From: Burton Rosenberg
To: Brian Coomes
Re: Computers
Date: October 13, 1994

   This memo documents the efforts made over the past two years to rationalize the file system structure of the department's computers. This rationalization was needed because of the quick growth of computing resources within the department. The old system did not extrapolate well during this growth. Although we would like to remain as typical as possible in our file system layout, reducing to a minimum customization and complicated site specific rules, our computer network is apparently much more decentralized and less vendor loyal than other installations, because the standard recommendations by do not completely suit this department's needs.

Overview

Our goal is to have a uniform space of names throughout the network independent of the physical layout of files on any machine. The reasons for this are,

1. It is too time-costly to insure a single physical standard throughout the network. Old machines would have to keep pace with new machines in so far as software updates.

2. Machines can exist on the network with operating systems that do not correspond well with the dominate model, which at this time is Sun-OS.

3. Machines can exist on the network with special requirements or privileges.

4. The physical layout corresponds more to the requirements of backup and efficient use of disks and networking than the logical layout of exported home directories and software products.

The separation of physical and logical location of files is achieved using *indirect automount.* This is an NFS software technology which uses configuration files, typically `/etc/auto.master` and `/etc/auto.directory-name`, to automatically mount remote file systems and create symbolic links from the automount directory to the mounted file.

The proposed directory structure, which to some extent has already been put into practice, is more easily explained after the introduction of some terminology. The logical locations where everyone will find shared resources are called *well known directories.* They are subdirectories of the root, some of which are configured using automount. Currently there are two well known directories configure by automount, *home* by file `/etc/auto.home`, and *local* by file `/etc/auto.local`. A machine in the network might or might not conform to our naming scheme. A machine which is fully conforming is said to have a *model directory structure.* An additional well known directory of a model directory structure is *exp*, also known as *export* on Sun machines. In the model, all site-specific software physically resides as a descendent of the exp directory.

Home directories are organized under subdirectories of the `/home` well known directory. It our experience, home directory systems come in two flavors: *shared*, or more correctly *machine inspecific*, and *private*, or more correctly *machine specific.* Each shared home directory system is given a name, for example alachua, and linked onto `/home`. For example, at `/home/alachua`. If the model directory structure is followed, alachua will reside at `/exp/home/alachua` on whatever server is exporting this home directory system.

The local software packages are slightly more difficult to organize. To achieve efficiency while allowing flexibility, we define a *canonical /usr/local*, one of which is maintained for each machine architecture available. In the present situation, the architectures are mips and sparc. If any computer wants a canonical `/usr/local`, it is available at `/local/mips` or `/local/sparc`, according to architecture desired. Transparency and backward compatibility are maintained by a soft link from `/usr/local` to `/local/arch-type`. The physical location of the canonical `/usr/local` is on some computer, its model

name is `/exp/local/mips` or `/exp/local/sparc`. (Or `/export/local/sparc`.)

There are three other standard subdirectories of the local well known directory. For completeness, all four are enumerated here:

1. `/local/arch-type`

2. `/local/package-name`

3. `/local/machine-name`

4. `/local/contrib`

Certain packages are exported under their own name. Machines which have there own local file system can remain consistent with the overall naming scheme if their `/usr/local` is linked to `/exp/local/machine-name`. This maverick file system's import-export location would be `/local/machine-name`, using automount as demonstrated in the next section.

Examples

The department's shared home directories are named alphabetically by Florida counties. Currently alachua, broward, collier and dade exist. Alphabetically, Florida counties continue: escambia, flagler, glades, hendry, indian-river, jackson, lee, manatee, nassau, osceola, pasco, st-lucie, taylor, union, volusia and wakulla.

Almost every machine has its machine-specific home directory, and this is especially important if the machine has significant disk capacity. For the machine passaic, the home directory is logically at `/home/passaic`, an all machine reference it at that location. Physically on passaic, it is located at `/exp/home/passaic`. To maintain backwards compatibility, `/usr/users` is linked to `/exp/home/passaic`. In general, a link from `/usr/users` to some appropriate server is made for each machine.

Among the older directories, `/home/paris` is the most desperately non-conforming to this new naming scheme. `/home/paris` was the root of both home directories and exportable packages. Some packages were actually on `/export/package-name`, a few were physically at `/package-name`. The home directories exported by paris were at `/home/paris/users`. Compatibility with both situations was maintained by remount the old `/home/paris` on

/tempf/paris and arranging symbolic links for backward compatibility. The situation is similar for mthvax, its wild-card filesystem is at /tempf/mthvax on paris.

The automount file /etc/auto.master indicates which files to consult further for configuration information. The file /etc/auto.home configures the well known home directory. Here are /etc/auto.home files for several computers:

```
#  /etc/auto.home for cs.cs
#
#key     option  location
alachua          cs:/exp/home/alachua
broward          cs:/exp/home/broward
collier          paris:/home/collier
dade             paris:/home/dade
paris            paris:/home/paris/users
passaic          passaic:/home/passaic

# auto.home map for paris
#
# key                location
paris                paris-gw:/tempf/paris
#
alachua              cs:/home/alachua
broward              cs:/home/broward
collier              paris-gw:/export/home/collier
dade                 paris-gw:/export/home/dade
#
passaic              passaic:/home/passaic
anatolia             anatolia:/home/anatolia
atlanta              atlanta:/home/atlanta
cal                  cal:/export/home/cal.home
israel               israel:/home/israel
michelle             michelle:/home/michelle
nanjing              nanjing:/home/nanjing
oakland              oakland:/home/oakland
soho                 soho:/home/soho
```

```
sweden                    sweden:/home/sweden
warsaw                    warsaw:/home/warsaw

# auto.home map for passaic.cs
#
#key              mount-options    location
paris                              paris:/home/paris/users
cal.fac.burt                       cal:/cal/fac/burt
alachua                            cs:/home/alachua
broward                            cs:/home/broward
passaic                            passaic:/exp/home/passaic
dade                               paris:/home/dade
```

Canonical /usr/local's are maintained for mips and sparc architectures at `/local/mips` and `/local/sparc`, respectively. It is exported by a server from this locating and imported by a client at this location. For backwards compatibility, a link from `/usr/local` to `/local/x` is created, where x is the appropriate architecture code. The tex package, including Amstex, Latex, Metafont, dvips and xdvi, is located at `/local/tex`. This package supports multiple architectures by having bin directories dot qualified by the architecture code, `/local/tex/bin.mips`, for instance.

Not all software traditionally located in `/usr/local/lib` is architecture insensitive. So the idea of having a bin.arch for `/usr/local` is incorrect. However, for random, goof-ball software that can exist under such a restriction, and wishes to be widely distributed, a `/local/contrib` directory is maintained. All binaries appear simultaneously in this directory, with architecture choice made correctly setting the user's search path correctly according to the architecture type returned by arch or mach. Furthermore, a machine requiring an entirely independent `/usr/local` can locate it at `/exp/local/machine-name`, link `/usr/local` to this location, and make it available at `/local/machine-name`.

The local well known directory is configured by `/etc/auto.local`. I would like to remind the reader that auto.local can give several locations from which to mount a copy of `/local/tex` or `/local/sparc`. This provides automatic load-balancing and redundancy for important file services. If one server crashes, the user will remount the package from a running server.

To maintain greatest transparency, even the server should define a `/local/mips` and a `/local/tex` location. Clients request packages are these locations.

5

The software, according to the rule for site-specific software, is in fact physically located at an appropriate subdirectory of the exp well known directory. The naming convention is consistent with that of home. There would be an `/exp/local/tex`, an `/exp/local/sparc`, and so on.

Here are the `/etc/auto.local` files for several of the department's computers. As you can see, nothing here really conforms to the standard. We had only recently come to a decision about the local software structure and have not completely updated the machines.

```
# /etc/auto.local on paris
#
#key     option  location
tex              paris:/export/local/tex

# /etc/auto.local on passaic.cs
#
#key     option  location
passaic          passaic:/exp/local/passaic
contrib          cs:/local/contrib
tex              cs:/local/tex \
                 passaic:/exp/local/tex
cs       -suid   cs:/usr/local

#  /etc/auto.local for cs.cs
#
#key     option  location
passaic          passaic:/local/passaic
tex              cs:/exp/local/tex
contrib          cs:/exp/local/cs
mips             cs:/exp/local/cs
```

Backup

It is necessary to backup each night home directory software. Perhaps tar or cpio is a good choice, since the computer does not have to be brought to single user mode. The local directories need to be backed up after each installation of new local software. This can be done on demand, again cpio or tar seems sufficient.

The remaining partitions need to be backed up only once, after installing the system. And dump should be used from single user mode. There is a small exception of the /var/spool/mail, /var/log and /etc subdirectories. These are normally changing directories, important for the functioning of the system. Perhaps weekly tar or cpio's are all that is required.