

System Administration and Maintenance of Fully Configured Workstations

Robert E. Van Cleef

Workstation Subsystem Manager for Operations

General Electric Corporation ¹

Moffett Field, CA 94035

October 6, 1988

GE Technical Information Series – TIS 88C1S005

Abstract

NASA's Numerical Aerodynamic Simulation Program (NAS) supports over eighty UNIX ² workstations, the majority of which are high-end graphics workstations used by scientific researchers. This paper discusses the administration tools and procedures used to maintain the system files on a large number of fully configured workstations.

¹This work was supported by contract NAS 2-11316 of the National Aeronautics and Space Administration to General Electric Corporation.

²UNIX is a trademark of AT&T Bell Laboratories.

1 Overview

NASA's Numerical Aerodynamic Simulation Program (NAS), at the Ames Research Center in Moffett Field, California, has three main objectives: [1, 2]

- Provide a national computation capability available to NASA, DoD, industry, and other government agencies and universities, as a necessary element in ensuring continuing leadership in Computational Fluid Dynamics (CFD) and related disciplines.
- Act as a pathfinder in advanced, large-scale computer system capability through systematic incorporation of state-of-the-art improvements in computer hardware and software technologies.
- Provide a strong research tool for the NASA Office of Aeronautics and Space Technology.

An important aspect of the work at NAS is the use of Graphics Workstations as a tool for Computational Fluid Dynamics research[3]. This has led to the installation of over forty Silicon Graphics IRIS workstations for use by local researchers and programmers. The majority of these systems are classified as production systems. This means they receive the same level of support as any other production system, such as a Cray-2 or a VAX.

The goal of the Workstation Support Group is to increase the usability and stability for the research scientists by maintaining the systems in a stable configuration and minimize downtime. A number of tools have been collected and developed to assist in this task and configuration standards have been implemented to simplify this task as much as possible.

This paper provides an overview of support policies, configuration standards and some of the tools that are used by the Workstation Support Group. Although this paper primarily addresses workstation issues, many of the concepts and policies mentioned in this paper are also used in the administration and maintenance of the larger NAS systems, see Table 1.

2 Workstation Support

Each workstation is provided a certain level of support depending upon the user requirements of that system. The level of support is directly related to

Quantity	System Type	Version of UNIX
2	Cray 2	UNICOS 4.0
1	Cray Y-MP	UNICOS 4.0
1	Amdahl 5880	UTS 1.2.2
1	ETA 10Q	Unix Sys. V
4	DEC VAX 11/780	BSD 4.3
8	Silicon Graphics IRIS 3030	SGI GL2-W3.5
9	Silicon Graphics IRIS 3130	SGI GL2-W3.5
25	Silicon Graphics IRIS 2500T	SGI GL2-W3.5
16	Silicon Graphics IRIS 4D/60	SGI 3.2
11	Sun 3/50	SunOS 3.5 or 4.0
1	Sun 3/60	SunOS 3.5
6	Sun 3/260	SunOS 3.5 or 4.0
6	Sun 3/280	SunOS 3.5 or 4.0

Table 1: Current NAS Production Computer Systems

the use of the system and level of operating system control the user desires. It is understood that the user of a system may reject any level of restriction – such as the restriction on the distribution of *superuser* passwords, but in doing so, they also reject the corresponding level of support.

2.1 Support Classification Levels

There are four classifications of systems; fully supported (FS), modified for application development (MAD), systems application development (SAD), hardware only supported, engineering development (HOSED). The classification of a system dictates what services the Workstation Support Group will NOT supply, and what flexibility users are allowed in modifying the system files.

1. Fully Supported (FS)

The FS systems are fully administered by the Workstation Support Group, which supplies all hardware support, maintains all configuration files, and supports all standard system and application software. The Workstation Support Group also establishes and maintains all accounts, and performs regular backups. *superuser* access is controlled

and limited to designated individuals who are in support positions, such as members of the Workstation Support Group and other staff or vendor system support personnel.

2. Modified for Application Development (MAD)

The MAD systems are used to develop application software or for *beta* testing of new system software. These systems normally have experienced users who require *superuser* access.

The Workstation Support Group supplies all hardware support, maintains all configuration files, and supports all standard system and application software. The Workstation Support Group also establishes and maintains all accounts, and performs regular backups. *superuser* access is NOT limited to support personnel.

3. Systems Application Development (SAD)

The SAD systems are used to develop system software or test hardware where the users require the right to modify and change the system configuration as part of their work.

The Workstation Support Group supplies as much support as is practical, given the changing nature of the system. If the user maintains the underlying system in a standard NAS configuration, the Workstation Support Group will supply the same level of support as for MAD systems, but they reserve the right to deny support if the system is too far out of alignment with the baseline configuration.

The Workstation Support Group establishes and maintains all standard accounts, and supplies regular backups when the development activity does not prevent the use of normal backup procedures. *superuser* access is controlled by a designated system manager (normally the primary user) and procedures have been established to allow the Workstation Support Group personnel *superuser* access.

4. Hardware Only Supported, Engineering Development (HOSED)

The HOSED systems are those used in any manner that precludes being maintained in the standard NAS configuration, or will not allow *superuser* access to members of the Workstation Support Group. In

Type of System	FS	MAD	SAD	HOSED
SGI IRIS 2500T	19	4	2	
SGI IRIS 3030/3130	13	4		
SGI IRIS 4D	5	10		1
SUN 3	20		1	3

Table 2: Workstations in each support category

such a case, the Workstation Support Group will only help coordinate the analysis and repair of hardware problems.

A listing of what types and numbers of systems are in each category, Table 2, shows that the majority of the users have elected to relinquish flexibility and control of their systems in exchange for reliability and support.

2.2 Support Functions Available

1. Hardware Support

All systems are under a maintenance agreement for vendor hardware support and the Workstation Support Group coordinates the handling of all hardware problems.

2. System Software Support

System software is supported for those systems under Workstation Support Group control. As discussed above, the level of control will determine the level of support.

3. User File Backups

It is our intention to provide backup services if possible. However, this depends on simple logistics, network performance, the availability of the proper tools, and man power. Our goal is to backup user files on a monthly basis. The current schedule for the IRIS workstations is one full backup per month, with an incremental backup each week.

The performance of backups also depends on the classification of the system and the ability to install, control, and maintain the needed tools.

4. *superuser* Access

Access to the *superuser* account depends on the classification of the system. Access to *superuser* determines the classification of the system as described above. In a UNIX operating environment, there are no access or write restrictions for the *superuser* account, so there are no protections against accidental or intentional damage to the system by individuals using the *superuser* account. Limiting access to the *superuser* account helps insure the integrity of the system files and software.

5. Account Control

Due to the implementation of Network File System (NFS), all users on NAS systems are required to have the same User ID (UID) number on all systems where they have an account. There is a procedure to allow users who control accounts on their own workstations to get UID and Group ID (GID) numbers for any accounts that they need to add. User accounts may not be added without getting valid UID and GID numbers for that account.

6. Network Addresses

Network addresses are assigned by the NAS Network Manager. All workstations are assigned a generic designation, such as wk00 for an IRIS, but may also have an alias, such as “chewbaka”. To have an alias entered into the host table, the user simply needs to send an electronic mail request to the Network Manager.

If at all practical, the Workstation Support Group would prefer to maintain all of the network configuration files on all systems. Because the files are constantly being updated by the Network Manager, a set of tools have developed to allow the automatic update of the network configuration files on all of the systems.

7. System Support Accounts

Certain system support accounts are required on all systems connected to the NAS network. These are standard user accounts that allow the operations support team to monitor and diagnose problems that have an impact on the entire network.

2.3 Installation and Classification of a New System

The Workstation Support Group is available to assist in the installation of any new systems. If the systems are to be classified as either FS or MAD systems, the Workstation Support Group will perform the installation and configuration of the system. If the systems are to be classified as SAD, they will assist in the installation and will install the needed system support tools. If the systems are to fall into the HOSED category, they will assist in the installation, only if requested.

3 Configuration of NAS Workstations

3.1 Standardization of file system structure

The policy at NAS is to keep the UNIX distribution directories as close to the original vendor distribution as possible, so that any new system upgrades can be installed with a minimum of system modification. Therefore, local files are not normally placed in the distribution directories. Unfortunately, for technical and historical reasons, some local files have to be placed in the system file areas, such as all of the configuration files in */etc*.

3.1.1 Executable programs

Executable programs fall into four classifications.

1. Vendor Distributed System Files.

These are files and programs that are supplied by the manufacture of the system and are maintained in their specified locations, such as */bin* and */usr/bin*.

2. Locally Installed and Supported Programs

These include either purchased products or locally developed tools that NAS has agreed to install on a supported basis. They are installed under */usr/local*.

3. Unsupported Programs

These include any programs or tools that NAS has not agreed to support. They are installed under */usr/unsupported* and can be installed and maintained by individuals who have access to the *unsupp* group.

4. Personal Programs

Users may obtain or develop their own tools. These are maintained in their own personal file system.

3.1.2 File Backup and Recovery

Full backups are done on a monthly basis, with weekly incrementals. Tape retention is for three months.

All locally modified system files, such as */etc/passwd*, */etc/group*, and */etc/sys_id*, are “archived” daily to an archive directory under the home directory of a system support account. In addition, selected files, such as */etc/passwd*, are copied nightly to a central, administrative system.

3.1.3 User’s Home Directories

The user files on all systems are contained in a three level, generic directory structure. The first level is */u*. The name of the second level consists of two letters such as *aa*, followed by the user’s home directory in the third level. For example, */u/aa/george*, or */u/nc/george*. Currently, the second level directory names are keyed on the name of the system (i.e., *aa*, *ab*, *ac* for the system ALPHA, *ba*, *bb*, *bc* for the system BRAVO, and *wk* for all workstations).

As a result, this directory structure provides the following capabilities:

- All user directories are located under a single top level directory
- All user directories reside at the same level
- New file systems can be added under */u* with little or no impact to the user community
- Home directories are not bound by organizational constraints, such as having all programmers in */usr/develop* and all non-programmers in */usr/people*, providing more flexibility in disk space distribution.
- The file systems are easily managed by system administrators, including the tracking of disk usage, disk space allocation, and load balancing.

3.2 Standardization of Disk Partition Sizes

The goals are to simplify the maintenance and repair of the system partitions and to make the maximum amount of disk space available to the workstation users. By keeping the system partitions, *root* and */usr*, as small and as identical as possible, it is only necessary to maintain one master backup for all of the workstations, greatly simplifying system restorations and decreasing the time spent on backing up the system files.

This is done by moving local files (where applicable) to the */u* partition by using symbolic links to redirect files to two directories on the */u* partition (*/u/.private* and */u/.links*).

More specifically, the following local files have been moved to the directory */u/.private* :

- local, volatile files (i.e., user files such as those in */usr/mail* and log files such as those in */usr/adm*)
- workstation specific files which are located under the *root* partition or the */usr* file structure (configuration files)
- system configuration files (i.e., */usr/lib/sendmail.cf*) are only relocated if possible. Note that the critical system files under */etc* are not relocated.

The following non-system files are moved to the */u/.links* directory:

- most of the non-system related files that are normally located on the */usr* partition (i.e., man pages, large font libraries, etc.)
- the directories */usr/local* and */usr/unsupported*

This configuration allows a small partition for the */usr* system files, while allowing many of the larger file trees (such as */usr/local* and */usr/unsupported*) to be placed on the larger */u* disk partition. By having the large, non-system related files gathered together under one directory, */u/.links* becomes a prime candidate for a NFS mount point. By using symbolic links to point to files under a single directory, we have been able to use a single NFS mount point to supply over 100 megabytes of common files to the workstation.

4 Workstation Support Tools

4.1 System Tools for File System Maintenance and Cleanup

One of the most time consuming aspects of system administration is the maintenance of the file system. Besides the traditional running of the program *fsck* to check file system consistency on reboot, the system administrator has to insure that unneeded files are deleted, log files are kept small, and that the system files remain intact, with the correct permissions and ownership. This section describes some of the tools use on the NAS program to automate and simplify these tasks.

4.1.1 skulker

A major part of file system maintenance is the removal, on a regular basis, of files that clutter the system and waste disk space. There is a program called the *skulker* [4] that is run daily at 02:30 A.M., which finds and deletes files with names that match specifically designated patterns and have not been accessed within a specific time.

The name pattern for the files on each system, and the lifetime for each file, can vary. The actual list of files and their periodic deletion rates can be obtained on each system by looking at the script */usr/local/adm/skulker*. Some examples are:

- All files with names that begin with a comma ',' are considered to be temporary files, with a maximum retention time of eight (8) days.
- All files that are located in the */tmp* and */usr/tmp* directories are considered temporary files, subject to removal at the discretion of the system administrators. These files are removed after each reboot of the system and with a maximum lifetime of eight (8) days.
- Any of the scratch files left behind by the *emacs* editor, including files ending in *.BAK* and *.CKP*, which are deleted after eight (8) days.

4.1.2 `agelog`, `Archive.logs`

Another important aspect of file system maintenance is the control and use of log files. To assist in controlling the growth of the system log files, all log files are archived by the program `/usr/local/adm/Archive.logs`. This archiving is done at regular intervals and is determined by how fast the log actually grows. `Archive.logs` maintains a collection of back copies of system log files in the directory `/usr/spool/log/archive`. Even if a log file is not under the control of `Archive.logs`, the system administrator is requested to make a notation of the existence of the log file, in the form of a comment, in the `Archive.logs` script. This makes it easier to file log files. In addition, as part of the effort to make log files easy to locate an attempt is made to adjust system programs to maintain their log files in the directory `/usr/spool/log`.

`Archive.logs` uses the script `agelog` to archive the files. `agelog` appends a “00” serial number to the basename of the log file name after “incrementing” the numbers on previously `aged` log files in sequential order so they are kept in order by age. `agelog` is an enhanced version of the `syslogswap` script [5], which supports two digit numbers and an optional directory where the archived log files are stashed.

4.1.3 `rdist`

Installing and updating files on over thirty systems is a long, time consuming process. If done manually, it can also lead to many errors. The BSD 4.3 program `rdist` (remote file distribution) simplifies this aspect of system maintenance. `rdist` was ported to the IRIS workstations as part of an effort to develop tools to aid in system administration [6]. `rdist` compares file statistics between workstations, using one system as the baseline, and either reports all changes which are necessary or updates the target system to match the baseline system. The major advantage is only files that do not match are updated and only files that are missing are installed. When validating and updating fifty files in a directory, the ability to skip over the unchanged files greatly shortens the validation and update time.

4.1.4 `trsh`

`trsh` (Treeshell) is a locally developed program [7] that simultaneously connects a terminal to interactive shell processes on a collection of workstations.

It allows a system administrator to simultaneously execute a command on multiple workstations at the same time. *trsh* is very helpful for routine operations such as creating a new directory on all of the systems or updating the file permissions for one file on all workstations.

4.1.5 **binaudit**

When dealing with a large number of systems, knowledge of any modifications or changes to files in the system directories is critical to prevent or correct system problems. Run nightly by *cron*, *binaudit* [8, 9, 10] scans a list of file systems and compares the results with information in a master file, which is created the first time *binaudit* is run. *binaudit* ignores any changes (additions, deletions) to the file system if the name of the changed file matches one of a set of specific *egrep* patterns. When completed, *binaudit* mails a description of all other differences found to the auditor.

The *binaudit* package can maintain the master files on a central mainframe or minicomputer system for a large collection of workstations. The NAS workstations are currently audited from one central system. However, as currently implemented, *binaudit* audits each system on its list sequentially, which leads to the audits of some systems still being processed during the day. In the future, the systems will be divided into file server-related clusters, and audits on each system will be run from their primary file server. Note that both *binaudit* and the *perms* program mentioned below, only take a “snap shot” of the system each time they run, and compare it with a previous “snap shot”. This means they are only useful for comparing and monitoring long-term changes, rather than short-term (hourly) changes.

4.1.6 **perms**

Based on the program in the UNIX System Security book [11], *perms* is used to set and check the ownership and mode of directories. It is used to create all of the needed system directories such as */usr/spool/log* when a new system is installed, and is run once per week to insure that the permissions and ownership of the system directories are not accidentally changed.

4.2 System Tools for System Administration

This section describes some of the tools that we use to simplify the day to day system administration tasks.

4.2.1 `addacct`

addacct [12] is designed to install, delete or change user accounts. When invoked without an argument, it simplifies and streamlines the process of adding accounts by collecting and verifying all the new user information, updating the password file and creating the new user's home directory. *addacct* then runs a simple shell script (*addacct.sh*) which copies a set of default files to the new user's home directory and other desired actions, such as sending a welcoming electronic mail message to the new user. When called with an argument, *addacct* can be used to disable or change an existing account.

4.2.2 `time.sh`

The IRIS 2500T, 3030, and 3130 workstations do not have a real time clock, and often will reboot with the wrong time set. *time.sh* is a shell script which polls a master system to determine the time. It is run at reboot time and twice each day to reset the workstation's clock.

4.2.3 `netmotd`

netmotd is a shell script that allows the system administrators on the central administration machine to update and control the "message of the day" (*motd*) file on all of the NAS production systems. The script allows the updating of the *motd* file on individual systems, a class of systems, such as workstations, or all of the systems.

4.2.4 `network tools`

A set of shell scripts are used by the NAS Network Administrator to maintain the network tables on all of the systems. These scripts use a standard system account to transfer the files to the supported system where a *cron* command updates the actual system files.

4.2.5 `lsu`

Enhancing the standard UNIX `su` program, the locally developed programs `su`, `nsu`, and `lsu` [13] incorporate the ability to restrict access to alternate accounts by account name, time periods, terminal ports, and baud rate. `su` is identical in performance to the standard UNIX `su`, bugs and all, while `nsu` properly sets all of the needed environmental variables to make it appear that you actually logged in to the selected account. The `lsu` program allows access to an alternate account to be given to individuals on a selected machine, without giving them the password for that account. This is accomplished by using the password for the invoking account as the validity check, instead of the password for the target account. On the NAS workstations, the `superuser` password is normally the same for all of the systems in one administrative cluster. By using the `lsu` program, `superuser` access can be given to one user on one system, without telling them the password that would give them `superuser` access on all of the systems in that cluster or without having to give their system a unique password. Another alternative would be to give each of these users a personal `superuser` account. Experience has shown that users are more likely to use personal `superuser` accounts for routine activities than when they use a program such as `su` or `lsu` to get `superuser` privileges.

4.2.6 `serverd`

The `serverd` [14] monitors the remotely mounted file systems on the local host. The `serverd` checks the file systems listed in the `imports` file and if the file system is no longer accessible, it will unmount the bad file system and attempt to replace it with a file system from an alternate file server.

4.2.7 `userval`

`userval` is part of a collection of shell scripts that copy the password file for all systems to a central archive point, report any significant changes, and build a text file for the generation of user account reports.

4.3 User Tools for System Administration

4.3.1 `usrboot`

With the sophisticated graphics programs that are being used by the workstation users, it is quite possible for a user to get his/her workstation into an unusable condition, requiring a system reboot to reset everything. A user executable program, *usrboot*, has been supplied to the users to allow them to reboot the system without the assistance of a workstation system administrator.

4.3.2 `nfsmount`

To simplify the sharing of data files between workstation users, a user executable program, *nfsmount* [15], has been supplied to the users to allow them to mount, via NFS, the user file systems on different workstations without the assistance of a workstation system administrator. Currently, *nfsmount* limits the user to five remote mounted file systems.

4.3.3 `sob`

sob (simple operator backup) [16] is used to back up files using a *tar* format tape archive. *sob* has two major features from a system administrative point of view. The first is the ability to copy an arbitrary list of files onto a 40 megabyte cartridge tape drive on the workstations, prompting for the necessary number of tapes, one at a time. The second feature is that *sob* maintains a listing of the tape contents, allowing the system administrators to identify the location of a file they have been asked to restore. *sob* was developed to support system backups, but has been found useful by the user community for making copies of their files without having to worry about the tape size limitations.

4.3.4 `gethome`

The *gethome* program returns the full path name of a user's home directory. Since all system administration scripts are required to run on all systems, the home directory of the support account will be different on every system. Using *gethome* in the script enhances the portability of the script. Also, on

the larger systems it is often necessary to relocate a user's home directory to a different disk partition. The users are advised to use the program *gethome* in all of their shell scripts.

4.4 Availability of Workstation Support Tools

Local Package	Available	Comments
addacct	yes	available
agelog	yes	available
binaudit	yes	available
gethome	no	not packaged
lsu	yes	available
netmotd	no	not packaged
network tools	no	not packaged
nfsmount	yes	available
perms	yes	available
serverd	no	not packaged
skulker	no	this is a simple script, see [4]
sob	yes	available
time.sh	no	available
trsh	yes	available
usrboot	yes	available
userval	no	not packaged

Table 3: Local Package Availability

Some of the locally developed software packages are available for distribution. To obtain a copy of one of these packages, a letter on corporate stationery requesting the individual packages is required. They will be supplied for use by the requestor only and should not be distributed outside of the United States. Please note that these packages are still under development and constructive suggestions regarding alterations or improvements would be appreciated. Table 3 lists the status of the various program packages.

References

- [1] F. Ron Bailey. NAS – current status and future plans. In *Supercomputing in Aerospace*, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035, 1987.
- [2] Bruce T. Blaylock and F. Ron Bailey. Status and future developments of the NAS processing system network. In *Third International Conference on Supercomputing*, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035, 1988.
- [3] Michele Crabb. *CFD Graphics Workstation Software System Overview*. Technical Report Project 2000-949 Technical Note No. 5, Sterling Federal Systems, Inc., 1121 San Antonio Road, Palo Alto, CA 94303, 1988.
- [4] Dr. Rebecca Thomas. The wizard’s grabbag. *Unix/World*, April 1988.
- [5] Dr. Rebecca Thomas. The wizard’s grabbag. *Unix/World*, January 1987.
- [6] Eric Raible. Remote workstation administration in a supercomputing environment. In *3rd Annual Networking and Supercomputer Workshop*, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035, 1987.
- [7] David A. Tristram. *trsh – Treeshell*. NAS Systems Division, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035. Private communications.
- [8] Matt Bishop. The RIACS intelligent auditing and categorizing system. Research Institute for Advance Computer Science, NASA Ames Research Center, Moffett Field, CA 94035. – Submitted to 1988 Workshop on Unix Security.
- [9] Matt Bishop. Security monitoring. 1987. Research Institute for Advance Computer Science, NASA Ames Research Center, Moffett Field, CA 94035.
- [10] Matt Bishop. Auditing files on a network of UNIX machines. In *UNIX Security Workshop*, USENIX, 1988.

- [11] Patrick H. Wood and Stephen G. Kochan. *UNIX System Security*. Hayden Book Company, 1985.
- [12] Jonathan Hahn. *addacct*. General Electric Corporation, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035. Private communications.
- [13] Matt Bishop. Managing superuser privileges under UNIX. 1986. Research Institute for Advance Computer Science, NASA Ames Research Center, Moffett Field, CA 94035.
- [14] Gary Veum. *serverd*. Sterling Federal Systems, Inc., NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035. Private communications.
- [15] Gary Veum. *nfsmount*. Sterling Federal Systems, Inc., NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035. Private communications.
- [16] David A. Tristram. *SOB – Simple Operator Backup utility*. NAS Systems Division, NASA Ames Research Center, Mail Stop 258-2, Moffett Field, CA 94035. Private communications.