programs that lock or modify the boot sector or when running OS/2 or the OS/2 Boot Manager.

Boot sector viruses spread by booting from an infected disk or diskette. The best way to protect against boot sector viruses is to never boot from a diskette. Although this will prevent boot sector viruses, mistakes will always happen, such as accidentally powering on the system with a diskette that was left in the drive. Booting from this diskette, if infected, will infect the hard disks on the system.

If you must boot from a diskette, use antivirus tools to check the diskette before booting. Note that when a system's hard drive becomes contaminated with a boot sector virus, it will infect the boot sector of any other diskette inserted into the system. While it is easier to prevent boot sector viruses from entering your system, once they have infected the hard drive, they are usually more difficult to remove. Sometimes boot sector viruses move the original boot sector to an already used area of the disk, destroying data or file pointers such as the File Allocation Table (FAT). If

this data is corrupted, you will lose files and data. Remember to back up frequently!

File-infecting viruses place themselves in memory and infect other programs on disks or diskettes to which they have access. Because it is common to share programs with other users, especially within a business environment, this type spreads more easily than boot sector viruses. An infected program can easily spread to many systems before it is discovered.

Users need several types of virus protection tools to effectively detect and eradicate viruses.

Antivirus Tools

Most antivirus tools provide protection for both types of viruses and also handle the new Stealth and Polymorphic viruses. These new viruses use sophisticated techniques to hide from antivirus tools or to disguise themselves. Viruses employ techniques such as encryption and self-modifying code to reduce the chance of detection.

Users need several types of virus protection tools to effectively detect and eradicate viruses. The basic tools include an active scanner and a passive Terminate-and-Stay-Resident (TSR) program that detects when an infected program is about to be executed and prevents the virus from spreading. Most commercial tools employ both types of techniques. The *scanner* uses a specific string of characters called a *virus signature* to check programs for known viruses. Since the number of viruses is con-

stantly growing, it is important for the vendor to provide updates or new virus signatures regularly. Without these updates, the scanner soon becomes unable to detect and protect against newer and more complex viruses. Scanners also have begun to use techniques such as heuristics to detect new or unknown viruses by watching for behavior that is characteristic of some classes of known viruses.

It is important for the scanner not to give false positives, such as notifying users that they have a virus when the virus does not actually exist. These false positives create unnecessary work and concern. Antivirus tools, such as the IBM AntiVirus Scanner, provide signatures of both known and analyzed viruses, as well as heuristics to detect possible unknown viruses. No tool will detect all viruses that will be written. This is not a limitation of technology but a recognition of the amount of energy going into creating undetectable viruses.

TSR-based tools prevent the virus code from executing, and therefore spreading to other programs. IBM DOS Shield is one such program that is uniquely able to remove the virus code from memory, enabling the user to safely execute the infected program. Although this can be done, it is recommended that users immediately disinfect their systems before any other programs are run.

Another kind of antivirus program is a *disinfector*, which removes the virus from infected files or boot sectors. Although these are occasionally incorporated into scanners or TSRs, it is usually better to have them as a separate program. This is because the scanner usually scans only for a partial signature or suspicious activity so it can scan the files in a reasonable amount of time. The disinfector should always compare the signature found

in the suspected infected file or boot sector to the complete signature on a byte-by-byte basis before attempting to remove the virus. This ensures that the infection is not a variant where the removal technique could damage the program.

Deployment of Antivirus Tools

Users must decide whether to install antivirus tools on all systems or only on a subset of systems or critical systems. This is similar to deciding how much insurance to buy—you are weighing the risk of damage from viruses against the cost of deploying the tools. Another consideration is the cost of cleaning up the virus versus the cost of the tools. Recently, virus tools have lowered in price, and the cost of cleanup is increasing since this is labor intensive. In a major outbreak, the odds of reinfection are great. Some vendors provide a “full-service offering,” including advice and consulting during a virus incident, assistance with new or unknown viruses, volume discount on antivirus software, and access to a bulletin board for virus alert information and documentation. This offering lowers the cost of protecting all machines in a large business and includes automatic updates to the antivirus programs.

Whatever decision you make, it is critical that servers be protected. Servers may never become infected themselves, but they can store infected programs that can rapidly infect many clients. TSRs are not as useful on a server since the server may never execute an infected program, but scanners should be used regularly. However, if TSRs are used on all requesters, they provide some protection since they will prevent the spread and notify the user of the infected program.

IBM Tools

IBM has two antivirus programs: IBM AntiVirus/DOS and IBM AntiVirus/2. They provide state-of-the-art protection for DOS, Windows, and OS/2 environments. The IBM AntiVirus/DOS covers DOS and Microsoft Windows. It includes a file modification detector, a scanner, a disinfecter, and the TSR program called IBM DOS Shield. AntiVirus/DOS uses an “install and forget” technique. It can be set to automatically run at predetermined times, protecting the system even when the user forgets to run it.

IBM AntiVirus/2 is an OS/2 program similar to IBM AntiVirus/DOS. Although there are no known OS/2 viruses at this time, DOS viruses can infect programs through OS/2's DOS mode. Some boot sector viruses can infect High-Performance File Systems (HPFSs) by corrupting the data, causing the OS/2 system to be unable to boot, or destroying data.

IBM provides a comprehensive AntiVirus Service that uses these tools. The service also provides enterprise licenses, access to a toll-free virus bulletin board, automatic updates to the antivirus programs, and advice and consultation when a virus incident occurs. IBM will reverse engineer any newly discovered virus reported by customers within 48 to 72 hours of its detection and will add the signature to the detection programs.

To order the IBM AntiVirus programs and updates, call toll-free (800) 551-3579. For information about the IBM AntiVirus Service, call (800) 742-2493.

More Information

- Cohen, Fred. “Computer Viruses: Theory and Experiment.” *Com-*

puters and Security, Vol. 6 (1987) pp. 22–35. This is the first paper that defined viruses and related topics.

- *Communications of the ACM*, Vol. 32 No. 6 (June 1989). Contains several good articles on the Internet Worm incident.
- Dataquest. *Computer Virus Market Survey for the National Computer Security Association*. 1991. Recent survey about viruses and antivirus tools.
- Hoffman, Lance J. (editor). *Rogue Programs: Viruses, Worms, and Trojan Horses*. New York: Van Nostrand Reinhold, 1990. ISBN 0-442-00454-0. This book has several articles about many aspects of the virus problem.
- Kephart, J. O. and White, S. R. “Directed-Graph Epidemiological Models of Computer Viruses.” *Proceedings of the 1991 IEEE Computer Society Symposium on Research in Security and Privacy*, Oakland, California, May 20–22, 1991. pp. 343–359. This article was written by members of the IBM T. J. Watson Research Laboratory who support the IBM antivirus tools.

Darrel Knaus, 4007 Ashley Court, Colleyville, TX 76034. Darrel is an independent consultant specializing in information asset protection and antivirus protection. He retired from IBM after more than 30 years of service, holding many technical and management positions. Darrel has a BS in mathematics from Oklahoma State University and a master's degree in information and computer science from Georgia Institute of Technology.

Migrating from APPC/PC to Networking Services/DOS

John Q. Walker II
IBM Corporation
Research Triangle Park, North Carolina

This article describes how to migrate APPC/PC application programs to their CPI-C equivalents for use with Networking Services/DOS.

Networking Services/DOS (NS/DOS) is a new communications software package that was shipped by IBM in August 1992. It offers the Common Programming Interface for Communications (CPI-C) running atop an Advanced Program-to-Program Communications (APPC) Low Entry Networking (LEN) node. It runs with DOS Version 3.3 or later, or DOS with Microsoft Windows Version 3.0 or later. NS/DOS supports many Data Link Controls (DLCs), including Token Ring and Ethernet LANs, 3270 coax cards, Synchronous Data Link Control (SDLC), and asynchronous modems. It requires comparatively little RAM and uses DOS upper memory if available.

CPI-C (pronounced "sip'-ick") is a library of calls easily used to create portable communicating programs. CPI-C is available on all types of computer systems, from DOS, Windows, and OS/2 personal computers to major midrange and mainframe systems, as well as UNIX workstations. CPI-C offers the advantages of high performance, high flexibility, and high reliability. It also offers programmers a high degree of portability for their communications code. The CPI-C Application Programming Interface (API) has been published by the X/Open standards group and is available on many IBM and non-IBM computers.

CPI-C applications that use NS/DOS can be painlessly migrated to OS/2. If character-based programs are developed as family-API applications, a single .EXE program will run on both NS/DOS and the OS/2 Communications Manager. Similar portability can be achieved between Windows and the OS/2 Presentation Manager by using CPI-C calls and one of several packages (such as XVT[®] from XVT Software) to achieve portability between Windows calls and OS/2 Presentation Manager calls.

NS/DOS was developed by starting with the PC Support/400 "router," fixing its Logical Unit (LU) 6.2 architecture deviations, and adding the CPI-C support from the OS/2 Communications Manager. Several extensions beyond those required to achieve architectural correctness also were added. For example, NS/DOS includes single session support, dynamic numbers of partner LUs and sessions, user-defined Transaction Programs (TPs), predefined and user-defined modes, and a program launcher for incoming Attaches (when running with Windows).

The NS/DOS development team performed extensive testing and had the field experience of millions of PC Support/400 users. Testing included the LU 6.2 compliance tests, millions of CPI-C calls with severe random testing, connectivity testing with many partner LUs and partner link

stations, and several DOS and Microsoft Windows applications and hardware configurations.

NS/DOS was not designed to replace IBM's earlier APPC/PC product for DOS; the more natural migration path for APPC/PC applications is to OS/2's APPC verb control block interface. However, to gain the advantages of CPI-C, begin now to convert applications written to the APPC/PC interface to the open, portable CPI-C interface.

Source code changes must be made to programs written to APPC/PC for them to use NS/DOS. Usually, the changes are limited to coding instead of design changes. Following is a summary of the changes between APPC/PC and NS/DOS:

- APPC/PC's requests for APPC services are changed from a DOS system interrupt (Int 68) to a CPI-C call interface. The call interface is reached by linking a small CPI-C library to the application program. The NS/DOS documentation also describes an optional interrupt interface to the CPI-C API.
- Instead of fields in a verb control block, information is supplied and returned as parameters of the CPI-C call. The Pascal calling convention is used for exchanging parameters with CPI-C on the stack. Since all requests to CPI-C are made by calls, the AH register request code and AL register return code used by APPC/PC are not employed.
- APPC/PC's programmer-supplied application subsystem is changed to simple ASCII configuration files in NS/DOS.
- APPC/PC requires programmers to write code to convert Systems Network Architecture (SNA) names (such as LU, TP, and mode names) from ASCII to EBCDIC when supplying them as parameters on verbs. This conversion is

not required in NS/DOS; with CPI-C, SNA names are supplied in their native ASCII.

- APPC/PC requires programmers to write application code to handle incoming Attaches. With NS/DOS, an attach manager is supplied. With Windows active, the attach manager can start programs; with only DOS active, the attach manager holds requests until an `Allocate` is issued to handle them.
- No APPC user exits are available to NS/DOS application programs.
- Only a single local LU is available in NS/DOS. This is not an undue restriction since all CPI-C (and APPC) applications should use the default local LU. An associated Physical Unit (PU) in each workstation has the same fully qualified name as the local LU.
- NS/DOS can communicate only with independent partner LUs.
- No LU-LU security is available in NS/DOS. Full conversation security capabilities are available, matching the CPI-C counterparts in Extended Services for OS/2.

Using CPI-C

Using CPI-C is easy for programmers. For example, for C programmers, there is a `.H` header file to be included when compiling a program and a `.LIB` library with which to link. NS/DOS has many useful example programs (with source code) to shorten the time needed to begin producing useful CPI-C applications.

Although APPC/PC can be used from any programming language that can issue DOS interrupts, only header files for the PC macro assembler are packaged with it. CPI-C can be called from any programming language with a call-return interface. NS/DOS is shipped with header files for C and COBOL programming languages. No assembler files are furnished with

NS/DOS. Assembler programs are not expected to use CPI-C since assemblers are not portable between machines with different hardware architectures.

CPI-C provides an extensive set of defaults. It is much easier to start writing distributed applications using these defaults instead of APPC/PC's verb control block interface. For example, the default conversation type is mapped; if mapped conversations are used by an application (and they should be), no additional calls are needed. However, if an application uses basic conversations, it must explicitly issue a `Set_Conversation_Type` (CMSCT) call, setting the conversation type to the constant named `CM_BASIC`. When moving from APPC/PC to CPI-C, be sure to issue the appropriate `Set_` calls for any default values that must be changed. For information about CPI-C, see the current *IBM SAA CPI-C Reference* (SC26-4399). For additional books, see the references at the end of this article.

There are two classes of verbs in APPC/PC: those related to actual data exchange (conversation verbs) and those related to the underlying runtime environment (application subsystem verbs). The remainder of this article is divided into two corresponding sections showing the mapping from APPC/PC to NS/DOS for these two classes of verbs.

APPC/PC Conversation Verbs

The following verbs are related to conversations; each verb either supplies or returns a conversation ID as a parameter.

ALLOCATE and

MC_ALLOCATE: These verbs correspond with a sequence of at least two calls in CPI-C, `Initialize_Conversation` (CMINIT) and `Allocate` (CMALLC). Before issuing the `Allocate` call in CPI-C, a

program has the option of issuing one or more of the following calls to set the parameters of `Allocate` to values other than the default:

- **CMSCT:**
`Set_Conversation_Type`
- **CMSMN:** `Set_Mode_Name`
- **CMSPLN:**
`Set_Partner_LU_Name`
- **CMSRC:** `Set_Return_Control`
- **CMSL:** `Set_Sync_Level`
- **CMSTPN:** `Set_TP_Name`

CONFIRM and MC_CONFIRM:

These verbs correspond with the CPI-C `Confirm` (CMCFM) call.

CONFIRMED and

MC_CONFIRMED: These verbs correspond with the CPI-C `Confirmed` (CMCFMD) call.

DEALLOCATE and

MC_DEALLOCATE: These verbs correspond with the CPI-C `Deallocate` (CMDEAL) call.

Before issuing the `Deallocate` call, a program has the option of issuing one or more of the following calls to set the parameters of `Deallocate` to values other than the default:

- **CMSDT:** `Set_Deallocate_Type`
- **CMSLD:** `Set_Log_Data` (basic conversations only). The maximum log data buffer size in APPC/PC is 65,535 bytes; in CPI-C, the maximum log data length is 512 bytes.

FLUSH and MC_FLUSH:

These verbs correspond with the CPI-C `Flush` (CMFLUS) call. The `Set_Send_Type` (CMSST) and `Set_Deallocate_Type` (CMSDT) calls also can be used to combine `Flush` with `Send_Data` and `Deallocate`, respectively.

GET_ALLOCATE: This verb corresponds with the CPI-C `Accept_Conversation` (CMACCP) call. The `GET_ALLOCATE` verb

enables programmers to control the handling of incoming Attaches with their own procedures. NS/DOS furnishes its own code for handling incoming Attaches.

GET_ATTRIBUTES and MC_GET_ATTRIBUTES: Three parameters can be extracted using CPI-C calls, `Extract_Mode_Name` (CMEMN), `Extract_Partner_LU_Name` (CMEPLN), and `Extract_Sync_Level` (CMESL). The local LU name and user ID fields are not available using CPI-C calls.

GET_TYPE: This verb corresponds to the CPI-C `Extract_Conversation_Type` (CMECT) call.

POST_ON_RECEIPT: Not supported. NS/DOS currently offers no provision for posting received information.

PREPARE_TO_RECEIVE and MC_PREPARE_TO_RECEIVE: These verbs correspond with the CPI-C `Prepare_To_Receive` (CMPTR) call. Before issuing the `Prepare_To_Receive` call, a program has the option of issuing the following call, which affects the `Prepare_To_Receive` call's function:

- **CMSPTR:**
`Set_Prepare_To_Receive_Type`

RECEIVE_AND_WAIT and MC_RECEIVE_AND_WAIT: These verbs correspond with the CPI-C `Receive` (CMRCV) call. Before issuing the CPI-C `Receive` call, the `receive_type` must be set to `CM_RECEIVE_AND_WAIT` using the `Set_Receive_Type` (CMSRT) call. The maximum data buffer size in APPC/PC is 65,535 bytes; in CPI-C, the maximum requested length is 32,767 bytes. Before issuing the `Receive` call, a program has the option of issuing the following call, which affects the `Receive` call's function:

- **CMSF:** `Set_Fill` (basic conversations only)

RECEIVE_IMMEDIATE and MC_RECEIVE_IMMEDIATE:

These verbs correspond with the CPI-C `Receive` (CMRCV) call. Before issuing the `Receive` call, the `receive_type` must be set to `CM_RECEIVE_IMMEDIATE` using the `Set_Receive_Type` (CMSRT) call. The maximum data buffer size in APPC/PC is 65,535 bytes; in CPI-C, the maximum requested length is 32,767 bytes. Before issuing the `Receive` call, a program has the option of issuing the following call, which affects the `Receive` call's function:

- **CMSF:** `Set_Fill` (basic conversations only)

REQUEST_TO_SEND and MC_REQUEST_TO_SEND: These verbs correspond with the CPI-C `Request_To_Send` (CMRTS) call.

SEND_DATA and MC_SEND_DATA: These verbs correspond with the CPI-C `Send_Data` (CMSEND) call. The maximum data buffer size in APPC/PC is 65,535 bytes; in CPI-C, the maximum send length is 32,767 bytes. Before issuing the `Send_Data` call, a program has the option of issuing the following call, which affects the `Send_Data` call's function:

- **CMSST:** `Set_Send_Type`
- **CMSPTR:**
`Set_Prepare_To_Receive_Type`
- **CMSDT:** `Set_Deallocate_Type`

SEND_ERROR and MC_SEND_ERROR: These verbs correspond with the CPI-C `Send_Error` (CMSERR) call. Before issuing the `Send_Error` call, a program has the option of issuing the following call, which affects the `Send_Error` call's function:

- **CMSD:** `Set_Error_Direction`

- **CMSLD:** `Set_Log_Data` (basic conversations only). The maximum log data buffer size in APPC/PC is 65,535 bytes; in CPI-C, the maximum log data length is 512 bytes.

TEST and MC_TEST: These verbs correspond with the CPI-C `Test_Request_To_Send_Received` (CMTRTS) call for the `REQUEST_TO_SEND_RECEIVED` returned parameter value. Since CPI-C offers no provision for posting received information, `POSTED` is not supported in CPI-C.

WAIT: Not supported. NS/DOS currently offers no provision for posting received information.

APPC/PC Application Subsystem Verbs

APPC/PC leaves configuration and setup as an exercise for the systems programmer. These functions are more user-friendly in NS/DOS, where they are handled with simple ASCII configuration files.

The number of changes needed to migrate a TP to NS/DOS depends in part on whether these application subsystem functions are separate from the TP functions. If the functions are separate, fewer changes are needed.

ACCESS_LU_LU_PW: Not supported. NS/DOS currently offers no provision for session-level security.

ACTIVATE_DLC: This verb corresponds to the following two network configuration file keywords in NS/DOS.

- **DLC_NAME:** Use the NSDC keyword in the network configuration file to specify the DLC type.
- **ADAPTER_NUMBER:** When using a LAN adapter, use the TRAN keyword to specify the adapter number.

The DLC is activated when the network configuration file is read as part of the NSD START command.

ATTACH_LU: This APPC/PC verb is used to define a local LU, a set of partner LUs, and the modes used to reach those partners. Only a single local LU can be defined in NS/DOS.

- **LU_NAME:** Use the NSDN keyword in the network configuration file to specify the fully qualified local LU name.
- **LU_LOCAL_ADDRESS:** NS/DOS currently supports only independent LUs; dependent LUs are not supported. Therefore, only the value X'00' from APPC/PC is supported.
- **LU_SESSION_LIMIT:** Not supported. The available memory limits the maximum number of active sessions in NS/DOS.
- **CREATE_TP_EXIT:** Supported by entries in the DEFINETP.NSD file.
- **SYSTEM_LOG_EXIT:** All NS/DOS messages are logged in the MSGLOG.NSD file.
- **LU_LU_PASSWORD_EXIT:** Not supported. NS/DOS currently offers no provision for session-level security.
- **MAX_TPS:** Not supported by NS/DOS.
- **QUEUE_ALLOCATES:** Supported by entries in the DEFINETP.NSD file.
- **QUEUE_DEPTH:** Supported by entries in the DEFINETP.NSD file.
- **PARTNER_LU_NAME:** There are three ways to specify the name of the partner LU with NS/DOS.
 - As an entry in the SIDEINFO.NSD file
 - As a parameter on a Set_Partner_LU_Name CPI-C call

— As an option on the NSD ACTIVATE command

- **PARTNER_LU_SESSION_LIMIT:** Not supported by NS/DOS. In NS/DOS, session limits are specified by mode, not by partner LU.
- **PARTNER_LU_MAX_MC_SEND_LL:** Not supported. In CPI-C, the maximum send length is always 32,767 bytes.
- **PARTNER_LU_DLC_NAME:** Only one DLC at a time can be used with NS/DOS. Use the NSDC keyword in the network configuration file to specify the DLC name.
- **PARTNER_LU_ADAPTER_NUMBER:** Use the TRAN keyword in the network configuration file to specify the adapter number.
- **PARTNER_LU_ADAPTER_ADDRESS:** Use the TRLD keyword in the network configuration file to specify the remote adapter address.
- **PARTNER_LU_SECURITY_CAPABILITIES:** Not supported. NS/DOS currently offers no provision for session-level security.
- **MODE_NAME:** Supported by entries in the MODE.NSD file. Whenever possible, CPI-C programs should use one of five predefined mode names: #INTER, #BATCH, #INTERSC, #BATCHSC, or all blanks.
- **MAX_RU_SIZE:** Supported by entries in the MODE.NSD file.
- **PACING_SIZE:** Supported by entries in the MODE.NSD file.
- **MODE_MAX_NEGOTIABLE_SESSION_LIMIT:** Supported by entries in the MODE.NSD file.

ATTACH_PU: NS/DOS provides a single local LU. It is associated with a PU of the same fully qualified name.

- **PU_NAME:** Not separately supported in NS/DOS. Use the local LU name specified on the NSDN keyword in the network configuration file.
- **NET_NAME:** Use the NETID portion of the fully qualified local LU name specified on the NSDN keyword in the network configuration file.
- **SYSTEM_LOG_EXIT:** All NS/DOS messages are logged in the MSGLOG.NSD file.
- **RETURN_CONTROL:** Not supported by NS/DOS.

CHANGE_LU: Not directly supported by NS/DOS.

CNOS: A Change Number of Sessions (CNOS) call is not required for most applications. To explicitly change the number of sessions in NS/DOS, use the NSD ACTIVATE and NSD DEACTIVATE commands. Otherwise, it is done implicitly when a session is required with a new partner LU and mode.

CONVERT: Not supported. This APPC/PC verb eases the conversion of SNA names from ASCII to their EBCDIC equivalent. With CPI-C, however, this conversion is not required. SNA names used by CPI-C in NS/DOS (such as LU, TP, and mode names) are supplied directly in native ASCII as parameters on their respective calls.

CREATE_TP: NS/DOS supplies its own code for handling incoming Attaches.

DETACH_LU: In NS/DOS, the single local LU is terminated by issuing the NSD STOP command.

DETACH_PU: In NS/DOS, the single local LU is terminated by issuing the NSD STOP command. Use NSD STOP /F to force sessions and conversations to end immediately.

DISABLE/ENABLE_APPC: In NS/DOS, use the NSD STOP /U command to force unused sessions to end immediately, to release all links, and to unload NS/DOS. If conversations are active on those sessions, the /F option also must be used. Use the NSD START command to initiate the operation of NS/DOS.

DISPLAY: In NS/DOS, use the NSD LIST command to display information about LUs, link stations, sessions, queued inbound Attaches, mode definitions, side information, and local SNA information.

PASSTHROUGH: Not supported by NS/DOS.

SET_PASSTHROUGH: Not supported by NS/DOS.

SYSLOG: All NS/DOS messages are logged in the MSGLOG.NSD file.

TP_ENDED: Not supported by CPI-C. CPI-C currently offers a one-to-one correspondence between TPs and conversations.

TP_STARTED: Not supported by CPI-C. CPI-C currently offers a one-to-one correspondence between TPs and conversations.

TP_VALID: Not supported by CPI-C.

TRACE: In NS/DOS, tracing is started and stopped by using the NSD TRACE command.

TRANSFER_MS_DATA: Not supported. NS/DOS currently offers no provision for sending network management data.

Alternative Method

If you are not absolutely ready to convert to CPI-C calls, there is an alternative. Buried inside NS/DOS is the

original APPC verb control block interface that was part of the PC Support/400 router. Since this API was originally based on the APPC/PC API, this may provide the easiest migration path. However, the APPC verb control block APIs have no future. They are different from product to product—even between these two products that are similar. It is strongly recommended that you migrate to CPI-C as quickly as possible.

Even with that, if you are still considering the APPC API that is embedded in NS/DOS, it is well-documented in the following two books:

- *PC Support/400: Application Program Interface Reference* (SC41-8254)
- Glowacki, Janice R., Knight, Stephen A., and Strom, Desiree S. *Cooperative Processing with AS/400 PC Support (DOS)*. New York: Van Nostrand Reinhold, 1991. ISBN: 0-442-00624-1.

Summary

NS/DOS offers significant speed, memory, and flexibility advantages over APPC/PC. Its open CPI-C programming interface gives you portability of source code and programming skills across many different computing platforms. The migration path from NS/DOS to the OS/2 Communications Manager is particularly easy using CPI-C, since you can build family-API programs.

If you are considering APPC for new DOS or Windows systems, NS/DOS should be your platform of choice. If you also have some existing APPC/PC programs that you plan to continue to maintain and modify, start today with your port to CPI-C using the guidelines previously discussed. Finally, the APPC forum on CompuServe (reached by entering "GO APPC") contains a whole section for ques-

tions related to CPI-C programming, as well as libraries of useful programs with source code. Discuss your questions with CPI-C experts and other users, 24 hours a day, worldwide.

References

- *IBM SAA CPI Communications Reference* (SC26-4399)
- *Networking Services/DOS Version 1.0 User's Guide and Reference* (S20G-0438)
- *APPC/PC Programming Guide* (75X1084)
- *IBM Extended Services for OS/2 APPC Programming Reference* (S04G-1025)
- *The APPC Resource Book* (G325-0055). If you are doing anything related to APPC, APPN, CPI-C, or LU 6.2, be sure to get this book. It contains extensive information about education, manuals, applications, development tools, and platforms.

John Q. Walker II manages architecture, development, and marketing departments responsible for APPC, CPI-C, and APPN. He recently managed one of the teams responsible for this software on OS/2. He joined IBM in Rochester, Minnesota in 1978 where he did software development and testing for the IBM System/38. In Research Triangle Park, John was an architect for the Token-Ring local area network and served as a co-editor of the corresponding IEEE 802.5 standard in 1983 and 1984. He received BS, BA, and MS degrees from Southern Illinois University and a PhD in computer science at the University of North Carolina.

OS/2 2.0 Resources

Many types of resource materials are available for OS/2 2.0 from IBM and other independent sources. Information is available in both electronic and printed formats. This article provides a selected list of recent publications including technical materials, books, current magazines devoted to OS/2, and white papers from IBM. Selected recent videos and bulletin boards for OS/2 also are provided. The publications listed in this article are separately available materials—those not included in the software package itself.

This article also provides ordering information for IBM and independent publications.

How to Obtain Publications

IBM documents and publications, CD-ROMs, brochures, videos, and other types of publications can be ordered by fax or telephone. The ordering process is shown in Figure 1. You can use a VISA, MasterCard, or Diners Club credit card; a check or money order; or your IBM customer number (in some cases) for your order. Publications also can be ordered through SLSS¹ by calling the numbers listed in Figure 1.

Prices for documents vary, but the average price of IBM publications is \$25. Orders can be placed via telephone from 6:30 A.M. to 6:00 P.M. mountain standard time.

IBM publications also can be ordered through IBM branch offices or your

IBM marketing representative. Some IBM publications (or subscriptions) are available through independent vendors. Telephone numbers are listed, when available, for ordering these publications.

Non-IBM publications are available through the source listed or at your local bookstore.

Planning Guides

IBM publishes product and planning guides for new products. These guides provide a comprehensive overview of the main features of the products but are not intended to be in-depth

instruction manuals. They provide extensive listings of additional information sources—both electronic and printed materials—available from IBM and other independent sources. These guides are an excellent source of information about the product and also serve as a good resource for identifying the most recent documentation available on the product. The guides that relate to OS/2 2.0 are as follows:

IBM OS/2 2.0 Information and Planning Guide (G326-0160). Provides an overview of OS/2 2.0 functions plus assistance in estimating memory and disk space requirements. It is recommended for anyone planning to install OS/2 2.0 on several machines.

IBM Extended Services for OS/2 Information and Planning Guide (G326-0161). Provides information

Ordering Documents via Fax:

- Obtain a form from your IBM branch office or create your own.
- Include the following information on your fax order:
 - Publication numbers for all items you are ordering
 - IBM customer number (if you have one)
 - Credit card type (VISA, MasterCard, Diners Club), number, expiration date, and name on the card
 - Name, phone number, and fax number of person placing the order
 - “Ship to” address, including the name of the person to receive the order
- Fax to (800) 284-4721, toll-free in the U.S.

Ordering Documents via Telephone:

- Have the information listed above available.
- Call (800) 879-2755, toll-free in the U.S. You will reach “IBM Software Manufacturing and Delivery.”
- Select option 1 and follow the directions.

Figure 1. Ordering IBM Publications

¹ IBM employees should continue to order publications through their own access to SLSS. For questions about how to access SLSS, call the number listed and choose option 1.

for those planning to install and use OS/2 2.0 Extended Services, including Communications Manager, Database Manager, and Query Manager.

IBM LAN Server Version 2.0 Information and Planning Guide (G326-0162). Provides product and planning information about OS/2 Local Area Network (LAN) Server Version 2.0. LAN Server 2.0 includes OS/2 LAN Server, OS/2 LAN Requester, DOS LAN Requester, LAN Adapter and Protocol Support, LAN Support Program, and various utility programs.

OS/2 Technical Compendium ("Red Books")

The five-volume *OS/2 Technical Compendium* provides comprehensive information about OS/2 2.0. It is published by the IBM International Technical Support Center in Boca Raton, Florida—the same location as the development laboratory for OS/2 2.0. It is essential reading for anyone needing an in-depth understanding of OS/2 2.0. The complete five-volume set can be ordered as *OS/2 Technical Compendium* (GBOF-2254), or separately as follows:

- *Volume 1: Control Program* (GG24-3730). Provides detailed information about the Control Program component of OS/2 2.0. Describes memory and task management, debugging support, and enhanced Application Programming Interfaces (APIs).
- *Volume 2: DOS and Windows Environment* (GG24-3731). Provides details about the Multiple Virtual DOS Machines (MVDM) feature and support for Microsoft Windows programs. Describes 8086 emulation, device drivers, extended memory support, DOS settings, using specific versions of DOS, and the architecture of MVDM.
- *Volume 3: Presentation Manager* (GG24-3732). Introduces the

Presentation Manager (PM) and describes the enhanced graphics appearance of the operating system and the enhanced help facilities. Discusses programming considerations for 32-bit or a mix of 16- and 32-bit code application development.

- *Volume 4: Application Development* (GG24-3774). Provides a general introduction to object orientation, modularization, naming conventions, and other structural considerations for programs designed to run in the OS/2 environment. Discusses programming and migration considerations for 32-bit, 16-bit, and mixed applications.
- *Volume 5: Print Subsystem* (GG24-3775). Describes the internal workings of the print subsystem, including the spooler.

Another important Red Book, *OS/2 Remote Installation and Maintenance* (GG24-3780), contains practical instructions and various approaches for installing OS/2 across a LAN. Two other books that supplement this one are as follows:

- *Automated Installation for CID-Enabled OS/2 Version 2.0* (GG24-3783)
- *Automated Installation for CID-Enabled Extended Services 1.0, LAN Server 3.0, and Network Transport Services/2* (GG24-3781)

Other Red Books that cover OS/2 and its extensions are as follows:

- *LAN Server 2.0: New Functions and Features* (GG24-3875)
- *NetWare from IBM: Network Protocols and Standards* (GG24-3890)
- *Extended Services for OS/2 Database Manager: New Features* (GG24-3794)
- *Developing a CUA Workplace Application* (GG24-3580-01)

- *Evaluation of OS/2 Application Development Tools* (ZZ81-0295)
- *Practical Introduction to Object-Oriented Programming* (GG24-3641)
- *Migrating from a DOS/Windows Environment to OS/2* (GG24-3822)
- *Multimedia Application Enablers and PS/2 Ultimeidia* (GG24-3749)
- *IBM Personal System/2 Multimedia Fundamentals* (GG24-3653)

Magazines

Several magazines dedicated primarily to OS/2 are listed below:

OS/2 Developer (previously *Personal Systems Developer*). The *OS/2 Developer* is a quarterly publication from IBM's Developer Assistance Program (DAP). The *Developer* contains articles on many OS/2 topics, including OS/2 2.0, system extensions such as the Database Manager and LAN Server, and material on OS/2 applications and development tools. Subscriptions for \$39.95 annually are available by calling (800) WANT-OS2 in the U.S. or (708) 647-5960 internationally.

OS/2 Monthly. This independent magazine is dedicated to news, views, and technical information about OS/2. It is published by JDS Publishing, P.O. Box 4351, Highland Park, NJ 08904. Phone: (800) 365-2642 or (908) 247-0952. Subscriptions are \$39 annually.

OS/2 Professional. This independent magazine for OS/2 professionals is published bimonthly and sent to qualified OS/2 users. Requests for qualified subscription forms can be sent to *OS/2 Professional*, 6129 Executive Boulevard, Rockville, MD 20852. Phone: (301) 770-7302. Fax: (301) 770-2327.

Personal Systems Technical Solutions. The IBM Personal Systems Competency Center in Dallas, Texas, publishes the *Personal Systems Technical Solutions* quarterly. *Solutions* covers both hardware and software, including many articles on OS/2 2.0. Subscriptions are available for *Solutions* (\$40 for one year and \$72 for two years) by calling (800) 551-2832 in the U.S. or (415) 948-3140 internationally. Fax: (415) 948-4280.

Newsletters

The Cobb Group produces a monthly newsletter called *Inside OS/2*. It is designed for computer novices or users with some OS/2 experience who want to learn more about OS/2 and get more from their OS/2 system. Subscriptions are available for \$39 annually from the Cobb Group, P.O. Box 35160, Louisville, KY 40232. Phone: (502) 491-1900.

White Papers

The following papers were first issued by IBM in April 1991; a second set appeared in January 1992. They provide an excellent guide to IBM's OS/2 strategy and future product directions.

OS/2 2.0 Considerations. An excellent guide to OS/2 2.0's features, market positioning, application migration, and future directions.

OS/2 System Performance Management. An overview of current OS/2 and LAN Server parameters and tuning facilities, as well as future directions.

OS/2 System Management. An overview of current IBM Systems Application Architecture (SAA) System-View facilities in OS/2 environments and future directions.

OS/2 Database Manager Highlights and Directions. An overview of current database offerings and future directions.

OS/2 Communications Manager Highlights and Directions. A detailed overview of the Communications Manager function.

OS/2 Performance Considerations. A guide to some of the issues involved in tuning performance for OS/2 2.0.

OS/2 2.0 Windows Environment. An explanation of how the Win-OS/2 environment is architected, and a description of the benefits of the environment versus Windows 3.x.

OS/2 LAN Server Positioning. A guide to the relative strengths and weaknesses of LAN Server versus Microsoft LAN Manager 2.0 and Novell NetWare 3.11.

OS/2 LAN Server Migration. A discussion of how to move both clients and servers to OS/2 LAN Server 2.0 from several previous IBM LAN products.

OS/2: The Bigger Picture. An overview of IBM's OS/2 strategy and future directions.

Upgrading to OS/2 2.0. A review of the various options in upgrading from DOS, Windows, or OS/2 1.3 to OS/2 2.0, and the requirements for each migration path.

Getting Started with the OS/2 Workplace Shell. A brief introduction to the capabilities of the Workplace Shell. (This is worth reading before starting on the shell for the first time.)

Client/Server Computing. An overview of various IBM products that addresses the client/server environment, including Extended Services for OS/2 and OS/2 LAN Server.

OS/2 2.0: The Development Platform of Choice. Reasons why OS/2 2.0 is an excellent target for developers,

including a review of the range of various development tools available.

White Papers can be obtained from your IBM representative or your IBM branch office. They are on IBM's internal MKTTOOLS disk as WPAPERS and WPAPERS2 packages. They can be copied and distributed to anyone who wants to understand OS/2.

OS/2 2.0 Technical Library

The entire technical library contains a series of publications on various OS/2 2.0 topics.

General Information

Application Design Guide (S10G-6260). Provides an overview of OS/2 programming concepts, including help using the System Object Model (SOM) to develop applications and create Workplace objects. Useful when building executable files or dynamic link libraries, when writing code for an object-oriented environment, or when migrating from DOS or OS/2 16-bit applications.

Programming Guide. This three-volume set includes the following:

- *Volume I* (S10G-6261). Describes the Control Program for programming functions that are internal to applications, including the file system, memory management, exception management, and multitasking.
- *Volume II* (S10G-6494). Describes the Presentation Manager windowed user interface, including messages and message queues, window classes, frame windows, control windows, and window controls. Describes how to write a PM application that conforms to Common User Access (CUA) guidelines.
- *Volume III* (S10G-6495). Describes the graphics programming interface, including graphics primitives, graphics segments, bitmaps, and

transformation functions. Describes printing and device support.

Information Presentation Facility (IPF) Guide and Reference (S10G-6262). Provides help in using the IPF tagging language and IPF compiler, and serves as a reference for window functions, dynamic data functions, and help manager messages.

System Object Model Guide and Reference (S10G-6309). Contains the same content as the online reference. It is for programmers experienced in developing object-oriented programs.

Control Program Programming Reference (S10G-6263). Provides a hardcopy of the online *Control Program Reference*.

Presentation Manager Programming Reference. This is a three-volume set as follows:

- *Volume I* (S10G-6264). Provides an alphabetical listing of the dynamic data format (Ddf), device (Dev), dragdrop (Drg), graphics (Gpi), profile (Prf), and spooler (Sp1) API functions.
- *Volume II* (S10G-6265). Provides an alphabetical listing of the window API functions and the new Workplace methods.
- *Volume III* (S10G-6272). Contains related information such as graphics-orders, data types, applications hooks and procedures, and PM messages.

REXX Information

Procedures Language 2/REXX User's Guide (S10G-6269). Contains two parts: Basics includes frequently used features; Advanced Topics describes special features and includes examples.

Procedures Language 2/REXX Programming Reference (S10G-6268). Describes the REXX functions supported by OS/2 2.0.

Device Driver References

Physical Device Driver Reference (S10G-6266). Provides category, function code, and calling conventions for I/O control (IOCTL) functions, including those needed for DevHlp routines.

Virtual Device Driver Reference (S10G-6310). Provides information on virtual DevHlp routines and describes virtual device driver architecture, operations, and inter-device driver communication. Includes a detailed description of each virtual device driver available with OS/2.

Presentation Driver Reference (S10G-6267). Describes the internal interface between the PM interface and the driver, and between the driver and the I/O subsystem. Contains information about queue and port drivers, and detailed descriptions of control structures, data structures, and I/O formats.

CUA Interface Design Guides

Systems Application Architecture: Common User Access (CUA) Guide to User Interface Design (SC34-4289). Describes the principles, components, and techniques of user-interface design in general and the process of designing a product with a CUA interface.

Systems Application Architecture: Common User Access Advanced Interface Design Reference (SC34-4290). Lists all fundamental and recommended guidelines for designing and developing a product with a CUA interface.

Related Information

OS/2 2.0 Keyboards and Code Pages (S10G-6312). Provides support to those who use code-page switching for files that are received from or sent to other countries.

OS/2 2.0 Command Reference (S10G-6313). Provides a hardcopy of the online version being shipped with the operating system. Describes how to use OS/2 commands and contains information about the syntax and purpose of each command.

OS/2 2.0 Using Bidirectional Support (S41G-8688). Describes bidirectional support, instructions and considerations for using bidirectional functions, and key assignments for and summaries of bidirectional functions.

OS/2 2.0 Remote Installation and Maintenance (GG24-3780). Provides instructions for installing OS/2 from a LAN.

Extended Services for OS/2

Extended Services components offer comprehensive communications support, a full-function relational database with query functions, access to remote sources, and access authorizations. Components include the Communications Manager and the Database Manager. Publications that provide information about these components are listed below.

Extended Services for OS/2: New Functions and Features (GG24-3794). Provides information about the new functions and features in the Extended Services and the Distributed Database Connection Services/2 products.

IBM OS/2 Vendor Integration Guide (S33F-9404). Provides information about selected equipment manufactured by non-IBM vendors that is compatible with IBM PC AT and Micro Channel workstations. Documents the differences in OS/2 installation procedures, keyboards, operation, and migration by vendor model.

Extended Services Command Reference (S04G-1020). Contains three

volumes describing the syntax and other information about the commands:

- *IBM Extended Services for OS/2 Communications Manager Command Reference*
- *IBM Extended Services for OS/2 Database Manager Command Reference*
- *IBM Extended Services for OS/2 Query Manager Command Reference*

Communications Manager

IBM Extended Services for OS/2 ACDI Programming Reference (S04G-1023). Introduces and defines the procedures for using the Asynchronous Communications Device Interface (ACDI). Serves as a reference for programmers who want to use the asynchronous programming functions to create application programs using C and Macro Assembler languages.

IBM Extended Services for OS/2 APPC Programming Reference (S04G-1025). Provides information for creating application programs in C, COBOL, or Macro Assembler that use Advanced Program-to-Program Communications (APPC) programming interface functions.

IBM Extended Services for OS/2 Conventional LU Application Programming Reference (S04G-1028). Helps the communications system administrator configure for conventional Logical Unit (LU) application sessions.

IBM Extended Services for OS/2 Server-Requester Programming Reference (S04G-1026). Provides information needed to create requester application programs that use Server-Requester Programming Interface (SRPI) functions in Communications Manager.

IBM Extended Services for OS/2 X.25 Programming Reference (S04G-1030). Provides information needed to create application programs for the X.25 API.

IBM Extended Services for OS/2 EHLLAPI Programming Reference (S04G-1027). Defines the Emulator High-Level Language Application Programming Interface (EHLLAPI). Applications written to this programming interface can access and control the 3270 and 5250 host presentation spaces.

IBM Extended Services for OS/2 Real-Time Interface Coprocessor Multiport/2 Device Driver Programming Reference (S04G-1029). Provides information for writing programs that support the Real-Time Interface Coprocessor Multiport/2 adapter. Discusses the dynamic link routines, the application loader, and the device driver interface.

IBM Extended Services for OS/2 ACDI Redirection Guide and Information (S04G-1024). Describes the Asynchronous Communications Device Interface (ACDI) Redirection services available to Extended Services LAN workstations that are using ACDI and the IBM LAN Asynchronous Connection Server.

IBM Extended Services for OS/2 Service Point Application Router and Remote Operations Service Guide (S04G-1021). Provides installation, configuration, and maintenance information needed to write application programs for Remote Operations Services.

Database Manager and Query Manager

IBM Extended Services for OS/2 Database Manager Programming Guide and Reference (S04G-1022).

Provides information needed for designing and coding application programs that access Database Manager. Also includes information needed to write programs for the Query Manager callable interface in C, COBOL, and FORTRAN programming languages and in REXX.

LAN Server 2.0

LAN Server 2.0 includes OS/2 LAN Server, OS/2 LAN Requester, DOS LAN Requester, LAN Adapter and Protocol Support, LAN Support Program, and various utilities. LAN Server provides resource sharing for files, printers, and serial devices across a LAN. LAN Server also provides comprehensive LAN capabilities to interconnected OS/2 and DOS workstations on IBM Token Ring, IBM PC Network, and ETHERAND networks.

IBM Operating System/2 Local Area Network Server Version 2.0 Application Programmer's Reference (S04G-1046). Describes the API provided by LAN Server 2.0. Assists application and system programmers in developing software for use in the OS/2 LAN Server environment. Includes an installable online version.

IBM Operating System/2 Local Area Network Server Version 2.0 Command Reference (S04G-1045). Documents all LAN Server, DOS LAN Requester, and User Profile Management (UPM) commands.

Books

Albrecht, Robert and Plura, Michael. *Stepping Up to OS/2 2.0*. Grand Rapids, MI: Abacus, 1992. ISBN: 1-55755-160-X.

Camarda, Bill. *OS/2 in the Fast Lane*. Carmel, IN: New Riders, 1992. ISBN: 0-56205-126-1.

Chapman, Robert B. *OS/2 Presentation Manager Programming for COBOL Programmers*. Wellesley, MA: QED, 1992. ISBN: 0-89435-439-6.

Conklin, Dick, ed. *OS/2 Notebook*. Redmond, WA: Microsoft Press, 1990. ISBN: 1-55615-316-3.

Dill, David M. *The COBOL Presentation Manager Programming Guide*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01293-4.

Franken, Peter. *OS/2 2.0 Complete*. Grand Rapids, MI: Abacus, 1992. ISBN: 1-55755-157-X.

Gopaul, Mitra V. *C Programming in the OS/2 2.0 Environment*. New York: Van Nostrand Reinhold.

Haber, John and Haber, Herbert R. *OS/2 2.0: The Usable, Portable Guide*. Winchester, MA: Usable Portable Publications, Inc., 1992. ISBN: 0-945965-27-4.

Ivens, Kathy and Proffit, Brian. *OS/2 Inside and Out*. Berkeley, CA: Osborne McGraw-Hill, 1993. ISBN: 0-07-881871-0.

Knight, Stephen A. *Learning to Program OS/2 2.0 Presentation Manager by Example: Putting the Pieces Together*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01292-6.

Kogan, Mike and Deitel, H.M. *The Design of OS/2*. Reading, MA: Addison-Wesley, 1992. ISBN: 0-201-54889-5.

Levenson, Steven. *Now That I Have OS/2 2.0 on My Computer, What Do I Do Next?* New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01227-6.

Mastrianni, Steve. *Writing OS/2 Device Drivers in C*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01141-5.

Mentor Technologies Staff. *Mentor Notes—Software Learning Made Easy: Introduction to OS/2 2.0*. Columbus, OH: Mentor Technologies, 1992. ISBN: 1-56494-005-5.

Minasi, Mark; Little, John W.; Semple, Marlene; Camarda, Bill. *Inside OS/2 2.0*. Carmel, IN: New Riders Publishing, 1992. ISBN: 1-56205-045-1.

Moskowitz, David, et al. *Converting Applications to OS/2*. New York: Brady Press. ISBN: 0-13-171943-2.

Nance, Barry and Chicres, Greg. *Using OS/2*. Carmel, IN: Que Books, 1992. ISBN: 0-88022-863-6.

Nance, Barry. *OS/2 2.0 Quick Reference*. Carmel, IN: Que Books, 1992. ISBN: 1-56529-068-2.

Orfali, Bob and Harkey, Dan. *Client/Server Programming with OS/2 2.0*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01219-5.

Proffit, Brian. *OS/2 Application Development Tools*. Stratford, CT: Premier Publishing, 1992. ISBN: 1-881899-00-4.

Shafe, Laurence Dr. *OS/2 in the Corporate Environment: A Management Guide to Client/Server Computing*. Intelligent Environments, Inc., 1992. (508) 640-1080.

Shafer, Dan. *Smalltalk Programming for Windows and OS/2*. Rocklin, CA: Prima Publications, 1992. ISBN: 1-55958-280-4.

Tate, Bruce; Malkemus, Tim; Gray, Terry. *Comprehensive Database Per-*

formance for OS/2 2.0's Extended Services. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01325-6.

Tyne, Maria. *OS/2—The Workplace Shell: A User's Guide and Tutorial for Release 2.0*. Wood Dale, IL: Computer Information Associates, 1992. ISBN: 1-881435-47-4.

Tyson, Herb. *Ten-Minute Guide to OS/2 2.0*. Carmel, IN: Alpha Books, 1992. ISBN: 1-56761-000-5. 225 pages.

Winn, Graham. *OS/2 Presentation Manager*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-00468-0.

Zach, William. *Integrating Applications with OS/2 2.0*. New York: Van Nostrand Reinhold, 1992. ISBN: 0-442-01234-9.

Videos

Comsell, Inc. *Using OS/2 Version 2*. Combines motion video, sound, and animated graphics to provide multimedia training on your PC on the concepts, features, and functions of OS/2. Most users will complete this course in four to six hours. A comprehensive workbook accompanies the course. Available for \$2,195 from Comsell, Inc., One Buckhead Plaza, Suite 1500, 3060 Peachtree Road, NW, Atlanta, GA 30305. Phone: (404) 262-3500.

Comsell, Inc. *Working with OS/2 Version 2* (90 minutes). Designed for beginning users of OS/2. Provides a basic understanding of how OS/2 2.0 works and how to manage it. Available for \$99 from Comsell, Inc., One Buckhead Plaza, Suite 1500, 3060 Peachtree Road, NW, Atlanta, GA 30305. Phone: (404) 262-3500.

BBS Contact	Name	Phone	Location	Baud Rate	Modem Type	FidoNet Node
Patrick O'Riva	AsmLang and OS/2*	(408) 259-2223	CA	14.4	HST, MNP V.32 (and/or V.42)	Yes
Steve Lesner	Bullet BBS*	(203) 329-2972	CT	9600	HST, MNP V.32 (and/or V.42)	Yes
		(203) 322-4135	CT	9600	HST, MNP V.32 (and/or V.42)	Yes
Jim Dailey	Cajon Zone OS/2*	(619) 588-6634	CA	9600	MNP V.32/V.42bis	Yes
Bob Germer	Capital City BBS*	(609) 261-0689	NJ	14.4	HST, MNP V.32bis/V.42bis	Yes
Emmitt Dove	Fernwood*	(203) 483-0348	CT	9600	HST, MNP V.32bis/V.42bis	Yes
		(203) 481-7934	CT	14.4	HST, MNP V.32 (and/or V.42)	Yes
Bill Cook	Greater Chicago Online*	(708) 895-4042	IL	9600		Yes
Bogie Bugsalewicz	I CAN! BBS	(312) 736-7434	IL	9600	HST, MNP V.32bis/V.42bis	Yes
		(312) 736-7388	IL	2400		Yes
N/A	IBM National Support	(404) 835-6600	GA	2400		
		(404) 835-5300	GA	9600	HST, MNP V.32bis/V.42bis	
Robert McA	Live-Wire	(214) 307-8119	TX	9600	HST, MNP V.32 (and/or V.42)	Yes
Dave Fisher	LiveNet*	(918) 481-5715	OK	16.8	HST, MNP V.32bis/V.42bis	Yes
Chuck Gilmore	Magnum BBS*	(818) 706-9805	CA	9600	HST, MNP V.32bis/V.42bis	
Joe Salemi	Max's Doghouse*	(703) 548-7849	VA	2400	HST, MNP	Yes
Paul Breedlove	Multi-Net*	(503) 883-8197	OR	9600	HST, MNP V.32bis/V.42bis	
Ron Bemis	Nibbles & Bytes	(214) 231-3841	TX	9600	HST, MNP	Yes
Craig Swanson	OS/2 Connection*	(619) 558-9475	CA	14.4	MNP V.32/V.42bis	Yes
Pete Norloff	OS/2 Shareware*	(703) 385-4325	VA	9600	HST, MNP V.32bis/V.42bis	Yes
		(703) 385-0931	VA	9600	HST, MNP V.32bis/V.42bis	Yes
Brady Flowers	Oberon Software*	(507) 388-1154	MN	14.4	HST, MNP V.32bis/V.42bis	
Paul Beverly	PMSC Online Resource*	(803) 735-6101	SC	2400		Yes
Ken Rucker	Ruck's Place/2*	(817) 485-8042	TX	14.4	HST, MNP V.32bis/V.42bis	Yes
Ed Barboni	System-2 RBBS	(215) 631-0685	PA	9600	MNP V.32/V.42bis	Yes
		(215) 584-1413	PA	9600	MNP V.32/V.42bis	Yes
Mark Lehrer	The Akron Anomoly	(216) 688-6383	OH	9600	HST, MNP V.32bis/V.42bis	Yes
Bill Schnell	The Asylum BBS*	(918) 832-1462	OK	9600	HST, MNP V.32 (and/or V.42)	Yes
Bill Hatton	The Monster BBS	(908) 382-5671	NJ	9600	HST, MNP	
Woody Sturges	The OS/2 Woodmeister*	(314) 446-0016	MO	14.4	HST, MNP V.32bis/V.42bis	Yes
Troy Kraser	The Other World	(904) 893-2404	FL	9600	MNP V.32bis/V.42bis	Yes
Art Fellner	The Soldier's Board*	(713) 437-2859	TX	9600	HST, MNP V.32bis/V.42bis	Yes
Bill Andrus	The Systems Exchange*	(703) 323)7654	VA	9600	HST, MNP	Yes
N/A	WSI BBS	(901) 386-4712	TN	2400		

* OS/2 is the primary interest of the board.

Figure 2. Independent OS/2 BBSs

IBM. *CUA Vision* for VHS PAL (GV26-1004) or for VHS NTSC (GV26-1003). This video and booklet (G242-0215) outline IBM's vision for the way user interfaces will develop. The booklet includes a diskette with a graphical demonstration of the principles discussed.

IBM. *Moving to the Workplace Shell* (S41G-5097). Guides the new user of OS/2 2.0 through the Workplace Shell. Topics include using folders and objects, migrating, customizing, associating, and creating shadows and templates.

IBM. *Screen Reader for OS/2: Highlights* (GV21-9052). Provides an overview of Screen Reader for OS/2.

ViaGrafix. *Getting Started with OS/2 2.0* (54 minutes). Available for \$39.95 from ViaGrafix, 31 S. Adair, Pryor, OK 74361. Phone: (918) 825-6700.

ViaGrafix. *Productivity with OS/2 2.0* (63 minutes). Available for \$49.95 from ViaGrafix, 31 S. Adair, Pryor, OK 74361. Phone: (918) 825-6700.

Other Documents

IBM *OS/2 Applications Solutions Directory* (G362-0002). This guide lists most of the applications available for OS/2 today. There is a supplement of 32-bit applications (G362-0029).

I.V. League Catalog. This catalog provides a listing of OS/2 materials including books, videos, courseware, and other items available from independent vendors. The catalog is available free by calling (800) 342-6672 or writing I.V. League, P.O. Box 525, Dearborn, MI 48121-0525.

OS/2 Frequently Asked Questions. Maintained by Timothy F. Sipples (sip1@ellis.uchicago.edu). An excellent basic reference to many common questions and answers about OS/2. It is available in electronic form on most of the bulletin board systems (see "Bulletin Boards" below).

Bulletin Boards

IBM-Sponsored BBSs

IBM sponsors the National Support Center Bulletin Board Service (BBS), located in Atlanta, Georgia. This BBS provides conferencing capabilities for customers and the ability for customers to submit problems for resolution. The BBS can be reached using the following numbers:

(404) 835-6600	First available modem
(404) 835-6296	First available Hayes Ultra
(404) 835-5300	First available USR V.32bis with ASL
(404) 835-5578	First available IBM 7855 Model 10

The Canadian Software Support Center within IBM Canada supports BBSs specializing in OS/2 and OS/2 support. Those numbers are as follows:

(604) 664-6464	Vancouver
(416) 946-4244	Toronto
(416) 946-4255	Toronto
(514) 938-3122	Montreal

CompuServe

IBM provides technical support and general OS/2 information on three CompuServe forums. New DAP enrollees can enter GO OS2DAP to register. OS2DF1 and OS2DF2 are for OS/2 developer support.

The CompuServe Information Service offers the IBMOS2 conference for general and technical discussions with other OS/2 software developers and computer professionals at IBM and other companies. The OS2DEV conference is for software development-oriented technical discussions, including technical support from IBM. Both conferences include shareware programs, tools, technical tips, and sample code.

Independent BBSs

Independent BBSs throughout the U.S. are sponsored by various organizations and individuals. The primary interest of many of these BBSs is OS/2. Figure 2 provides a current list of those available.

IBM FAX Information Service

IBM FAX can be used to obtain up-to-date product information (specification sheets, brochures, and so on) through a computerized fax machine and a voice response unit.

Information is available on products ranging from personal systems to large systems, education, application software, IBM services, and general information.

Call (800) IBM-4FAX or (800) 426-4329 to select a document containing the product information of your choice and have it automatically sent to your fax machine. IBM FAX is available 24 hours a day, seven days a week. There is no charge for this service.

Acknowledgement

This article is based on material provided by Martin McElroy at IBM Corporation in Basingstoke, United Kingdom.

Little Solutions



We invite you to share your "little solutions" in this column. Send them to us in care of the editor.

LAD/2: A Simple Way to Implement CID

Anyone who is familiar with IBM's LAN strategy for Configuration, Installation, and Distribution (CID) knows that the strategy is rich and complex. Someone who is new to CID must learn the intricacies of redirected response file capability to perform LAN-based installations. Recognizing a need to simplify remote installations and insulate users from the complexity of CID, IBM has developed LAN Automated Distribution/2 (LAD/2).

LAD/2 ties together and automates the rich functions of CID through an easy-to-use Presentation Manager (PM) interface. The user interface captures configuration data for LAN clients. LAD/2 then uses this data to automatically build the response files

and LCU command file for installation. The files generated by LAD/2 require no additional manipulation by a user.

LAD/2 supports several products and configuration features. These products and their associated features are summarized below.

Configuration, Installation, and Migration for the following products:

- **OS/2 2.0:** All selective configuration features
- **Network Transport Services/2 (NTS/2) LAPS**
 - 802.2
 - NetBIOS
 - NetWare
 - IBM Token-Ring Network Adapters

- IBM PS/2 Adapter for Ethernet Networks
- 3Com 3C523 Etherlink/MC Adapters
- Ungermann-Bass NIUps Adapters

- **OS/2 Communications Manager**

- Up to four 3270 DFT sessions
- Up to four 3270 Non-DFT sessions
- One 3270 DFT host printer session
- One 3270 Non-DFT host printer session
- Advanced Peer-to-Peer Networking (APPN)

- **OS/2 Database Manager**

Following are the database configuration types:

- Stand-alone
- Client
- Server
- Server/Client

- **LAN Server 3.0 (Advanced)**

- LAN Requesters
- Domain Controllers and Backup Domain Controllers
- Additional servers
- First Failure Support Technology/2 (FFST/2) support

- **CID-Enabled Applications**

Configuration and Distribution for the following:

- **NetWare for OS/2**
 - Network Interface Card

- Source Routing
- Sequential Packet Exchange (SPX) support for OS/2 sessions
- NetBIOS emulation for OS/2 sessions
- NetWare support for DOS and Windows applications
- Remote Named Pipes support
- **OS/2 Applications**
 - Modification supported for the Desktop, CONFIG.SYS, and STARTUP.COM to uniquely configure each application
- **DOS LAN Requester**
- **Personal Communications/3270 (PC/3270):**
 - Up to five 3270 terminal sessions
- **DOS Database Requester**

- **DOS Applications**

- Modification supported for AUTOEXEC.BAT and CONFIG.SYS

Along with these features, LAD/2 also supports partitioning/formatting client hard drives, automatic creation of client boot diskettes, and virus scanning to protect the integrity of installations.

- **Who Needs LAD/2?**

LAD/2 will benefit anyone who must perform remote installations or distributions for multiple workstations on a LAN. LAD/2 enables you to take advantage of IBM's CID strategy without having to know all the technical aspects of CID. Contact your IBM representative or call (800) 547-1283 for more information about LAD/2.

— Douglas Kuelbs, IBM Corporation, Roanoke, Texas

Shutdown without a Mouse

Some OS/2 2.0 users do not use a mouse. This can make it difficult to perform a proper shutdown, especially if you do not know the key sequence. It only takes the following six steps to perform a proper shutdown without using a mouse:

1. Press the Ctrl + Esc keys to bring up the Task List.
2. Use the arrow keys to select the OS/2 2.0 Desktop Icon View.
3. Press the Enter key.
4. Press the Ctrl + \ keys to deselect all items and give the focus to the Desktop.
5. Press the right Shift + F10 keys.
6. Select the option to shut down.

— Tiffany Groom, IBM Corporation, Roanoke, Texas

Programming for Display Resolutions

Unwary programmers may encounter pitfalls by not planning for various display resolutions available today. An ever-increasing number of display adapters and displays offer many different resolutions, scan rates, and colors. Application programmers must support these choices. Spending time and effort in the design stage to understand how different resolutions and display sizes may affect the functionality of the application can help avoid surprises that may render the application unusable later.

This article highlights the importance of considering different display resolutions when developing OS/2 2.0 Presentation Manager (PM) applications. It also considers the meaning of the phrase "device independence" and the confusion surrounding it. Fonts contribute their share of trouble, so suggestions for avoiding some of those problems are provided. The article also addresses the use of C and other high-level language programming and some 4GL products to see if they help or hurt.

Device Independence

When asking several colleagues their definition of "device independence" when applied to displays, I received

almost as many different ideas. One who works very closely with OS/2 said, "What do you mean by device independent programming?" Another thought it should be called "intelligent programming." A third said, "Device independence is writing an application so that it behaves *in the way intended*, regardless of the type of display (and adapter) used by the end user." I agree with that statement and will use it as the definition for the remainder of this discussion.

The italicized phrase in the above definition, *in the way intended*, means that the programmer has considered every facet of how the application will appear on the display. It

does not mean that the application will be tested on each combination of system, display adapter, and display at each resolution available. It means that the programmer has considered the effect of different resolutions (even those not yet available) and has written the application with that in mind.

This discussion concerns only the user interface to the application via Presentation Manager windows, controls, and text, and the way those items are displayed. The application's logic and processing are not part of this topic.

Choices

There are three primary options for programming a Presentation Manager application's visual interface:

- **Constant Pels¹:** A window (or another object) can be created by specifying its size and position by pels. This is an absolute reference to the size and position of the item. The object will always display at the same location, in pels, relative to the lower left corner of the monitor. It will always be the same size, in pels, as originally defined. For example, a window defined as 200 x 300 pels, located 150 pels from the bottom, and 150 pels from the left side of the screen will always be displayed that way—regardless of the resolution of the screen. Figure 1 shows an example.
- **Constant Percentage:** The next choice is to always maintain a constant percentage. This is what we will use to create an object that will be one-fourth the width of the screen and one-third the height; we always want to maintain those proportions. This can be done easily by finding the width and height in pels using `WinQuerySysValues`, and then multiplying by the desired percentage.

- **Constant Size:** The third choice is to create an object whose size is measured by some system that is not native to the computer. This means using inches, centimeters, or any other system that is primarily used for measuring physical objects. For example, if you create a five-inch line on the monitor screen and place a ruler next to the line, it will measure five inches.

To find the height and width of the display in inches, use the `DevQueryCaps` Application Programming Interface (API) and obtain pels-per-meter for both dimensions. With a little math, we know the physical measurements of the display area. This is more difficult to implement than it first appears, because some 17- or 19-inch displays look like 8514 displays to the system when connected to an XGA adapter. Since these displays are larger than an 8514, the value returned in pels-per-meter is incorrect. The solution to this problem is called Display Mode Query and Set (DMQS). It is available on some XGA systems now

and will become more widespread in the future. OS/2 2.1 supports DMQS as part of the operating system.

There are several caveats relating to these three choices. Suppose we create a visual interface for data entry on a VGA system, and decide to use the full display area. By using the constant percentage method (100% in this case), this application will always use the whole screen, even on an XGA system where the resolution is greater.

Some may see this as a terrible waste of space. It was not as important when using an 80-column by 24-line display, but with graphical systems such as the Presentation Manager, more resolution is the equivalent of a larger display area. In this case, the extra resolution will not be used. However, if we created a smaller window using the constant pels method similar to the example in Figure 1, the result may be a window that is uncomfortably small for some users

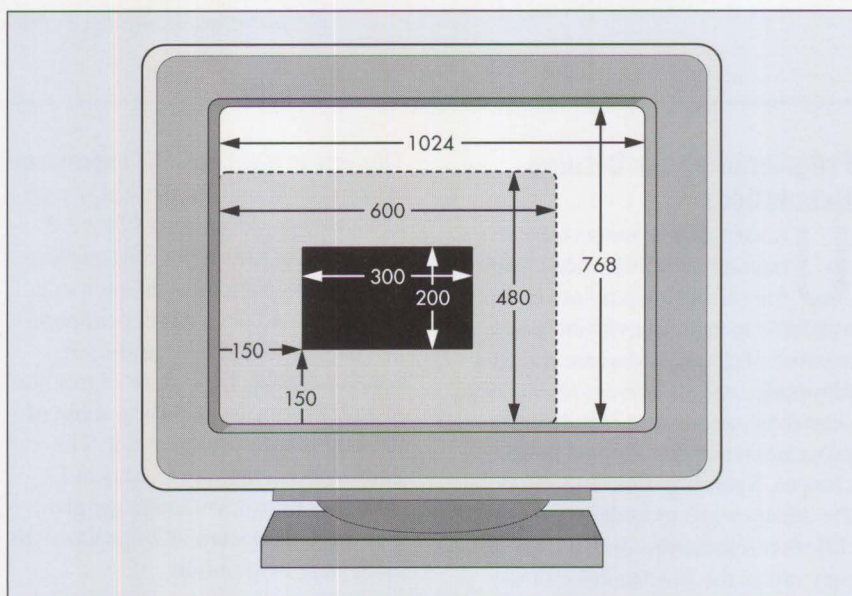


Figure 1. Constant Pels Method

¹ A pel is the smallest display unit of a display screen. The resolution of a display is measured in pels on both the horizontal and vertical axes.

when the resolution is changed from VGA to XGA.

Each of these three choices may individually have some disadvantages. If none of the options above seem appropriate, make adjustments to the method already being used. This is a fourth choice: *intelligent adjustment*. It means that we can use any of the methods already discussed and change to another when it makes sense. This may even mean changing fonts, especially if we are using controls in the client area, not in dialog boxes.

We have not yet considered the effect of a different system font or using dialog units in dialog boxes. Now let's consider what can be done to anticipate the positive aspects of our options, allowing us to create applications that work well in any circumstance.

Fonts and Dialog Boxes

The *Programming Guide* suggests that a dialog box is for an application to have a dialog with the user. I consider this as a short conversation, not for "heads down" data entry, or worse yet, as the basis for the main part of the application. Dialog boxes provide convenience, but at a price. They manage the cursor when tabbing from one control to another, allow groups of similar objects to be defined, and provide the ability to define their layout by using .RC files.

One of the problems with dialog boxes was actually designed to be its strength. Dialog boxes use a measurement system based on the size of the average character of the default font. The unit itself, a dialog unit, is one-fourth the width and one-eighth the height of the default standard character. By using dialog units when designing a dialog box, we were assured that the application would behave in a predictable way on systems with different resolutions and

monitor sizes. Although this was supposed to be device independent, it usually is not. When OS/2 is installed for a different resolution or it encounters a different monitor size, different system fonts are used. When the system font is changed, the dialog unit size changes, and the size of the dialog boxes and controls also change.

Intelligent adjustment means that we can use any of the methods and change to another when it makes sense.

Therefore, some dialog boxes no longer fit in the client window, and sometimes text does not display properly. There are several reasons for this:

- Crowding too many controls into a limited area and not allowing room for the windows and its controls to expand in size.
- When a control changes size, fonts other than the system font remain the same.
- Not explicitly testing on other resolutions to see the actual result.

The solution is limiting the use of dialog boxes to their intended purpose of short conversations with the application, and not crowding the objects within them.

Why Do the Fonts Change?

When a smaller display is used (8515 versus 8514) or the resolution of the system increases, the system font becomes larger. This prevents the system default from becoming so small that it causes discomfort to the user. International Standards Organization (ISO) compliance, introduced with

the ServicePak, brought yet another change in system fonts.

These changes protect the user. After all, who wants to leave at the end of the day with a bad case of "user-squint"? However, programmers can control the situation by choosing not to use the default font for controls (or perhaps for anything else), but by dynamically selecting a more suitable font for the purpose. Programmers need to assume the responsibility for ensuring that their application is still comfortable to use.

Controls in the Client Area

We can create controls in any window. Although the dialog window offers the programmer some relief from controlling tabbing, we occasionally want to place controls in a client area. All controls have a close relationship with text. What would an entry field or a list box be without text? Or a drop-down list, combo-box, or spin button? Try using a set of pushbuttons without labels.

By not using a dialog window, we lose the convenience of using dialog units. Our control windows are measured in pels and conform to the methods already discussed. When the default system font changes because a different size monitor is used, or the display resolution changes, any control that uses the system proportional font (the default system font) runs the risk of not fitting. Dialog boxes automatically adjust for this, as we have discovered, but now the control is no longer large enough to hold its text.

What Can We Do?

There are several ways to implement "intelligent adjustment." An example is the behavior of Microsoft's Excel for OS/2. When Excel is run on a VGA system, columns A through F or G are displayed. When Excel is used on an 8514/A system with an

8514 display, columns A through L are displayed. However, on an XGA adapter with an 8515 display, column J is the last displayed. Both the 8514/A and the XGA adapters have a horizontal resolution of 1024 pels. The difference appears to be that the size of the display was considered, and an adjustment was made when the 8515 was detected. Clearly, the application's behavior was predetermined and did not happen accidentally.

When designing an application's main functions, use the GPI interface of Presentation Manager if you are coding in C or another high-level language for maximum flexibility. Retain as much control as possible. Use dialog boxes for preferences, settings, font selection, and file management. It is best to limit the use of dialog boxes to short, quick interactions with the application.

Consider the *real* size of the application's visual elements. When there is potential for an object to become too small to be useful, make it larger. If Excel displayed the same number of columns on the 8515 display as on the 8514, each column would have been smaller. Someone decided that the usability of Excel may be enhanced if there were a slight increase in width, and made an adjustment. The result is a compromise between the VGA column count and the narrower columns that would have resulted if nothing had been done.

Language Options

To some, there is a *perceived* steep learning curve associated with Presentation Manager programming using C. But using C is where you find flexibility and control. There is more work at the beginning, but you can control all aspects of the application, including the appearance of the interface at runtime. This is done by asking the system for the resolution

and size of the display that will be used and, based on the information returned, making the adjustments that maximize the application's usability.

Using code generators or 4GLs introduces some restrictions and makes it difficult to use any solutions that are implemented with a high-level language. Often, an application created on a VGA system using a 4GL cannot be used on an XGA system without some changes. Each system would require a different version of the application. Work must be done on each system to ensure that the result is what was intended.

When designing an application's main functions, use the GPI interface of PM if you are coding in C or another high-level language.

To demonstrate one problem with 4GLs, use a system that allows a window to be drawn interactively, such as EASEL® or CASE:PM™, and one that generates source code that can be examined. Somewhere in the generated code, you can see where it has created the control and set its size and position. If a 4GL application is created on a VGA system, there will be a set of values similar to those in Figure 1. If an XGA system is used, those values will be much different.

Most developers do not go into this code and add the necessary logic to make the size and position dynamic, based on monitor size and resolution. Although it may be possible to do, it may not be practical. Other 4GLs keep the code internal so it cannot

even be viewed. The 4GLs also limit or completely block access to other fonts for list boxes, pushbuttons, and drop-down combo boxes. Use any of the 4GLs with care and caution, and understand the limitations.

Conclusion

This has been a brief look at a most complex issue. It is very important to build an application with your eyes and mind open to the potential problems that may be encountered.

I know of one situation where the programming staff, using a 4GL, spent nearly two years developing an application only to find that it would not run acceptably on an XGA system.

Another application was based on templates created for Excel for OS/2 on an 8514 monitor. The application could not be used when XGA systems and 8515 displays were chosen for the roll-out.

The end of the project is an expensive time to discover that major changes in the application are required, or that the hardware that has been purchased is not compatible with the application.

By using the methods described in this article and looking at every aspect of the user interface, you can protect yourself from these unpleasant situations. Select the least obstructed path to the problem's solution, and realize that certain investments are worth the time and trouble in the long term. If using a 4GL, be certain that the limits are known and understood before one of those limits is reached. It is possible to combine high-level languages and 4GLs to realize the strengths of each, and keep the risks to a minimum.

—Larry Pollis, IBM Corporation,
Roanoke, Texas

New Products

Database2 OS/2 and SAA DDCS/2

IBM announced two new relational database products for the 32-bit OS/2 operating system environment:

- IBM Database2 OS/2 (DB2/2) Version 1
- IBM SAA Distributed Database Connection Services/2 (DDCS/2) Version 2

Both provide single-user and client/server versions.

IBM DB2/2 is a relational database management system and a member of the IBM relational database family of products along with DB2 for MVS, SQL/DS for VM and VSE, and SQL/400[®] for OS/400. DB2/2 extends IBM's SAA relational database technology to the single-user workstation and client/server LAN environment. It is a key component in IBM's Information Warehouse[™] framework.

DB2/2 supports access to OS/2 database servers from OS/2, DOS, and DOS/Windows database client workstations. DB2/2 is a 32-bit product that includes the functions previously provided in the OS/2 Extended Services 1.0 Database Manager. DB2/2 also includes additional new functions that are focused on application portability, DB2 compatibility, SQL and industry standards compliance, new connectivity options, integrity enhancements, reliability, availability, systems management, and performance.

DB2/2 and DDCS/2 provide relational database access solutions for line-of-business, decision support, and

online transaction processing applications that may reside on a LAN or in a host environment. Application Programming Interfaces (APIs) and tools are provided for programmers as well as a database command-line processor. Query Manager is also provided for use by both novice and experienced users.

DDCS/2 provides workstation application access to enterprise data. DDCS/2 enables users to develop and run application programs that connect to, access, and update host databases from OS/2, DOS, and DOS/Windows database client workstations. The same database clients installed to access DB2/2 database servers on a LAN also can access and update host databases through DDCS/2. To a user or application program, the host database can be considered as an extension of the OS/2 database server capabilities.

DDCS/2 is a 32-bit product, including the same functionality provided in DDCS/2 1.0. Connectivity to Database 2 (DB2) Version 2 Release 3, Structured Query Language/Data Services (SQL/DS) Version 3 Release 3, and Operating System/400 (OS/400) Version 2 Release 1.1 host relational databases is supported. DB2/2 and DDCS/2 can be acquired separately, but used with the new IBM Communications Manager/2. DB2/2 and DDCS/2 are designed to run with IBM OS/2 2.0 with ServicePak level XR06055, or later. DB2/2 and DDCS/2 also are designed to run on selected IBM and non-IBM hardware that support OS/2 2.0 with Service-

Pak level XR06055, or later. General availability will be announced at a later date.

Highlights:

- Exploitation of the 32-bit OS/2 2.0 operating system and its application development platform
- Improved DB2 compatibility and SAA SQL Level 2 language extensions
- Application programming language enhancements to aid in programmer productivity
- Scalability enhancements
- Additional connectivity options to support IBM's recently announced Network Transport/Services/2 (NTS/2) product and Communications Manager/2 products
- Systems management enhancements including expanded support of the First Failure Support Technology/2 (FFST/2) and IBM's LAN NetView systems management products
- Remote installation support of DB2/2 components and DDCS/2 using IBM's Configuration, Installation, and Distribution (CID) technology
- Inclusion of the functions previously found in the Extended Services 1.0 Database Manager
- Continued support of 16-bit OS/2 database applications
- Database client-enabling distributed feature support for OS/2, DOS, and DOS/Windows 3.0 or 3.1 workstations, provided with the client/server version with DB2/2
- Additional National Language Support for Latin-2, Turkey, Arab countries, and Israel

- A Query Manager distributed feature included as part of the client/server version of DB2/2, for OS/2 client-only workstations, to access remote database servers on a LAN using Query Manager
- Access to Extended Services 1.0 database servers from DB2/2 database clients
- Access Extended Edition 1.0 database servers from DB2/2 database clients
- Access to DB2/2 database servers from Extended Edition 1.3 and Extended Services 1.0 OS/2, DOS and/or DOS/Windows clients
- OS/2, DOS, and DOS/Windows database client access to DB2, SQL/DS, and OS/400 databases using DDCCS/2 and the Distributed Relational Database Architecture™
- Flexible price/performance packaging options
- Overall quality and performance improvements; DB2/2 and DDCCS/2 protect customer investments in current database management applications and provide a foundation for expansion of future database management requirements

Letter # 293-037, January 26, 1993

IBM PS/ValuePoint 6384 System

IBM is responding to the preferences of individual customers and customers working in large, medium, and small business environments. The five new PS/ValuePoint™ 6384 models have IBM DOS 5.0 and Microsoft Windows 3.1 preinstalled. Packaged in a versatile five slot/five bay mechanical package, these models provide customers with expansion choices and investment protection.

The PS/ValuePoint 6384 has SVGA 16-bit graphics. Models Fxx, Mxx, and Wxx support 640 x 480, 800 x 600, 1024 x 768 up to 256 colors.

MFI modes 2, 3, and 4 are supported. MFI mode 5 (132 columns) is partially supported in some graphics modes. *Note:* The monitor attached must support the same modes. The following coprocessor/processor upgrade support is provided:

- Models F23 and F43: 487SX 25 MHz or 486 25/50 MHz overdrive upgrade
- Models M43 and M53: 486 33/66 MHz overdrive upgrade

The PS/ValuePoint 6384 has four full-length and one half-length AT-bus 8/16-bit adapter card slots for expandability. When additional expansion and hard disk storage are required, these new 6384 models can be upgraded with an additional 5.25-inch, 1.2 MB half-high diskette drive and one 80, 170, or 212 MB hard disk. PS/ValuePoint 6384 Model F20 enhancement is effective immediately: the 6384 Model F20 will be shipped with a 120 MB hard disk (previously announced October 20, 1992, with an 80 MB hard disk). The price remains the same.

Letter # 193-010, January 25, 1993

IBM Entry LAN-to-LAN Wide Area Network Program

The IBM Entry LAN-to-LAN Wide Area Network Program Version 1.0 is a program product that interconnects remote LANs across a Wide Area Network (WAN). Applications written to the IEEE 802.2 interface, such as Lotus Notes and others that use the IBM NetBIOS frame protocol, will be able to communicate with remote LANs using the wide area network as a transport medium. This program provides the interface between the LAN and the WAN and communication will be via the LU 6.2 facilities provided by OS/2 Communications Manager.

Highlights:

- Improves workgroup computing by enabling customers with small remote offices to interconnect their OS/2 LAN Server with NetBIOS applications, such as Lotus Notes
- Provides filtered access to resources on the LAN
- Extends LAN connectivity by its use of data link switching
- Offers accounting functions of bandwidth usage for charge-out purposes
- Enables NetBIOS connectivity to a WAN from a single workstation

Letter # 293-005, January 26, 1993

IBM PS/2 Server 295 Preinstalled Software

Additional preinstalled software options can now be selected for the build-to-order PS/2 Server 295 system. Customers have the choice of ordering OS/2 LAN Server 2.0 Advanced or OS/2 LAN Server 3.0 Advanced preinstalled on the PS/2 Server 295 system at the manufacturing site. This improves system setup time and provides customers with a turn-key hardware/software solution.

Letter # 192-297, December 15, 1992

LAN Resource Extension and Services/VM Release 2.0

LAN Resource Extension and Services/VM (LANRES/VM) establishes a server environment on VM to allow NetWare clients transparent access to mainframe resources. With LANRES/VM, S/390™ or System/370™ VM systems, NetWare LAN servers, and heterogeneous clients, can be integrated in a way that helps maximize user productivity and provides unique business solutions. Customers who use NetWare can save time and leverage skilled

resources while they provide their NetWare clients with improved services.

With this release, LANRES/VM is enhanced to support Systems Network Architecture (SNA), Transmission Control Program/Internet Protocol (TCP/IP), and VM Programmable Workstation Communication Services (VM PWSCS) connectivity options. Indirectly attached NetWare LAN servers (NetWare LAN servers attached to a NetWare LAN server that is registered to use LANRES/VM) are supported for all LANRES/VM services and connectivity options. Programmable workstation client support is expanded to include all NetWare clients for all LANRES/VM services. Disk serving is enhanced to allow multiple NetWare servers to share read-only disk images on VM. In addition, enhancements to improve LANRES/VM usability have been incorporated.

Highlights:

- Provides business solutions through enhanced ability to leverage S/390 or S/370 resources into the workstation environment
- Increases productivity by providing transparent access to VM DASD and services to end users on a greater variety of programmable workstation platforms
- Enables growth by supporting read-only disk sharing among multiple NetWare servers and by supporting new user exits for printing on LAN printers
- Enhances systems management through simplified license control via an interface to the IBM Software License Monitor/MVS and VM and through improved tracing and log file support
- Protects the customer's investment in a wider range of environments

via new connectivity options to support SNA, TCP/IP, and VM PWSCS, and by supporting indirectly attached NetWare LAN servers for all LANRES/VM services and connectivity options

Letter # 292-693, December 8, 1992

New Release of Distributed Application Environment OS/2 Family Products

New releases of the primary OS/2-based Distributed Application Environment products are being provided to further enhance application development productivity, and improve usability and operational performance. In addition to functional enhancements, such as Presentation Manager-based operation utilities and support for both 16- and 32-bit applications, support has been enhanced for implementing and operating a seamless, distributed network environment that also includes AIX/6000-, DOS-, OS/400-, and VM-based applications.

IBM Distributed Application Environment provides common APIs, services, and tools to develop, install, and maintain applications that operate in a distributed, client/server network. With these standard application program interfaces and services, application development and maintenance are accomplished with reduced time and resources.

In addition to the OS/2 operating platform, the product family provides these benefits to the VM/SP, VM/ESA™, AIX/6000, OS/400, and DOS platforms, allowing these diverse architectures to be included in application investments. Applications developed using these APIs and installed on any one of these platforms may then be used on the others without concern for the hardware or software on which the application runs.

Highlights:

- Provides consistent application program interface and system services to support user productivity
- Supports growth through portability of application programs
- Allows the user to integrate resources and data, building an enterprise's business solution
- Provides the means for an enterprise to keep track of and manage its integrated systems
- Protects the user's investment through key architectures and products, portability, and National Language Support

Letter # 292-690, December 1, 1992

IBM Workstation Interactive Test Tool Version 2

Workstation Interactive Test Tool (WITT) Version 2 is the follow-on product to WITT Version 1 record and playback tool. This new version includes support for record and playback of OS/2 2.0 Presentation Manager applications along with the new OS/2 2.0 controls (slider, value set, container, and notebook). WITT Version 2 will continue to support record and playback of MVS, VM, VSE (VSE/ESA™ and CICS/VSE™), and OS/400 applications.

This new version of WITT also includes an online tutorial that helps users become productive more quickly. IBM is also making available displayable softcopy publications for select programs.

Highlights:

- Regression test becomes more reliable and consistent with WITT's record and playback ability, thereby improving application developers' productivity and the quality of applications.

- New code test becomes easier with WITT's edit, compare, and save functions, thereby improving application developers' productivity.
- WITT Version 2 supports OS/2 2.0 windowing.

Letter # 293-091, February 23, 1993

OASAS Disk Array Management Software 2.0 For OS/2 2.0

OASAS I Disk Array Management Software Version 2.0 for OS/2 2.0 manages the disk array subsystem and provides optional fault tolerance, disk spanning, and minimized downtime. OASAS I 2.0 software supports from three to eight Small Computer Systems Interface (SCSI) drives on one to four SCSI adapters as a single array. It can support up to four SCSI adapters and four arrays per system, with a maximum of 27 SCSI drives. When mirroring fault tolerance is implemented, each array has an even number of drives from two to sixteen with data capacity of two to eight drives. A non-fault tolerant array can have from two to eight drives.

OASAS I 2.0 provides the following choices for fault tolerance:

- Fault tolerant data striping—RAID/5
- Mirroring and duplexing—RAID/1

OASAS I Version 2.0 provides a solution to the need for low-cost, fault-tolerant storage on a PS/2 system for mission-critical applications. OASAS I 2.0 provides the following:

- Data striping across drives in the array with or without fault tolerance

- Mirroring fault tolerance (With the proper hardware, OASAS I 2.0 allows replacement and recreation of a single drive without shutting down the system when individual drive enclosures are used.)

Highlights:

- Provides multiple disk array fault-tolerant solutions to mission-critical business applications
- Supports any standard IBM SCSI adapter and any standard IBM SCSI drive
- Provides investment protection for PS/2 systems by adding additional capabilities to the SCSI disk storage subsystem
- Increases user productivity by providing better disk storage performance, size and reliability, while decreasing downtime
- Allows for greatly improved disk systems management on PS/2 systems by providing many utilities for disk array management

Letter # 293-092, February 23, 1993

IBM Time and Place/2 Version 1.0

IBM Time and Place/2 is an SAA, OS/2 2.0 LAN-based time management program for coordination and scheduling of work-day activities. Time and Place/2 gives the user control over events, "to-do's" and memos, and allows the user to schedule meetings when operating in a LAN environment.

Time and Place/2 provides a generalized time manager that can be used to manage any corporate resource, from

people to equipment. The user interface is patterned after common day/week/month desktop calendars. These different views can be printed in a variety of text or graphical forms. The functions provided by Time and Place/2 include:

Group functions:

- Scheduling of events for one or more members of a group
- Free time search across a group
- Time zone support across different time zones
- Assigning "to-do" items to one or more group members
- Replicating events across time periods or multiple days
- Security authorization to control access to a user's calendar events and "to-do's"

Personal functions:

- Ability to customize daily, weekly, monthly, and list views
- Events, prioritized "to-do" list, and memos for each day
- Pop-up monthly calendar for navigation or reference
- Alarms with tunes (OS/2 client) or beep (Windows client) and a "snooze" option
- Users can authorize other Time and Place/2 Version 1.0 users to view and/or update their calendar

Letter # 293-094, February 23, 1993

OS/2 32-Bit Application Migration Workshops

IBM offers software developers a unique opportunity to help migrate their existing DOS, Windows, UNIX, and 16-bit OS/2 applications to 32-bit OS/2. The IBM OS/2 32-Bit Application Migration Workshops give developers hands-on, in-depth training and personal assistance in migrating their applications to OS/2.

Leading-edge software developers can take advantage of OS/2 32-bit features such as thread management, multitasking, robust memory management, and full 32-bit Application Programming Interfaces (APIs) to make their applications run better and faster, and to position their applications for portability.

As more and more users demand OS/2 applications, developers who participate in IBM's OS/2 Application Migration Workshops will get a substantial jump-start on migrating their own applications to OS/2. The workshops minimize the learning curve associated with a new operating system. They also accelerate migration efforts by providing industry experts who instruct and assist in the migration process.

Find out for yourself how indispensable the OS/2 32-Bit Application Migration Workshops are for migrating your applications to 32-bit OS/2. Read the workshop descriptions, then decide which one is best for you.

The workshops are held at the Omni hotel in West Palm Beach, Florida, except for the UNIX to OS/2 32-Bit Workshop, which is held at IBM in Austin, Texas. The workshop includes meals, software, and use of hardware during the workshop—all at an affordable cost.

Windows 3.x to OS/2 32-Bit Presentation Manager Native Workshop (5 Days)

This workshop is for Windows 3.x application developers who want to expedite migrating their Windows code to the 32-bit OS/2 platform. This native port will enable Windows 3.x applications to exploit the power of OS/2 memory management, multithreaded design, and the Common User Access (CUA) Presentation Manager interface. The workshop

“Until this one, we had never attended a workshop that has paid back tenfold the time and money spent.”

involves extensive hands-on porting of the developer's source code, including instruction and assistance from a migration expert. Topics include the following:

- Architectural changes needed to port Windows 3.x applications to 32-bit OS/2
- Memory management
- Multithreaded design
- Introduction to PM
- Interprocess communication
- OS/2 DOS APIs versus C runtime
- Common controls
- IBM Multimedia Presentation Manager/2 (MMPM/2)
- Introduction to migration tools

Prerequisites for this workshop are Windows 3.x and C language programming experience, and an existing Windows 3.x application. Workshop participants receive the following:

- The latest release of OS/2
- IBM C Developers Workset/2
- IBM Systems Application Architecture (SAA) Common User Access Controls Library/2 (CCL/2)
- IBM MMPM/2
- Migration workbook

DOS to OS/2 32-Bit Presentation Manager Workshop (7 days)

This workshop is for DOS developers who want to expedite migrating their existing applications to the OS/2 32-Bit Presentation Manager platform. This native port will enable DOS applications to exploit the power of OS/2 memory management, multithreaded design, and the CUA Presentation Manager interface. The workshop involves extensive hands-on porting of the developer's source code, including instruction and assistance from a migration expert. Topics include the following:

- Mapping BIOS-based functions to PM APIs
- INT 21 functions and their 32-bit OS/2 equivalents
- Memory management
- Multithreaded design
- Introduction to PM
- Interprocess communication
- OS/2 DOS APIs versus C runtime
- Common controls
- IBM Multimedia Presentation Manager/2 (MMPM/2)

Prerequisites for this workshop are DOS and C language programming experience and an existing DOS-based C language application. Workshop participants receive the following:

- The latest release of OS/2
- IBM C Developers Workset/2
- IBM MMPM/2
- Migration workbook

System Object Model/Workplace Shell Workshop (5 days)

This workshop provides detailed information about OS/2 32-bit System Object Model (SOM) programming. It shows how to interact with the Workplace Shell and to take advantage of future OS/2 SOM enhancements with minimal impact to the source code. The workshop features extensive hands-on application development, with instruction and assistance from a SOM expert. Topics include the following:

- Similarities and differences between SOM and C++
- Interaction between SOM and the Workplace Shell
- Architectural changes needed to migrate from a typical OS/2 application to a SOM/Workplace Shell application

Prerequisites for this workshop are 32-bit OS/2 and C language programming experience, and knowledge of the principles of object-oriented programming. Workshop participants receive the following:

- The latest release of OS/2
- IBM C Developers Workset/2
- Migration workbook

OS/2 16-Bit Presentation Manager to OS/2 32-Bit Presentation Manager Workshop (5 days)

This workshop enables OS/2 developers to migrate their existing C language applications to the OS/2 32-bit PM platform. This migration implements the flat memory model, better interprocess communication, and the performance advantage of 32-bit execution versus the overhead of "thunked" 16-bit code. The work-

shop involves extensive hands-on porting of the developer's source code, including instruction and assistance from a migration expert. Topics include the following:

- Differences between OS/2 16-bit and OS/2 32-bit subsystems
- Memory management
- Multitasking
- Interprocess communication
- OS/2 DOS APIs versus C runtime
- Common controls
- Performance enhancement aids
- IBM Multimedia Presentation Manager/2 (MMPM/2)

"We succeeded in porting 75% to 80% of our application during the single week in attendance."

Prerequisites for this workshop are experience with OS/2 1.3 and C language programming, and an existing OS/2 1.3 application. Workshop participants receive the following:

- The latest release of OS/2
- IBM C Developers Workset/2
- IBM MMPM/2
- Migration workbook

UNIX to OS/2 32-Bit Workshop (5 days)

This workshop is for UNIX programmers who want to migrate or reimplement their applications on the 32-bit OS/2 platform, and for programmers who want to include OS/2 in their UNIX environment. The workshop fosters a broad understanding of OS/2 components by comparing and contrasting UNIX and OS/2. This

knowledge will help developers understand how to provide existing application function in the OS/2 environment, and how to exploit unique features in OS/2 to extend or enhance existing function. Capabilities and interfaces common to both UNIX and OS/2 are emphasized to expedite the migration process and to preserve application architecture and design wherever possible. Topics include the following:

- OS/2 product overview
- Programming/development environment and process
- OS/2 base operating system concepts
- Communications
 - Shared memory, pipes, queues, semaphores
 - TCP/IP, Sockets, Remote Procedure Call (RPC), NetBIOS, SNA
- Interoperability between UNIX and OS/2
- X-Windows to PM
- Security in stand-alone and LAN environments

Prerequisites for this workshop are UNIX/X-Windows and C language programming experience, and familiarity with standard development tools. Workshop participants receive the following:

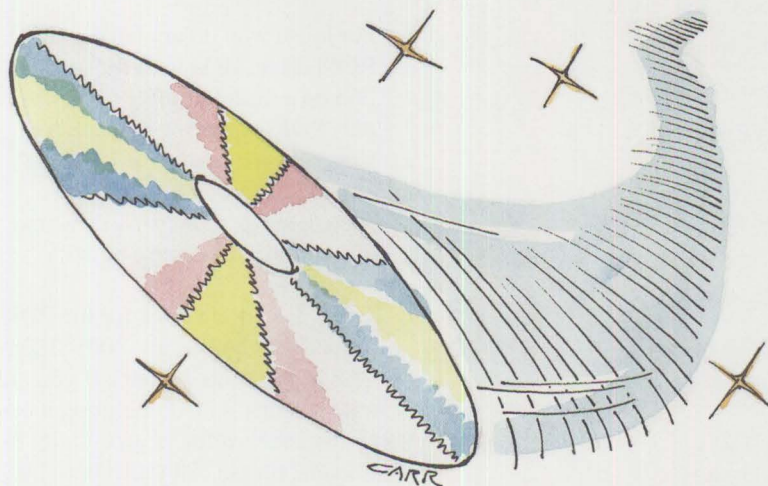
- The latest release of OS/2
- IBM C Developers Workset/2
- TCP/IP

For More Information

Contact the One Up Corporation and ask about the IBM OS/2 Application Migration Workshops. One Up's toll-free number, (800) 678-31UP (3187), is available Monday through Friday, 8:00 A.M. to 5:00 P.M. central time.

— Art Goddu, IBM Corporation, Boca Raton, Florida

IBM OS/2 Products Available on CD-ROM



IBM has produced a limited edition CD-ROM containing beta-level OS/2 tools. This CD-ROM, the Professional Developer's Kit (PDK), is available to all developers worldwide who join or are currently members of the Developer Assistance Program (DAP). In addition, it is being distributed at many computer shows.

The CD-ROM is currently in its fourth edition; each edition is updated with OS/2 products at beta level. The first PDK was introduced at the Software Developers' Conference in Boston in September 1992. The second was introduced at the IBM OS/2 Developers' Conference in New York City in October 1992. The third was available at COMDEX® in Las Vegas in November 1992. The fourth debuted at the OS/2 Technical Interchange Conference in Phoenix in February 1993.

The CD-ROM includes a wide range of OS/2 products. Some products can

be installed directly from the CD-ROM (denoted by C in Figure 1); some can be installed by creating diskettes first (denoted by D). Some can be installed both ways. This will be useful when you want to give diskettes to someone who does not have a CD-ROM drive. You can make up to nine copies of the code.

The February PDK contains the products listed in Figure 1.

The tools and utilities shown in Figure 2, installable from the CD-ROM, are also included. Some of the tools and utilities have been written by IBM employees and have had limited availability outside IBM.

Another plus on the CD-ROM is the vast library of IBM documentation, which includes the Red Books and the OS/2 Technical Library. This

Installation Method	Product
C D	IBM OS/2 2.1
C D	Communications Manager Client Server, Single User
D	Communications Manager Application, Debugger
C D	IBM C/C++ for OS/2 2.1
C D	IBM Developer's Toolkit for OS/2 2.1
C D	IBM OS/2 2.1 Kernel Debugger
C	IBM Multimedia Presentation Manager, Toolkit, Enhanced Samples
D	IBM Multimedia Presentation Manager Base
C D	IBM WorkFrame/2
D	NetWare Workstations Kit for OS/2 2.1
D	TCP/IP
D	Distributed Computing Environment
D	Network Services for DOS
D	32-bit PL/1 compiler
D	Multi-Protocol Transport Services

Figure 1. Beta OS/2 Products

Utility	Description
ALPHAL	Code browser and analysis tool
APPCUTIL	APPC Utilities (PING, TELL, Remote Execution)
APPCGAME	Included are REVERSI, CYCLES, and MANDPM
ASDT32	IBM Application/System Debug Tool (ASDT32)
CTFORMAT	Filter that formats Assembler, C, and text files
CDEXPL	A compact disc digital-audio explorer
DBMRPW	Remote Password Administrator utility
DINFO	Monitors the SWAPPER.DAT file
DIRSTAT	LAN Adapter DirStatus
EPM	An updated Enhanced Editor with macro support
FORBROWS	A PM forum browsing tool
HEXDMP	A binary data display and conversion utility
HYPERWRITE	32-bit PM WYSIWYG Help file editor
IBROW	Image browser, using the image toolkit
IPFCPP	IPFC Preprocessor
ITSCLABS	Samples of ITSC laboratory classes
MAGNIFY	A PM magnifying glass
PENPM	Toolkit for writing pen-based applications
PMATE	PM automated test tool
PMDCLIP	Clip art for PMDRAW
PMDRAW	Drawing tool for creating overheads for presentations
PMDRAW Samples	Sample PMDRAW images
PMFTRM	An async terminal emulator
PMGLOBE	An OS/2 Presentation Manager World Globe
PMPRTF	32-bit Printf programming support
PMTIMER	Timed execution of programs
PMTREE	Graphical display of interaction with PM windows
QCONFIG	Tool to capture machine hardware in a system
RXAPPC	Communications Manager APPC API from REXX programs
RXMATH	Basic math functions, subset of C Set/2 compiler
RXNETB	Gives access to the OS/2 NetBIOS API from REXX programs
SCRAP	Screen-capture utility, using the image toolkit
SLICK	Demo of a powerful programmer's editor for OS/2
SOMII	System object module preprocessor for C++, builds WP Shell
TINYED	Tiny OS/2 and DOS editor
TXT2PS	Printer filter to convert plain ASCII files to PostScript
VIEWALL	A tool that presents all the books on the bookshelf
VREXX	Visual REXX

Figure 2. Beta OS/2 Tools

paperless library can be read using Library Reader/2, a tool shipped with the PDK, and the Information Presentation Facility, which is part of OS/2 2.1. You can print selected sections of each book and even copy parts to the clipboard.

For those who do not have a CD-ROM drive, IBM includes information on a discount offer to purchase one. PDK support is provided on CompuServe in section 12 of the OS2DF2 forum. Questions are posted, suggestions are given, and tips and techniques are noted.

There is a nominal fee for the PDK. To get your copy, call (800) 3IBM-OS2 in the United States. If you are not currently a DAP member, there is an instant enrollment procedure. In Canada, please call (800) 465-2222. In European, Middle Eastern, and African countries, contact Mike Gove, Manager, International OS/2 Users' Group at 44-285-641175 (telephone) or 44-285-640181 (fax). In the Asia/Pacific Region (except Japan), contact Rohaini Cain, IBM Australia at 61-2-354-7684 (telephone) or 61-2-354-7766 (fax). In Japan, fax requests to Hiroshi Yasukawa, IBM Japan, at 81-3-3495-2045. In Latin America and South America (except Brazil), fax to Juan Carlos Fernandez, IBM Mexico, at 52-5-395-7812. In Brazil, fax requests to Carla F. Resenda Coelho, IBM Brazil, at 55-21-546-5082.

—Midge Portney, IBM Corporation, Boca Raton, Florida

Index to Past Issues of IBM Personal Systems Technical Solutions

January 1993 (G325-5020)

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 BIOS

October 1992 (G325-5019)

Exploring File Server Performance
PS/2 3.5-Inch Rewritable Optical Drive
Programming the XGA Video POS Registers
Video Monitoring on Personal Computers
Memory Address Space
OS/2 2.0 Installation and Performance Considerations
OS/2 2.0 Application Support
Cleaner Installation of Applications Under OS/2
Creating Resizable Pushbuttons
Configuring Parallel Ports for OS/2
Performance Characteristics of ES 1.0 Database Manager
AlertVIEW
Screen Reader/2

July 1992 (G325-5017)

IBM PS/2 Server 295: New Thresholds for Client/Server Networking
Comparing Architectures: Micro Channel and EISA (Part 2)
Synergy by Design
Pen-Based Computers
Why Doesn't My Portable's Battery Last Longer?
Planning Guidelines for Token-Ring Cabling
Installing and Migrating Applications in OS/2 2.0
Printing Under OS/2 2.0
Installing the IBM 4029 LaserPrinter Under OS/2 1.3
Serviceability Tools in OS/2 2.0
Online Communication Using the OS/2 2.0 PM Terminal
IBM Extended Services Database Manager
NetWare for SAA
Using the IBM DOS 5.0 Driver EMM386.EXE and Upper Memory
The Solutions Evaluation Tool

April 1992 (G325-5015)

Comparing Architectures: Micro Channel and EISA
Portable Computer Trends and Directions
LCD Panel Technology
The OS/2 Workplace Shell
New Applications in OS/2 2.0
Unattended Installation of OS/2 2.0
OS/2 Communications Manager Trace Events
IBM and Novell LAN Software Coexistence
IBM 8209 LAN Bridge Connects Ethernet Clients to Novell and IBM Servers
Backup and Restore in an IBM NetWare Environment
The DOS Protected-Mode Environment
DOS Disk Management
Customizing Alphanumeric Screen Dimensions

January 1992 (G325-5014)

Additions to the IBM PS/1 Family
IBM LaserPrinter 4029 Series Print Quality Enhancements
OS/2 2.0: The Integrating Platform
Multiple Virtual DOS Machines
IBM OS/2 LAN Server 2.0
OS/2 2.0 Memory Management
Coding for Performance Under OS/2 Version 2.0
Extending the Functions of OS/2 REXX
Protecting User Exits Under OS/2 1.X

GDDM-OS/2 Link
IBM Upgrade Enhanced Install Utility/DOS 5.0
Advanced Peer-to-Peer Networking: An Overview
Using IBM SAA Networking Services/2
The AAI Family of Products
Securing the Enterprise Workstation

Issue 4, 1991 (G325-5013)

Power Factor: Non-Linear Loads and the Power Distribution System
Database Manager: Highlights and Direction
OS/2 Communications Manager
Improving OS/2 Application Performance
Creating PM Windows with Dialog Templates
REXX Program for OS/2 LAN Server
Micro Focus COBOL/2 and the DOS Database Requester
IBM DOS 5.0 Facts and Features
IBM DOS 5.0 Upgrade
DOS 5.0 Performance Improvements
DOS Memory Management Facilities
Disk Caching Under DOS
NetWare Client-Server Interaction
LANACS Protocols

Issue 3, 1991 (G325-5012)

PS/2 Model L40 SX Laptop Portable Computer
OS/2 2.0 Considerations
Comparing PC-DOS, OS/2, and AIX PS/2 - Part 2
Using Dual Displays with OS/2
Local Area Networks: The New Utility
And You Thought LANs Were Just for the Office!
Remote LAN Management Tools
New Horizons for IBM's Shielded Twisted Pair Cabling
Tuning and Self-Tuning Features of OS/2 LAN Server
NetWare Communications and Routing Protocols
Little Solutions for LANs

Issue 2, 1991 (G325-5011)

IBM PS/2 Model 90 XP 486 and Model 95 XP 486
Choosing an I/O Bus Architecture
The Network Is the Message
Invoking Printer Job Properties
Comparing PC-DOS, OS/2, and AIX PS/2
Programming PM Using the COBOL/2 Bindings
Installing and Using the DOS Database Requester
OS/2 LAN Server 1.3 Overview
IBM Windows Connection 2.0
SNA Definitions for 3270 Emulators - Part II
IBM THINKable™

Issue 1, 1991 (G325-5010)

XGA - Raising Video Expectations
Choosing betw. Shielded and Unshielded Wiring for Data Transmission
Compatibility of LAN Servers and Requesters
Running DOS LAN Requester and Novell NetWare Concurrently
Breaking the 640 KB DOS Memory Barrier
Understanding an OS/2 CONFIG.SYS File
OS/2 EE 1.2 Database Manager Performance
OS/2 EE 1.2 Competitive Performance
An Intelligent Front-End EASEL® Application
Enabling Software for National Language Support
SNA Definitions for 3270 Emulators
Diskette Failures Caused by Contamination

Issue 4, 1990 (G325-5009)

First Look at the New IBM PS/1 Computer
Using the IBM 4019 LaserPrinter Effectively
Micro Channel Interface Chip Sets
Token Ring Bus Master LAN Adapters
Extension of Wiring Rules for 4-Mbit/s Token Ring Using UTP Lobes
SCSI and DISK.386.SYS
Operating System Platforms: A Business Perspective
Minimum OS/2 1.2 DASD Requirements
User Profile Management
Understanding OS/2 1.2 LAN Server Performance
PM: An Object-Oriented Approach
DOS 4.00 SHARE
A "C" Programming Model for DOS Device Drivers
An Electronic Bulletin Board for PC Users

“XGA-2 offers non-interlaced scanning at a high frame refresh rate of 75 Hz at 1024 x 768. (page 2)

“IBM’s goal for the 1990s is to bring information and people together. (page 4)

“It is easy to use REXX to customize the WPS as part of a CID installation. (page 22)

“The IBM LAN NetView Management products provide a framework and applications to implement OS/2-based distributed systems management solutions. (page 33)

“The IBM LAN NetView Start program represents networks graphically. (page 41)

“A client playing full-motion audio and video with CD-ROM quality from a server on the Token Ring uses an average delivery rate of 1.2 Mbits per second, giving 30 video frames per second. (page 49)

“The printer driver cannot be changed on a network printer because it is controlled by the print server. (page 57)

“WITT is an automated test driver that tests an application from an end user perspective by simulating end user keystrokes and mouse movements. (page 62)

“Users need several types of virus protection tools to effectively detect and eradicate viruses. (page 75)

“We succeeded in porting 75% to 80% of our application during the single week in attendance at the OS/2 Application Migration Workshop.” (page 100)

G325-5021-00

