Patricia Seybold Group

Editor-in-Chief
Michael A. Goulde

INSIDE

Formerly Unix in the Office

OPEN INFORMATION SYSTEMS

Guide to Unix and Other Open Systems

Vol. 7, No. 6 • ISSN: 1058-4161 • June 1992

Digital's DECworld Gems

Alpha and Accessworks Shine

By Michael A. Goulde

IN BRIEF: Preproduction prototypes of systems built around Digital's Alpha RISC technology were the star of the show at DECworld this year. However, the interoperability star was a product that makes use of Digital's NAS services and database management tools to provide access from mixed desktop environments to heterogeneous data sources. Appropriately named Accessworks, this combination of hardware and software combines client and server functions in one box to help address the daunting challenge of supporting multiple communication and data access protocols so that desktop clients and remote hosts don't have to be burdened with that overhead. While Accessworks doesn't introduce any new technology, it does represent a pioneering effort to provide users with a pre-configured, pre-tested, integration platform that can be used to build effective client/server applications.

Report begins on page 3.

Welcome to Open Information Systems

Evolution, Not Revolution

AS YOU PROBABLY noticed when you first picked up this month's newsletter, we have changed its name from *Unix in the Office* to *Open Information Systems*. We took this action after evaluating the responses we received to the reader survey that was sent out in March.

The vast majority of subscribers who responded to our survey indicated that they do not feel that the content of this publication should be limited either to "Unix" or to "the Office." They indicated that a name change would be consistent with the content of the publication as it has evolved over the past few years. As a result, we made the decision to rename the newsletter to reflect its editorial direction more accurately.

Unix in The Office began its life in February 1986 with a feature article that reviewed three Unix word processors. As early as May of 1986, *Unix in The Office* featured an article about a DBMS, Unify, reflecting the fact that the scope of office applications implied more than word processing.

Just as the application scope has broadened over the years, so has the scope of platforms broadened. Unix used to be synonymous with open systems. However, with the advent of technology-independent system interface standards, that is no longer the case. It is important now to focus on the key benefits users seek from open systems—interoperability, transparent access to data, manageability, investment protection, etc.—independent of the technologies required to achieve those benefits.

The mission of *Open Information Systems* will continue to be to provide in-depth information and analysis of commercial open systems. The distinction between commercial and technical/scientific systems is justified because of the unique characteristics and requirements of each. Commercial applications tend to be more focused on the meaning of data—its information value—than on the data per se. Account balances, flight reservations, orders, inventory, and other data elements represent knowledge about business conditions that must be captured, stored, processed, and shared. The information and the knowledge it imparts is the focus, not the data itself.

The systems requirements for commercial and technical applications overlap in some areas and are different in many others. For example, floating point performance isn't important when reserving an aisle seat, but it is very important in designing the seat. On the other hand, two phase commit is not very important in 3D modeling, but it is extremely important in ordering the real thing from inventory.

The mission of *Open Information Systems* will continue to be to bring its readers up-to-date information and considered analysis about the trends, products, technologies, and standards that are enabling the migration from closed, proprietary information systems to distributed open systems that are based on formal and de facto standards. We will cover key issues facing management, like migration strategies, systems management, and client/server development and continue to provide in-depth coverage of the most critical and interesting topics.

Issues faced in the information management industry change over time. What is not now will be passé in a few years. As the issues evolve, so does the focus of this newsletter. That has always been true and, in a sense, nothing has changed but our name.

OPEN INFORMATION SYSTEMS

Editor-in-Chief
Michael A. Goulde

MCI:
MGoulde
Internet:
mgoulde@mcimail.com

Publisher
PATRICIA B. SEYBOLD

Analysts and Editors
JUDITH R. DAVIS
ROSEMARY B. FOY
DAVID S. MARSHAK
RONNI T. MARSHAK
JOHN R. RYMER
ANDREW D. WOLFE, JR.

News Editor
DAVID S. MARSHAK

Art Director
LAURINDA P. O'CONNOR

Sales Director
PHYLLIS GUILIANO

Circulation Manager
DEBORAH A. HAY

Customer Service Manager
DONALD K. BAILLARGEON

Patricia Seybold Group
148 State Street, 7th Floor,
Boston, Massachusetts 02109

Telephone: (617) 742-5200
Fax: (617) 742-1028
MCI: PSOCG
Internet: psocg@mcimail.com
TELEX: 6503122583

Open Information Systems (ISSN 0890-4685 is published monthly for $495 (US), $507 (Canada), and $519 (Foreign) per year by Patricia Seybold Group, 148 State Street, 7th Floor, Boston, MA 02109. Second-class postage permit at Boston, MA and additional mailing offices.

POSTMASTER: Send address changes to *Open Information Systems*, 148 State Street, 7th Floor, Boston, MA 02109.

Digital's DECworld Gems

Alpha and Accessworks Shine

Will DECworld Calm the Turmoil?

Considering the bad press that Digital has been receiving over the past six months, one might have expected Boston's World Trade Center, the site of DECworld 1992, to have been filled with tension, chaos, and confusion. Instead, visitors were pleasantly surprised by a host of interesting vertical applications, forward-looking new technologies, and opportunities for hands-on with new products that Digital hopes will carry it back to better times.

Demonstrations of Alpha Systems

Visitors had their first glimpse of production prototypes of four new systems based on Digital's Alpha RISC technology. Desktop and deskside workstations were shown along with departmental and enterprise servers. The thrust of the Alpha display was to demonstrate OSF/1 and VMS running on Alpha processors and to show that ISVs are well along the way in porting their applications to Alpha. The ISVs at the show indicated that the task of porting their software to Alpha has been straightforward and that they anticipated no major delays in having their applications ready when Alpha is delivered to customers. However, Digital currently has only a minority of its software partners committed, so the Alpha exhibit was aimed as much at them as it was to customers.

In the Alpha demonstrations, Digital went so far as to actually show comparative performance of an Alpha deskside workstation against an HP 9000 Model 750, an IBM RS/6000 POWERstation 550, and a Sun Sparcstation 2. As expected, the Alpha system easily outperformed the other systems running a fractal drawing program. What was remarkable about this demonstration was not the performance, but that at this early point in development Digital was willing to run preproduction Alpha systems and preproduction operating systems side-by-side against the competition. Either Digital is incredibly confident about the progress of Alpha development, or the company figures it has nothing to lose at this point by making the comparisons. In either case, off the exhibit floor, a number of ISVs readily admitted that Alpha performance is definitely impressive.

Doughnuts to Dollars

In other demonstration areas, Digital used real-world simulations to highlight applications from Digital as well as from third parties. Ross Systems' manufacturing software (See *Unix in the Office, May 1992)* was featured running a doughnut factory that produced real (not great, but real) doughnuts for visitors. Other industry areas were arranged to be equally lifelike, adding to the relevance of the demonstrations, and, hopefully, from Digital's perspective, to the ability of Digital's salespeople to close business.

New Products and Technology Galore

In addition to the Alpha systems and many recently announced products, Digital was demonstrating a variety of technology and product concepts, and soliciting feedback from visitors on requirements and design details. There were Intel-based super servers with massive RAID configurations, smaller servers with attractively painted cabinets and solid mahogany doors, and personal computers with extremely avante garde enclosure designs. However, lurking among all the flashy hardware products was an equally impressive software solution that seemed to hold particular interest for visitors looking for answers to open systems

challenges. It was Accessworks, an integration and management solution for heterogeneous client workstations retrieving information from multiple databases.

Accessworks: Dealing with Client/Server Complexity

While standards-based methods for remotely accessing SQL data are evolving within organizations like ANSI and the SQL Access Group (SAG), users are faced with immediate needs and are looking for products today that can bring data from multiple data sources, including non-SQL data, to any client application. Although many point-to-point solutions are available on the market, all suffer a common shortcoming—as the mix of different clients that simultaneously access different servers becomes more complex, the complexity of managing multiple operating systems, network protocols, and access methods increases geometrically. For example, providing a network of PCs and Macintoshes with access to DB2 and Oracle data requires four different access protocols: PC to DB2, PC to Oracle, Macintosh to DB2, and Macintosh to Oracle. Adding one more client to the picture, a Unix workstation, for example, now requires six different access methods, adding Unix to DB2 and Unix to Oracle to the others. In addition, each client may have to support multiple transport protocols, depending on the platform on which the server is running. For example, a PC may have to run both TCP/IP and SNA protocols concurrently. The complexity of this situation is shown in Illustration 1. In the real world, the cost and difficulty of managing heterogeneous networks of this sort is enough to scare away all but the most daring users.

Compounding the management problem is that, with the addition of each new client protocol, the database server is given the incremental load of additional protocol processing and the load of handling a different access method. If one adds SQL dialect conversion to the puzzle, then the challenge of dealing with inconsistent functionality across different database back ends makes the goal of heterogeneous database integration almost unreachable.

NAS to the Rescue

Digital has entered this breach with a packaged solution based on the concept of adding an integration server as a middle, third tier between clients and servers, running a preconfigured collection of Digital's Network Application Support (NAS) and database products. The combination of hardware and software that constitutes Digital's branded middleware is called Accessworks, and it reflects an entirely new definition of three-tier architecture. In the past, three-tier implied a hierarchical architecture. In Accessworks, all three tiers are operating in peer-to-peer fashion. The only role distinctions are among servers and clients.

Accessworks Brings Data to the Desktop

The Accessworks Program and Strategy

Digital's strategy is to use "an expandable, manageable, coherent, and simple architecture for multivendor systems integration based on open standards." The goal is to "provide easy, transparent, global access of enterprise-wide data in a client/server, networked computing environment to a variety of client platforms" and to offer product implementations of this strategy today.

The origins of Accessworks are an interesting combination of product planning and sytems integration. Many of the components of Accessworks were being assembled as a part of a project to build a custom database integration server for the University of California in the fall of 1991. The University asked the Digital project people to explore whether their requirements could be met by existing Digital products so they could have product-level support for this server instead of having it maintained as a custom project. When they discovered that plans were in the works for an integration server, the University of California group collaborated with the product group on the product. Recognizing a market opportunity for what had initially been thought a specialized system integration solution, Digital

brought the rest of the components into the product program and announced Accessworks in March 1992.

Adding a Third Tier for Integration

Accessworks is a product, but it is best understood conceptually by considering it as a framework for enabling client/server applications under the NAS architecture. Digital approached the problem described above by interposing a middleware server between the client workstations and the database servers, creating a third, peer-level tier in the conventional two-tiered client/server architecture. The server acts as both an intermediate client for remote databases and a server for client workstations.

Framework for Client/Server Applications

Providing Access to Remote Hosts

The Accessworks framework provides LAN support on the client side and LAN/WAN connectivity to the remote servers. Using only one transport and one API, clients communicate with the Accessworks site server as though it were the actual provider of the requested data. In actuality, the data may be coming from any of a number of back-end servers, but the Accessworks server resolves all the protocol, syntax, dialect, and format conversions for clients before passing the request for data on to the remote server. In effect, the

The Heterogeneous Data Access Dilemma

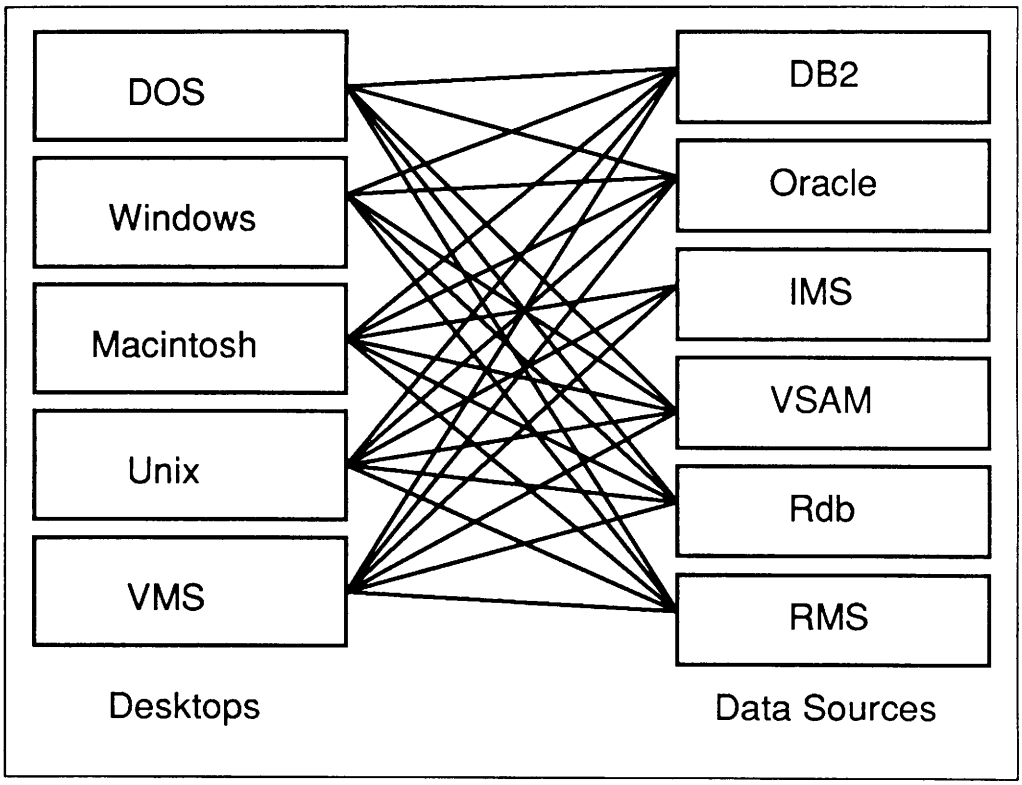

Illustration 1. This illustrates the dilemma created by heterogeneous desktops accessing heterogeneous data sources. Each client has to run different protocols for each server being accessed, and each host has to handle queries, protocols, and conversions for large numbers of clients.

Accessworks site server acts as a proxy for the desktop client application to retrieve requested data, communicating with a single protocol to that host on behalf of all clients, regardless of the protocol the client uses to connect to the Accessworks server. This relieves

the host system and application from performing any conversions. The manner in which this simplifies data access is shown in Illustration 2.

Site Server Handles the Translations

From the client perspective, an application makes a call for data to any of the supported APIs. This request is handled by the site server, which makes all the translations and protocol conversions necessary to pass an appropriately structured request to the remote host. The remote host then passes the data back to the server, which, in turn, passes it back to the client. It is possible to configure a single client logon for the site server that gives access to all the remote servers. In some cases, however, separate logons may be desired, and that can be done as well.

Server and Client Functions Performed

The server functions both as a client of the remote databases and as a server to desktop clients. It provides a combination of gateway services, protocol conversion services, and intermediate datastore functions that simplify the work of both desktop clients and back-end servers. Digital's Rdb relational database is included in the product to act as an intermediate data store. SQL Services, included in the VAX Rdb/VMS Runtime environment, performs SQL access functions.

Implement the Framework

The Accessworks framework is currently implemented as a family of products that integrate hardware, software, and network support to link multivendor client and database server platforms. The foundation for Accessworks products is a combination of Digital's VAX computers and Network Application Support (NAS) packages. NAS provides services for application access, communication and control, and information and resource sharing.

NAS Service Packages

Accessworks is more than NAS, but NAS provides the foundation for connectivity. NAS is available in three levels, called NAS 200, 300, and 400. (See *Unix in the Office*, July, 1991). NAS 200 provides basic workstation-server connectivity, NAS 300 adds support for client/server application development, and NAS 400 adds system tuning and data integrity features to NAS 200 and NAS 300.

Desktop workstation clients can use any application that supports one of the data access programming interfaces supported by NAS. Clients run a transport for application access as well as a LAN transport. Client applications are unaware of where their data is coming from, in terms of both the physical location of the data and the specific database in which it resides. Those issues are all resolved by the site server.

Site Server Architecture

The Interface to Clients

The software architecture of the site server is shown in Illustration 3. Each layer of the architecture is defined by standard or published interfaces, and each layer's service may be implemented by a variety of different products. Beginning at the layer that interfaces with clients, the Site LAN Transport may be one or more transports, depending on the mix of clients being supported. The LAN Transport may be any combination of TCP/IP, DECnet, AppleTalk, NETBEUI, or IPX, depending on what is necessary to communicate with the clients. The network transport is generally Ethernet, although the VAX 4000-based servers can be attached to Token Ring because their Q bus supports that topology. The LAN transport receives packets from the client and passes the data on to the Site Transport Server layer. Digital's SQL Services provides the Site Transport Server services, receiving incoming requests and performing the necessary transformations to prepare transactions for the Site Transaction Manager. For applications that are transaction-based, the Site Transaction Manager provides control services either through Application Control and Management System (ACMS), DECMessageQ, or Reliable Transaction Router (RTR) to ensure data in-

tegrity. The Site Format and Protocol (FAP) service then translates these requests into the proper format for the site server. By July, the FAP services will include a FAP client that is compliant with the FAP defined by the SAG.

Simplified Data Access

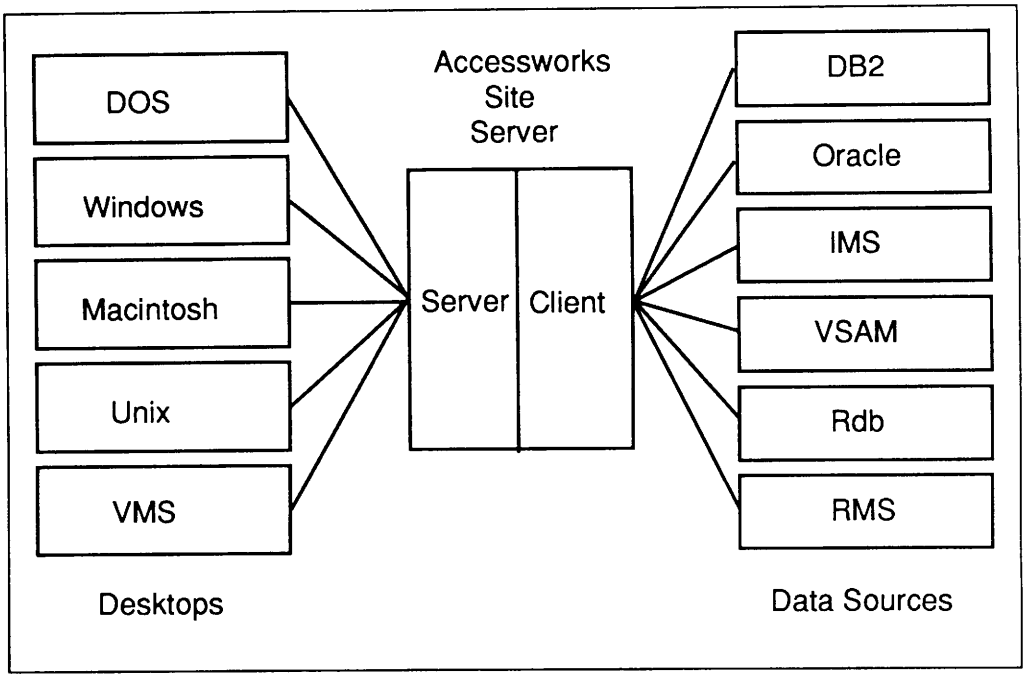

Illustration 2. This shows simplified data access through the use of a three-tier architecture. Desktops and remote hosts work within one environment while the Accessworks server handles all the complexity.

Active Server Functionality

The site server provides data management and application-level services that extend beyond those ordinarily provided by a gateway. The site server engine can be any application that has an API and can make its services available to clients on the network. Depending on the applications being supported, the site server can be any combination of Rdb, Application Control Architecture (ACA) services, Digital's SQL Services, CDD/Plus or CDD/Repository, a storage server, or a workflow server. The site server is under programmatic control and therefore can carry out a wide variety of activities on behalf of the clients. For example, the site server could be programmed to periodically update an extract of a database and respond to queries from clients without necessarily having to access the host database.

Connectivity and Interoperability with Remote Host

Actual routing of queries and data is handled in the Site Remote Router layer through services provided by Rdb Dispatch. These transactions are then sent to the Site Remote Format and Protocol service, which prepares queries in the proper format to be sent to the remote host. These services are provided by the client side of a format and protocol service that is either tailored for a specific database or is compliant with SAG standards for FAP. For example, the client side may be provided by VIDA/Client for IBM databases or by the FAP client that will be available in July 1992. The Site Remote Format and Protocol service handles the conversion of formats and protocols to and from the host format to a format the site server can handle.

Multiple Transport Support

The Site Remote Client layer, which prepares data to be passed to the remote host, may be provided by LU 6.2 for SNA hosts or by SQL•Net for Oracle databases. Finally, the transport for the site server's communications with the remote host is provided by the Site Re-

mote Transport layer. The services of this layer may be supplied either through one of a number of IBM Gateways or by a TCP/IP service. Accessworks/SNA-ST (Synchronous Transport) provides network-to-network communications for medium traffic loads over a DECnet network to one or more IBM SNA hosts. For heavier loads, Accessworks/SNA-CT (Channel Transport) operates as part of both the DECnet and SNA networks, providing a high-capacity communications capability. Unlike other Accessworks components, however, SNA-CT runs on dedicated server hardware.

Accessworks Site Server

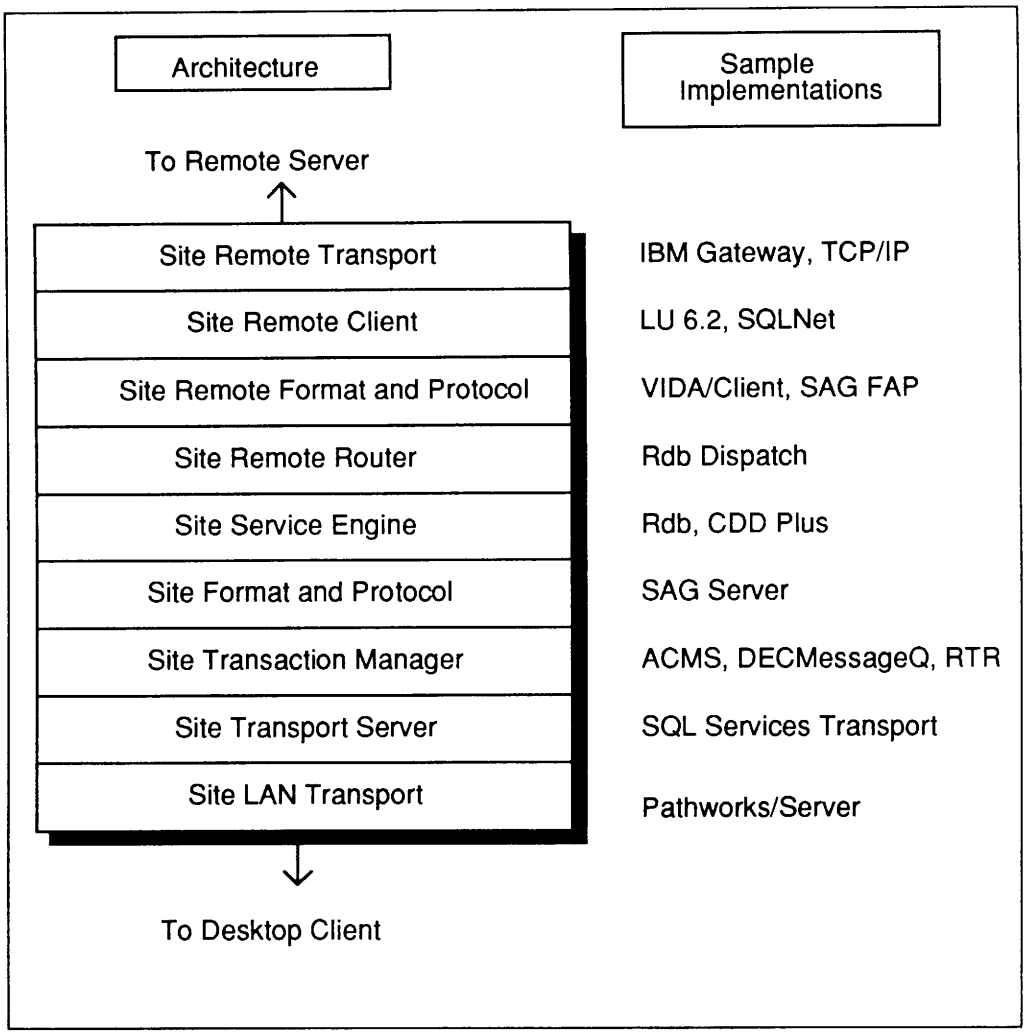

Illustration 3. In the Accessworks server architecture, the server is actually a client of the remote data source and a server to the desktop client application.

Site Server Configured according to Application Requirements

APPLICATION SEGMENTS. Accessworks site servers are currently available in seven VAX configurations sized to meet the needs of four application segments: Team Computing, Workgroup Computing, Organizational Computing, and Enterprise Computing. These segments have been defined primarily on the basis of the number of users who need to be supported and the load they could be anticipated to place on a server. Team and Workgroup segments are assumed to require the support of a single site server for a relatively small number of clients. The definitions of Organizational and Enterprise Computing are somewhat loose, but they roughly correspond to departmental and company-wide applications. Enterprise servers are distinguished by being offered with high-availability platforms. Or-

ganizational and Enterprise segments may require the services of multiple Accessworks servers and are configured with the higher-level NAS packages.

VAX PROVIDES THE FOUNDATION. The hardware component of each Accessworks package is configured to match the number of clients that are expected to be supported. The specific hardware platforms offered with Accessworks are scaled from within Digital's offerings to meet the different requirements of each of the four segments. The specific products for each segment will change over time as new models with new processors are introduced. Digital's market segmentation is less important in selecting the right platform than the performance characterizations it has done on each server to estimate capacity under actual workloads. Although Accessworks site servers are currently all VAX computers, it is reasonable to expect that Digital will move the full set of services to Alpha-based systems when they are available. This will extend the upper range of performance, allowing much heavier workloads to be supported for approximately the same cost per user.

NAS AND PATHWORKS PROVIDE SERVICES. One of the three different NAS packages is included with each server. NAS 200 is installed on the Team Computing server. NAS 300 is supplied for Workgroup and Organizational Accessworks servers, and NAS 400 is provided on Enterprise Accessworks configurations. All servers have Digital's Pathworks server software installed to provide connectivity for PCs. Depending on the type of client support and remote access required, a variety of optional services are available in the form of layered products that are preinstalled and preconfigured.

Client-Side Support

Clients using the services of an Accessworks server actually require no special Accessworks software. DOS, Windows, Macintosh, Unix, OS/2, and VMS clients are currently supported by Accessworks. DOS, Windows, OS/2, and Macintosh client workstations must have the appropriate Pathworks client software installed on them, whereas Unix clients need only TCP/IP. VAXstation clients use DECnet as their client software. Applications used on the clients can be virtually any package that provides one of the supported APIs. For example, the client desktop may use Paradox, SQL Link, and Pathworks to get data from a DB2 database either through the Accessworks server or data that has been extracted to the server. Other suitable applications might be TeamLinks, Lotus 1-2-3, or Excel. Illustration 4 shows how the client software fits the Accessworks architecture, even though no specific Accessworks software is required.

Support for NetWare clients is planned as a future enhancement that will allow them to access the server without running Pathworks. When that support is available toward the end of the year, the site server will look like a NetWare server to NetWare clients. NT client support will be available as part of the Pathworks for NT offering, probably in the first half of 1993.

Remote System Configuration

Minimal software is required to be run on the remote system or host. The remote architecture is shown in Illustration 5. The Remote Transport would generally be IBM's VTAM in SNA networks. The Remote Translation Server could be Digital's VIDA for DB2 or the Data Transfer Facility for both MVS and VM VSAM file access. The Remote Transaction Monitor could be CICS or IMS/DC. The Remote Format and Protocol server is provided in the architecture to accommodate future SAG servers and includes Remote Database Access (RDA) support.

Site Server Architecture

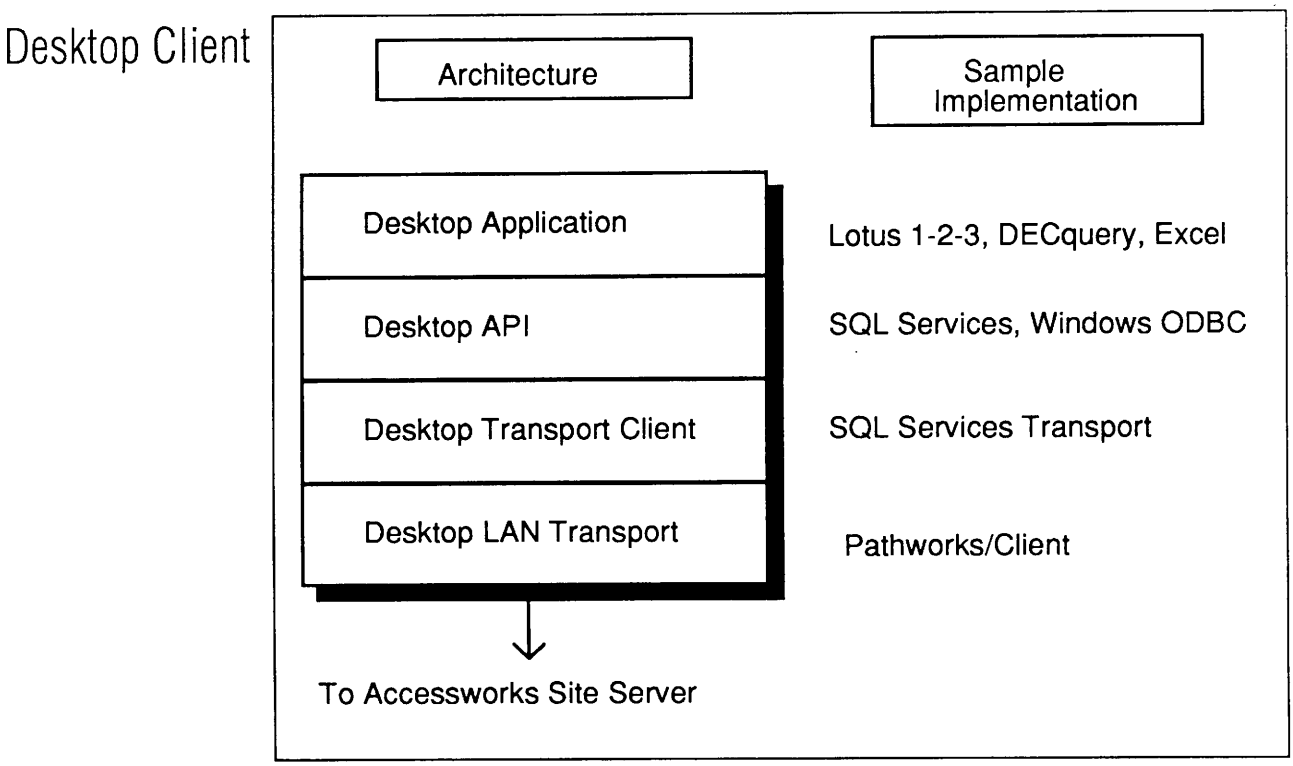

Illustration 4. This illustrates architecture and sample applications implementing the architecture on the desktop client.

Accessworks/Data Store

Since Accessworks has a local database, it offers some interesting application possibilities. The Rdb component of Accessworks creates a local image of one or more remote databases. In decision support applications, this could allow queries to be run locally against an extract of multiple remote databases, which could shorten response time and limit the load placed on the server database. This would be particularly helpful when complex joins over the network are required or when such joins would overwhelm a PC. A local view of the data from heterogeneous databases can be created by the database manager, giving users easy access to data from multiple sources.

Database functions at the site server are currently provided by Digital's Rdb. Accessworks' Data Store includes VAX Data Distributor and Rdb Interactive. However, there is nothing in the Accessworks server that limits it to using Rdb. In fact, Digital is currently in discussions with other database vendors about providing their database engines in the site server. By doing this, Digital would not only gain additional credibility for Accessworks as an open platform but also provide users with functionality in the server that Rdb lacks.

Rich Assortment of Data Sources

In addition to Digital's own Rdb and RMS data sources, Accessworks applications can reach data stored on IBM mainframes in DB2 databases and IMS and VSAM files. The current release also supports Oracle data, and support for other databases will roll out over time. We suspect that support will probably begin with Sybase, followed by Ingres and then Informix in short order.

Although access to DB2 and Oracle is currently read-only, an enhancement is planned that will allow Accessworks applications to write to DB2 and Oracle databases.

Communications Protocol Support

All the communications protocols supported by NAS are available in Accessworks. The core services available with VMS are DECnet/OSI and TCP/IP. Unix systems are supported with NFS and RPC support to promote resource sharing, networking, file access, and applications development. SNA networks can be reached with gateways. AppleTalk networks are supported through Pathworks, as are NETBEUI transports.

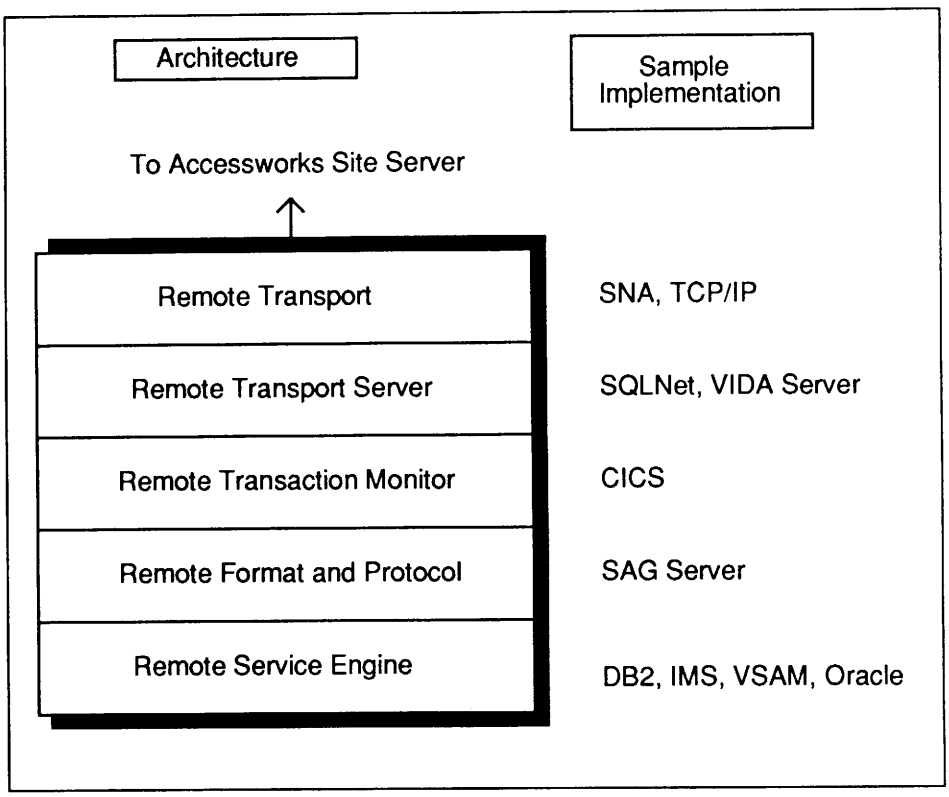

Illustration 5. This illustrates the remote data server architecture and examples of product implementations. There is no Accessworks-specific software on the remote host. Desktop client and site client are transparent to the Remote Service Engine.

Key Packaging Goals

It can be argued that, while Accessworks contains no unique technology or products, it is certainly unique in concept. In addition to supplying a preconfigured layered software environment that comes virtually ready to use, it addresses a whole host of issues when complex software environments are considered. Although middleware is rapidly becoming a key architectural component for most vendors, Digital, in a rare marketing coup, has already gone out and created a brand-name middleware product. In addition to branding, Accessworks packaging meets objectives in several key areas: version control, characterization, new technology enhancements, performance optimization, integrated server management, controls and security, and CD-ROM distribution.

Version Control

Ensuring compatibility across versions of so many different products is difficult because an upgrade to any one piece can easy break compatibility with other components. Having a

Key Packaging Goals

clearly defined architecture and staying within standard APIs is nice conceptually, but in the real world where actual code has to run, problems appear all too often. By packaging all the pieces together, Digital can incorporate each of the upgrades to individual pieces as a tested upgrade to the whole package. In fact, one of the benefits of Accessworks is that, although it began its life as a customized system integration effort, it now has product-level support and quality assurance. As a result, users should find the kind of reliability and integrity they are demanding in their applications.

Characterization

This is attractive to users who are extremely concerned about capacity planning for client/server applications. When elements of an application come from multiple sources, it is hard to estimate what kind of server configuration will be required to support various combinations of middleware. Accessworks controls the middleware variable and allows various workload mixes to be varied so that configuration requirements can be determined.

New Technology Enhancements

A layered architecture combined with a packaged product allows Digital to introduce new technology into Accessworks more easily and sooner than if the various parts of the solution were less predictable. It also makes it easier to support new platforms, both on the client side and the remote server side, and to support new services within Accessworks servers themselves.

Performance Optimizations

Predictable configurations on known hardware platforms allow Digital to tune and optimize for maximum performance in the environment in which Accessworks servers function. Communications hardware and software can be tuned for client/server activity, and memory and storage configurations can be optimized for database activity.

Integrated Server Management

Accessworks provides a common management interface for all its components. This addresses users' concerns in integrating their own solutions with multiple, inconsistent management environments. Ultimately, the OSF's Distributed Management Environment (DME) will address this, but, for now, Digital's approach simplifies matters for site administrators.

Controls and Security

Consistent control over access, logons, privileges, and accounts is easier with an integrated package, as is interfacing this function to server management facilities.

CD-ROM Distribution

Digital can distribute one-shot software upgrades to all the server components on CD-ROM, simplifying installation, licensing, and distribution.

Fine-Grain Capacity Planning

Capacity planning for client/server applications is a major challenge for users. There are no easy yardsticks, and no one has enough experience with client/server installations yet to be able to set down any rules of thumb. Digital is in the process of characterizing various Accessworks servers under typical workloads in order to gauge their capacity under standard workload conditions. Using three types of workloads with varying degrees of complexity in the queries, it is possible to vary the front and back ends independently and gauge capacity requirements and characterizations. This approach has the caveat that real user workloads may vary from those used in Digital's testing and that results may be different from those obtained. However, the methodology is a useful one and could probably be used to test configurations with individual customer workloads to gauge capacity requirements that are unique to them.

In a client/server application, adding clients is a certainty. The intermediate server approach used by Accessworks allows capacity to be increased in much smaller increments than would be possible if the database server itself had to be upgraded when its capacity was reached. In addition, Accessworks servers can be upgraded either by adding processors to

the multiprocessor servers or by adding additional clustered Accessworks servers to the network. This would be an ideal application for the distributed database technology Digital is developing and demonstrated at DECworld.

Future Directions

Feature Enhancements

A deferred query capability is planned for a future release of Accessworks that will allow queries to be stored in the server and executed according to a predetermined schedule or event. Deferring a query would allow the execution of queries that are expected to return a large data set during periods of light loads on the host.

Deferred query will include support for an event-triggered query that would refresh the data set on the server when a programmed event occurs. This same capability could be used to continually update the database extract held on the server to provide up-to-date data for decision support. Deferred update, the complement of deferred query, will also be supported in a future release.

Digital announced plans in March to enhance the site server with a graphical schema browser that will allow users to look at host database schema and customize their query and reporting applications. This will allow users to access desired data more easily without needing the support of a database administrator.

Other expected enhancements that we expect from Digital this year are support for Information Builders Incorporated's (IBI's) EDA/SQL and Sybase Open Client APIs.

Expanded Capability

Over the coming year, Digital has a number of other enhancements planned for Accessworks, most of which are the result of work being done on component products like Rdb and Pathworks that Accessworks will inherit. For example, the ability to write to DB2 and Oracle databases that will be included in a future release will take Accessworks beyond decision support applications. Support for SAG's FAP, mentioned above, will be followed by support for the Call Level Interface (CLI), for which the specification is nearing finalization.

Digital plans to enhance the management capabilities of Accessworks as Digital's Enterprise Management Architecture (EMA) converges with the OSF's DME architecture. As this occurs, Accessworks servers will begin to manage back-end databases as well as databases on the server. Digital's Management Control Center (MCC) manages the network piece today, but it does not have the ability to manage the database component. In the meantime, real-time database access monitoring will be added to Accessworks that will allow performance to be monitored with a graphical display and provide integrated database administration and performance planning tools. The Motif-based graphical user interface (GUI) will display all linkages to and from clients and servers. It will be able to start and stop all servers. The manager performs real-time monitoring, polling all the functions on all the servers. If a failure is detected using preset thresholds, the manager will try to restart the server automatically. Included is a messaging server that can then take additional action as necessary.

But Is It Open?

Is Accessworks a proprietary product or an open product? It is probably some of both. To the extent that it supports industry-standard interfaces on both the client and the remote server sides of its architecture, it is the ultimate open systems product. To the extent that it is a tightly integrated package running on a Digital server, it is proprietary. The pragmatic user will see Accessworks as a means to a highly desired end—access to heterogeneous data

sources from heterogeneous desktops using standard APIs and protocols. On the other hand, the open systems purist will see Accessworks as an attempt by Digital to interpose a proprietary layer of middleware between clients and servers and make users dependent on the proprietary Digital middleware solution.

We believe that as long as client applications are written to standard APIs and communicate with the Accessworks server over standard transports, and remote server applications are not being specially modified, then Accessworks is completely replaceable, and, therefore, open. In the long run, when client applications and remote service providers are all supporting a common set of interfaces and protocols, Accessworks may become obsolete. In the interim, however, it provides a way to achieve open systems objectives, even when all the methods are not available.

Digital's rise in the mid-1980s was based on its ability to deliver highly functional networks for computers. As the value of proprietary computers fell at the beginning of the 1990s, so did Digital's fortunes. Accessworks represents a return to Digital's roots, the network. By resolving many of the complex interoperability issues in one box, Digital confronts one of the most costly issues facing users. The key for Digital will be to position Accessworks as a complete middleware package based on standards and carrying the Digital brand. Customers would much rather buy a packaged solution for their interoperability challenges than develop their own solutions.

Next month's *Open Information System* will address
Integrating Applications in the Real World.

For reprint information on articles appearing in this issue,
please contact Donald Baillargeon at (617) 742-5200, extension 117.

Open Systems: Analysis, Issues, & Opinions

FOCUS: NETWORK MANAGEMENT

Sun Fortifies Its Position in SNMP Management

Sun Running Strong

SunConnect, (Billerica, MA) the networking products subsidiary of Sun Microsystems, recently announced major new versions of its Simple Network Management Protocol (SNMP) management platforms. SunConnect is seeking to build on the strength of its 4,000 existing licensed installations and 73 third-party applications providers.

SunConnect has a two-year lead on Hewlett-Packard's OpenView in the market. SunNet Manager was the first real SNMP applications platform; now, SunNet Manager 2.0 builds on SunConnect's first release. It does not introduce startling new features: Its major enhancement is the performance tuning of Sun's proxy agents to support larger networks and the addition of new ease-of-use improvements.

At stake in this competition is a controlling position in the future of network management applications. SNMP is the most widely used network management protocol, and SunNet Manager and OpenView are the most widely used SNMP platforms for multivendor management. As users migrate to more functional network management solutions based on the International Standard Organization's Common Management Information Services/Protocol (CMIS/CMIP), both HP and SunConnect will be in position to capture that business as well.

SunNet Manager 2.0: Incremental Improvement of a Stable Product

SunNet Manager was really the first modern network management platform. It was announced as an SNMP product for TCP/IP networks in late 1989, and it immediately began to change the rules of the network management game. Sun's strategy for SunNet Manager was to make it so easy, robust, and inexpensive that management solutions providers would build on top of SunNet Manager rather than build their own individual management platforms. In the process, Sun believed, hundreds of Sun workstations running SunNet Manager would be sold into customer sites.

Sun's strategy did not include provisions for porting SunNet Manager to multiple hardware platforms or for licensing the core technology to other systems vendors for use in their management solutions. It was a solution that would help sell Sun hardware, pure and simple.

SunConnect, which inherited SunNet Manager from Sun Microsystems in the big corporate reorganization of 1991, followed through on this product strategy. As a result, HP has been able to grab the role as the leading multivendor management platform. With the announcement of SunNet Manager 2.0, SunConnect has begun fighting back by challenging HP OpenView's relatively simple feature set and HP's ability to manage large networks in the near future. In addition, SunConnect is pledging to provide versions of SunNet Manager on the IBM RS/6000, Intel-based platforms, and the HP 9000 at an unspecified future date.

SunConnect has also done something that, for Sun, is extraordinary. SunConnect has pledged to support interoperability with DME—not implement DME, but rather interoperate with it. In the old days, Sun wouldn't have made even as bland a statement of support for an OSF product as that. However, SunConnect's pledge has some substance behind it: The company has recently begun working with Tivoli (Austin, TX) to integrate Tivoli Wizdom with SunNet Manager, using SunSoft's distributed object management technology as the integration medium.

Major Features of SunNet Manager 2.0

Aside from the statements of intent to deliver on a multivendor strategy, the announcement of SunNet Manager 2.0 did not represent a departure from what Sun has been doing with its management platform for the last two years. Rather, SunConnect has made improvements to the basic product. Those improvements fall into two general areas: usability and support for larger networks.

NEW USABILITY FEATURES. SunNet Manager 2.0 has usability features that HP OpenView doesn't have yet. These features offer network administrators some nice conveniences. The major innovation in usability is the coupling of autodiscovery of IP addresses with an "automanagement" feature. This means that once SunNet Manager 2.0 discovers a node, it automatically begins reporting on whether that node is available. The user (the network administrator) doesn't have to do anything to get this basic level of management function. Network administrators will have to provide further definitions of individual nodes to expand beyond the basic management functionality.

SunNet Manager 2.0 also incorporates a flexible user environment. The major improvement is a single Control Window that allows administrators to view all active management queries from a single vantage point. SunNet Manager makes it easy to build management queries of network elements as well. In addition, SunNet Manager 2.0 now allows administrators to define general behavior, such as alarm thresholds, for whole classes of devices and events. Previously, administrators had to define behavior for each episode.

Last, SunNet Manager 2.0 has multiple levels of topological maps. The most interesting of these is a "perspective view" option that allows administrators to create custom views of the network. For example, an administrator can use this feature to very quickly present a view of all routers on the network. Perspective views are simple to define.

SUPPORT FOR LARGER NETWORKS. SunNet Manager 2.0 also responds to the growing complexity of the environments SunConnect finds its product in. SunNet Manager was originally developed as an SNMP manager running on TCP/IP networks. Since then, Sun has rolled out support for DECnet and FDDI networks. SunConnect's ISV and OEM partners have added support for other protocols, including AppleTalk and SNA.

Sun's response to this has been to tune its existing architecture for managing distributed networks. SunNet Manager does not require that all management functionality be resident in a single manager console. Rather, SunConnect delivers "intelligent management" components that localize management polling for information about a device's status to a single protocol. The intelligent management component can filter this one protocol and report back to the central manager console only preselected information.

With SunNet Manager 2.0, SunConnect is bringing out new, streamlined intelligent management components. They perform faster than the previous generation.

This approach achieves the goal of reducing the amount of network traffic required for the manager console to keep in touch with agents around the network. It also succeeds in distributing some management intelligence. But it is clearly only a step toward a distributed management solution. The ultimate solution will be to have distributed but coordinated multiprotocol management consoles.

SunConnect's Multivendor Support Plan

The big news for SunConnect is that it has a plan to support platforms other than Sun hardware for its management software. Beyond this statement of direction, the company is holding its plans close to the vest. The IBM RS/6000 and the HP 9000 are targets, but we don't know when Sun plans to deliver.

It does seem obvious, however, that SunConnect is following closely the lead of SunSoft, which develops and markets the Solaris distributed operating environment in pursuing its multivendor strategy. SunSoft plans to port Solaris to Intel-based computers during 1992, and SunConnect will be there with SunNet Manager when Solaris for Intel is ready.

Beyond this obvious step, however, SunConnect is positioning the Object Management Group's Common Object Request Broker Architecture (CORBA) technology as its foundation for future multivendor interoperability and integration of new functions into SunNet Manager. SunSoft is a primary implementor of CORBA technology and a natural source of it for SunConnect.

The first need SunConnect plans to use CORBA technology to solve is the integration of its network management platform with Tivoli's Wizdom platform. Wizdom uses a facility very much like CORBA to manage systems elements. Indeed, the primary challenge of the OSF DME project is to integrate Wizdom object management services with HP OpenView object management services. SunConnect plans to take a different approach: Rather than integrate SunNet Manager management services with Wizdom's equivalent services, it will use Solaris's CORBA implementation to allow the two to talk.

The remaining question is about the notion of a common management API. SunConnect currently offers APIs for its topology map, its manager services, and its agent services. How will SunConnect provide a single set of calls to both network management and system management functions? The answer is, SunConnect doesn't plan to do so. Rather, it will seek to support multiple APIs for major management functions with a common base in Solaris's object management services.

Over time, then, SunConnect may seek to merge new management functions into its own API structure.

Pricing, Availability, and Packaging

SunNet Manager has never been a costly product, and SunConnect isn't changing this market posture with SunNet Manager 2.0. The list price for the network management platform is $3,995. That's less than Novell's new NetWare Management System for PC LANs. SunNet Manager 2.0 is scheduled for July 1992 availability.

Conclusions about SunNet Manager 2.0

SunConnect's SunNet Manager 2.0 announcement is most interesting for its least substantive elements—the new multivendor strategy. As far as the product goes, SunConnect has clearly decided that it can reverse HP's momentum by delivering features that network administrators will treasure and attacking the management of large-scale environments before HP can bring DME technology to market. These are variations on the existing SunNet Manager theme, not the substance of a major new announcement.

SunConnect's strategy has at least a fighting chance. It is likely that DME will be widely adopted by major corporate users; however, there's no guarantee that DME will be an overnight success. We won't know until we see the implementations from IBM, HP, Groupe Bull, Olivetti, and others. In the meantime, SunConnect is ahead of the pack in delivering management functionality to users. This is the main reason that Boeing recently anointed SunNet Manager as its choice to manage its network.

Where SunConnect's strategy begins to fall apart is in large multivendor networks. Sun has a strategy for dealing with multivendor requirements, but we can't judge its chances of succeeding for lack of details. For Sun to regain the momentum it has lost to HP in network management, it will have to demonstrate that it can handle large, multivendor networks and work well with other management systems. SunConnect didn't do this with the SunNet Manager 2.0 announcement. Stay tuned. *—J. Rymer*

FOCUS: SYSTEM SOFTWARE

Unix Desktop Battle Is On

Three Contenders—One Challenger

The battle for domination of the Unix desktop has begun in earnest with Santa Cruz Operation (SCO) announcing the availability of Open Desktop 2.0 and SCO Unix Version 4.0, Unix System Laboratories (USL) introducing Unix System V Release 4.2, and SunSoft shipping Solaris 2.0 for Sparc. These three variants of USL System V will all be competing for many of the same desktops, although each company believes that it is uniquely positioned to dominate in at least selected market segments.

FINDING A COMFORTABLE NICHE. SCO believes it will continue to prevail in the commercial workstation and small-business multiuser market for Intel-based hardware. These are its areas of traditional strength, and the company believes that the breadth of its hardware support and its large application catalog will help it maintain its position. SunSoft hopes that its leadership in RISC workstations will extend to Intel-based commercial desktops on the basis of its strengths in networking, object-oriented technology, and software applications. Unix System Laboratories is positioning its desktop Unix as the universal solution, targeting all popular processor architectures, as well as popular PC networking environments.

COMMON THREAT FROM NT. All three Unix suppliers face a common threat from Microsoft's Windows NT, which will be aimed at similar markets and will address a similar, though narrower, range of architectures. The key difference is that NT has a DOS and Windows heritage and the others share a Unix heritage. NT's main advantage is the common API it shares with Windows, Win32, which will allow developers to bring thousands of graphical applications over to NT in a relatively short time. This makes it a simple matter for many of the 10 million Windows users to upgrade to NT rather than switch to a new Unix environment.

There is no doubt that a market requirement exists for a Unix desktop operating system. The question is whether the market will support at least three different offerings. While each shares compliance with a common set of standards and APIs, they are different enough to require a choice to be made among them.

SCO Open Desktop/Unix Release 4

SCO has an installed base of over 650,000 licenses supporting more than 5 million users. Its annual shipment

rate is currently running at 200,000 units. While its $170 million in software sales in 1991 may appear small, that software was installed on $5 billion worth of systems. Open Desktop now represents 30 percent of SCO's revenue, whereas the older, 16-bit Xenix product still accounts for 35 percent of revenue. SCO's key industry segments are banking, finance, securities and trading, transportation, manufacturing and distribution, retail, and miscellaneous service industries. Although its sales have been weighted toward small businesses, that trend is shifting in the direction of departmental and enterprise applications. In fact, the company has established a major account sales force whose compensation is based on sales in excess of 1,000 units into each account.

NEW IN OPEN DESKTOP. Open Desktop is SCO's single-user desktop operating system for Intel-based personal computers. It is built on top of the new Version 4 of SCO's Unix System V Release 3.2 base and now includes the Motif 1.1 graphical user interface, the Motif toolkit, and IXI's X.desktop 3.0. Although the LAN Manager Client that is included is the older 1.1 release, it will be upgraded to LAN Manager 2.1 in the Fall. MS-DOS services now include DOS 5.0 support, including support for volumes larger than 32 MB, and EMS support. Windows 3.0 applications are supported in Standard Mode, and Windows 3.1 will be supported by the end of the year.

SCO Unix Upgraded. Version 4 of SCO's Unix now supports up to 512 MB of RAM, unlimited SCSI host bus adapters, and disk drives larger than 1.2 GB. The package includes the newest releases of TCP/IP, NFS, and NIS. Multiprocessing support comes in the form of the SCO MPX extension, which is purchased separately. Installation has been simplified and can now be done from CD-ROM.

The current release continues to support over 800 peripherals. Among them is a broad range of hardware options, including many different display types, mouse devices, disk drives, and tape drives, as well as over 200 different Intel-based platforms. In addition, there are 17 vendors supporting SCO's mutiprocessor option.

SCO continues to build on its System V Release 3.2 base. It has made no commitment to support System V Release 4 (SVR4) even after shelving its OSF/1 development (see *Unix in the Office,* May, 1992) Part of its marketing message has been that a higher release number does not necessarily signify a better product. Now, by naming the newest version of its Unix operating system Version 4.0, it seems to be intentionally creating confusion as to whether it is based on System V Release 4.0, which it is not. It is compliant only with SVID 2.0, which makes it SVR3-compatible.

Enhancements for Developers. The Open Desktop Development System includes an updated Microsoft C6.0 compiler, CodeView debuggers, DOS as well as OS/2 libraries, and dbXtra, a graphical debugger. Interfaces and libraries are included that support long filenames, symbolic links, and Network Information Services.

PRODUCT PACKAGING. SCO has repackaged its offerings with this release. There are now three separate products: SCO Open Desktop Personal System for individual users, Open Desktop Development System for developers, and Open Desktop Server system for multiuser systems and server applications. In a change from earlier versions, Open Desktop 2.0 no longer bundles the Ingres DBMS.

FUTURE DIRECTIONS. SCO is evaluating its future technology directions in everything from the operating system kernel to distributed object management. It is looking at both the OSF microkernel and the USL SVR4 technology as well. Its goal is to provide both SVID3 and OSF Application Environment Specification (AES) compliance. To the extent that those two environments merge, its task will be made easier.

Sparc to Intel: SunSoft's Solaris 2.0

Announced last year, Solaris 2.0 for Sparc is now available to customers along with development and migration tools. Solaris 2.0 for Intel is still due by the end of 1992. Solaris 2.0 is compliant with Sparc Compatibility Definition 2.0 (SCD 2.0), SVID3, POSIX 1003.1 and supports ANSI C and the Device Driver Interface/Device Kernel Interface (DDI/DKI) specifications.

Although most applications written for SunOS 4.2 will run on SunOS 5.0, most developers will want to rework their software to take into account the differences between the old and new operating systems. Networking is now based on Transport Layer Interface (TLI) and Streams instead of Sockets, although Sockets are supported in a compatibility library. Device drivers must now conform to the DDI/DKI specification. Sun C has been replaced as the primary programming language by ANSI C. SunView applications are supported in a compatibility mode, and new SunView application development is not supported.

Once an application is moved over to the Solaris environment, its source can be compiled for the Intel version. Compatibility mode features are not supported in the Intel version.

SOLARIS 2.0 CONTENTS. Included under the Solaris 2.0 banner is foundation technology, a developer environment, and a user environment. This packaging is not unlike the packaging of SCO Open Desktop.

Foundation Technology. The foundation technology comprises SunOS 5.0 and related services for distributed computing, security services, support for real-time applications, and integrity features for commercial applications. SunOS 5.0 is the result of the effort to merge System V Release 4, Xenix, BSD, and SunOS. It is fully SVR4 compatible and is SVID3 compliant. The kernel is multithreaded and supports symmetric multiprocessing. Support for disk striping, mirroring, and concatenation will appeal to commercial users. Internationalization is assisted by 8-bit clean characters, support for international keyboards, and local language formatting and messaging.

The foundation includes enhancements to ONC and NFS. Support for the TI RPC (See *Unix in the Office*, April, 1992) is included, as are Solaris Federated Services, which provide an interface for third-party network services, such as NetWare and the OSF's Distributed Computing Environment (DCE), which would be supplied by third parties. Solaris 2.0 features three security options: Kerberos and SunSoft's ARM and ASET.

Developer Environment. The Solaris Developer Environment is provided as an extension to the foundation and is aimed at development for both the Sparc and Intel platforms. Included in the developer environment are OpenWindows Version 3, ToolTalk, and OpenWindows Developers' Guide (DevGuide).

User Environment. For users, SunSoft has a bundle of goodies designed to make the system useful and easy. Included is DeskSet Version 3, a collection of 15 graphical applications and utilities. Also included in the user environment are MultiMedia Mail, an Audio Tool, Workgroup Calendar Manager, and Magnify Help.

PC VENDORS SUPPORT SOLARIS 2.0 FOR INTEL. A number of PC vendors have announced support for Solaris 2.0 for Intel-based computers, including Dell, Zenith, CompuAdd, Everex, Olivetti, AST Research, and Toshiba. Other Intel system vendors, including NetFRAME NCR and ICL, have indicated support. Hardware support for devices is not an area that SunSoft is concentrating on, placing the PC vendors in the position of having to do the bulk of the driver development for Solaris 2.0 to run on their systems. Therefore, having PC vendors support Solaris is important if it is going to be installable on a sizable portion of the installed base. SunSoft needs to get Compaq and IBM into the picture, but we believe that support will have to come from third parties.

Solaris for x86 will be in controlled release over the summer. Early access for developers will begin in the third quarter with user availability late in 1992 or early in 1993. The hardware that SunSoft will support is limited, considering the wide variety of options available on Intel machines. In each category, such as SCSI cards, SuperVGA chipsets, network adapters, and mouse devices, only one or two vendors' products are being supported by SunSoft. Relying on third parties to develop drivers for the vast array of PC components is problematic because driver development under Solaris 2.0 requires writing to a relatively new specification. There will be a definite learning curve for developers.

SUNSOFT STRATEGY. Skepticism abounds regarding how serious SunSoft is about the Intel market. Many analysts refuse to believe that SunSoft is free to make marketing decisions independent of its Sun Microsystems parent. We believe that SunSoft is serious about its Intel strategy, but it is only beginning to learn the ropes in PC land. SunSoft does not see itself competing against SCO or USL—its competition is Microsoft and Windows NT. In that case, it has to compete as a shrink-wrapped operating system. Shrink-wrapped operating systems come with hundreds of device drivers and are purchased in computer retail stores and by mail order. Post-sale support is provided by telephone by the vendor, often at little or no cost. SunSoft will have to be willing to play by these rules if it wants to win the PC game against NT.

SVR4.2 AND THE NEW USL

There have been dramatic changes at USL following Roel Pieper's taking over leadership of the owner of Unix. The company is now driven by market requirements rather than pure technology and has given up religious purity for the sake of market impact. USL's entry into the desktop Unix market was carefully considered as one component of a broader attempt to provide better-quality technology that would reduce its customers' time to market and improve their market position. The apparent stalling of OSF/1's momentum has been one noticeable result of this effort.

DESTINY USURPS THE 4.2 DESIGNATION. USL's announcement of the product that had been code-named Destiny as Unix System V Release 4.2 was the opening of its campaign to gain dominance in low-end operating systems. The original road map had Release 4.2 as the merger of the 4.1 Enhanced Security and Multiprocessing releases. Since Destiny made it to market first, it has assumed the 4.2 mantel.

USL's strategy at the desktop has five main components:

- leverage Univel, Novell, and NetWare
- drive Unix System ABI-compliant operating systems onto low-end Intel systems
- define applications development tools that focus on client/server applications
- converge the market on a single desktop manager API/product
- leverage the Unix System's strengths in client/server and peer-to-peer computing for both LANs and WANs

The SVR4.2 product and the work that the Novell-USL joint venture, Univel, is doing to port NetWare to SVR4.2 are the implementation of that strategy.

SYSTEM V RELEASE 4.2. The base for this release is the 4.1 Enhanced Security release (See *Unix in the Office*, February, 1992). It features a graphical user interface that is user selectable for either the Open Look or Motif look and feel. Drag and drop is supported for many desktop operations, such as sending mail, printing, and many administrative tasks. Adobe Type Manager is embedded in the operating system, supporting Type 1 fonts for accurate screen representation of typefaces. Emulation utilities for DOS and Windows 3.0 have been built into the kernel to provide support for applications and data formats.

This release has dynamically loadable kernel modules that let printers, disks, and other devices be installed while the system is running. Dynamic loading and unloading also reduces system RAM requirements and improves performance on low-end systems. The desktop installation of SVR4.2 requires as little as 4 MB of RAM and 80 MB of disk.

A new journaling file system provides rapid startup times, higher performance, and greater integrity in I/O intensive applications. For internationalization, SVR4.2 supports 8-bit and multibyte characters, Extended Unix Characters (EUC), ANSI C internationalization enhancements, alternate date and time formats, an extended internationalized curses library, XPG3 Messaging Facilities, and Streams support for international input methods.

PACKAGING OPTIONS. Since USL does not sell directly to end users, its packaging strategy is aimed at the requirements of its OEM licensees. USL has repackaged the source code of SVR4.2 into foundation, utility, and feature sets allowing system vendors to customize their offerings more easily. To do this, USL extended the modularity of SVR4.1 by isolating processor-specific source code from the main body of common code. As a result, SVR4.2 will support multiple architectures from a single source tree, and only the processor-specific code has to be rewritten. This will enable the operating system to be ported more easily by USL's porting partners to Sparc, Mips, PowerPC, and eventually PA RISC and Alpha.

COMPATIBILITY. SVR4.2 is binary compatible with applications written for the Intel ABI for SVR4 and SVR4.1. It is also source-code compatible with applications written for System V, SunOS, Xenix, and BSD. Naturally, it is SVID3 compliant, because SVID3 is based on SVR4 interfaces. It is POSIX 1003.1 compliant, and USL will work to implement the OSF's Application Environment Specification as well.

DEVELOPMENT TOOLS. Although there is no separate packaging of a developer edition, there are several new tools for developers in SVR4.2. A new C Optimized Compilation System, Release 2.0, is provided, along with a new graphical debugger that takes advantage of the desktop environment. The MoOLIT Toolkit allows developers to create applications that can be dynamically switched to either Motif or Open Look based on a user's preference at runtime. A Windowing Korn Shell extension enables rapid development of graphical applications using standard Korn shell programming. Also included is the Application Builder, based on Integrated Computer Solutions' (ICS) Builder Xcessory interactive design tool. This environment allows rapid development of interface designs, generates C code, and reduces application development time.

DISTRIBUTION CHANNELS. Shrink-wrapped SVR4.2 will be distributed by Univel through its dealer network and by other system software vendors. In addition, USL is providing binary-ready packages to OEMs for inclusion with their systems. OEMs can also license source code or master binaries that give them the flexibility to use the scalability inherent in SVR4.2 to deliver it on non-desktop systems.

The shrink-wrapped SVR4.2 from USL will be the same product Univel sells. The work that Univel does to integrate NetWare with SVR4.2 will be licensed back to USL and available in binary form to OEMs through USL.

WHICH WILL IT BE? Will the market support three different end-user desktop Unix systems? We believe that the

competition among the three competitors would be healthy for the Unix industry if it weren't for the fact that they are all going to be competing against Microsoft's Windows NT. Even if the first release of NT is not ready for prime time, customers will begin getting a sense of its capabilities in 1993 and may defer moving to any other 32-bit multitasking, multithreaded operating system until NT is sufficiently robust. If the Unix vendors are serious about convincing customers that Unix is a viable alternative on the desktop, then they must ensure complete binary compatibility within architectures and source compatibility across architectures all the way up to the GUI. Without that kind of uniformity, the 3,000 developers at the NT developers' conference will put Unix at the bottom of their development list, right next to OS/2. —*M. Goulde*

DISTRIBUTED OBJECT DEVELOPMENT TOOLS WATCH

SynchroWorks Makes Sense of Visual Object Programming

OOP for the Uninitiated

Oberon Incorporated (Cambridge, Massachusetts) is completing a development tool that promises to make object-oriented programming (OOP) accessible to more organizations. OOP is quite different from other disciplines of software development: Not only are the languages and tools different, the way the developer views a software application is different. Because there are relatively few software engineers who know the object-oriented viewpoint, many organizations have been unable to realize the gains in modularity, reliability, and reusability that OOP provides. With SynchroWorks, Oberon is seeking to put OOP capabilities into the hands of non-OOP developers.

Oberon calls SynchroWorks a fifth-generation language, or 5GL. In fact, SynchroWorks does represent another step along the path from 3GLs to 4GLs and beyond. 5GLs take basic 4GL capabilities and add object-oriented development features, such as encapsulation of objects, abstract data typing, and object class hierarchies. Developers access these features using a graphical development environment, not by writing code.

If this simple description sounds familiar, it should. Geode from Servio Corporation (Alameda, California)—see Vol. 7, No. 2, February 1992—is a similar approach. This visual approach to OOP represents one of the most promising avenues for object technology to penetrate customer sites. Both SynchroWorks and Geode significantly reduce OOP entry costs by minimizing the need for retraining development staff. But while Geode is more directly focused on database application development, SynchroWorks supports general application development. There are other distinctions between SynchroWorks and Geode that we discuss below.

Applications without Program Text—*Really*

SynchroWorks allows a developer to construct applications within a graphical environment without ever having to drop into textual programming. It is important to emphasize this feature because of the long history of products that claimed to eliminate software coding but didn't. The enabling technology for this key feature is Oberon's object-oriented approach, specifically, object encapsulation. Encapsulation formalizes an object so that it conforms to an abstract definition of its composition and behavior. While an object may be implemented in C++, the application developer sees that object only in the abstract and uses it without employing any C++ code of his or her own.

Primitive SynchroWorks objects are, in fact, written in C++. Specialists in systems programming or in some distinct discipline can build SynchroWorks objects in C++ for others to use. (Under Geode, primitive objects are built in Servio's dialect of Smalltalk.) But the application developer can simply breeze along in the visual programming environment, rarely, if ever, touching application program texts.

Using SynchroWorks to Program

SynchroWorks developers build applications on a "canvas," which is partly the backdrop and partly the wrapper for the application. Developers take an object definition from an object library and drag it onto the canvas. Oberon calls these "building blocks" (as does Servio for Geode).

SynchroWorks building blocks are not always objects in the pure sense of the word. Some building blocks, such as simple comparison operators, have no stored data, but they are crucial to implementing logic at the object level. Other SynchroWorks building blocks may be more complicated than the usual OOP objects, such as aggregates of other objects or building blocks. The developer can put any building block, primitive or high-level, onto the canvas and put it to work.

After the developer has placed the objects onto the application canvas, he or she needs to assemble them into a working system that performs the intended application function. This is done by connecting the building blocks at their "ports." Those familiar with OOP will be

tempted to think that a SynchroWorks port is the same as an object method. However, object methods are inherently passive; that is, they can only be invoked. A method can't activate itself to run. Ports, on the other hand, are "object-smart" communications sockets that can be active or passive. Simple ports have a data type, such as "file name," a label, such as "mail-merge template file," and an indication of whether they are "input" or "output" ports.

EXAMPLE: A MAIL-MERGE APPLICATION. A developer might, for example, use SynchroWorks to implement a mail-merge program using building blocks representing a word processor, a file selection dialog box, a customer-database view, and a printer. The developer would draw a connection from the file selection dialog box output port to "mail-merge template file" input on the word processor building block, and from the database to a "mail-merge data" input port, and then route the word processor print output to the printer's input port. (See Illustration A.) The resulting SynchroWorks application displays a dialog box, such as "Please select a mail-merge template file," and then the word processor is invoked to run the mail merge with that file and data from the database view. The results are sent to the printer.

Sample SynchroWorks Mail-Merge Application

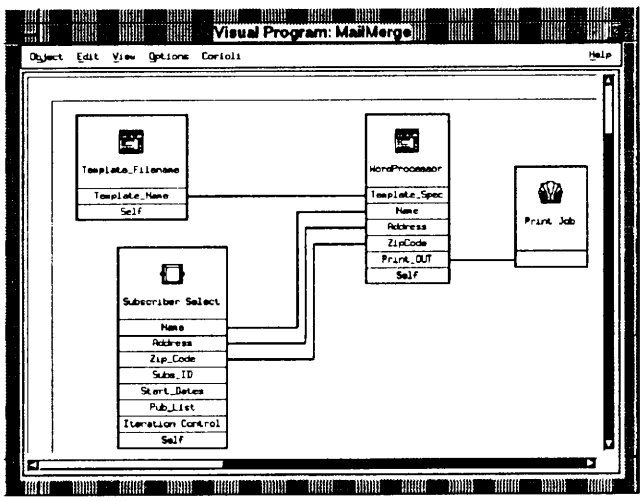

Illustration A. This is a draft of a mail-merge application built using SynchroWorks. Lines join ports of the building block objects. This prototype shows how the visual interface displays all pertinent aspects of the application objects.

Ports and Connections Reduce Object Dependencies

The SynchroWorks definition of object building blocks is a distinct advance over traditional object-oriented systems that make objects more independent and allow application development to take place at a high level. In conventional object systems, an object communicates with another object by explicitly addressing it. Under some systems, the object invokes some unnamed instance of the target object class. SynchroWorks further abstracts this operation by having objects invoke only their own output port. The word processor no longer has to invoke a printer by "HallwayLaserPrinter.Print" or by issuing a command such as ":printerVariable.Print." Instead, it simply writes to the output port. The destination is completely in the hands of the developer. This greatly enhances the modularity and reusability of building blocks under SynchroWorks.

PROTOCOL RESOLUTION BETWEEN OBJECT PORTS. The discussion of building blocks, ports, and connections may give the impression that SynchroWorks is little more than a data-flow builder. After all, you simply plug a file selection block into another block that manipulates named files to define application logic. But while SynchroWorks certainly supports such an elementary, data-flow-like construction, this is only the beginning of its real power in assembling a group of objects to create an application.

Oberon perceived that the set of methods and outputs of an object fundamentally constitute an object protocol. After all, they constitute a set of rules, formats, and conventions for communication and cooperation with outside entities. This view naturally suggested the implementation of ports with both active and passive characteristics, sometimes client-like and sometimes server-like. For example, the printer building block in the above example could send messages back to the word processor such as "send some more data" or "stop transmitting data."

But the port protocol does not need to be a single, concrete protocol. It can be an abstract protocol instead. Instead of specifying that the printer's input use Hewlett-Packard's PCL as a print protocol, the printer building block might have a port for input of type "image." Under this type, the port might accept PCL, Adobe PostScript Level 1, and raw text with carriage returns.

Similarly, the word processor's output port for printing might be of type image, generating PostScript Level 1 or raw text. SynchroWorks will automatically resolve the available protocols between the ports you connect,

so that the most powerful communication format is used.

Migrating Existing Unix Code

Oberon provides a tool that simplifies the migration of existing Unix software to a SynchroWorks object system. Its Unix Interface Builder, depicted in Illustration B, is a GUI-based utility that helps a developer construct a "wrapper" around an existing Unix program. The wrapper defines inputs, outputs, and parameters for that program according to the Unix conventions for invoking programs. When finished, this wrapper becomes a SynchroWorks building block.

The developer can then use the "wrapped" Unix program in the SynchroWorks graphical application builder like any other building block. The inputs, outputs, and parameters become the ports on that building block.

Taking the mail-merge example, the word processor building block is actually a SynchroWorks wrapper around an existing word processor program. That is, SynchroWorks allows the developer to integrate standard, unmodified Unix commands, utilities, and applications into his or her object system.

Application Management in SynchroWorks

Like Geode, SynchroWorks is built on top of an object-base, in this case, ObjectStore from Object Design Incorporated (Burlington, Massachusetts). All SynchroWorks objects are defined and stored in ObjectStore. The object management facilities in ObjectStore can be employed directly to keep track of SynchroWorks objects or building blocks.

ObjectStore also allows SynchroWorks developers to build objects using multiple inheritance. The developer can combine several types of objects into a new object class, and the composition of that class will evolve not only through direct modification, but also through modification of the object classes from which it is derived. Multiple inheritance affords tremendous power in developing object-oriented applications and systems.

VERSIONING IS A SYNCHROWORKS WEAKNESS. Because messaging can occur between any two objects, and because of multiple inheritance, the dependency network in an object-oriented application can be very complex. Object versioning is very important in this environment because it allows a development manager to propagate object changes carefully and incrementally through applications and groups of applications. Unfortunately, the control available to SynchroWorks efforts is not very precise at this point. Until it is improved, we expect large-scale development efforts to face clumsiness and workarounds in configuration management.

Oberon's Unix Interface Builder

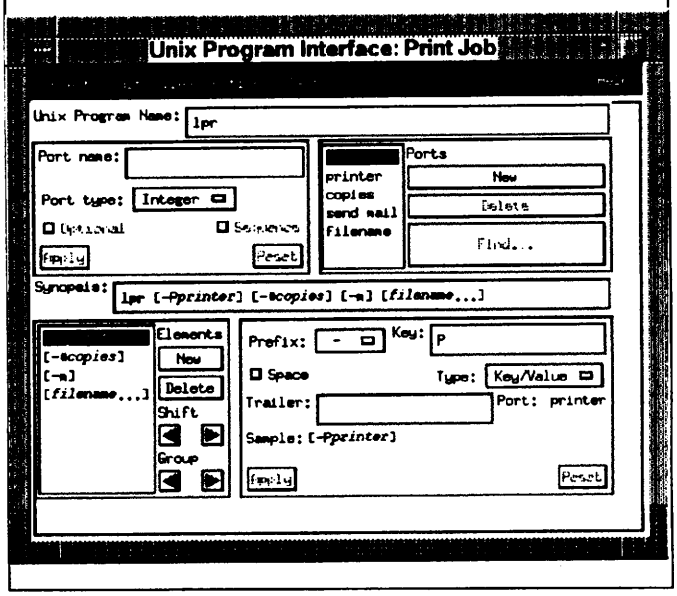

Illustration B. The SynchroWorks Unix Program Interface Builder provides a form-based GUI mechanism for integrating existing Unix programs into SynchroWorks applications. The Interface Builder lets you define the Unix program according to its command specification, almost like a manual page.

Oberon's CORBA Strategy

SynchroWorks, by virtue of its use of ObjectStore, supports development of client-server applications, in which client applications have access to server-based objects. However, SynchroWorks does not yet distribute application logic across multiple nodes. However, Oberon consciously constructed the underlying SynchroWorks messaging mechanism to be compatible with the Common Object Request Broker Architecture (CORBA) specification from the Object Management Group (Framingham, Massachusetts). CORBA defines a distributed facility; thus, when ORB implementations become available, SynchroWorks will be able to exploit them to build distributed object applications.

CORBA compatibility was also a factor in Oberon's employment of ObjectStore. Object Design is also directing its product toward CORBA compliance. Oberon's is a savvy approach that we think should pay off handsomely for itself and its customers when CORBA facilities become more widely available.

Is SynchroWorks Tied to One Platform?

SynchroWorks embodies some tremendous advances in the development of object applications. However, it is starting off on a somewhat restricted footing. It will be a Sun-only product for the near future, and it is dependent on ObjectStore to run. Servio's Geode operates almost as an add-on to The Gemstone Objectbase Management System, even as a high-level object DBMS interface, and sales of the two products reinforce each other. SynchroWorks depends on ObjectStore, but it doesn't have the mutual product reinforcement that would occur if both were made by the same company.

Oberon does plan to move SynchroWorks to other platforms over time. Oberon feels that its implementation reduces rehosting effort to little more than quality assurance. However, most Unix software is introduced on two or three platforms—a sort of proof to buyers that the product is portable enough to follow hardware advances. We would feel a bit more comfortable with SynchroWorks if the proof of its portability was in its availability on a second platform.

Conclusions about SynchroWorks

SynchroWorks is a very promising product. It supports object-oriented programming at a very high level, allowing developers to work more like architects and less like coders. Its visual programming approach is one of the few that is really powerful enough to encompass most of the semantics of textual programming facilities like C++. Moreover, being object-oriented, the visual programming is inherently extensible.

Meanwhile, the graphical metaphor is simple enough for non-OOP developers and even astute end users to employ. While we are disappointed in its (currently) nondistributed implementation, its weak support of configuration management, and its single-platform delivery, we feel that Oberon has shown its ability to deal with these issues effectively. For example, its CORBA-compatible messaging will solve most of the distributed system issues when an ORB becomes available. In our view, the distinct conceptual achievements Oberon has made in SynchroWorks, particularly its protocol-based port abstraction, make it an extremely fertile product for creating sophisticated object-oriented applications.

—A. Wolfe