M. KRAMER

From: CLEX1::DOB "Hans Doebbeling, GSI, +49-6151-35954" 18-APR-1991 0
9:26:20.86
To: @guumail
CC: DOB
Subj:

EUROPEAN ORGANIZATION FOR NUCLEAR RESEARCH

CERN - CN/C5/90/68
Revised November 30,1990

Recommended Standard for Unix Workstation Environment Setup

Rainer Toebbicke, Philippe Defert

CN Division
CH-1211 Geneva 23

Abstract

This document proposes a standard Unix workstation user environment and system setup, including recommendations for system installations, for a practical split between centrally supplied services versus autonomous workstation operation, and for the user environment such as a preferred shell, command search paths, etc.

The recommendations have been chosen to be as practical as possible on all Unix(1) workstations that may gain importance at CERN in the medium term, namely

- HP/Apollo, in either of the two possible Unix environments,
- Digital's DecStations and VaxStations running Ultrix,
- The NeXT system,
- IBM RS6000 systems,
- Sun workstations running SUN OS,
- Silicon Graphics,

without loosing touch with the existing or expected mainframe Unix services, e.g. CRAY, cernvax and AIX/370. The goal is to define an environment

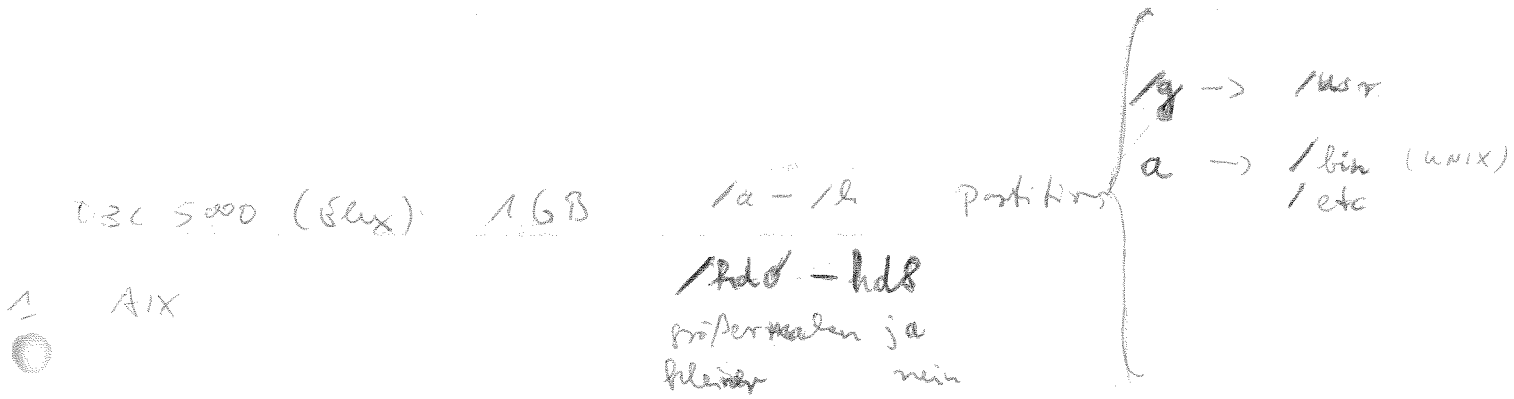
- where the Unix user can approach whatever workstation is accessible to him(2) regardless of its brand and resume his usual work after a minimum adaption period,
- where locally provided software and utilities can quickly be ported with small effort and so be largely non-impacted by the complexity caused by the variety of Unix workstations to be expected.

---

(1) Throughout this document only the interface as experienced by the end

user is considered, irrespective of implementation dependencies. The Apollo DOMAIN/OS SR10, Ultrix and MACH are Unix systems in this sense despite radically different implementations.

(2) please read him/her, he/she etc. where applicable



ov

It is in the spirit of this document that the openness and versatility of Unix systems be respected and that users, who for whatever reason choose to deviate from the proposed recommendations, be supported as well as possible in their environment.

1. A certain number of inconsistencies exist with the growing number of Apollo systems installed at CERN, which can be configured to any of the three environments Aegis, BSD Unix and Unix System V: roughly half of the Apollo user community currently runs in a Unix environment by default. The number of systems installed with the BSD environment is slightly lower than the number of System V installations. However, System V has often been chosen without any particular requirements and despite the fact that many of the CERN provided utilities either favor or even require a BSD environment. Many nodes have even been "hacked" to allow for BSD facilities to be called from the default System V environment, regardless of the inconsistencies this may involve (e.g. when a BSD utility invokes another command and enters the slightly incompatible System V version) and of the increased maintenance effort required.

2. A long relationship with Digital has motivated traditional VMS users to buy DECstations which only support the Ultrix system. The Unix user community is therefore likely to be widened by a traditional VMS user base.

3. The Unix operating system is a common denominator between workstations from different vendors. The differences between these systems can be largely hidden by correct setting up and by providing a set of common utilities. The user can thereby be offered a uniform appearance, increasing the value of training and documentation and making the system more easily accessible and more efficiently usable. On the other hand, differences in system setup and working environments complicate documentation and locally provided software.

CONTENTS

TABLES

PART 1.

USER ENVIRONMENT

1. THE USER'S DEFAULT SHELL

The choice of a shell is probably a less critical standardization item and could be left to the user. Locally provided software will normally run independent of the surrounding shell. However, documentation for local software, support, training of new users will be easiest if an agreement on a recommendation could be found. Again, universal availability is an important issue.

- The C-shell is available on all systems we know of. It is superior in functionality to the Bourne shell, however it does not implement all its features. Although its command re-editing facilities are not easy to learn for the novice user, the C-shell is the traditional shell of choice on BSD systems for interactive use.
- The Bourne shell /bin/sh is the basic Unix shell and available on all implementations. It does not contain the C-shell's command re-editing facilities, but is adequate for command scripts and as a batch job control language.
- The Korn shell is a superset of the Bourne shell including command re-editing facilities with functionality comparable to the C-shell but superior command re-editing facilities. It is however not widely available, and its implementations on different systems (e.g. Apollo and AIX) are not completely compatible.
- There are several other non-vendor-supported shells such as the Bourne-again-shell (bash).

RECOMMENDATION: For portability reasons we recommend that command scripts be written in the Bourne shell (/bin/sh) language. The Bourne shell will be the reference shell for beginner's documentation. New and novice users will be set up with the Bourne shell as default interactive shell, leaving them to change to a C-shell or other newer shells such as ksh or bash on their own initiative.

Support will be given on both Bourne (or even Korn) shell and C-shell problems.

2. WINDOWING SYSTEM AND WINDOW MANAGER

Most Unix workstations provide a bitmapped graphics display, which the applications access through some high level windows interface. In the general case this native interface will be different among different brands of machines.

In the interest of providing CERN software implementable on workstations of different brands, a standard graphics screen interface is necessary which is

available on all workstations. Since the most obvious candidate is the X11 windows system it is recommended that the X-server (i.e. the display interface driver in X terminology) be started on all Unix workstations.

The Apollo user is expected to normally use the Apollo display manager for efficiency and compatibility reasons. The X-server will share the display with the display manager, which will allow simultaneous interaction to X and display manager applications. The setup can in fact be changed by the user himself to running a complete xdm/motif system instead of the Apollo display manager. However, currently this can hardly be recommended, owing to the incomplete status of the software and the high memory requirements (minimum recommended is 12 or 16 Megabytes).

RECOMMENDATION: All Unix workstations should start the X-server. Additionally, CERN software, even when used in non-graphical modes, e.g. from a dumb terminal, will assume that an installed version of the X11 software (e.g. libraries, fonts, etc.) is accessible.

2.1 WIDGETS AND TOOLKITS

Users writing programs which make use of the window system should be encouraged to use documented X11 interfaces. Besides portability, the obvious advantage to this is that in this manner expertise is built up in a way useful for other systems and that help will be more readily available on known interfaces. A recommendation as to which set of widgets or toolkits to use should be governed by widest availability. A few points for discussion:

- OSF/Motif will be integral part of the OSF/1 system and available on many others. Its "look and feel" is common to the Personal Computer world with MS-Windows and OS/2 Presentation Manager.
- SUN offers X11 but many applications use the "Open Look" toolkit. The toolkit is freely available in source form but not easily portable to all environments.
- Digital's DecWindows is mostly a superset of X11 X-Windows. The portability of programs developed under DecWindows will depend on the usage of special DecWindows fonts or functionality. Programming to special DecWindows functions should therefore be avoided.
- The NextStep graphical interface is not based on X11 but on Display PostScript.

RECOMMENDATION: We recommend that a library of widgets (e.g. for the Motif toolkit) used commonly at CERN be collected and published among CERN users.

3. SYSTEM V OR BSD

The following discussion has its origin in certain inconsistencies in Apollo systems installed at CERN regarding the choice of the user environment, namely BSD versus System V. It is mentioned here since the problem appears in most of the Systems this document addresses. The Apollo system appears as the most drastic case, since a clear cut between a pure BSD 4.3 and a System V.3 appearance exists and in practice the user has to make an (almost) definitive choice. The other systems are normally primarily based on one of the environments and offer features of the other as extensions. Often the user is not required to choose (as long as no incompatibilities appear), e.g. the date command on the BSD-based Ultrix system offers System V-typical formatting options. In other cases compiler (and loader) flags allow to choose include files and libraries affected by an environment change.

It is in the user's and in the support personnels' interest to agree on a common recommended Unix environment. On the first look the user will rarely be impacted by the subtle differences. However, they make the task for support more difficult and complicate centrally provided software and command scripts. The 'xprint' shell script for example relies on certain BSD facilities which are not available in the System V environment. This requires explicit switching to the BSD environment, and it will fail if the environment is not installed.

The Berkeley 4.x system and the AT&T System V have remained distinct after the merge of a multitude of Unix variants over the past years. As a practical choice and without speculating on whether either of these system does today more resemble a hypothetical unique system of tomorrow than the other, we propose to prefer the BSD environment to the System V environment for the following reasons:

1. Many CERN developed facilities and most of the "public domain" software which make Unix so attractive require the BSD environment.

2. With

   - Apollo's Domain/OS
   - Digital's Ultrix which is a BSD 4.2 based system,
   - NeXT(3)
   - SUN OS, which continues to offer BSD 4.3 facilities,
   - CRAY's Unicos, which started as System V with BSD extensions will even move closer to BSD as of the next release,
   - IBM's AIX(3)

   the majority of systems this document is targeted at support a BSD environment.

---

(3) The NeXT system is nearly BSD and IBM sometimes call AIX a System V implementation. Both however show enough differences to make a straight comparison difficult.

3. The coming AT&T System V.4 will include many BSD facilities and for the user resemble more closely a BSD 4.3 system than the System V.3 does today.

Regardless of this, programmers are encouraged to follow the relevant POSIX (IEEE 1003.1, ISO 9945/1) standards where possible rather than rely on specific BSD, System V, or even vendor specific facilities.

RECOMMENDATION: We recommend to use BSD facilities on systems where there is a choice. For Apollo users, the Apollo System V environment will be supported as long as Domain/OS is.

4. USER IDENTIFICATION

NFS and the BSD r-facilities (r for remote, rsh, rcp,rlogin, etc.) require that the Unix internal userid and group id are the same on all systems! Each workstation owner has to maintain the /etc/passwd and /etc/group files for each user who should have access to the station, maintaining a consistent mapping of user names to the numeric user and group ids. Other facilities, for example CERN provided tape access mechanisms, may have a similar need to identify users on a user and group basis.

It is therefore essential that on each Unix workstation as well as the Apollo Domain registry users be defined with the same Unix user- and group numbers (uid/gid).

The central computer users database (CCDB) has been set up to uniquely identify CERN's computer users and to enable unique uid/gid attribution (purposely called NFS numbers) on all systems.

RECOMMENDATION: It is recommended that an effort be undertaken to investigate the implications of using a centrally controlled, distributed administration service such as the Hesiod name service, which will be available with Ultrix 4.0 and OSF/1. Meanwhile workstation administrators are strongly encouraged to coordinate attribution of Unix accounts and numbers with the central database.

5. SESSION AND SHELL INITIALIZATION

Each new user account is created with a .login/.profile standard login scripts that sets up a number of reasonable defaults in the CERN environment. The login script should only contain the instructions necessary to execute commands stored in some centrally administered location, e.g. /usr/local/bin. The user can than be encouraged to define his own preferences after those provided centrally. The initial definitions could include e.g. command search path, man pages search path, cern library version, etc...

RECOMMENDATION: The user's command search path should contain (in that order):

1. the current directory "."

2. the user's &tilde./bin directory
3. the CERN provided /usr/local/bin directory
4. standard places:
   a. /usr/ucb
   b. /usr/bin
   c. /bin
   d. /usr/bin/X11

Shell scripts should set their own command search path instead of relying on the user's setting. Templates for other initialization files providing reasonable defaults should be created automatically for new users and at least offered to the existing user community, e.g. for

- shell initialization files (.cshrc, .shrc) with definitions for aliases (abbreviations), history control, etc., many of which are commented out, well documented and can be activated by the user if he wishes,
- window system and window manager defaults, e.g. .Xdefaults, .mwmrc, with definitions for menus, fonts, colour, etc., xhost/xset and similar startup commands,
- news and mail,
- popular user maintained software packages like emacs or GNU utilities.

6. DIRECTORY STRUCTURE FOR CERN SOFTWARE

6.1 THE CERN PROGRAM LIBRARY

The CERN Program Library is created as a stand alone tree and distributed in this fashion to the various systems. It is sufficiently monolithic and isolated from other locally installed software that there is hardly any reason why the distributed structure could not be adequate. It would typically(4) look as follows:

Table 1. CERN Program Library tree

/cern					
./new	./old	./pro	./gks	./nag	...etc...
new, old and current releases of CERN Program Library			graphics and numerical algorithms		others?

The distribution must allow for restrictions on local file system space, e.g. on systems where the root file system is too small to hold the whole tree. The tree could then either be installed in another file system and referenced via symbolic links or mounted as its own (read-only) file system.

---

(4) it would have to be implemented as a symbolic link on systems where the root file system is not big enough to hold the whole tree

6.2 OTHER CERN PROVIDED SOFTWARE

Being a popular operating system, Unix has been extended by many programs and utilities that users will consider to be a standard part of the system. Typical examples are TeX, emacs, tn3270, and many more. A de facto standard has evolved where such programs, include files and libraries reside in subtrees of a /usr/local tree, in the same manner as vendor supplied facilities can be found in /bin, /usr/bin, /lib, etc.

Nowhere is this standard clearly documented, but it can be guessed easily by looking at the installation procedures for the most popular public domain programs. It is obviously in the interest of CERN's users that this de facto standard structure be followed as closely as possible.

Table 2. Local software tree: /usr/local

/usr/local				
./bin	./lib	./include	./emacs	...etc...
commands: scripts and executables	libraries, configuration files, utilities used by commands in /usr/local/bi	C, Fortran, etc. include files for libraries	complete trees with additional files for a command in /usr/local/bin	

The structure described is the structure as it should appear to the user. The actual placement of software may be different as long as it fits the structure with e.g. symbolic links, and could take into account considerations such as load distribution, networks and support for different processors in parallel such as VAX/MIPS, Motorola/Prism (/SPARC), 370/Risc/i-386, etc...

The /usr/local tree should also be used where possible when a product is installed on a machine for a group of users.

7. PUBLICLY AVAILABLE FREELY COPYABLE SOFTWARE

One of the most attractive aspects of Unix is the huge amount of "public domain" or at least freely copyable software which is available owing to its popularity. Through networks countless programs and packages are available on hundreds of sites allowing "anonymous" access to retrieve them.

We recommend to install a central repository for at least the most popular of these programs. An adequate publicity and a properly maintained index would not only mean savings in network traffic as dozens of users will avoid downloading the same package from a server in the U.S., but also give users the possibility to check whether the port of a program to a certain hardware has already been done or at least what problems are to be expected. Also, we

believe that many users are willing to accept the sometimes inferior quality of this type of public programs, when they can benefit from their functionality for free. This should justify a certain coordination effort and hardware expenses.

The scope and visibility of such a service has yet to be determined. We propose:

1. Scope: the repository could be maintained jointly by CN for popular packages supported on most platforms and interested volunteers for other packages under at least the following rules:

   a. Every package should be identified by an index entry containing a 1 line description of what purpose it serves.
   b. For each package a README type file further describes its functionality. The same or another file describes the status (recent changes) and for which platforms an installation procedure (see below) exists.
   c. When a package has been ported to a particular platform, an installation procedure (e.g. "make install") allows easy installation of the executables and libraries.
   d. The complete original distribution kit is kept in some efficient format (e.g. compressed tar) for users who prefer to start from scratch.

2. Visibility: it should be decided whether the service should be open CERN internal only, to the HEP community or world wide. This decision should be based on network bandwidth cost, the processing power required implied by the number of simultaneous users expected and the maintenance effort required to keep it running.

   As estimated from experiences of other sites having opened a world wide service with often considerable resources, it is currently felt that such a service, without even considering the question whether it is CERN's role to provide it, would be incompatible with CERN's resources in both network bandwidth and support personnel.

   We therefore propose to open an anonymous FTP service restricted to CERN local access, with the corollary that any user "known" to CERN, e.g. with an account on one of the central VM, VMS and Unix services can access it by logging in through external TCP/IP.

7.1 NEWS

The "Usenet" news should be made available centrally (i.e. on a single machine) for all Unix machines. News readers (e.g. rn, nn) should be installed where possible to provide access through the appropriate nntp protocols.

As has already emerged as a de facto standard practice, CERN news can be locally merged into the Usenet news system. The standard .login/.profile/.newsrc files should be set up in a way that by default users automatically enter local news groups of general interest upon login in the same manner as on the VM and VMS services.

8. DOCUMENTATION

An introductory user's guide should be produced to complement the vendor supplied documentation and popular Unix books. The guide should

- provide initial help for novice users,
- contain a recommended reading list for standard Unix books and tutorials,
- document CERN specific facilities such as the CERN library and other available CERN supplied programs,
- outline the common aspects of Unix workstations, in contrast to vendor specific extensions. Users should be encouraged to use standard facilities where possible.

PART 2.
APOLLO SYSTEM SETUP

We recommend that Apollo workstations are set up so that the BSD environment is the standard environment. Users are recommended to specify the BSD environment and SYSTYPE in their &tilde./.environ file.

For historical reasons the System V environment will continue to be supported to the best of our ability. Users should however be aware that CERN written software and many public domain programs may expect a working BSD environment. Where possible, CERN software will switch to the BSD environment when required. It should however be emphasized that this may provoke inconsistencies and lead to situations where the correct functioning of the software cannot be guaranteed.

9. LOCAL COPIES OF SYSTEM SOFTWARE VERSUS NETWORK LINKS

The Apollo system software can be installed either on a local hard disk or simply directory entries referring to a server node on the network. A system installed with "all links" can be as small as about 20 Megabytes, whereas a completely installed system exceeds 150 Megabytes. Various intermediate stages are possible between those two extremes, normally grouped by functionality.

The following concerns should govern the choice of the amount of system software being installed locally:

- the amount of system software likely to ever be used is low compared to the total amount available, but it's hardly predictable which components are going to be used,
- access to the local disk is normally faster and more reliable,
- CERN provided facilities and documentation may refer to locally modified or customized parts of the operating system, which are normally maintained only on central servers for practical reasons

A reasonable breakdown of locally installed and linked components could therefore be the following:

Table 3. Locally installed versus network linked software	
locally installed	basic BSD Unix commands Fortran compiler or C compiler with system include files X-Windows server
linked	other compiler help/man pages central administration data (e.g. mail, TCP/IP) System V and Aegis commands CERN facilities: CERN library, GKS, PATCHY, etc. Motif window manager, library and sample applications
debatable...	dde (debugger) X-Windows clients (/usr/bin/X11)

A set of reliable central servers has been installed and will be operated by central operations. The servers will contain a full copy of the current operating system level and reference copies of common software where licensing restrictions permit.

10. X-WINDOWS

The X-Windows server is started automatically sharing the display with the Apollo display manager. In this manner Apollo users will primarily work with the Apollo display manager but are capable of running X-Windows applications.

This setup may be reviewed in the light of evolution of the manufacturer's software.

11. NETWORKING

11.1 TCP/IP

All nodes are set up to use the centrally supported TCP/IP name server. The /etc/hosts table is no longer maintained.

Users should be encouraged to have NFS installed on their nodes for increased interoperability with non-Apollo systems.

11.2 ELECTRONIC MAIL

A single node (cernapo) receives all mail and is the only link to CERN's mail gateway. Individual users should not run SMTP but make use of the mail server. A user's mail is stored in the mail server's spool space until removed with the "mail" program.

To avoid outgoing mail that cannot be replied to, all nodes should be set up such that sender addresses are ALWAYS of the form "user@cernapo". This can be achieved by a local rewrite of the sender address or by routing all mail from Apollos through cernapo.

11.3 PRINTING

Unix printing is done using the BSD print subsystem via links to a service node, i.e. //bsd_print. The node provides spool space and controls transfer to remote print sites, e.g. to central print servers such as the "LWPRINT" server or to printers attached to other Apollos driven by the Aegis print software. /etc/printcap should be a link to //bsd_print/etc/printcap, /usr/spool/lpd a link to //bsd_print/usr/spool/lpd

Domain/OS printing is truly distributed printing using NCS as communications protocol. Any node attaching a printer can be a print service node. The commands however are different from standard Unix commands, and printing is restricted to the Domain/OS world.

The different printing facilities are grouped together and uniformly accessible via the CERN-developed xprint facility, which adds printing via DECnet into the Vax and IBM world.

12. CERN PROVIDED SOFTWARE

Historically locally (CERN) provided software has been made available through links in the directories /cerncom and /cern. /cerncom contains various utilities and programs (i.e. read_news, 3270, etc.) whereas /cern is dedicated to the CERN library. "Standard" Unix systems normally present local software in the directory tree /usr/local, which means that many public domain tools are difficult to install owing to absolute path references to /usr/local/something in their Makefile.

Locally available software other than the CERN library should be accessible through a /usr/local tree, the current /cerncom directory could be a link to /usr/local/bin. We propose to install a /usr/local tree as part of the normal system installation process and on all existing "disked" nodes, containing the standard ./bin, ./lib, ./include, etc. entries as links to a central server. Users offering a product to a group of their colleagues will be suggested to install the software into a subtree of their /usr/local directory.

13. MAINTAINING SYSTEMS INSTALLED USING REMOTE LINKS

The general problem with non-locally installed components is that their availability is impacted by network problems and the speed of access. Linked components should therefore not include critical and often used files.

Another problem arises when a user installs a product on his node that adds a file to a non-local directory. Normally the installation procedures recognize this and omit the file, so that the server (and in case of a "replace" its integrity) is not affected. But non-Apollo products come with their own installation scripts not necessarily conforming to this rule so care should be taken.

Therefore splitting software into locally versus remotely installed parts may increase the installation effort in certain cases:

- Include files may present a problem depending on whether they have been installed locally or remotely. In case of server-installed include files new software adding include files has to be installed on the server.
- Man pages, help files and system documentation will be installed on central servers and should be linked remotely instead of installed locally, since for them neither speedy access nor 100 % availability is critical. Man pages for CERN supplied software will be made available on the central servers only. A problem can arise when a user installs programs locally on his machine, since then he is unable to install the man pages as well. In these cases at least the man pages for such programs will be installed on the server.

14. SYSTEM V TO BSD MIGRATION

The BSD and System V environments are in fact very similar. In the CERN environment the differences would pass unremarked in the majority of cases, if the ls command did not format terminal output into columns by default

under BSD. Therefore the move from System V to a BSD environment is hoped to be painless for most users. The following considerations may make a user choose to stay on System V.

1. CRON is a system process that allows periodic automatic execution of commands driven by a time table. The format of the table and the availability to users differs from System V to BSD. Under System V every user can add an "entry" into the cron table to be executed under his own userid, the crontab command insures that no user can impact other user's entries. Under BSD cron is reserved to the system manager.

   In the BSD environment the "at" service can be used to simulate access to CRON services a la System V. "at" allows the user to schedule a command for deferred execution. Periodic execution can be achieved by surrounding the command by a script that insures re-scheduling via "at". On the Apollo system a single machine running System V could be used by all users to use cron services, a simple script can be provided to update the tables and cause commands to be executed on the user's behalf on his own machine.

2. System V streams are not available under pure BSD. On the Apollo, however, like on some other systems (e.g. Sun) they are supported but require the development of streams "drivers" in the System V environment.

3. alas, the date command has less formatting options under BSD.

4. under BSD the /bin/sh does not allow for shell functions.

PART 3.
ULTRIX SYSTEM SETUP

The setup problem of most Unix systems differ from the Apollo Domain/OS environment owing to the fact that their data sharing mechanism is NFS based. Throughout the following, the Ultrix setup is therefore discussed representing a typical non-Apollo Unix system, the observations being largely applicable to e.g. SUN OS or AIX systems.

15. NFS ISSUES

Other than in the Apollo Domain/OS, file systems on different workstations are not automatically embedded into a global structure above the traditional local root file system. The file system on a workstation (or parts of it) must be actively exported and imported INTO the interested partner's existing structure. This considerably complicates setup and management, to a point where such a truly distributed approach is rarely chosen in practice.

Instead, by tradition users are grouped together and assigned disk space and home directories on a common server. In this case stations can be diskless, or a local disk used only for paging space, for a copy of the system directories for faster performance, or perhaps for non-backed up, unshared personal data.

The size and type of the server and its location would be chosen as to optimize

- network traffic -- traffic may cause less problems when the server is on the same Ethernet segment as most of its users,
- operational facilities -- for backups to tape to be run off the server, for problem diagnosis for critical around the clock availability,
- budgeting and flexibility -- a single big server may be easier to manage than several small ones.

15.1 CONTROLLING AND GROUPING NFS MOUNTS

We propose to mount all foreign file systems into a common highly situated directory, e.g. "/Net"(5). Such a convention has a number of advantages over having foreign mounts about just anywhere in the structure, which are among others

- commands like df and du don't descend into foreign file system trees, which is normally not expected,
- if several users work on a single station, and especially in case of a mainframe, the mounts and cashed data buffers are automatically shared between users,

---

(5) as suggested on the NeXT system

- the suggested directory structure is easy to accommodate (see below).

This would not mean that parts or even whole trees of foreign file systems could not be referenced from some other convenient place (e.g. /usr/local/bin !), since almost no system today does not offer symbolic links.

15.2 SERVER AND WORKSTATION DIRECTORY STRUCTURE

We propose the following directory structure to accommodate user files as seen on the workstation and the server:

1. User directories on the server are stored in file systems of the form /u/groupcode/userid, e.g. /u/c3/rtb. Splitting the directory structure into groups is not mandatory(6) , but has the following advantages:

   - users in a same group can share some common files and libraries, after the model of the G-disk on the VM system, e.g. /u/c3/bin,
   - a "group administrator" can manage entries in the tree and control space allocation for systems with quota,
   - the du would conveniently show how much space a group uses.

2. Upon boot, a workstation would automatically mount the servername/u file system as /Net/servername from as many servers as necessary,

3. A user's home directory entry in the password file would be /Net/server/group/userid, e.g. /Net/CN_Server/c3/rtb. Although in this way the server name is part of the directory entry and requires an additional activity (chhd command (change home directory)) if for any reason the user's home directory is to be moved, this does not impose any restriction on the number of groups per server or the number of servers per group!

The scheme will not break down for users with their home directory on the workstation's local disk -- where such an approach might be preferred for its simplified management or critical availability. The /etc/passwd would point directly to a local directory without passing through the NFS server. Where required, the NFS server could mount the workstation's file system in the appropriate place and reference the workstation's user tree via symbolic links.

---

(6) For historical reasons the current structure on Apollos does not follow this scheme. It would be possible, though not trivial, to introduce an additional level without seriously affecting the users

16. DISK PARTITIONING

For (mostly) historical reasons many Unix systems are installed on hard disks in several distinct file systems (or disk partitions). This may cause a problem with the suggested directory structure for the /cern library (see above) just under the root filesystem. For workstations with a local copy or for the server the tree therefore has to be installed on its own filesystem or somewhere else and the /cern link replaced by a symbolic link.