Chapter 8 Migration between SunOS Releases

This chapter briefly describes the differences between SunOS releases and notes release improvements that can affect applications software.

8.1 SunOS Release Overview

Software release cycles

In order to keep OEMs and customers informed of upcoming system software releases, Sun has instituted a new software release cycle. The objectives of this plan are to establish a mechanism for accurate schedules and timely SunOS releases. The three components of the new release plan are major releases, minor releases, and micro releases.

A major release is a SunOS version that could require an application porting effort because of new features, architectural changes, or other significant improvements. The contents of a major release and the availability of unbundled software will dictate the schedule.

A minor release's primary purpose is to support new software and hardware products. In most cases, a minor release will not require an application porting effort. Minor releases will generally be available on a 6- to 9- month schedule.

A micro release, also known as a dot-dot or bug fix release, consolidates individual bug fixes and applies only to the relevant product(s). No new hardware or software functions are included in a micro release.

SunOS has been evolving over the last few years, and its powerful innovations will become incorporated into a new unified UNIX discussed later in this chapter. An aspect of SunOS enables multiple clusters of networks communicating over a network while also supporting different releases of SunOS. (A few rules apply to diskless clients and their servers.) All machines can access other filesystems on different parts of the network through NFS.

8.2 SunOS 4.0

Features of SunOS 4.0

SunOS 4.0 began shipping in May 1988. It contains many improved features and is a sound architectural foundation for future releases. See Figure 8-1 for SunOS 4.0 features.

For more details, please refer to the *SunOS 4.0 Change Notes* (part of the SunOS 4.0 release documentation), *SunOS 4.0 Release Report*, and System V and SunOS 4.0 documents.

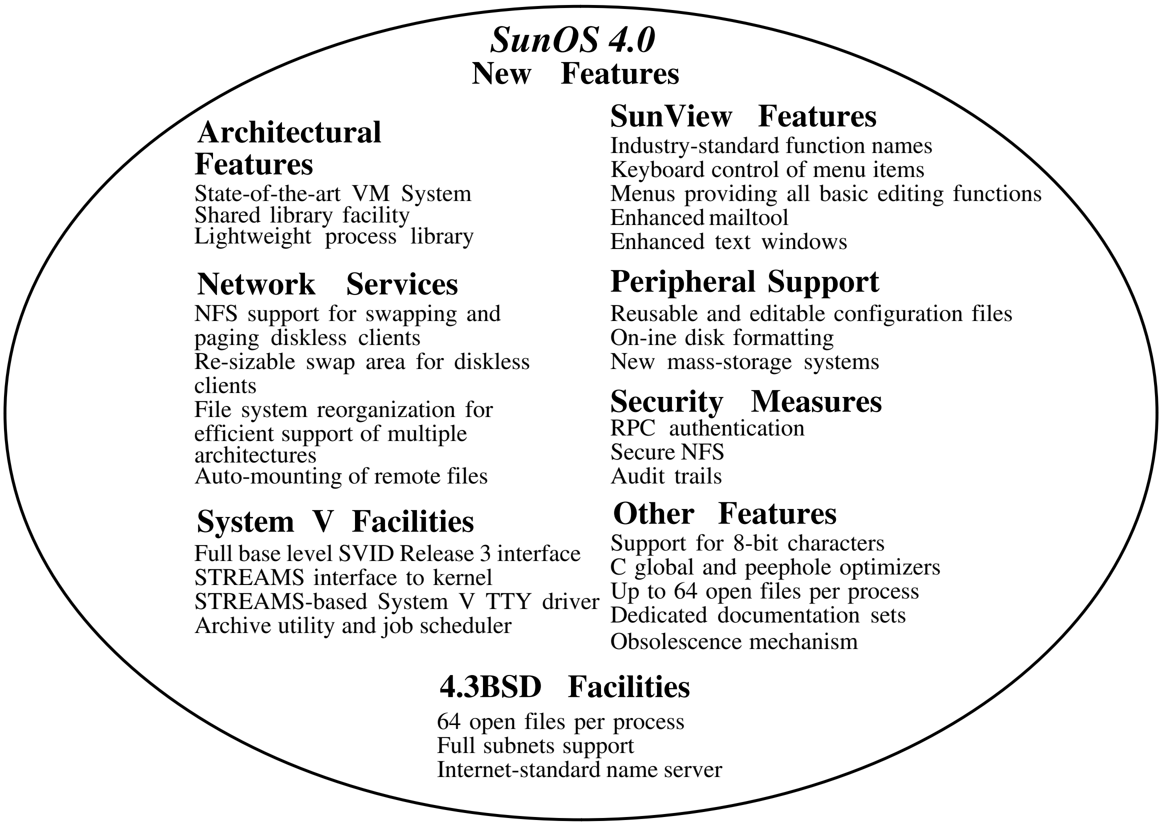

Figure 8-1. SunOS 4.0 Features

8.2.1 SunOS 4.X

A forthhcoming release of SunOS will incorporate the Base Level System V Interface Definition (SVID) and pass the System V Verification Suite (SVVS). SunOS 4.1 includes new features such as improved ease of use and basic international support will also be provided. Additional SunOS 4.X features will be announced later.

8.3 The Sun and AT&T Joint UNIX Implementation

In October 1987, Sun and AT&T announced an alliance to integrate SunOS and its Berkeley UNIX (4.3/4.2BSD) features into a single, open UNIX System V standard. This will be released as UNIX System V Release 4.0. This approach uses industry standards to define and describe programming interfaces, the result of which is inherently competitive and multivendor. The goal of this venture is to provide a standard, stable application development environ-ment and, to encourage application software creation. The overall result will be a consistent user interface across different applications, better functionality via defined OS interfaces, and preserved investment in software and people.

Note that UNIX must run on any hardware architectures. This new, unified UNIX conforms to this requirement with a comprehensive programming interface for development tools, user interface (OPEN LOOK), and networking. (See Figure 8-2.) Also note that Xenix extensions will be present in this UNIX because of an agreement between AT&T and Microsoft.

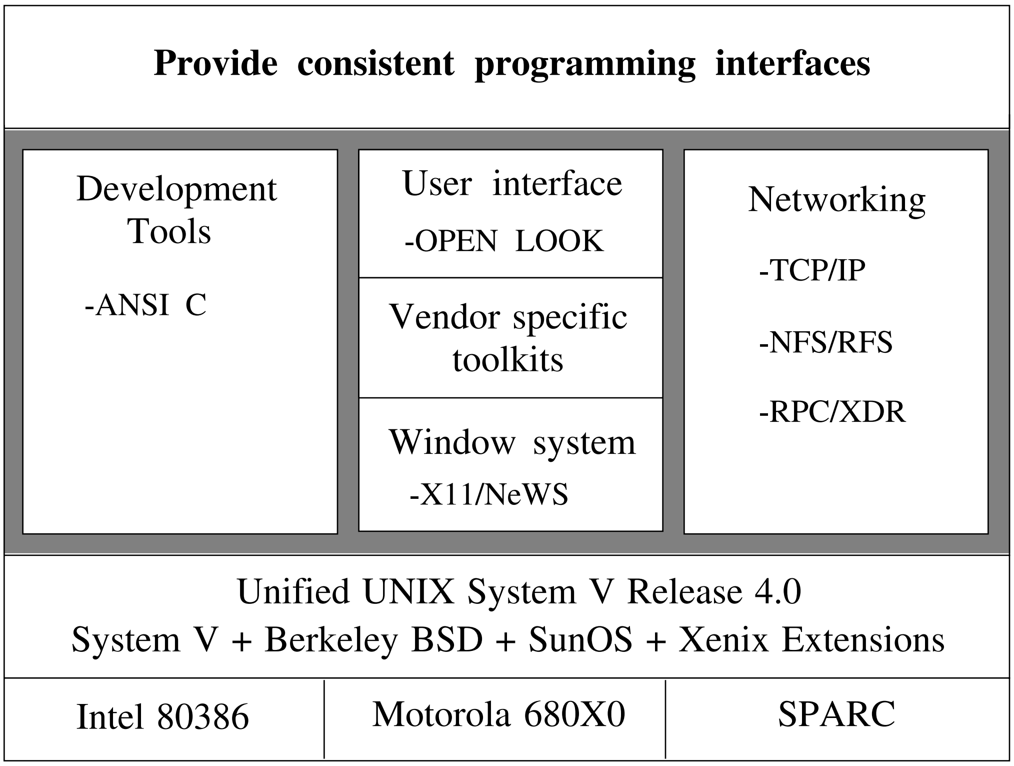

Figure 8-2. UNIX System V Release 4.0

Incorporation of SVID into SunOS 4.X

Sun and AT&T's efforts for unified UNIX are divided into three phases. For the first phase, Sun will deliver a SunOS version that will contain most of the features and functions of the unified UNIX system. It will comply to the Base Level SVID and will support software developers who want early exposure to the new System V UNIX. Phase 2 is the AT&T effort to supply the unified UNIX system to the world. The beta version (3B2 source-based) will appear in 1989. Phase 3 is an effort to provide continued evolution of UNIX while maintaining the consistent programming interfaces. The interfaces are the most important in this phase, not the kernel implementation.

8.4 Porting from SunOS 3.X to SunOS 4.0

SunOS 4.0 has several new features appealing to most OEMs, users, and system administrators. Numerous enhancements make the migration from 3.X to 4.0 different from previous

SunOS installations or upgrades. Most SunOS 3.X binaries can run under SunOS 4.0.

Do not use pre-SunOS 4.0 object files (.o and .a files) with SunOS 4.0 without recompiling these files for SunOS 4.0. Compatibility issues arise with programs that depend on the filesystem layout, kernel data structures (/dev/kmem or /dev/mem), and certain system call semantics (mmap). The full details are listed in the *SunOS 4.0 Change Notes*, which is suggested reading for any porting effort of this type.

8.5 Suggestions for Future Releases beyond SunOS 4.0

Open UNIX System V standard: UNIX System V Release 4.0

The SunOS 4.0 release provides many opportunities for application developers to insulate themselves from the possible incompatibilities of future major releases. These product migration strategies can ease the transition to the unified UNIX:

- Install the System V compatibility package option and become familiar with its conventions.
- Set your $PATH variable to choose BSD behavior (/usr/ucb) or System V behavior (/usr/5bin). Further details are in the System V and SunOS 4.0 document.
- Design your package to install in either environment (BSD or System V).
- If you are writing new code, write to the System V Interface Definition (Issue 2, Volumes I, II, III) if possible.
- Try to avoid writing your own custom kernel modules; write to the libraries or interfaces instead. SunOS 4.0 gives you new tools to make this job easier.

8.6 Migrating from SunOS 3.X to SunOS System V Default Release

Product migration strategies for transition to unified UNIX

Moving from SunOS 3.X to the System V default release is an addition to a SunOS 4.0 to System V default release migration. The System V default environment means that the System V commands, libraries, and behavior is the default. Please see below.

8.7 Migrating from SunOS 4.0 to SunOS System V Default Release

As the System V and BSD environments converge, it becomes important for users to be aware of the default system environment. In SunOS 4.0, the BSD commands, libraries, and behavior defaults to BSD operation.

When Sun releases its OS including System V Release 4.0, the System V behavior will become the default environment. This does not mean the BSD environment will go away. It does mean that users preferring BSD will need to prepend /usr/ucb to the $PATH variable before /bin, /usr/bin, and /usr/5bin in SunOS 4.X. Because of this upcoming change, users need to be aware of programming practices that can complicate a transition to the next software release.

To avoid application incompatibilities, changing the $PATH variable is the first step. In cases where the application depends on commands or options not in both SunOS and System V, you should try to remove the dependencies. Otherwise if you require the BSD or SunOS semantics, you may have to specify the BSD compatibility package for your application to work with the System V default release. The obvious areas of incompatibilities are commands currently present in both /usr/5bin and /usr/bin (*ar, cat, cc, cflow, chmod, col, cxref, date, diff3, du, echo, expr, grep, lint, ls, m4, nohup, od, pr, sed, sort, stty, sum, time, touch, and tr*). Consider the example with the echo command:

BSD
```
echo "hello world on line1";echo -n "hello world on line2"
```

System V
```
echo "hello world on line1\nhello world on line2\c"
```

8.8 Compilation, Makefile, and Build-Related Issues

To ensure that existing programs build independently of the environment preference of the invoking process, most developers 1zwant to explicitly specify a BSD environment by setting CC=/usr/ucb/cc and passing CC to makefiles. The /usr/5lib is a symbolic link to /usr/lib, with conflicting files moved to a new directory called /usr/ucbinclude. Similarly, /usr/5include is a symbolic link to /usr/include with all files there moved to /usr/include. Conflicting files are in the /usr/ucb/include directory.

8.8.1 Other Changes

The underlying system administration will change to be more compatible with System V. Count on the accounting record format changing along with uucp administration. To be safe, do not rely on the conventions of /single/init for initialization files associated with starting the system.