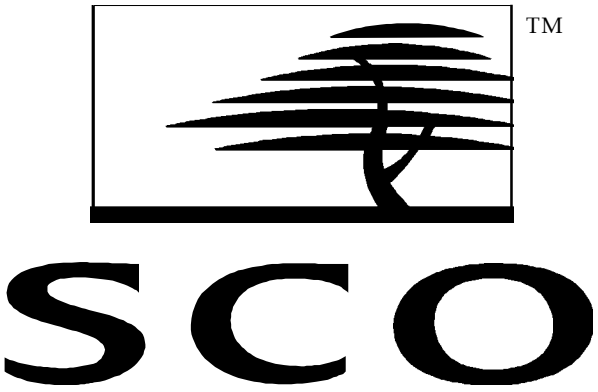
SCO® UnixWare® 2.1
Technical Summary

An SCO Technical White Paper

February 1996



Introduction

This document is meant to provide a detailed look at the components of SCO UnixWare 2.1, part of an important new phase in commercial computing: networked, multi-processing UNIX® systems. Businesses increasingly face more information processing decisions as information itself becomes more valuable and available. The need for reliable integration between different systems and networks is often one of the most important (and costly) issues to address. The UNIX® operating system will increasingly be at the heart of commercial and enterprise information strategies due to its expandable, adaptable nature. And the technologies of SCO UnixWare 2.1 will reach many thousands of businesses and millions of customers due to its effective interoperability with existing environments and its far-reaching technological advances.

SCO UnixWare 2.1 is based on the highly acclaimed UNIX® System V Release 4.2MP, first developed by UNIX System Labs® (USL) and later enhanced and productized by USL and Novell engineers. SCO UnixWare 2.1 has the most advanced and comprehensive implementation of Symmetrical Multi-Processing (SMP)™ technologies available that has yielded consistent industry-leading performance and price/performance. Layered on top of this advanced SMP kernel is a complete set of standard interfaces and tools that make SCO UnixWare 2.1 one of the purest incarnations of the UNIX operating system available. Also layered onto this kernel is a far-reaching set of networking technologies that allows UnixWare 2.1 to function effectively, and cost-effectively in todays' heterogeneous computing environments. With the addition of powerful graphical administration and user-interface features, UnixWare 2.1 stands as a clear symbol of the ascendancy of open systems technology in business critical environments.

SCO UnixWare 2.1 Strengths

The introduction of UnixWare 2.1 combines breakthrough SMP engineering from USL, aggressive productization by Novell, and the effective delivery of solutions by SCO. UnixWare 2..1 provides SCO customers with a highly **reliable, advanced, secure, high-performance, easy-to-use, cost effective, totally networked open systems** platform with a **strong migration** path to 64-bit UNIX and beyond.

Reliable	SCO UnixWare 2.1 features a journaling file system and provides enterprise-level RAID features through the On-line Data Manger add-on product.
Advanced	State of the art Symmetrical Multi-Processing, process threading, file-system technology, networking and system management
Secure	SCO UnixWare 2.1 has built-in security that can monitor system, data, directory and file access down to the system call level.
High Performance	SCO UnixWare 2.1 continues to lead the industry in price/performance in key areas like database, transaction processing and file serving, with consistent world-records in key benchmarks like AIM and TPC.
Easy to use	Graphical tools allow administrators to accomplish most tasks through the well-designed graphical interface without sacrificing the power of getting direct access into the internal workings of the system .
Cost Effective	A server platform without hidden costs. World class performance, world-record price/performance and outstanding value
Totally Networked	Full support for TCP/IP, NFS®, NIS, SLIP/PPP as well as Novell® NetWare 3.X and 4.X client support as well as NetWare 4.1 file, print and directory services.
Open	SCO UnixWare 2.1 maintains SVR4 compatibility and is available in source-code for OEMs to adapt for specific computing tasks like embedded systems and real-time applications. This gives application developers and customers the security that something that works on your UNIX system now will keep working as UNIX evolves. SCO UnixWare 2.1 will conform to the UNIX 95 standard later this year.
Migration	SCO will work with HP® and other OEMs to bring the UNIX-on-Intel® platform into the 64-bit age without unacceptable disruption of customer environments.

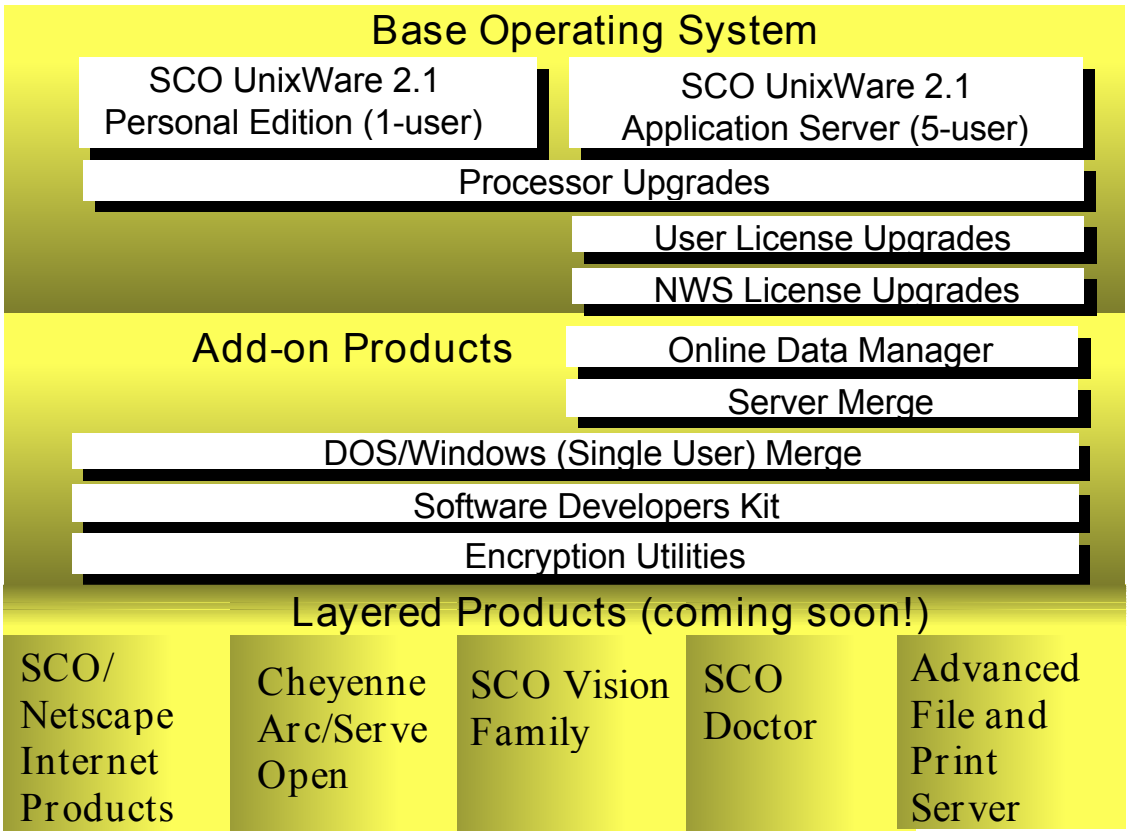
What's new in SCO UnixWare 2.1

SCO UnixWare 2.1 introduces several new enterprise enabling technologies as well as significant new NetWare integration features. These capabilities are summarized below.

High Availability and Reliability	• RAID 5, staged I/O and support for large file systems has been added to the Online Data Manager for UnixWare 2. • Additional high-end availability capabilities are added through efficient handling of multi-level SCSI bus resets and SCSI bus time-outs • Hot insertion and removal of disk and tape drives allows for enhancement a systems hardware capabilities without having to take the system down to do it. • SCO UnixWare 2.1 also support large disks and up to 256 IRQs for larger server environments.
NetWare Integration	• Novell NetWare 4.1 File, Print and Directory Services - SCO UnixWare 2.1 includes popular NetWare 4.1 file, print and directory services. Users can use standard NetWare tools and commands (like NWADMN and NLIST) to administer NetWare services (NWS) on SCO UnixWare 2.1. • Graphical Administration of NWS - setting up and configuring NWS is done through the UnixWare desktop interface. This includes configuring NetWare Directory Services (NDS), the NWS file system and NWS user configuration. • NetWare UNIX Client (NUC) enhancements ⇒ Incorporate NDS awareness into NUC tools (nwlogin, nwlogout, whoami, setpass, nlist) ⇒ Remove dependency on bindery emulation ⇒ Set user context via the nwcx command ⇒ xauto provides authenticated NDS connections to NDS printers and volumes ⇒ standard UNIX mount command to mount NDS objects. ⇒ map NDS objects (named objects) to SCO UnixWare 2.1 directories for seamless access.
Enhanced Networking	• Fast Networking - SCO UnixWare 2.1 introduces bundled support for faster network devices like FDDI and 100 megabit Ethernet. • Shared Interrupt support - SCO UnixWare 2.1 NICs can share interrupts for easier installation and management. • PPP Enhancements - Dynamic Address Configuration and CHAP Authentication has been added which conforms with the majority of Internet providers in the market. • IP Fail-over - lays the groundwork for clustering and future high-availability environments (allows other systems to assume the IP of a system if the primary system goes down. • IP Aliasing - allows multiple IP addresses for a single network interface card. • NIC Configuration - the niccfg command allows the configuration of Network Interface Cards without having to reboot the system or rebuild the entire kernel.
Expanded Hardware Support	• New SMP architectures from IBM®, Compaq®, Olivetti®, Tricord® and Intel® • Support for Intel Multi-processing specification (Spec 1.4) • Support for Pentium Pro™ processor • Support for the latest PCI specification (PCI 2.1)

	• Support for PC-Card Ethernet and modem cards (PCMCIA) • Support for cost effective IDE CD-ROMS • Support for new SCSI and video cards • Support for 30 new Network Interface Cards (NIC)
Installation Enhancements	• Supplemental Installation Diskette - a standard interface to provide patches for kernel modules and installation tools • Network Installation and NIC Configuration ⇒ auto-detection of non-ISA network cards • information preserved for NIC configuration
Improved Software Development	• C++ Compiler and Libraries ⇒ pre-compiled header support ⇒ revised options for template instantiation ⇒ language interpretation updates ⇒ additional keywords ⇒ include most SCO UnixWare 2.1 preprocessor and assembly-level extensions • C and C++ Enhancements ⇒ Pentium Pro processor code generation ⇒ Pentium Pro processor optimizations ⇒ General optimization improvements
DOS/Windows Application Support (through Advanced Merge add-on)	• Supports Enhanced Mode ⇒ Supports Win32s applications ⇒ Additional application support ♦ WordPerfect™ 6.x ♦ NWAdmin ⇒ Windows for Workgroups 3.11 support • TCP/IP support through standard winsock API • Includes NetWare 4.1 Clients for DOS/Windows • New Merge_Setup GUI • More comprehensive documentation • desktop graphics enhancements

SCO UnixWare 2.1 Product Line Overview



The SCO UnixWare 2.1 product line consists of two Operating System offering and a number of add-on products described below. SCO layered products like SCO Doctor and SCO/Netscape® Internet products will not be discussed in this paper.

The **SCO UnixWare 2.1 Application Server** is a multi-user server configuration of UNIX® System V release 4.2MP. It includes support for 5-users and will accept user upgrade licenses of 10, 25, 100, 500, and unlimited users. It can be effectively used as a database server, an Internet node, a factory automation server, a transaction processing engine, a multi-user development host or an application server for departmental or office computing. UnixWare 2.1 also now features NetWare 4.1 file, print and directory services through the NWS component. Other SCO products supported on the Application Server is the SCO Online Data Manager, SCO UnixWare 2.1 Processor Upgrades, SCO NWS License Upgrades, SCO Advanced Merge, SCO Server Merge, SCO UnixWare 2.1 Encryption Utilities, the SCO UnixWare 2.1 Software Developers Kit. These packages will be described below.

The **SCO UnixWare 2.1 Personal Edition** is desktop/workstation configuration of UNIX System V release 4.2MP. It includes support for 2 users and does not accept user license upgrades. It can effectively be used as a development workstation or as a highly reliable client platform for business critical client/server applications. In an environment when frequent desktop reboots are not an option, UnixWare 2.1 Personal Edition makes sense.

The SCO UnixWare 2.1 **Online Data Manager (ODM)** provides enterprise level storage management features for the SCO UnixWare 2.1 Application Server. It provides RAID 0,1 and 5 capabilities, dynamic re-sizing of file systems, dynamic backup and restore and dynamic disk defragmentation. It provides a host of other enterprise-level storage management features that used to be available only in mainframe class systems. These powerful features are delivered in an intuitive and powerful graphical interface bringing advanced storage management down to earth.

The SCO UnixWare 2.1 **Software Developers Kit (SDK)** is a comprehensive development environment featuring C and C++ compilers, graphical debugging tools, performance profiling tools and libraries and headers for all aspects of system and application programming. UnixWare 2.1 continues in the tradition of an open and powerful development platform and provides powerful tools and documentation for the professional developer.

SCO UnixWare 2.1 **Processor Upgrades** allows users to add support for additional processors to their systems. Both the Application Server and the Personal Edition include support for 2 processors. For systems above 2 processors, the Processor Upgrades are used. The upgrades are additive, so a system that has 4 processors would require 2 Processor Upgrades, a 6 processor system would require 4 Processor Upgrades and so on.

SCO UnixWare 2.1 **User Upgrades** are license disks that are also additive in nature. The user upgrades come 10, 25, 50, 100, 500, and unlimited user licenses and are available for the Application Server only.

SCO UnixWare 2.1 **NWS License Upgrades** provide access to the NWS facilities to additional users. Upgrade are available in 5-user additive upgrades.

SCO UnixWare 2.1 **Advanced Merge** provides application support for DOS and MS® Windows® 3.1 applications including enhanced mode applications like Microsoft® Office, WordPerfect Perfect Office and Winsock compliant networking apps written for MS Windows 3.1. It enables users at UnixWare development stations or user desktops to run personal productivity applications while still having direct access to the power of the UNIX environment. It is available for both the Application Server and the Personal Edition.

SCO UnixWare 2.1 **Server Merge** provides multi-user access to MS Windows Applications. This enables UNIX workgroups or X Terminal users access to DOS and Windows applications from a central server, thus reducing the amount of client administration. Server Merge is only available for the Application Server.

The SCO UnixWare 2.1 **Encryption Utilities** is a group of libraries, drivers and commands that allow the encryption of files, messages and data. For businesses, companies or government agencies that need an extra layer of protection and security for sensitive materials like company records, payroll and personal information. The Encryption Utilities is only available for the SCO UnixWare 2.1 Application Server and is not available outside of the US.

SCO UnixWare 2.1 System Overview

Installation requirements SCO UnixWare 2.1

Processor	Intel® 80386 (25 MHz or greater), 80486, Pentium®, Pentium Pro®
Bus architectures	ISA, EISA, Microchannel®, PCI, VESA, PCMCIA
Diskette	3.5-inch diskette required
Display	VGA, SVGA graphics adapters and monitors
Memory	12 MB minimum for Personal Edition 16 MB minimum for Applications Server
Disk space	150 MB minimum for Personal Edition 200 MB minimum for Application Server
Licensing	SCO UnixWare 2.1 Personal Edition - 2 users (not upgradeable) SCO UnixWare 2.1 Application Server - 5 users (upgradeable) User License upgrades available for Application Server of 10, 25, 50, 100, 500 and unlimited
Keyboard/mouse	Standard keyboard. Mouse recommended.
Installation media	CD-ROM or QIC-160 tape drive Network Installation also supported

Application Support

SCO has both amassed a huge catalogue of support applications for the OpenServer® product line and most of those 11,000+ applications will run unmodified on UnixWare 2.1. UnixWare 2.1 also can boast over 3000 native applications with that number growing quickly. Both native applications formats will be supported in Gemini.

Applications supported by UnixWare 2.1 include:

- Native (UnixWare 2)
- Existing UnixWare 1.x
- BSD 4.3 UNIX®
- UNIX System V
- SCO UNIX, XENIX™
- DOS/Windows 3.1 (enhanced mode and winsock apps included)
- Third party applications
- Optional Novell® products

The SCO UnixWare 2.1 Kernel: Inside a high-performance engine

The SCO UnixWare 2.1 operating system is based on highly innovative SMP kernel technology developed by the engineers of UNIX Systems Laboratories® and Novell UNIX Systems Group. It provides a fully parallel environment via multithreading, multitasking, 32-bit computing, asynchronous I/O, and a fully pre-emptable kernel, all with support for multiprocessing.

Some of the key features/services offered by the SCO UnixWare 2.1 operating system in support of powerful application servers include:

- symmetrical multiprocessing (SMP)
- consistent kernel-level and user-level multi-threading
- rich device driver support
- asynchronous I/O
- flexible file system support

UnixWare 2.1 Symmetrical Multi-Processing and Threads

As mentioned above, the SCO UnixWare 2.1 kernel was designed and developed as a multi-processor system at a time when multi-processing technology is becoming remarkably more powerful, yet less expensive (especially on Intel architectures) Working closely with Intel Corporation and other SMP pioneers, engineers from USL and Novell developed a kernel that is portable and flexible enough to quickly support emerging next-generation SMP server architectures from a number of leading companies. The result was the SVR4.2MP kernel, the basis for SCO UnixWare 2.1. The product of a long development and testing process, the SCO UnixWare 2.1 kernel was designed to maximize performance and scalability in SMP servers. Since it's release, SCO UnixWare 2.1 has garnered many performance benchmark records and accolades from press and analysts.

SMP-based systems are very flexible and have the potential to minimize processor bottlenecks and allow maximum system efficiency. The SCO UnixWare 2.1 kernel is designed to take advantage of any multiprocessing hardware design using the Intel architecture. Key technologies that make the UnixWare 2 kernel so outstanding include:

- **Re-entrant code** to enable multiple processors to run without affecting the state of other execution contexts.
- **Fine-grained locks** to protect critical data from simultaneous modification.
- **Fully pre-emptable kernel** allows processes of a higher priority pre-empt any kernel process of a lower priority. The degree of pre-emption in the kernel improves the average process performance of the overall system and of threaded applications in specific (like databases).
- **Multithreading of all major kernel components** to optimize performance.
- **User-level threads -** the inclusion of user-level threads allows SCO engineers to make more components of the operating system SMP-aware. It also allows application developers access to the performance improvements o be gained from taking advantage of SMP capabilities.
- **Lightweight Processes -** a finer gradation of addressable process entities that allow parallel aspects of a single process to be executed simultaneously.
- **Memory-mapped files** are an efficient, transparent way to transfer data from memory to files. This is especially valuable when dealing with large files.
- **Flexible scheduling mechanisms** to support scheduling and load balancing across different processors.
- **Virtual memory management enhancements** to handle multiprocessing, LWPs, and large databases.
- **Processor cache affinity based scheduling** - provides a highly efficient means of scheduling processes in and out of multiple CPUs.
- **Auto configuration and dynamic loading for device drivers** allows portions of the kernel to be loaded, services and then reloaded without having to rebuild the kernel or shut the machine down.

SCO UnixWare 2.1 Device Driver Support

SCO UnixWare 2.1 provides three device driver interfaces, all of which are based on a single specification, the Device Driver Interface/Device Kernel Interface (DDI/DKI). The DDI/DKI is a set of kernel level interfaces designed to reduce the cost of releasing drivers by providing source and binary compatibility guarantees. In addition to the traditional block and character device driver interfaces, and the STREAMS™ interface introduced to support network device drivers, SCO UnixWare 2.1 provides the Portable Device Interface (PDI) for the support of storage device drivers.

The three major types of device drivers supported in SCO UnixWare 2.1 are:

- Host Bus Adapter (HBA) - conforms to PDI framework.
- network - STREAMS-based Data Link Provider Interface (DLPI) or the Novell Open Data-Link Interface (ODI) specifications
- video drivers - use UNIX Screen Interface (SI) standards to support multiple Screen Display Drivers (SDD).

The Portable Device Interface (PDI) is a framework, architecture, and set of kernel interfaces for developing storage device drivers. It supports a large class of devices including hard disks, tape drives, CD-ROM drives, and WORM drives. New drivers can easily be developed, installed, configured and administered using this framework.

Asynchronous I/O

Asynchronous I/O provides a user process with the ability to overlap processing with I/O operations. This is particularly significant in real time, database and transaction processing environments where throughput and determinism on a per process/application basis are important. The asynchronous I/O APIs provided in SCO UnixWare 2.1 conform to the latest Draft of the POSIX 1003.1b Real-time Extensions Standard. Asynchronous I/O may be used by single-threaded or multithreaded applications. Leading database vendors depend on Asynchronous I/O to gain application performance and scalability.

UnixWare 2.1 File System Support

The UnixWare system supports a variety of filesystem types using the Virtual File System (VFS) architecture. This modular architecture facilitates the integration of new filesystem types and makes it possible to support the dynamic loading of file system modules into a running kernel.

The UnixWare system supports three major categories of file systems: disk-based, distributed and pseudo. Disk-based file system types supported include:

- VERITAS file system (VjFS)*
- UNIX file system (UFS)*
- Secure file system (SFS)*
- System V file system (S5)*
- CD-ROM file system (CDFS)*
- Boot file system (BFS)
- Memory file system (memfs)
- DOS File System (dosfs)
- XENIX filesystem (xxfs)*

* denotes multi-threaded component for increased performance

Distributed file systems provide the user access to files stored on remote systems. The file system types supported include:

- Network File System (NFS®)

- NetWare UNIX Client file system (NUCFS)

Pseudo file systems provide an easy-to-use internal interface to the programmer, and are typically not used directly by a user. The pseudo file system types supported include:

- Proc filesystem (/proc)
- Processor filesystem (profs)
- Fifo file system (fifofs)
- File descriptor file system (fdfs)
- Special device file system (specfs)
- Name file system (namefs)
- Xenix semaphore and shared data segment file system (xnamfs)

All of the basic filesystem code in the kernel has been multithreaded and uses fine-grained locking to boost overall system performance. The VERITAS journaling file system (VjFS) is installed as the default disk-based filesystem to improve filesystem reliability and support fast filesystem recovery when the system is booted. The S5, UFS, XXFS and SFS file systems are supported for backward compatibility and security.

The Network File System (NFS) is supported to provide access to files on other systems on the network and is the de facto standard distributed file system in UNIX environments. Integrated NetWare UNIX Client (NUC) file system support allows transparent access to NetWare files. To enhance access to PC file systems, integrated support of DOS FAT file systems is also provided.

Most database and transaction processing applications do not use the UNIX file system because of its integrity and performance limitations. They typically implement highly optimized, cache management algorithms which are not benefited by using the UNIX buffer cache. The UnixWare system solves the performance problem by allowing direct I/O to bypass the file system cache for I/O operations being performed on files used by database systems.

The Veritas File System

The Veritas file system (vjfs) is the default file system for UnixWare 2.1. Among the benefits of the Veritas File System are:

- quick recovery on system startup through **intent logs**
- robustness due to full structural verification of the filesystem on reboot
- support of shutdown using power off
- high-performance
- transparency to users and applications
- support as the root file system
- dynamically loadable
- support of Access Control Lists (ACLs).

The Veritas file system is an **extent-based** file system in which contiguous disk space is allocated to a file, as opposed to a block at a time. Using extents allows sequential I/O operations to be considerably faster than block-at-a-time operations. The resulting throughput can more than double traditional UNIX file system performance.

The Veritas file system provides recovery only seconds after a system failure by using a tracking feature called **intent logging**. This is a logging scheme that records pending changes to the file system structure in an **intent log**. Upon recovery from a system failure, the Veritas File System scans the intent log, nullifying or completing the file system operations that were active when the system failed. A full structural verification of the file system is not necessary. The mechanism is transparent to the user and the system administrator. The result is a file system that is much less likely to be damaged due to catastrophic system shutdown like power spikes or failures.

CD-ROM Filesystem (cdfs)

The **cdfs** filesystem type supports file systems residing on CD-ROM devices and supports:

- High Sierra and International Standards Organization (ISO) 9660 CD-ROM file system format specifications
- SUSP extensions to ISO-9660
- XCDR Support Component extensions to ISO-9660.

The **cdfs** contains a Volume Descriptor List which contains many volume descriptors. It provides dynamic transparent access to files in all of the above formats.

Network File System (NFS)

NFS is the de facto standard distributed file system in the computing community and allows computer systems, both UNIX systems and proprietary environments, to share files in a transparent manner. SCO UnixWare 2.1 provides a multi-threaded implementation of most NFS-related processes (like lockd, nfsd et al). SCO UnixWare 2.1 also provides for dynamically creation and deletion of NFS-related processes to dynamically balance NFS loads.

NetWare UNIX Client File System (NUCFS)

The NetWare UNIX Client file system (NUCFS) is a UnixWare network file system that provides NetWare file services to a UnixWare platform, giving UnixWare system users direct and transparent access to NetWare files and directories on remote NetWare servers. The NUCFS file system distributes UnixWare file service requests which reference file services on a remote NetWare server platform to the NetWare network operating system. Other features of NetWare integration and interoperability will be discussed below in the section entitled **UnixWare 2.1 NetWare Integration**.

UnixWare 2.1 Networking

The UnixWare system includes the standard Internet protocol suite, utilities, and services, as well as an integrated set of NetWare network services. The built-in services for both IPX/SPX and TCP/IP networks allow UnixWare to be fully integrated and connected in a distributed environment. Add to this SCO's Advanced File and Print Services™ for UnixWare and customers will find that the UnixWare system will enhance all popular network environments.

The Internet networking facilities supported in the UnixWare system include:

- TCP/IP protocol suite
- Internet application services
- Network API services
- Service Access Facility (SAF)
- Authentication services
- Network File System (NFS) (see UnixWare 2.1 File Systems)
- Name services (DNS and NIS)
- Remote printer support (System V, BSD, NetWare) (see UnixWare 2.1 Printing)
- Mail services (SMTP, UUCP, MHS) (see UnixWare 2.1 Electronic Mail)
- Network management (SNMP, SMUX) (See UnixWare 2.1 Network Management)

TCP/IP Protocols

The TCP/IP communications protocol suite supports packet-switching networks, connecting machines in a local-area network (LAN) and connecting LANs across wide-area networks (WANs) to enterprise-wide networks. The Transmission Control Protocol (TCP) and Internet Protocol (IP) are industry- and government-standard communications protocols and are the foundation of The Internet and many network-based applications.

Members of the Internet protocol family in UnixWare 2.1 are:

Internet Protocol (IP)	standard internet datagram delivery protocol and addressing method
Transmission Control Protocol (TCP)	standard internet connection-oriented protocol. Provides reliable, flow controlled, in-order two-way transmission of data
User Datagram Protocol (UDP)	high-speed, connectionless datagram protocol
Address Resolution Protocol (ARP)	maps IP addresses to Ethernet addresses
Reverse Address Resolution Protocol (RARP)	maps Ethernet addresses to IP addresses
Internet Control Message Protocol (ICMP)	used to report status and errors in IP communication
File Transfer Protocol (FTP, TFTP)	standard internet protocol for uploading and downloading files
Routing Information Protocol (RIP), Exterior Gateway Protocol (EGP)	standard internet routing protocols
Bootstrap Protocol (BOOTP)	protocol that allows multiple systems to have IP addresses dynamically assigned by a BOOTP server.
Serial Line IP (SLIP), Point-to-Point Protocol (PPP)	standard point-to-point protocols for delivering IP over serial links

Internet Application Services

The TCP/IP protocols are used by Internet applications, the X Windows System™ and the NFS distributed file system.

The Internet applications supported on the UnixWare system include:

- ARPA utilities: telnet, ftp
- Berkeley r* utilities: rlogin, rcp, rsh, rwho, ruptime, rdist, finger
- Berkeley lpd printer daemon
- Router programs: route, routed, gated
- BIND domain name service (DNS)
- Berkeley sendmail
- Simple Mail Transfer Protocol (SMTP)
- BNU/UUCP
- Remote execution facility (REXEC)
- Simple Network Management Protocol (SNMP) with SMUX extensions.

Network API Services

SCO UnixWare 2.1 supports three key network service Application Programming Interfaces (APIs) for use by application programs which require access to distributed services. They are:

- Direct Transport Service (sockets and TLI/XTI)
- Remote Procedure Call Service
- NetWare NOS Call Service

The **Direct Transport Service** is implemented via the Transport Layer Interface (TLI) and X/Open® Transport Interface (XTI), and Sockets Interface. TLI/XTI is designed to make applications independent of the underlying transport protocol, while the sockets interface is more closely tied to the TCP/IP suite of transport protocols.

The **Remote Procedure Call Service** is implemented via the Transport Independent Remote Procedure Call (TI-RPC) API. The TI-RPC distributes procedure calls from client applications to server application procedures. RPC has been made transport-independent to facilitate the portability of distributed applications across networks with varying transport layers. Thus, client-server applications can use the RPC interfaces regardless of the underlying transport layer (IP, NetWare, etc.).

The **NetWare NOS Call Service** is implemented via the NetWare Calls (NWCalls) API. It allows application programmers to sends requests into the UnixWare system kernel for connection management services to NetWare servers including attachment, authentication, task threading, serialization, and reliable delivery.

Service Access Facility

SCO UnixWare 2.1 supports two distinct styles of network service provision. One is the Berkeley style, implemented as "r* commands", and available on most UNIX systems.. The other style is the new Networking Applications Architecture (NAA). This architecture allows applications to run on different networks and still present a consistent interface to the user. The port monitors are managed by the **Service Access Facility (SAF)** on the server.

The Service Access Facility (SAF) provides a flexible administrative framework for managing service requests received on physical ports connected to computer systems running UnixWare. SAF manages connection requests from outside the system and issues instructions to the various system processes that create and maintain the connections.

Each port (terminal or network connection) on a UnixWare system can have a monitor associated with it that can be used to make the port active and otherwise manage its use. The monitor for a port grants/denies access, keeps track of usage, and provides information about availability to administrative programs that need to use the port. Two basic port monitors are provided with the system:

ttymon	serial port monitor
listen	network monitor that provides connection services regardless of a network's communications protocol.

The **ttymon** port monitor performs the functions that getty and uugetty performed in earlier releases of the UNIX system. It has the following functions:

- initializes and monitors terminal ports
- sets terminal modes and line speeds for each port it monitors
- identifies and authenticates users
- invokes the service associated with a given port whenever it receives a connection request on that port.

The use of ttymon provides much more flexibility than the use of getty/uugetty. It can provide a configurable set of services in addition to login service.

The **listen** port monitor monitors a connection-oriented transport network, receiving incoming requests, accepting them, and invoking the services that have been requested. It differs from other listeners (such as inetd) in several important ways:

- it allows private addresses for services.
- passes connections to standing servers.
- supports socket-based services.
- supports RPC-based services and dynamic addressing.

Authentication Services

Identification and authentication services are supported in SCO UnixWare 2.1 to maintain network application security. A basic connection server process is run to support the establishment of connections for all network services that communicate over network and dial-up connections. The connection server :

- receives requests for network services from client machine applications,
- establishes connections to the server-machine ports associated with the requested services
- passes the connections back to the application.

Before passing a connection to an application, the connection server may invoke an authentication scheme.. The cr1 identification and authentication scheme is used on SCO UnixWare 2.1 to protect systems from unauthorized access. cr1 uses DES encryption by default, but it can use ENIGMA encryption and other encryption schemes as well to satisfy export restrictions.

Name Services

Distributed naming services are necessary to achieve a fully interconnected enterprise. SCO UnixWare 2.1 provides a complete set of name services via the **Network Information Service (NIS)** system and the **Domain Name System (DNS)** facility. These are the two most widely-implemented distributed lookup services supported on UNIX platforms. The use of these name services allows a user to access a service anywhere within the enterprise or even external to the enterprise by using the appropriate name service to locate and access the service.

DNS provides a distributed lookup service that allows TCP/IP network administrators to centralize the information regarding which host names map to which TCP/IP addresses on an IP network. DNS is the main naming service for The Internet. Large DNS databases are distributed throughout the world to map names like “sco.com” or “whitehouse.gov” to their TCP/IP address.

The **NIS** service manages a set of distributed databases that centralize and simplify management of common administrative information on a network, such as user accounts, user groups, host addresses, and network domains. NIS is commonly used to allow UNIX systems to centrally maintain configuration information that can be accessed on the network. It is also commonly used to centralize the administration of UNIX user account information.

UnixWare 2.1 NetWare Integration

A discussion of UnixWare 2.1 NetWare integration can be broken down into two logical headings: UnixWare 2.1 as a NetWare client and UnixWare 2.1 as a NetWare Services (NWS) provider.

UnixWare 2.1 as a NetWare client

UnixWare provides built-in services for both IPX/SPX and TCP/IP networks. This enables SCO UnixWare 2.1 users to receive and distribute information throughout an organization that has one or both of these networking environments. SCO UnixWare 2.1 Application Servers can easily integrate into existing NetWare auto networks to provide clients with access to corporate data processing.

The standard installation procedures lets SCO UnixWare 2.1 Application Servers connect immediately to an existing NetWare environment, allowing them to share e-mail, data and printers with NetWare. This enables NetWare clients to access line-of-business applications running on the SCO UnixWare 2.1 Application Server using the network services and data located on a NetWare server. NetWare Integration with SCO UnixWare 2.1 provides graphical login, single network login and authentication for easy connection to multiple servers.

The integration of NetWare into SCO UnixWare 2.1 is provided by the following major facilities:

- NetWare protocols (IPX, SPXII, NVT2)
- NetWare Protocol Stack Daemon (NPSD)
- Service Advertising Protocol Daemon (SAPD)
- IPX auto discovery
- NetWare UNIX client (NUC)
- NUC File System (NUCFS)
- NUC Auto-Mounter (NUCAM)
- NetWare authentication
- Single UnixWare/NetWare login
- Command line utilities
- NetWare API's
- File name/permission mapping
- NetWare printer support
- Access to NetWare server console
- Remote backup on NetWare server
- SNMP support and instrumentation
- DOS and Windows terminal emulator support
- Remote applications over SPX (client and server support)

NetWare Protocols

The NetWare protocol stack facilitates communication between the UnixWare system and NetWare over IPX. It allows UnixWare clients to communicate with NetWare servers and devices. It uses the native NetWare Core Protocols (NCI) to deliver requests over the network to the NetWare kernel.

In addition, the NetWare protocols allows NetWare DOS clients to access SCO UnixWare services via terminal emulation, and it allows DOS emulators to access NetWare services. Components of the NetWare protocols are as follows

- IPX/RIP driver
- SPXII driver
- NVT2
- NetWare Protocol Stack Daemon (NPSD)
- Server Advertising Protocol Daemon (SAPD)
- NetWare UNIX Client (NUC) driver requester

The **Internetwork Packet Exchange (IPX**) protocol provides NetWare's native datagram service. It allows communication as an unreliable, connectionless, datagram protocol among network nodes. The IPX driver in UnixWare implements both IPX and **RIP (Router Information Protocol).**

Enhanced **SPXII** provides a connection-oriented, reliable session protocol and functions on top of the IPX protocol, adding session services. In addition to the services provided by SPX, SPXII supports true protocol windows and negotiation of packet size upon the establishment of a connection.

Novell Virtual Terminal 2 (NVT2) establishes connections between DOS and MS Windows workstations and UnixWare host systems by guaranteeing packet delivery and providing connection-oriented services.

The **NetWare Protocol Stack Daemon (NPSD)** starts up and links the protocol stacks required for NetWare, as well as starting the SAP daemon and the NetWare management stack daemon.

The **Service Advertising Protocol (SAP) Daemon (SAPD)** collects service advertising packets that can be accessed to perform server name to address mapping and get a list of advertised services. It also performs those functions of the SAP agent that are required for IPX routers, such as building and maintaining the Server Information Table

When UnixWare is installed on a system with a Network Interface Card already installed, the **auto-discovery** feature is used to automatically detect the IPX network number, frame type, and network device name for the network(s) connected to the node.

NetWare UNIX Client (NUC)

The **NetWare UNIX Client (NUC)** enables SCO UnixWare 2.1 to be a client to NetWare servers (3.x and 4.x). The NUC permits UnixWare users to access a NetWare file system as if it was mounted on a UnixWare system. It also supports the ability to print on a NetWare printer from a UnixWare system. The NUC provides NetWare Core Protocol (NCP) connectivity from a SCO UnixWare 2.1 client to NetWare servers.

NUC features are enhanced over those supported on other NetWare client stations (DOS, MAC™, OS/2®, Windows) by the multi-user, multitasking and multiprocessor capabilities of UnixWare. The NUC system provides a multi-user, multitasking, multiserver client environment for UnixWare users. Each user on the local UnixWare system can access remote resources on many different servers simultaneously.

NUC maps UnixWare services into NetWare, providing a transparent interface between the UnixWare operating system and the NetWare operating system, giving the user access to directories, files and printers on remote NetWare servers. Local UNIX operating system service semantics are preserved without compromising NetWare security.

Major components of the NUC are:

Graphical NUC client user interface	desktop sup[port for NetWare client features
NUC drivers and libraries	name-to-address mapping and kernel interfaces
NUC daemon (NUCD)	coordinates mounting and file system services on UnixWare (NUCFS, NUCAM)
NUC file system (NUCFS)	networked NetWare file system
NUC automounter (NUCAM)	controls automatic mounting and unmounting of NetWare Volumes
command line utilities	provide command-line users access to NetWare services.

NUC File System (NUCFS)

The NUCFS is a SCO UnixWare 2.1 network file system (see Section 4.2 for more details) that provides access to NetWare file services to a UnixWare platform, providing UnixWare system users with direct access to NetWare files and directories on remote NetWare servers as if they are local. The NUCFS is a stateful file system, i.e., the NetWare server knows about its clients and keeps information about the state of operation between it and its clients.

NUC Auto-Mounter (NUCAM)

NetWare volumes may be mounted using the standard UNIX mount(1) utility, e.g.

```
$ mount -F nucfs server_name/volume_name: /path
```

or using the advanced auto mounting capability provided by the **NUC Auto-Mounter (NUCAM).** The NUCAM automatically mounts NUCFS file systems. The UnixWare Desktop Manager File Viewer or the command-line interfaces, e.g. cd(1), can be used to navigate a NetWare volume. Double-clicking on a NetWare server folder causes the auto-authenticator panel to appear if the user is not already authenticated to the NetWare server.

The **single login** feature is supported by the synchronization of names and passwords between a NetWare server and UnixWare. Any UnixWare utility which performs user login (such as login, rlogin, telnet, rsh, nvt, or the Desktop graphical login) stores the user's name and password in a form that can be retrieved by the NUC.

Command Line Utilities

In addition to the graphical interface provided for the NUC, the following command line utilities are provided:

- nlist - displays list of NetWare users, file servers, or volumes.
- nwlogin - allows a user to log in to a NetWare server.
- nwlogout - allows a user to log out of a NetWare server.
- nwwhoami - displays the NetWare server to which a user is attached.
- setpass - allows a user to change his/her password on a NetWare server.
- mount_nucam - allows user to mount the nucam file system.
- mount_nucfs - allows user to mount nucfs file system.
- nwprimserver - allows user to access and set the primary NetWare server.

NetWare API's

The SCO UnixWare 2.1 NWCalls APIs track the DOS client NetWare 4.1 API set, increasing application portability. The APIs have the same names and parameters on all client platforms (SCO UnixWare 2.1, DOS, Windows, OS/2, NT), and the same NCP engine is used on all platforms. Services provided by these APIs include:

accounting	bindery	bindery	bindery	subdirectory
directory entry	file I/O	file server statistics	name space	file server
synchronization	transaction tracking	transaction	directory services	auditing
		volume		

File Name/Permission Mapping

The methods that NetWare uses to control access to files and directories are different from those used by the UnixWare system. NetWare maintains more control over files and directories and sometimes the filename and access control mapping between the two environments is not exact. The mapping of these differing access controls is handled by the NUC and the **NUC.NLM**. The NUC.NLM is a module that runs on the NetWare server that maintains file permissions and ownerships for UNIX users who access them. NUC and NUC.NLM together provide the ability to access files on NetWare volumes and a permissions or rights translation mechanism that allows users to use SCO UnixWare commands to view and change access control attributes of NetWare files.

Access to NetWare Server Console

The NUC provides the ability to access a NetWare server console from UnixWare. This requires the loading of a number of NLM's from the UnixWare system onto the NetWare server. Once this is done, the administrator can use telnet(1) from a UnixWare terminal window, or use the **Xconsole** application from the Desktop, to act as a remote NetWare server console. This allows the administrator to execute console commands, load NLMs, and perform other functions available only from the NetWare server console.

Remote Backup on NetWare Servers

A UnixWare **Target Service Agent (TSA)** permits a UnixWare system to participate as a client of NetWare's Storage Management Services (SMS). The TSA supports remote backup and restore of UnixWare files as well as of NetWare files.

SNMP Support and Instrumentation

UnixWare 2.1 includes SNMP instrumentation, tools and modules that enable client applications to manage and view many of the UnixWare client resources over either TCP/IP or IPX/SPX. This includes management consoles like Novell's Network Management System (NMS™) IBM® Netview™ and HP® Openview™ products.

DOS and Windows Terminal Emulator Support

UnixWare includes two terminal emulation packages. These can be used from a NetWare client running DOS or MS Windows to log in to a UnixWare system on a NetWare network using IPX/SPX protocols. Host Presenter runs in the MS Windows 3.x environment while TNVT220 runs in the DOS environment. Each of the terminal emulation packages enables a PC station to emulate a DEC VT220/VT100/VT52 or PC ANSI terminal.

NetWare Services for SCO UnixWare 2.1 (NWS)

System Requirements for NetWare Services (NWS)

System Requirements	UnixWare 2.1
Installation	bundled with UnixWare 2.1 CD-ROM, or supplemental QIC-150 tape
RAM	24 MB + 1MB for each 10 NWS clients
Licensing	1-user license is included for evaluation purposes. User license upgrades are available in 5 user additive licenses
Disk Requirements	NetWare Server - 70 MB NetWare Docs - 62 MB NetWare Clients - 16MB = 148 MB+

NetWare Services (NWS) overview

The NetWare Services (NWS) provides Novell's NetWare 4.1 File, Print and Directory services on the UnixWare 2.1 platform. The features and components that are included with this technology include:

- NetWare Directory Services™ 4.1 (NDS)™
- NetWare 4.1 File Services
- NetWare 4.1 Print Services
- Graphical Administration
- Integrated backup and restore
- Hybrid user management
- User Licensing

NetWare 4.1 Directory Services™ (NDS)

NetWare Directory Services (NDS) is a database of network resources that can be displayed and manipulated in a direct and systematic way. The NDS database is distributed (that is, different portions of

the databases can be on different computers and in different locations). The database is also replicated so that several copies of all or parts of the tree can be kept to remove the danger of a single point of failure if a server housing the database went down. NDS provides direct and managed access to all network resources regardless of where they are physically located. This presents a single information system to users and administrators.

One of the many benefits of NDS is that users no longer must log into each server they wish to access. Users log directly into the network as a whole and will have automatic access to servers, printers and volumes that the network administrator allows them to see. This broader view of the network is much easier to manage since network objects can be manipulated graphically using the standard NetWare 4.1 administration tools.

The directory consists of **objects,** which are the elemental pieces of the network. Examples of objects would include users and printers but would also include print queues, disk volumes and even NetWare servers.

There are numerous different objects types and these are differentiated by the types of **properties** these objects have. Object properties define what an object is and, to an extent, how it acts within the network. The nature of specific objects is held in the object's **values**. So, for example, when a new user is added to the network, a new object of type "user" is added to the NDS database, some of the user object's properties would include user name, phone number and FAX number and the new user object would be assigned appropriate values for the name, phone number and so on.

NetWare 4.1 File Services

With **NetWare 4.1 File Services,** NWS provides network storage capabilities for standard NetWare clients. NetWare clients would notice no difference between a native NetWare 4.1 server and NWS in the normal operations of saving and retrieving files over the network. Files reside on standard UnixWare file systems and can be manipulated using standard UNIX tools as well as NetWare client tools like **ndir** and **ncopy**.

NetWare uses **name spaces** to support native file services for a variety of different clients. This is necessary because of the different file storage requirements of the different NetWare clients. For instance, DOS only supports file names of 8 characters with a 3 character extension, while UNIX allows 256 character names and Macintosh™ supports long names with embedded spaces. NWS supports names spaces for DOS/Windows 3.1, Macintosh, UNIX and OS/2 (note: Windows 95 and Windows NT use OS/2 name space).

Access Control to NWS volumes can be handled by either the UnixWare system or by the NetWare mechanism. This is configurable through the **NetWare Volume Setup** desktop tool that is installed with the system.

NetWare 4.1 Print Services

NetWare 4.1 print services are totally integrated with native UNIX print services. UnixWare 2.1 users can utilize standard NetWare print services including **print servers**, **print queues** and printers. Print Servers are processes running either on UnixWare or on NetWare that route print requests to the correct queues and printers. Print Queues are temporary holding areas where print jobs reside before they are sent to a printer.

Because NWS provides standard NetWare 4.1 printing, users have access to printers anywhere they reside on a NetWare or a UnixWare server. This was accomplished by implementing the NetWare **NPrinter** process on UnixWare 2.1. And because of the NDS features within NWS, NetWare administrators can use standard NDS tools to add new printers to the UnixWare environment and control user access to those printers.

Graphical Administration of NWS

The administration of NWS is accomplished by standard NetWare 4.1 administrative tools like Nwadmin as well as a number of easy to use tools on the UnixWare desktop:

NetWare_Setup - provides a single view of all NetWare-related administrative capabilities on UnixWare 2.1. It includes the ability to administer parameters relating to IPX/SPX, NWS, NUC, network management and Print Server functions.
NWS_Status - shows operating status of NWS and allows administrators to turn the server on and off, and send messages to NetWare client screens
NWS_Volume_Setup - allows users to create, delete, repair, optimize and get detailed information about NWS volumes
DS_Install - allows administrators to install or remove NDS databases in NWS.
DS_Repair - allows standard NetWare repair facilities to be run on the NDS tree created on NWS systems.
NWS_Licensing - allows for adding or deleting user license upgrades for NWS client access.
NetWare_Client_Disks - allows users to create standard NetWare 4.1 client disks for DOS/Windows and OS/2 clients.

NetWare 4.1 Volume Management

NetWare volumes are analogous to UNIX partitions. Volumes contain the root of the NetWare file system and a single NetWare server can have up to 64 volumes. NWS utilizes the **NetWare Portable File System (NPFS)** to implement NetWare volumes on UnixWare. The NPFS is a software layer that allows native UnixWare file systems (like vxfs and ufs) to act as NetWare volumes. NPFS keeps track of NetWare specific filesystem (volume) features and emulates those features on top of the UNIX file system.

NWS **volume management** is done using the **NWS_Volume_Setup** desktop utility. This utility allows administrators create, delete, repair and optimize NWS volumes. By default, NWS allows up to 10 volumes to be specified although the system can be tuned to allow up to 64 (the current NetWare 4.1 limit).

NWS User Management

NWS introduces the concept of **Hybrid User** which provides a linkage for users wanting to access files both as NetWare clients and UNIX users. The hybrid user function handles situations of assigning file and directory ownerships when a user exists in both the UnixWare and the NetWare environments. Administration of this feature is done through the UnixWare 2.1 graphical administration tools. Users will have normal, **native** access to files and directories in both environments.

NWS Backup and Restore

All files and components of the NWS environment can be backed up using standard UNIX backup technologies that have been enhanced to take NetWare extended file attributes into account. NDS can also be backed up and restored using tools supplied with the system.

UnixWare 2.1 Installation

Installation of the UnixWare 2.1 system is fast and largely automatic. Designed to be easy, intuitive and flexible the installation process features:

- **Installation is Fast and Easy -** installation of the system can be accomplished in under an hour without extensive training or even UNIX experience. Easy menus and graphics walk the user through the install while still allowing expert user a great deal of flexibility.
- **Auto-configuration** to identify and configure platform and peripheral hardware with minimum user interaction
- A **Device Configuration Utility (DCU)** that allows administrators to view and change hardware configuration settings during the installation process.
- A **single boot floppy** that talks to the BIOS, contains the bootstrap and sets up the installation interface

- **Network installation** from a UnixWare Application Server on either an IPX/SPX or a TCP/IP network
- Extensive use of **default parameter** values eliminating the need for most user input
- "**Front loaded" user interaction** so that configuration information is supplied without waiting between lengthy install operations
- **Non-destructive upgrade** and overlay installations

Device drivers for the most common hardware are provided on the installation diskette. Additional SCSI and non-SCSI drivers are supplied on the Host Bus Adapter Drivers diskette supplied with the system. Manufacturers will often make new drivers available to users. If the peripheral you wish to use is not on the current SCO supported hardware list for UnixWare 2, contact the peripheral manufacturer.

UnixWare 2.1 Printing

UnixWare supports a large number of different printer types using both UNIX SVR4- and BSD-style print system commands. Local printers are supported using either direct or dial-up connections. Remote printers are supported using System V, BSD or NetWare IPX/SPX print protocols

One of the distinguishing features of the UnixWare printing system is the tight integration between the traditional UNIX print system and the NetWare print system. Bi-directional printing between the UnixWare and NetWare printing systems is fully supported. While continuing to provide the full functionality of lp, UnixWare provides the necessary connectivity to take advantage of NetWare printers, and for NetWare print servers to utilize printers defined through UnixWare (NPrinter).

The standard lp model that is now part of UNIX SVR4 includes the **lp** and **lpsched** modules. The user command, **lp**, (from "line printer") is used to submit print requests from the user environment to the lp print system. The print scheduler process, **lpsched**, removes print jobs from the print system (queues) and sends them to the appropriate destination. The print system management also includes utilities to configure printers, control printing via forms or filters, and check the status of print requests.

Network print requests on UnixWare are handled by the **lpNet** program, working in conjunction with lpsched. The lpNet program supports the SVR4 protocol, the BSD protocol and the NetWare IPX/SPX protocol, using the printer configuration information to determine which type of communication is appropriate for each request. The network protocol, in turn, takes control of passing print requests to the remote printer system.

In UnixWare, remote printers, and their corresponding communication protocols, are defined either by using the printer setup graphical utility, or by using the lpsystem command line utility. The system handles each identically allowing administrators to allow or deny print access to individual users.

A NetWare PServer process is also able to deposit NetWare print jobs into the UnixWare lp print system. This is accomplished with the UnixWare daemon **NPrinter**. The NPrinter process communicates with NetWare PServer processes to pass jobs from the NetWare queues, through NPrinter, into the lp print system. From there, print jobs are handled as any other UNIX print job.

UnixWare 2.1 Electronic Mail

UnixWare mail service provides electronic communications between users on the same computer or between computers connected together on a network. The user may manage (read, compose, send) mail on the local machine using either a Desktop tool provided for this purpose or the familiar "mail" command from the command line. The administration of the mail system can be done using the convenient **Mail_Setup** utility on the UnixWare desktop.

For transmitting mail between networked systems, UnixWare uses the **Simple Mail Transfer Protocol (SMTP**) and the **UNIX-to-UNIX-copy (UUCP)** protocol for Internet and dial-up lines, respectively. The UnixWare mail services have been enhanced to exchange mail with Novell **Message Handling Service (MHS)** using an MHS gateway running on UnixWare.

Using the **MHS gateway**, UnixWare users are given access to all of the messaging environments supported by MHS. These include SNADS for IBM AS/400's and mainframes, X.400 for access to public networks, MHS

internetwork messaging protocol over a standard IPX-connected network for communication with another NetWare Global MHS server or a NetWare MHS host, and MHS asynchronous messaging protocols when using an asynchronous connection. **SMTP** is provided for access to UNIX systems and TCP/IP networks, and also works between UnixWare systems over SPX.

The UnixWare MHS gateway is a set of programs that allows UnixWare and NetWare mail systems to send and receive messages from an MHS mail system. The gateway translates between the Internet RFC822 mail format and the MHS SMF71 (and SMF70) mail formats.

The routing of mail messages on UnixWare systems is basically controlled by the **/bin/rmail** program and a number of configuration files. Incoming mail from SMTP, UUCP or MHS sources is queued and then processed and delivered by the UnixWare mail subsystem.

Outgoing mail is queued by the appropriate outgoing transport programs, or if the destination is unknown, the mail message is routed to a "smarter host" for further processing. A "smarter host" is a UNIX system to which remote mail can be shipped if the mail destination is unknown to the local computer.

The mail system on UnixWare has been enhanced to support a very sophisticated and efficient mail delivery system. A cluster of computers can be established for purposes of mail delivery so that mail sent from any of the computers in the cluster will appear to be from a cluster name, as opposed to an individual member of the cluster, hiding the internal machine names from the outside. NIS and DNS name services can be used to locate addressing information not found locally. Multimedia attachments can be added to messages using the Multipurpose Internet Mail Extensions (MIME) specifications.

UnixWare 2.1 Security

UnixWare has security designed in as an integral part of its system architecture. The system is designed in a modular fashion, with each module itself secured so that it contains protective firewalls. Safeguards at the lowest-level functions provide a foundation for building secure applications on the system. UnixWare is National Computer Security Center C2 certifiable and provides B2 extensions that meet the most demanding government and enterprise system security needs. The user can customize each system configuration to meet his security needs. UnixWare provides a secure operating environment by means of these key features:

- identification and authentication
- Access Control Lists
- least privilege/trusted facility administration
- auditing

The **identification and authentication facility (IAF)** provides a framework for modular replacement of identification and authentication schemes. UnixWare 2.1 supports two different types of schemes: the familiar login/password scheme and a new bilateral scheme called cr1 (Challenge-Response 1).

Access control lists (ACLs) are supported (on SFS and VxFS file systems only) as an extension to **discretionary access control** (traditional UNIX permissions) . ACLs provide finer grained access control (down to a specific user) than existing file permission bits. ACLs allow the owner of a file to grant or deny different access permissions (read, write, execute) to the owner, specific users, specific groups and/or everybody else. ACLs can be associated with files as well as IPC objects such as semaphores, shared memory and message queues.

The concept of **least privilege**, or the ability to execute a system administration task with the least amount of privilege necessary to execute the task, has been introduced into UnixWare. The **least privilege/trusted facility administration** feature splits the single privileged role of root into well-defined, but less powerful roles. This allows system administrators to be assigned specific tasks and have their administrative responsibilities restricted. This avoids inadvertent errors and/or the compromise of confidential information.

Auditing provides the ability to track all pertinent system and user activity that may affect the security of a computer system. This mechanism can be used to identify instances of unauthorized access control and occurrences of unusual activities.

UnixWare 2.1 Network Management

Network Management provides the ability to view and manage many of the UnixWare resources from Network Management SNMP consoles like HP OpenView, IBM Netview or Novell Network Management System (NMS). The Simple Network Management Protocol (SNMP) is the most widely used means for gathering information on the status of individual nodes on the network and making changes in the operational parameters of the nodes. UnixWare SNMP provides a solution for IP- and IPX-based networks. The system administrator can display network information and manage the network on any SNMP management console. Instrumentation, or gathering of statistics, for each object provides network management with the information needed for the management process. Network resources include such components as the file system, requester protocol stacks, and devices. In UnixWare, network management is comprised the following components:

- SNMP agent
- SMUX protocol and library
- MIB files
- MIB compiler (mosy)
- command line utilities

The **SNMP agent** is the module that maintains the database for managed objects, allocates and manages the memory for the database, allows objects to register themselves as managed objects, and sends information about these objects to management consoles, as requested. The SNMP agent runs over UDP/IP and IPX. UnixWare uses the**SNMP Multiplexing protocol (SMUX**) as its instrumentation interface to SNMP. This interface can also be used by third parties for instrumentation.

Management Interface Base (MIB) files form a hierarchical tree of managed objects and attributes. These MIB files can be compiled into a definition file, using the**MIB compiler (mosy)** , that the SNMP agent understands. The objects instrumented in UnixWare include the following:

- Host resource MIB - system, storage, device, software run, software run performance, software installed (RFC 1514).
- NetWare protocol stack MIBs - IPX, RIP/SAP, SPX, diagnostics.
- MIB II - UnixWare protocol stacks, SNMP agent statistics.

Command line utilities are provided to monitor and manage the SNMP agents. These include utilities such as:

- **getone** - get one attribute or object in the MIB.
- **getnext** - get the next attribute or object in the MIB.
- **getmany** - get groups of attributes or objects in the MIB.
- **getroute** - extract routing information from an SNMP entity.
- **setany** - sets one attribute or object.
- **snmpstat** - show network status for the SNMP agent.
- **trap_rece** - receive traps from a remote and/or local SNMP agent.
- **trap_send** - send SNMP traps.

UnixWare 2.1 adds the ability to instrument network management features for the bundled NetWare Services subsystem. This would appear as a separate server on standard network management consoles.

Internationalization (I18N) and Localization (L10N)

Internationalization is the process of making software independent and portable across a variety of languages and regions. UnixWare 2.1 is an internationalized operating system that supports many different language/cultural environments without modification to the base system. Localization is the

process of customizing software for a specific language or region. UnixWare has been localized for Japanese, French, German, Italian, Spanish and English; support for other languages can be added by the user. Multibyte representations of characters as well as single byte representations of non-ASCII characters are supported in UnixWare.

UnixWare supports the concept of a **locale**, a language and country combination, to represent the unique local requirements and provides the mechanism for installing locales and switching among them. Typically a locale describes those conventions that are used every day in business communication and in data processing in a particular country or geographical area. This information is used by the applications running on the system. The conventions include the area of keyboard input and screen output; data, time, and monetary and numeric conventions; character sorting order; and the language of the user interface.

UnixWare supports both a wide range of**code sets** and multiple code sets simultaneously. A code set (or coded character set) is defined as the computer representation of each character or symbol within a set of characters by a unique numeric value. Each language requires a different set of characters and symbols - for instance, the Latin, Greek, Cyrillic alphabets or Japanese ideographs. The code sets which are in use for a particular user are determined by the locale setting. Code sets included with UnixWare 2.1 are:

USA, Western Europe	ISO 8859-1
Japan	eucJP, JIS X201, JIS X208
Other locales	various code sets

Various **keyboard input methods** are supported by UnixWare. This allows the user to enter characters not found on his keyboard by typing in a special sequence of keystrokes. The keystrokes are processed by an "input method" to produce the desired character. All Western European locales use a common input method. The Japanese input method is only available with the Japanese version of the UnixWare product.

The internationalization and localization features of the system provide users with a fully localized operating system and GUI interface from installation through runtime. Localized applications can run on UnixWare 2, enabling users to run UnixWare 2 in the environment in which they are most comfortable and productive.

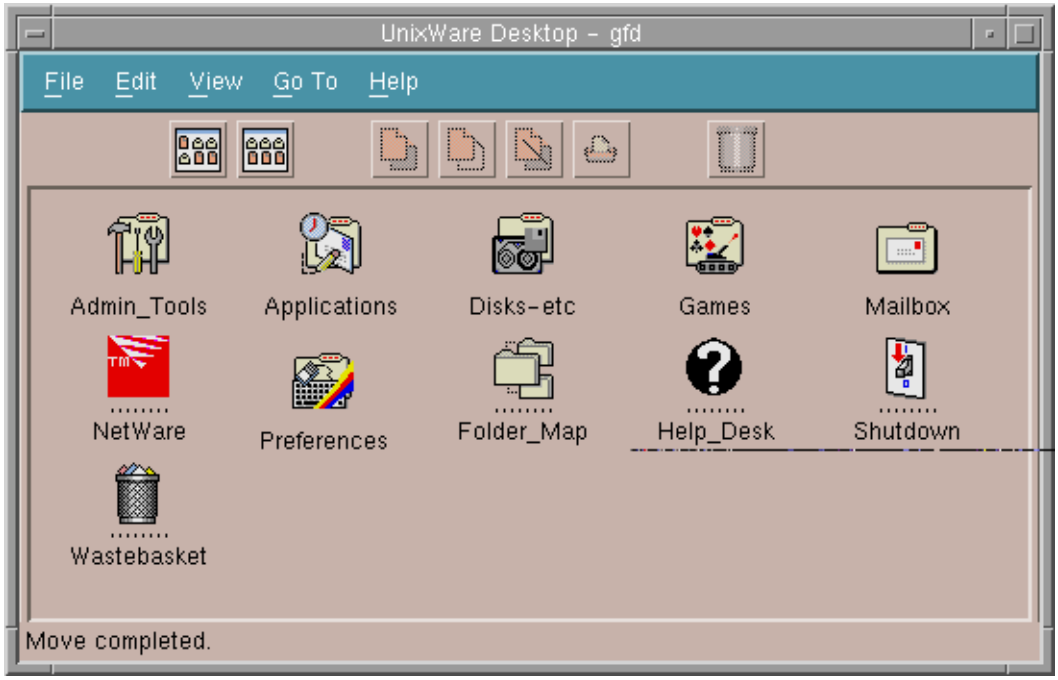
Desktop Utilities and Administration

The SCO UnixWare 2.1 desktop provides powerful tools for creating highly reliable, custom desktop environments. It is based on the**X Windows System ™ (X11r5)** and uses the Motif™ v1.2.3 look and feel as developed by the Open Software Foundation® (OSF).

The UnixWare 2.1 desktop provides context-sensitive on-line help, a point a click interface to select and execute documents and applications, and drag and drop capabilities that include the dragging of documents onto printer or editor icons for their respective actions. This provides a significant improvement over most other UNIX systems while still allowing expert users access to more advances workstation type of desktop functions.

The UnixWare 2.1 desktop functions as a**metaphor** for the underlying system so that files and directories appear as file and folder icons on the desktop. Users still have access to file permissions and other information, but much of the cryptic or complex nature of UNIX is shielded from them.

Desktop Windows

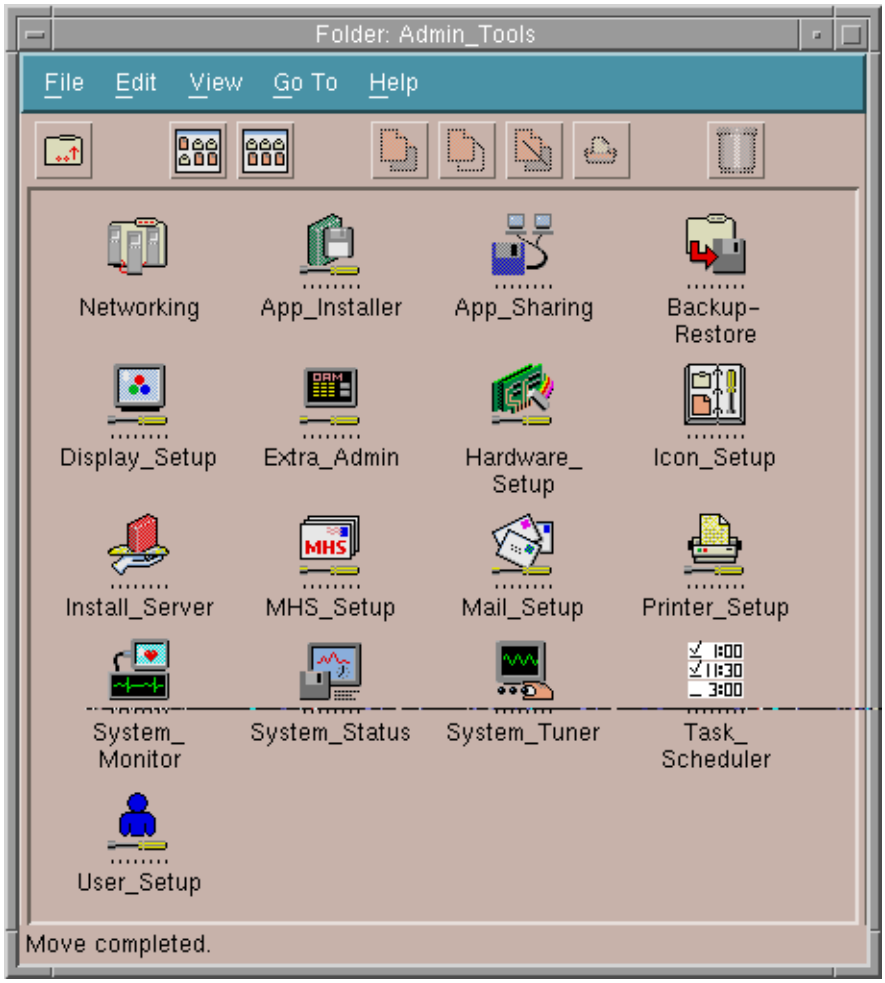


The UnixWare Desktop Window

The "UnixWare Desktop" window appears by default upon login, presenting the user with icons for all of the files, folders (representing directories), applications and utilities that come with UnixWare:

- **Admin Tools**: utilities for system administration
- **Applications**: text editors, command line shells, mail, remote logins, DOS and Windows applications, etc.
- **Disks-etc**: access to tapes, CD ROM and diskettes
- **Games**: Xtetris and Puzzle
- **Mailbox**: Mail application, saved mail, files transferred via uucp
- **NetWare:** access to NetWare servers
- **Preferences:** tools to customize your Desktop
- **Folder Map**: tools to graphically view and manage the file system
- **Help Desk**: on-line help and documentation for each icon
- **Shutdown:** utility to gracefully shutdown the system
- **Wastebasket**: "deleted" files

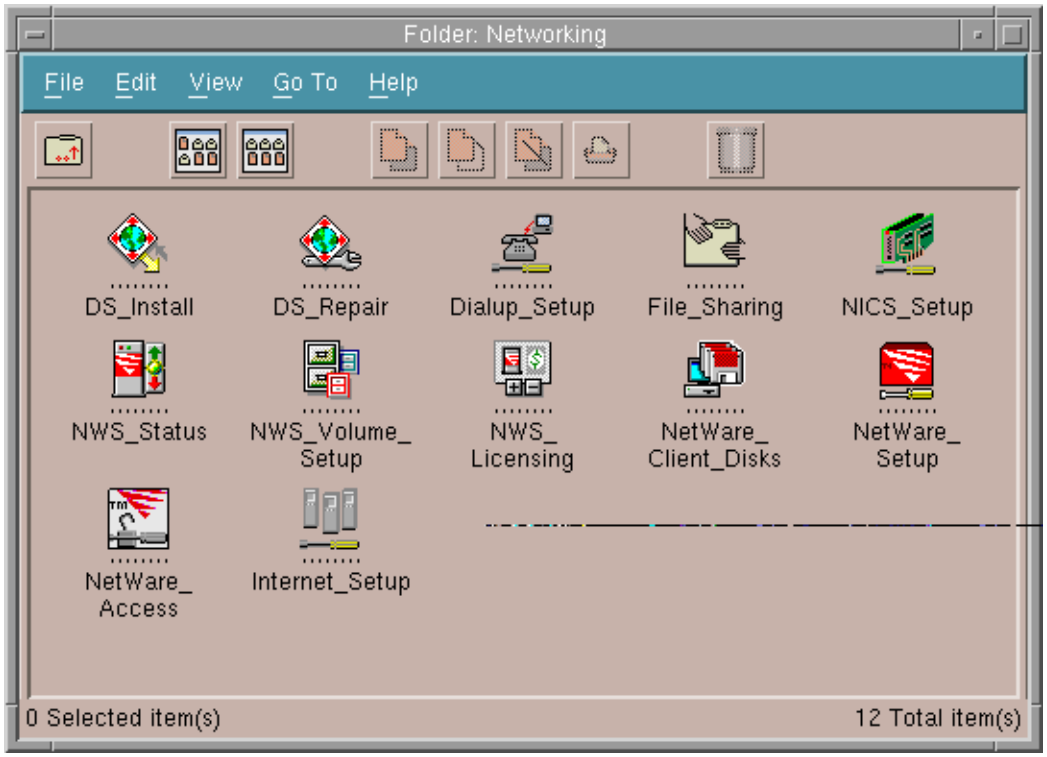
The Admin Tools Folder [desktop/Admin_Tools]



App_Installer - Provides a graphical interface to standard UNIX software packaging interface (pkgadd). This will install application from Corel Draw to Oracle databases using a consistent interface.
App_Sharing - allow users to share applications over the network to other UnixWare system by simply dragging and dropping.
Backup-Restore - provides a graphical way for users to do full or partial backups of the UnixWare system.
Display_Setup - allows users to change the screen resolution of their UnixWare 2.1 desktop and automatically detects the graphics chip-sets that are used on the video board.
Extra_Admin - allows users access to certain legacy administration tools commonly used on earlier UNIX releases. It provides a menu-driven interface to the UNIX sysadm environment.
Hardware_Setup - allows the administrator to view and change hardware driver setting (like IRQs, RAM addresses and DMA channel settings).
Icon_Setup - create icons for applications or commands on the system. It also allows users to customize how the icon acts when it is double-clicked and when a file is dropped on to it.
Install_Server - (Application Server only) - allows administrators to set up the Application Server to install other UnixWare 2.1 systems over the network.
MHS_Setup - (Application Server Only) allows administrators to set up the Application Server as a gateway for MHS/Internet mail users in NetWare environments.
Mail_Setup - (Application Server Only) allows administrators to set up standard UNIX mail through a convenient interface.
Printer_Setup - allows a single, consistent interface for adding local, remote UNIX and remote NetWare system printers. Printer icons can be installed on the desktop so that users can simply drag and drop documents onto the icon to print them out.

Processor_Setup - enables an administrator to enable or disable processors on the system.
System_Monitor - provides real-time performance analysis in a graphical format. Various system processes and parameters can be monitored simultaneously.
System_Status - allows users to view status of disk storage, processor type and other system indicators.
System_Tuner - allows administrators to view and change kernel parameters, displaying current values, system limits and defaults.
Task_Scheduler - can be used as a desktop alarm-clock to notify the user when important events are scheduled.
User _Setup - allows administrators to add, delete or modify users on the UnixWare system.

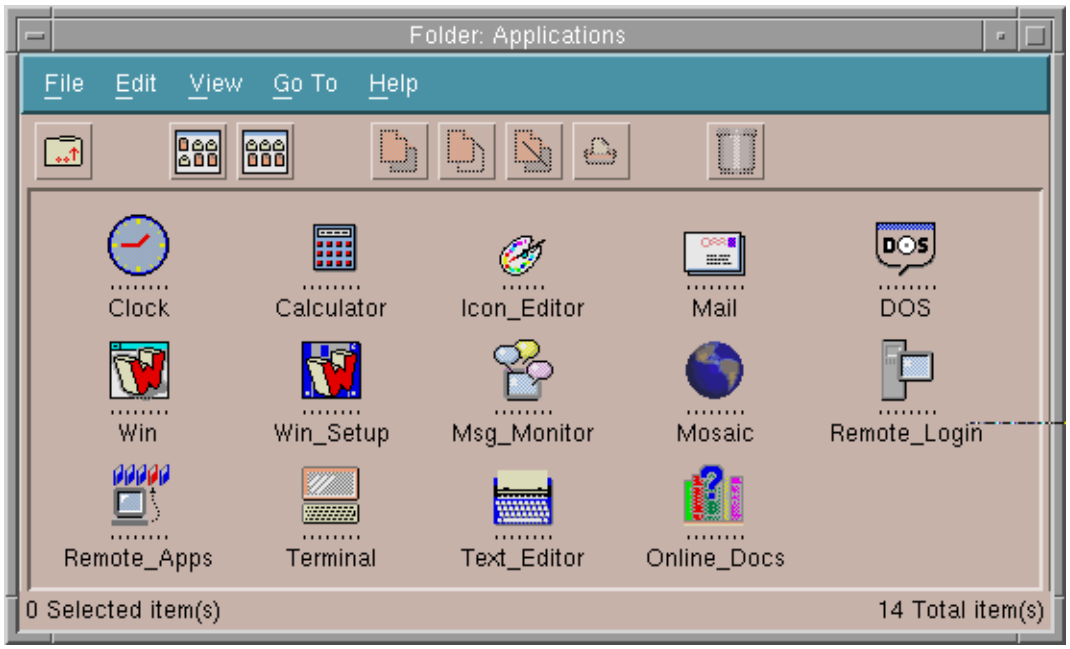
The Networking Folder [desktop/Admin_Tools/Networking]



DS_Install - installs, configures or deletes NetWare Directory Services for the NWS processes.
DS_Repair - the standard NetWare utility that allows administrators to repair NetWare Directory Services if NDS becomes out of synch with other NDS servers.
Dialup_Setup - allows users to configure modems and the standard UUCP (UNIX to UNIX copy) files.
File_Sharing - provides a graphical interface to the Network File System (NFS). This allows administrators to advertise or mount remote resources from the desktop.
NICS_Setup - allows for the addition, deletion and configuration of Network Interface Cards (NIC)
Internet_Setup - Allows users to set up various aspects of TCP/IP operations. This includes routing, netmask, IP addresses, hosts files, DNS client setup and others.
NetWare_Setup - provides a single point of administration for all NetWare client and server configuration. This includes setting IPX/SPX parameters, specifying the internal server address
NetWare_Access - allows users to view the current status of NetWare connections. It allows a user to login, logoff and to query NetWare servers for volume or user information.
NetWare_Client_Disks - allows administrators to create standard NetWare 4.1 client disks. This includes protocol stacks and utilities for access NetWare services.
NWS_Status - allows administrators to start or stop the NWS services. It also allows for sending messages to NetWare client consoles.
NWS_Volume_Setup - allows the creation and administration of NetWare volumes for NWS operations.

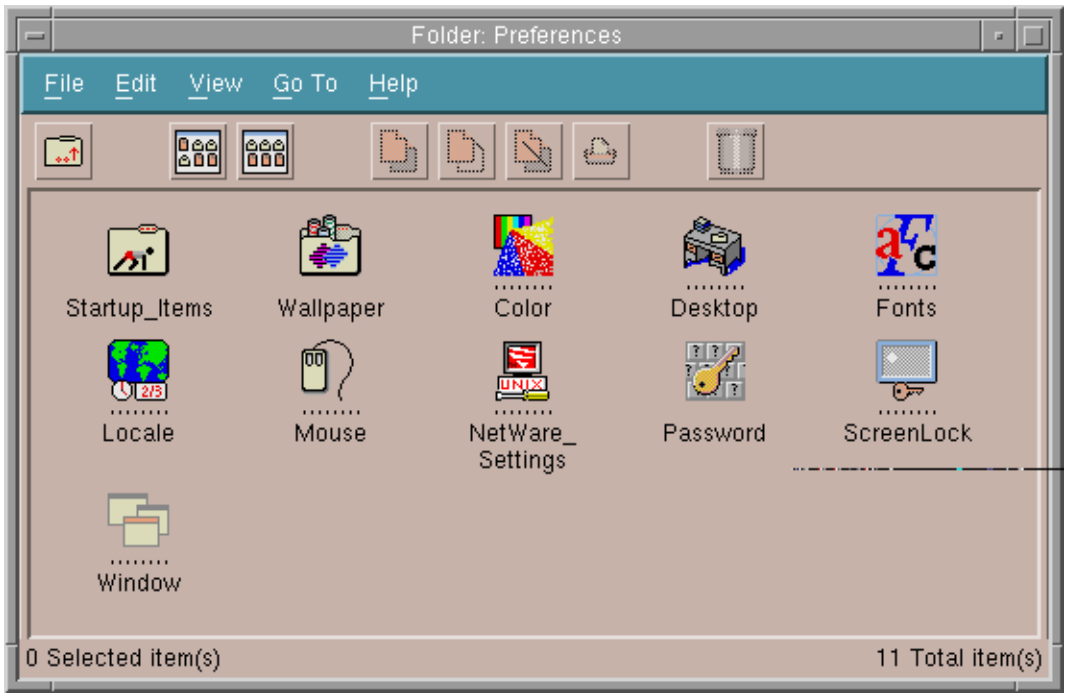
NWS_Licensing - allows administrators to add or delete NWS license upgrades. License upgrades are provided in 5-user, additive licenses.

The Applications Folder [desktop/Applications]



Clock - to display the time on the desktop (even when iconified)
Calculator - to do basic arithmetic operations on the desktop.
Icon_Editor - to customize and colorize desktop icons.
Mail - a full featured graphical e-mail application. Allows aliases, mail folders, cut and paste and other standard mail features.
DOS - allows users to run DOS applications on the desktop (requires Advanced Merge add-on)
Win - allows users to run MS Windows 3.1 applications on the UnixWare desktop (requires Advanced Merge add-on).
Win_Setup - sets up a users MS Windows environment. Users can change the amount of memory Windows uses, binds UNIX printers and COM ports to Windows sessions.
Msg_Monitor - allows users to view UNIX console messages in real time.
Mosaic - runs the NCSA Mosaic graphical World Wide Web browser.
Remote_Login - allows users to create desktop icons that log directly into remote UNIX systems
Remote_Apps - allows a user to execute (or create desktop icons that execute) remote applications shared from other UnixWare system with the App_Sharing tool mentioned above.
Terminal - provides a command line interface window on the desktop. Many can be opened and active at the same time.
Text_Editor - provides a quick and convenient way to write, edit and print text documents and files.
Online_Docs - users can read system documentation and handbooks using this tool. It provides for searching, annotating, marking and printing document pages and uses Dynatext™ technology, the same interfaces used for NetWare on-line documentation.

The Preferences Folder [desktop/Preferences]



The **Preferences** folder holds all of the desktop utilities for customizing and personalizing the user desktop.

The **Startup_Items** folder is a folder where users can drag and drop applications to have the applications execute each time a user logs into the desktop.
The **Wallpaper** utility allows the user to select from a number of backdrops for the desktop environment.
The **Color** utility allows the user to customize the colors of the desktop environment (window borders, text color, window background, etc.
The **Desktop** utility allows a user to customize operations of the desktop like mouse keyboard preferences.
The **Fonts** utility allows the user to choose the icons used by the desktop. Font sizes can also be changed for better readability .
The **Locale** utility allows the user to notify the system of the local for correct time stamps and languages.
The **Mouse** utility allows the user to test and modify mouse settings.
The **Password** utility allows a user to changes his/her UNIX password.
The **Screenlock** utility allows the user to select the type of screen saver to be used when the desktop is left inactive for a pre-determined interval.
The **Window** utility allows the user to specify behavior of the desktop windowing environment.

Online Data Manager Product Overview

Online Data Manager System Requirements

System requirements	UnixWare 2.1 Application Server
Installation	CD-ROM drive required
disk space	5 MB free and .5 MB free on each disk is recommended

Online Data Manager Product Description

The UnixWare 2.1**Online Data Manager (ODM**) provides enterprise-level storage management that is easy and intuitive to use, yet is still provides powerful high-availability features. Features include:

- on-line backup
- on-line administration for defragmentation,
- on-line file system expansion and shrinking
- preallocation and fixed extent sizes
- unbuffered I/O and other caching options
- additional mount options
- NFS acceleration.

One of the key features that make ODM such an invaluable tool for business critical applications is that the majority of its administration and even changes in configuration can be accomplished in real-time without any interruption of applications and users on the system. Large-scale storage management procedures can be accomplished without having to shutdown or reboot the system.

On-line Data Manager components

ODM consists of three components:

- Advanced File System (VxFS)
- Volume Manager (VxVM)
- Visual Administrator (VxVA)

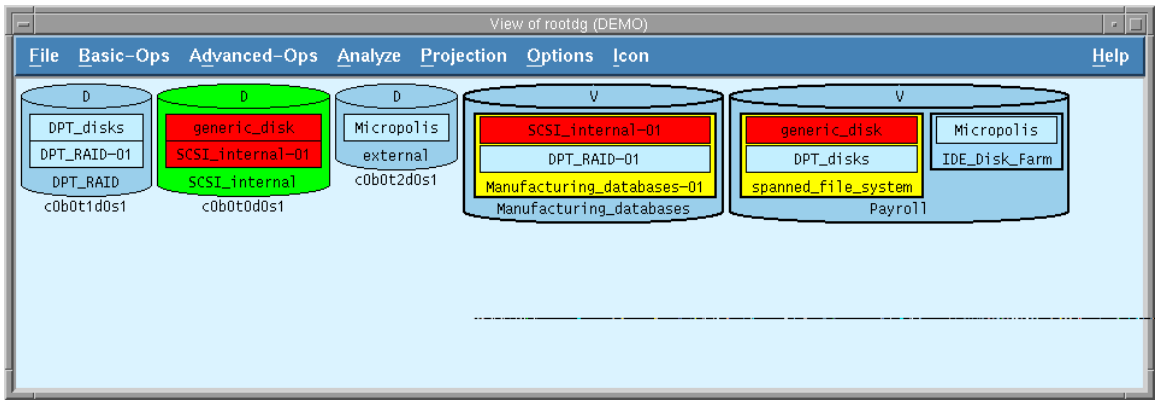
The **Advanced File System** provides a superset of the features of the journaling filesystem (vjfs) that is the default file system for UnixWare 2.1. Vxfs has more advanced reliability features and it supports the on-line administrative capabilities of the Online Data Manager. Other features include:

- **Intent logging (journaling**) - maintains file system integrity even after cold shutdowns. Protects system and user data.
- **Synchronous writes -** causes unlogged changes to be written to disk when a file is closed. Thus once a file is closed, the data is safe.
- **Extent based file allocation -** an efficient file allocation scheme that allocates contiguous disk space rather than just blocks. Thus reducing disk seek time and improving performance.
- **Dynamic i-node allocation -** system i-nodes are increased on an as needed basis, requiring no administrative intervention.
- **Direct I/O support -** provides a bypass to disk aching for database and transaction processing applications for faster raw disk accesses.
- **Access Control List support -** ties into the advanced security features of UnixWare 2.1

The **Volume Manager (VXVM)** builds virtual disk volumes on top standard UnixWare file systems so that they span multiple disks or multiple slices. After creating a volume, the Volume Manager operates on the virtual disk volumes, rather than the physical device. This provides greater flexibility to administrators who need dynamic control of disk resources. VxVM permits transparent and dynamic reorganization of the virtual disk volumes. This reorganization allows administrators to optimize performance, change volume size, add mirrors, perform backups or other administrative tasks without interrupting system users.

The Volume Manager provides:

- **Disk striping (RAID 0)** - increases file system performance by distributing files and file systems across different disks, allowing portions to be read and written in parallel, as opposed to being done serially.
- **Disk mirroring (RAID 1)** - the ability for a second to take over if the primary fails.
- **Disk striping with distributed parity (RAID 5) -** provides ability to re-generate data on failed drives using multiple "parity" disks rather than having to mirror all drives in question. Parity information kept on parity drives is used o reconstruct the data lost from a disk going off-line.
- **Volume resizing -** grow and shrink file systems in real time.
- **Hot sparing** allows administrators to designate a disk drive as a spare so that if a mirrored drive goes down, ODM will rebuild the mirror onto to the designated spare. This reduces the likelihood of critical data being lost due to a lack of mirroring.
- **Disk defragmentation -** performed on-line and transparent to users and applications.
- **Staged I/O** - provides a more efficient method of routing storage management requests through the UnixWare kernel, thus improving system performance and I/O throughput.



The Veritas **Visual Administrator** (VxVA) is a graphical system management tool used to display, organize and optimize disk usage, and guard against media failure. The VxVA application is based on the Motif graphical user interface. To make the job of the system administrator easier, VxVA displays an icon-based interface for the manipulation of system resources. By implementing the use of color, monitoring capabilities, and user-definable default settings, VxVA provides an efficient environment for day-to-day disk administration. VxVA works together with VxVM to help manipulate entries defined by the Volume Manager. VxVA displays both physical and logical disk resources, providing the administrator a consolidated view of the storage management subsystem.

The true power of this tool is in the using. Few enterprise customers who see the capabilities Online Data Manager fail to purchase it.

UnixWare 2.1 Software Development

UnixWare 2.1 provides a powerful set of development tools for building advanced applications, and supporting new uni-processor and multiprocessor systems and new hardware devices. The UnixWare system comes with a powerful set of utilities, tools and command languages that form the basis of a rich working environment for users and applications developers.

Development technologies available for UnixWare 2.1 are:

- shell programming tools

- Windowing Korn Shell (WKSH)
- Software Development Kit (SDK)
- Application Programming Interfaces (APIs)
- Application Debugger
- Kernel Debugger
- Performance management tools
- IHV Development Kit
- Platform Support Kit (PSK)
- Software Packaging Tools

Shell Programming Tools

The shell, or command line, interface is the traditional method used to enter commands and work with files and processes in the UNIX operating system. It is often preferred by many professional software developers, system administrators or users who have some background in computer science or experience with earlier versions of the UNIX system. UnixWare provides all of the standard UNIX shells for the experienced UNIX user:

- Bourne shell (sh)
- C shell (csh)
- Korn shell (ksh)

The **Bourne shell**, written at AT&T Bell Telephone Laboratories was the standard shell with earlier releases of UNIX system software. The **C shell** uses a programming syntax that is similar to that of the C programming language and supports command line and history editing facilities. The **Korn shell** borrowed heavily from the Bourne shell and from the C shell, but added facilities to support better interactive use. The Korn shell provides a rich set of command control facilities, including I/O redirection, job control, aliases, variable substitution, option setting and other environment control features.

Windowing Shell (WKSH) Programming Tools

The Windowing Korn Shell (WKSH) is designed for UNIX system shell language programmers intending to develop applications using a "windowed" front-end. It is built on top of the X Window System. WKSH gives shell programmers the ability to build graphical applications using the extensions to the standard Korn shell provided by WKSH. Using WKSH provides several benefits over Motif programming in C or C++:

Faster - library and process initialization are done automatically
Interpreted - because it is a shell environment, there is no wait for compiling.
Easier to Debug - standard shell debugging techniques can be used.
Easy to Extend - c-code can be linked into WKSH to add to its capabilities

UnixWare 2.1 Software Development Kit (SDK)

The SDK (an optional add-on product) consists of the application programming interface libraries as well as the basic tools that the applications developer can use to create applications for the Application Server or for the Personal Edition version of UnixWare. The tools are particularly suitable for developing graphical and network applications. The major components include:

- C compilation system (part of base product)
- Application Programming Interfaces (APIs) and libraries
- desktop manager development
- enhanced debugger
- kernel debugger
- IHV development kit
- graphics development package
- Motif development package

- network management software development kit
- Software Development Kit documentation
- software packaging tools

C and C++ Compilation Systems

The **C and C++ compilation systems** have been optimized for the Pentium and Pentium Pro processors. The C++ compiler and the C++ Standard Components Class Library enable the use of modern object-oriented programming techniques. The C and C++ compilation systems support both shared object libraries and dynamic linking.

Application Programming Interfaces (APIs)

A complete set of application programming interface (API) library routines and related toolkits and documentation are provided to simplify the creation of distributed client-server applications, and networked, graphical and multithreaded applications. The libraries and toolkits provided with the SDK for this purpose include:

- **User threads support library:** an enhanced set of APIs for providing high performance synchronization, scheduling and user-level threads facilities. Included are routines to create threads, terminate threads, wait for threads, and adjust threads' scheduling characteristics.
- **Thread safe libraries**: the libraries libc, libm and libnsl have been made thread-safe to support multithreaded applications. This ensures that parallel threads which call the same library function will not interfere with each other.
- **X Server Graphics Development Package** : provides basic library and header file support for programming in the X11R5 environment. This includes the font server, X server, Xlib and X intrinsics. Clients and utilities used to support UnixWare X Server development are also included.
- **Desktop Manager Development Package** : provides the desktop manager API support including the drag-and-drop and help APIs. It provides a variety of graphics libraries and tools to support the graphics capabilities in UnixWare.
- **Motif Development Package**: provides the Common Desktop Environment Motif toolkit, based on OSF Motif 1.2.3. It provides the tools necessary to develop applications using Motif.
- **Network Development Package**: provides the header files and libraries for writing networking programs which use XTI/TLI, RPC, sockets, network selection facility, and name-to-addressing APIs.
- **NetWare Development Package**: provides development header files and libraries for writing and debugging network management, IPX/SPX, and NetWare-aware client applications.

Application Debugger

The SDK includes graphical windowing debugger for source-level application debugging that supports multi-threaded applications and LWPs. The graphical debugger can track multiple threads and processes executing on independent processors simultaneously. The user is given complete control over the execution of the program and the examination of its state. The debug program provides both a command line interface and an X Windows based graphical user interface.

The graphical debugger has the capability of configuring the user interface to meet the developer's current needs. By default the graphical debugger displays the following nine panes in a series of six windows:

- process (list of processes and threads)
- stack (stack trace of current object)
- symbols (visible symbols in current context)
- registers (machine registers for current process or thread)
- disassembly (disassembly of current function)
- source (current source file)
- event (stop, signal, syscall events)

- command (entry of debugger commands)
- status (status of current process or thread)

The look and layout of the debugger windows in the graphical environment is highly customizable, allowing developers to zero in on problems more quickly and easily.

Kernel Debugger

A threads-aware multiprocessing kernel debugger is provided to assist in the porting and debugging of kernel modules and drivers by allowing developers to examine and control a running kernel. This benefits system developers by reducing the development time needed to incorporate new modules into the system.

Performance Management Tools

A new profiling tool,**fprof,** is provided to support flow and time profiling. Flow tracing is a form of tracing used to analyze performance and locality of reference of text. The information produced by this profiling tool can be used for performance analysis or locality of reference tuning. The **lprof** tool is another profiling tool which displays line-by-line execution count profile data for a source file. In addition, the basic **prof** profiling tools now support C++ and dynamic shared libraries.

A set of Metric Access Support (MAS) functions are provided to allow applications to advertise and gather performance metrics.. The raw metric data can be analyzed to determine how best to improve the performance of a specific application, or even of the operating system.

IHV Development Kit

An **Independent Hardware Vendor (IHV) Development Kit** , packaged as part of the SDK, provides the tools, documentation, instructions and sample code required to write device drivers for new hardware devices. The Kit contains an extended set of sample software code and development tools to assist IHV developers when creating/updating UnixWare device drivers. The kit contains:

- sample drivers for video screens.
- screen interface (SI) modules, Host Bus Adapter (HBA) drivers, and networking drivers, in both uni-processor and multiprocessor versions.
- README files, containing details about each of the included drivers and other related technical information.
- HBA and LAN pre-certification kit, which facilitates testing the integrity of the HBA and network driver floppies used to install HBA and LAN driver packages.
- DDI conformance tool, a utility that can be used to ensure that the driver complies with DDI/DKI interface standards.
- packaging utilities and source code for creating driver installation floppy disks.

Device driver developers can also use the auto-configuration mechanism provided with the UnixWare kernel. This allows device drivers to take advantage of the automatic detection and configuration capabilities available in some HBAs and devices. Auto-detection enables the device driver to see whether the device it is to control actually exists on the system.

Platform Support Kit (PSK)

The UnixWare **Platform Support Kit (PSK**) provides a quick, easy and cost-effective way for OEMs whose products are based on the Intel 80*86 architecture family to ensure that UnixWare supports their uni-processor and multiprocessor hardware platforms. PSK provides :

- sample source code

- documentation
- compatibility tests
- certification process for Platform Support Modules (PSMs)

Platform Support Modules (PSMs) are the layer of software that sits between the UnixWare 2.1 kernel and the hardware. The PSM mechanism allows for greater system portability onto new architectures.

Software Packaging Tools

Software Packaging Tools provide the tools to create and modify installable UNIX software packages. UNIX package support (also known as pkgadd format) is a standard mechanism for installing applications, drivers and files onto the UnixWare system. A complete set of tools are provided to install, remove, display information about, and check the contents of a package. The popular custom mechanism is also supported on UnixWare 2.1.

UnixWare 2.1 Standards Conformance

UnixWare is a standards-based operating system that drives and complies with many of the important Open Systems standards and specifications that have been adopted in the computer industry to date. These include standards in the area of operating system interfaces, commands and utilities, languages, and networking areas such as TCP/IP, IPX/SPX, SNMP, etc. The standards to which UnixWare conforms can be classed as binary and source. Source standards enhance compatibility of source code across all implementations of the UNIX system, whereas binary standards enhance compatibility of binary code across processor family implementations.

Standards-based Application Programming Interfaces (APIs)

UnixWare's powerful applications runtime environment conforms to industry standards in order to support most off-the-shelf applications. Key standards are:

- POSIX - Portable Operating System for UNIX (POSIX.1, POSIX.2 and POSIX.4)
- FIPS 151-2 - Federal Information Processing Standard
- SVID3 - System V Interface Definition, Issue 3
- XPG4 - X/Open Portability Guide
- iABI - Intel Application Binary Interface
- iABI+ - Intel Application Binary Interface +
- iBCS2 - Intel Binary Compatibility Specification, Version 2
- COFF - Common Object File Format
- ELF - Executable and Linking Format
- System V ABI, 3rd Edition, and Intel Processor Supplement 3rd Edition, and ABI+ for Intel Architectures Specification 3.0.
- Novell DDI/DKI, up to Version 7.
- X Consortium ICCCM.

Source Standards:

- System V Interface Definition, Third Edition (SVID3).
- ISO 9945-1:1990, Portable Operating System Interface (POSIX): System Application Program Interface (IEEE Std 1003.1-1990).
- Federal Information Processing Standard (FIPS) 151-2 (IEEE Std 1003.1-1990 with all extensions).
- ISO/IEC 9899:1990 C Language Standard (plus ISO MSE amendment - EN 29899).
- Department of Defense National Computer Security Center (NCSC) C2 Trusted Computing Base.
- X/Open Portability Guide Issue 4 (XPG4) Base Profile Brand.

CORPORATE HEADQUARTERS
The Santa Cruz Operation, Inc.
400 Encinal Street, P.O. Box 1900
Santa Cruz, California 95061-1900
SALES AND INFO: (800) SCO-UNIX (726-8649)
(408) 425-7222 • FAX: (408) 458-4227
TWX: 910-598-4510SCO SACZ
EMAIL: info@sco.COM
SCO HOME PAGE http://www.sco.com

EUROPEAN HEADQUARTERS
The Santa Cruz Operation, Ltd.
Croxley Centre, Hatters Lane
Watford WD1 8YN United Kingdom
+44 (0)923 816344 • FAX: +44 (0)923 817781
TELEX: 917372SCOLON G
EMAIL: info@sco.COM
U.S. Federal Systems Group
Reston, Virginia (703) 715-8700
SCO Canada, Inc.
Toronto, Ontario (416) 922-1937
SCO Eastern Europe and Central Asia
Düsseldorf, Germany +49 211 5768 41
The Santa Cruz Operation (France) Sarl.
Issy-les-Moulineaux +33 1 46 48 85 00
The Santa Cruz Operation (Deutschland) GmbH.
Bad Homburg v.d.H. +49 6172 48670
The Santa Cruz Operation (Italia) Srl.
Cassina dé Pecchi, Milano +39 2 95301383
The Santa Cruz Operation, Ltd. - Nordic Region
Holte, Denmark +45 4242 5775
SCO Middle East
Salam, Dubai +97 14 482 801
The Santa Cruz Operation Pty., Ltd.
North Sydney NSW, Australia +61 2 911 7711
SCO, Kabushiki Kaisha
Meguro-ku, Tokyo. +81 3 5486 5120
The Santa Cruz Operation (Asia), Ltd.
Singapore +65 471 2221
The Santa Cruz Operation Mexico S. de R.L. de C.V.
Mexico, DF (525) 566-1781