The PPPL Unix Cluster

Charles Karney

1 Introduction

The PPPL Unix Cluster is a collection of Unix computers available for general use. This document gives some basic information on how to use the Unix Cluster.

The computers discussed in this document are paid for by the taxpayer, used by many people, and connected to the Internet. Since these computers are used in research, relatively few restrictions are placed on the way you use these computers. Thus, as a user of these computers you have certain obligations:

- Make sure you pick a good password (see Section 1.4 [Choosing and Setting Passwords], page 3). We need to be sure that only authorized users gain access to the computers. Passwords are the main line of defense against computer abuse. A password that is too short or is a common word can be guessed by malicious outsiders.
- Please be judicious in your use of the computers. It's easy, for example, to overwhelm the network, fill up the disks, etc. If someone notices you doing this, you may be asked to modify your behavior. We'll try to find other ways for you to accomplish your work efficiently.

1.1 Goals

The Unix Cluster has following characteristics:

- A single account grants you access to all computers. Once you've logged into one of the computers, gaining access to the other computers usually does not involve re-entering your password.
- There's a single "home" directory for each user which is available on all the computers. You can edit your files on one machine and compile and run them on another machine.
- There's a single E-mail address for each user, and a single system file for E-mail.
- Most of the file systems are shared between computers.
- Most of the same software is available on all the computers.
- Various other services are provided by the cluster and are available to users from any computer. These include: printing, news, web publishing, anonymous ftp, fax transmission and reception.

This means that it usually doesn't much matter which computer you use. Typically, you might be content just to use a single computer. However, if that computer becomes too slow or is scheduled to be down, then you can move to another computer more-or-less transparently.

1.2 Members

The following computers are members of the Unix Cluster

Name	Make	Model (CPUs)	Operating System
lyman	Sun	Sparc 10-521 (2)	Solaris 2.5.1
carl	Sun	Sparc 10-521 (2)	Solaris 2.5.1
orion	Sun	Ultra 2 (2)	Solaris 2.6
lupus	Sun	Sparc 10-421 (2)	Solaris 2.5

ron	Sun	Sparc 20	Solaris 2.5
tom	Sun	Sparc 10-41	Solaris 2.5.1
frodo	Digital		OSF1 4.0b
wave	Digital		OSF1 4.0b
hydra	Digital	(4)	OSF1 4.0d
tern	Intel	Pentium 166MHz	Linux 2.0.36
jupiter	Digital		OSF1 4.0d
venus	Digital		OSF1 4.0d
mars	Digital		OSF1 4.0d
saturn	Digital		OSF1 4.0d
mercury	Digital		OSF1 4.0d
neptune	Digital		OSF1 4.0d
taurus	Sun	Enterprise 450 (2)	Solaris 2.6
solo	Sun	Sparc 5	Solaris 2.6
ead01	Sun	Ultra 5/10	Solaris 2.6
ead02	Sun	Ultra 5/10	Solaris 2.6
ead03	Sun	Sparc 20 (2)	Solaris 2.5
pppl	Sun	Sparc 2	SunOS 4.1.4

The fastest computers in the Cluster are taurus and the "planets" (venus, mercury, etc.). Use these for your heavy number crunching needs. pppl (which is also known as pobox) is reserved for handing E-mail, and you should not normally use this computer. The other computers are fine for interactive use.

The full Internet names for these computers are formed by adding ".pppl.gov" to the name, e.g., "taurus.pppl.gov". (But "pppl.pppl.gov" is really called "pppl.gov".)

1.3 Accounts

For an account on the Unix Cluster, contact Lena Scimeca (x2833, E-mail: `unixadmin@pppl.gov`). She will need to know:

1. Your full name
2. Your prefered username. Your username is limited to 8 characters. By default, the choice of username is your first initial and last name; however, if that is more than 8 characters, or if you already have accounts under another username, you may want to choose another username (such as your last name only or just your initials).
3. whether you already have an account on NERSC (and if so, what your username is there).

You will be given a *home directory* when your account is created which will be called '`/u/`*username*', however in many contexts you can also refer to your home directory by the abbreviation '`~`'. (Similarly you can refer to home directory of user jsmith either with '`/u/jsmith`' or '`~jsmith`'.)

By default, there's a quota of 20MB on your home directory.

1.4 Passwords

Your account will be assigned a password initially. You should change this as soon as possible after getting your account set up. On the Unix computers, type

```
passwd
```

while on any of the computers in the Unix Cluster (except for Linux computers).

You will be prompted for your old password and asked to select a new password (which you need to enter twice, to guard against typos—the passwords are not echoed on the screen). This changes your password on *all* the Unix Cluster computers.

Here are some guidelines for choosing a password:

- Passwords must be at least 6 characters long.
- Passwords must be changed at least once a year.
- Passwords should not be a word, username, or personal name.

In addition, the following recommendations will make your password more difficult to guess:

- The longer the better—however, only the first 8 characters count on Unix systems.
- Use the password generator on the VAX. (Type `SET PASSWORD/GENERATE=6` on the VAX cluster and keep hitting ⟨RET⟩ until you see a password you like. Type *`CTRL-z`* to get out.)
- Mix upper and lower case; include digits or punctuation.

1.5 E-mail

E-mail is the standard way for users to communicate with one another on computer systems. In addition messages from the computer (indicating, for example, that you need to change your password) are sent via E-mail.

On the Unix Cluster, your E-mail address is simply

username`@pppl.gov`

Important: You must ensure you can (and do) read mail sent to this address. You have three choices for how your mail is handled:

1. You leave the mail on the Unix Cluster and you use some mail reader on the Unix Cluster to read your mail.
2. You leave the mail on the Unix Cluster and you use a POP client on a desktop computer (e.g., Eudora on a Mac) to access your mail via a POP server on pppl.gov.
3. You have your mail forwarded to another computer system.

1.5.1 Forwarding

Forwarding of mail is governed by a *forward* file. On traditional Unix systems, this is called '`~/.forward`'. However on the Unix Cluster, you should place your forwarding address in '`/usr/mail/.forward/`*username*'. This file should contain the complete E-mail address of where you want the E-mail sent. E.g.,

```
echo zarnstorff@vaxcluster.pppl.gov > /usr/mail/.forward/zarnstor
```

A partial verification of the forwarding is provided by `finger`

```
$ finger zarnstor@pppl.gov
...
Mail forwarded to zarnstorff@vaxcluster.pppl.gov.
```

However, you should also test it in case you've created a forwarding loop (where the mail is sent around in circles). Send yourself some mail and verify that it arrives OK.

If you want to stop the forwarding of mail, just delete '`/usr/mail/.forward/`*username*'.

If forwarding doesn't look like it's working, it may be that a system-supplied E-mail alias is over-riding it. Check with `unixadmin@pppl.gov`.

For more information see forward(4).

1.5.2 Aliases

For those people whose usernames do not conform to the standard of initial+lastname, there should nevertheless be an *alias* which directs E-mail from initial+lastname to the correct destination. For example, John Krommes' username is krommes and so his E-mail address is krommes@pppl.gov. However jkrommes@pppl.gov is an alias of krommes@pppl.gov. You can verify the existence of such an alias by using `finger`:

```
$ finger jkrommes@pppl.gov
The alias jkrommes expands to John Krommes <krommes@pppl.gov>.
```

Aliases are set up by a system administrator. If you think the aliases for you are not correct, send E-mail to `unixadmin@pppl.gov`.

1.6 Getting Help

Since Unix is now more than 20 years old, there are several disconnected ways in which documentation is provided to the user. The main ways are man pages, info, the web, command-line prompts and other on-line aids, specialized document viewers, miscellaneous files, and finally printed manuals and books. Finally, it sometimes necessary to ask someone for help.

1.6.1 man pages

The original way in which most aspect of Unix was documented was via "man" pages. Traditionally these were bound into "sections", with section 1 containing user commands, section 2 containing system calls, section 3 containing library routines, etc. A man page designation ls(1)then documents the command "ls" in section 1.

The simplest way of reading a man page is via the command man(1).

1.6.2 info

1.6.3 Documentation on the web

1.6.4 Other documentation

1.6.5 Human Help

The `makeinfo` and `texinfo-format-buffer` `http://w3.pppl.gov` See ls(1). commands transform a Texinfo file such as this into an Info file; and TeX typesets it for a printed manual.

2 File systems

2.1 Home directories

2.2 Other writable directories

2.2.1 scratch

2.2.2 tmp

2.2.3 work

2.2.4 VMS

2.3 AFS

AFS, the Andrew File System, is a wide area file system providing

1. a single namespace under '`/afs`'
2. caching to a local disk to improve performance
3. access control lists on directories

Online documentation is accessible at `http://w3.pppl.gov/misc/afs/unix/Html/index.htm`

Printed documentation is accessible at `http://w3.pppl.gov/misc/afs/unix/PostScript/`

AFS Frequently Asked Questions

3 Networking

Here we describe some of the networking commands in detail. These enable you to access other systems from the Unix Cluster and vice versa.

3.1 Logging into another computer

3.1.1 telnet

3.1.2 rsh and rlogin

3.1.3 slogin and ssh

ssh(1)and slogin(1)are "secure" replacements for rsh(1)and rlogin(1). You may wish to use them because:

- They provide a convenient method for accessing the Unix Cluster from off-site, since (a) rsh access to the Unix Cluster is restricted to a few domains, and (b) ssh optionally offers compression which offers improved throughput on low bandwidth connections.
- Finally, ssh knows how to forward AFS tokens to remote machines, so you may wish to use ssh within the Unix Cluster.

The primary documentation is via the manpages: ssh(1), sshd(8), slogin(1), scp(1), ssh-keygen(1), ssh-agent(1), ssh-add(1).

The main ssh web site is `http://www.ssh.fi/sshprotocols2/`. (Note that the Unix Cluster is currently running version 1.2.x of ssh.)

A nice tutorial on ssh is at `http://www.tac.nyc.ny.us/~kim/ssh/`. This includes pointers to more information.

There is also `http://w3.pppl.gov/~karney/ssh-sem.txt` with PPPL specific information.

There are three primarly ways in which a connection to a remote host can be authorized by ssh.

1. by prompting for a password for the remote host
2. by an rsh-style file named '`~/.rhosts`' (or '`~/.shosts`') on the remote host. See rhosts(4)
3. by the use of private and public keys, so-called *RSA authentication*

The first two should be easy to use. The third requires some a little more work to set up, but is more secure and does not require that the host you're connecting from be already known to the remote host.

3.1.3.1 ssh with RSA authentication

You need to do the following *once*:

1. Run ssh-keygen as follows

   ```
   $ ssh-keygen
   ...
   Enter file in which to save the key (~/.ssh/identity): [RETURN]
   Enter passphrase: [ENTER A GOOD PASS PHRASE TO PROTECT THE PRIVATE KEY]
   Enter the same passphrase again: [DITTO]
   ...
   ```

   This creates a private-public key pair in ‘`~/.ssh/identity`’ and ‘`~/.ssh/identity.pub`’. The first file is protected by (a) Unix permissions, and (b) your pass phrase. The second file only needs to be protected against writing by anyone except you.

2. Create ‘`~/.ssh/authorized_keys`’:

   ```
   cp ~/.ssh/identity.pub ~/.ssh/authorized_keys
   ```

3. Create ‘`~/.ssh/authorized_keys`’ on any remote systems you want to access. E.g., on hecate.princeton.edu

   ```
   mkdir ~/.ssh
   chmod 755 ~/.ssh
   ```

   and use ftp(1)to transfer the local copy of ‘`~/.ssh/authorized_keys`’ to hecate.princeton.edu.

4. Near the beginning of your ‘`~/.xsession`’, put

   ```
   eval `ssh-agent -s`
   ssh-add < /dev/null &
   ```

   OR

   ```
   eval `ssh-agent -c`
   ssh-add < /dev/null &
   ```

   depending on whether your ‘`~/.xsession`’ is a ‘`/bin/sh`’ or ‘`/bin/csh`’ script. Look at the first line of the file to find out.

   At the end of ‘`~/.xsession`’ (*after* the invocation of the window manager), put

   ```
   eval `ssh-agenet -k -s`
   ```

   or

   ```
   eval `ssh-agenet -k -c`
   ```

   Your ‘`~/.xsession`’ may then look like:

   ```
   #! /bin/sh
   . /etc/env.default

   eval `ssh-agent -s`                # ADDITION
   ssh-add < /dev/null                # ADDITION

   xrdb $HOME/.Xresources

   xterm -ls -T `hostname` -n `hostname` -geo 80x50+105+5 &
   twm
   ```

```
eval `ssh-agenet -k -s`            # ADDITION
```

5. [OPTIONAL] Create a file '~/.ssh/config' containing

   ```
   Host *.nersc.gov
   User u12345

   Host hecate.princeton.edu
   Port 1515
   ```

   where 'u12345' is your NERSC user name. (This step means that you won't have to supply a '-l u12345' argument to ssh.) The port-specification for hecate connects you to the AFS-aware version of sshd on that machine. If you want to use compression on certain connections, this can be configured here too (see ssh(1)).

Once per X session

When you start an X session on your X terminal, the ssh-add command in '~/.xsession' will pop up a window for you to enter your pass phrase.

3.1.3.2 ssh examples

3.1.3.3 Installing ssh on other machines

In order to access the Unix Cluster from a remote machine, you will need to have ssh installed on that machine. You should ask the system administrator to do this. In addition, if your site has AFS, ask for the AFS patches to ssh to be made. The AFS patches are available at `http://www.monkey.org/~dugsong/ssh-afs-kerberos.html`.

[In fact, most of the functionality of ssh is available even if it's not installed by a system administrator.]

3.2 Accessing files on another computer

3.2.1 ftp

3.2.2 rcp and scp

3.2.3 rsync

4 X windows

4.1 xrsh and xssh

xrsh(1)and xssh(1)are convenience programs which

1. use rsh or ssh to connect to a remote system
2. set DISPLAY appropriately on the remote system
3. invoke xterm(1)or other X client on the remote system

1. To get a window on a.nersc.gov, do

   ```
   xssh -auth ssh a.nersc.gov
   ```

   xssh is a ssh-capable replacement for *xrsh*. You will need to ensure that your PATH on a.nersc.gov is set in *.cshrc* and that it included the directory with xterm lives. For that reason, I suggest you move

   ```
   if ( -e /etc/cshrc.nersc ) then
     source /etc/cshrc.nersc
   endif
   ```

   from '`~/.login`' to the end of '`~/.cshrc`'.

   On a.nersc.gov, if you do

   ```
   echo $DISPLAY
   ```

   you will see it set to something like '`a:4.0`'. This is a special display setting used by ssh for encrypting your X traffic. Incidentally the encryption is only in effect between NERSC and the machine you ran xssh on. The traffic between this machine and your X terminal is still unencrypted.

2. To start a window on another machine in the Unix Cluster with your AFS tokens carried over:

   ```
   xssh lupus
   ```

   With this invocation, the X traffic is not encrypted. You may wish to insert these xssh commands into your .twmrc as follows

   ```
   menu "Programs"
   {
   "Machines"      f.title

   "lupus"         !"xssh lupus"
   "a.nersc.gov"   !"xssh -auth ssh a.nersc.gov"

   }
   ```

5 Miscellaneous software

A wealth of software is available on the Unix Cluster. The following include *very* brief descriptions of the software together with pointer to additional documentation.

5.1 emacs

Emacs is powerful text editor. If running under X, it will normally use its own window, so you would start it in the background with

```
emacs &
```

To prevent emacs from using its own window, specify the `-nw` option as in

```
emacs -nw file
```

Emacs is fully documented in section "Overview" in *The Emacs Manual*. However, for most people, 90% of what they need is covered by the online tutorial which is accessed by typing `C-h t` (control-H followed by the letter t) while in emacs. In addition here's a handy reference card. Print this on a duplex printer, if you have access to one.

5.2 TeX

TeX is a scientific typesetting package. We run the teTeX distribution version 1.0. This is documented at `http://w3.pppl.gov/misc/teTeX`. LaTeX latex(1)is the recommended macro package. The run it, use the command

```
latex file.tex
```

Useful (indispensible) books are:

- LaTeX, a Document Preparation System, by Leslie Lamport (2nd Ed, Addison Wesley, 1994)
- The LaTeX Companion, by M. Goosens, F. Mittelbach, and A. Samarin (Addison Wesley, 1994)
- The LaTeX Graphics Companion, by M. Goosens, S. Rahtz, and F. Mittelbach (Addison Wesley, 1997)

If you have problems running TeX, please check your environment variables, in particular `TEXINPUTS`. Make sure these do NOT contain references to '`/usr/local/share/texmf/...`', '`/usr/local/lib/texmf`', etc. If you have private macro files in '`~/tex`', then you will want to define

```
TEXINPUTS=.:~/tex: export TEXINPUTS
setenv TEXINPUTS .:~/tex:
```

in sh or csh syntax. (The default `TEXINPUTS` will be implicitly inserted after the trailing colon).

This version provides better support for producing documents for online viewing:

1. pdflatex produces a pdf file instead of a dvi file.
2. the hyperref package has support for carrying the cross-reference information and references to URLs in a TeX document over to the dvi or pdf file; for example, you reference your home page like this

```
\href{http://w3.pppl.gov/~jsmith}{J. Smith}
```

3. xdvi can follow links (cross-refs and URLs).

The following are possible ways of producing pdf files from LaTeX

1. In the '`.tex`' file, put
   ```
   \usepackage{times}
   \usepackage{hyperref}
   ```
   Process with `latex`, then with `dvips -Pwww -z`, then with `distill`.
2. In the '`.tex`' file, put
   ```
   \usepackage{times}
   \usepackage[dvips]{hyperref}
   ```
   Process with `latex`, then with `dvips -Pwww`, then with `distill`.
3. In the '`.tex`' file, put
   ```
   \usepackage{times}
   \usepackage[pdftex]{hyperref}
   ```
   Process with `pdflatex`.

When you view the '`.dvi`' file produced in method (1) with `xdvi`, the links will be underlined (and you click on them to follow them).

5.3 python

Python is an interpreted, interactive, object-oriented programming language. Version 1.5.1 is installed on the Unix Cluster (under '`/usr/local`') and on hecate (under '`/usr/pppl`').

The main documentation is python(1). The web site is `http://www.python.org`. There's a reference manual available at '`/usr/doc/python-1.5.1/Doc/ref/ref.ps`'. Our local expert is Greg Hammett.

In addition we have the LLNL Python Extensions (release 6) installed. These are describe at `http://xfiles.llnl.gov/python.htm`. In order to use these you will need to specify

```
PYTHONPATH=/usr/local/lib/python1.5/NumPy export PYTHONPATH
setenv PYTHONPATH /usr/local/lib/python1.5/NumPy
```

in sh or csh syntax. The extensions that we have installed are

- Numerical: provides a fast numerical facility for Python
- Graphics: enables use of gist(1)from Python
- PyPDB: an interface to PACT/PDB database files. See Section 5.4 [pact], page 16.
- PyHistory: time history data from Python simulations
- RNG: random number distributions for Python

5.4 pact

pact is a "portable application code toolkit". See `http://www.llnl.gov/def_sci/pact` for details. To use this at PPPL, you will need

```
PACT=/usr/local/pact export PACT
setenv PACT /usr/local/pact
```

in sh or csh syntax.

5.5 ghostview

ghostview is a viewer for PostScript and PDF files. It may be invoked as gv or ghostview. We are running version 3.5.8. Documentation is via the man page gv(1). There is also documentation available at '`/usr/doc/gv-3.5.8/gv.html`' . The main web site is `http://wwwthep.physik.uni-mainz.de/~plass/gv/`.

5.6 MATLAB

(From the documentation:)

MATLAB is a high-performance language for technical computing. It integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. Typical uses include:

- Math and computation
- Algorithm development
- Modeling, simulation, and prototyping
- Data analysis, exploration, and visualization
- Scientific and engineering graphics
- Application development, including Graphical User Interface building

To run MATLAB, type

```
matlab
```

We are running version 5.3 (also known as Release 11). Matlab is supported on all the computers in the Unix Cluster. PPPL has a single floating license for this product. However, if this license is in use, MATLAB will automatically use one of the Princeton University licenses, which should only be used for educational purposes.

Documentation is obtained from matlab(1), the command `matlabdoc` (type `matlabdoc -h` for documentation), and at `http://w3.pppl.gov/misc/matlab/helpdesk.html`. The last includes manuals in PDF format. You can see many examples of using MATLAB by typing `demo` when running the program.

The vendor's web site is `http://www.mathworks.com/products/matlab/`.

5.7 MexCDF

(From the documentation:)

What is MexCDF?

MexCDF is a mex-file interface between NetCDF and MATLAB that invokes the complete C-Language NetCDF interface, as described in the NetCDF Users Guide. This allows MATLAB users to read, write, and manipulate NetCDF data files in a general, efficient manner.

Access to m-netcdf files is obtained by typing

```
addpath /usr/local/mexcdf/netcdf
ncstartup
```

somewhere at the top of the MATLAB scripts.

Web sites are

http://crusty.er.usgs.gov/~cdenham/MexCDF/nc4ml5_guide.html
http://crusty.er.usgs.gov/mexcdf.html

5.8 Maple

Maple is a system for doing algebra. We are running Maple V Release 5 under a University Site which lets it be run on any Unix Cluster machine (as well as PC's, personal Linux machines, etc.) The are two interfaces

bullet text-only—accessed with the command `maple` (see maple(1))

bullet graphical—accessed with the command `xmaple` or `maple -x`

The main documentation for Maple V are the books published by Springer-Verlag (1991–91):

bullet Maple V, First Leaves: A Tutorial Introduction

bullet Maple V, Language Reference Manual

bullet Maple V, Library Reference Manual

In addition to the man pages, the graphical interface has a `Help` menu.

The vendor's web site is `http://www.maplesoft.com`.

5.9 Mathematica

Mathematica is a system for doing algebra. We are running version 3.0.2. PPPL has a 4-user node-locked license for taurus. It can also be used for educational purpose on other Solaris systems using one of the Princeton University licenses. There are two interfaces

bullet text-only—accessed with the command `math` (see math(1))

bullet graphical—accessed with the command `mathematica` (see mathematica(1))

The main documentation for Mathematica is the book "Mathematica, a System for Doing Mathematics by Computer", by Stephen Wolfram, Addison-Wesley (1988).

In addition to the man pages, the graphical interface has a `Help` menu.

The vendor's web site is `http://www.wolfram.com/products/mathematica/`.

5.10 TotalView

The TotalView debugger is part of a suite of software development tools for debugging, analyzing, and tuning the performance of programs, including multiprocess multithreaded programs.

As of July 5, 2001, the current release of Totalview is version 5.

The vendor's web site is `http://www.etnus.com`.

Man pages man be read by typing

```
man -M /usr/local/totalview/man totalview
man -M /usr/local/totalview/man tvdsvr
```

THE TotalView programs and its library are in

```
/usr/local/bin/totalview
/usr/local/bin/totalviewcli
/usr/local/bin/visualizer
/usr/local/lib/libdbfork.a
```

Totalview is available on the Unix Cluster Linux Alpha, Linus Intel, Digital Unix and Solaris platforms. Documentation including a user guide is in:

`file:///usr/local/totalview/doc/pdf`

`file:///usr/local/totalview/doc/html`

The Release Notes describe new features, have a list of known problems, and how to describe report new problems.

The CLI Guide describes the new Command Line Interface.

The User_Guide is a 350+ page manual describing all features.

5.11 xbp Restoring files

The Netbackup Utility backups files on the Unix Cluster. User home directories and the scratch disk directories are amongst those backed up. Netbackup currently runs on the Solaris and on the Digital Unix systems.

Man pages describing the system are bp, xbp, bpbackup, bprestore, bparchive, and bplist.

xbp is used to restore backed up files. The following files give complete documenation.

```
file:///usr/openv/netbackup/help/xbp/Tutorial

file:///usr/openv/netbackup/help/xbp/Altpath

file:///usr/openv/netbackup/help/xbp/Config

file:///usr/openv/netbackup/help/xbp/Confirm

file:///usr/openv/netbackup/help/xbp/Help

file:///usr/openv/netbackup/help/xbp/Index

file:///usr/openv/netbackup/help/xbp/Main

file:///usr/openv/netbackup/help/xbp/Progress

file:///usr/openv/netbackup/help/xbp/Viewselect
```

EXAMPLE:

To restore backed up files that were in your home directory:

```
o Go to computer where home directory resides (now orion)

  % ssh orion

o Change to the directory where the files were located.

  % cd ~/TestMDSplus/LRanderson_004

o Invoke xbp
```

```
    % xbp -r &

  In xbp

    Choose File/Browse backups (restore)
    Choose Edit/Update Display
    In files directory
      Highlight (click) files to restore -- Note archive date
    In Backup Images
      Highlight (control-click) 'Standard Class'  images to use for
      restore. Incr means incremental backup, Cinc means cumulative
      incremental backup, and Full means full backup.
    Choose Restore/Restore selected files and directories...
      Note the log filename.
      Highlist 'Overwrite existing files' if appropriate.
      Click on OK
    Select Restore/Report progress of restore...
      Enter name of Log file
      Click on Automatic Refresh
```

5.12 axiom

axiom is a system for doing algebra. We are running version 2.1. PPPL has a site license as part of the NAG Academic Site License Initiative. It can be run on any Unix Cluster machine and on Linux and Windows PCs. To run it, type `axiom` (see axiom(1)). This brings up a documenation browser. It's also documented in the book "Axiom, the Scientific Computation System", by Richard Jenks, and Robert Sutor (Springer-Verlag, 1992).

The vendor's web site is `http://www.nag.com/sumbolic/AX.html`.

5.13 Alpha Linux Fortran

The alpha linux machines are currently loki, machine3, thor, freda, and tyr. You can verify you are on a linux alpha machine by:

```
$ uname -ms
Linux alpha
```

The recommended fortran compiler is the Compaq Fortran Compiler for Linux Alpha. It is accessed via the *fort* command. Sources with *.f*, *.for*, or *.FOR* suffixes are assumed to be FORTRAN 77. Sources with *.f90* or *.F90* suffixes are assumed to be Fortran 90 source files. This fortran compiler closely resembles the Fortran used on the Digital Unix machines (i.e. the planets). One difference is that when compiling Fortran 99 sources, the switch -assume no2underscores is generally required.

Compaq Fortran documentation is in

- `file:///usr/local/doc/alphalinux/Fortran/index.html`
- `file:///usr/local/doc/alphalinux/Fortran/docs/cf/index.html`
- `file:///usr/local/doc/alphalinux/Fortran/docs/cxml/index.html`

- `file:///usr/local/doc/alphalinux/Fortran/docs/cpml/index.html`

Man pages are invoked on the alpha linux machines via:

```
% man fort
% man -S3f intro
```

Currently available libraries include blas, lapack, nag, ncar, and netcdf. These are available in directory '`/usr/local/lib`'.Documentation can be found in directory '`/usr/local/doc`'. MP, the Message Passing Interface, is available in directory '`/usr/local/mpich`'. The mpich users guide is in file '`/usr/local/mipch/doc/guide.ps.gz`'.

Hard copies of the fortran documentation are also available.

- The Fortran Language Users Guide is at `file:///usr/local/doc/alphalinux/Fortran/docs/cf/dfuum.ps`
- The Fortran Language Reference Manual is at `file:///usr/local/doc/alphalinux/Fortran/docs/cf/dflrm.ps`
- The Extended Math Library documentation is at `file:///usr/local/doc/alphalinux/Fortran/docs/cxml/cxml-ref.ps`
- The Fortran Intristic Functions documentation is at `file:///usr/local/doc/alphalinux/Fortran/docs/cpml/cpml-ref.ps`

5.14 Linux Fortran

For the results of a study of Linux Fortran compilers made here during February 2000 see: `http://w3.pppl.gov/~dmccune/papers-reports/linux_f90.txt`

Available on the Linux Intel members of the Unix Cluster are:

1. Fujitsu Fortran
2. Lahey-Fujitsu Fortran
3. Portland Group Fortran
4. Numerical Algorithms Group (NAG) Fortran
5. Message Passing interface (MPI)
6. Fortran specific libraries such as blas, lapack, nag, and netcdf.

A description follows.

1. The Fujitsu Fortran Linux compiler is invoked via:

   ```
   % setenv PATH            /usr/local/ffc/bin:$PATH
   % setenv LD_LIBRARY_PATH /usr/local/ffc/lib
   % f90 -o test_example test_example.f
   ```

   - *frt*, *f90*, and *f95* are equivalent commands used to compile
   - The Fujitsu debugger is invoked via *fdb*
   - Fujitsu Fortran specific libraries are in `/usr/local/ffc/lib/...`

   See the following for Fujitsu Fortran documentaion

   - User and Reference manuals are in `/usr/local/ffc/doc/...`
   - Man pages are in `/usr/local/ffc/man`. As an example, to obtain documentation on program *f90* or on subroutine *ifix*:

     ```
     % man -M /usr/local/ffc/man f90
     % man -M /usr/local/ffc/man ifix
     ```

     Available subroutines can be listed via:

     ```
     % ls /usr/local/ffc/man/man3
     ```

- Vendor web site is at `http://www.fujitsufortran.com`

2. The Lahey-Fujitsu Fortran Linux compiler is invoked via:

```
% setenv PATH            /usr/local/lff95/bin:$PATH
% setenv LD_LIBRARY_PATH /usr/local/lff95/lib
% lf95 -o test_example test_example.f
```

- *lf95* is the command used to compile
- The Lahey-Fujitsu debugger is invoked via *fdb*
- Lahey-Fujitsu Fortran specific libraries are in `/usr/local/lff95/lib/...`

See the following for Lahey-Fujitsu Fortran documentaion

- User and Reference manuals are in `/usr/local/lff95/doc/...`
- Man pages are in `/usr/local/lff95/man`. As an example, to obtain documentation on program *lf95* or on subroutine *ifix*:

```
% man -M /usr/local/lff95/man lf95
% man -M /usr/local/lff95/man ifix
```

Available subroutines can be listed via:

```
% ls /usr/local/lff95/man/man3
```

- Vendor web site is at `http://www.laheyonline.com`

3. The Portland Group Fortran Linux compiler is invoked via:

```
% setenv PATH            /usr/local/pgf90/bin:$PATH
% setenv LD_LIBRARY_PATH /usr/local/pgf90/lib
% f90 -o test_example test_example.f
```

- *f90* or *pgf90* are equivalent commands used to compile.
- *totalview* will invoke the TotalView debugger.
- The Portland Group F90 Fortran specific libraries are in `/usr/local/pgf90/lib`.

See the following for Portland Group Fortran documentation

- man pages at: pgf77(1), pgf90(1). pghpf(1), pgprof(1), pgdbg(1), and Xpgdbg(1).
- Vendor documenation at: `/usr/local/pgf90/doc/....`
- Vendor WEB documentation at: `http://w3.pppl.gov/misc/pgi`.
- Vendor web site at: `http://www.pgroup.com`.

4. The NAG Fortran compiler is invoked via:

```
% setenv PATH            /usr/local/nagf95/bin:$PATH
% setenv LD_LIBRARY_PATH /usr/local/nagf95/lib
% f95 -o test_example test_example.f
```

- *f77*, *f90*, and *f95* are equivalent commands used to compile.
- *dbx90* will invoke the dbx debugger.
- The NAG F95 Fortran specific libraries are in `/usr/local/nagf95/lib`.

See the following for NAG Fortran documentation

- Man pages are in `/usr/local/nagf95/man`. As an example, to obtain documentation on program *f95*:

  ```
  % man -M /usr/local/nagf95/man f95
  ```

- Vendor web site at: `http://www.nag.com`.

5. The Message Passing Interface

   MPI, the Message Passing Interface, is specific to each Fortran compiler. Depending of the compiler use either `/usr/local/ffc/mpich/...` , `/usr/local/lff95/mpich-1.2.1/...` `/usr/local/pgf90/mpich-1.2.1/...`, or `/usr/local/nagf95/mpich-1.2.1/...`

   See the following for Message Passing Interface documentation

   - A Users Guide is at '`/usr/local/ffc/mpich/doc/guide.ps.gz`'
   - Man pages are in `/usr/local/ffc/mpich/man`. As an example, to obtain documentation on program *mpif90* or on subroutine *MPI_Bcast*:

     ```
     % man -M /usr/local/ffc/mpich/man mpif90
     % man -M /usr/local/ffc/mpich/man MPI_Bcast
     ```

     Names of available MPI man pages can be listed via:

     ```
     % man -M /usr/local/ffc/mpicn/man -k MPI
     ```

   - WEB documentation is at

     `http://www-unix.mcs.anl.gov/mpi/index.html`

     `http://WWW.ERC.MsState.Edu/labs/hpcl/projects/mpi/`

6. Fortran specific libraries

   Fortran libraries specific to each Fortran compiler are in `/usr/local/ffc/lib`, `/usr/local/lff95/lib`, `/usr/local/pgf90/lib`, and `/usr/local/nagf95/lib` .

   Current available libraries include blas, lapack, nag, and netcdf.

5.15 AVS

AVS Express is a high-end visualization systems. We are running version 4.2. It runs on Solaris and OSF/1 platforms. We are licensed for 6 concurrent users and for 1 user of the "developer" version. To use it, type

```
express
express -nohw
express -usage
```

The first form is for display supporting 3-d graphics hardware. The second form is for other X displays. The final form types out a usage message. Extensive documentation is available via the `help` menu while running AVS express. There are also printed manuals available.

Our local expert is Scott Klasky. To use the developers edition, type

```
env XP_FEATURE=EXPRESS express
```

Please coordinate this use with Scott.

AVS is an older product whose use is discouraged. We are running version 5.4. We have a one user license that can be used on Solaris and OSF/1 machines. Documentation is in the manpage avs(1).

5.16 NCAR Graphics Library

The NCAR Graphics Library is a Fortran and C callable Graphics library. PPPL has a site license which provides the library on all Unix Cluster machines (Sun, Digital, Linux) and on hecate.princeton.edu (the Origin 2000 machine on campus). We are running version 4.1.1. The Linux version assumes the use of g77. This will be changed to use the Portland Group Compiler.

Before using NCAR Graphics you need to define

```
NCARG_ROOT=/usr/local export TEXINPUTS
setenv NCARG_ROOT=/usr/local
```

in sh or csh syntax.

On hecate replace '`/usr/local`' with '`/usr/pppl`'. On hecate, you will also need '`$NCARG_ROOT/bin`' in your '`PATH`' and '`$NCARG_ROOT/man`' in your '`MANPATH`'.

Online documentation is available via the WWW at `http://w3.pppl.gov/ngdoc/ng/ngdochome.html` (also at `http://ngwww.ucar.edu/ngdoc/ng/ngdochome.html`).

The vendor's web site is `http://ngwww.ucar.edu/index.html`

There are also manpages in $NCARG_ROOT/man. Start with ncargintro(5).

Postscript versions of the documentation are available at `http://w3.pppl.gov/ngdoc/ng/nggenrl/psdocs` or in the directory `file:/c/usr/local/doc/ncarg`.

The sources for this version of NCAR Graphics are in the directory `file:/c/usr/sources/ncarg`.

To compile and link, use ncargf77(1)and ncargcc(1)in place of f77(1)and cc(1). To view the "CGM" graphics files produced by NCAR Graphics use ctrans(1)or ictrans(1). CGM files can be printed directly with the `lpr` command on the PPPL Unix Cluster (`ctrans` will be invoked by the spooler).

Examples can be run with ncargex(1)and ng4ex(1), e.g.,

```
cd /tmp
ncargex wmex14
ng4ex cn12f
```

5.17 Swing

swing is part of the java development kit. Vendor's web site is `http://java.sun.com/products/jfc` and `http://java.sun.com/products/jfc/swingdoc-api-1.1.1/`

The swing area on the unix cluster is '`/usr/local/swing-1.1.1`'. For further information, see '`/usr/local/swing-1.1.1/README.txt`'.

5.18 Java

Java is available on the Digital Unix, Solaris, and Linux members of the unix cluster.

On COMPAQ Tru64 UNIX (Digital UNIX)

```
% /bin/java -version
java version "1.1.8-1"

% /bin/man java
```

WWW page is: http://www.digital.com/java/

On Solaris

```
% /bin/java -version
java version "1.1.7"

% /bin/man java
```

WWW page is: http://java.sun.com/products/jdk/1.2/docs/index.html

On Linux

```
% /usr/local/jdk1.1.7/bin/java -version
java version "1.1.7B"

% /bin/cat /usr/local/jdk1.1.7/README.linux
```

WWW page is: http://www.blackdown.org/java-linux/docs.html

5.19 StarOffice

StarOffice 5.1 is a desktop office product.

A network installtion of StarOffice 5.1 is availabe for User Installation on Solaris and Linux members of the Unix Cluster.

Web sites are

http://www.sun.com/products/staroffice/overview.html http://www.sun.com/products/staroffice/faq.html

An installation manual is available in /usr/local/doc/staroffice51/setup.pdf

An example user installation is:

```
On TAURUS
$ /bin/mkdir /u/<username>/sun5
$ cd  /usr/local/staroffice51/bin
$ ./setup

$ cd /u/<username>/sun5/Office51/bin
$ ./soffice
```

5.20 grace

Grace is a WYSIWYG tool to make two-dimensional plots of scientific data. We are running version 5.0.4, which is install on all Unix Cluster machines. The main WEB site for grace is http://plasma-gate.weizmann.ac.il/Grace/.

Documenation is via the manpages: grace(1), xmgrace(1), gracebat(1), and grconvert(1).

Other information is in (these are also available under the Help menu in xmgrace):

- Tutorial (hard copy version).
- Users' Guide (hard copy version).
- FAQ (hard copy version).

Examples are available in the under Help menu in xmgrace.

5.21 pgplot

Pgplot is a Graphics Subroutine Library. We are running version 5.2 which is installed on Solaris and OSF/1 machines in the Unix Cluster. The main WEB site for pgplot is `http://astro.caltech.edu/~tjp/pgplot/`. The users' manual is `http://astro.caltech.edu/~tjp/pgplot/contents.html`

To use pgplot, set

```
PGPLOT_DIR=/usr/local/pgplot export PGPLOT_DIR
PGPLOT_DEV=/xwin export PGPLOT_DEV
```

or

```
setenv PGPLOT_DIR /usr/local/pgplot
setenv PGPLOT_DEV /xwin
```

depending on whether your shell understands sh or csh syntax. Setting PGPLOT_DEV is optional—it just provides the default device type.

C programs will need the header file

```
#include <cpgplot.h>
```

and will need to be compiled with

```
-I$PGPLOT_DIR
```

To link Fortran and C programs, use

```
-L$PGPLOT_DIR -lpgplot -lX11
-L$PGPLOT_DIR -lcpgplot -lX11
```

respectively.

A summary of routines is available in `http://w3.pppl.gov/misc/pgplot/pgplot.html` (also available as a text version and as a hardcopy version).

6 Printing

All Unix Cluster hosts are now running software that has some nice features:

1. The printing software auto-recognizes many different file types:

   ```
   text
   TeX dvi (no need for the -d flag)
   PostScript, PDF
   gif, jpg, ppm
   compressed (.gz or .Z)
   man pages (e.g., lpr /usr/man/man1/csh.1)
   ```

   (However, printing man pages with

   ```
   /usr/local/bin/man -t csh | lpr
   ```

   is probably better.)

2. You can enable various features with

   ```
   lpr -C<feature1>,<feature2>
   ```

   Possible features are:

   ```
   duplex -- two-sided printing
   twoup -- print two pages/side
   tumble -- bind on short edge (implies duplex)
   tray0 or upper -- tray selection
   tray1
   tray2 or lower
   manual -- manual feed
   high -- high resolution printing (e.g., for Color Epson printers)
   ```

   You can divide your consumption of paper by four with

   ```
   lpr -Ptheoryw -Ctumble,twoup -h FILE
   ```

3. You can submit a job with lpr on machine A and remove it with lprm on machine B.

4. You can print to printers on other systems with

   ```
   setenv PRINTER queue@remote-host
   ```

   or

   ```
   lpr -Pqueue@remote-host
   ```

   The system administrator on remote-host may have to grant you permission to do this. (However, there's no need to set up a queue on our machines.)

Concept Index

I

Introduction 1

Table of Contents

1 Introduction . . . 1
- 1.1 Goals . . . 1
- 1.2 Members . . . 1
- 1.3 Accounts . . . 2
- 1.4 Passwords . . . 3
- 1.5 E-mail . . . 3
  - 1.5.1 Forwarding . . . 3
  - 1.5.2 Aliases . . . 4
- 1.6 Getting Help . . . 4
  - 1.6.1 man pages . . . 4
  - 1.6.2 info . . . 4
  - 1.6.3 Documentation on the web . . . 4
  - 1.6.4 Other documentation . . . 5
  - 1.6.5 Human Help . . . 5

2 File systems . . . 7
- 2.1 Home directories . . . 7
- 2.2 Other writable directories . . . 7
  - 2.2.1 scratch . . . 7
  - 2.2.2 tmp . . . 7
  - 2.2.3 work . . . 7
  - 2.2.4 VMS . . . 7
- 2.3 AFS . . . 7

3 Networking . . . 9
- 3.1 Logging into another computer . . . 9
  - 3.1.1 telnet . . . 9
  - 3.1.2 rsh and rlogin . . . 9
  - 3.1.3 slogin and ssh . . . 9
    - 3.1.3.1 ssh with RSA authentication . . . 10
    - 3.1.3.2 ssh examples . . . 11
    - 3.1.3.3 Installing ssh on other machines . . . 11
- 3.2 Accessing files on another computer . . . 11
  - 3.2.1 ftp . . . 11
  - 3.2.2 rcp and scp . . . 11
  - 3.2.3 rsync . . . 11

4 X windows . . . 13
- 4.1 xrsh and xssh . . . 13

5 Miscellaneous software 15
5.1 emacs 15
5.2 TeX 15
5.3 python 16
5.4 pact 16
5.5 ghostview 17
5.6 MATLAB 17
5.7 MexCDF 17
5.8 Maple 18
5.9 Mathematica 18
5.10 TotalView 18
5.11 xbp Restoring files 19
5.12 axiom 20
5.13 Alpha Linux Fortran 20
5.14 Linux Fortran 21
5.15 AVS 23
5.16 NCAR Graphics Library 24
5.17 Swing 24
5.18 Java 25
5.19 StarOffice 25
5.20 grace 25
5.21 pgplot 26

6 Printing 27

Concept Index 29