Patricia Seybold Group

Editor-in-Chief
Michael A. Goulde

INSIDE

OPEN INFORMATION SYSTEMS

Guide to Unix and Other Open Systems

Vol. 8, No. 12 • ISSN: 0890-4685 • December 1993

OLTP Monitors

Key to Client/Server Robustness

By James J. Johnson and David Hudson

IN BRIEF: Maintaining openness in the design and implementation of open systems-based online transaction processing systems depends on equal parts of careful architecting and adherence to standards. Transaction managers are a critical component for managing transactions across distributed and even heterogeneous databases. As the technology evolves, transaction managers supporting key X/Open standard interfaces have become an important foundation for application development. Customers migrating from proprietary environments have their options as well, with transaction monitors supporting legacy applications being ported over to open systems.

Report begins on page 3.

Migrating to Open Systems

Part II

LAST MONTH, I wrote about our migration from DECnet to TCP/IP, fully anticipating that we would be connected to the Internet by this month. Little did I know that it would be "The Phone Company" that determined when we connected. After all, 56Kbs lines don't just appear overnight. Now I know all about DEMARCs, punch-down blocks, and phone closets. Next month, for certain, we'll be on the net.

While Archie, Gopher, Veronica, and Mosaic are all very interesting to us, the strategic reason for our migration is related to our data, databases, and applications. Our two key applications are subscription fulfillment and contact management. The former is currently managed by an application called Scribe, which runs on a PDP 11. It is the mirror image of everything one would want in a customer information system. Our contact management information is spread throughout the company in six different applications, as well as on paper and Post-it notes.

Our first attempt at correcting this three years ago, the KnowledgeBase, was an ambitious and comprehensive undertaking. It was client/server, with a Windows front end and a relational back end. But it was designed and built before many of the issues with large-scale client/server applications were understood. Performance was unacceptable, and the system was inflexible and hard to maintain. Even worse, it was monolithic and had to be completely discarded.

We learned a lot from the KnowledgeBase experience which we will be putting into practice in our new, open environment. One of the key things we learned was modularity. Each piece of the information architecture, as well as the applications themselves, has to be pluggable. We need to be able to remove any component with a minimum amount of bother in order to replace it if something comes along with better functionality. It also has to be extensible, so that we can add functionality without having to change any of the existing architecture.

Our business relies on information about people, companies, and technology and about their relationships. We manage information about subscriptions, orders for reports, consulting projects, conference registrations and speakers, and the marketing efforts related to all of these. Like any business, our information requirements call for consistent information, entered once, maintained by its users, managed in a consistent fashion, and managed independently of applications.

Our design for the future is a three-tier architecture, with user interface, application logic, and data management handled independently but integrated through consistent standards and interfaces.

Data management will be handled by an SQL-based RDBMS. User interfaces will be implemented using tools that embed minimal or no application logic, presenting just screens and interface widgets to the user. Determining a strategy for isolating application logic has been the hard part.

The notion of separating application logic from the client is not new. Many database companies favor this kind of "Skinny Client" strategy. The problem is that their solution is usually a "Fat Server" with application logic hard-wired into database triggers and stored procedures. 4GLs don't provide a much better option because of the lack of standardization. Once written, a 4GL application may be redirected to a different database, but it is generally hard-wired into the user interface.

We would like our middle tier to have application objects independent of user interface but acting on behalf of the interface against whatever database happens to be installed at the back end. Whether these need to be true "object-oriented" objects remains to be seen. We are pursuing alternatives at the moment and will keep you posted on the outcome.

OPEN INFORMATION SYSTEMS

Editor-in-Chief
Michael A. Goulde
MCI: MGoulde
Internet:
mgoulde@mcimail.com

Senior Editor
Wayne Eckerson
MCI WEckerson
Internet:
weckerson@mcimail.com

Publisher
PATRICIA B. SEYBOLD

Analysts and Editors
JUDITH R. DAVIS
STANLEY H. DOLBERG
MITCHELL I. KRAMER
DAVID S. MARSHAK
RONNI T. MARSHAK
JOHN R. RYMER

Copy Editors
ALBERT C. D'AMATO
MIRIAM F. D'AMATO

Art Director
LAURINDA P. O'CONNOR

Marketing Director
STEVE BLUMBERG

Sales Director
PHYLLIS GUILIANO

Circulation Manager
DEBORAH A. HAY

Customer Relations Manager
DONALD K. BAILLARGEON

Patricia Seybold Group
148 State Street, 7th Floor,
Boston, Massachusetts 02109
Telephone: (617) 742-5200 or
(800) 826-2424
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.

FEATURED REPORT BY: J. JOHNSON & D. HUDSON

OLTP Monitors

Key to Client/Server Robustness

No one appreciates the reliable and sound environment offered by centralized mainframes more than the seasoned information technology (IT) professional. These veterans know that the same measure of reliability and integrity must be maintained when moving mission-critical applications to open, distributed environments. The key to maintaining reliability and integrity in distributed environments lies with a mainstay of centralized computing, the online transaction processing (OLTP) monitor.

In a distributed environment, an OLTP Monitor supports critical features not found in other system software products: data integrity, high-availability operations, performance-tuning, load-balancing, and scalability. To date, most companies attempting to implement transaction-oriented client/server applications without the benefit of an OLTP monitor have either had to build these features into their application themselves or, more often, abandon these functions altogether. The primary benefit of incorporating an OLTP monitor system into a client/server or distributed application environment is the support of a development, execution, and management environment for mission-critical applications.

Application Design Considerations

Distributed application development entails the partitioning of processing functions among both logical and physical system resources. Basically, three current models can be applied to this partitioning:

1. The Remote Data Access (RDA) model
2. The Database Server (DBS) model
3. The Application Server (APPS) model.

In the RDA model, the application processing, along with the presentation services, is executed on the client device. Data are manipulated through SQL statements shipped over the network to a remote back-end database. The DBS model, on the other hand, executes application processing in the database using native stored procedures. Finally, in the APPS model, the presentation is implemented on the client, data management in the database, and a third logical tier—the application server tier—is used to isolate the business application logic.

The APPS architecture is summarized as follows:

- Clients are used to implement presentation functions via standard GUI and terminal APIs. Clients can also be implemented to invoke remote application services using standard network APIs and protocols.

- Application Servers model and implement business functions in the form of application services. They support well-defined interfaces to business functions, which are, in turn, expressed in standard 3GL and 4GL languages. Clients and databases are integrated via standard transaction processing systems and via APIs and protocols supporting the development, execution, and administration environment for critical business processes. Designs can be optimized for performance, availability, integrity, and the expression of complex business functions.

Key to Client/Server Robustness

- Databases are designed to model complex data relationships and manage the access to—and integrity of—the data themselves. Data access is provided via standard APIs and protocols such as SQL, the International Standards Organization's (ISO's) RDA, and IBM's Distributed Relational Database Architecture (DRDA). Databases integrate into the application server system (i.e., the transaction processing system) via standard integration interfaces such as X/Open's XA for distributed transaction processing.

Shortcomings of RDA and DBS

The problem with both the RDA method and the DBS method is that they do not approach the same level of integrity and reliability found in mainframe centralized computing (the model and precursor for APPS methods). Neither evolved to serve high-volume, high-integrity environments. The RDA method was developed for a less-demanding PC environment; the DBS environment was initially used for decision support. Each has several limitations when applied to transaction-oriented processing, and they share a common deficiency—limited scalability. The RDA model is network intensive, filling the client network with SQL-protocol messaging traffic. The DBS method pushes the bottleneck into the database, loading the RDBMS engine with application logic processing when it should be optimized for query and I/O functions.

The X/Open TP Model

X/Open (Reading, England) has defined a process model and related service interfaces for distributed transaction processing (DTP) applications. The model relates the various components of a distributed transaction application including: Resource Managers (RMs), applications (Apps - as opposed to the APPS model discussed above), and Transaction Managers (TM). An extension to the model currently under consideration adds Communication Resource Managers (CRMs). The process model is a straightforward one in which applications perform work on resource managers (most often relational database management systems, or RDBMSs) while the transaction manager components coordinate the commitment or rollback of the work. CRMs may also be accessed, particularly if transaction context needs to be passed to other participating transaction systems. For example, if a transaction within a Tuxedo transaction system domain requests work from an application routine within an ACMSxp system, an OSI-based communications channel would be used to distribute the necessary context, such as global transaction IDs, between the two systems.

Illustration 1 shows the so-called four-box X/Open model.

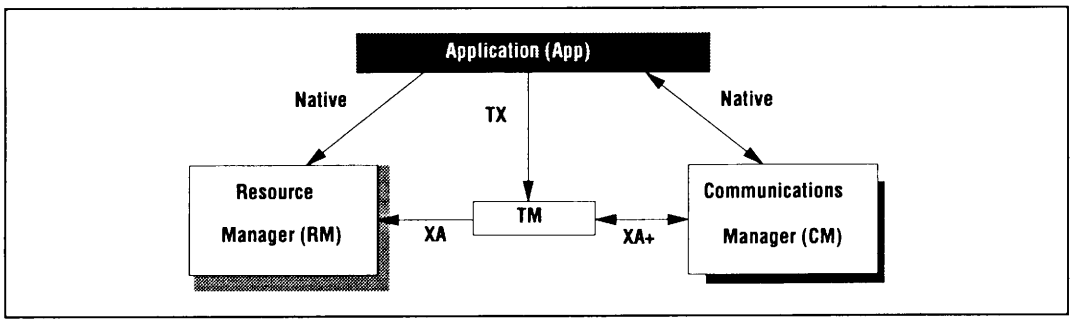


Illustration 1. X/Open's "four-box" model. Applications can communicate with Resource Managers, typically databases, and Communications Managers directly through native interfaces or through a TX-compliant Transaction Manager.

DTP and Principles of Clean Layering

The interfaces defined by X/Open (X/A, TX) isolate system dependencies and allow independently-developed system components to be integrated seamlessly. When the X/Open model is incorporated into an OLTP monitor, it offers a high level of software independence.

By masking dependencies, programmers are free to create a clean separation between the software support platform and the business application. This development method is called *Clean Layering*. It is also a set of principles designed to ensure software independence while adhering to industry standard APIs and protocols. Clean Layering is equally important for designing and implementing a new application or modifying legacy applications. The goal of Clean Layering principles is to protect and maintain the large investment made in application solutions. OLTP monitors offer protection for managing change in technology, products, features, business requirements, and enterprise IS organizations.

The following Clean Layering principles aid in isolating software dependencies while creating reliable, transaction-oriented client/server applications:

X/A-Compliant OLTP Monitors. The cornerstone for Clean Layering is an X/Open DTP, X/A-compliant, OLTP monitor/transaction manager. X/Open has defined the X/A interface for integrating heterogeneous resource managers (i.e., databases) to heterogeneous OLTP monitors. The X/A interface is virtually transparent to application developers but performs the vital task of coordinating application-driven resource manager updates as well as rollback and recovery in the event of system failures. The developer writes to the TP-system and database APIs, while, in the background, the database and transaction manager communicate via X/A protocol to ensure correct coordination.

X/A- and SQL-Compliant Databases. On the other side of the OLTP monitor is the X/A-compliant database. By using databases and TP monitors that are both X/A compliant, you have a "pluggable" standard interface offering independence for the component on either side. Increasingly, organizations will need to support multiple heterogeneous resource managers, including RDBMSs, object databases, store-and-forward queuing systems, and communications managers. The X/A interface allows each of these resources to be managed in complex distributed computations.

Programming in C, C++, and Cobol. C, C++, and Cobol are ANSI-standard languages, or tools that "emit" standard language code needed to develop critical business process logic.

COSE or POSIX Operating System. Using a standard operating system like the Common Open Software Environment (COSE) or POSIX in conjunction with standard APIs offers independence from both operating systems and the underlying hardware.

Common Graphical User Interfaces. The use of widely-accepted standard graphical user interfaces (GUIs) can provide a separation of presentation services from the application logic within the software environment.

Common Transaction Interfaces. X/Open DTP is developing a standard model and APIs for transaction-oriented computation. The X/Open interfaces support a flexible and rich model for structuring applications and are now garnering the support of many key transaction system vendors.

Common Transport Protocols. The TCP/IP network stack, IBM's LU6.2 protocol on SNA networks, and the emerging OSI network stack are the leading network transports. Various APIs allow programmers to write distributed applications over these network protocols using interfaces that include X/Open CM, XATMI, peer-to-peer, and TxPRC.

Open OLTP Monitor Products

The key to the reliability and scalability of the APPS client/server model is the OLTP monitor. With APPS, client devices invoke business applications (via some messaging or RPC mechanism), and results are returned. The business functions in the application server

communicate with the databases, effectively confining the processing/network-intensive SQL processing within a single node or an isolated LAN segment.

OLTP monitors incorporating standard application programming interfaces (APIs) and protocols (such as those of X/Open's Distributed Transaction Processing model) provide a number of key features specifically tailored to enable client/server and distributed processing applications across heterogeneous platforms. These include the coordination of heterogeneous databases within a single distributed operation (i.e., unit of work), as well as support of interoperability between autonomous, heterogeneous processors.

Currently, there are several OLTP monitors on the market. In this report, we will discuss 10 of these monitors: IBM's CICS/6000; Novell Corporation's (Provo, Utah) Tuxedo; Transarc Corporation's (Pittsburgh, Pennsylvania) Encina; NCR Corporation's (Dayton, Ohio) Top End; Digital Equipment Corporation's (Maynard, Massachusetts) ACMS; Micro Focus Group PL's (Newbury, England) Transaction System (MTS); Integris's (Billerica, Massachusetts) UniKix; VISystems Incorporated's (Dallas, Texas) VIS/TP; Gresham Telecomputing PLC's (South Hampton, England) tp+; and Allinson Ross Application Systems Incorporated's (Toronto, Canada) TIP/ix.

IBM's AIX CICS/6000

IBM's Network System Division, based in Staines, England, develops, supports and distributes Customer Information Control System (CICS), the most widely-used OLTP monitor. Some estimates are that over 60 percent of the world's electronic transactions are being processed by CICS. The product is currently generating over $680 million annually in license and maintenance fees for IBM across thousands of sites. The pervasive market penetration of CICS is the envy of every system vendor and TP-system developer.

IBM's AIX CICS/6000 is an interface-compatible implementation of CICS functionality that runs on IBM RS/6000 platforms, with support for HP-UX and other vendors' Unix systems to follow. It provides support for open systems, client/server computing, and distributed transaction processing. (See *Open Information Systems*, Vol. 8, No. 3, March 1993, for more information about CICS/6000.) CICS/6000 is layered on top of the Open Software Foundation's (OSF's) Distributed Computing Environment (DCE). CICS/6000 supports standards-based computing and implements aspects of the X/Open Distributed Transaction Process model. Though leading the market with the CICS family of transaction systems, IBM is not resting on its laurels with CICS/6000. The company wants CICS to be the standard in open OLTP monitors. IBM is actively working to cultivate partnerships to this end. For example, Hewlett-Packard Company (Palo Alto, California) has licensed CICS and markets CICS/HP on both its 9000/HP-UX and 3000/MPE-ix platforms. Currently, CICS/6000 only operates on the RS/6000 and HP, but more partnership ports are expected in the near future.

IBM has constructed CICS/6000 over the OSF DCE and parts of Transarc's Encina Toolkit. The Encina Toolkit is a "transactional" augmentation of the OSF DCE. DCE allows CICS functional components to be distributed on multiple nodes in a "DCE cell." A highly-distributed implementation is outlined in Illustration 2.

The CICS/6000 implementation is a departure from the current CICS mainframe architecture. The CICS/6000 Application Server hosts a CICS monitor server process at startup, subsequently spawning the following functions:

- A recovery manager process responsible for recovery and restart operations for the node

- An applications manager process which, in turn, manages its "child" processes—dynamically "spawning" and "killing off" application server processes based on start-up configuration parameters and dynamic loads
- A transaction scheduler "daemon" process that receives requests from clients (local or distributed) and routes the request to an appropriate application server process

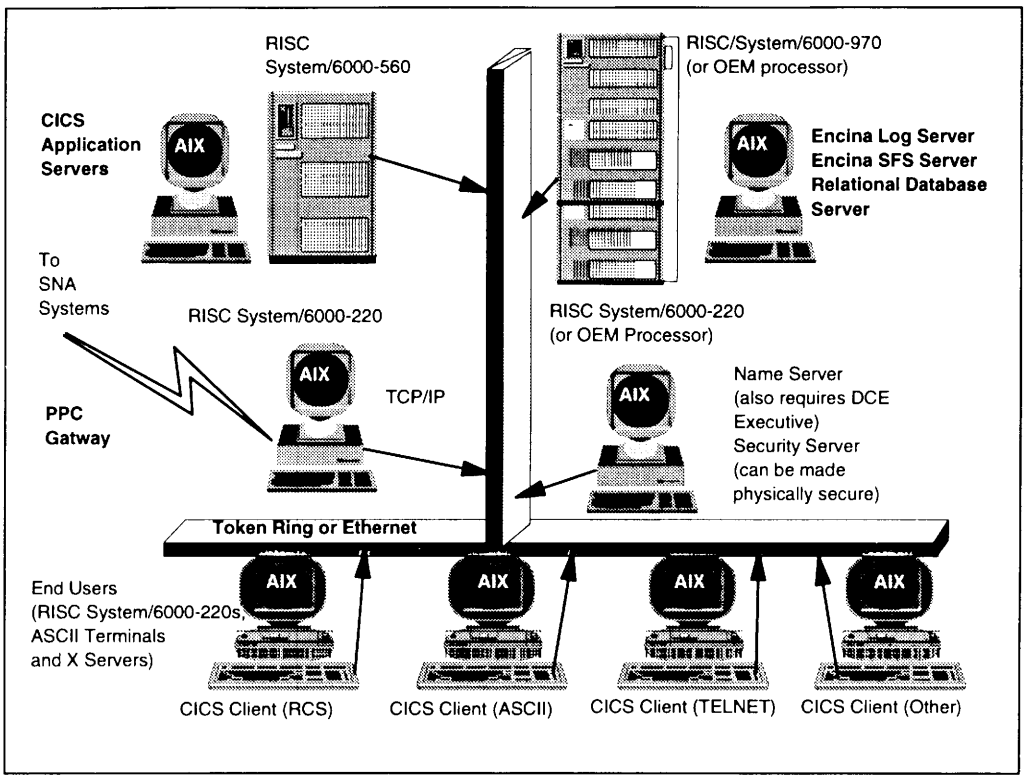


Illustration 2. CICS/6000 in a distributed environment. Clients may be terminals, PCs, or workstations. Communications may be either over Token-Ring or Ethernet with SNA gateways to the mainframe. CICS/6000 uses the DCE name server and security services.

Because of IBM's entrenchment within its base of mainframe CICS customers, CICS/6000 should win the day when it comes to market share for distributed transaction processing technology, but several caveats remain in the scenario. The OSF DCE is new technology, as are the Encina Toolkit and the new CICS/6000 API layer. CICS/6000's initial performance and stability expectations must be tempered, for it may take a generation or two of DCE, Encina, and CICS/6000 releases before all the bugs shake out and performance tuning becomes a reality.

Novell's Tuxedo System

The Tuxedo System, now a Novell product following the acquisition of Unix System Laboratories (USL), is in its fourth generation of product development. The Tuxedo System group now reports into the AppWare division of Novell, as opposed to the Unix System Group (USG). Initially developed by AT&T/USL for internal Unix-based network-switching and directory assistance applications, the Tuxedo System is now a widely-adopted, open, transaction processing system with over 30 original equipment manufacturer (OEM)

licensees. The Tuxedo System supports a distributed transaction processing model closely tracking standards work within the X/Open DTP. (See Illustration 3.)

Historically, USL was strictly a TP-system source code provider to OEM system vendors, systems integrators, and independent software vendors (ISVs). OEMs take the Tuxedo System source, customize it for a specific hardware/software environment, and possibly add value in the form of tools. Some of the OEM vendors market the Tuxedo System under a different company-specific program name (e.g., Open/OLTP, BOS/TP, etc.). A trademark program is in the works to help raise the visibility of the underlying Tuxedo System in these cases. A Tuxedo System logo will become a more common sight on marketing collateral.

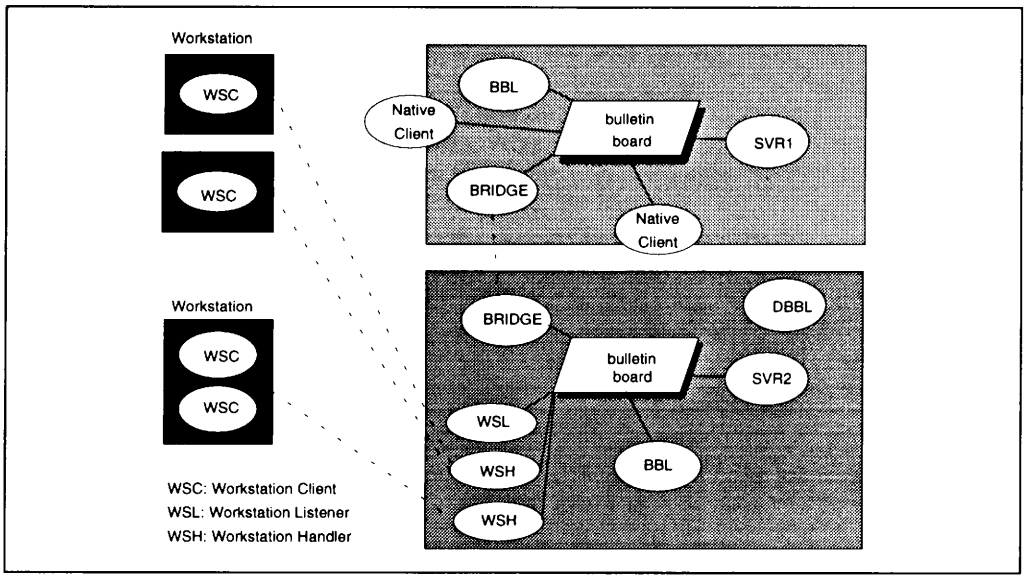


Illustration 3. Tuxedo System Process model. Server applications register service names and addresses. Client transaction applications send service request messages that are then routed according to the BB mapping information. A "BBL" system process is used to securely administer this critical shared-memory region.

Tuxedo Services

Tuxedo provides traditional transaction monitor services: client/server binding, distributed transaction coordination services, request scheduling, data logging and data recovery, and application program and resource manager interaction APIs. The core element within Tuxedo is the shared memory Bulletin Board (BB) name server. Server applications register service names and locations. Client transaction applications send service request messages, which are then routed according to the BB mapping information. Other system processes gather BB statistics to implement features such as dynamic throughput tuning and load-balancing across distributed nodes.

Novell's Opportunities

The acquisition of the Tuxedo System presents an interesting opportunity for Novell. An immediate consequence is the acceleration of a previously planned Tuxedo System NetWare Loadable Module (NLM). This is now a high priority for the Tuxedo System development organization. What remains to be seen is the positioning/utilization of the Tuxedo System within the AppWare strategy. AppWare comprises Novell's architectural plans for becoming a strategic platform for business-critical client/server application development, administration, and execution. This is a long way from Novell's network operating system heritage of providing file-sharing, mail, and print services. The Tuxedo System (like most other distributed transaction processing systems) is now being positioned not only as a

transaction system but also as a design center and execution engine for all types of critical, client/server application development.

There is significant overlap between the functionality and positioning of the Tuxedo System and the strategic direction of AppWare. Although the Tuxedo System is now part of AppWare, there has been no public positioning or definition of the role its technology will play in the AppWare architecture. The possibilities are there, and we assume Novell is in the process of sorting it all out. Obviously, a central position for the Tuxedo System within AppWare would be a tremendous boost for the technology. We'll just have to wait and see if this happens.

NCR's Top End

NCR's Top End is an open systems-based distributed transaction processing system closely aligned with the X/Open DTP model. Over the past year, NCR has released Top End 2.0 and has worked quietly and diligently to establish the product as a leading transaction processing system. With release 2.0, Top End now supports 3270 terminals, MS Windows environments, and enhanced network routing and workload features. Top End has also been "parallelized" and is generally available on NCR's strategic 3600 parallel processing platform.

Top End runs on several competing open systems platforms, including the Pyramid MIServer, HP 9000, Sun SPARC, AT&T StarServer, Unisys U Series, and IBM RS/6000. It implements a network-based distributed transaction model. Refined in previous generations of NCR OLTP technology, the message-passing architecture enables Top End to be ported to a variety of distributed system architectures, including the loosely coupled, massively parallel NCR 3600. Top End is designed around the X/Open DTP model, and NCR has extended this model to include a communication resource manager. NCR and allies are now working to formally move the current three-box X/Open model to a four-box model.

Top End Components

The fundamental Top End component is the Application Component. (See Illustration 4.) Application Components run as single (Unix) processes and are composed of application program logic linked to distributed transaction processing service libraries for transaction management, resource management, and client/server interaction (i.e., communications). Boundaries depicted within the Application Component represent interfaces to these service library routines and closely model the X/Open DTP application process model.

Increasingly, system vendors such as NCR are porting and/or licensing key software for implementation on competitive hardware platforms. The goal of these vendors is to establish a broad market presence for the critical software components while gaining account penetration in competitor's domains and establishing a nice licensing revenue stream. The theory is, "Where the software sale leads, perhaps the hardware sale will follow." NCR, IBM, and Digital Equipment are each employing this strategy (to a greater or lesser degree) within the transaction processing arena. Fortunately for the customer, it's a win-win situation: Choose a TP-system and choose your platform; one choice doesn't lock in the other. For NCR as a vendor, this type of licensing broadens market opportunity, demonstrates a commitment to openness, and gives confidence to users.

So far, the strategy appears to be paying off. Over the past year, Top End's deployment has been greatly accelerated. In most instances, both NCR and non-NCR platforms have been involved, with Top End support for "foreign" platforms a primary prerequisite.

NCR's Top End

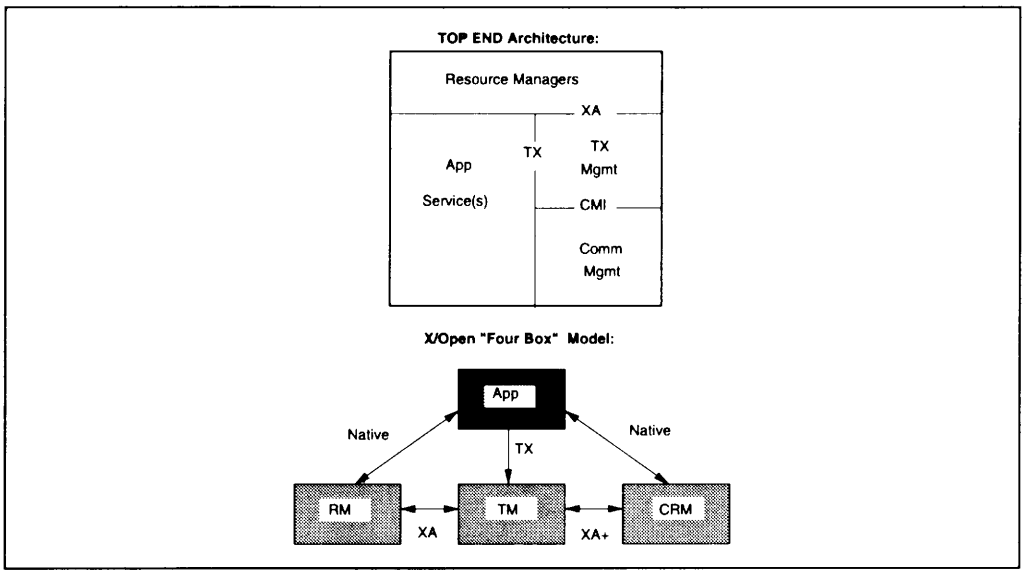


Illustration 4. Top End and the X/Open DTP model. The services within Top End closely parallel the architecture of the X/Open model and extend the model with a communication resource manager.

Tuxedo and Top End

Before Novell's acquisition of USL, AT&T was in the awkward position of having two competing transaction monitor products in its stable. NCR's choice to stick with Top End over USL's Tuxedo was tough to explain. The two technologies have many similarities. Both are Unix-based, X/Open XTP compliant, and non-CICS compatible. However, Top End seems to have better distributed system management features than Tuxedo, and, while Tuxedo uses a centralized bulletin board to manage distributed transactions, Top End uses a more distributed model. On the other hand, Tuxedo has far more OEMs and wider distribution than Top End and more management and development tools as well. However, Top End supports more transaction models.

The greatest boost to Top End's success may be changes in positioning that Novell introduces for Tuxedo—tying it closer to UnixWare, for instance. Although Tuxedo OEMs would rather not have to think about changing directions, Clean Layering would work well for them if they should have to replace Tuxedo with something else, like Top End.

Transarc's Encina

Transarc Corporation formally announced the Encina family of distributed transaction processing products in September 1991. The products became generally available in late 1992, and 1993 witnessed the continued roll-out of the platform amid growing industry support for Transarc's technology. Encina is noted for its leverage and augmentation of the OSF DCE. Transarc has developed several tiers of TP-system tools and services layered over DCE's services, including a TP monitor, index file system, client and server application development libraries, logging and recovery services, and a host of integrated components comprising a complete DCE-based transaction processing environment. Encina is also noted for its prominent OEM licensees, including IBM, HP, Stratus Computer (Marlboro, Massachusetts), and NEC Corporation (Boxborough, Massachusetts).

Transarc's Encina

Encina Architecture

Transarc adds layers of distributed transaction and client/server application development tools and services to an OSF DCE network underpinning. Each component is designed as a service with well-defined, accessible application programming interfaces. This allows application designers to select the services needed to develop a broad range of network applications up to and including high-availability, high-security distributed transaction processing applications.

Transarc has extended the OSF DCE remote procedure call (RPC) to be transactional. With this key technology, programmers can use the DCE RPC facility to invoke transaction services on network servers. Transarc's approach piggybacks all of the needed transaction processing state over the client/server "transactional" RPC connection. Importantly, particularly for commercial transaction processing, Transarc provides security all the way down to the RPC level by utilizing the automatic authentication and privacy inherent in the DCE RPC. In fact, Encina is built directly on DCE; other implementations are sort of bolted on. Encina uses the full DCE product, including name services, threads, and security services. Other implementations tend to use just the RPC.

The Encina product family comprises two tiers: the Encina Toolkit and the Encina Extended Services. See Illustration 5. The toolkit is composed of integrated, low-level services required to support transactional computation in an OSF DCE environment. The Extended Services are built on top of this layer and support communications, resource manager, and execution environment (i.e., TP-monitor) components.

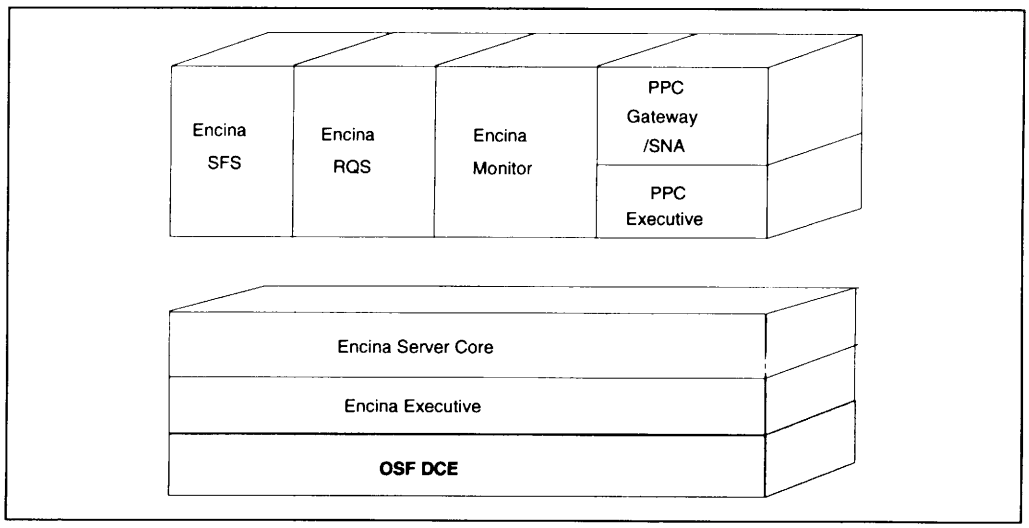


Illustration 5. The Encina product family is made up of layered services built over OSF's DCE. These services include the POSIX-compliant Encina Executive; Encina Server core, which provides transactional integrity; Encina OLTP monitor; Encina Structure File system (SFS); Encina Recoverable Queuing Service (RQS); and Peer-to-Peer Communication (PPC) services.

Transarc's Market Position

Transarc carries a lot of weight for a small, privately-held, start-up company. The principal reason for this can be spelled out with three familiar letters: I-B-M. Not only are IBM salespeople marketing Encina, but also IBM's own strategic CICS/6000 was developed using select parts of Transarc's layered Encina Toolkit technology. Of relevance here is the credibility bestowed upon Transarc's Encina, otherwise an as-yet-unproven technology. IBM's use of the toolkit as a key component of its strategic CICS direction catapulted Transarc into prominence.

Digital's ACMS

Digital Equipment Corporation has restructured to become a more tightly focused, customer-driven company. The organizational structure of the new Digital centers around individual business units, with six of its nine named units focusing on particular vertical market industries with worldwide profit-and-loss responsibilities. Complementing the organizational restructuring is a product development and marketing strategy built around "frameworks." Currently, Digital's six strategic frameworks comprise groups of related products implementing a particular computing paradigm and/or business function. One of the six, the Production Systems framework, incorporates the interrelated areas of transaction processing, reliable messaging, database and database integration, and storage management.

ACMS Is Application Friendly

Digital's Application Control and Management (ACMS) is evolving to meet the company's strategic direction. ACMS is characterized by its inherent support for modular, highly-structured transaction application designs. Integral to this model is Digital's unique Structured Transaction Definition Language (STDL), which provides constructs for defining the "workflow" of a transaction application. A new version of ACMS, ACMSxp, is fully integrated with the OSF DCE. Although Digital has chosen to implement its own transactional semantics and its own transactional RPC, we do expect that Digital's transactional RPC and Transarc Encina's transactional RPC will be compatible in the future.

In the ACMSxp design model, an application is composed of a set of discrete tasks. Each task may require the execution of a series of steps, either sequentially or with conditional branching. ACMSxp formalizes this hierarchy by providing an environment and the specialized STDL to define the applications, the associated tasks, and the discrete callable steps executed within the task.

STDL is a structured, TP-specific language for defining transaction applications, and it constitutes the ACMSxp API. The STDL programming model, shown in Illustration 6, separates the TP-computation into three parts:

1. Tasks that describe the transaction demarcation/semantics, the overall execution flow, and exception-handling conditions
2. Front-end presentation service procedures
3. The back-end application-logic server procedures providing transactional access/modification to shared resources

ACMS Is Strategic

ACMS will continue to play a key role in Digital's transaction processing future. The development priorities for ACMSxp include ports to Digital's three strategic platforms: OpenVMS, OSF/1, and Windows NT on Alpha AXP platforms. Though ACMS will remain strategic to Digital, the company will also support production system frameworks centered on other transaction processing systems.

Micro Focus's MTS

Micro Focus Group PLC (Newbury, England) was founded in 1976 and is most widely recognized for its Cobol Workbench for DOS and OS/2. In 1986, Micro Focus made a leap—from straight Cobol to CICS Cobol—with the CICS Option for the Cobol Workbench. The CICS Option provides an integrated development environment for mainframe online transaction processing applications by "emulating" the CICS environment on a desktop system.

Micro Focus's MTS

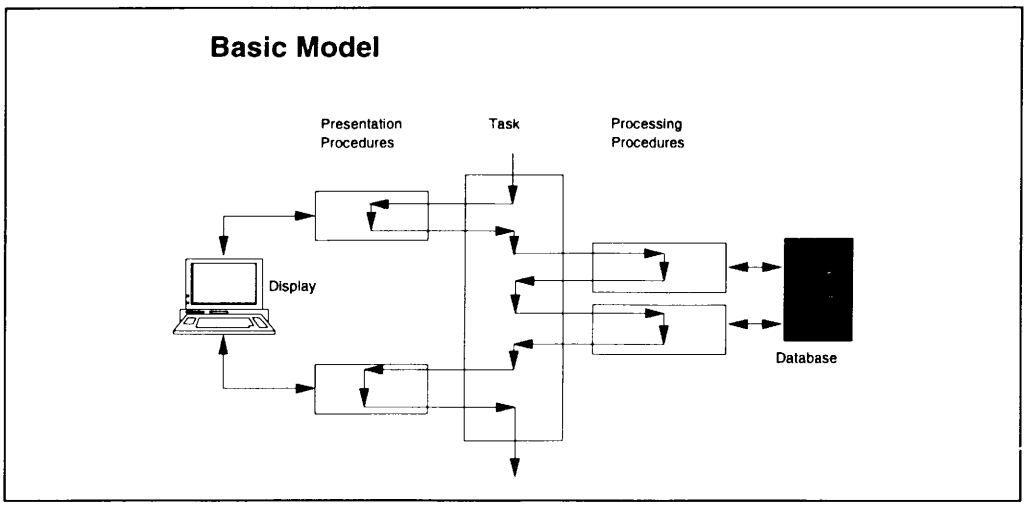


Illustration 6. The STDL programming model separates transaction processing into three components. Tasks execute application logic, communicating with presentation procedures on the client side and with processing procedures on the database side. Processing procedures handle transactional access to shared resources.

MTS Provides CICS Compatibility

Now, Micro Focus is making another leap with the Micro Focus Transaction System (MTS). The MTS takes Micro Focus from the development/debugging offloading realm to full production application execution—a CICS-compatible, command-level transaction processing system. Though a new product, MTS is already generally available on the RS/6000 and SCO/Intel platforms and runs under Unix, OS/2, and DOS. Micro Focus has implemented CICS "cooperative processing" functions as defined by the CICS Inter-Systems Communications (ISC) APIs and should be able to reside in large CICS networks as a fully participating member. ISC represents a set of features supported by IBM's CICS transaction system to provide for client/server and distributed transaction processing across multiple CICS regions and nodes. Support for ISC enables support for the IBM distributed model. MTS is not layered over the OSF DCE, as is IBM's CICS/6000, which is either good, bad, or neutral, depending on one's perspective. Micro Focus believes that, in the near-term, avoiding DCE has performance advantages.

Micro Focus is now a full system software vendor after years of playing the role of premiere CICS Cobol development "offloading" vendor and IBM partner. In addition to running on Unix, OS/2, and DOS platforms, MTS is fully integrated with the Micro Focus Cobol Workbench/CICS Option and the Animator symbolic debugger. The battle between IBM's CICS/6000 with DCE/Encina and Micro Focus's MTS, which has none of these dependencies, should prove to be an interesting struggle.

Integris's UniKix

UniKix is a CICS-compatible transaction processing system developed and marketed by the Integris division of Groupe Bull. Integris is a systems integration organization focusing on networking and "Smartsizing" (their trademarked term) mainframe applications, i.e., migrating existing CICS applications to Unix hosts using UniKix. UniKix Release 4.0 is the current product iteration. Basically, UniKix supports the execution of mainframe-developed CICS Cobol VSAM and DB2 programs using the Micro Focus Cobol compiler and runtime system. The mainframe code is first translated (parsing the command-level CICS directives) and then compiled with Micro Focus tools.

Integris's UniKix

A UniKix system consists of various components:

- UniKix Main Program
- UniKix Terminal Processor
- UniKix Transaction Processor
- UniKix Start Processor
- UniKix Recovery Processor
- UniKix Debug Processor
- UniKix Print Processor
- Various UniKix Utility Processors

Application Development under UniKix

The Application Development Environment ties all other UniKix utilities together as well as various other functions (e.g., the editor, compiler, and so on) to provide a familiar environment for programmers coming from a mainframe world. From an input Cobol program, the Command Language Translator translates any EXEC CICS statements into a series of Cobol statements and a call to a UniKix interface routine. The interface routine then transfers control to the proper UniKix processing routine.

UniKix Differentiation

Of course, Integris believes that its CICS implementation has particular merit, especially when compared to those of some of the other CICS vendors, most notably IBM. The primary distinction is simplicity. If a user has CICS Cobol/VSAM and DB2 applications (and it's probably easier to count the large system users that don't) and wants to "re-host" the applications in order to save money, then, by Integris's thinking, UniKix is the right product. Integris is quick to point out that IBM's AIX CICS/6000 has skill prerequisites and network infrastructures (Unix, DCE, TCP/IP, and multiple workstation processors) that many shops don't—and won't—have in the short term. Thus, UniKix enters the scene, offering IS professionals a simple, straightforward solution. This, to some degree, is also the position taken by Micro Focus with MTS.

VISystems' VIS/TP

VISystems' VIS/TP is an IBM CICS-compatible transaction system environment. VIS/TP is also positioned to facilitate CICS application migration to a variety of Unix-based platforms. The product has implemented some of the mainframe CICS interoperability and distributed transaction processing features (i.e., ISC). The VIS/TP product has been used in production applications for four years and now boasts some high-profile success stories.

CICS Execution on Unix

VIS/TP is a collection of products providing a CICS application execution environment on a variety of open system platforms. VIS/TP supports translation/recompilation of command-level CICS applications, the migration of mainframe VSAM data sets, and interoperability with mainframe CICS systems.

The VIS/TP application migration model provides for the translation and conversion of CICS/Cobol and VSAM data sets, respectively. CICS Cobol source code is run through the VIS/CICS translator/precompiler which, in turn, emits structured C code. The C code can then be compiled using standard C compilers for the target platform. The C code is linked with VIS/Executive libraries supplying the runtime execution environment for transactions. Similarly, VSAM files are run through a conversion filter to produce a "VIS/VSAM" file. Basic Mapping Support (BMS) screen definitions are handled by the VIS/Screen Maker utility. This utility can translate BMS to VIS screen source for interactive modification and then export either BMS code (for uploading to the mainframe) or VIS definitions for use in the VIS/TP execution environment.

With these utilities, the CICS/Cobol-to-VIS/TP execution environment conversion is relatively straightforward. Essentially, a command-level CICS application with VSAM data

VISystems' VIS/TP

access calls and BMS macro-defined screens can be translated to the VIS/TP environment with few problems.

Target Markets and Partners

VISystems focuses its direct sales activity on two areas: Fortune 500 companies that are downsizing applications and ISVs targeting Fortune 2000 companies with low-cost transaction delivery engines. The company has marketing relationships with Hewlett-Packard, Digital, Data General Corporation (Westboro, Massachusetts), and Motorola (Tempe, Arizona). The vendor relationships will gain in importance as the dowhsizing market heats up and systems vendors realize that VIS/TP's software tools will help them sell hardware into IBM accounts, Of course, those same vendors can currently license CICS/6000 from IBM or bring in Micro Focus. The CICS world is getting very crowded.

Gresham Telecomputing's tp+

Gresham Telecomputing PLC has expertise in transaction processing, client/server database processing, and open systems computing. The company develops and markets the TPS and tp+ transaction processing environments. The tp+ product is being used by Dun & Bradstreet Software (Framingham, Massachusetts) as an enabling technology for the Unix implementation of its current terminal-based "Millennium" financial application products. In addition to tp+, Gresham has several other interesting TP-related products that it licenses to users and other TP-system vendors. These products include ISAM-XA, an X/Open DTP XA-compliant ISAM file manager; TP Session Server, a transaction session/context manager built over ISAM-XA; and DataServe, an RPC service call mechanism for OSI network client/server computing.

Straightforward Approach

The tp+ product provides a very simple, straightforward, nondistributed TP system for mainframe application conversion/migration to Unix platforms. Transaction applications are written in Cobol or C. Architecturally, each terminal/user is allocated a process that executes the complete transaction, linking in all the terminal-handling, application, file manager, and system code required to execute transactions. Application code is implemented as read-only code that is loaded once in memory and shared among all processes in the system. The tp+ monitor code supports transaction control mechanisms, such as commit and abort. Each concurrent user need not have identical application process copies. Several different application programs with different sets or mixes of services can be supported.

Positioning—Access to ISAM Files

Gresham's tp+ is specifically positioned as a migration and/or porting tool for mainframe-based, nondistributed transaction applications accessing a single ISAM file system. Applications implemented on this model can be easily migrated to tp+. The advantage of tp+ is that it is a very lightweight monitor that uses minimal resources, an important factor in smaller installations. In addition, the Dun & Bradstreet relationship makes Gresham the only Unix monitor company, other than IBM, with a major ISV customer.

Allinson-Ross's TIP/ix

Allinson-Ross offers TIP/ix, an OLTP system derived from a Unix implementation of TIP/30. a transaction processing environment originally developed for the Unisys/Sperry System 80 hardware platform and its associated OS/3 operating system. TIP/ix also supports applications written to the Unisys IMS transaction environment (also System 80 based) and the Unisys-supplied IMS/1100 and TIP/1100 for the 1100/2200 environment.

TIP/ix first appeared in December of 1991. This release is a single-system implementation without distributed transaction processing capabilities. The system is essentially a port of the Unisys System 80-based (OS/3 operating system) TIP/30 product, which supports a native API and the Unisys IMS API. A recompile and migration strategy is supported.

Allinson-Ross's TIP/ix

A second release, scheduled to go out for beta testing in January 1994, will add some distributed processing functionality, as well as interoperability features with existing System 80 TIP/30 installations. Distributed processing will include a transaction-routing feature to allow transparent transaction execution on heterogeneous Unix platform implementations of TIP/ix. TCP/IP will be the transport. Function shipping, the ability to connect to remote data stores as if they are local, will also be supported. A new System 80 interoperability feature will allow remote login from a TIP/ix Unix platform. Two other major additions to the API in V.2 include a peer-to-peer API for establishing sessions/conversations across TIP/ix applications and a reliable store-and-forward message queue system. Allinson-Ross plans to add more general open systems DTP functionality in future releases. This level of open systems functionality will begin to emerge in 1994.

Allinson-Ross's product is of interest primarily to Unisys System 80 customers. Although Unisys has a strategy for taking care of these customers as they migrate to Unix, some customers will find Alinson-Ross's approach more efficient and cost effective.

Summary

Client/server application development/deployment has gotten off on the wrong foot. The prevalent remote data access model of client/server suffers from integrity problems, and, from all indications, will have serious problems scaling upward. The push by database vendors to facilitate newly-developed or rewritten applications as database stored procedures is also problematic. Stored procedures offer a primitive execution environment, and they limit flexibility for application design. In addition, in many transaction-oriented applications, the database is already the bottleneck. Further burdening the database execution engine with application processing responsibility is not the solution.

The current generation of transaction processing systems supports client/server application development with a (logical) three-tier model. Desktop processing functions include presentation, input collection/validation, and business function invocation. The application server tier—within the execution domain of a TP monitor—implements modular business functions, which, in turn, operate on the database resource managers. Partitioning applications in this manner greatly enhances performance and scalability. In addition, the highly functional execution environment supports load-balancing, failover, and dynamic performance-tuning attributes—attributes that will be required if organizations want to get the most out of their distributed computing resources.

Well over a dozen Unix-based (and multi-OS) transaction processing systems are available today. Most are converging with key open systems industry standards for distributed computing and transaction processing.

James H. Johnson is the chairman of the Standish Group International, and David Hudson is a director. The Standish Group publishes the Annual Open OLTP Monitor report, currently in the 1993 edition. The Standish Group International Incorporated also provides market research and education on open systems and online transaction processing products and services.

Next month's *Open Information Systems* will address
OSF's DCE: Application Models

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: CONSORTIA

OSF Forges Ahead

With Unix International officially closing its doors at the end of the year, the Open Software Foundation (OSF) appears to be the survivor of the Unix wars. In reality, it has been the consortium's work in user interface and distributed computing, not OSF/1, that has provided the most benefit and has had the most far-reaching impact on the industry.

DCE, DME Advance

At times, it seems that OSF is destined to move the proverbial two steps forward, one step back in its efforts to promulgate its distributed technology throughout the industry. Steps forward receive less media attention, lending support to the perception that no progress is being made. The reality, however, is that OSF is making inexorable progress toward key objectives, both through its own actions and as a beneficiary of actions by others in the industry.

DCE Benefits from Joint Development. A recent joint announcement by Microsoft Corporation (Redmond, Washington) and Digital Equipment Corporation (Maynard, Massachusetts) of their intention to bring their distributed object strategies into synchronization is actually very good news for the OSF in general and for OSF's Distributed Computing Environment (DCE) in particular. The Common Object Model will assure interoperability between software built using Microsoft's Object Linking and Embedding (OLE) 2.0 and software built using Digital's DCE-based ObjectBroker with its Common Object Request Broker Architecture (CORBA)-compliant interfaces.

Although Microsoft does not license OSF code for its remote procedure call (RPC) mechanism, the RPC it is using in the current release of NT and in the next object-oriented release, code-named Cairo, is compatible with the DCE RPC. Little wonder, since the person at Microsoft responsible for its distributed computing efforts, Paul Leach, was one of the original developers of the Apollo Network Computing System (NCS), which provided significant genetic material to DCE, including the NCS RPC, which provided the foundation for the DCE RPC.

DCE Adoption. DCE's usefulness is starting to become evident in work being done by corporate and third-party developers on both applications and tools. DCE users are starting to emerge, including Citicorp (trading system), Charles Schwab (order entry), Lawrence Livermore Labs, and Jet Propulsion Labs in the United States; Secom in Japan (facilities, as in buildings, security); Barclay's Bank in the United Kingdom (trading system); and Hewlett-Packard Company (worldwide order entry system).

DCE Interoperability. Customers will be implementing open information systems with components from different vendors. Therefore, ensuring interoperability among DCE implementations is a critical requirement. The OSF is putting in place a program to demonstrate interoperability along the lines of its Challenge '93, and it plans to do more testing in 1994. (See *Open Information Systems*, Vol. 8, No. 7, for more information on Challenge '93.) We would like to see more formalized conformance-testing to ensure that application programming interfaces (APIs) are correctly supported and executed. Instead, OSF will certify that a vendor's implementation of DCE has passed a set of tests that OSF publishes and that it has successfully completed interoperability tests in the OSF lab setting. Last year, the lab tests were a one-time deal in conjunction with Challenge '93. OSF now plans to establish a somewhat scaled-back lab, with 5 to 10 systems instead of 17, operating year-round to test actual vendor DCE products.

At present, although the DCE specifications are complete, but the validation test suites for all of the components are not. Version 1.0.1 of the Validation Test Suites (VTS) for the DCE RPC will be available in January 1994. OSF plans to make available additional validation tests for the directory, security, time, and distributed file system (DFS). The consortium has plans for certification of DCE implementations in the short term, and for DCE-based applications in the long term, but no plans for DCE branding. The issues involved in branding are too complicated, particularly from a lega perspective. Ultimately, we need to be able to test applications that use DCE services to ensure interoperability.

Enhancements for Future Releases of DCE

DCE is not standing still, and work is progressing on the next release. Being added is an interface to allow nonsecure environments to interoperate with secure DCE environments. Additional extensions are being evaluated to allow DCE to interoperate with RDBMSs, CORBA, transaction processing (TP), and other environments. These will be areas in which value will be added to DCE through a number of different mechanisms. For example, HP has proposed extensions to the DCE security model for CORBA environments, online transaction processing (OLTP) extensions that will probably be adopted by the OFS. In the TP area, convergence of the work of a number of transaction monitor suppliers, including Transarc Corporation (Pittsburgh, Pennsylvania) with Encina, NCR/AT&T (Dayton, Ohio) with Top End, and Novell Corporation's (Provo, Utah) Tuxedo, will promote DCE as a foundation for distributed OLTP. Transarc's Encina is currently the most DCE compliant. The others have further to go but will all make the trek.

Version 1.03 Maintenance. Version 1.03 of DCE, available in December 1993, is a maintenance release with no new functionality added. It concentrates on providing an industrial-strength distributed file system and improving the robustness of the other components. Having a solid DFS is critical for interoperability among Unix suppliers, since, to date, each has had to implement DCE using its own file system. It is less meaningful for other operating system platforms because integrating DFS with non-Unix kernels is more of a challenge.

Major Update in Release 1.1. Release 1.1 of DCE, slated for late 1994, will have significant reliability, availability, and serviceability enhancements, including auditing and accounting features. This release will have faster, tighter code, an improved runtime environment, and improved serviceability. It will provide better load management and connectivity management within the RPC mechanism. Also to be included are replication features that will improve security and additional directory services to provide increased availability. Release 1.1 will also include serviceability instrumentation so that performance monitors can be built. Improved internationalization is on the docket as well in the form of double-byte character-set support in the naming services. Consistent administration functions will be included along with a single, consistent, more forgiving environment for administering DCE services. An audit facility is being integrated that will track any events in the environment.

All of these improvements will be compatible with existing DCE development work. We believe that Release 1.1 will go a long way toward answering many of the objections the industry has had to DCE in its current form.

Although DCE implementations have been appearing mostly on Unix platforms so far, they will become available next year on proprietary platforms, including MVS, AS/400, HP MPE/ix, Hitachi, Bull GCOS 7 and 8, and SNI BS2000. (Next month's feature will focus on how DCE is being used in building distributed applications.)

A Different Sort of DME Arrives

Rumors of the Distributed management Environment's (DME's) death have been exaggerated—not greatly exaggerated, but exaggerated nonetheless. What OSF has finally shipped in DME 1.0 isn't exactly what DME was supposed to look like when the results of the DME request for technology (RFT) were made public three years ago. That can largely be traced to OSF's original decision to go with object-based management and events in the object-oriented segment of the industry.

DME 1.0 Released. DME 1.0 had previously been available to OSF members as a part of the OSF early access program, and it is now generally available for licensing. It supplies a set of distributed services, including:

- **Software distribution (SDS) and license management services (LMS)**, based on technology from HP, which enable the distribution and tracking of application software to be managed in distributed environments.
- **PC management services**, which tie PCs into license management and event management services using agent technology developed by Gradient Technologies (Marlboro, Massachusetts). This component is designed to run on systems that are not running DCE.
- **Event management services**, based on Event Logger technology sold to Banyan Systems Incorporated (Westboro, Massachusetts) by Wang, which monitor and track significant events at any supported node in a distributed environment for purposes of systems management, troubleshooting, and remote problem resolution.
- **Subsystems management services (SMS)**, which is the management component of DCE for monitoring and controlling its various components. As such, SMS does not use DCE services itself. DES notifies SMS. DCE environment is sensitive with regard to the order in which servers go up or down. SMS

defines the order of shutdown and start-up. The basic technology in SMS comes from IBM.

Shift from Original Plan. OSF's original plan for DME was for an object-oriented framework on which management services and applications could be built. The original selection of the Tivoli Systems Incorporated's (Austin, Texas) WizDom (which later became Tivoli Management Environment, or TME) was aimed at providing an object framework to serve as a platform for integrating system and network management with a combined API. That selection preceded the completion of the work done by the Object Management Group (OMG) on the CORBA 1.1 specification for object request brokers (ORBs).

After OMG's specification was announced, key OSF members made it clear that their companies' directions were to support the CORBA specification and that the OSF would have to have a CORBA-compliant object request broker in DME. The Tivoli TME's object request broker was not CORBA compliant at the time, but probably a more important consideration was that the individual vendors did not want OSF in the ORB business, particularly when interoperability had not yet been addressed in the CORBA specification.

DME was restructured to focus more on using the lower-level distributed services of DCE and defining management services APIs above the ORB. Tivoli's ORB was removed (probably to Tivoli's relief). OSF and Tivoli continue cooperating to define the upper layer API for management services. This is the work that has been submitted to X/Open.

DME in Three Phases. DME 1.0 is the first of three phases of work the OSF will do. It is a set of services with APIs for applications to use those services. Phase two is termed the Network Management Option. It represents an implementation of the X/Open XMP standard. Work on this is being done by Groupe Bull (Paris, France) along with Hewlett-Packard Company (Palo Alto, California), which is providing CMIP and SNMP support, and Siemens-Nixdorf Informationssysteme (SNI, Germany), which is doing X.500 for CMIP while HP integrates the SNI work into the whole.

The goal of phase two is to have a single consistent implementation to meet requirements that are being proposed to the OSF, particularly by the telecommunications industry. This work is scheduled for release in mid-1994, although there may be some earlier release of code before then under controlled circumstances.

Critical Phase Three. The third phase of the DME work actually consists of two stages. The first is the support mentioned above that will be added to DCE to better support CORBA 1.1-compliant ORBs. The approach that OSF has taken is to provide better support for the ORB environment by supplying services that are in CORBA but are not currently in DCE. These include dynamic invocation interfaces, adding capabilities like persistence (object storage) services, and others. The DCE Interface Definition Language (IDL) also will be enhanced so that it can more easily map to the OMG CORBA IDL.

In addition, late in 1994, the OSF will release a set of lower-level management services that will run on any CORBA 1.1 ORB to provide maps, collections, display services, discovery, etc. It will also provide an API that will ensure portability.

The second stage of phase three is the development of object-based management services, the APIs that will constitute the interfaces for development of DME management applications. These two stages should both be complete by early 1995 and will provide the foundation on which management applications, working through vendor-supplied ORBs, can be delivered. OSF no longer has plans to provide an ORB. All the major vendors have their own ORBs, which don't interoperate right now anyway, so "Why complicate things further?" figures OSF.

OSF Submission to OMG. OSF is submitting the APIs for the future enhanced DCE services to the OMG as a part of the CORBA 2.0 RFT. Its objective is to have them incorporated as the transport protocol and services required for interoperable ORBs. Under this proposal, DCE would be the standard way to communicate across ORBs. This is likely to stir up some controversy within the OMG, since some groups there think all that is required is an RPC, others want name and security services, and still others want no part of DCE (or ONC).

Tivoli Submission to X/Open. In the meantime, Tivoli has submitted the APIs that comprise its Management Services to the X/Open System Management Working Group. This submission has been reviewed by many industry groups, including the OSF, which supports a consistent Management Services Interface Specification.

Goal of DME Unchanged. The ultimate goals for DME have not changed: to provide a platform for applications for integrated system and network management for both DCE systems and systems that are not DCE based, and also to support different networks. Much of DME's work is aimed at providing better integration with DCE, but the scope of DME remains broader than simply managing DCE.

OSF Survival A Matter of Evolution

Although the trade press often speculates about the long-term viability of the OSF, as a member-driven organization, the consortium will exist as long as it serves a purpose for its members. Today, that membership has reached a level where 60 percent are user organizations. Those companies have a different agenda than the vendors who founded the OSF, and they are likely to drive their agenda into the forefront of OSF's work. With Unix International now defunct and the Unix wars officially over, OSF has an important role to play in the management of industry processes that result in the availability of interoperable, distributed computing architectures.

OSF has an unexpected "partner" in that pursuit in the form of Microsoft. With DCE components lurking around the integration of Microsoft's OLE and Digital's Object Broker, the OSF ends up benefiting from the partnership. The DCE's RPC and IDL are key parts of the Microsoft-Digital effort, although they are not implemented directly. Microsoft's DCE has common formats and interfaces, although it does not use OSF code. Microsoft's IDL is the DCE IDL with a few extensions. We believe that interoperability with Microsoft is critical for open systems. With Digital playing an intermediary role, OSF technologies begin to play an important part in the scenario of open, distributed computing environments bridging open standards with de facto standards.

OSF survival is not a question. How its role will evolve is very much a matter of speculation. — *M. Goulde*

FOCUS: INDUSTRY

X Consortium Goes Private (Sector)

Ten years ago, computer scientists at the Massachusetts Institute of Technology (MIT, Cambridge, Massachusetts) embarked on an ambitious computing experiment called Project Athena. The goal was to see how computing equipment, dispersed throughout the campus, would influence the educational environment.

To achieve this goal, Project Athena developers focused on creating a coherent set of software and a common platform with which engineers could build the educational software to run across this distributed network. The X Window System, one of the by-products of the Project Athena experiment, became a licensed technology in 1985. Since then, it has helped research institutions and various companies implement networked computing across disparate computing platforms.

Under New Management

Having grown the X Window System and watched it mature as a windowing system for Unix, the MIT X Consortium decided this past September to spin off the organization from the MIT campus and give it a new name: the X Consortium. In the beginning, MIT was not comfortable running the consortium but agreed to house it to help foster its growth as an independent organization. The X Consortium relocated itself and named Luther Able as the new director. Abel, known in the Unix industry as a former president and CEO of Unix System Laboratories, takes the helm from the departing Bob Scheifler.

Filling the vacant shoes of Scheifler will be no easy task. Scheifler not only had a good technical grasp of the technological issues but also of the politics involved in maintaining the consortium as a vendor-neutral organization.

A New Agenda

In April 1994, the X Consortium will begin distributing X11 Release 6 (R6), the newest release of the X Window System. Like any technology that has matured, X11 R6 features several new capabilities important to today's users and markets.

Some of the new features found in R6 include a low-bandwidth protocol (LBX) to support asynchronous serial links, an imaging extension (XIE) to handle various types of imaging applications, object-oriented technology (Fresco) designed to bring greater ease of use and interoperability with other operating environments, multithreading to support multiprocessing machines, and other general features, like session management.

High on the consortium's agenda for the next year is addressing the issues posed by the growing interest in the commercial marketplace among MIS departments as well as making sure that shortcomings of the X Window system are addressed in future releases. The X Consortium believes that some of the new features in X11 R6 will help with the transition to a more market-driven entity that is sensitive to the needs of the commercial marketplace.

Overall, the rich set of new features embodied in R6 are expected to strengthen X and bring it more in line with the features and functions found in other windowing environments, like Microsoft Windows and the Macintosh.

LBX: Asynchronous Communication

Today, many X users are limited to running X over Ethernet and other high bandwidth networks. This poses problems for users who want to connect to their X environment from a remote site over a telephone line, perhaps to read their electronic mail. That's where LBX comes into the picture.

Today, users can employ Serial Line Internet Protocol (SLIP) or XRemote to allow this type of remote asynchronous communication. However, these de facto standards lag in the area of performance. LBX is designed to improve the overall performance of X over serial lines by cutting down on the network communication that occurs between the client and the server. Basically, LBX provides an intermediate site, called the proxy, to cache information such as window properties, keyboard and mapping tables, and font metric properties.

When implemented, LBX allows multiple clients to be linked to the proxy via a high-bandwidth link while the proxy and server are linked via a low-bandwidth link. In this way, application requests are kept local and travel from the client to the proxy and back without having to traverse the entire length of the wire between the client and the server. Additional efficiency is achieved with data-compression algorithms that help reduce the number of bytes sent across the wire.

The objective of LBX is to allow users to enjoy good performance when connected to a host machine with a 9600-baud modem connection. LBX runs over any underlying network transport protocol, such as PPP, CSLIP, DDCMP, and TCP. In addition, LBX can coexist with other protocols, such as telnet, NFS, and font service.

Multithreading

X Window users have been trying to find a way to harness the processing power offered by multiprocessor machines. However, until this release, users could not benefit from multiple CPU machines because the X server code has been single threaded, thus allowing the execution of only one task at a time.

In order to take advantage of multiprocessor machines, R6 will include support for multithreading. Simply, multithreading allows for interrupts. A user, for example, can be reading mail in one window while a complex 3D image is being drawn in another window. Users will also benefit from R6's multithreading by gaining greater interactivity with their applications. Multithreading will become especially important to computation-intensive applications in graphics and imaging that require multiple CPU processing power.

Imaging Comes to X

Another important piece of technology that has been missing from X since its creation is imaging. With the release of R6, users will finally be able to work with images and create applications that can exploit such elements as contrast, sharpness, zoom, filtering, mixing, dithering, and color space conversions.

These imaging capabilities will surely accelerate development of imaging applications for such markets as banking, insurance, telecommunications, oil, and medicine. Users and developers will be able to rotate, scale, compress, and decompress images, thereby reducing the processing burden of the host system. The imaging extension also supports compressed image transport, such as G3/G4 fax, Joint Photographics Expert Group (JPEG), and Tagged Image File Format (TIFF).

For those users who don't need the complexity and size of a full-fledged imaging extension, R6 will provide a Document Image Subset (DIS) designed for bitonal images, such as scanned paperwork and faxes.

In the past, some X terminal vendors, including Network Computing Devices Incorporated (NCD, Mountainview, California) and Digital Equipment Corporation have implemented bitonal support in X servers. The X Consortium and X terminal vendors estimate that 80 percent of the X users require only bitonal imaging support.

Enhanced Interapplication Communication

Perhaps one of the most important enhancements found in the new release is its support for cut-and-paste between windows. For some time, users have been anxious to have the same cut-and-paste capabilities found in the Microsoft (MS) Windows and Apple Macintosh (Mac) environments.

In the previous version of X11 R5, there was support for cut-and-paste between windows. However, the new release provides cut-and-paste along with drag-and-drop between different windowing environments, such as the Mac and MS Windows.

As Microsoft's Object Linking and Embedding (OLE) technology gains momentum on the desktop, X users are coming to expect R6 to permit some degree of object embedding. Users, for example, will be able to display both a 3D solid model and a wire frame model of a car in the same window simultaneously. Using this capability,

users can access multiple views of the same data at the same time.

Another general feature implemented in R6 is disability access, permitting disabled people who can't use their hands to type to use X Window functions. A sticky key feature means that disabled users won't have to hold down a control key and another key at the same time. For deaf users, the screen will flash instead of beep in the event of an error message. Other disability access features include repeat control, pseudo mouse, and slow keys.

In R6, session management has been enhanced to give users the ability to save the state of the program and the screen they were working on at the time of a crash. The new release provides a uniform mechanism for users to save and restore their sessions. Users can then resume in the place where they left off. This technology is designed to be entirely independent of X so that any application can utilize it.

Window System Independence

Having recognized the fast acceptance and promise of interoperability with object-oriented technology, the X Consortium has planned its own migration to object technology in future releases. It is not clear how much, if any, of the Fresco object-oriented technology will be available in X11 R6. Clearly, the aim of providing object-oriented technology in future releases will permit easier integration and coexistence of MS Windows, Mac, and X environments. In addition, Fresco is designed to use the PostScript graphics model to provide resolution-independent printing. It will also incorporate distributed objects as described in the Common Object Request Broker (CORBA) specification.

The goal of the X Consortium's Fresco technology is to support all of the application, whether it is menus and widgets on the outside of the application or text and graphics in a document inside of the application. Data or numbers from inside an application will be differentiated from the presentation method used to display a view of the data, like a graph or table.

Another complaint about X Window frequently heard from developers is the lack of font support in the X Window environment. The new release is expected to remedy some of the font shortcomings through its support of several predefined fonts.

And OSF Readies Motif 2.0

The Open Software Foundation's (OSF's) Motif is the standard implementation of the X Window System. Striving to bridge the operating environments of OS/2, MS Windows, and the Macintosh, OSF plans to deliver Motif 2.0 by the middle of next year. The new release is designed to permit users to integrate widgets from these different operating environments more easily.

In February of next year, OSF plans to release new widgets that will make interoperability among these three environments much more manageable. Those widgets will provide extensions that allow developers to create new widgets using C and C++ languages. And for those developers looking to embed one object into another, OSF will provide a container widget.

Two of the key technology suppliers for Motif 2.0 are Lotus Development Corporation (Cambridge, Massachusetts) and Computer Associates International Incorporated (Islandia, New York). Both vendors are expected to deliver technology that helps improve the consistency of Motif with IBM's Common User Access (CUA) workplace shell and MS Windows. One improvement designed to make Motif easier to use in mixed environments of MS Windows is the use of a combo box to facilitate interoperability between MS Windows and X Window.

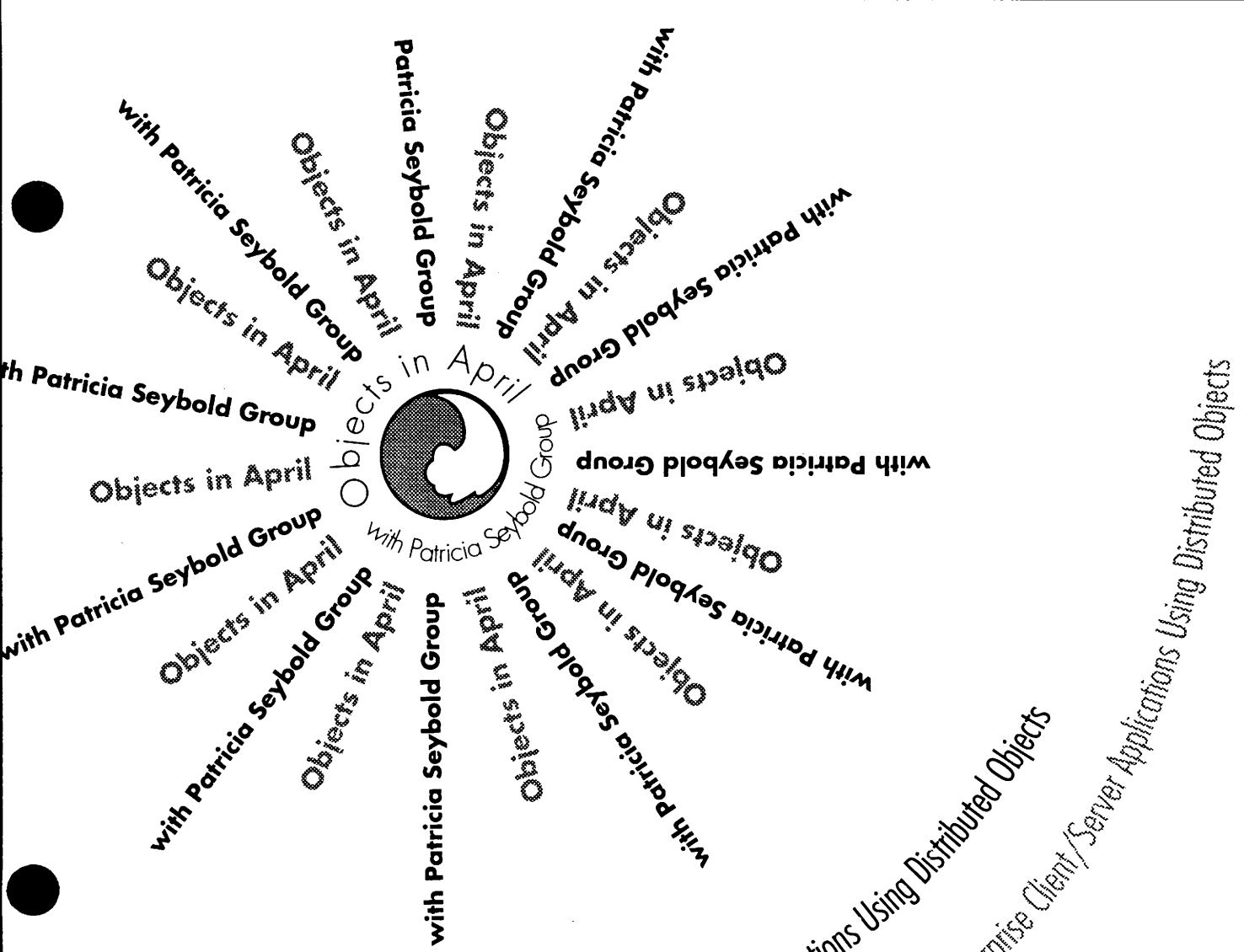
Motif 2.0 will also provide support for internationalization, so that, once a developer has written an application in one language, it can easily be converted into other languages.

In sum, Motif 2.0 is being enhanced to reflect the reality of the market; technologies will survive only if they are easy to use and can coexist alongside other de facto standard technologies, like Windows.

Can X Window Hold Its Own?

Although the Unix graphical user interface (GUI) wars are over, the market as a whole is entering a new round of UI contention. Taligent Incorporated (Cupertino, California) will enter the fray in 1994 with its new environment, and Microsoft will evolve the Windows interface with significant changes in Chicago in the same time frame. Although Unix vendors have united around Motif on top of X Window, X Window is in third or fourth place, depending on how you count IBM's Workplace Shell. However, X Window and Motif are still unique in their distributed orientation. This will continue to be a key differentiator for years to come.

— L. Bruno



Join the Best Minds in the Industry at

Patricia Seybold's Technology Forum

April 25-27, 1994

Cambridge Marriott, Cambridge, Massachusetts

The Only Client/Server Conference Where User Experience is the Object!

Patricia Seybold's Computer Industry Reports Order Form

Please start my subscription to:		U.S.A.	Canada	Foreign
☐ *Workgroup Computing Report*	12 issues per year	$385	$397	$409
☐ *Open Information Systems*	12 issues per year	$495	$507	$519
☐ *Distributed Computing Monitor*	12 issues per year	$495	$507	$519

Please send me a sample of: ☐ *Distributed Computing Monitor* ☐ *Open Information Systems* ☐ *Workgroup Computing Report* ☐ *Paradigm Shift—Case Studies in Distributed Computing*

Please send me information on: ☐ Consulting ☐ In-Depth Research Reports ☐ Conferences

Total $_____ ☐ **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: _____

Fax No.: _____ 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 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. **IO-1293**

Send to: Patricia Seybold Group: 148 State Street, 7th Floor, Boston MA 02109; FAX: (617) 742-1028; Phone: (617) 742-5200; MCI Mail: PSOCG

1992 and 1993 Feature Reports from Patricia Seybold Group
To Order these Back Issues, Fax (617) 742-1028, or Call (617) 742-5200.

Workgroup Computing Report

Volume 16 issues—$50

- ❒ 7/93 Issues in Remote Computing—Notebook PC Boom Raises Question: How to Handle Detachable Computing
- ❒ 8/93 Scheduling-Based Workflow—IET's W-6 Attacks the Complex Area of Time-Dependent Processes
- ❒ 9/93 Portfolio's Office IQ and ProdeaSynergy—Innovative New Products for Workgroups
- ❒ 10/93 Rules-Based Messaging—Using Electronic Mail as the Vehicle for Customized Workflow Applications
- ❒ 11/93 Strategies for Corporate E-Mail—Backbone, Client, and Other Considerations

Back Issue Total $_____

Open Information Systems

Volume 8 issues—$50

- ❒ 5/93 Unix and PC Interoperability—Toward the Utility Era of Computing
- ❒ 6/93 Open Electronic Mail—Interoperability through Standards
- ❒ 7/93 SAP's R/3 Software Suite—Architecturing Open, Integrated Software for Distributed Enterprises
- ❒ 8/93 Tomorrow's Microkernel-Based Unix Operating Systems
- ❒ 9/93 Apple Computer and Open Systems—The Other Desktop Standard
- ❒ 10/93 Progress Software's Version 7—Solving Client/Server Migration Issues
- ❒ 11/93 Oracle's Version 7—Can It Leapfrog the Competition?

Back Issue Total $_____

Distributed Computing Monitor

Volume 8 issues—$50

- ❒ 7/93 Detachable Computing—Vendors and Users Scramble to Support Occasionally Connected Computers
- ❒ 8/93 Modeling Network Behavior Using Objects—Horizon Strategies' AROs Increase Simplicity and Flexibility of Distributed Applications
- ❒ 9/93 Migrating to ATM—Four Scenarios to Take Users from Today's Networks to Switched Gigabit "Fabrics"
- ❒ 10/93 The Distributed Enterprise—The Logical Next Step in Distributed Object Computing Is Shared, Reusable Business Models, Not Just Objects
- ❒ 11/93 Novell AppWare/Expersoft XShell—Two New Efforts to Make Construction of Distributed Applications Transparent

Back Issue Total $_____

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION. **1A. Title of Publication:** Open Information Systems. **1B. Publication no.:** 1058-4161. **2. Date of filing:** December 1, 1993 **3. Frequency of issue:** Monthly. **3A. No. of issues published annually:** 12. **3B. Annual subscription price:** $495.00. **4. Complete mailing address of known office of publication:** 148 State St., 7th Floor, Boston, MA 02109. **5. Complete mailing address of the headquarters of general business offices of the publisher:** 148 State St., 7th Floor, Boston, MA 02109. **6. Publisher:** Patricia B. Seybold, 148 State St., 7th Floor, Boston, MA 02109. **Editor:** Michael Goulde, 148 State St., 7th Floor, Boston MA 02109. **Managing Editor:** Ronni T. Marshak, 148 State St., 7th Floor, Boston MA 02109. **7. Owner:** Patricia B. Seybold, 148 State St., 7th Floor, Boston MA, 02109, **8. Known bondholder, mortgagees, and other security holders owning or holding 1 percent or more of total bonds, mortgages, or securities:** None. 9. **For completion by nonprofit organizations authorized to mail at special rates:** Not applicable. 10. **Extent and Nature of Circulation:** A. Total no. copies printed Average 12 Mos. 600 Single Issue 500 B. Paid Circulation 1. Sales through dealers and carriers, street vendors and counter sales Average 12 Mos. None Single Issue None 2. Mail Subscriptions Average 12 Mos. 600 Single Issue 500 C. Total Paid and or Requested Circulation Average 12 Mos. 420 Single Issue 295 D. Free distribution by mail, carrier or other means samples, complimentary, and other free copies Average 12 Mos. 425 Single Issue 40 E. Total Distribution Average 12 Mos.465 Single Issue 335 F. Copies not Distributed 1. Office use, left over, unaccounted, spoiled after printing Average 12 Mos.135 Single Issue 165 2. Returns from news agents Average 12 Mos. None Single Issue None G. Total Average 12 Mos. 600 Single Issue 500 I certify that the statements made by me above are correct and complete. (signed) Patricia B. Seybold Publisher

STATEMENT OF OWNERSHIP, MANAGEMENT AND CIRCULATION. **1A. Title of Publication:** Patricia Seybold's UNIX in the Office. **1B. Publication no.:** 1058-4161. **2. Date of filing:** December 1, 1993 **3. Frequency of issue:** Monthly. **3A. No. of issues published annually:** 12. **3B. Annual subscription price:** $495.00. **4. Complete mailing address of known office of publication:** 148 State St., 7th Floor, Boston, MA 02109. **5. Complete mailing address of the headquarters of general business offices of the publisher:** 148 State St., 7th Floor, Boston, MA 02109. **6. Publisher:** Patricia B. Seybold, 148 State St., 7th Floor, Boston, MA 02109. **Editor:** Michael Goulde, 148 State St., 7th Floor, Boston MA 02109. **Managing Editor:** Ronni T. Marshak, 148 State St., 7th Floor, Boston MA 02109. **7. Owner:** Patricia B. Seybold, 148 State St., 7th Floor, Boston MA, 02109, **8. Known bondholder, mortgagees, and other security holders owning or holding 1 percent or more of total bonds, mortgages, or securities:** None. 9. **For completion by nonprofit organizations authorized to mail at special rates:** Not applicable. 10. **Extent and Nature of Circulation:** A. Total no. copies printed Average 12 Mos. 600 Single Issue 550 B. Paid Circulation 1. Sales through dealers and carriers, street vendors and counter sales Average 12 Mos. None Single Issue None 2. Mail Subscriptions Average 12 Mos. 390 Single Issue 377 C. Total Paid and or Requested Circulation Average 12 Mos. 390 Single Issue 377 D. Free distribution by mail, carrier or other means samples, complimentary, and other free copies Average 12 Mos. 45 Single Issue 35 E. Total Distribution Average 12 Mos. 435 Single Issue 412 F. Copies not Distributed 1. Office use, left over, unaccounted, spoiled after printing Average 12 Mos. 165 Single Issue 138 2. Returns from news agents Average 12 Mos. None Single Issue None G. Total Average 12 Mos. 600 Single Issue 550 I certify that the statements made by me above are correct and complete. (signed) Patricia B. Seybold Publisher