White Paper

Windows-family Integration with UNIX Systems

Using the Windows-family in UNIX environments

Microsoft

Windows-family Integration with UNIX Systems

Using the Windows-family in UNIX environments

A Whitepaper from the Business Systems Technology Series

February 1995

About the Microsoft Business Systems Technology Series

The Microsoft Business Systems Technology Series consists a of number of inter-related white papers dedicated to the task of educating IT professionals on technological issues surrounding Windows NT and the Microsoft Back Office family of products. While current technologies used within Microsoft products is often included, the real purpose of this series is to give the reader an idea of where major technologies are going, how Microsoft is leveraging off of those technologies in ways that are complementary and what this means to information technology planners doing real world IT implementations.

Contents

Overview

Organizations are efficient when they focus on their business goals instead of on the technologies they use. For this purpose, it is necessary to increase application usability and reuse, enable easy access to data from different sources, and reduce development time. UNIX systems are already present in the enterprise in significant numbers while advances in information technology are occurring on inexpensive PCs. MIS organizations must move forward, but also leverage their investments in UNIX systems, so many must integrate existing UNIX systems with the Microsoft® Windows® family of operating systems. Integration, in this case, means enabling network connectivity, cross-platform development, object services, messaging, database access and system management across Windows-family and UNIX systems.

Users get network connectivity between Windows-family and UNIX systems through technologies and products like Network File System (NFS), LAN Manager for UNIX (LMU), TCP/IP, and the X Window System™ protocol. Microsoft® has made the Windows application programming interfaces (APIs) available to developers on UNIX systems through the Windows Interface Source Environment (WISE) licensing program. With WISE products, users will be able to run shrink-wrapped Windows applications on UNIX systems. Microsoft's object technology, OLE, will also become available on UNIX systems through WISE. Users and developers get cross-platform database services between Windows-family and UNIX systems through Open Database Connectivity (ODBC) and Open Data Services. Users can send and receive email between the Windows-family and UNIX systems using products like Microsoft Mail and Microsoft Exchange Server. System managers can manage heterogeneous systems using Microsoft Systems Management Server and third-party Windows-family based Simple Network Management Protocol (SNMP) software such as OpenView and NetView®.

Users benefit from integration since they can focus on solving business problems without worrying about hardware and software details. Developers benefit since they can leverage their platform-specific knowledge for multiple platforms. System administrators can manage heterogeneous systems easily and remove administrative problems for users and developers. Integration helps MIS managers and corporate purchasing reduce costs for software, hardware, software maintenance, and training.

The case for integration

Aligning information strategy with business goals is one of the foremost challenges of IT organizations. Therefore, it becomes important to provide easy access to information from multiple sources, reduce software development time, increase application usability and re-use, and incorporate new technologies while leveraging existing assets. More than 2.5 million UNIX systems are installed in businesses, while Windows is the desktop operating system of choice, with an installed base of more than 60 million.

Early implementations of client/server technologies in MIS organizations occurred on UNIX systems and this led to a major investment in UNIX systems. Meanwhile, major advances in information technology are happening on inexpensive PCs running Windows and Windows NT™. MIS organizations must move forward with new technologies while still leveraging their previous investments in UNIX systems. It is therefore necessary to integrate existing UNIX systems in enterprises with Windows family systems.

UNIX systems also have a significant presence in financial, scientific, engineering, and academic organizations. These organizations are also increasingly turning to Windows and Windows NT systems, which must be integrated with existing UNIX systems.

Forms of integration

At a minimum, integration of Windows-family and UNIX systems must provide simple *network connectivity* between the systems. Users must be able to access files across platforms over a network, and applications on different systems must be able to communicate with one another.

Better integration will facilitate cross-platform *application development*, *object services*, *database access*, *messaging*, and *system and network management*.

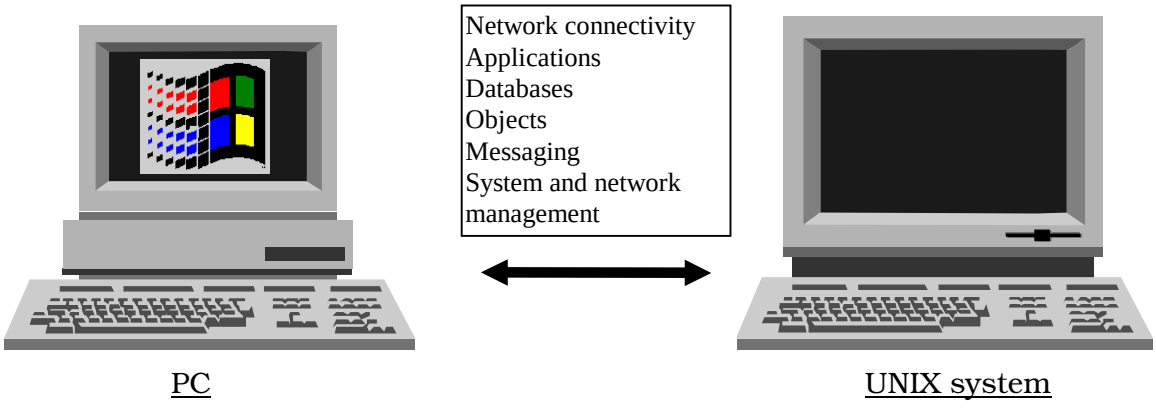


Figure 1: **Integration occurs at many levels**

With cross-platform application development, developers will be able to write platform-independent applications. Cross-platform object services let software components communicate across platforms easily and make users

more productive. System administrators will be able to manage heterogeneous systems easily if system-management software can provide, in one place, management information about heterogeneous systems. Cross-platform database and messaging services provide users with a means for easy, platform-independent information exchange.

This paper is organized into sections for the following readers:

- Users: Integration Shifts Focus to Tasks, Not Technologies
- MIS Managers: Move Forward While Leveraging Existing Investments
- System Administrators: Integration Offers Consolidated System and Network Management
- Developers: Integration Speeds and Simplifies Multiplatform Development

Throughout the paper, "Windows family" refers to Windows 3.1, Windows for Workgroups and Windows NT.

Contact information for third-party software vendors is provided in Appendix A.

2. Users: Integration Shifts Focus to Tasks, Not Technologies

End-users regard technology as a means towards achieving their goals. Generally it is best to hide as many software and hardware details from them as possible. Integration between Windows-family and UNIX systems lets users work with different systems without knowing much about the underlying details.

Accessing information across platforms

Users of Windows-family systems can access files on UNIX systems easily without understanding the protocol used to provide access to the file across platforms. For example, a user working with Microsoft Excel on a PC can open a text file on a UNIX system across a network in exactly the same way he or she would open the file on the PC using the File Open dialog box. Applications hide cross-platform technical complexity so users can focus on the applications they work with. Similarly, UNIX-system users can access files on Windows-family systems.

Using applications on other systems

Users often run both personal productivity Windows-based applications and some vertical or in-house developed software on a UNIX system. The user needs to access both applications from the same system.

With third-party products, users on UNIX systems or X terminals can run remote Windows NT-based applications or remote X Window System applications on Windows NT. Similarly, a user of a Windows family system can run remote X Window applications on UNIX systems. Many third-party software vendors support this option, so users have a wide choice. Section Four, "System Administrators: Integration Offers Consolidated System and Network Management" describes options for application services across systems.

Using the same application on different platforms

Users on UNIX systems who need applications like word-processors and spreadsheets either buy expensive UNIX applications, or buy a PC and PC-based applications. Those who buy UNIX applications are restricted to using the application only for that particular version of the UNIX system. Users cannot change to another UNIX system and continue to use the application they bought earlier.

Windows Interface Source Environment (WISE) emulators let users run inexpensive, shrink-wrapped Windows-based applications on UNIX and Macintosh® systems. WISE is a Microsoft licensing program that enables integration of Windows-based applications with UNIX and Macintosh systems. Microsoft has licensed Windows source code to Insignia Solutions Inc. and Locus Computing Corporation. Insignia's product, Softwindows, lets users run shrink-wrapped Windows and OLE-based applications on Macintosh and non-Intel® based UNIX systems. The Locus product, Merge, lets users work

with shrink-wrapped Windows-based applications on Intel-based UNIX systems.

WISE emulators enable users to run inexpensive, shrink-wrapped Windows-based applications on UNIX and Macintosh systems. WISE emulators make more than ten thousand off-the-shelf Windows-based applications available to users on Macintosh and UNIX systems.

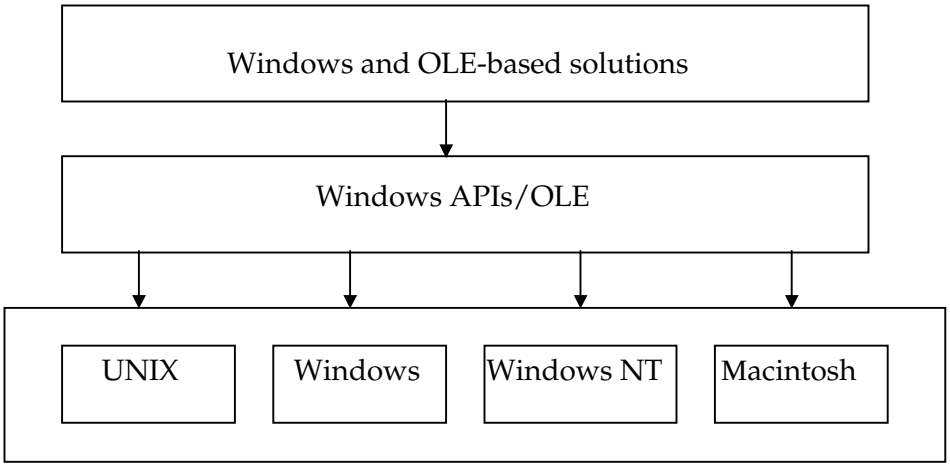


Figure 2. **Windows Interface Source Environment (WISE) provides open cross-platform flexibility**

WISE offers:

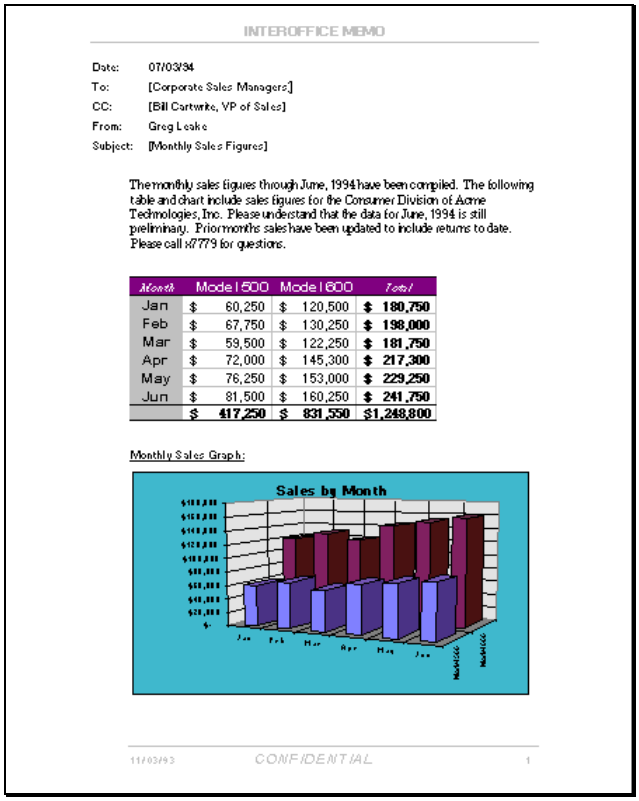
- Compatibility: Because WISE emulators use Windows source code licensed from Microsoft, they provide maximum compatibility with the many thousands of shrink-wrapped Windows-based applications on the market. Users can choose from several *inexpensive* shrink-wrapped Windows-based applications off the shelf and use them on UNIX and Macintosh systems.
- Consistency: End users who need to use existing Macintosh and UNIX solutions can use company-standard Windows-based applications to simplify their jobs. Users can thus use a consistent set of applications across several platforms.
- Confidence: Users can adopt a solution that will evolve along with future versions of Windows, taking full advantage of improving technology. Microsoft has committed to providing WISE licensees with future versions of Windows source code, thereby continuing to provide the best application compatibility and performance for today's and tomorrow's applications.

Object technology

OLE

OLE is Microsoft's object technology that lets users work easily with multiple applications. OLE is based on the Component Object Model, an underlying system software model that enables interoperability between software components. With OLE applications, users receive the following features:

- OLE Documents: OLE Documents improve the content of business documents and the process of creating them. OLE Documents can contain any type of information, including text, spreadsheet tables, pictures, graphics, video, or sound. Users can create the information contained in an OLE Document with any supporting OLE-enabled application, such as a spreadsheet application, graphics application, or multimedia application. These applications can be supplied by different software vendors, because OLE components work together. OLE Documents not only enhance user productivity by creating a seamless computing experience, but also let users communicate their ideas more effectively. As a user edits an OLE Document that contains different types of information, the specific tools necessary to edit the different types of information are made available to the user automatically within the context of the document. This is called Visual Editing. With OLE Linking, a document can contain information that maintains a data link to another document.



INTEROFFICE MEMO

Date: 07/03/94
To: [Corporate Sales Managers]
CC: [Bill Cartwrite, VP of Sales]
From: Greg Leake
Subject: [Monthly Sales Figures]

The monthly sales figures through June, 1994 have been compiled. The following table and chart include sales figures for the Consumer Division of Acme Technologies, Inc. Please understand that the data for June, 1994 is still preliminary. Prior months sales have been updated to include returns to date. Please call x7779 for questions.

Month	Model 500	Model 600	Total
Jan	$ 60,250	$ 120,500	$ 180,750
Feb	$ 67,750	$ 130,250	$ 198,000
Mar	$ 59,500	$ 122,250	$ 181,750
Apr	$ 72,000	$ 145,300	$ 217,300
May	$ 76,250	$ 153,000	$ 229,250
Jun	$ 81,500	$ 160,250	$ 241,750
	$ 417,250	$ 831,550	$1,248,800

Monthly Sales Graph:

11/03/93 CONFIDENTIAL 1

Figure 3: **OLE Document: This memo was created in a word-processing application. It contains an embedded spread sheet table and an embedded spreadsheet chart.**

- OLE Drag and Drop: OLE Drag and Drop allows users to exchange information between applications without worrying about saving files to disk or converting information to different formats. For instance, a user can select an embedded spreadsheet in a document and drag it over to another document in another application. By making data exchange graphical and intuitive, OLE Drag and Drop increases productivity.
- OLE Controls: OLE Controls are OLE-enabled software components that users can purchase to extend and enhance an application's functionality. Users can use OLE Controls in custom or off-the-shelf OLE-enabled applications. Most popular development environments, including the next version of the Microsoft Visual Basic® programming system, will support OLE Controls as an efficient means to build business applications using high-quality, prefabricated software components.
- OLE Automation: Automation enables applications to expose command sets that operate within and across applications. For example, a user can invoke a command from a word-processing program that sorts a range of cells in a spreadsheet created by a different application.

OLE is available today on Windows-family and Macintosh systems.

OLE on UNIX systems

Through the WISE licensing program, Microsoft will enable full OLE support on all major UNIX systems. Users can use OLE-enabled applications and realize all the benefits of OLE on UNIX systems.

Common Object Model (COM)

COM is a non-graphical layer of software built on the OLE Component Object Model. It was defined by Microsoft and DEC® in order to provide cross-platform object services. Working with DEC, Microsoft intends to make COM available on HP-UX®, SunOS, IBM® AIX®, OpenVMS, Ultrix™, and OSF/1.

In future, COM will let users use components across networks. For instance, a user working on a Microsoft Excel spreadsheet could bring real-time stock information directly into the cells of the spreadsheet from a remote UNIX server that supports COM. The source of the information would be transparent to the user, and the user would work as though the information was being generated locally.

Messaging

As more companies use heterogeneous systems, it becomes more important for users to be able to send messages from one system to another. Microsoft Mail enables users on PCs, Macintosh and UNIX systems to exchange messages with one another.

Communication between users on different systems is easier if the users make use of similar applications. With WISE emulators, users on UNIX and Macintosh systems can send and receive email using the same Windows-

based mail application on all systems, improving communication and increasing productivity.
Microsoft Exchange Server, the next messaging product from Microsoft, will also enable messaging between PCs, Macintosh and UNIX systems.

Databases

PC users can access databases from companies like Oracle®, Sybase®, or Informix® that are running on UNIX servers. The user does not need to know anything about the system on which the database resides. Users on UNIX-client systems can also transparently access database servers like Microsoft SQL Server™ running on Windows NT Server.
Several UNIX system based databases such as those from Oracle and Informix are also available on Windows NT, so users can choose from a variety of inexpensive database systems.

3. MIS Managers: Move Forward While Leveraging Existing Investments

Early implementations of client/server technologies in MIS organizations occurred on UNIX systems and this led to a major investment in UNIX systems. Meanwhile, technological advances in information technology are happening rapidly on inexpensive PCs running Windows and Windows NT. MIS managers must move their organizations forward with new technologies while still leveraging existing investments in UNIX systems. Integration between Windows-family and UNIX systems helps MIS managers ease the transition to open systems and reduces the costs that result from a heterogeneous environment.

Reducing software and hardware costs

Many MIS professionals use UNIX systems for running some applications and PCs for personal-productivity applications. Such users can use Windows Interface Source Environment (WISE) products to run the personal productivity applications on their UNIX systems. WISE thus eliminates the need for extra PCs used in this manner.

Because Windows-based applications are inexpensive, MIS managers can use WISE products to run them on Macintosh and UNIX systems and thus reduce costs.

Because solutions are available for application services between Windows-family and UNIX systems, MIS managers need not buy multiple systems to provide users access to both Windows-family and UNIX applications.

Reducing product cycle time and software maintenance costs

WISE lets MIS departments easily develop and maintain applications simultaneously for Windows, Macintosh and UNIX systems. Developers write to Windows APIs and OLE and use the resulting applications on Windows, Macintosh, and UNIX systems. Developers maintain a single code base for an application running on Windows, Macintosh, and UNIX platforms.

Similarly, developers already proficient with UNIX systems can write to UNIX APIs and use the application on Windows NT with the help of the UNIX environments available for Windows NT.

With the availability of cross-platform object technology, developers can reuse software components across platforms and be more productive.

Developers writing database applications can become more productive by using Open Database Connectivity (ODBC), which is available on Windows family, Macintosh, and UNIX systems. With ODBC, developers write to the same interfaces independent of the database they are accessing.

MIS managers thus reduce product cycle time and maintenance costs for such multiplatform application development.

Reducing training costs

Integration reduces costs because users do not need to be trained in different systems to access information on these systems. For example, a user on Windows can work with Microsoft Excel on a PC and open a file on a UNIX sys-

tem across a network exactly as he or she would open the file with Microsoft Excel on the PC. The user does not need to know anything about UNIX systems to perform this task. Similarly, Windows NT provides file-access utilities such as ftp, telnet and so on, that are already familiar to UNIX system users, so these users need not be trained in new techniques for information access.

Developers writing database applications can use ODBC on both Windows-family and UNIX systems, which gives them access to more than 50 database servers. Developers need not be trained for the different databases they are accessing, or for the system they are using — they simply write to the ODBC APIs.

WISE products let users and developers trained in Windows and OLE leverage their training for Macintosh and UNIX systems as well. Similarly, developers and users already proficient on UNIX systems can use UNIX environments on Windows NT with little training.

Because users need not learn to use different technologies, they will need less help from MIS in day-to-day activities. MIS helpdesks will thus receive fewer help requests.

Making communication easier

Several cross-platform messaging solutions are available, so users on different platforms can easily exchange messages between Windows-family, Macintosh and UNIX systems. MIS professionals can use the same applications used by senior management and the sales force, although MIS may be working on different platforms. This is because WISE emulators enable shrink-wrapped Windows-based applications to run on Macintosh and UNIX systems. This increases communication between MIS, senior management, and sales.

Choosing from among many vendors

MIS professionals can choose from more than ten thousand inexpensive shrink-wrapped Windows-based applications offered by many different vendors. These easy-to-use applications run on Windows, Macintosh, and UNIX systems and increase productivity.

Reducing client/server migration costs

While downsizing to a client/server system, MIS may need to migrate mainframe applications to heterogeneous systems. This requires expertise in the targeted platforms and the extra work of migrating the same mainframe application to multiple platforms. Now, MIS professionals can migrate mainframe applications once to Windows and use WISE products to run the migrated software on UNIX and Macintosh systems also. On the other hand, MIS can choose to migrate mainframe applications to UNIX systems and use UNIX environments on Windows NT to run the applications on Windows NT.

4. System Administrators: Integration Offers Consolidated System and Network Management

System administrators in a multiplatform environment need to enable users to access information, run applications, access databases, and exchange messages across platforms. Also, they need to manage heterogeneous systems from a single place.

Providing file services across platforms

At a minimum, system integrators must provide users with a means of accessing files between the Windows-family and UNIX systems. System integrators must enable Windows-family systems to provide file services to UNIX clients and vice-versa.

Windows NT Server as a file server for UNIX clients

Windows NT Server provides file services to PCs through the Server Message Block (SMB) protocol. File service for UNIX clients is available through the Network File System (NFS), SMB protocols and the ftp service.

- NFS: Third-party NFS servers such as DiskShare from Intergraph Corporation, BW-Connect from Beame & Whiteside Inc., NFSWare from Process Software, and Chameleon32NFS from NetManage Inc. are available for Windows NT. These servers enable Windows NT Server to provide file service for PCs, UNIX workstations or other systems acting as NFS clients. They provide support for the NTFS, FAT, CDFS and HPFS file systems and some also enable the exporting of network drives. A Windows NT Server system could, for instance, connect to a NetWare® server and share out the NetWare server's files to NFS clients. Support for the Intel®, Alpha AXP™, and MIPS® platforms is available and PowerPC™ support will be available in future.
- SMB: Free SMB clients for UNIX systems are provided by SMBServer, which is available on *nimbus.anu.edu.au* in the directory *pub/tridge/server*.
- ftp, rcp: Windows NT Server provides the ftp service, which enables bidirectional transfer of ASCII and binary files. The ftp service is not installed by default since the service could cause a security breach to the Windows NT system. Because UNIX users are already familiar with the ftp service, they can easily access files from Windows NT Server. rcp provides a mechanism to unidirectionally copy files between two systems of which one or both can be remote. A third-party rcp server for Windows NT is available from Software Innovations Inc.

UNIX servers as file servers for Windows family clients

UNIX servers can provide file services to Windows-family systems through NFS, SMB, ftp, and rcp.

- NFS: Third-party NFS clients such as PC-NFS®/Windows from Sun Microsystems®, PC-NFS/Windows NT from Intergraph, BW-Connect from Beame & Whiteside, and Chameleon32NFS from NetManage are available for the Windows family. With such software PC users can ac-

cess files from PCs, UNIX systems and other systems acting as NFS servers.

- SMB: Users from Windows or Windows NT systems can access files on UNIX systems running LAN Manager for UNIX (LMU). Microsoft has licensed the source code for LAN Manager for UNIX and the source code for Windows NT Server to AT&T® GIS. AT&T GIS will sublicense the code to other vendors to provide Microsoft Networking on UNIX systems. Advanced Server for UNIX (ASU) is the next generation of LAN Manager for UNIX being developed by AT&T. The ASU technology is fully interoperable and functionally compatible with the networking technology incorporated in Windows NT. ASU is equivalent to Windows NT Server in terms of establishing and managing accounts and providing support for trusted domains. System administrators can use the Windows NT Server tools to administrate the UNIX servers. DEC has a license for Advanced Server for UNIX and has stated that it will implement it on the OSF/1 and VAX™ systems. Unipress will provide Advanced Server for UNIX on Solaris. SCO® announced an agreement to license ASU from AT&T GIS and provide it on SCO-UNIX. Bull, SNI and Olivetti® will also provide Advanced Server for UNIX on their UNIX systems.

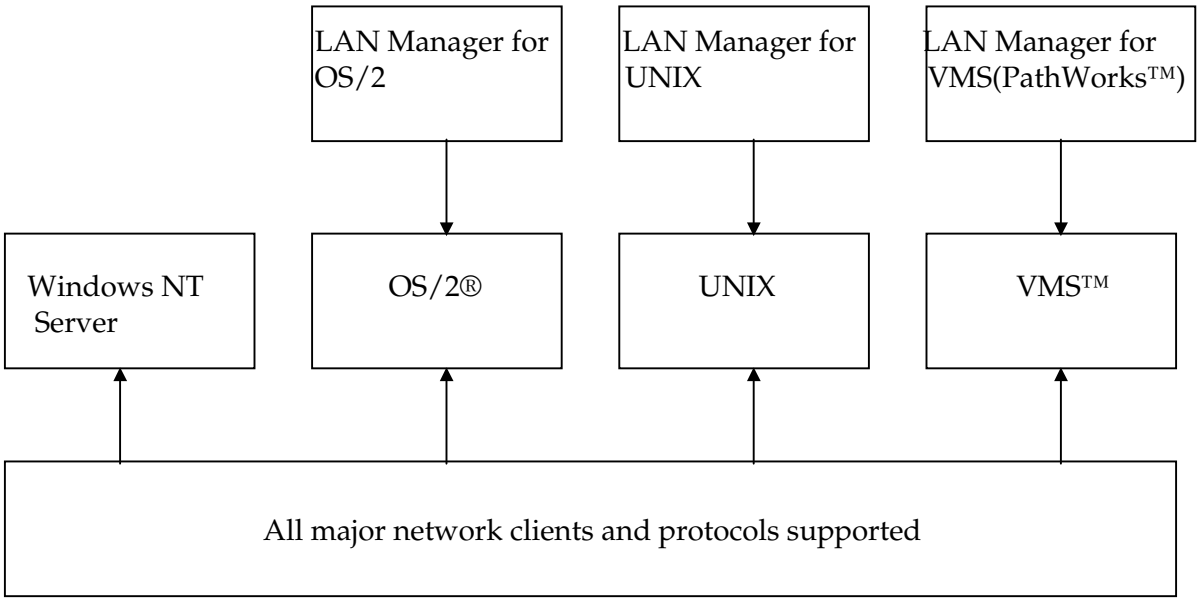


Figure 4: **Microsoft Networking**

- ftp, rcp: Windows NT provides ftp and rcp clients, and ftp and rcp clients for Windows are available from third parties like Walker Richer & Quinn, Wollongong, and Frontier Technologies. Both ftp and rcp clients are functionally similar to the ones available on UNIX systems. With the ftp client, a user on Windows NT can access ASCII and binary files on

remote systems. The remote system(s) must be configured to trust the rcp requests by configuring a file such as a .rhosts file.

Printing across platforms

Windows NT provides TCP/IP printing services based on RFC1179, which contains the communication specification for line printer remote (lpr) and line printer daemon (lpd) printing. The lpd and its components in RFC1179 are known as Berkeley Style Daemons. Windows NT enables printing to UNIX servers that support the Berkeley style TCP/IP printing daemons. Windows NT provides an lpd server that lets UNIX machines with Berkeley-style lpr daemons to send print jobs to Windows NT.
lpr and lpd support for Windows is available from third parties like NetManage, Wollongong, SunSelect, Walker Richer & Quinn, and Frontier Technologies.

Running applications across a network

Corporate users often use personal productivity applications on Windows as well as workstation applications on UNIX systems. In such situations, system administrators must provide users access to both the personal productivity applications and the UNIX applications from the same system.

Windows NT Server as an application server for UNIX clients

Windows NT Server can provide application services to Windows-family systems over a network. Users on Windows family clients connect to a sharepoint on a Windows NT Server and use applications on the server. Application services to UNIX clients are provided as follows:

- telnet, rlogin, rsh, rexec: telnet and rlogin offer basic terminal emulation to remote systems over TCP/IP. rsh provides a simple mechanism to execute a process on a remote system over TCP/IP. rexec provides similar functionality as rsh with the addition of cleartext password authentication. Windows NT does not provide any of these services because these could cause a security loophole. When a user on another system needs to telnet, rlogin, rsh or rexec to a Windows NT system, the password for the Windows NT account will be passed in cleartext over the network. A malicious sniffer on the network could get hold of the user's password and compromise security. If users want these services on Windows NT, they can use third-party telnet and rlogin server software available from Ataman Software, Seattle Lab, and Software Innovations. Software Innovations also provides rsh and rexec servers for Windows NT.
- X Window System clients on Windows NT: X Window System client software on Windows NT is available from third parties such as Datafocus Inc. Users of UNIX systems can run X Window System applications on Windows NT over a network.

- Running Windows-based applications remotely: Citrix Systems, Prologue, and Adonis Corporation will make multi-user versions of Windows NT available to customers. Using software being developed by these companies, users of UNIX systems or X terminals can run Windows family-based applications on Windows NT Server systems over a network.

UNIX servers as application servers for Windows family clients

- telnet, rlogin, rsh, rexec: Windows NT provides telnet, rsh, and rexec clients. Third party rlogin, rsh, rexec, and telnet clients are available for the Windows family from vendors like Wollongong, Walker Richer & Quinn, FTP Software, and Frontier Technologies.
- X Window System servers on Windows NT: Users on Windows and Windows NT systems can run X Window System applications on UNIX systems using X server software. Third party X server software such as XoftWare from AGE Logic, eXcursion from DEC, X-One from Grafpoint, eXceed from Hummingbird Communications, Multiview/X from JSB Corporation, Chameleon from NetManage, PC-Xware from Network Computing Devices, X/Vision from VisionWare, Reflection/X from Walker Richer & Quinn, and eXodus from White Pine Software are available for the Windows family.

Connecting to the Internet

Internet services

Windows NT provides the ftp service that lets users on the Internet transfer files to and from Windows NT. This ftp service is similar to the one in use on UNIX systems on the Internet, so UNIX users can use it easily.
Windows NT 3.5 resource kit will contain two public-domain Internet information servers: a World Wide Web (WWW) server and a Gopher server. Gopher offers distributed file-serving features with a good user interface. Web servers provide plain and hypertext documents with links to other documents around the world. These servers are available for Intel, MIPS and Alpha AXP platforms. Public domain WWW and Gopher servers are available as shown below.

Application	***Site***	***Directory***	***File name***
GopherS	emwac.ed.ac.uk	pub\gophers	gs*.zip
	sunsite.unc.edu	pub\packages\infosystems\gopher \servers\nt-goph	GS*.zip
HTTPS	emwac.ed.ac.uk	pub\https	hs*.zip
	sunsite.unc.edu	pub\packages\infosystems \www\servers\https	HS*.zip

Wide Area Information Server (WAIS) servers offer content-indexing services to clients who connect over the Internet. A public domain WAIS server will be available in the Windows NT 3.5 resource kit. This server is also available on *emwac.ed.ac.uk.edu* in the directory *pub\waiss*.

Internet clients

Windows NT Workstation ships with ftp, telnet, rcp, finger, rexec, and rsh clients.
Public-domain Gopher clients are available from several sites as follows:

Application	***Site***	***Directory***	***File name***
Gophbook	boombox.micro.umn.edu sunsite.unc.edu	pub\gopher\Windows\gophbook pub\micro\pc-stuff\ms-windows\winsock	gophbook*.* gophbook.zip
BCGopher	bcinfo.bc.edu	pub\bcgopher	bcg08a3.exe
WSGopher	boombox.micro.umn.edu	pub\gopher\Windows	wsg-11.exe
HGopher	gopher.ic.ac.uk	Networking	(see index)

Public domain WWW clients for Windows NT are available as shown in the following table:

Application	***Site***	***Directory***	***File name***
NCSA's Mosaic	ftp.ncsa.uiuc.edu	Web\Mosaic\Windows	wmos20a7.zip
	sunsite.unc.edu	pub\packages\infosystems\www\clients\Mosaic\Mosaic-NCSA\Windows	wmos20a7.zip
Cello	ftp.law.cornell.edu	pub\LII\Cello	*.*
WinWeb	sunsite.unc.edu	pub\packages\infosystems\www\clients\WinWeb	winweb.zip
Mosaic Netscape	ftp.mcom.com	pub\netscape\windows	nscape09.zip

Using different network protocols

Increasingly, system administrators must provide interoperability between systems supporting different networking protocols. With the Windows family, system administrators get support for protocols like TCP/IP, NetBEUI, IPX/SPX, NetWare Core Protocol (NCP), Systems Network Architecture (SNA), LAN Manager, X Window System, and NFS. Using this rich protocol support, system administrators can integrate Windows-family systems into an heterogeneous environment in an enterprise.

Managing heterogeneous systems

System administrators can effectively manage heterogeneous systems if they can get information about the systems at one place. Administrators can use management software on Windows family or UNIX systems to get information about UNIX and Windows-family systems on the network.

SNMP-based system management

SNMP is part of the TCP/IP protocol and is used for exchanging management information among UNIX systems. SNMP management software runs on central servers and enables system administrators to get information from client systems by using the SNMP service running on UNIX client systems. HP® OpenView, Sun SunNet manager, IBM NetView are examples of SNMP management software. These products are available on a variety of operating systems.

Microsoft provides the SNMP service in Windows NT and will provide the SNMP service in Windows® 95, to better integrate the Windows family with UNIX systems. SNMP agents on Windows NT and Windows 95 systems will provide SNMP management software on other systems with information about themselves. System administrators on UNIX systems will thus be able to use SNMP management software such as HP OpenView on UNIX systems to manage Windows NT and Windows 95 systems.

The SNMP service in Windows NT provides support for the Internet Management Information Base-II (MIB-II) and LanMan MIB II. Support for the Ethernet MIB, X.25 MIB and Host MIB is planned for future versions of Windows NT.

HP OpenView and IBM NetView are examples of SNMP-based management software available on the Windows family. Using such products, system administrators using the Windows-family can manage UNIX clients.

Microsoft Systems Management Sever

Using Microsoft Systems Management Server, system administrators on Windows NT Server can distribute and install software, get hardware and software configuration, and do remote performance analysis and troubleshooting for Windows-family systems. Using Microsoft Systems Management Server with DEC Polycenter AssetWorks on Windows NT Server and UNIX client software from DEC, system administrators on Windows NT Server can get hardware and software information from and distribute or install software to UNIX systems. DEC provides Microsoft Systems Management Server clients for OpenVMS, Ultrix, SunOS, and OSF/1. Clients for Solaris, HP-UX and AIX will be available in the future.

Microsoft Systems Management Server also enables information exchange with system-management software like OpenView and NetView.

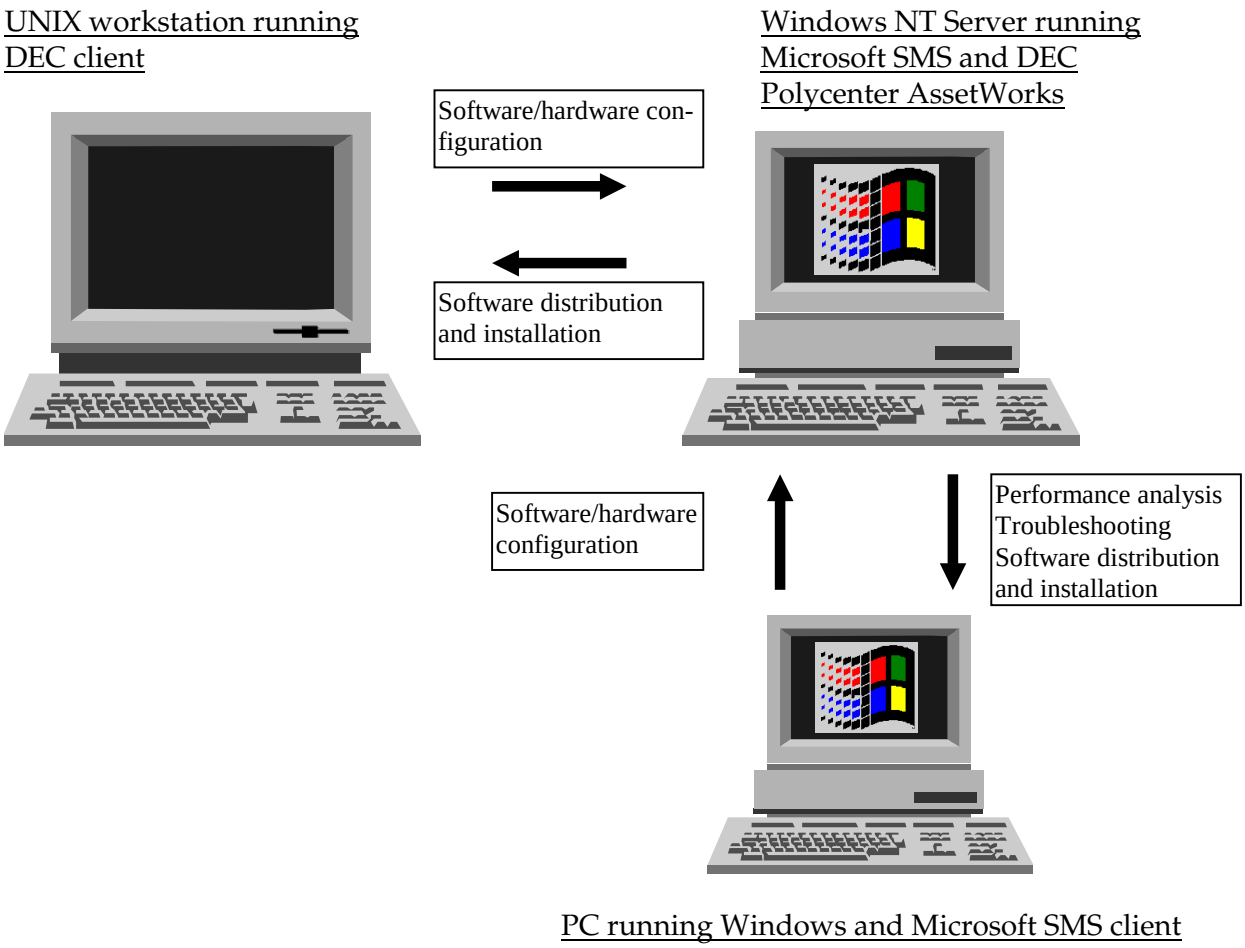


Figure 5: **System management using Microsoft Systems Management Server**

Managing networks

Windows NT Server provides support for the Dynamic Host Configuration Protocol (DHCP). With this service, Windows NT Server can provide IP addresses, subnet masks, and default gateway information automatically to Windows-family systems on a Windows NT network. This makes administration of TCP/IP networks easier.

A Windows NT hosted Domain Name Service (DNS) server will be included in the Windows NT 3.5 resource kit. The DNS server handles mapping between host names and IP addresses and makes the information available over the entire Internet. The DNS server thus enables communication between Windows-family and UNIX networks.

Windows Internet Naming Service (WINS) servers configure and maintain the mapping between IP addresses and names of machines in a TCP/IP based Microsoft network. The DNS server provides transparent gatewaying of DNS queries to WINS servers. Hence, UNIX systems can look up names of

Windows family machines in the WINS database and need not use IP addresses for communication with Windows-family systems.

Messaging across platforms

System administrators can enable users to send email between Windows-family and UNIX systems by using Microsoft Mail and gateways. Microsoft Mail can be used to send mail to SMTP, X.400 and uucp based email applications on UNIX systems using the Microsoft Mail Gateway to SMTP, and the Microsoft Mail Gateway to X.400. The SMTP gateway supports Multipurpose Internet Mail Extensions (MIME), several attachment handling options, message forwarding and multiple local account aliases. The X.400 gateway is compliant with the 1984 X.400 standard.
In addition to sending mail to UNIX systems, users can use Microsoft Mail and gateways to send email to Macintosh, MVS™, VMS and several other systems.

Microsoft Exchange, the upcoming messaging product from Microsoft, will provide clients for Windows family, Macintosh and UNIX systems. Using these clients, users can send and receive messages between PCs, Macintosh and UNIX systems. Microsoft Exchange Server provides native support for SMTP and X.400. System administrators can thus set up communication between Microsoft Exchange and both X.400 and SMTP-based messaging applications on UNIX systems.

Database access across platforms

System administrators can enable PC users to have access to databases from companies like Oracle, Sybase, and Informix running on UNIX servers. Users on UNIX systems can also access database servers running on Windows NT Server. Microsoft has enabled this integration between Windows-family and UNIX systems through Open Database Connectivity (ODBC) and Open Data Services (ODS). Gateways between Microsoft SQL Server and a wide range of databases on MVS, UNIX, VMS, and AS/400® systems are also available from third parties such as Micro Decisionware, Information Builders and Showcase. Using these gateways, users can access data transparently from MVS, UNIX, VMS, or AS/400 systems. For instance, an organization could store information in two different databases — a Microsoft SQL Server database on a Windows NT Server and an ORACLE database on a UNIX server. A user at a Windows-family or UNIX client system could request data from the Microsoft SQL Server database. If the user requests data that is not present in the Microsoft SQL Server database, the gateway communicates with the UNIX server and gets the data from the ORACLE database and passes it back to the client. This integration with the UNIX server is completely transparent to the user, who is only aware of getting data from Microsoft SQL Server.

5. Developers: Integration Speeds and Simplifies Multiplatform Development

Developers benefit from integration between Windows-family and UNIX systems because they can build solutions once and leverage them for multiple platforms.

Developing cross-platform solutions

Corporate developers in a multiplatform environment either develop applications separately for each platform or develop applications for one platform and port the applications to the other platforms. These developers must understand the differences between the Windows, Macintosh, and UNIX operating systems, as well as the differences between the several UNIX systems themselves.

Windows Interface Source Environment (WISE)

Windows Interface Source Environment (WISE) is a Microsoft licensing program that enables integration of Windows solutions with Macintosh and UNIX systems. Microsoft has licensed the Windows family source code to Mainsoft Corporation and Bristol Technology Inc. Using the products developed by Mainsoft and Bristol, developers will be able to write to the Win32® API and OLE on different UNIX platforms.
WISE Software Development Kits (SDKs) enable developers to write to Windows APIs and use the resulting applications on Macintosh and UNIX systems. To get a Windows application running on a Macintosh or UNIX system using a WISE SDK, the application *source code* must be compiled on those systems. Some benefits of WISE SDKs to developers are:

- Compatibility: Developers build and maintain one code base compatible with several platforms and can use tools provided by WISE SDKs to easily move existing Windows applications to Macintosh and UNIX systems.
- Consistency: Developers write to a *standard*, *consistent* and *well-documented* set of APIs and leverage their solutions across the Windows family, UNIX and Macintosh platforms.
- Confidence: Developers can build solutions that will evolve along with future versions of the Windows family, taking full advantage of evolving 32-bit technology. Microsoft has committed to providing WISE licensees with future versions of Windows family source code, thereby continuing to provide the best application compatibility and performance for today's and tomorrow's applications.

A WISE/UNIX SDK provides the Windows libraries, Microsoft Foundation Classes (MFC) for UNIX, a help compiler, resource compiler, and tools for porting Windows-based applications to UNIX systems.

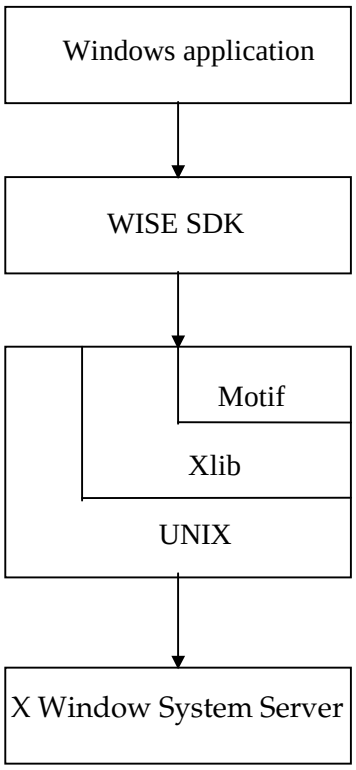


Figure 6: **WISE SDK enables developers to write to Windows APIs on UNIX systems.**

UNIX environments on Windows NT

Developers can write to UNIX APIs on Windows NT by using either the POSIX subsystem of Windows NT or third-party products such as NuTCRACKER from Datafocus Corporation and Portage from Consensys Corporation.

The POSIX subsystem of Windows NT is certified by the National Institute of Standards and Technology (NIST) to be compliant with the IEEE 1003.1 POSIX specification.

NuTCRACKER provides support for UNIX APIs on the Win32 subsystem of Windows NT and Portage provides UNIX System V Release 4 support on top of the Win32 subsystem. Both these third-party products provide several UNIX utilities on Windows NT. The products can be used either as porting tools to port UNIX applications to Win32 or for simultaneous cross-platform development. NuTCRACKER and Portage provide support for UNIX processes, signals, semaphores, message passing, sockets, file management, and the C runtime libraries.

Developing components using OLE

OLE is Microsoft's object technology that enables developers to integrate off-the-shelf components with custom software to build solutions. OLE is based on the Component Object Model, an underlying system software object model that allows complete *binary* interoperability between components written by different companies, possibly in different languages. These software components can be purchased, replaced, enhanced and re-used at any time during the business-system life cycle. By writing to OLE, developers can build applications that offer users features such as:

- OLE Documents: Visual Editing and OLE Linking
- OLE Drag and Drop
- OLE Controls

With OLE Automation, developers can create macros and other command sets that work *across* applications. OLE Automation enables developers to programmatically use components and functionality provided by other applications. Existing application macro languages can be extended easily to include new sets of commands and function calls that other applications can execute. For example, a word-processing document can send commands to a spreadsheet document to automatically open a spreadsheet, create a chart, and embed or link the chart into the word-processing document. As another example, a spreadsheet application could use OLE Automation to send commands to a statistical analysis package to perform special analysis functions.

OLE is available today on Windows-family and Macintosh systems.

Developing with OLE on UNIX systems

Through the WISE licensing program, Microsoft will enable full OLE support on all major UNIX systems. Mainsoft and Bristol will provide OLE support on all major UNIX systems. Developers can realize all the benefits of OLE on UNIX systems also.

Common Object Model (COM)

COM is a layer of non-graphical software built on top of the OLE Component Object Model. It was defined by Microsoft and DEC in order to provide cross-platform interoperability. Working with DEC, Microsoft intends to provide COM on HP-UX, SunOS, IBM AIX, OpenVMS, Ultrix, and OSF/1. In future, developers will be able use COM to build solutions that use components across platforms over a network.

Interoperability with Common Object Request Broker Architecture (CORBA)

CORBA is a specification created by the Object Management Group (OMG) to define interfaces for distributed object computing services. Microsoft plans to provide interoperability between OLE and CORBA across a network when consistent implementations of CORBA become available. Microsoft, DEC, and Candle Corporation have provided a proposal to the OMG to

use the COM wire protocol as the wire protocol for interoperation between OLE and CORBA. This will require an object adaptor on the CORBA server. With this interoperability, developers will be able to write applications that use components developed using CORBA-compliant object technologies.

Writing distributed applications

Developers can use the remote procedure call (RPC) mechanism provided by Windows NT to write distributed applications. Microsoft RPC interoperates with the Distributed Computing Environment (DCE) RPC, so developers can write applications that communicate with UNIX applications that make use of DCE RPC.

Using the Winsock interface provided by Microsoft, developers on Windows-family systems can write applications that communicate with UNIX systems. The Winsock interface is similar to the Berkeley sockets interface available on most UNIX systems, so developers already proficient on UNIX systems can easily use Winsock on Windows NT.

DCE on Windows NT is available from DEC. This provides RPC, directory, time, security and thread services. A DCE client for Windows is available from Gradient Technologies.

Developing database applications across platforms

Developers on PCs, Macintosh, and UNIX systems can write software to access a Microsoft SQL Server database running on Windows NT Server. Similarly, developers on Windows and Windows NT systems can write client software to get data from databases like SYBASE and ORACLE running on UNIX servers.

Open Database Connectivity (ODBC)

ODBC is an open, vendor-neutral, and powerful interface that application writers can use to write applications that access database systems from different database vendors. Developers need not learn multiple programming interfaces; they can use the universal set of interfaces provided by ODBC. Using ODBC, developers can write software that can access over 50 database servers including ORACLE, Informix, SYBASE and Microsoft SQL Server. ODBC support is available in many popular front-end applications and development tools like Microsoft Visual Basic, Powersoft PowerBuilder™ and Knowledgeware® ObjectView. ODBC support on UNIX systems will be available from Visigenic software. ODBC support on the Macintosh is available from Apple®.

Open Data Services (ODS)

ODS is an event-driven application program interface that provides the foundation for integrated client-server business solutions. Developers can use ODS to integrate almost any external data with Microsoft BackOffice and Microsoft SQL Server.

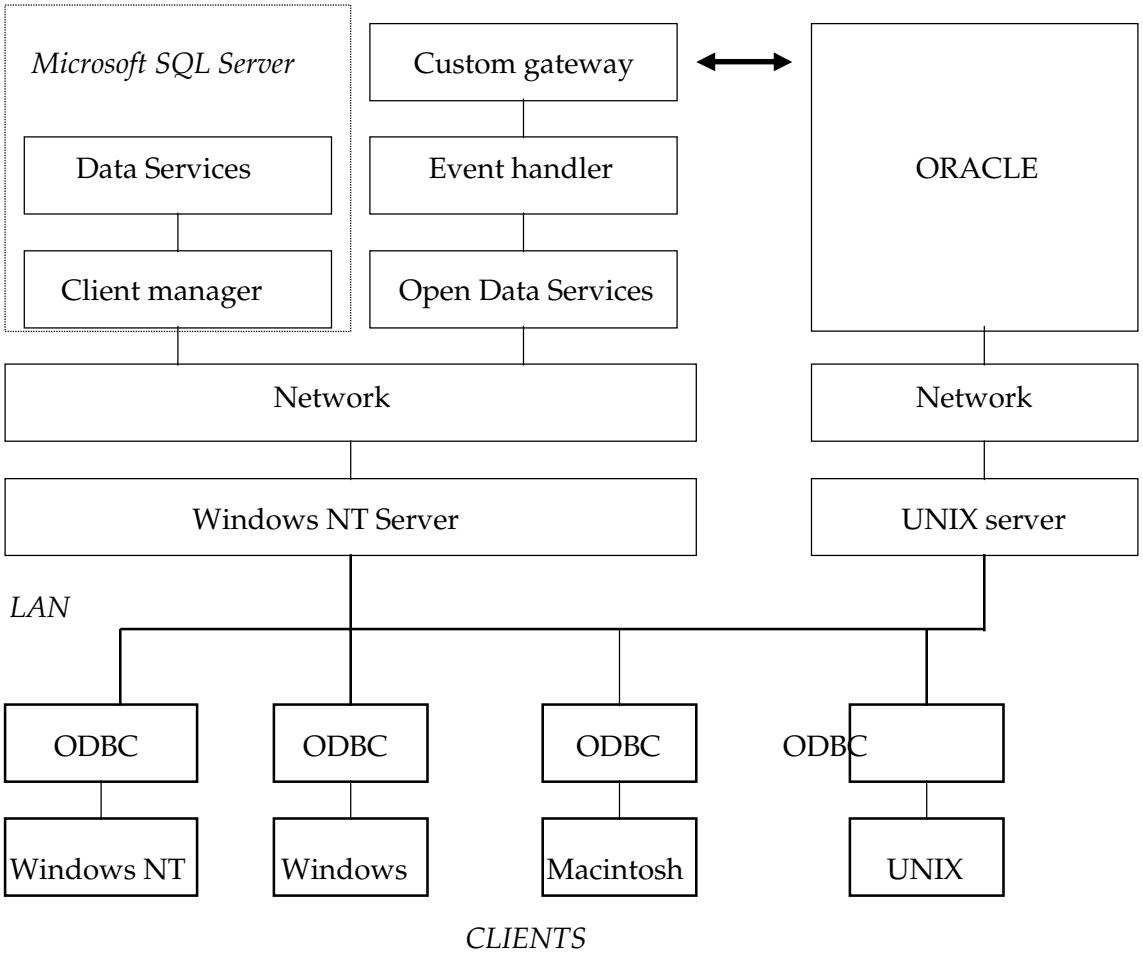


Figure 7: **Building integrated database solutions with ODBC and ODS.**

Developers can use ODS to build gateways between Microsoft SQL Server and a wide range of databases on MVS, UNIX, VMS, and AS/400 systems. For instance, a developer could write an application on Windows NT Server using ODS so that when a Microsoft SQL Server client requests data, the application sends the request to another system and passes the reply back to the client. For the client, the reply seems to come from Microsoft SQL Server, but the information actually comes from the ODS application that could, for example, be getting the data from a UNIX server.

ODS can be also used to build server applications such as a security application to verify login to a Microsoft SQL Server database, an auditing application to maintain a record of client transactions, or an application for notifying a user of changes in a Microsoft SQL Server database.

Appendix A: Third-party Contacts

Adonis Corporation	(206) 827-6467
Ataman Software	(801) 583 9132
Beame & Whiteside	(919) 831-8989
Bristol Technology Inc.	(203) 438-6969
Citrix	1-800 437-7503
Consensys Corporation.	(905) 940-2900
Datafocus Inc.	(703) 631-6770
Digital Equipment Corporation (for Polycenter Asset Works)	(603) 881-2223
Frontier Technologies	(508) 851 5700
Gradient Technologies	(508) 624 9600
Information Builders	(212) 736-4433
Insignia Solutions	(415) 694-7600
Intergraph Corporation	(205) 730-6112
Locus Computing Corporation	(310) 670-6500
Mainsoft Corporation	(408) 774-3400
Micro Decisionware	(303) 443-2706
NetManage	(408) 973-7171
Process Software	1-800 722-7770
Prologue (France)	(1) 69 29 39 90
Seattle Lab	(206) 828 9001
Showcase Corporation	1-800 829-3555
Software Innovations Inc.	1-800 946-6688
SunSelect	(508) 442-2300
Unipress	(908) 287-2100

Visigenic Software	(415) 286-1900
Walker Richer & Quinn Inc.	1-800 872 2829