Patricia Seybold's

VOL. 5, NO. 6
ISSN: 0887-3054
JUNE 1990

UNIX IN THE OFFICE

PRODUCTS • TRENDS • ISSUES • ANALYSIS

INSIDE

Desktop Unix

Open Desktop Aims to Be the Rallying Point for ISVs

By Ronni T. Marshak

THE BATTLE FOR the desktop continues to heat up. OS/2 was gaining on DOS and appeared to be the clear favorite. But then delays on the 32-bit version of OS/2 were announced. At the same time, sneak previews of Windows 3.0 were generating excitement. DOS gained momentum, OS/2 slipped. But what about Unix on the desktop? Can it succeed? Or is Unix destined to remain exclusively as one of the primary server technologies, distributing applications and services to DOS and OS/2 clients? *(continued on page 3)*

• E D I T O R I A L •

Spotlight: Distributed Computing

OSF's Decision Will Focus Industry Attention on DCE Options

By Judith S. Hurwitz

NO MATTER which side you've taken on the distributed computing issue, you must admit one thing: By issuing a Request For Technology (RFT), the Open Software Foundation (OSF) has propelled distributed computing to center stage. Distributed computing technology is a vital next step toward allowing users to make more practical use of the technology they have and finally to achieve the transparency they have been demanding for the last 10 years.

OSF's decision to adopt Apollo/HP's Network Computing System (NCS) RPC was a sound one. NCS supports interoperability among different operating systems, not just Unix, which makes it an appealing choice for enterprise-level computing environments. Companies like Hewlett-Packard, Digital Equipment, and IBM have already begun to make NCS part of their enterprise computing infrastructures.

We do not expect that OSF will have an easy time convincing the industry that its selection was unbiased and fair. Many (especially the Sun faction) will protest that the deck was stacked before the analysis started. One of the problems, in fact, is that some fierce market competitors began the process of melding their technologies together (the DeCorum proposal) before the selection process began. Who could have imagined three or four years ago that HP, Digital, and IBM would agree on a common technology base? But they did. The industry has become more accustomed to consortia of vendors. But we should not be suspicious of these joint projects. Several vendors working together before submitting a proposal may result in an extra measure of interoperability among components.

Ironically, had each vendor independently offered its technology to OSF, there would have been less controversy. Even if OSF made the same choice, it would have looked more independent. But the vendors involved felt strongly about the need for the technology they proposed and didn't want to risk the possibility of OSF not seeing how well these technologies complemented each other.

Despite the controversy, we believe that the OSF process of selecting technology worked. Members of the review team included representatives of AT&T and major Unix systems users. It appears that the team came to an agreement on technology and disagreed only on implementation and roll-out issues.

However, whether or not this team of experts performed well, and whether or not the technology selected was the best the industry had to offer, the controversy is bound to continue. Sun Microsystems is on the public relations warpath and will maintain its push for the de facto adoption of its distributed computing model, Open Network Computing (ONC). Sun is planning, though, to change ONC to provide many of the functions available in the OSF model. But Sun will not be able to deliver the new functionality until 1992. Unix International points to existing technology in System V.4, including NFS, RFS, and Streams, as its response to the OSF selection.

The bottom line is that achieving transparent distributed network computing will not be easy. Users will be confused by three different organizations (OSF, Unix International, and Sun Microsystems), each of which proposes a different model. Users need distributed computing, and they would like a single model to emerge. But at least the technology is becoming available and the goal of distributed computing will be realized—even if the initial implementations are incompatible.

Patricia Seybold's Office Computing Group

Publisher **PATRICIA B. SEYBOLD**

Editor-in-Chief **JUDITH S. HURWITZ**

Managing Editor
DOUGLAS A. FREEMAN

Senior Editors
LAURE BROWN
JUDITH R. DAVIS
Telephone: (617) 861-3926
RONNI T. MARSHAK
MICHAEL D. MILLIKIN
JOHN R. RYMER

News Editor
DAVID S. MARSHAK

Sales Director
RICHARD ALLSBROOK JR.

Circulation Manager
DEBORAH A. HAY

Customer Service Manager
DON BAILLARGEON

148 State Street, Suite 612, Boston, Massachusetts 02109 Telephone: (617) 742-5200 FAX: (617) 742-1028
MCI: psocg / 312 2583 Internet: psocg@dcm1sg.das.net TELEX: 6503122583

Patricia Seybold's UNIX in the Office (ISSN 0887-3054) is published monthly for $495 (US), $507 (Canada), and $519 (Foreign) per year by Patricia Seybold's Office Computing Group, 148 State Street, Suite 612, Boston, MA 02109. Second-class postage permit at Boston, MA and additional mailing offices. POSTMASTER: Send address changes to *Patricia Seybold's UNIX in the Office*, 148 State Street, Suite 612, Boston, MA 02109.

•SCO•

(continued from page 1)

The Santa Cruz Operation (SCO) is betting on both client and server roles for Unix. To that end, the company, along with Digital Equipment Corporation, Locus Computing Corporation, Relational Technology Incorporated, and Tandy Corporation, introduced Open Desktop in early 1989, which SCO is calling a *complete graphical operating system.* Open Desktop is actually an integrated offering of SCO Unix, DOS, networking services, and SQL relational DBMS (RDBMS), all operating under a graphical user interface and Desktop Manager. The offering runs on a 386 or 486 workstation with 6MB and a 100MB hard disk. SCO hopes the product will become the de facto Unix desktop and that ISVs will flock to the platform, thus providing customers with retail, shrink-wrapped Unix applications to run on their PCs.

Product Philosophy

When SCO et al. took a good look at the market in 1988, it was about the time hopes were being pinned on the OS/2 desktop environment. The collected companies recognized that OS/2 might not prove to be the savior desktop platform that it claimed to be. Naturally, with its Unix bias, SCO started considering the possibility of Unix claiming some of the desktop real estate in commercial markets.

PIECES ALL IN PLACE. According to Doug Michels, executive vice president of SCO, all the pieces were there. Customers were becoming perfectly willing to invest in 386 hardware with VGA monitors, a configuration which is satisfactory for graphical windowing environments. (Michels acknowledges that a 386 is not necessarily the optimum platform for robust windowing applications, but it is, he states, satisfactory.) In addition, memory prices were dropping, so the earlier limits were going away. The way Michels sees it, the hardware was there, TCP/IP and NFS were there, Unix was there, X Window was just about mature enough, and the only missing piece was a graphical user interface. Then along came Motif from OSF. According to Michels, "Motif was a natural for the PC part of the industry." The look and feel is much like OS/2 Presentation Manager's, and the Motif toolkit was stable enough for ISVs developing applications.

MISSING: TARGET DEVELOPMENT PLATFORM. Still, something was holding back Unix on PCs—a lack of a target platform for developers. ISVs didn't know what to write to. Even though the industry had pretty much settled on Unix V.2 as a de facto standard, no single graphical user interface (GUI) or networking protocol had emerged as a clear leader. Even if a software developer picked a platform and protocol, the user was stuck with elaborate installation procedures to make sure every piece of the architecture worked together correctly. You couldn't just install the application and get going. You had to install all the drivers, all the transports, etc. for each application. The installation process became very complicated. "If we left it to the users, they might never get it configured quite right," warns Michels.

SCO, along with many others in the industry, recognized the need for a common platform that provided three aspects: the GUI, the operating system, and the transports—a single platform environment for desktop Unix. In addition, the environment needed to include a networked SQL engine. According to Michels, "Modern applications will be written to assume underlying corporate data. Embedded SQL is the current standard—the only one we've got. Networked SQL has to be part of any modern platform."

The Digital Connection

THE STORY GOES like this. When OS/2 and, more specifically, OS2EE were introduced, Digital saw IBM attempting to ramrod a proprietary client operating system to the industry. Never a Big Blue fan, Digital did not at all appreciate the idea of IBM owning the desktop. Some innovative Digital marketeers got together and realized, hey, wouldn't it be great if there was a non-OS/2 alternative for the 386! They pulled in some people from Ingres, with whom they already had a close working relationship; Tandy, which supplies Digital with its PCs; SCO, which virtually owns the PC Unix operating system market as well as the distribution channels; and Locus, which could provide DOS connectivity. Digital itself would provide the user interface tools with the DECwindows toolkit. And, just for good measure, the plan was to add TCP/IP and NFS to the proposed offering. Thus, the concept of an integrated Unix desktop was born. Just before Open Desktop, as the product came to be called, was announced, OSF announced Motif, so the user interface portion was represented with the Motif toolkit. SCO was then handed the ball and charged to run with it.

Now that the product is out, Digital has fully endorsed it to run on its 386 platforms, the DECstation 316, 325, and 333 (all manufactured by Tandy). Digital will also resell Open Desktop and provide support to Digital customers.

INDUSTRY CATCH-22. So, basically, the Unix desktop market faces a catch-22: Customers won't buy into the platform because there are so few applications; on the other hand, ISVs don't see sufficient market potential to justify development costs. As a result, few developers are building innovative graphical networked Unix applications.

With a clear target platform, Michels anticipates, developing these types of applications will become more attractive to software vendors. And, when innovative commercial applications are available, more customers will buy.

Michels sees the resolution of this catch-22 as a particular boon for those companies committed to running on open systems, including the U.S. government.

THE ROLE OF OPEN DESKTOP. SCO took a bold step, integrating together a number of full-featured software offerings at a very attractive price and going great guns after VAR and ISV commitments. Open Desktop is the realization of this target platform for this market. SCO is counting on the product to open up the shrink-wrapped Unix desktop market, thus giving ISVs a target market segment sufficiently large to be worth their attention.

Actually, SCO was somewhat surprised at the reception Open Desktop got, not from ISVs, who are very interested, but from the excitement generated from corporate MIS users. Among the initial orders for the Open Desktop development system (see Illustration 2) are a number of orders from Fortune 500 companies, some of whom are ordering hundreds of systems. These MIS types are planning to use Open Desktop as the target platform for in-house end-user systems. Some, such as Harris Corporation, plan to use Open Desktop as a platform for system integration for their customers as well as for internal applications (see box, page 14).

As Michels says, "We've got a little snowball rolling right now. Hopefully, it will become a big snowball. That is, if it doesn't rain."

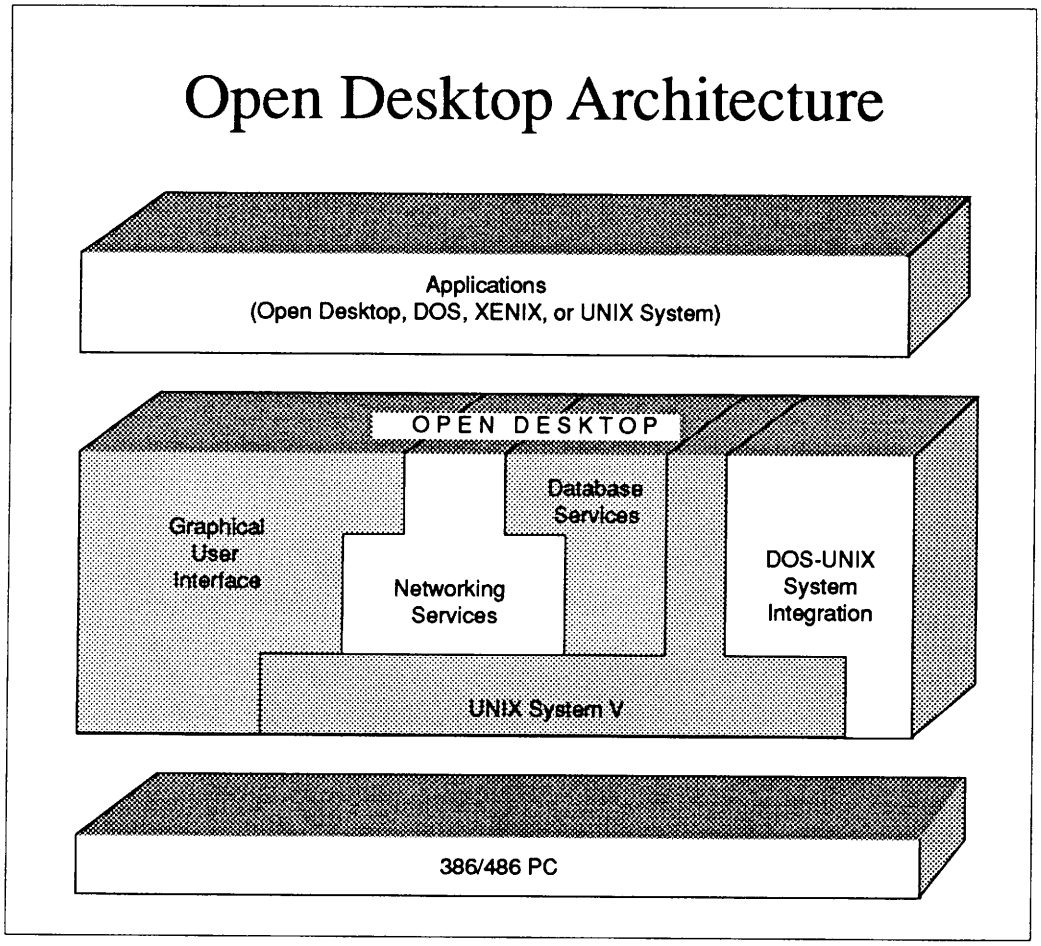


Illustration 1. Open Desktop provides the horizontal services that underlie DOS and Unix applications written to an Intel 386 or 486 platform. The platform includes a graphical user interface and desktop manager (Motif and IXI's X.desktop), networking services (TCP/IP, NFS, and LAN Manager), database services (Ingres RDBMS including networked SQL), Unix System V (SCO Unix V/386 Release 3.2), and DOS and Unix system integration (Merge from Locus and Xterm—part of the X.11 window system).

Architecture

Open Desktop is not an application, but a collection of industry-standard horizontal services that underlieUnix applications written to an Intel 386 or 486 platform (see Illustration 1).

There are three versions of the Open Desktop family (see Illustration 2 on p. 5):

- Open Desktop. The end-user product includes SCO Unix System V/386 Release 3.2, Motif, a Desktop Manager, TCP/IP, NFS, LAN Manager Client, SQL RDBMS with interactive SQL user services and Networked SQL, DOS 3.3, and DOS-Unix integration services. A User's Guide and an Administrator's Guide are included.

- The Open Desktop Server is a supplementary upgrade to the Open Desktop operating system. The upgrade allows a 386 or 486 workstation to act as a server for distributed client-server applications. The workstation can also act as a de-

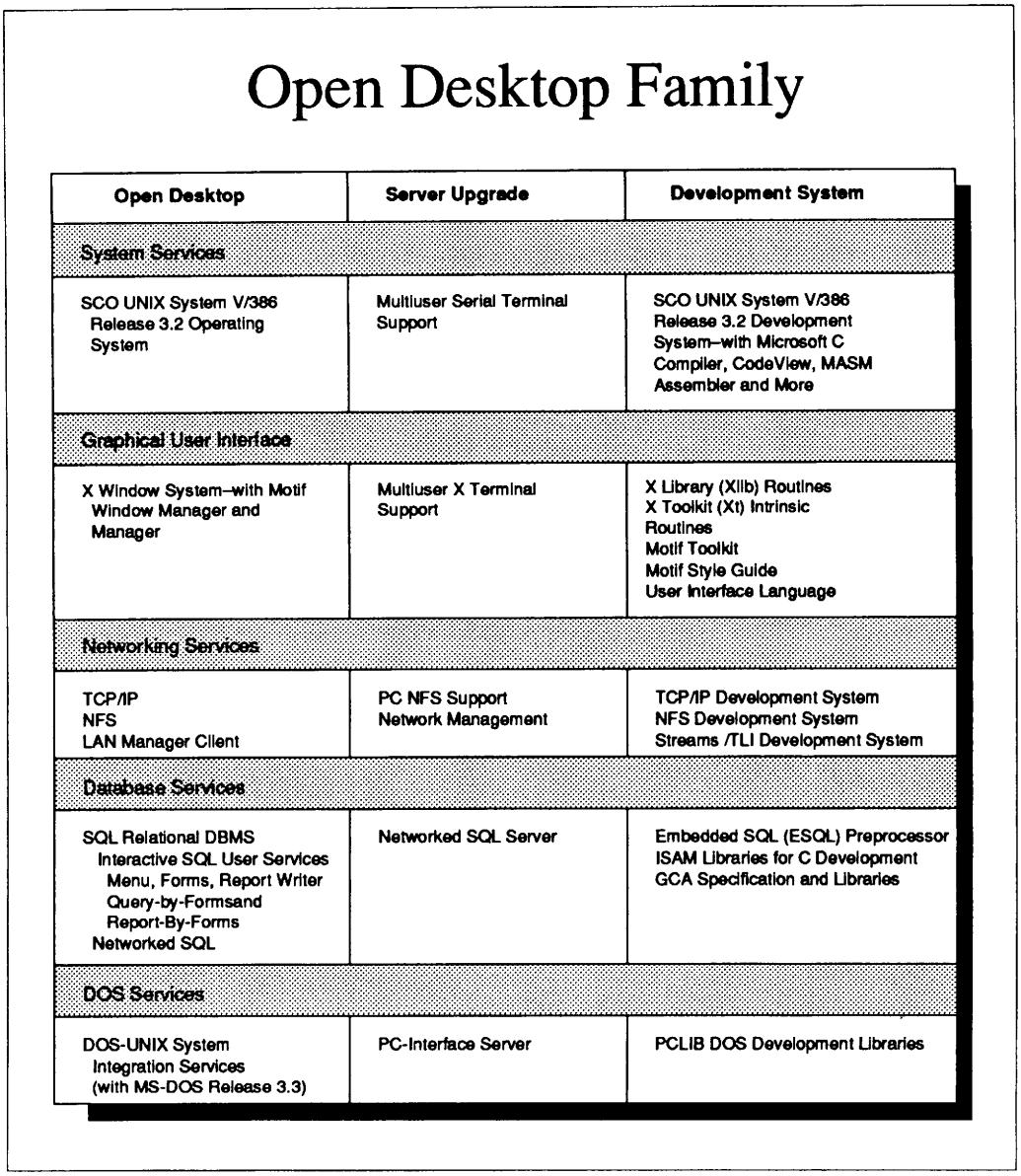


Illustration 2.

partmental server for DOS, OS/2, Xenix, Unix, or Open Desktop networks. The upgrade includes support for multiuser serial terminals, multiuser LAN access, and X terminal, SQL, and PC-Interface. The basic Open Desktop package is required. Complete documentation is provided.

- The Open Desktop Development System is a supplementary upgrade to the Open Desktop operating system that provides standard APIs for the graphical interface, networking, database, and DOS-Unix integration services. The upgrade includes development-system versions of each Open Desktop component, interactive debugging tools, and complete documentation. The basic Open Desktop package is required.

SYSTEM REQUIREMENTS. Open Desktop runs on either a 386 or 486 system based on ISA, EISA, or MCA. The minimum memory requirement is 6MB RAM and 100MB hard disk (see Illustration 3 on p. 6). Users must remember, though, that the minimum configuration will not be sufficient to run robust applications (e.g., Ingres, Uniplex, Alis, etc.) with any sort of reasonable performance. The base system, a 386 with 6MB, will run Open Desktop and, say, a word processor or spreadsheet.

Performance Issues. SCO is the first to admit that the current version of Open Desktop is slow in certain circumstances. Speed on a 25 MHz machine was compared to the lower end of the Sun line, about a Sun3. Though other factors play a part in performance, including I/O speed, graphics adapter speed, memory, and application, in general, to achieve really zippy performance, you need about a 33 MHz system. Doug Michels doesn't feel that 33 MHz is an outrageous expectation. "Not too long from now, that will be the minimum speed," he anticipates.

The performance seems especially slow compared to today's RISC machines, which are boosting users' standard expectations of performance.

But, Michels points out, you pick the price and performance you want. And SCO has not finished optimizing performance. The new version of SCO Unix, due this month, promises to be much speedier. Similarly, both X.11 Release 4 and Motif 2.0 will include significant performance improvements.

Still, one ISV we spoke to warned of the danger of seeing Open Desktop as a PC replacement, and, thus, buying MHz and memory as you would for a PC. Rather, it is a workstation replacement, requiring the speed and the memory commonly associated with that market.

User Interface and Desktop Behavior

DESKTOP MANAGER. SCO has chosen to go with Motif as the Open Desktop look and feel (see Illustration 4 on p. 7). But Open Desktop's graphical windowing environment, the Desktop Manager, based on X.desktop from IXI, goes beyond the pure display capabilities of Motif, providing some management capabilities as well, including directory manipulation and online help.

System Requirements

Open Desktop	Server Upgrade	Development System
CPU: 386 or 486 Computer Based on ISA, EISA, or MCA		
Media: 5.25" or 3.5" Disk or QIC 24 Tape		
Display: EGA,VGA, Extended VGA, Hercules Monochrome, or Selected, High-Performance Adaptures		
Mouse: Bus or Serial		
Network Card: 3C501, 3C509, WD8003E		
RAM: 6 Mbytes **Hard Disk:** 100 Mbytes	**RAM:** 8 Mbytes (plus .5 Mbyte per user) **Hard Disk:** 180 Mbytes	**RAM:** 8 Mbytes **Hard Disk:** 140 Mbytes

Illustration 3.

Default Desktop. Each user has a default desktop that can be modified—new items can be added and existing items can be returned to their original directories (see Illustration 5 on p. 9). The objects on the default desktop include:

- Home directory, as assigned during installation.
- Unix Window, which provides a character-based window into the SCO Unix V/386 Release 3.2 environment. Multiple sessions in multiple windows can be supported.
- DOS window. Multiple DOS sessions in multiple windows may be accessed.
- Xman. Xman is a manual browser that comes with the X toolkit. The product lets you browse through online manuals.
- Root directory.
- Help. Online help on Open Desktop functionality.
- Print, which is activated through drag and drop. To print a file, drag it onto the print icon and release the mouse button.
- Trash, which is activated through drag and drop. To delete a file, you drag it onto the Trash icon and release the mouse button. The left mouse button moves the file into the Trash directory. The right button permanently deletes the file.
- Data access, which is access into Ingres. Other SCO Unix-based RDBMSs may also be accessed from the Desktop by assigning a data access icon to the executable DBMS file.
- Mail, which provides immediate access to Unix mail. Users can substitute their primary mail systems for the default mail system. Files to be sent via mail without a cover message can be dragged and dropped on the mail icon. Customers can configure the mail system to use the word processor of choice to compose messages. The word processor can be either Unix based or DOS based. But if it is a DOS application, you need to include the conversion utility that is provided with the Open Desktop DOS integration functionality to ensure that every mail recipient can read the message.
- Editor. The default editor is vi, but the user can substitute any Unix or DOS editor by changing the editor variable in the .profile file in each home directory.
- Sysadmsh, system administration shell, is actually part of SCO Unix. The shell features ring menus for all common system administration tasks. For example, a new user can be added without having to access the /etc/passwd file. The system administrator needs to understand his or her system, but doesn't need to know Unix.

Desktop Manager Menu. The Desktop Manager Menu provides access to window functions including:

- Align, which aligns icons on an invisible grid
- Reorganize, which lines up icons at the top of the window
- Select All, which selects all icons

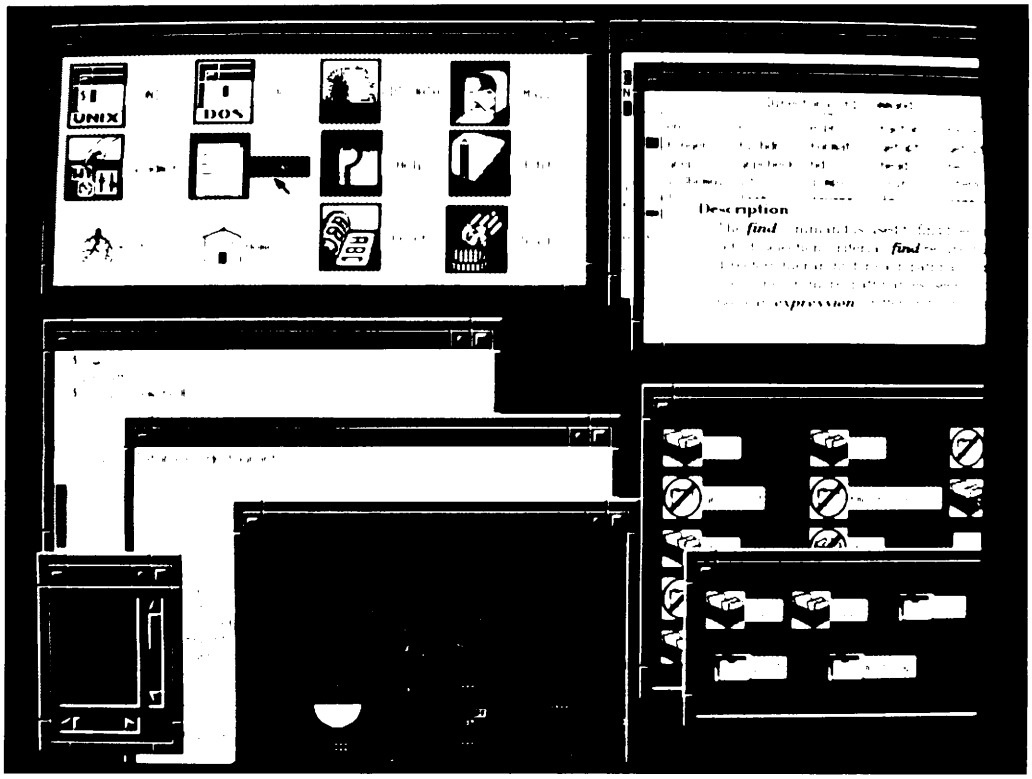


Illustration 4. The Open Desktop screen features the Motif look and feel, which is similar to the look and feel of OS/2 Presentation Manager. Notice that both a DOS and Unix Window are active on the screen.

- Put Back, which returns any selected icons to their original directories
- Icon Info, which displays information about selected icons and lets you change file permissions
- Shell window, which opens a new Unix Window
- Stop Desktop, which quits Open Desktop

Directory Menu. Each directory window has a Directory Menu, which acts on the contents of that specific directory. Functions include:

- Displaying the directory contents in alphabetical order by the time each file or directory was created or modified, by class (directory files first, executable next, text and data last), or by size (smallest to largest)
- Selecting all files and directories in the window
- Icon information, including setting file attributes such as read/write permissions
- Duplicating selected files
- Creating a new directory within the current directory
- Creating an empty file
- Closing the directory window itself

An icon is also provided for switching between an icon view of files and directories and a character view of the same directory. While the icon view is preferable for manipulating files (dragging and dropping), the character mode takes up less room and allows more filenames to be displayed within the window.

BEHAVIOR. While the Open Desktop desktop has the standard Motif look, certain paradigms are the province of the Desktop Manager and can differ from other Motif-based windowing systems.

Drag and Drop. The Desktop Manager supports the use of icons, using a drag-and-drop paradigm for icon manipulation. For example, to move a file from one directory to another, you simply drag the file icon from its original directory to the new directory and drop it there.

Icon Creation. A predefined set of standard icons is provided within Open Desktop. This is a limited set, so, for example, all executable files would have identical icons. Alternate icons may be created or modified using bitmap, the (limited) X Window icon editor.

One particularly nice bit of icon usage is the symbol for a file to which you have no access or permission: This universally recognized ghostbuster-type symbol appeals to our sense of whimsy.

Click to Activate. Open Desktop supports the "click to activate" paradigm as the default. This means that you must click within the window you wish to activate. You can choose to configure the system with an alternative mode called the "real estate" method: Whatever window happens to contain the mouse pointer (without clicking) is active; if you move your mouse accidentally, you find yourself positioned in some other window. (You can probably tell which method I prefer. But both paradigms have strong supporters.)

Ingres Server Functionality

By Laure Brown

IT'S HARD TO GIVE you a picture of the Open Desktop database module without giving you an update on the Ingres relational database management system (RDBMS). That being the case, we're providing one (albeit a brief one).

PORTABLE APPLICATION DEVELOPMENT. Ingress excels especially in its server functionality. It has two mechanisms for dealing with heterogeneous databases: its general communications architecture (GCA), an implementation of ISO's remote data access (RDA) specification for the client/server protocol and a gateway server that runs on the platform with the DBMS server. These servers can be Ingres, IBM's DB2 or SQL/DS, or Digital's Rdb.

There's also Open SQL, a set of SQL statements that is common to the servers supported in GCA. Ingres also provides an Open SQL toolkit that has a language preprocessor, so developers can embed Open SQL statements into an Open Desktop application. The application can then access any of the databases mentioned above; the gateway makes the appropriate translations. In other words, the application becomes DBMS independent.

Limitations. Aside from the fact that GCA doesn't support rival databases such as Oracle, Informix, etc., Ingres's solution has a few shortcomings. The structure demands that the application be an Ingres application, and the Ingres-developed gateway server is necessary to translate SQL statements, datatypes, error codes, and the data dictionary. It also cannot access non-SQL-based DBMSs. Furthermore, Open SQL is, by nature, lowest-common-denominator technology. Ingres has tried to overcome this last limitation by implementing some features that aren't supported by all the DBMSs included in GCA. For example, Open SQL supports repeat queries and emulates them for DBMSs that don't support those queries.

SERVER FUNCTIONALITY. Ingres includes advanced distributed data management capabilities. In addition, Ingres offers optional extensions for object and knowledge management. Not only do these features put the product significantly beyond what Oracle currently offers, it also implements features that have impressed us about Sybase.

Ingres maintains a multithreaded, multiserver architecture which was designed to take advantage of the powerful multiprocessors emerging in the market. It also features database procedures, which, like Sybase's stored procedures, are 4GL functions that are compiled, stored, and managed by the server rather than the application. They are typically used for predefined transactions. The point is to let a single procedure be accessible by multiple applications, thereby reducing network traffic as well the amount of code and logic that must be pumped into an application. Ingres also has a distributed query optimizer and a number of I/O reduction techniques, such as fast commit, group commit, and multiblock reads and writes.

Recent enhancements have made the Ingres server functionality more complete. The most notable of them are distributed transaction support and online backup. Ingres adopted a two-phase commit protocol for distributed transactions. The protocol separates a commit into two phases (prepare to commit and commit), which ensures that all participating nodes can actually take part in the transaction before it tries to commit. Online backup guarantees high availability—a must for mission-critical environments.

Optional Features. Ingres has also developed extensions that improve the functionality of its server. Unfortunately, these options are not part of the core database product. Therefore, they're not part of Open Desktop and must be purchased separately.

One is an object management extension that offers user-defined datatypes, such as geographic coordinates, temperatures, weights, and time-series data. User-defined datatypes are pretty handy in that they spare programmers from having to learn the rules for storage and manipulation of unconventional datatypes and from having to program them for each client application. Furthermore, since the rules are server based, you're guaranteed that they have been implemented correctly. Ingres has made them even handier by letting users define operators and functions to be used with these datatypes—to calculate the distance between two locations, convert pounds to ounces, and calculate volume. For instance, the user could teach Ingres about inches, feet, and yards, and how to add and multiply them. These datatypes can then be manipulated with standard SQL.

Ingres has also recently released a knowledge management extension, which, basically, adds a rules system to the database environment. In other words, it captures referential integrity rules and business policies (or data thresholds your business requires). You may include an unlimited number of rules per table. Furthermore, rules may be nested and recursive—both to an unlimited level. Knowledge management also adds resource and access control systems to the server. Resource control prevents users from using up their quota of resources before they actually start a database activity. Access control lets you give groups and applications permission to access data.

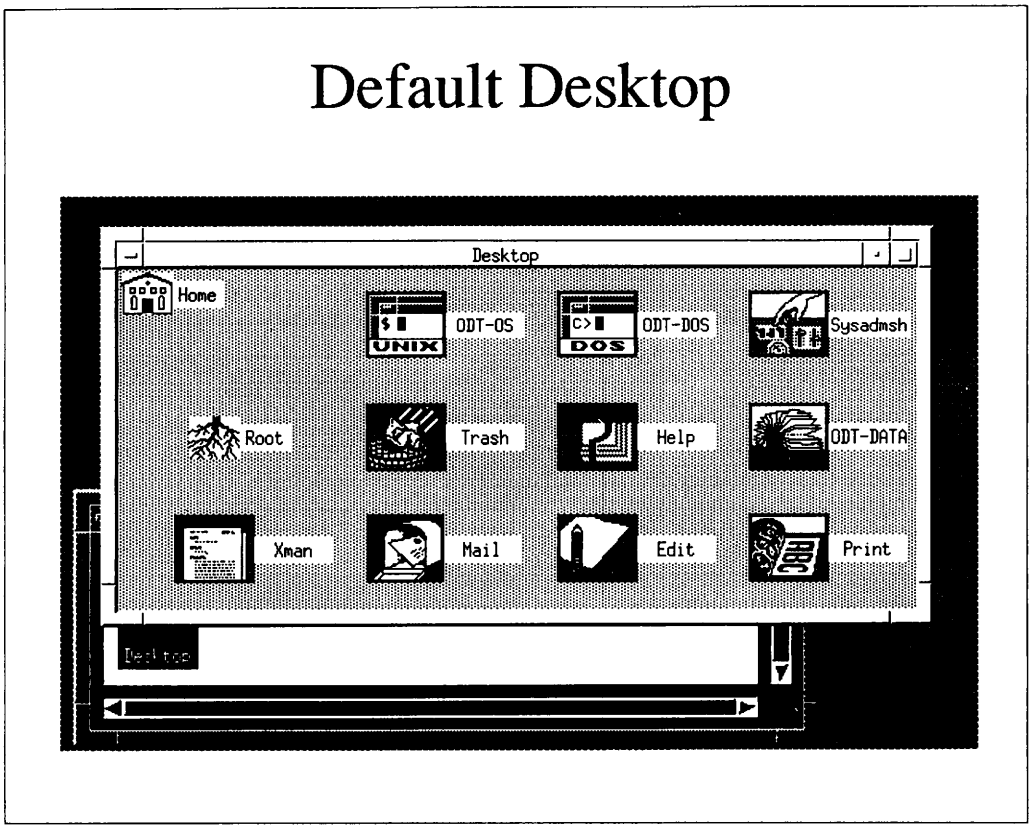


Illustration 5. The default desktop of Open Desktop allows you to access horizontal services, such as print and mail, by dragging and dropping document icons onto the service icon. Your standard editor and SQL database manager are also available from this initial desktop. Commonly used applications may be moved to the desktop by dragging and dropping the application icon from its original directory. Actually, none of the applications, services, or directories resides in the default desktop. The icons represent pointers to their actual locations.

Inconsistencies with Motif. Currently, the Desktop Manager is only "Motif'd" on the surface. The window 3-D shading and icons look like Motif, but menu operations differ. The X.desktop interface features pop-up rather than pull-down menus. This can be disconcerting. Even if all the applications running on your system sport Motif (though they do not have to), you go into a very different paradigm when manipulating desktop windows and directories. To further complicate matters, the Help feature sports yet another interface: ring menus, no scrollbars, no mouse support. And Xman features still one more interface including a left scrollbar and a three-button mouse paradigm.

The next release of X.desktop will feature a complete Motif interface. Open Desktop's Desktop Manager should get its Motif interface-lift soon after the new release is completed.

Mouse Usage. The product is designed for a two-button mouse, though a three-button mouse may be installed. In the three-button model, the middle and right buttons perform the same functions. Some programs are designed to work with three-button devices. While the two-button mice can access the applications, some features may be unavailable. Two programs that come as part of Open Desktop, Xman and Xterm (the UniX Window emulator), require a three-button mouse. For these programs only, using the Shift key plus the left button emulates the middle button.

Open Desktop uses the "double click (of the same mouse button) to execute" paradigm. In general, the left button selects and the right button extends. When you open an application or a directory, the left button creates a new window for the application; the right button substitutes the contents of the new application into the active window.

While mouse-button conventions can be changed (for example, left-handed users often want to switch the function of the buttons), it isn't an end-user task. One ISV commented, "You have to really not like it to want to change it."

The Open Desktop Database

Currently, Open Desktop comes complete with Version 6.1 of the Ingres relational database system. Version 6.2 will be included in Open Desktop Version 1.1 (due before the end of the year), and Version 6.3 of Ingres will be part of Open Desktop Version 2.0 (due in 1991). The rationale behind using Ingres as the underlying relational database for Open Desktop is more obvious from SCO's point of view than from Ingres's. SCO correctly assumed that custom applications will be increasingly dependent on corporate data. So the company sought a robust, distributed SQL DBMS that would accommodate other databases, and Ingres fit the bill. Application tools were not really an issue—which isn't to say that the Ingres toolset is not up to snuff, because it is. The point is, however, that SCO doesn't want potential customers hemming and hawing over whether or not to move to Open Desktop because they have standardized on a different database. Ingres is the one that comes with the environment, but you can install whatever database you want.

Open Desktop is advantageous for Ingres as well. The

company has recently gone public and is on a huge marketing blitz. Trying to emerge from its image as a good product that's shadowed by Oracle's marketing success, Ingres needed a vehicle for market recognition. Enter Open Desktop. If the platform succeeds the way SCO hopes, Ingres would penetrate the market much more deeply than it could as a standalone product.

DATABASE TOOLS. Besides the back-end administrative facilities, Open Desktop provides tools for developers and users. For developers:

- A module for embedding SQL statements, forms, menus, and other tools into a host language program; access to embedded query languages, the visual forms editor, and a report writer
- A visual forms editor
- A report writer

For users:

- A menu-driven interface that ties together the database subsystems
- A forms package for querying
- A forms-based report writer
- A more robust, command-driven report writer
- An interactive, forms-based SQL editor

DATABASE INTEGRATION. However, you don't necessarily need to be inside the database module to perform certain operations, and you don't necessarily need to use the Ingres database. Open Desktop has some on-the-fly data access facilities from a desktop window—or Data/WindowView, as it's referred to in Open Desktop. Within Data/WindowView, you can type in SQL commands, select fields from a form, cut and paste text, run a report, and query by form. These Motif-based, menu-driven facilities can access any Open Desktop database or other database supported by an Ingres gateway.

Obviously, SCO doesn't want Ingres's involvement in Open Desktop to stop other RDBMSs from writing to the environment.

NO ISV SUPPORT YET. Obviously, SCO doesn't want Ingres's involvement in Open Desktop to stop other RDBMSs from writing to the environment. Therefore, SCO is stressing Open Desktop's relational capabilities instead of its Ingres-specific applications. Yet, none of the database companies we spoke with intends to write specifically to Open Desktop. Oracle, for example, has "no plans whatsoever" to write to Open Desktop. We expected as much; Oracle wouldn't be enthusiastic about playing on Ingres's back yard. Oracle doesn't support Motif, so, at the moment, it couldn't run in the environment anyway. Sybase, for another example, doesn't yet support either Motif or SCO Unix 3.2. In most cases, it doesn't seem to be that other database companies are reluctant because of Ingres; it's just that they're technically unprepared to support Open Desktop any time in the near future. Either they don't yet support Motif or they don't support SCO Unix.

Access to DOS and Unix. Access to DOS is provided through Locus's Merge 386 and PC-Interface server products. Multiple DOS windows may be open and active simultaneously. All you have to do is double-click on the DOS icon in your default desktop. You end up staring at the familiar DOS C> prompt. In addition, DOS disks can be accessed. In fact, you can boot a DOS disk in an Open Desktop window.

A native Unix Window is similarly accessed from the desktop. This takes you to the good old $ prompt. Any Unix application can be loaded through this window. This is very significant; it means that you aren't limited to using applications written to the Open Desktop specifications. As long as the application is written to SCO Unix System V/386 Release 3.2, the application will run.

Networking Protocols

Open Desktop uses TCP/IP transport and comes complete with NFS and LAN Manager client software, allowing connectivity to standard Unix and OS/2 and DOS networks, as well as any proprietary network that supports either NFS or LAN Manager. The product supports both Ethernet and Token-Ring networks.

In the next release, NFS server capability will be included in the base offering of Open Desktop, so each workstation or PC on the network can access files distributed across other Open Desktop machines on the network.

SCO has not yet determined which protocol to support for remote computing. Basically, the company is waiting to see what emerges as the de facto standard. Our guess is that Open Desktop will incorporate whatever remote computing protocol is chosen by OSF.

Similarly, no networking services, such as authentication, security, time services, etc., are part of the product. This area is "under investigation." Again, SCO is waiting for a clear standard to emerge.

Finally, in the area of network management, SCO is again playing the waiting game. But the company does plan to offer a graphical front end to whatever network management system is

offered. Network management, when available, will be included in the Server Upgrade version of Open Desktop, most likely in the next release.

Open Desktop Components

We asked why the current suite of products was chosen to make up the Open Desktop platform. Doug Michels explained that there were several considerations. First, the components had to be consistent with emerging standards. Because it was in contact with most of the players in the industry, SCO could choose those products that it felt best met the demands of the industry. But another major consideration was reputation of the company and prior relationships with SCO. Michels admits that, in some cases, there might be better products out there in some areas, but he considers this to be a short term advantage. Open Desktop is a long-term strategic product. Thus, the kind of company, the pricing, and the commitment to Open Desktop were important factors in the decision. The companies which are providing the components are all committed to "working together in the long term to syncronize the evolution of Open Desktop," according to Michels.

OPEN DESKTOP COMPLIANCE NOT NEEDED. If you would prefer to use few or none of the Open Desktop services, you can choose to bypass applications in the installation process. For example, you can bypass the data service (Ingres) installation in favor of using Oracle.

And Oracle doesn't need to be written to Open Desktop to function properly. As long as it's written to SCO Unix, it will operate fine in the Open Desktop environment. If the database manager is also written to Motif, it will look like the rest of the environment, but it won't change the functionality of the application within the environment. The advantages to specifically writing to Open Desktop are more marketing—the ability to promote an Open Desktop version—and the installation considerations. With Open Desktop, a range of services are installed. Open Desktop-compliant applications assume that these services exist, and the user doesn't have to specifically tell the application about them.

Using Open Desktop

THE INSTALLATION PROCESS. Open Desktop is not intended to be installed by casual end users. It is, after all, a Unix product, and the multitasking, multiuser nature of Unix still requires system administration, much as the workstation market does. To that end, SCO is requiring that all VARs and Open Desktop distributors (and their resellers) preinstall the system and ship it fully loaded. SCO has strict requirements in training and installation for their resellers to ensure that the customer gets a fully loaded system that will be up and running quickly and effectively.

For those customers who don't buy through a distributor, the installation process is geared more for the nontechnical administrator instead of the Unix groupie.

Instructions are very clear, though not graphical. The process looks too much like Unix for our taste. There is one point where the interface changes and you are presented with a menu screen for installing the VGA monitor. This, we were told, is the interface for SCO's own applications. It would be nice if the entire process worked with these nicely designed, though character-based, menus. But, even as it is now, the process is straightforward. In fact, with a list of device names for the installation, even a novice can handle it.

On the other hand, the installation process uses well over 40 diskettes, a number of which are reused several times. It takes hours of swapping floppies to get Open Desktop up and running. An installation tape is available, but, for tapeless systems, be prepared to shuffle diskettes.

Our solution to the installation blues is bundling agreement with PC vendors. Then, when I buy my 386, Open Desktop is already installed. All I have to do is indicate which devices are specific to my network (by menu, of course). SCO would not commit to any specific bundling agreements, but we suspect several are already in the works.

SCO is also working on providing a CD-ROM-based installation.

Open Desktop is not intended to be installed by casual end users. It is a Unix product, and the multitasking, multiuser nature of Unix still requires system administration.

WORKING IN THE ENVIRONMENT. As mentioned earlier (see "Inconsistencies with Motif" page 9), you run into inconsistent behavior in the desktop manager when you go past the top level. The different interface paradigm, however, isn't difficult to master, and it actually works pretty well with Motif.

The interface within the applications is left totally to the application itself, though Open Desktop-compliant programs are Motif based. This means that they most likely will follow the Motif style guide in designing their interface. (Incidentally, the Motif style guide is the same as the Presentation Manager style guide.) We had no difficulty opening several applications within multiple windows, nor moving between them. Some applications were easier to use than others, but that's because of the complexity of the software, not the environment.

Opening multiple DOS and Unix sessions is very simple. There is the shock of moving into the character-based world, but, at least in DOS, you could go into Windows, which operates much like Motif.

Single-Source Documentation. One of the selling points of Open Desktop is the user documentation. Though not all the

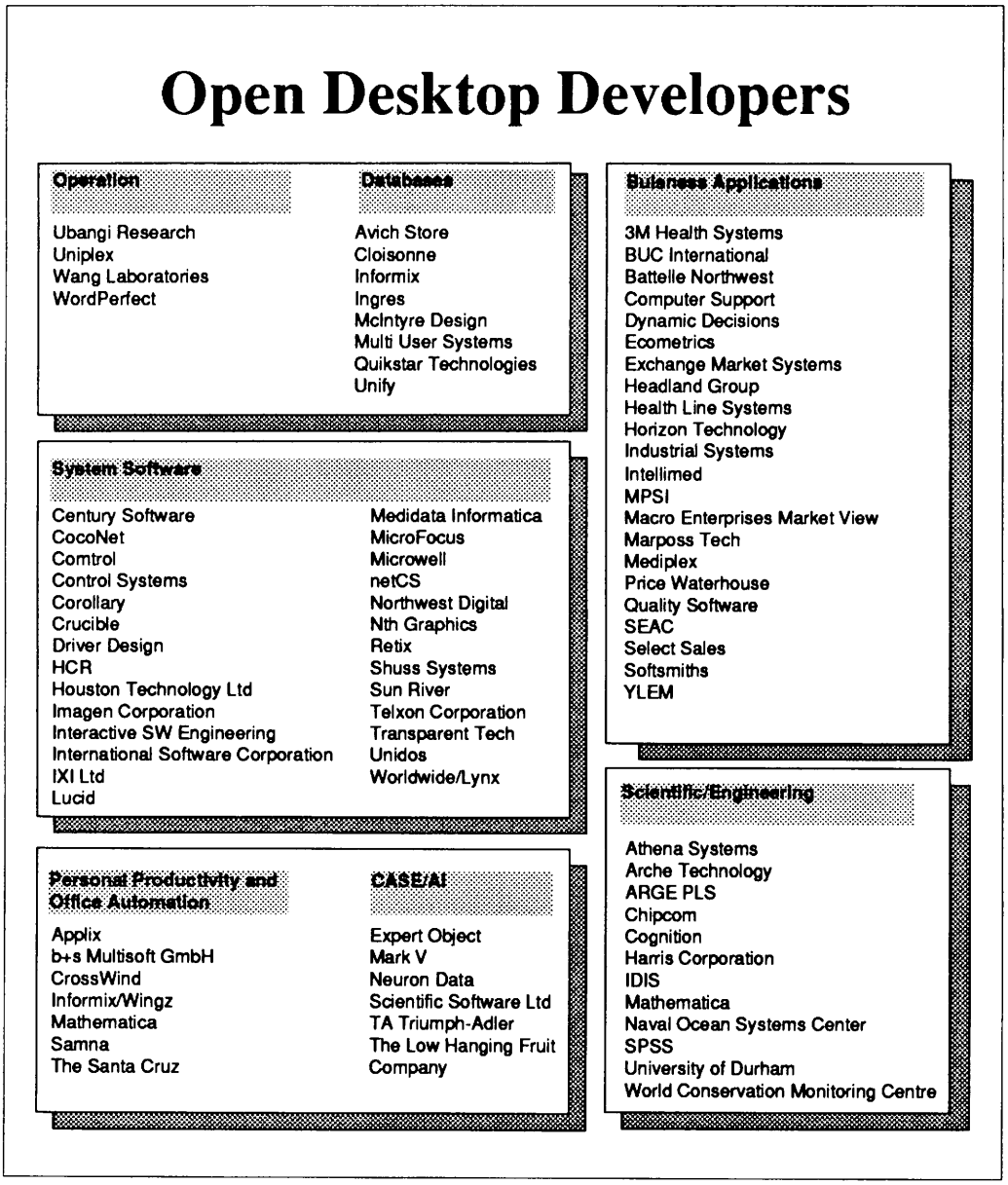


Illustration 6.

services operate in the same manner, SCO provides a single set of documentation, written in a single style, for all components of the environment. This can be a big advantage to users because they don't have to keep switching from one style of reference material to another. SCO states that its documentation brings "Unix to the user."

Interoperability. A big advantage to working within a predefined environment is the interoperability between applications. For example, Windows and PM have DDE links; the Mac environment has cut-and-paste buffers as well as MacroMaker, a low-end, cross-application macro facility; NewWave has, or will shortly have, agents and hot links between applications. Unfortunately, at this time, Open Desktop does not offer any additional functionality on this score other than the cut-and-paste capability found in X Window. This can be reasonably functional, though. Because both DOS and non-Open Desktop Unix applications are opened in X Windows, contents of these applications can be cut and pasted with each other and with Open Desktop applications.

However, we would like to see SCO focus on providing this sort of interoperability. It would allow the platform to truly add value, rather than simply being a target design center.

ISV Issues. For two years now, Open Desktop has been the talk of UniForum. In 1989, SCO joined with Digital Equipment Corporation, Locus Computing Corporation, Relational Technology Incorporated, and Tandy Corporation to announce the concept behind the product and the timetable for availability. Then, this January, the SCO booth was buzzing with activity as the product, complete with Beta versions of Open Desktop applications, was demonstrated. The number of applications actually running was disappointing. SCO had initially hoped to have about 30 applications to show off. In fact, 17 applications were presented, but only a handful were actual alive and working. Most were canned demos. We were assured by both SCO and ISVs that this was not due to problems with the developer's toolkit. SCO had underestimated the difficulty of creating a stable desktop Unix environment. As Doug Michels said, "It was hard!" And SCO was determined to do it right—all the pieces working together seamlessly and transparently—understanding that if it wasn't done well, no one would write to it. So delays resulted. The original timetable called for Open Desktop to be available to developers in April of 1989 and to customers in the third quarter of the same year. In actuality, developers didn't get the full toolkit until September '89 (though prerelease versions were available since April '89), and the retail version wasn't available until January of this year.

But now the news is much better. Many applications are currently in Beta with scheduled ship dates (SCO would not give us a specific—or even a ballpark—number), and, in general, ISVs are happy with the quality and stability of the Open Desktop toolkit.

OPEN DESKTOP TOOLKIT. SCO has not written new APIs to the standard services which make up Open Desktop. But the Open Desktop Toolkit does include its own documentation, which Allen Ginzburg, director of strategic marketing at SCO, states has "cleaned up" a lot of the instructions for those APIs.

In addition to the APIs for Motif, X lib, Ingres, TCP/IP, and NFS, Open Desktop provides Locus's PC Interface Library. This is a DOS library which generates DOS and OS/2 binaries and lets you write DOS programs that use both DOS and Unix facilities both locally and over a network.

ISV ENDORSEMENTS. We spoke to a number of ISVs, asking them why they chose to endorse the Open Desktop platform. All the developers we interviewed said the same things:

Grateful for a Target. The positioning of Open Desktop as a target platform for desktop Unix is supported by the ISV community. "This is the way Unix-based window applications will finally move through commercial distribution channels. Open Desktop is the next vital step on the road to shrink-wrapped Unix," states Larry Warnock, director of marketing at Uniplex. He continues, "An unbundled solution can be really complicated for the user. Open Desktop takes care of all those installation problems." Uniplex is in Beta testing with its Open Desktop version of the UnipleX Windows Office System.

Chris Knudsen, vice president of marketing at CrossWind Technologies, concurs, and adds, "Products like SCO's Open Desktop straddle the Unix and PC markets. The lines between the two platforms are becoming fuzzy." CrossWind is about to ship its Open Desktop version of the Synchronize multiuser time management tool.

The hardware vendors are equally pleased with a target platform. According to Kenneth Osowski, director of Unix systems product management at Wang, "Open Desktop brings standards to the market. And this is a big help, because picking a Unix software platform can be very confusing." Osowski sees PCs running Open Desktop as a major step in the success of the business market for Unix. Wang will support Open Desktop on its PC 300 series, where it can run as a single-user system or as a multiuser server. Open Desktop will also be supported on the Wang OpenServer line.

Riding SCO's Coattails. One major factor in the decision to endorse Open Desktop seems to come not from an industry need, but from a strategic partnering decision. Developers seem unanimous in their endorsement of SCO as the star to which they want to hitch their Unix wagons. Jim Morton, Applix's product manager of 386 Alis, is clearly a fan. "SCO has established itself as *the* 386 Unix supplier, wiping out all the competition. SCO has the clout to take Unix into the office for the average mortal person." The Open Desktop version of Alis was about to be shipped as we went to press.

Wang's Osowski says, "Look at SCO's track record. Its commitment [to desktop Unix] will be a big push for Unix business software."

"Realistically," Uniplex's Warnock points out, "SCO has a large industry presence and marketing channel. But it needs applications. And applications need distribution. It works out well for both sides."

And SCO has just bitten off an even bigger piece of the Unix operating system pie by acquiring HCR Corporation, a Toronto-based concern that is reported to be the leading Unix software company in Canada. SCO Canada, Inc., as the independent subsidiary will be called, will provide programming tools under Open Desktop, as well as supplying consulting and contractual services.

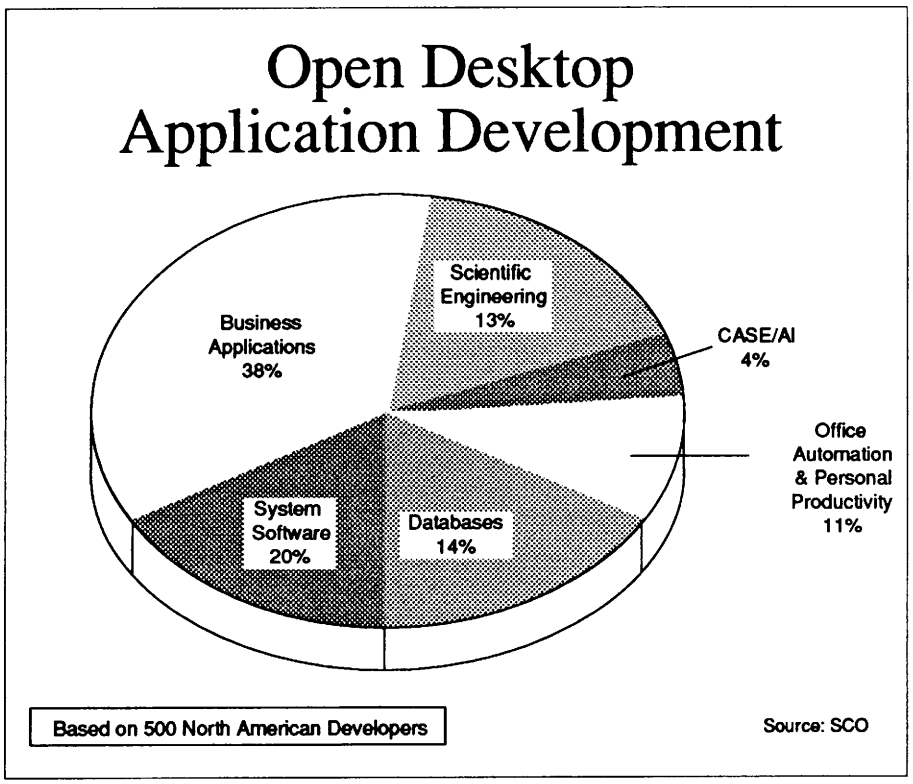


Illustration 6.

Open Desktop in the Real World

HAVING ANTICIPATED primary support from the ISV and reseller communities, SCO has been somewhat surprised by the enthusiastic response to Open Desktop from the corporate world. Though SCO would not provide a list of corporate customers, it did provide a few names and stories.

Harris Corporation

Harris Corporation, a worldwide electronics and communications company finds that Open Desktop addresses its three requirements for an advanced operating system platform:

- Flexibility and powerful tools required to keep ahead of the competition
- The ability to merge disparate operating systems, communications protocols, and hardware platforms
- The ability to downsize systems from mainframes and minis to networked workstations

The integration of de facto standards was also a big selling point for Harris.

Harris plans to use Open Desktop to develop internal mission-critical systems, though no specific applications were mentioned. Open Desktop will also be used for systems integration for Harris customers and for development of commercial software such as a network management system.

Eastman Christensen

Eastman Christensen is an oil-field service company which is planning to use Open Desktop on GRiDCASE 1530 Laptops to direct offshore drilling activities around the world. Real-time decisions will be made based on tracking the path taken to reach a target drill site. Open Desktop's graphical user interface will be used to depict the path of each well in three-dimensional geographical coordinates, and the relational database functionality will store the relevant drilling parameters for dynamic and static analysis. Eastman Christensen had considered developing its own system by linking a workstation computation engine, relational database, TCP/IP, and X Window environment, but the time and work required was daunting. Open Desktop, which combines all the required services, nicely fits the bill.

The Right Combination

In both cases, Open Desktop was chosen because it offered the right combination of standards at a low cost. Both companies plan to use Open Desktop as an internal development platform for mission-critical applications. And, in both cases, the companies would probably have put together a similar platform based on standards, but at considerable cost and effort. SCO made it much easier for them to get to the business of running the business.

Other Motivaters for Endorsement. There were a variety of other reasons given for Open Desktop development, including the following:

- Low-cost platform for X Window. Applix's Morton comments, "Open Desktop is a great low-cost platform for our product and for other X-based applications."
- Alternative to OS/2. "When SCO first announced Open Desktop, we saw it as a solution alternative to OS/2. Multitasking, but a lot more stable and mature than OS/2," comments Warnock from Uniplex. "Also, SCO is selling Open Desktop through the reseller channel where customers can get a turnkey solution. That's really what desktop users want."
- Alliance with Microsoft. Morton finds SCO's alliance with Microsoft (Microsoft owns 10 percent of SCO) attractive. "The migration from the DOS world will be easier as a result of the alliance between SCO and Microsoft."

ISV CONCERNS. While most of the ISVs are optimistic about Open Desktop's potential, a few concerns were expressed.

Unrealistic Expectations. The major concern voiced was the fear that users would have unrealistic expectations regarding platform architecture and performance. "While SCO Unix is very fast, X and Motif slow it down a lot," mentions one ISV. "Users have to understand that you can't use a really cheap configuration and run workstation-type applications."

Another stresses that the Open Desktop and SCO Unix product should "do well for where it is targeted—at the workstation product level. It is not meant to replace 386 Xenix running an eight-user, character-based accounting system."

Still another stated, "Customers and retailers need to understand that Open Desktop is a workstation product complete

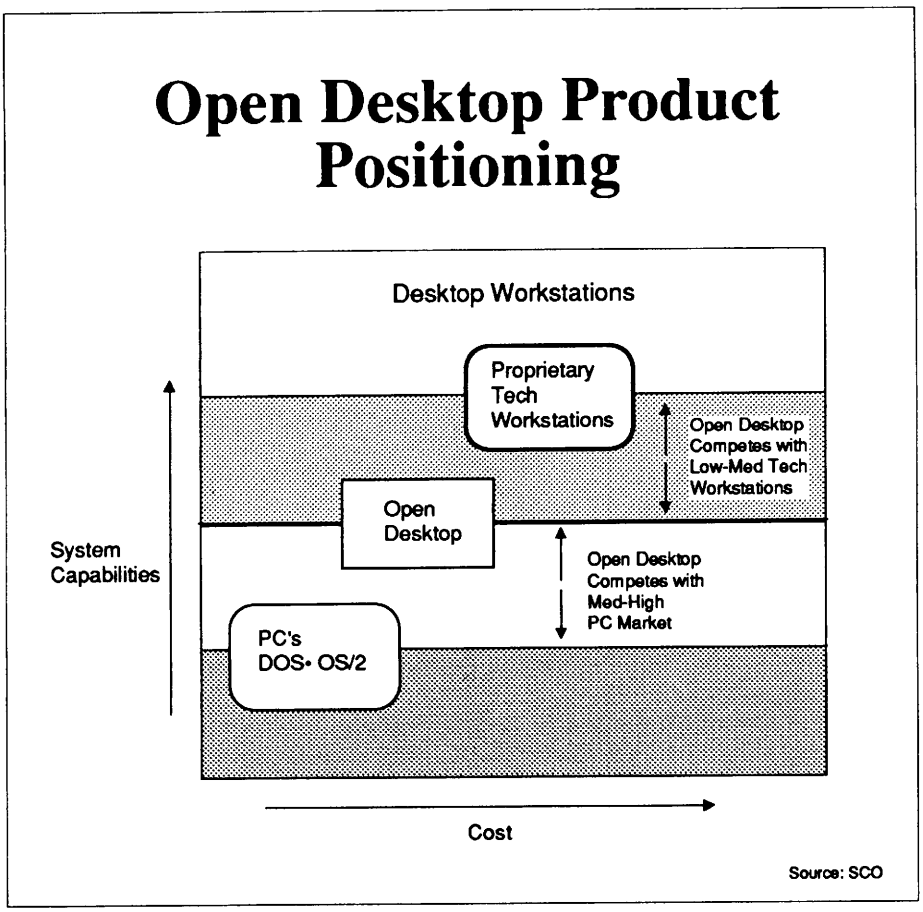


Illustration 7.

with networking. If people don't need workstation capabilities, then don't buy Open Desktop."

Most feel that the danger is customers getting attracted to the low price of a minimal configuration—386, 6MB, 100MB disk—without realizing that robust X-based applications will not perform well. In addition, there are problems of real estate on a screen. The marketing director of one software vendor comments, "X costs you two inches, one at the top of the screen and one at the bottom. So, to realistically run most Motif applications, you need to go for a larger, high-resolution monitor. Then you need to load your 386 with a whole lot of memory and processor speed—20 MHz is rock bottom. For four users, you have to plan on 33 MHz. Once you pay for all this, you've lost your price/performance advantage. You're in the same cost ballpark as a SPARCStation 1."

"People are going to be disappointed if they look for too much of a bargain," one ISV summed up.

Competition from RISC. Several comments were made about the competition from RISC platforms, which will continue to go down in cost, improving the price/performance ratio vis a vis the 386 and 486 platforms.

SCO, while not committing to any specific plans for migration to RISC, has stated that it is committed to the Intel product line and that it will most likely move in the direction of Intel's i860 RISC platform. However, SCO also said that it plans to stay in sync with the market. We anticipate that the company may have to move off an exclusive Intel platform and embrace the Motorola 88K or MIPS RISC platform if SCO does want to keep in line with the marketplace.

Slow ISV Development. A few of the ISVs we interviewed expressed concern at the slow development pace for third-party Open Desktop applications. "Unless there is a flurry of applications, Open Desktop will go nowhere," one vendor remarked.

But others were very optimistic for the same reason. "SCO has a vested interest in working with ISVs. No ISV support, no applications. So support from SCO has been excellent."

System Stability. We did hear one very negative reaction from an ISV who claimed that the Open Desktop software was very buggy. "Oh, sure, we're going to write for it, but, around here, we call the product 'Open Deathtrap!'"

Packaging. Finally, one ISV pointed out the problem in packaging a shrink-wrapped Open Desktop application. "It's not like DOS, where you take it out of the box and run it. You have to make an attractive and clear package that will move through the distribution channel. But how do you make clear the difference between a 16- and a 32-user system, for example? There are a lot more packaging issues that we need to resolve."

Marketing Strategy

SEEDING THE MARKET. Open Desktop is not really ready for prime-time retail sales to end users. SCO admits this. But the company felt strongly that a stable, usable version for developers needed to come out quickly to seed the market. Waiting too long could result in some competitor stealing the market or in the window of opportunity closing because no applications were ready. However, SCO did postpone release of the product to make sure it was, indeed stable, if not polished. "You can seed the market or you can poison it," said Doug Michels.

MARKET THRUST. SCO does not consider ISVs as a major revenue target for Open Desktop. Though early developers paid for the system, Michels acknowledges that this charge was

necessary to cover the high costs of supporting and updating early development systems, and, now that money is coming in from customers, the company can afford to be more generous with ISVs. "ISVs are part of the infrastructure of the market. We need them," commented Michels.

There are two initial marketing thrusts for Open Desktop: large corporate customers and the workstation market. Corporate customers are buying multiple copies of the development system and are designing custom applications. The end-user versions will go on the users' desks as soon as these in-house applications are ready.

The workstation user is accustomed to paying premium prices and is "ready-made for a lower cost platform," says Michels. And this user's expectations are not as high as those of PC or Macintosh users. Thus, workstations users won't be put off by the inconsistencies in the current version of Open Desktop. However, he does not view this as a long-term market. The DOS PC market has a lot more potential in size and revenue.

Retail Is the Goal. Ultimately, the target customer for Open Desktop is the retail shopper. But Michels doesn't anticipate much movement in this market until over a year out. "When applications are ready, the target machines are out, and the next level of polish and distribution support is there, then we can address the retail customer. The PC, Mac, OS/2 level of user."

Pricing. Whether or not you approve of the concept of Open Desktop, you have to like its price tag. For $995, you get almost $5,000 worth of software. The actual price breakdown of all the Open Desktop services looks like this:

- SCO Unix - $595
- Ingres version 6.1 - $1,495
- X.desktop, including X Windows and Motif - $395
- TCP/IP - $595
- NFS - $595
- LAN Manager - No price given
- DOS Merge, including DOS 3.3 - $495

Open Desktop is priced to sell. Ingres, reportedly, practically gave away the software to be included in SCO's product. Obviously, the vendors involved have great confidence in the future of Open Desktop and in its potential for opening new markets for them.

Futures

Though SCO has not yet announced what will go into the next release of Open Desktop, Mike Foster, Open Desktop marketing manager, states that the next version of the product will concentrate on three areas: user interface—specifically, enhancements to the help and system administration function, as well as complete Motif implementation; integration—better integration of the links between the components of Open Desktop; and performance optimization.

With the problems of inconsistency addressed, SCO needs to focus on issues of interaction and interoperability—some sort of data exchange mechanism such as Windows' and PM's DDE is necessary.

SCO needs to look at what value Open Desktop can add as an environment and to the applications written to the environment; cross-application macros are a good example. After all, the problem of adding value is usually limited because not all applications are written to the same platform and protocols. Open Desktop applications are, and that is the primary function of the product—being a common platform.

But users must realize that Open Desktop does not, at this point, add anything to the applications themselves. There is no interoperability above what comes standard in X Window; the behavior and look and feel of applications are no more consistent than any nominally written to Motif (full Motif implementations will have a higher degree of consistency). Rather, the platform provides all the underpinnings necessary for a multitasking, networked, graphical environment. Writing to Open Desktop eliminates many installation hardships for users and, even more significantly, establishes a de facto standard for a Unix desktop platform. ◐

NEWS ANALYSIS

PRODUCTS • TRENDS • ISSUES • ANALYSIS

• INSIDE •

• OSF •

DCE Selections

The Open Software Foundation (OSF) has released its selections for the first phase of a Distributed Computing Environment (DCE). The technologies:

- Network Computing System (NCS) 2.0 from HP/Apollo with Digital enhancements for the remote procedure call mechanism
- Andrew File System (AFS) 4.0 from Transarc for a distributed file system
- Support for diskless workstations, from Transarc
- Kerberos, with HP extensions for authentication
- LAN Manager for Unix (LM/X) from Microsoft and PC-NFS from Sun for PC integration
- DECdns from Digital and Siemens DIR-X for X.500 directory service
- CMA from Digital for threads
- Digital's DECdts for a time service

As with Motif, the OSF is putting some work into integrating the selected technologies. As a result, the technologies will be available as the OSF Remote Procedure Call, the OSF Distributed Naming Service, The OSF Distributed Time Service, the OSF Threads Service, the OSF Distributed File System, the OSF Distributed Security Service, and the OSF Personal Computer Integration Service.

The components selected and designed by the OSF will all work together as a *system*. For example, the Naming Service is implemented on top of the OSF RPC and is integrated with the Security Service. The Threads Service, which provides portable facilities supporting concurrent programming, is used by a number of OSF services: RPC, Security, Naming, Time, and Distributed File System.

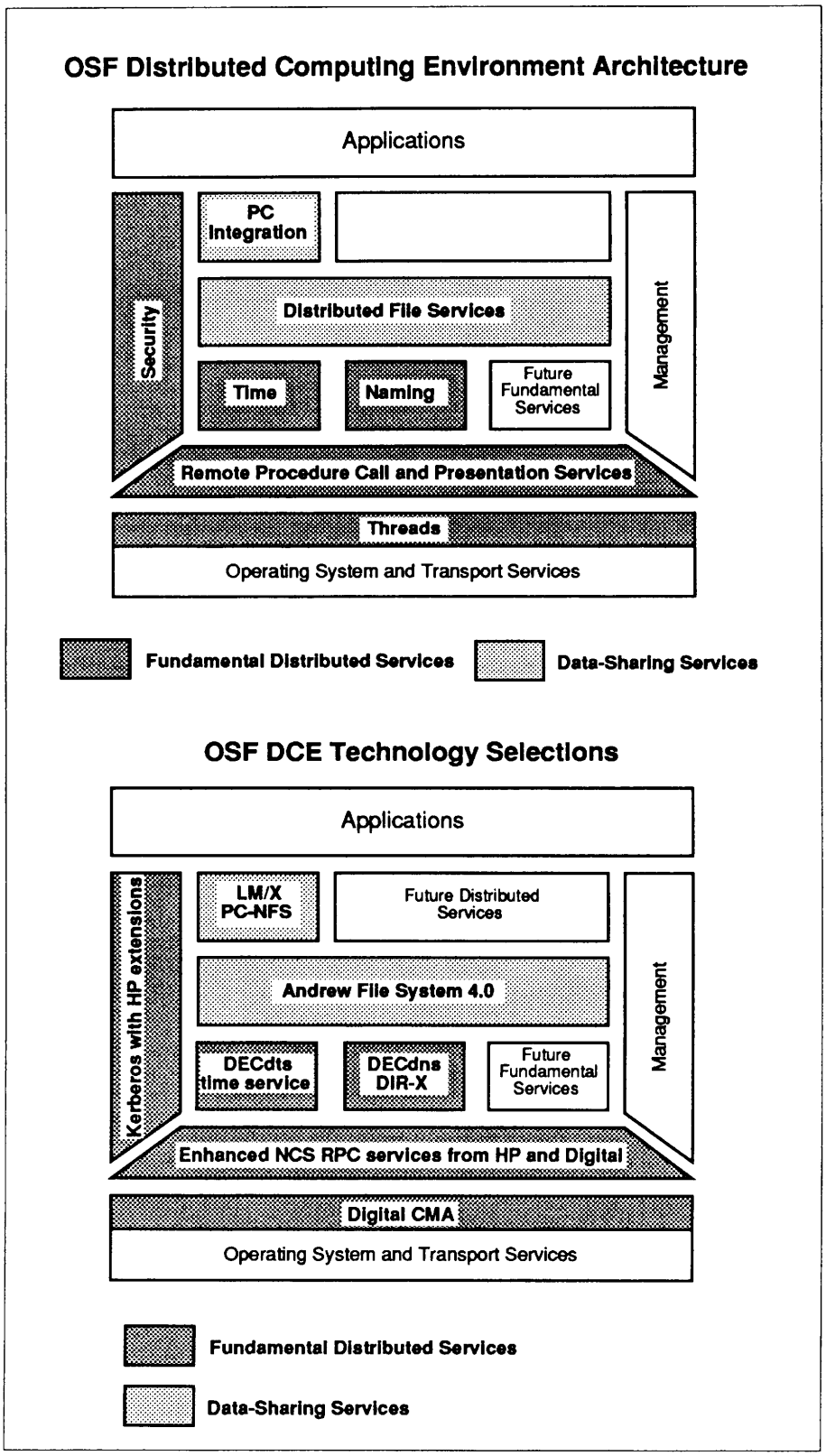
Within the architectural framework of a distributed computing environment, the selected technologies meet two primary needs: data sharing and fundamental distributed services (see illustrations on page xx).

The final selection does appear quite similar to the massive DeCorum submission (primarily from OSF members). That alone will be enough to raise innumerable objections from the opposition camp: Sun, AT&T, and Unix International.

Rumors have been bubbling up (through the conversational mud surrounding issues such as this selection) to the effect that the technical and the management camps within OSF were divided on the selection. (In other words, management allegedly strong-armed the technical selection.)

Not so, according to the OSF technical folks. The technical consultants apparently were in agreement on the technical quality and worth of the selected technologies. There was some disagreement on what that actually should mean in terms of implementation and rollout, however.

But the net of the selection is that OSF has a strong set of technologies that, in their very design, are intended to span multiple platforms. One of the critical selection points for the OSF DCE was platform and operating environment independence. As we've pointed out before, this isn't a Unix issue; the DCE has to transcend platforms. That desire for a platform-independent, portable solution is really a characteristic of many of the OSF companies. The IBMs, HPs, and Digitals all need solutions that they feel scale well within their current mixed-systems networks. The OSF DCE selection process wasn't about the best Unix network platform; it was about the best set of enablers for a mixed distributed net-



OSF's architectural structure for a Distributed Computing Environment consists of number of components. As seen in the diagram, the selected technologies slot into the architecture to provide a fairly comprehensive foundation for implementing this type of networked system.

work environment.

For its part, Sun plans to enhance its Open Network Computing (ONC) environment with the same set of services being offered by OSF, although at a slightly slower rate, since Sun intends to develop much of it itself.

Sun also agrees with the notion that a distributed computing environment is platform independent. However, the reality of Sun's base is that Unix dominates. Sun's work with ONC is clearly targeted to its perception of the needs of its customers.

So, apparently, there will be two contending DCE platforms: the one from Sun and the one specified by the OSF. The barriers between the two aren't absolute, but there is less compatibility than there should be in an ideal world. Some users won't be happy to hear about further disagreement. However, the OSF suggestion should certainly please the enterprise customers who now have the prospect of having advanced interoperability among the systems in their heterogeneous networks. And that interoperability was OSF's target in the first place.

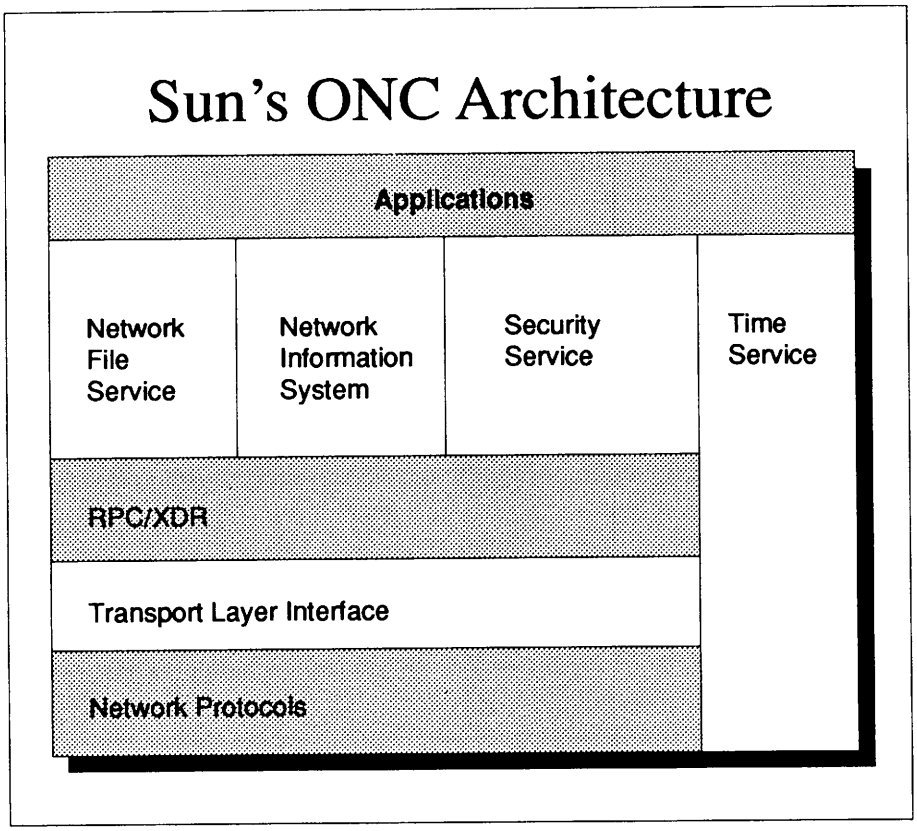
Furthermore, Sun's "I'll do it myself" attitude carries with it the burden inherent in any such situation: i.e., it takes longer to reinvent something on your own than to adopt a suite of technologies proffered as a standard. Unixcamp micturition contests aside, the vendors who now will adopt OSF DCE recognize that it would take them too long and cost them too much to develop the entire set of requisite functionality by themselves.

—M. Millikin

•RPC•

OSF Be Damned: Sun Runs Alone with ONC

There will be no truce in the RPC wars. Sun Microsystems is striking out on its



own to build a distributed computing framework. Sun will do so by extending and enhancing its Open Network Computing (ONC) basket of products. Sun has chosen not to partake of the Open Software Foundation's Distributed Computing Environment (DCE) initiative, but rather to compete with it.

ONC has at its heart Sun's Network File System (NFS) and the RPC Tool from Netwise Incorporated of Boulder, Colorado. Sun acknowledges that it needs much more than this—naming services, time synchronization services, and authentication services to start—and promises to build additional components into ONC.

Sun recently announced its "Road Map" for extending ONC. The plan is vintage Sun, stressing time to market and aggressive marketing. However, Sun also signaled that it will produce the ONC distributed environment using internal development projects, as opposed to working with partners. Licensing Netwise's RPC Tool, then, was an anomaly. This approach contrasts with the Open Software Foundation (OSF) strategy, which is built on technology partnering (see illustration above).

SUN'S RATIONALE. Corporate users have been clamoring for an end to the RPC wars. At the Patricia Seybold's Technology Forum in April, the call for a single standard was loud and clear. Yet Sun is portraying its ONC strategy as serving the needs of users who have already invested in NFS and ONC.

Sun has a responsibility to its installed base, and it is committed to carrying forward ONC. Company executives note that Sun has signed up many more users and can claim more applications than the Apollo division of Hewlett-Packard, its archrival, for its competing Network Computing System. How, they ask, could the OSF turn its back on such market support?

The answer to that question reveals the risks of Sun's approach. The industry's transition to distributed computing has just begun. Sun's foundation for this new era is NFS, a solution originally designed to support workgroups that is, for the most part, untested in larger settings. Sun is saying to its customers: Trust us to add the services you'll need in the future to what you have installed today.

The OSF and the participants in its Distributed Computing Environment want to make sure that customers have a solid architectural foundation as they start implementing distributed computing applications in a big way. OSF's platform describes services to support both localized and enterprise-scale applications. Interestingly, the components of Sun's architecture for distributed applications are the same as those identified by the OSF for its DCE.

Sun's marketing rationale is also disingenuous because it fails to consider migration as an option for users. For years, IBM, Digital Equipment Corporation, Hewlett-Packard, and others have been crafting migration paths to bridge customers' existing investments in technology and emerging technologies. Migration is a fact of life as technology changes. According to Sun, customers will migrate, too, but within ONC. Sun believes its customers are loyal and will follow its ONC path.

WHAT'S IN THE ONC PLAN. There's not much new in Sun's ONC Road Map. Sun is committing to deliver new time/synchronization services, new authentication services, and improvements to its Network Information Service (formerly called Yellow Pages) and NFS during the next two to three years (see table on page 20).

The Road Map doesn't provide very many new features. Sun had previously announced its intention to move to Netwise's RPC Tool, superseding its own RPC/GEN. Netwise's RPC Tool will compile into Sun's External Data Representation's (XDR's) canonical format and eventually give users the option of compiling to OSI's ASN.1 format. The Transport Layer Interface (TLI) incorporated in Unix System V Release 4 will provide the transport independence.

A Base Architecture. The Road Map is notable in setting out Sun's view of

what comprises a distributed computing environment. Sun's approach is more focused than OSF's. Then again, at some point, Sun will have to figure out how to add some or all of the services outlined in the OSF architecture.

The following are the other improvements Sun plans to make on the ONC we know of today:

RPC Improvements. Along with the RPC Tool, Sun will introduce two new RPC features to its users. First, it will supply an asynchronous call facility. This allows an application to continue working even as it waits for the results of an RPC to be returned. Previously, Sun's RPC "blocked" the flow of an application while an RPC was serviced. Asynchronous RPCs will first be used to take advantage of multiprocessing hardware, next in parallel processing.

Sun has also committed to implementing the threads model for RPCs next year. Currently, Sun offers only the lightweight process model. Again, threads are in line with the requirements of multi- and parallel-processing.

NFS Enhancements. Sun has committed to four improvements to NFS. First, Sun will use a combination of tuning and local disk-caching to increase performance. One of Sun's goals in improving NFS's performance is to allow more desktops per NFS server.

Second, Sun will make NSF networks easier to administer with new tools and with integration of existing tools into its SunNet Manager network management software. Third, Sun will add file replication as a way to provide read-only NFS operations, thus improving overall reliability.

Last, Sun will add the Kerberos authentication scheme as a security option for users. Actually, Kerberos fits into a separate group of authentication service enhancements in the Road Map.

Authentication Services. MIT's Kerberos, which is also part of OSF's DCE, is an interim solution within Sun's ONC. Sun will offer Kerberos as well as its existing Secure RPC as authentication services. Kerberos maintains a database of clients, each of

ONC Feature	2nd half '90	1st half '91	2nd half '91	1992 and beyond
RPC	• Transport Independence • Asynchronous call support • Netwise RPC		•Multithread support	
Naming		• Improved consistency • Improved security • Location - binding interface		•X.500 interoperability • Object-oriented approach
File System		• Local disk caching • Performance improvements	• Security improvements • Replicated file system support	• Admin. improvements
Security		• Kerberos Authentication		
Time Synchronization Protocols	•Network Time			

which has a private key known only to it. The Kerberos scheme also provides for message encryption. Secure RPC is an optional mode of Sun's RPC that provides much of the function of Kerberos. In the secure mode, the RPC enforces user name and password security in an encrypted format. Sun currently offers the federal Data Encryption Standard (DES) format.

Sun is offering Kerberos because Kerberos has emerged as a standard. The strategic direction for ONC, however, is to improve its Secure RPC in combination with security enhancements to its Network Information Service (NIS). Sun is particularly concerned about accommodating both private keys and the public keys of outside clients. In its ultimate authentication model, RPCs will retrieve keys from a secure naming service. Sun is working on this enhancement itself.

Naming Services. Sun's Network Information Service will be the basis for a distributed Naming Service in ONC. Sun is improving the consistency of its current product by adding a synchronization mechanism between directories and allowing multiple instances of a name. It is also adding a location and binding interface that will allow network objects to automatically register themselves with NIS and allow application clients to locate and bind to the network services.

Sun is also moving toward an object-oriented naming approach by adding security attributes to the information NIS stores about users and resources attached to the network.

Sun is positioning NIS as a superset of the X.500 standard for directory services, and it is committing to interoperate with X.500 directories.

Time/Synchronization Services. To synchronize the time on networks with world standard time, Sun is adopting the Internet's Network Time Protocol (NTP) as part of ONC. NTP provides an absolute time throughout the network and coordinates that time with the world time standard.

CONCLUSIONS. Despite our preference for an immediate end to the RPC war, we're encouraged that Sun's new ONC Road Map at least brings it into closer alignment with OSF. The Road Map identifies a subset of the services OSF included in its DCE. There's hope that, as time passes, Sun's architecture will evolve to be virtually indistinguishable from OSF's. Sun itself acknowledges that the OSF initiative caused it to whip its uncoordinated research projects to extend ONC into a structure and a set of priorities.

But, for the moment, Sun will fight OSF and the backers of its DCE. Given its position as the market leader, this is certainly understandable. There's no business imperative for Sun to join in an industry-wide effort right now.

Sun has always operated with a lot of self-confidence. The danger for Sun is letting its confidence become corporate hubris. The latest slogan invented by Sun CEO Scott McNealy is, "All of our wood behind one arrowhead," with the wood being the SPARC RISC chips, Unix System V.4, ONC, and OpenLook. Sun is developing three of these four technologies by itself, which is a big, big job even for a multibillion dollar company. We expect Sun to relax its defiant stance over time to grab the technologies it needs to satisfy its market imperatives. *—J. Rymer*

•OMG•

An Open Framework for Object Systems

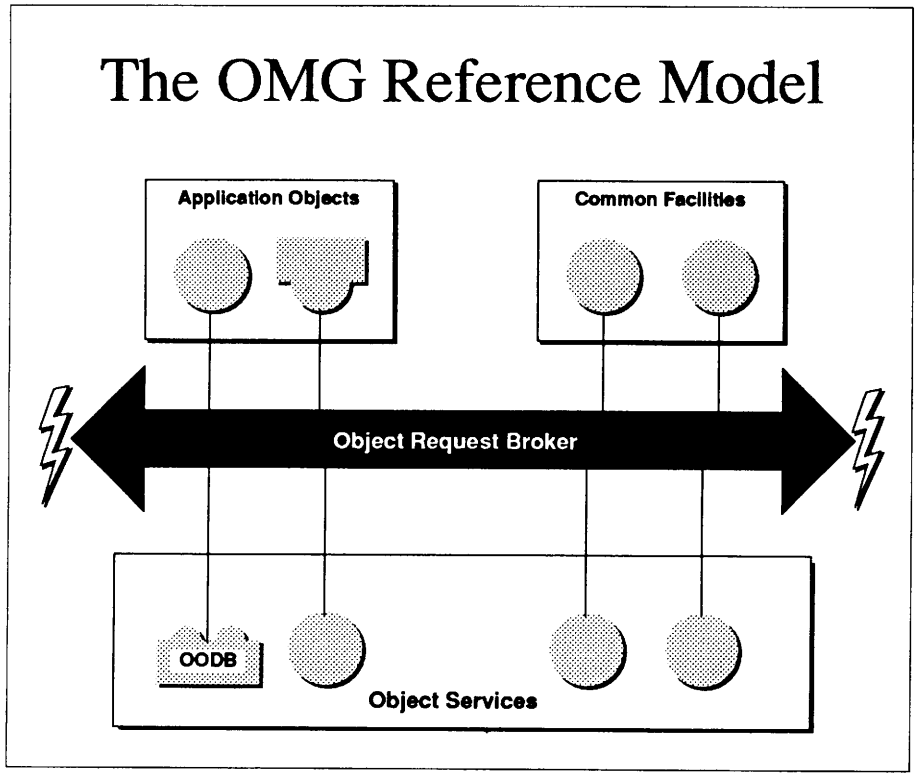
Everyone believes the object-oriented paradigm will be instrumental to the information systems of the '90s. But the paradigm has been dogged by doubts about just how it will be applied to large commercial systems. Progress in developing object technology has been slowed by questions about how differences between object models can be reconciled, how objects and services will be managed in distributed networks, and, indeed, what an object is.

Fresh off the presses, the Object Management Group's (OMG's) new Standards Manual attacks these problems by offering a common ground: A set of root concepts and a system architecture. The OMG's manual seeks to clarify how system developers can ensure that their object-oriented systems work together. The manual includes a "reference model" for object systems that defines the crucial interfaces for interoperability, as well as an abstract object model that represents the first agreement of its kind.

The OMG's reference model is a critical piece in the search for ways for object-oriented systems to work together despite diverse computer architectures, operating systems, and programming languages.

COMMON GROUND, DELICATE BALANCE. Since its founding a little more than a year ago, the Object Management Group has been called everything from the salvation of commercial object-oriented technology to a Hewlett-Packard booster club. During its first year, the organization has signed up an impressive roster of members from around the world and forged important links with standards bodies like X/Open and the European Community's Esprit initiative. With the publication of its Standards Manual, the OMG has proven it can forge consensus on how object technology should be guided during its formative stages to achieve interoperable products without stifling innovation.

OMG's approach to standards is minimalist and is geared to serve companies developing technology today. The challenge for the organization was to find the common conceptual ground for a variety of object technology implementations. Its models don't seek to define implementations, but, rather, the concepts and functions that are essential to interoperability. OMG uses existing technology as the basis for its standards recommendation. In this ap-



proach, it is quite different from the International Standards Organization (ISO), which defines standards based on what is ideally possible.

The approach is evident in the first version of the OMG Abstract Object Model. The model distinguishes between the semantics of an object and the implementation of those semantics in data and code. The OMG manual spends more detail on the semantics, leaving the implementation details open to each developer.

ABSTRACT OBJECT MODEL. The Abstract Object Model is a conceptual framework that will guide the OMG's design choices in proposing and adopting specific technologies. Its view of what objects are and how they interact encompasses both the pure object-message-object models used in today's languages and more generalized views of how objects operate.

In the OMG model, clients issue requests specifying an operation and parameters, any of which can identify an object, to the system. Methods as well as the operation may be selected by any of the objects identified in the request. This is a superset of the language model that has objects issuing messages to other objects to perform tasks. It is also a superset of the remote procedure call (RPC) model. The OMG model is descended from the model used by the Common Lisp Object System (CLOS).

The model's fundamental concept is the separation of the semantics of objects and the implementation of those semantics in data and executable code.

Object Semantics. The object semantics defined by the OMG model include the following items:

- OMG objects. An OMG object is a set of operations and their associated states that are characterized by a behavior triggered in response to a request by a client. An object is created by explicit action and has a distinct identity. Objects are identified by their names and handles. A *handle* is a name that refers specifically to a particular object; a *name* is a more general identifier. For example, a name might refer to the nearest available printer, while a handle would refer to a specific printer in a particular location. An object may have more than one handle in a single or in multiple contexts. The model allows a third kind of identity—abstract identity—for certain special tasks.

- Literals. The OMG model does not require all entities that can be referenced to be objects. A *literal* is a named entity that is not an object—such as an integer.

- Requests. In the OMG model, clients issue requests for services to be provided by one or more objects. ("Provided," in this sense, implies a wide degree of participation in the provision of a service.) A *request* comprises an operation and zero or more parameters, or values. The parameters name which objects will provide the requested service.

- Operations. An *operation* names the service to be provided to the client. Each operation is created by an explicit action and is distinct from other operations. Like objects, operations can have identity. Each operation has a signature that may restrict the meaningful parameters in requests naming it. Signatures are specified when an operation is created. The signature concept is similar to the strong typing imposed by some programming languages.

- Behavior. A *behavior* is generally some sort of computation, which may produce a final result or create another request.

- Type. In the OMG model, *type* corresponds to the notion of object classes used by some languages—except that it is a higher level of abstraction. A type is a boolean function used in a signature to restrict a parameter or characterize a result. The extension of a type is the set of values that satisfy the type. An object type is a type with an extension that is a set of values that identify objects.

- Interface. An *interface* describes the possible uses of an object—the potential requests in which the object

can participate. An interface type is a type satisfied by any object that satisfies a particular interface. Each object has a principal interface.

The OMG's model allows a single value to have multiple types, and classes are but one of the inheritance structures allowed by this model. The OMG model recognizes inheritance whenever one type conforms to another—any value that satisfies the first type also satisfies the second.

Object Implementation Concepts. Object Implementation concepts apply to implementation of objects—including methods, data structures, classes, and implementation inheritance.

- Methods. A *method* is the code that executes to satisfy a requested service. In executing, the code accesses stored data. These data represent the state of the system, and performing the service may change this state. The code itself is called a method.

- Binding. The OMG model allows the method and data to be bound both before the request for service is issued (static binding) and after the request is issued (dynamic binding).

- Object Implementation. The OMG model assumes the presence of mechanisms to define the definitions of data structures, methods, and how the system infrastructure, as defined by the OMG Reference Model, will select methods and data to execute in response to a service request. The model allows a single method to be executed for different objects.

- Inheritance. The OMG model allows different objects with the same behavior to share their implementations. These include: support for classes, the ability of a single method to be executed for different objects, incremental refinements of existing objects in new implementations, and delegation.

- Persistence. The OMG model defines persistent objects as those that are preserved in storage after the process they are involved in ends. Transient objects last only as long as the processes in which they participate.

- Data Models. The OMG confines its discussion of data models to desired features, including state integrity and consistency, referential integrity, behavior consistency, and atomicity.

What's Excluded from the Model. The OMG plans to invest its time in defining objects from the perspective of applications and databases, not from that of language. Thus, the OMG's object model doesn't mediate the competition between the models espoused by competing object-oriented programming languages such as SmallTalk and C++. The model is general enough to accommodate different programming language models.

The OMG model also leaves out some familiar concepts, specifically compound objects, attributes and links, copying of objects, change management, transactions, and a control and execution mechanism. It leaves these to specific application architectures.

REFERENCE MODEL. The OMG Reference Model is a "central design core" that ensures interoperability, portability, and extensibility in heterogeneous environments. The Reference Model defines the components and interfaces of the Object Management Architecture (OMA), which describes an infrastructure that mediates between clients issuing requests and services. The primary purpose of this infrastructure is to ensure that a system selects the appropriate method to perform a requested service, and provides access to the appropriate data in its execution.

The Reference Model leaves open many implementation possibilities. It does specifically describe how objects make and receive requests and responses, identifies which operations must be provided for every object, and defines object interfaces providing common facilities.

Object Management Architecture. The OMA defines an interface, a standard way for clients to issue requests to conforming objects and to receive responses. The architecture does not provide detailed interface and protocol specifications. These will most likely be provided later through the OMG's technology sponsorship process. The illustration on page 22 identifies the four major components of the OMA:

- Object Request Broker (ORB). The Object Request Broker allows objects to interact using the OMG's request-result concept. The ORB arranges for requests to be processed. Components of the ORB include name services, a request dispatch function, parameter-encoding facilities, a delivery mechanism, synchronization facilities, activation/deactivation facilities, exception-handling, and security mechanisms.

- Object Services (OS). Object Services provide the functions for realizing and maintaining objects. The services may be accessed via class interfaces or not, although the OMA Reference Model doesn't define non-class interfaces. Object Services may include class and instance management services, storage facilities, access control, integrity mechanisms, query facilities, and versioning.

- Common Facilities (CF). Common Facilities, accessible through class interfaces, provide functions useful across a variety of applications. Common Facilities could include object and class cataloging/browsing, agent facilities, printing/spooling, help, electronic mail, user interfaces, interfaces to external systems, and user preferences and profiles.

- Application Objects (AO). The Application Objects layer is the collection of classes specific to user appli-

cations. These may work in concert with Common Facilities; indeed, Application Objects may migrate to become Common Facilities.

A key characteristic of the OMA is accommodation of applications and services that are not implemented using the object paradigm. As long as these resources are retrofitted with an OMA class interface—using "wrapper" or "adapter" code—they can participate in an OMA system.

CONCLUSIONS. The OMG's Standards Manual is not complete. But, as an organic document, it may never be truly complete. The priority for future versions of the manual is the completion and approval of the Technical Requirements chapter, which, at press time, was expected by early summer. We can expect other additions as the group pursues its mission.

We've read through the first version of the OMG Standards Manual, and the first question that arises is: Where are the standards? There are, after all, no protocols or interfaces defined in the document. Those kinds of detailed standards will come later. For now, the industry's need is for a common systems context for implementing the object paradigm. And that is what the OMG has provided. *—J. Rymer*

•INGRES•

A Step toward Presentation Independence

With the introduction of Ingres/Windows 4GL, Ingres Corporation is tackling some tough issues in the development of database applications. The product also forms the foundation for the company's tools products for the '90s. Ingres/Windows 4GL will become part of the suite of tools for generating applications for the Ingres relational database management system (RDBMS). A workstation product, Windows 4GL provides a graphical development environment designed specifically for generating Ingres DBMS applications that run on workstations with a graphical user interface (GUI).

BUT DOES IT DO WINDOWS? The move to client/server architectures, the increasing use of intelligent front-end workstations (from the lowly but ubiquitous PC up through high-powered technical workstations), and the growing popularity of GUIs are all making life more difficult for the applications developer while easing the burden on the end user. Designing an application for a workstation that takes full advantage of the native window manager means that the developer must now understand the windowing system and its associated toolkit and style guide, and must write the DBMS application to the window system's programming interface. Otherwise, the developer is faced with the alternatives of merely moving a character-cell interface to the workstation, or of giving the workstation user access to only a small subset of the available interface features.

Ingres/Windows 4GL has solved this problem by abstracting the window user interface, providing 4GL access to user interface objects, and implementing visual interface editors.

The 4GL. The Ingres 4GL has been expanded to provide access to and control over all capabilities of the graphical user interface. These include the graphical user interface elements (e.g., entry fields, button fields, radio buttons, option lists, check boxes, list boxes), the use of multiple concurrent windows within an application, and integration with other applications/windows on the desktop (e.g., sharing data between windows).

Visual Interface Editors. The frame and menu editors provide a GUI for designing the application's user interface. The developer can interactively paint windows from a palette of standard

Availability of Ingres/Windows 4GL

	CPU	Window System
Initial platforms (third quarter 1990)	Sun Sparc Digital VAX/VMS version 5.3	OSF/Motif (running in the Open Windows environment on the X11/NeWS server) DECwindows
Future platforms (Sun's OpenLook will be first, followed by the others in 1991)	Sun Sparc DECstation/Ultrix HP 9000/300 SCO Open Desktop IBM RS/6000 DG Aviion IBM PC Apple Macintosh	OpenLook OSF/Motif OSF/Motif OSF/Motif OSF/Motif OSF/Motif Presentation Manager (OS/2) MS Windows (DOS) Mac Tool Kit

Ingres/Windows 4GL will be priced at 35 percent of the Ingres base price, which varies depending on the size and configuration of the user installation. As an example, on a 2- to 8-node Sun4 network, Ingres/Windows 4GL will cost approximately $1,000 per seat.

"widgets" (or interface elements) and design complex pull-down/pull-across menus without writing any code. A 4GL script editor is used to tie application code to an entire window or to encapsulate the application code with individual window elements.

Object Oriented. Ingres continues to incorporate object-oriented features within the Ingres RDBMS. With Windows 4GL, object classes may be defined and shared among applications, and 4GL code encapsulated with window fields and menus to reduce coding.

WHAT HAPPENS TO PORTABILITY? The existence of *multiple* GUIs in the industry compounds the interface problem for developers whose products and applications must run across multiple platforms. One of the primary selling points of the successful RDBMS is applications portability across dissimilar platforms. In the past, the major concern was developing an application that could run unmodified on different CPUs and/or operating systems. Now another area of dissimilarity has developed—the native windowing manager. Is it Motif, DECwindows, Presentation Manager, MS Windows, or OpenLook? Each of these has its own programming interface and "look and feel."

Ingres/Windows 4GL has been designed to address this broader development issue of *presentation independence*. An Ingres DBMS application built with Windows 4GL in the DECwindows environment, for example, can be deployed on OSF/Motif or any other supported windowing system without change. The ported application not only takes on the "look and feel" of the native interface, but also gains interoperability with other window-based applications on the desktop (e.g., cut and paste). Windows 4GL achieves this at two levels. First, it provides an abstraction of the window user interface, shielding the developer from the gory details of each interface. Second, it links the run-time version of Windows 4GL directly with the native window toolkit libraries on each platform, rather than with the lower-level graphical interface itself. Many of the native interoperability features are provided at the window toolkit level.

It is important to note that the issue of presentation independence is industry-wide, and not just confined to DBMS applications. Hewlett-Packard, for example, is looking at a layer of software to provide presentation independence for developers writing to the NewWave platform.

Style Guide Issues. One area where Windows 4GL cannot provide transparent portability is in accommodating style guide differences (for example, the use of "File View Edit" in an OpenLook main menu versus "File Edit View" in OSF/Motif). Since Windows 4GL separates the definition of 4GL events from window and menu descriptions, the developer can define different windows and menus for use on different systems without recoding any of the application logic.

APPLICATION MANAGEMENT. Windows 4GL also provides application management benefits. It supports multiple developers working on the same application with automated locking and version control of application element definitions. All application definitions/objects are catalogued in the Ingres Open Data Dictionary and stored within the DBMS. This allows common application elements to be shared among multiple applications.

WHAT'S MISSING. As always, there are some limitations. In Ingres/Windows 4GL, we note the following:

- The developer can only take advantage of window facilities that are common to all underlying GUIs. When asked if Windows 4GL offers a "lowest common denominator" solution, Ingres preferred to describe it as a "medium common denominator." If a particular toolkit doesn't support a particular function (for example, a sliding bar), Ingres can use lower level functions to build up the desired capability.

- The developer cannot deploy Ingres/Windows 4GL applications on character terminals. Although deployment on terminals may be included in a later release of the product, it isn't clear that doing so is a priority for Ingres. Thus, Windows 4GL is targeted only for applications that will run on workstations. This will be a major drawback for many organizations that have a mix of terminals and workstations and need an evolutionary migration path from terminal-based to client/server applications. For these customers, Ingres continues to recommend the use of its existing applications development products, such as Applications-By-Forms.

- Deployment is limited to those platforms/GUIs for which Ingres/Windows 4GL is available (see chart).

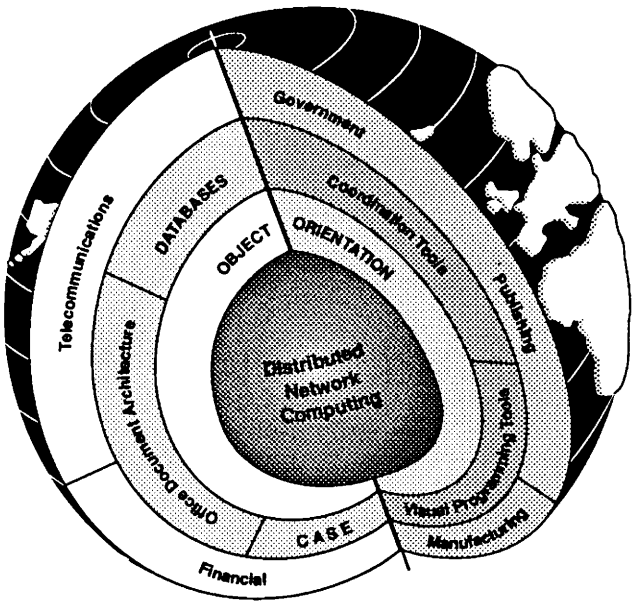
SUMMARY. With Windows 4GL, Ingres is offering an easy-to-use graphical development environment for developing applications *on* a workstation, a tool for developing GUI applications *for* a workstation, and a tool for developing applications that are presentation independent. We expect to see more of the DBMS vendors providing this type of software for presentation independence, and we are not surprised by the fact that Ingres got there first.

—J. Davis

Patricia Seybold's

Technology Forum '90/Europe

THE WORLD OF OBJECT-ORIENTED APPLICATIONS



October 18-19, 1990

Rai Congrescentrum,
Amsterdam,
The Netherlands

Registration Fee: $750.00

Sponsored by:
Patricia Seybold's Office Computing Group
and
The Object Management Group

WHY THIS CONFERENCE?

BUSINESS COMPETITION AND SURVIVAL

To survive in a business climate that is becoming more global in nature, users and vendors are turning to a new generation of information technology solutions.

If they are to compete effectively, users must enhance their access to global information as well as their ability to share this information. There is similar pressure on the vendors to provide these solutions and to help accelerate the industry out of its current stagnation. Information that is flexible and can reach across disparate systems is the key.

WHY OBJECTS ARE ESSENTIAL

Object-oriented technology will be fundamental to business success in the coming years. Object-oriented interfaces can free users to concentrate on their business tasks, rather than on working with software. Object-based applications give users the power to collect, analyze, and act on huge amounts of complex information through the creation of compound documents. Object-oriented programming and software design promise great increases in developer productivity and in the reduction of the application backlog.

UNIQUE INTERNATIONAL TECHNOLOGY SEMINAR

General office users are struggling to understand object technology and put it to work in a new generation of applications. "The World of Object-Oriented Applications" is the only international conference to address this sea of change in information technology and its implications for users and vendors.

FOR MORE INFORMATION:

- Call 617-742-5200
- Fax 617-742-1028
- Or write to:
  Patricia Seybold's
  Office Computing Group
  148 State Street, Suite 520
  Boston, MA 02109

P R O G R A M

DAY 1: OCTOBER 18, 1990

OPEN SYSTEMS IN THE '90S: BUSINESS RATIONALE AND OVERVIEW

Primary Purpose To establish the market imperative for the adoption of object-oriented technology.

Secondary Purpose To establish the important role of standardized interfaces as the foundation for Open Systems, rather than standard implementations.

OBJECT-ORIENTATION: DEFINITIONS, MODELS, AND IMPLEMENTATIONS

Primary Purpose To draw distinctions among the many applications of object-oriented terminology.

Secondary Purpose To discuss the current models in the market with an eye to enabling solutions and to providing business benefit.

BUSINESS VIEWS ON OBJECT-ORIENTED ENVIRONMENTS

Primary Purpose To give three perspectives — a user's, a developer's, and an executive's — on the benefits and pitfalls of object-oriented environments.

OBJECT-ORIENTED DATABASES

Primary Purpose To highlight the important role of a database in distributed object management and to explore issues in networking and distribution.

Secondary Purpose To explore the various approaches to object-oriented databases: reworking a relational DBMS with an object layer or Object SQL, or designing and implementing a highly granular object-oriented databases.

COMPOUND DOCUMENTS AND OBJECT-BASED PUBLISHING

Primary Purpose To highlight the important role of the compound document in future systems and to explore the intersection of compound documents and object technology.

Secondary Purpose To discuss the issues emerging in compound document interchange—and, by extension, object interchange and interoperability.

DAY 2: OCTOBER 19, 1990

BUSINESS BENEFITS OF WORKFLOW AUTOMATION

Primary Purpose To define workflow automation and to explore its business impact. To examine the potential for rethinking business strategy and organizational structure in terms of the technology.

Secondary Purpose To explore the role of object technology in enabling workflow automation.

OBJECTS AND EMERGING APPLICATION AREAS: HYPERTEXT, MULTIMEDIA, AND GROUPWARE

Primary Purpose To discuss the business importance of these new application areas, with particular emphasis on organizational performance and function.

Secondary Purpose To explore the relationship of objects and object technology to important emerging application areas.

STRATEGIC SOLUTIONS USING OBJECT TECHNOLOGY

Primary Purpose To explore object-based strategic solutions in five vertical application areas: Publishing, Government, Financial Services, Telecommunications, Manufacturing.

ENABLING THE DISTRIBUTED ENTERPRISE NETWORK

Primary Purpose To highlight the interrelation of distributed network computing and object orientation in delivering an advanced solution.

THE REVOLUTIONARY '90S

Primary Purpose To wrap up the conference with a call to action/implementation.

There will be numerous exhibits offering demonstrations of the latest in object technology. The exhibit hall will be open throughout both days of the conference.

Patricia Seybold's Computer Industry Reports

ORDER FORM

Please start my subscription to:

		U.S.A.	Canada	Foreign
☐ *Patricia Seybold's Office Computing Report*	12 issues per year	$385	$397	$409
☐ *Patricia Seybold's Unix in the Office*	12 issues per year	$495	$507	$519
☐ *Patricia Seybold's Network Monitor*	12 issues per year	$495	$507	$519
☐ *Paradigm Shift—Patricia Seybold's Guide to the Information Revolution*	12 issues & tapes per year	$295	$307	$319
☐ *Paradigm Shift—Patricia Seybold's Guide to the Information Revolution*	12 issues per year	$195	$207	$219

Please send me a sample of: ☐ *Network Monitor* ☐ *Office Computing Report* ☐ *Unix in the Office* ☐ *Paradigm Shift—Patricia Seybold's Guide to the Information Revolution*

Please send me information on: ☐ Consulting ☐ Special Reports ☐ Conferences

☐ **My check for $_____ is enclosed.** ☐ **Please bill me.** ☐ **Please charge my subscription to: Mastercard/Visa/American Express** (circle one)

Name: _____ Title: _____

Company Name: _____ Dept.: _____

Address: _____

City, State, Zip code: _____

Country: _____ Bus. Tel. No.: _____

Card #: _____

Exp. Date: _____

Signature: _____

Checks from Canada and elsewhere outside the United States should be made payable in U.S. dollars. You may transfer funds directly to our bank: Shawmut Bank of Boston, State Street Branch, Boston, MA 02109, into the account of Patricia Seybold's Office Computing Group, account number 20-093-118-6. Please be sure to identify the name of the subscriber and nature of the order if funds are transferred bank-to-bank.

IU-0690

Send to: Patricia Seybold's Office Computing Group: 148 State Street, Boston MA 02109; FAX: 1-617-742-1028; MCI Mail: PSOCG
To order by phone: call (617) 742-5200

Topics covered in Patricia Seybold's Computer Industry Reports in 1989/1990:

Back issues are available, call (617) 742-5200 for more information.

Office Computing Report

1989—Volume 12

#	Date	Title
9	Sept.	Tools for ESS Development—The Vendor Offerings
10	Oct.	OfficeVision—IBM Prepares for the Next Generation
11	Nov.	Cease Fire!—PC Wars Confuse Customers Needlessly
12	Dec.	IBM's Data Management Design—The Future Is Distributed and Relational

1990—Volume 13

#	Date	Title
1	Jan.	Office Redefined—A New Model for a New Decade
2	Feb.	Visual Programming—Is a Picture Worth 1,000 Lines of Code?
3	Mar.	Word Processing in the '90s—How MS Word for Windows Stacks Up
4	Apr.	Multimedia—The Promise, the Players, the Issues
5	May	Is X the Window? X Window and distributed Applications

UNIX in the Office

1989—Volume 4

#	Date	Title
11	Nov.	To Lead or Follow?—Which Route Have Uniplex, Applix, and Quadratron Chosen?
12	Dec.	Targon Office—Nixdorf's Object-oriented Platform for Application Integration

1990—Volume 5

#	Date	Title
1	Jan.	Oracle's RDBMS—The Product behind the Marketing
2	Feb.	Multiprocessing—Towards Transparent Distributed Computing and Parallel Processing
3	Mar.	IBM's Second Generation RISC—With its new RISC System/6000, IBM claims center stage.
4	Apr.	Hewlett-Packard's NewWave Office—Distributed Object Management in an Open Environment
5	May	Hewlett-Packard's NewWave Office—Distributed Object Management in an Open Environment

Network Monitor

1989—Volume 4

#	Date	Title
9	Sept.	AT&T's Accumaster Integrator—OSI Network Management Leader
10	Oct.	Digital's EMA—A Fresh Perspective on Managing Multivendor, Distributed Systems
11	Nov.	A New Calling—IBM's Strategy to Integrate Voice and Data
12	Dec.	TCP/IP Network Management—What It Was, What It Is, What It Will Be

1990—Volume 5

#	Date	Title
1	Jan.	Distributed Computing Environment—Interoperability Now!
2	Feb.	LAN Manager 2.0—Jumping Up to the Major Leagues
3	Mar.	A New View of OpenView—HP's Network Management Strategy
4	Apr.	A Study in Contrasts—Unisys's Open Networking Strategy
5	May	Xerox Panos—Delivering on Distributed Networking