Guide to Eos and Unity Computing

Dr. Ellen McDaniel
College of Engineering

INFORMATION TECHNOLOGY AND ENGINEERING COMPUTER SERVICES
COLLEGE OF ENGINEERING
NORTH CAROLINA STATE UNIVERSITY
RALEIGH, NC 27695-7901
The author gratefully acknowledges the manufacturers of the software listed in Appendix C and the authors of the World Wide Web sites referenced in this book, particularly those of the NCSU Information Technology Division.
Only connect . . .

E.M. Forster, *Howards End*

“No matter where you go, there you are!”

*Buckaroo Banzai*
ACKNOWLEDGMENTS

Guide to Eos and Unity Computing, 2003-04 Edition for Unix, Windows, and Linux is the principal user manual for the distributed academic computing environment at North Carolina State University. Formerly a College of Engineering system only, “Eos” was made accessible to other NCSU colleges in a project called “Unity.” In 1994, all NCSU students received accounts on what is now a fully merged campus-wide system called “Eos/Unity.”


Special thanks go to William E. Willis, former Vice Provost for Information Technology, Thomas K. Miller, current Vice Provost for Distance Education and Learning Technology Applications, and the NCSU College of Engineering (COE) Computer Committee for their vision in building the original Eos system. Tom Miller and Sam Averitt, Vice Provost for Information Technology, also have my thanks for their ongoing support of this user guide, along with William G. Scott and Charles Hunt, Director and Assistant Director of COE Information Technology and Engineering Computer Services (ITECS). Support and information from the staffs of ITECS and NCSU’s Information Technology Division (ITD), plus the COE Computer Committee, helped to make this manual more complete and comprehensive. I can no longer name all the people who helped; the list has grown too long over the years. Troy Hurteau has been my close partner in putting this guide online, and his Web talents are an inspiration. Thanks also to Tim Lowman for the original Hitchhiker’s Guide to the Eos System, to Marshall Brain for the first online Eos tutorials, and to the people who maintain the E115 Lab Manual. These guides provided the first information on the system, and their authors’ contribution to the education of our users is gratefully acknowledged.

The cover for this book is the work of Lou Harrison. I am lucky to have his permission to use the compelling image, Thru the Wall at http://www.csc.ncsu.edu/harrison/Artistic.html. I would also like to acknowledge the print sources I consulted in the writing of this manual:

Welcome to Athena, Working on Athena, Managing Your Athena Account, etc., http://web.mit.edu/olh/
# Table of Contents

## Getting Started

The Eos/Unity Academic Computing Environment ......................................... 1  
Legacy System and Unix/AFS Infrastructure ............................................. 1  
The Windows 2000 Platform ................................................................. 2  
Eos Linux: NCSU Realm Kit for Red Hat Linux ......................................... 3  
Student-Owned Computers ................................................................. 4  

## Account Eligibility and Privileges ....................................................... 5  
Account Identification ........................................................................... 5  
Account Privileges ................................................................................ 6  
Disabling and Deleting Accounts .......................................................... 6  
Summer Accounts .................................................................................. 7  
Account Names: Eos or Unity? ............................................................... 7  

## Computer Use Policies ........................................................................ 8  
Regulatory Limitations ........................................................................... 8  
Personal Use ......................................................................................... 9  
Use of Computing Facilities for Commercial, Advertising, and Broadcast Purposes ........................................................................................................... 10  
Violation of Policy ................................................................................ 10  
Application of Public Records Law .......................................................... 11  
Additional Rules ................................................................................... 11  
Computer and Network Use: Remote Access and Labs ............................. 12  
Unauthorized Copying or Use of Computer Software .................................. 13  
NCSU Guidelines and Procedures for Determining Ownership of Computer Software ................................................................. 13  
ResNet Terms and Conditions ................................................................. 13  
Policies in Eos Public Labs .................................................................... 13  

## Labs ..................................................................................................... 14  

## Workstation Basics ............................................................................ 17  
The Monitor and System Unit ................................................................. 18  
The Mouse ............................................................................................. 18  
The Keyboard ....................................................................................... 19  

## Logging In .......................................................................................... 20  

## The User Interface ............................................................................. 24  
The Solaris FVWM Interface .................................................................... 24  
Command Line Interpreter ..................................................................... 25  
Graphical User Interface ....................................................................... 25  
The Windows 2000 Interface ................................................................. 26  
The Linux GNOME Interface ................................................................. 27  

Interface to What? ................................................................. 28

Menus and Keyboard Shortcuts ............................................. 30
Window Control Menu .......................................................... 30
The Unix Root and Application Menus .................................... 31
The Start Menu on Windows .................................................. 32
Linux/GNOME Program Menus ............................................. 32
Keyboard Control ............................................................... 32

Windows ................................................................. 34
X, fwm2, MS Windows, and GNOME ....................................... 34
Selecting a Window ........................................................... 34
Grabbing and Moving a Window ........................................... 34
Sizing a Window .............................................................. 35
Iconifying (Minimizing) a Window ......................................... 35
Maximizing a Window ........................................................ 36
Restoring a Window .......................................................... 36
Scrolling a Window ........................................................... 36
Shuffling Windows ............................................................ 37
Closing a Window .............................................................. 37

Password Changes .......................................................... 38

Getting Help (help) ............................................................ 40

Log Out and Shut Down ...................................................... 42
Windows Shut-Down Procedure ........................................... 42
Solaris Shut-Down Procedure ............................................... 43
Linux Shut-Down Procedure ............................................... 43

The Eos/Unity Network and Software

Overview of Eos/Unity OSes, File System and Services .............. 46
Operating Systems ............................................................ 46
System Services .............................................................. 47
Andrew File System (AFS) .................................................. 47
Activation of System Software at Login ................................... 48

Application Software ....................................................... 50
Solaris (Unix) Application Menu ........................................... 50
Novell Application Launcher ............................................... 55
Software Documentation .................................................... 58
Commercial and Non-Commercial Software ............................ 58
Tables of Application Software ........................................... 58
Table 1: Licensed Commercial Application Software on Windows .......... 59
Table 2: Licensed Commercial Application Software on Solaris (Unix, *Linux) .... 60
Table 3: Selected Non-Commercial Software on Solaris and Linux ............ 61
Running Applications ........................................... 62
Launching Applications on Solaris (add) ....................... 62
Background and Foreground Processes (bg, fg) .................. 62
Launching Applications from Novell and Realm Kit Launchers . . . . 63

Network Overview .................................................. 64

Working with Files and Directories

Files and Directories ............................................. 70
Directories .................................................................. 70
Lockers ...................................................................... 71
File and Directory Naming .......................................... 71

Working Directories and Paths ...................................... 72

Working with Directories and Folders ............................. 76
Unix Directory Commands ......................................... 76
Make Directory (mkdir) ............................................. 76
Change Directory (cd) ............................................... 76
List Files in a Directory (ls) ........................................ 77
Remove Directory (rmdir) .......................................... 77
Attach and Add Directory (attach, add) ......................... 77
Specifying Directories .............................................. 79
Windows Directory Manipulation .................................. 79
Make Folder ........................................................... 79
Manipulating Folders ............................................... 80
Finding Folders with Windows Explorer ......................... 80

Unix Command Options and Arguments .......................... 81

Manipulating Files .................................................. 82
Unix File Commands ................................................ 82
Display File Contents (more, less) ................................. 82
Concatenate Files (cat) ............................................. 82
Copy Files (cp) ........................................................ 83
Move and Rename Files (mv) ....................................... 83
Remove File (rm) ...................................................... 83
Windows File Manipulation ......................................... 84
Display File Contents ................................................ 84
Copy and Paste Files ................................................ 84
Cut and Move Files .................................................. 85
Rename Files .......................................................... 85
Remove/Delete Files ................................................ 85
File Properties ......................................................... 85
Linux File Manipulation ............................................. 85

Shortcuts and Operators ............................................ 86
Copying and Pasting between Windows ......................... 86
Completing Command Entry without Typing .................................................. 86
Repeating Commands without Typing .......................................................... 86
Aliases (alias) ............................................................................................... 87
Useful Key Combinations ............................................................................. 87
Wildcards and Metacharacters ...................................................................... 88
Redirection of Input and Output ..................................................................... 88
Pipes and Pipelines ....................................................................................... 89

**Unix Man(ual) Pages** (man) ................................................................. 90

**AFS File-Sharing Commands** ............................................................. 91
The Root Directory and /afs Cells .............................................................. 91
Partition, Volume, Quota, and Mount Point ............................................... 92
Sharing Information (ls) ............................................................................ 92
Sharing Information with Groups (pts) ....................................................... 95
AFS on Windows ....................................................................................... 97
Quota ......................................................................................................... 97
Access Control Lists .................................................................................. 98
AFS from the Command Prompt ............................................................... 98
AFS Glossary ............................................................................................. 99

**Selected Applications and Services** .................................................. 102

**Printing** .............................................................................................. 102
Printing Facilities ..................................................................................... 102
Print Quota (ipquota) ............................................................................... 102
Printing Files (lpr) .................................................................................... 103
Previewing Files Before Printing ............................................................. 103
Printer Selection ....................................................................................... 103
Print Queue and Removing Jobs (lq, lprm) ................................................ 104
Using the Network X Print Manager (nxpr) .............................................. 105
Printing Selected Text from the Root Menu ............................................ 105
Color Printing ........................................................................................... 105
Printing Screens ....................................................................................... 106
Advice for Conserving and Sharing Print Resources ................................... 106

**Electronic Mail** .................................................................................. 107
IMAP Mail Access and Programs ............................................................... 107
Note to POP Users .................................................................................... 108
Preferred Mail Addresses .......................................................................... 108
Forwarding Mail ....................................................................................... 109
Using Mail ................................................................................................ 109
Outgoing Mail Relay Change for Commercial ISP Users .......................... 110
Viruses and Spam ..................................................................................... 110

**WolfWare Course Lockers** ................................................................. 111
What is WolfWare? .................................................................................... 111
Using WolfWare ....................................................................................... 112
Message Board .......................................................................................... 113
World Wide Web ................................................................. 116
The Web Browser ............................................................ 116
Publishing Your Web Pages ............................................... 119
Responsibility for Web Pages ............................................. 121
Creating HTML Documents ............................................ 121

SSH and PuTTY ................................................................. 124
Telnet ............................................................................. 124
Secure Shell SSH ............................................................. 124
SSH on Windows via PuTTY ............................................. 124

File Transfer Protocol (ftp) ................................................ 126
Ftp-ing to/from Your Eos/Unity Account ......................... 126
Ftp-ing via a Browser ...................................................... 127
Moving from ftp to Secure File Transfer ......................... 127

ResNet and Remote Access .............................................. 128
NC State ResNet ............................................................ 128
Remote Access to Eos/Unity ............................................. 129
Dialup Services .............................................................. 130
Service Providers ............................................................ 130
Types of Dialup Connections and Software .................... 130
What Can I Do on Dialup? ............................................... 131
What Can’t I Do on Dialup? ............................................ 131
Reading Mail from Dialup Machines ................................ 132
Compiling and Running Programs on Dialup .................. 132
Editing Files on Dialup .................................................... 132
Communicating Directly with Other Users ....................... 133
Principal NCSU Server Addresses .................................. 133
Eos Remote-Access Services .......................................... 134

Unix-only Applications and Customization

Zephyr Communication ................................................... 136
Locating a User (zlocate) ............................................... 136
Writing a Zephyr Message (zwrite) ................................. 137
Changing the Format of a Zephyrgram............................ 138
Copying and Pasting .................................................... 140
Subscribing to Other Messages (zctl) .............................. 140
The Help Instance ......................................................... 141
Locating Groups of Users (znl) ....................................... 141
Blocking Messages (zaway, zctl hide) ............................. 142

Common Text Editors .................................................... 143
NEdit ........................................................................ 143
Visual SlickEdit Programmer’s Editor ................................................................. 146
vi ......................................................................................................................... 147
Pico ..................................................................................................................... 148
asWedit HTML Editor ......................................................................................... 148
GNU Emacs ......................................................................................................... 149

XTRACS ............................................................................................................. 150

Corel WordPerfect ............................................................................................ 152
Starting WordPerfect ......................................................................................... 152
Creating a Document ......................................................................................... 152
Using the Pull-down Menus and Power Bar ....................................................... 153
Saving Your Document ...................................................................................... 154
Saving WordPerfect Files to a Floppy Disk ...................................................... 155
Saving in HTML with Internet Publisher ......................................................... 155
ExpressDocs Templates ..................................................................................... 156
Printing Your Document .................................................................................... 156
Tools ................................................................................................................... 157
Getting Help ....................................................................................................... 157
Quitting WordPerfect ....................................................................................... 157

Backing up and Restoring Files ........................................................................ 158
Backing Up Using MS-DOS Formatted Disks .................................................... 158
Formatting a Floppy Disk (fdformat, mformat) .................................................. 158
Writing Data to and from a Floppy Disk (mdir, mcopy) ...................................... 159
Man Pages on Mtools ....................................................................................... 160
Problems and Error Messages ......................................................................... 160
Restoring Lost Files (from NC State Help Database document 1332) ............... 161
Restoring Your Own Lost Files (fs examine, fs whichcell) ............................... 161
Requesting a Restore ......................................................................................... 162

The C Shell and Environment Variables ......................................................... 164
Environment Variables (printenv, setenv) ........................................................ 165
Shell Variables (set) .......................................................................................... 165

Customizing .................................................................................................... 166
Configuration Files ............................................................................................ 166
Levels of Session Customization ...................................................................... 167
Writing Configuration Files .............................................................................. 168
The .myshrc File ............................................................................................... 169
The .Xdefaults File ............................................................................................ 170
The .mylogin File .............................................................................................. 171
The .Xstartup File ............................................................................................. 172
.fwm2rc Window Manager .............................................................................. 174
Debugging and Fixing Configuration Files ...................................................... 175

Session Configuration ..................................................................................... 176
The X Windows Session Configuration ............................................................ 178
The TTY Session Configuration ....................................................................... 180
Managing Your Account .................................................. 182
Regular Checks ......................................................... 182
Managing Your Disk Quota ............................................. 182
Managing Your Print Quota ............................................ 183
Changing Your Password ................................................ 183
Protecting Your Account .............................................. 183
Sharing Information but not Your Password ....................... 183
Using Care with add and attach ...................................... 184
Your Personal Web Page ............................................... 185
Finding Other Users .................................................... 185

Appendix A: Trouble Shooting ......................................... 187

Appendix B: Summary of Commands ................................. 191
Unix and Local Commands ............................................. 192
Directory Commands ................................................... 193
Printing Commands ..................................................... 193
Zephyr Commands ....................................................... 193
User Location Commands .............................................. 193
AFS Commands .......................................................... 194
Vi Commands, distributed by UnixWorld ......................... 194
Vi Commands, Compiled by Tim Lowman ......................... 195

Appendix C: Licensed Commercial Software on Eos/Unity .......... 197
ADAMS Mechanical System Simulation
  Mechanical Dynamics, Inc. ......................................... 198
Adobe Acrobat, Illustrator, Photoshop, and Reader
  Adobe Systems, Inc. ................................................ 198
AMPL/CPLEX Optimization Modeling
  ILOG, Inc. ............................................................ 198
ANSYS Finite Element Analysis
  ANSYS, Inc. ........................................................ 199
ARC/INFO Geographic Information System Toolkit
  Environmental Systems Research Institute (ESRI) .............. 199
ArcView Geographic Information System
  Environmental Systems Research Institute (ESRI) .............. 199
ARENA Simulation System
  Rockwell Software, Inc. .......................................... 200
Aspen Plus Chemical Process Simulation
  Aspen Technology, Inc. ............................................. 200
AutoCAD Computer-Aided Design
  Autodesk, Inc. ..................................................... 200
AVR Studio
  Atmel, Inc. ........................................................ 201
Cadence Circuit Design Toolkit
  Cadence Design Systems, Inc. .................................... 201
CADRA Computer-Aided Design
  SofTech Systems, Inc. ............................................. 201
Dreamweaver Visual HTML Editor
  Macromedia, Inc. ................................................ 201
FactoryCAD Factory Layout Drafting
  EDS, Inc. .......................................................... 202
Fireworks Web Graphics
  Macromedia, Inc. ................................................ 202
Flash Web Animation
Macromedia, Inc. .......................................................... 202
FORTRAN 90 and 95 NAGware Compilers
Numerical Algorithms Group (NAG) ................................ 203
FrameMaker Word Processor and Publisher
Adobe Systems, Inc.......................................................... 203
HSPICE Electronic Circuit Simulator
Synopsys Corporation ....................................................... 203
Java Development Kit (JDK)
Sun Microsystems, Inc........................................................ 203
JMP Statistical Discovery Software
SAS Institute, Inc............................................................ 204
LINDO/LINGO Optimization
LINDO Systems, Inc........................................................ 204
LSF Load-Sharing Facility
Platform Computing, Inc.................................................... 204
Maple Symbolic Math Computation
Waterloo Maple Software .................................................. 205
Mathcad Calculation and Documentation
Mathsofts, Inc................................................................. 205
MATLAB Numerical Matrix Computation
The MathWorks, Inc........................................................ 205
Moldflow Plastics Injection Molding
Moldflow Corporation ....................................................... 206
NEeX Scientific and Engineering Spreadsheet
GreyTrout, Inc................................................................. 206
Norton AntiVirus Corporate Edition
Symantec Corporation ...................................................... 206
Office
Microsoft Corporation ..................................................... 206
OPNET Network Management and Modeling
OPNET Technologies Incorporated ...................................... 207
Primavera Project Planner
Primavera Systems, Inc..................................................... 207
Pro/ENGINEER and Pro/MECHANICA
Parametric Technology Corporation .................................... 207
SAS Data Analysis Applications
SAS Institute, Inc.......................................................... 207
Silvaco Integrated Circuit Design
Silvaco International ..................................................... 208
SolidWorks Computer-Aided Design
SolidWorks Corporation ................................................... 208
S-Plus Advanced Data Analysis
Insightful, Inc............................................................... 208
StarOffice Office Application Suite
Sun Microsystems, Inc...................................................... 208
Synopsys Digital Circuit Synthesis
Synopsys, Inc............................................................... 209
Tecline Interactive Plotting
Amtec Engineering, Inc.................................................. 209
TK Solver Plus Simultaneous Equation Solver
Universal Technical Systems, Inc....................................... 209
Together Control Center
TogetherSoft Corporation ................................................ 209
Visio Professional
Microsoft Corporation ..................................................... 210
VisualAge Smalltalk
IBM, Inc................................................................. 210
GETTING STARTED
North Carolina State University (NCSU) offers its students and faculty one of the best and largest distributed academic computing networks in the world. More than a decade of planning, development, expansion, and use has gone into delivering this massive and versatile product to campus. 1994 was the first year that all students and faculty received an account on Eos/Unity, which is now well established as a campus-wide system.

The system has notable underpinnings. In 1989, the NCSU College of Engineering (COE) Computer Committee, led by William E. Willis and Thomas K. Miller, traveled to MIT to investigate and assess the innovative Athena Project. Begun in 1983, this project was a decade-long effort by MIT, IBM, and Digital Equipment Corporation to develop a large-scale academic network with specifically designed software for educational and research computing.

Impressed by the power, security, scalability, and all-round functionality of the system for a large user community, the committee set about importing the Athena technology to NCSU. Since its first year of operation in 1990, students, staff, and faculty have been adapting and extending the system, taking it well beyond its Athena origins to what it is today: a large-scale, multi-platform network for 40,000+ users.

Legacy System and Unix/AFS Infrastructure

Named Eos after the Greek goddess of dawn by students in the former Computers and Technologies Theme dorm, the College of Engineering created a student computing environment for engineering (eos.ncsu.edu) that gradually grew into a campus-wide system (unity.ncsu.edu). With the addition of Unity, all NCSU students and faculty received accounts on the system, which came to be called Eos/Unity.

Unfortunately, this compound name is awkward and inaccurately implies that there are two systems rather than one. In fact, there has been only one system campus-wide since 1995, and the name is preserved simply because renaming so many files, computers, addresses, and accounts would be more disruptive than helpful.

The software that is primarily responsible for creating the common environment and uniting all the cells (eos.ncsu.edu, unity.ncsu.edu, and...
bp.ncsu.edu, or “backbone protocol”) is the Andrew File System (AFS), an implementation of Carnegie Mellon University’s Project Andrew (first owned by Transarc Corporation and now by IBM and OpenAFS). This software creates the shared file space that users move around in. The authentication system used to check that users are who they say they are is Kerberos, developed in MIT’s Project Athena.

Unix, specifically Sun Solaris, is the principal supported operating system. Eos/Unity has always run Unix in a client-server distributed architecture. Running on top of Solaris is the X Windows System, developed in the Athena Project, which uses the X client-server protocol.

Users can log in to the system from any Eos/Unity workstation and be admitted to a secure but common graphical and multitasking environment. Users can customize this environment as they like, and it is downloaded whenever they log in to any machine on the network, making popular the line from the movie, Buckaroo Banzai, “No matter where you go, there you are.”

The 75+ commercial applications on Eos/Unity include more than 30 for Sun Solaris (see Appendix C). Software is regularly upgraded to keep it current. Because students will use these programs again in the jobs and professions they enter after college, an introduction to this software at NCSU is a valuable addition to their education. Moreover, most of these packages are too big and expensive to acquire by any other means than a shared networked system. For an individual to buy, store, and maintain such a suite of software would be practically impossible.

In order to make such a suite of software available to all students, NCSU has installed workstation laboratories all across campus. Like software, the computing hardware is regularly upgraded. Each summer, hundreds of workstations are replaced in the public labs. Realm hardware and software are paid for out of the student Education and Technology (ETF) fee, with supplemental support from departments and colleges.

**The Windows 2000 Platform**

While Unix and its application software have supplied campus users with much of the functionality they have needed, this software has not been all things to all people. Unix and AFS have given the campus a secure and scalable infrastructure, but they have not been able to provide the many popular Windows-only applications that users have wanted. As a result, campus Information Technology Division (ITD) developed a non-Unix client for Eos/Unity built on Microsoft Windows.

In March 1998, ITD put 71 Windows NT workstations in the newly refurbished Language and Computer Labs, AKA “the Laundry.” ITD led the effort to develop the NT platform into a reliable and full-functioning client for campus, subsequently developing the GINA (Graphical...
Identification and Authentication) for a single Kerberized login and integrating it with NT AFS, Netware Directory Services (NDS), and the Novell Application Launcher (NAL) in fall 1999. The platform ran AFS as the file system and used the user’s home directory for data storage rather than the PC’s hard drive.

In 2000, ITD integrated Windows NT into the existing Eos/Unity and Novell Directory Services (NDS) infrastructures and stopped using the GINA. With the implementation of Windows 2000, users log in through NDS and KAUTH (Kerberos and Transarc AFS) to get access to their Eos/Unity AFS space.

This year, the College of Engineering released Wolfcall, a downloadable Kerberos for Windows + OpenAFS, used in the Eos labs and by users at home and in offices to connect to AFS. Users are encouraged to store their files in their AFS space rather than on the local hard drive because AFS space is backed up nightly. In labs, it is not possible to store on C:, the hard drive, so users must save to their K: and J: drives.

Each user has a roaming profile of individual preferences and application settings that is stored in the user’s NetWare profile space on the M: drive. Applications are delivered to the workstation using Novell’s ZENWorks and the Novel Application Launcher.

In simplest terms, MS Windows creates a familiar user front end with abundant applications, and Unix and AFS provide the back-end infrastructure harnessed to a large-scale campus network. User files, software, and customizations download to Windows clients similarly to the way they download to Unix workstations. An impressive suite of application software is available for this platform, see Appendix C and http://www.eos.ncsu.edu/software/.

**Eos Linux: NCSU Realm Kit for Red Hat Linux**

In 1999, Linux was added to the mix with an “Eos-ized” Red Hat Linux 5.2, thanks to support from Red Hat Software, Inc., and the Office of Information Technology and Engineering Computer Services (ITECS) in the NCSU College of Engineering. Developed by Textile Engineering Professor, Warren Jasper, and student, Matthew Wilson, an Eos Linux CD and installation manual were distributed in summer 1999.

Development continued in ITECS with the fall 2000 launch of Realm Kit for Red Hat Linux, based on Red Hat 6.2. This distribution had several campus add-ons in one install package (no CD, network install only). The Realm Kit was used principally in the College of Engineering and installed on all Dell lab computers in Leazar and Withers (more than 175 machines in all). These labs are the main labs for E115, CSC114, CSC112, and CSC116 classes.

The Realm Kit has moved through versions 7.1, 7.2, and 7.3. In 2002, development was picked up by the College of PAMS, which this year,
released the Realm Kit for Red Hat 9.0 with enhanced security and bug fixes. Users have access to their AFS file space, just as they do on the other platforms, and to such popular utilities as Zephyr (an Athena application for instant messaging). More information on the NCSU Linux Users Group (LUG) and the Eos Realm Kit for Red Hat Linux is at http://www.linux.ncsu.edu/.

**Student-Owned Computers**

NCSU has had a long-term commitment to providing facilities that meet all of the computing needs of its students. Because of its extensive campus resources, **NCSU does not require students to own computers.** However, surveys of incoming freshmen show that nearly all of them do (96%), and these student-owned computers provide additional benefits.

Students with their own computers are able to run software that is bundled with textbooks or which their instructors may distribute or require. They have 24/7 access to mail, Web, and network-based resources that their classes rely on. In addition, they have the advantage of being able to work at home rather than in the labs and install software of their choosing, which is not permitted on university-owned computers. If their computers are laptops, they can use them in classes and the library and take them home with them over holidays. For these reasons, the College of Engineering recommends that its students own or have access to a personal computer. Other colleges have their own guidelines, as does NCSU (see URLs in margin).

Students who purchase computers can consult *Do I need a computer at NC State?* for recommended specifications. Also, ResNet helps students in the residence halls take advantage of the computers they bring by offering network access and support.

The College of Engineering is developing its student-owned platform to better interface with the Eos/Unity network and environment so it can be used effectively in the classroom. COE established a pilot laptop program with freshman engineering students to work toward these goals. The program enters its third year in Fall 2003.

In conjunction with this program, COE has worked closely with IBM and Dell to obtain special pricing on laptop computers. These vendors offer special pricing for all NCSU students, faculty and staff, as well as for NCSU institutional purchases. Specifications and pricing for Dell and IBM laptop computers are available at http://www.eos.ncsu.edu/soc/.

North Carolina State University welcomes you to the powerful and versatile Eos/Unity computing environment. Any suggestions for the improvement of this manual may be sent to mcdaniel@ncsu.edu.
ACCOUNT ELIGIBILITY AND PRIVILEGES

Over 50,000 students and faculty are in the Eos/Unity computing realm. They access it through hundreds of workstations in public labs and many smaller clusters and offices on campus. Students pay for their access through the Education and Technology fee, which was put in place by the Office of the Provost. All registered students, faculty, and staff automatically get accounts on the system.

Account Identification

At one time, Eos and Unity accounts were handled differently and given different privileges. Now, the accounts have been merged into one realm and are virtually the same. The only differences that remain are: (1) a different location for engineering faculty, staff, and student accounts created before 1996, in /afs/eos/users/ rather than in /afs/unity/users/; (2) restricted access to equipment, which makes a difference where users can log in; and (3) special engineering applications in Eos Windows labs.

Since Fall 1996, accounts for engineering students, faculty and staff have been created in Unity file space. As for equipment access, engineering students pay an additional technology fee, which permits them to use the COE-only Eos labs in the engineering buildings. In addition, Eos Windows labs provide a number of engineering applications in addition to the standard Unity applications. This is true of other colleges also, which add their specialty applications to the base suite of applications that is provided with the standard Unity lab configuration across campus.

New student accounts are automatically generated before New Student Orientation and are fully activated before the beginning of the semester. Accounts remain active as long as students are registered. New accounts for faculty and staff are automatically generated when their employee information appears in the NCSU Human Resources (HR) database. Accounts remain active for the duration of faculty and staff employment.

Users, including faculty and staff, are issued a unique username (or userid) and password to use in logging in. The username is generally composed of the first and middle initials and the first six characters of the last name. For example, if the user’s name is John Q. Public, then the username would be jqpublic.

Knowing your Unity login name and password is essential for access to campus online services, forms, and secured web pages.

A user’s account identification is composed of two parts:

(1) a username, usually 8 characters composed of the user’s first and middle initials and the first six characters of the last name.

(2) an initial password, which is the user’s ID number. This password must be changed to a password of the user’s choosing as soon as s/he logs in for the first time.

For individual information about your account, including your file space, email, and print quotas, class schedule, etc:

Go to http://sysnews.ncsu.edu and select User Info → Account Information or go directly to:
https://sysnews.ncsu.edu/tools-bin/user-lookup
The password that users are given initially is their student ID number, which is typed in the password field without dashes or spaces. To prevent unauthorized access to their files and quotas, users must change their passwords at the password web site, http://www.ncsu.edu/password/ when they log in for the first time. After that, users should regularly change their passwords at least once a semester.

The facilities cannot handle more users than they were designed for and are restricted to NCSU faculty, staff, and fee-paying students only. **Users must never share their passwords and accounts with anyone!**

**Account Privileges**

Users access a common environment and suite of software. Some software may not be accessible to all workstations and users, but in general, users can use nearly everything on the system, including electronic mail and connection to the Internet and the Web.

An account includes 50 Megabytes (MB) of disk quota and 30 MB of IMAP email quota. Users may also acquire additional storage (see http://www.ncsu.edu/it/rulesregs/services.html). For printing, users must purchase print quota from WolfCopy centers on campus. One of the first things a user should do before logging in for the first time is to buy print quota ($5 minimum purchase).

**Deactivating and Deleting Accounts**

Student accounts are **deactivated** on census day (the last day of drop-add) of the first semester that a student is not registered. When an account is deactivated, no one will be able to log in with that userid or have access to files stored under it. Accounts are **deleted** one year after account deactivation. For example, if a student is not registered by census day of the spring semester his or her account will be deactivated. If that student is also not registered the following fall semester’s census day, the account is deleted. Summer is not considered a separate semester for this purpose.

Faculty and staff computing accounts are normally deactivated when an individual is no longer on the University payroll. Accounts and account files are deleted one year after deactivation.

**After graduation:** Student access to e-mail and Eos/Unity file space is extended for four months after graduation, free of charge. This policy was put in place as a courtesy to help students who use e-mail and Web resumes as part of their job search and career-planning strategies. Web pages on the www4.ncsu.edu server, where student Web sites reside, remain accessible via the Internet while accounts are active. However, recent graduates will not be able to log in to public workstations in campus computing facilities and labs.
Summer Accounts

Students who do not take classes during the summer but who pre-register for fall semester classes will continue to have computing privileges throughout the summer. This means that students who are pre-registered for the fall will continue to be able to use the workstations in labs, check e-mail, log in to campus servers, and transfer files to and from Eos/Unity file space using FTP software via ftp.ncsu.edu. Web pages on the www4.ncsu.edu server will also remain accessible during the summer. There are no extra fees.

However, students who are not pre-registered for the fall semester will lose their account access, including the ability to access their Eos/Unity e-mail from off campus. This procedure applies to graduate and undergraduate students registered for both spring and fall classes.

Users who leave the university for any length of time, even for a summer, should copy any important information they wish to keep to floppy, CD, or Zip disk. If an account is ever disabled, the user will not be able to access it until s/he is again registered for classes.

Account Names: Eos or Unity?

Eos/Unity, referred to as the “realm,” is an expansion of the College of Engineering’s Eos system. The campus-wide implementation of Eos outside of engineering is called “Unity.” Faculty, staff, and students are allowed only one realm account, which has come to be called the Unity account. There is no separate Eos account. The same basic resources are available to the entire Eos/Unity realm.

What’s in a Name? Eos, not EOS

The word “Eos” is a proper noun, not an acronym. The Project Eos computing system is named after Eos, the Greek goddess of the dawn. Those who gave Eos its name hoped that the project would introduce new technologies and approaches to computing on campus.

Users have been interested in the source of this name. Because MIT’s Athena project was the forerunner of ours, many assume that Eos follows Athena or is her daughter. In fact, Eos precedes Athena in Greek mythology.

In this mythology, the world began when Gaea, the Earth, bore a son, Uranus. Their union produced the first race of gods, the twelve Titans, six brothers and six sisters. The union of two of these siblings, Hyperion and Theia, produced Eos, Helios (god of the sun), and Selene (goddess of the moon). Two other Titans, Cronus and Rhea, gave birth to Zeus. It was from Zeus’ head that Athena, the goddess of wisdom and war, sprang full grown.

Eos’ marriage to Astraueus, the starry sky, produced the four winds, including Zephyr, the West Wind (also a member of the MIT and NCSU computing pantheon). All was well with Eos until the god Ares fell in love with her. This made the goddess Aphrodite jealous, and in her anger, she cursed Eos with a “fondness” for mortal men. From that time on, Eos fell for every man she saw.

One mortal, Tithonos, she liked especially well. Eos persuaded Zeus to make Tithonos immortal so they could live together forever. Unfortunately, she forgot to ask Zeus to grant him eternal youth. In time, Tithonos grew so old and dried-up that Zeus finally turned him into a cicada. Through it all, the saffron-robed Eos rode daily through the heavens in her purple chariot bringing the rosy-fingered dawn to the human world below.
COMPUTER USE POLICIES

I. Introduction

North Carolina State University’s (hereinafter "University") computer networks, equipment and resources are owned by the University and are provided primarily to support the academic and administrative functions of the University. The use of this equipment and technologies is governed by federal and state law and University policies and procedures.

II. Regulatory Limitations

A. Accounts are for the exclusive use of the individual to which they were assigned, and users may not allow or facilitate access to University computer accounts, equipment, or restricted files or systems by others. Users may not set up a proxy or anonymous remailer for purposes of allowing access to others.

1. Students and employees of NC State are authorized users unless access privileges have been revoked under University procedures.

2. Guest accounts may be authorized by the Associate Vice Chancellor for Finance and Information Systems or the Vice Provost for Information Technology, or their designees.

B. The University may examine electronic information stored on or passing over University equipment or networks for the following purposes:

1. To insure the security and operating performance of its systems and networks.

2. To enforce University policies or compliance with state or federal law where:
   a. Examination is approved in advance by a dean, vice chancellor, or vice provost, and either
   b. there is a reasonable suspicion that a law or University policy has been violated and examination is appropriate to investigate the apparent violation, or
   c. examination is necessary to comply with a state or federal law.

Computer users should have no expectation of privacy in the material sent, received, or stored by them on or over the University computing systems or networks when the conditions of subparagraph 1, or 2(a) and 2(b), or 2(a) and 2(c) above have been satisfied.

C. The University reserves the right to limit access when federal or state laws or University policies are violated or where University contractual obligations or University operations may be impeded.

D. The University may authorize confidential passwords or other secure entry identification; however, users should have no expectation of privacy in the material sent or received by them over
the University computing systems or networks. While general content review will not be undertaken, monitoring of this material may occur for the reasons specified above.

E. All material prepared and utilized for purposes of University business and posted to or sent over University computing equipment, systems or networks must be accurate and must correctly identify the sender, unless a University administrator (department head or higher) approves anonymity for a University business purpose.

F. Any traffic on the University’s networks, stripped of information content, may be monitored for operational or research purposes.

G. All material prepared for purpose of University business and posted to or sent over University computing equipment, systems, or networks must be limited to information needed for University business. Personal quotations or other personal statements in signature blocks are not permitted. Supervisors are responsible for enforcement of this provision.

III. Personal Use

Authorized users may access University computing equipment, systems and networks for personal uses if the following conditions are met:

A. The use is lawful under federal and state law.

B. The use does not violate any policy or directive of the Board of Governors, the NC State Board of Trustees, the UNC General Administration, or the NC State administration.

C. The use does not overload the University computing equipment or systems, or otherwise negatively impact the system’s performance.

D. The use does not result in commercial gain or private profit, except as allowed under University intellectual property policies and the external activities for pay policy. However, in no case may University computing resources be used for solicitation of external activity for pay.

E. The use does not violate any University licensing agreements or any law or University policy on copyright and trademark.

F. The use does not state or imply University sponsorship or endorsement.

G. The use does not violate laws or University policies against race, sex, religious, disability, or age discrimination, or harassment.

H. The use does not involve unauthorized passwords or identifying data that attempts to circumvent system security or in any way attempts to gain unauthorized access.

I. The use does not involve sending or soliciting chain letters, nor does it involve sending unsolicited bulk mail messages (e.g., “junk mail,” or “spam,” or “MLM.”).

J. The use does not result in any direct cost to the University.

K. Any creation of a personal World Wide Web page or a personal collection of electronic material that is accessible to others must include a disclaimer that reads as follows:
L. University computers must be registered with NC State in the ncsu.edu domain. It is forbidden to register a non-ncsu.edu domain for any computer which is connected to the NC State network without prior approval of the Associate Vice Chancellor for Finance and Information Systems or the Vice Provost for Information Technology. If such approval is given, it must be made clear that the non-ncsu.edu address is using NC State resources for delivery.

IV. Use of Computing Facilities for Commercial, Advertising, and Broadcast Purposes

A. No paid advertising will be allowed on official University Websites. However, an NC State Website may contain a simple acknowledgment of sponsorship by an outside entity in the following form: “Support for this Website [or university unit] has been provided by ____________.”

1. An “official University Website” is any World Wide Web address that is sponsored or endorsed or created on authority of a University department or administrative unit. Websites on University servers are either “University Websites” or personal websites allowed by the University.

2. “Paid advertising” means advertising or promotional information provided in exchange for legal consideration, including money or other valuable benefits.

B. Personal web pages that are maintained by University computer account holders may not contain paid advertising. This guideline is consistent with the University policy against use of University resources for private gain or commercial purposes.

C. University computer account holders may send advertisements to news groups dedicated to advertising. The ads may not refer readers to a University telephone number. Like other personal use of State computing resources, the ads must be sent on an employee’s own time, not during hours when they are being paid to work.

D. University computer account holders may not “broadcast” E-mail messages without prior approval from a University official with the rank of chancellor, provost, vice chancellor for finance and business, or their designees. “Broadcast” means transmission of a message to a significant number of computer accounts on a University server or servers; the intent is to prevent mass mailings from tying up employee time and computer resources.

E. Registered marks of the University (e.g., an image of a wolf) may be used in the Websites of University computer account holders on the conditions that (a) they are not used for or related to private profit or commercial purposes, and (b) they do not mislead or confuse viewers as to whether the Web page is University-sponsored.

F. The Chancellor or designee may approve specific exceptions to the prohibition on paid advertising.

V. Violation of Policy

A. Any violation of this policy by employees may be “misconduct” under EPA policies (faculty and EPA non-faculty), or “unacceptable personal conduct” under SPA policies. For students, violations
are misconduct under the applicable student disciplinary code. Violators may be referred to the appropriate disciplinary procedure, and violations of law may also be referred for criminal or civil prosecution. Sanctions may include revocation of access privileges in addition to other sanctions available under the regular disciplinary policies.

B. Apart from referrals to disciplinary procedures, a University system administrator (or designees) may suspend a user’s access privileges or suspend services to a computer for as long as necessary to protect the University’s computing resources, to prevent an ongoing threat of harm to persons or property, or to prevent a threat of interference with normal University functions. As soon as practicable following the suspension of access privileges, the system administrator must take the following actions:

1. The user must be sent written or electronic notice of the suspension of access and the reasons for it, and notice of the time, date, and location at which the suspension may be discussed with the system administrator.

2. The user must be given an opportunity to meet with the system administrator at his or her earliest convenience to discuss the suspension and present any reasons the user has why the suspension should be lifted. The system administrator must reconsider his or her suspension decision in light of the information received at this meeting.

3. Following the meeting, the user must be sent a copy of the system administrator’s decision upon reconsideration, and must be notified that the user may appeal to the system administrator’s immediate supervisor if the user is dissatisfied with the outcome of the meeting.

VI. Application of Public Records Law

All information created or received for University work purposes and contained in University computing equipment files, servers or electronic mail (e-mail) depositories are public records and are available to the public unless an exception to the Public Records law applies.

VII. Additional Rules

Additional rules and regulations on computer use may be adopted by various divisions/departments to meet specific administrative or academic needs. Any adopted requirement must:

A. Comply with applicable federal and state laws;

B. Be consistent with the policies of NC State and the University of North Carolina;

C. Be posted in writing or electronically in a manner that is available to all affected users; and

D. Be filed with the Office of Legal Affairs, the Associate Vice Chancellor for Finance and Information Systems, and the Vice Provost for Information Technology.

Authority: Issued April 9, 1999, by Chancellor Marye Anne Foxe. Changes or exceptions to administrative regulations issued by the Chancellor may only be made by the Chancellor.

**Computer and Network Use: Remote Access and Labs**

All users of North Carolina State University’s computer and network systems must abide by federal and state law and University policies and procedures. The University’s *Administrative Regulations – Computer Use* issued by the Chancellor and reproduced above, govern all computer use at NC State. The following additional rules govern the computer and network systems supported by campus Information Technology/Computing Services and the College of Engineering.

**Use of Remote Access Services**

Information Technology provides a limited number of free modem lines and remote access servers for use by the University. The remote access servers are commonly called dialup servers and telnet servers, and include, among others, *login.ncsu.edu* and *ssh.ncsu.edu*. The dialup/login machines are to be used for academic work.

Do not play games. Do not use chat programs such as IRC, ICQ, or AIM. Do not use MUDs, MUSHs, or other recreational Bulletin Boards. Do not log into a dialup and then log into other machines.

Running any CPU or memory intensive programs for prolonged periods of time or in the background is prohibited. This includes, but is not limited to setiathome, rc5des clients, and graphical web browsers (such as Netscape). Do not use the dialups as batch or compute servers.

Do not use the dialups as servers to support graphical login sessions. Running a single program back to your display is acceptable, as long as it doesn’t violate the above rules.

Violators of the above mentioned rules will be handled strictly according to the Administrative Computer Usage Regulations and Computer and Network Usage Policy, so heed this message. You could lose your Eos/Unity privileges.

**Use of University Computing Labs**

1. All those who use University facilities are expected to take proper care of the equipment. Any equipment malfunction should be immediately reported to staff on duty or to the organization responsible for the facility. Users of computing facilities may not move, repair, reconfigure, modify, or attach external devices to the computing systems. **No food or drink** is permitted in University computing facilities.

2. Recreational use of workstations in University computing labs during periods of light usage is permitted; however, no one may play games or engage in other recreational activities on workstations when others are waiting to use them for academic purposes. It is the responsibility of game players to recognize when resources are needed and to give up their seats to other users. It should not be necessary for someone to ask them to move.

3. Users must realize that they are in an academic facility and refrain from noise, sound effects, violent motion, etc., which may disturb others in the facility.

4. Individual Computing Services (Unity) computing labs and other University–owned computing labs may post additional operational rules and restrictions, which must be filed with the University’s Office of Legal Affairs. Users are responsible for reading and following these rules.

Violations of policy will be treated as academic misconduct, misdemeanor, or felony as appropriate.
Misdemeanor or felony violation charges will be prosecuted to the fullest extent of the law and may result in the immediate and permanent loss of privileges. Student disciplinary proceedings may also be initiated against violators.

**Computer and Network Use Regulations and Rules**

See [http://www.ncsu.edu/it/rulesregs/](http://www.ncsu.edu/it/rulesregs/)

**ResNet Terms and Conditions**


**Policies in Eos Public Labs**

- No food or drink.
- No smoking or chewing tobacco.
- No sharing accounts and passwords.
- No display of sexually explicit images.
- No noise or loud talking.
- No recreational use of workstations if lab is busy.
- Do not leave workstation unattended.
- Do not turn off or unplug workstations.
- Always log out before leaving.
- Leave workstation area clean.
- Take care of equipment and furniture.

Vandalism, theft, harassment, security violations, and misuse of computing resources are grounds for dismissal from the university. Computers are for academic use only. No commercial use of resources is permitted.

**Use of Cell Phones in Eos Public Labs, approved January 31, 2001, by the College of Engineering Computer Committee.**

Owners of cell phones should exercise courtesy to those around them by taking personal calls that are longer than 15 seconds to an area away from computing facilities, preferably outside the building or in lounge areas. Ringers should be set on the lowest setting or on vibrate. Voices should be kept low when speaking so that the labs remain conducive to work and study.

Complete and updated policies on campus computing are available at:

[http://www.ncsu.edu/it/rulesregs/](http://www.ncsu.edu/it/rulesregs/)
There are more than 45 public labs and 1500 Eos/Unity workstations on campus, not counting smaller department clusters and office workstations. The tables here provide information about:

1. location of the lab and the administrative unit, department, or college responsible for its operation (column 1);

2. hours the lab is open, subject to change (columns 2-5);


4. name of the public printer(s) in the lab (column 7).

Please read and follow all published and posted lab policies.

<table>
<thead>
<tr>
<th>All-Campus Labs</th>
<th>Mon-Thurs</th>
<th>Friday</th>
<th>Saturday</th>
<th>Sunday</th>
<th>Workstations</th>
<th>Printers</th>
</tr>
</thead>
<tbody>
<tr>
<td>104 Avent Ferry Residence</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>20 Sun 40 Win 1 Mac</td>
<td>afc-104-1</td>
</tr>
<tr>
<td>ITD, 515-2269</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>afc-104-2</td>
<td></td>
</tr>
<tr>
<td>125 Bagwell Residence</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>9 Sun 13 Win 1 Mac</td>
<td>bag-125-1</td>
</tr>
<tr>
<td>ITD, 515-3706</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2413 D.H. Hill Library</td>
<td>24 hrs</td>
<td>Closes</td>
<td>9am-10pm</td>
<td>Opens 9am Begin 24hrs</td>
<td>12 Sun 31 Win 4 Mac</td>
<td>dhl-2413-1</td>
</tr>
<tr>
<td>ITD, 515-3364</td>
<td></td>
<td>10pm</td>
<td></td>
<td></td>
<td></td>
<td>dhl-2413-2</td>
</tr>
<tr>
<td>122 Bldg. P. E.S. King</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>10 Win</td>
<td>kvp-122-1</td>
</tr>
<tr>
<td>Village, ITD, 515-2430</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>109 Language and Computer Labs (Laundry)</td>
<td>24 hrs</td>
<td>Closes</td>
<td>8am-6pm</td>
<td>Opens 1pm Begin 24hrs</td>
<td>17 Sun 63 Win 4 Mac</td>
<td>lau-109-1</td>
</tr>
<tr>
<td>ITD, 513-2371</td>
<td></td>
<td>midnight</td>
<td></td>
<td></td>
<td></td>
<td>lau-109-2</td>
</tr>
<tr>
<td>119A North Hall</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>13 Sun 14 Win 1 Mac</td>
<td>nd-119-1</td>
</tr>
<tr>
<td>ITD, 515-3651</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>nd-119-2</td>
</tr>
<tr>
<td>103 Sullivan Residence</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>10 Sun 22 Win 1 Mac</td>
<td>sul-103-1</td>
</tr>
<tr>
<td>ITD, 515-6886</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>sul-103-2</td>
</tr>
<tr>
<td>1004 Tucker Residence</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>10 Sun 23 Win 1 Mac</td>
<td>tuc-1004-1</td>
</tr>
<tr>
<td>ITD, 515-8648</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>tuc-1004-2</td>
</tr>
</tbody>
</table>

ITD—Information Technology Division (NCSU Campus Computing)  
320 Total
<table>
<thead>
<tr>
<th>Engineering Labs</th>
<th>Mon-Thurs</th>
<th>Friday</th>
<th>Saturday</th>
<th>Sunday</th>
<th>Workstations</th>
<th>Printers</th>
</tr>
</thead>
<tbody>
<tr>
<td>100 Leazar, ITECS Operator on duty 24/7 515-3923 (disabled access)</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>124 Linux 16 Sun 5 Win</td>
<td>lez-100-1 lez-100-2 lez-100-color1</td>
</tr>
<tr>
<td>301 Leazar, ITECS</td>
<td>8am-10pm</td>
<td>8am-10pm</td>
<td>8am-10pm</td>
<td>8am-10pm</td>
<td>33 Linux</td>
<td>lez-300-1</td>
</tr>
<tr>
<td>219 Daniels, ECE</td>
<td>Teaching lab</td>
<td>Teaching lab</td>
<td>Teaching lab</td>
<td>Teaching lab</td>
<td>24 Win</td>
<td>dan-242-1</td>
</tr>
<tr>
<td>242 Daniels, ECE Operator on duty 515-7345 (disabled access)</td>
<td>8am-10:30pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>24 Win</td>
<td>dan-242-1</td>
</tr>
<tr>
<td>247&amp;253C Daniels ECE, same as 242</td>
<td>8am-10:30pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>74 Sun</td>
<td>dan-242-1 dan-247-1 dan-253-1</td>
</tr>
<tr>
<td>306/320 Mann, CE</td>
<td>8am-10pm</td>
<td>8am-6pm</td>
<td>1-6pm</td>
<td>1-10pm</td>
<td>20 Linux 7 Sun 32 Win</td>
<td>mn-320-1 mn-320-2 mn-320-color1</td>
</tr>
<tr>
<td>1403 Broughton,MAE (disabled access)</td>
<td>8:30am-11pm</td>
<td>8:30am-9pm</td>
<td>Noon–6pm</td>
<td>Noon-11pm</td>
<td>56 Win</td>
<td>br-1403-1</td>
</tr>
<tr>
<td>2408 Broughton, MAE</td>
<td>8:30am-11pm</td>
<td>8:30am-9pm</td>
<td>Noon–6pm</td>
<td>Noon-11pm</td>
<td>45 Win</td>
<td>br-2408-1</td>
</tr>
<tr>
<td>4231 Broughton, MAE</td>
<td>8:30am-11pm</td>
<td>8:30am-9pm</td>
<td>Noon–6pm</td>
<td>Noon-11pm</td>
<td>31 Sun</td>
<td>br-4231-1</td>
</tr>
<tr>
<td>224 Withers, CSC with operator</td>
<td>8am-10pm</td>
<td>8am-10pm</td>
<td>Closed</td>
<td>Closed</td>
<td>12 Linux 11 Sun</td>
<td>wi-224-1</td>
</tr>
<tr>
<td>400 Withers, CSC</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>16 Sun</td>
<td>wi-400-1</td>
</tr>
<tr>
<td>118 Riddick, ChE</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>3 Sun 31 Win</td>
<td>rd-118-1 rd-118-2</td>
</tr>
<tr>
<td>239 Riddick, MSE</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>1 Sun 20 Win</td>
<td>rd-239-1</td>
</tr>
<tr>
<td>2114 Burlington, NE</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>9 Sun 2 Win</td>
<td>bu-2114-1</td>
</tr>
<tr>
<td>106 Park Shops, IE</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>24 Win</td>
<td>ps-106-1</td>
</tr>
<tr>
<td>122 Weaver, BAE</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>9 Sun 10 Win</td>
<td>dsw-122-1</td>
</tr>
<tr>
<td>300 EGRC, ECE</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>24 hrs</td>
<td>6 Sun</td>
<td>egrc-300-1</td>
</tr>
<tr>
<td>1903 Student Health ITECS/ITD (disabled access)</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>1 Linux, 1 Sun, 1 Win</td>
<td></td>
</tr>
</tbody>
</table>

**648 Total**

190 Linux, 274 Windows, 184 Solaris

ITECS–College of Engineering ITECS
CE–Civil Engineering
IE–Industrial Engineering
NE–Nuclear Engineering
BAE–Biological and Agricultural Engineering
CSC–Computer Science
MAE–Mechanical and Aerospace Engineering
EGRC Engineering Graduate Research Center
CHE–Chemical Engineering
ECE–Electrical and Computer Engineering
MSE–Materials Science and Engineering
<table>
<thead>
<tr>
<th>PAMS Labs</th>
<th>Mon-Thurs</th>
<th>Friday</th>
<th>Saturday</th>
<th>Sunday</th>
<th>Workstations</th>
<th>Printers</th>
</tr>
</thead>
<tbody>
<tr>
<td>120 Dabney, PAMS</td>
<td>8am-10pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>3 Linux 2 Sun 30 Win</td>
<td>dab-120-1</td>
</tr>
<tr>
<td>315 Dabney, PAMS/CALS</td>
<td>8am-10pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>15 Sun</td>
<td>dab-315-1</td>
</tr>
<tr>
<td>718 Dabney, PAMS</td>
<td>8am-10pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>12 Sun</td>
<td>dab-718-1</td>
</tr>
<tr>
<td>G100 Harrelson, PAMS/SICL</td>
<td>Teaching</td>
<td>Teaching</td>
<td>Closed</td>
<td>Closed</td>
<td>36 Sun</td>
<td>ha-100g-1</td>
</tr>
<tr>
<td>G108 Harrelson, PAMS/SICL</td>
<td>Teaching</td>
<td>Teaching</td>
<td>Closed</td>
<td>Closed</td>
<td>75 Win laptops</td>
<td>ha-108g-1</td>
</tr>
<tr>
<td>247/248 Harrelson, PAMS</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>6 Linux 26 Sun 6 Win</td>
<td>ha-247-1</td>
</tr>
<tr>
<td>269 Harrelson, PAMS</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>21 Sun</td>
<td>ha-269-1</td>
</tr>
<tr>
<td>314 Harrelson, PAMS</td>
<td>Teaching</td>
<td>Teaching</td>
<td>Closed</td>
<td>Closed</td>
<td>95 Win laptops</td>
<td>ha-314-1</td>
</tr>
<tr>
<td>1111 Jordan, PAMS</td>
<td>7:30am-11pm</td>
<td>7:30am-10pm</td>
<td>9am-10pm</td>
<td>1-11pm</td>
<td>10 Sun</td>
<td>jo-1111-1</td>
</tr>
<tr>
<td>2105 Jordan, PAMS</td>
<td>Teaching</td>
<td>Teaching</td>
<td>Closed</td>
<td>Closed</td>
<td>16 Mac</td>
<td>jo-2105-1</td>
</tr>
<tr>
<td>6124 Jordan, PAMS</td>
<td>8am-5pm</td>
<td>8am-5pm</td>
<td>Closed</td>
<td>Closed</td>
<td>25 Sun</td>
<td>jo-6124-1</td>
</tr>
</tbody>
</table>

PAMS—College of Physical and Mathematical Sciences  
CALS—College of Agriculture and Life Sciences

Additional general and college lab locations and information is at
http://www.eos.ncsu.edu/labs/
http://www.ncsu.edu/it/essentials/connections/unity_computer_labs/unity_labs.html
http://www.ncsu.edu/it/essentials/connections/campus_computer_labs/campus_labs.html

Additional information on assistive technology and access is at http://www.ncsu.edu/it/dss/

Not all labs are listed here. New ones open up and equipment changes. Labs also have holiday or intersession hours that differ from their regular schedules. Use the following table to record information about other Eos/Unity labs you use, or for alternate schedules. Consult the up-to-date information online to get current information and to find out about new labs.
WORKSTATION BASICS

Users have little or no contact with the network’s server computers, which are maintained exclusively by system personnel. However, the client workstation is a complete and powerful computer in its own right, not just a terminal connected to a remote computer.

Before 1998, Eos/Unity was Unix only, principally running on Sun workstations. However, in 1998-99, Intel machines running NT were installed, and in 2000, Linux was introduced in some Eos labs. In 2001, Windows 2000 was rolled out to Unity labs and this year runs in both Eos and Unity labs. There are now three supported operating systems: Solaris 8 running on Sun UltraSPARC workstations, and Microsoft Windows 2000 and Red Hat Linux 9 running on Dell computers.

What the user sees on the screen is the same for all workstations of its kind (or platform), unless the user has customized the environment. However, the Sun and Dell platforms are substantially different from each other. In terms of hardware, they have different keyboard layouts and key names, and their monitors and mice vary in size and shape. Their interfaces are also different, but they connect to much of the same application software, network and file services, etc.
**The Monitor and System Unit**

Eos/Unity monitors are large with high-resolution screens. On older machines, the monitor sits on a box-like system unit that contains a fixed hard disk and a floppy disk drive. On the Sun Ultra 10s and Dells, the system unit is a “tower” unit that sits on the floor or beside the monitor. The workstation remains on at all times (check the indicator lights on the front). If it has been powered off, turn it on via the switch at the back (Sun) or the button on the front (Dell) of the system unit. The monitor must also be turned on; it does not turn on automatically with the system. Because the power up/down cycle is generally more taxing on the system than leaving it on, **do not turn off the workstation after use**. The workstation has a “sleep” state it goes into with minimal power demands.

**The Mouse**

The mouse is a device for pointing, selecting, and initializing operations on-screen. It is the main device for working with windowed applications. The mouse moves and positions the on-screen pointer, which changes its appearance depending on the region of the screen it is in.

You use the mouse by placing your hand over it, gripping it in your palm, sliding it over the tabletop, and pressing or **clicking** one or more of the buttons. The buttons are referred to as MB1 for the first or left button, MB2 for the second or middle button, and MB3 for the right button, if there is one. Suns typically have three-button mice; Dells have two.

If the mouse seems to be getting away from you—that is, you have pushed it too far away, or the pointer has gone off the screen—simply pick it up off the table and set it down where you want it. The cursor will appear on-screen again, and you will be able to move the mouse more comfortably. You may have to do this periodically as you work.

The mouse performs four functions: **pointing**, **selecting**, **holding**, and **dragging**. Most are done with the first mouse button, MB1.

**To point.** Slide the mouse over the table top until the pointer is at the place on the screen where you want it to be.

**To select.** When the pointer is in place, **click** or quickly press and release a button to select. To “click on” something means to select or initialize an operation. Sometimes you must **double-click**, which means that you click the button twice in rapid succession.

**To hold.** Press and hold down a button. This generally suspends an action. For instance, holding down a button may keep a menu on the screen until you release the button. Sometimes you may hold down one button (usually MB1) and click another button.

**To drag.** Hold down one of the buttons and pull the mouse toward or away from you, as indicated. This action generally highlights and selects a region of the screen.
The Keyboard

The keyboard is connected to the system unit and is the main device for creating and entering information into the computer. A typical keyboard appears below.

QWERTY Typewriter Keypad: This keypad (named for the first six letters on the top row of letter keys) is the familiar typewriter keypad with Return (or Enter), Shift, Tab, Caps Lock keys and spacebar.

The delete or Back Space key is in the upper right corner of this keypad. It deletes characters to the left when it is pressed singly or held down (holding down a key “repeats” or speeds up the action). This keypad also has a Control (sometimes written as Ctrl) key, an Alt key, and a Meta key, which is the key with a diamond on it on Suns. These keys are often pressed in combination with other keys for specific operations.

Function Key Strip: This keypad is a group of twenty keys called function keys arranged in a row across the top of the keyboard. These keys are used for special operations defined by the various software.

Numeric Keypad: This keypad on the far right of the keyboard is used mainly in mathematics and calculation programs.

Editing and Cursor Keypad: This keypad has several specially named keys for editing and maneuvering. Also on this keypad are four arrow or cursor keys. The cursor keys move the on-screen cursor symbol in the direction that the arrow indicates. The appearance of the on-screen cursor changes as it moves, depending on the software or the cursor’s location on the screen. However, its function remains the same: to locate the point of insertion and help you know where you are on the screen.
LOGGING IN

When you approach a workstation, the screen may be black or dimmed. Moving the mouse slightly or pressing any key on the keyboard will bring up an opening login screen similar to the one below.

Select inside the **Username:** or **Login:** field (point and click the left mouse button) and type your username. Use lowercase letters only, and do not press the spacebar before you type. Press the **Tab** or **Return** key to move to the **Password:** field (or simply click inside the field) and type your password. The password does not appear on-screen when you type it in; instead, you will see a series of asterisks. Press **RETURN**, or click the **OK** button to log in.

The following is the login screen for a Solaris workstation.

If you log in correctly, the background disappears and is replaced by the screen you will work on, or “desktop.” If you do not enter the information correctly, you will get an error message.

Click **OK** to try again.
Selecting other menus on the login screen brings up additional information about the workstation and logging in.

Selecting **Session**, either on a Linux or Solaris computer, will give you options for logging in with different environments.

The Session menu provides you alternatives to a standard login in case something is not working correctly. The options are:

- **Normal**  
  No changes. Log in normally.

- **DefaultEnviron**  
  Log in without using any preferences you set. This session ignores your configuration files but does not change them.

- **ExtraLite**  
  Login without a window manager. A single window comes for use in a quick session or to fix something with your environment.

- **RepairDotfiles**  
  Normal, but restores your preferences to what they were when your account was first created. Your configuration files (dotfiles) are renamed so that they are not called. Use this option when you need to fix your configuration files before restoring them.
If you are on a Sun or Linux computer, you can bring up other session environments through the Session menu. You may choose the Default Environment or Repair Dotfiles options if you have been configuring your personal dotfiles and need to bypass or repair these files at login (a new user is not likely to choose these options). The Extra Lite session logs you in without the window manager and brings up only a single window.

**Default Environment** logs a user in with the standard session environment files, bypassing any special configuration files the user has created.

**Repair Dotfiles** renames all personal configuration files (e.g., a numbered extension is added to `.mycshrc` to become `.mycshrc.##`) and copies the system default files (`.cshrc` and `.login`) into your home directory.
Logging in to an Eos/Unity Windows workstation is similar to the Solaris/Linux login, but the interface is slightly different. You must first hold down the Ctrl, Alt and Delete keys at the same time to bring up the Novell Client login window below.

Click in the Username field and type in your Unity ID, e.g., jqpublic. Then Tab and type in your Password, which does not appear when typed. Select OK or press Enter.

If you ever lose access to AFS and your K: and J: drives, select the KAUTH or Wolfcall icon and log in to reauthenticate to AFS.

You are first logged in to Novell Directory Services (NDS). A local account is created for you on the workstation, and the password is matched to your NDS/Unity password to log you in. A user profile is set up so you can use the Windows software and services stored locally. Next, you are authenticated through Kerberos to AFS so you can use Eos/Unity realm software, file space, and services.

For more information on logging in and using the Linux computers in the College of Engineering, see http://courses.ncsu.edu/e115/common/.

The chapter, Overview of Eos/Unity OSes, File System and Services, includes more information on what happens at login. Also, a full description of the Unix login is in Session Configuration.
The “look and feel” of the workstation environment is defined by the operating system and interface software that govern how the user interacts with the system. Unix (Sun Solaris), Linux (Red Hat), and Windows have substantially different user interfaces, although they access common applications, network file space, and computing infrastructure. You can move among them with little problem, but you have to learn to do many things in different ways.

The Solaris FVWM Interface

When a user successfully logs in to the Sun Solaris workstation, the login screen disappears and the following screen takes its place. While not very colorful or inviting, this default interface is created by the F(?) Virtual Window Manager for X11. It is designed to minimize memory consumption, provide a 3D look to window frames, and a virtual desktop.

No applications are displayed on the desktop by default. Instead, applications are launched via a pop-up menu (middle mouse button in the background, see next chapter) or the command line.
The gray background defines an area called the root window. The small window that first appears in the upper left corner of the screen is called the console window. It contains system and error messages, and it also monitors the user’s session and what goes on in the system. Users do not work in the console window, and closing it will log the user off! Rather, they should watch it closely for messages and other information.

**Command Line Interpreter**

The Xterm window is the terminal window that appears in the center of the screen. It contains a shell prompt, which is the location of the command line where commands may be typed into the system.

Users will often have to instruct a Unix workstation through commands typed at the prompt. These commands are interpreted, executed, and passed on to the Unix operating system by a shell program, which controls the user’s interactions with the operating system. The shell prompt tells which shell the system is running. The percent sign (%) means that Eos/Unity runs a C shell (csh) (see *The C Shell*).

A command is entered by typing it after the prompt in an Xterm window and pressing the RETURN key. Commands must be typed in the precise syntax that the interpreting shell program accepts. When the shell finishes processing a command, it comes back and displays the prompt again, which asks (or prompts) the user for the next command.

**The Unix operating system is case-sensitive!** The user must be careful to use uppercase and lowercase letters as specified. In Unix, commands and filenames are usually lowercase. For example, the command to copy must be typed as cp, not CP. The same is true of the filenames README, readme, and Readme. All are viewed as different files to Unix.

**Graphical User Interface**

Users also work through a graphical user interface, or GUI (pronounced “gooey”), which keeps them from having to type commands into the system. On this pictorial plane are windows, menus, and postage-stamp images called “icons,” which a user manipulates with a mouse to perform operations. A GUI is an easier and often more engaging way to interact with a computer. It can be used without much initial instruction and does not require that users look up or memorize commands.

Although the dominant metaphor of the graphical environment is the window, the interface also employs the metaphor of the desktop. Windowed applications can be viewed as documents and folders that are moved, arranged, and stacked on a desk. The console and Xterm windows are two of many windows that can appear and run on the Solaris desktop, as seen below.
Each window is self-contained, and the applications running in them do not interfere with each other, even if their windows overlap, hide one another, or are scaled down to a size too small to read. All Eos/Unity platforms support multitasking, the ability to run more than one program at a time. Information can also be copied, moved, and shared among the windowed applications.

**The Windows 2000 Interface**

The Microsoft Windows interface is a “point-and-click” GUI. Rarely will the user need to work from the Command Prompt, the equivalent of the Unix command line. Users will find the Windows interface more familiar to them than Unix’s because of the widespread use of Windows at home and in the workplace. The desktop is a colorful plane of icons that can be double-clicked to launch programs or access data.

Unlike the Solaris platform, which always looks the same no matter where you log in, the desktop you see on 2000 may vary some depending on where you log in and what your NCSU affiliation is.

For example, students in the labs will see different desktops depending on the college they are in because the colleges provide and support different applications. Engineering applications in the Engineering Applications folder are shown in the figure below and are only available.
Unity Applications are principally general-use applications for all campus users. In Eos engineering labs running Windows. However, the applications in the Unity Applications folder and subfolders are available to everyone using the standard installation, so those applications do not change from lab to lab. Users should look in the other folders that appear on their desktops to see what else is available to them. Note: The NAL shown here is as it was organized and appeared at the time of publication and may be slightly different from what users see.

The Linux GNOME Interface

The Red Hat Linux operating system, which is used in the College of Engineering on over 170 student machines, is taught to entering engineering students in the course, E115: Introduction to Computing Environments (http://courses.ncsu.edu/e115/). Linux is a freely distributable open-source implementation of Unix that runs on the Intel (or PC) platform, a less expensive but very similar platform to the Eos/Unity Sun Solaris platform.

GNOME (GNU Network Object Model Environment) is the desktop environment that runs on this platform and creates the graphical user interface. It has been customized to give users easy access to Eos/Unity applications and utilities that run under Linux.
As seen below, the GNOME interface has a Windows-style look and feel with the Realm Kit Application Launcher (RKAL), icons on the desktop for easy execution of programs. It also has a programs menu available from the GNOME footprint icon in the lower left corner of the screen (similar to the Windows Start menu). However, also seen are the familiar Unix Xterm window for command-line entry and the console window.

**Interface To What?**

Unix has been the principal operating system on Eos/Unity for nearly a decade, and a number of Eos/Unity packages run on the Solaris platform only. However, some of them are also available for Linux and Windows (see Appendix C: Licensed Commercial Software on Eos/Unity). A few packages run on all three platforms (Matlab, Maple, Visual SlickEdit, NAGware Fortran compiler, and the Java Development Kit). However, the greatest number and most popular applications are on the Windows platform.

Most of the applications that run on Solaris are science and engineering programs. The management and scalability of the Solaris platform was so successful that it was decided that MS Windows and Linux must run similarly in order to be realm clients.
System administration is handled remotely via the network since it is not practical to configure each machine individually and on site. The configuration is downloaded to the machine upon login, giving the user access to a commonly configured interface, shared hardware devices like printers, legally licensed software, and both shared and personal file space on servers that are backed up nightly.

User customization of the interface and software is still possible but kept in user file space on servers rather than locally on the machine or with the software. These customizations are applied upon login, making the phrase from the movie *Buckaroo Banzai* apt in describing the system: “Wherever you go, there you are.”

However, less customization is possible on Windows than Solaris because of its reliance on the local C: drive, which is wiped clean upon logout from lab machines. As a result, much of the customization you might usually do in applications, or via the Control Panel and Accessories, is not available to you on the lab machines. (Faculty and staff are permitted administrative access to alter settings in their office machines, however.) You will get error messages when you try to reconfigure settings and devices.

By the same token, no backup of the local drives is made, either of software or data. This means also that all your data (the files you create) must be saved to your home directory (the K: drive on Windows machines) or to other AFS directories to which you have access (the J: drive on Windows machines).

Users should view the Unix, Linux, and Windows interfaces as different paths to a common campus network of shared software, hardware, data, file space, and services. The developers of Eos/Unity recognize the importance of maintaining standards across colleges and have worked to minimize the differences in platforms and interfaces as much as possible.
The windows on the screen, and the applications that run inside them, can be manipulated and controlled via menus. Menus enable the user to point to and select a function, rather than having to remember the command or key combination to perform it.

**Pull-down menus** usually open from the top of a window frame or application. Often, they are anchored to symbols on the frame or a row of words at the top of an application called a *menu bar*. Menus drop from these anchor points when they are clicked, usually with the left mouse button. **Pop-up menus** open anywhere on the screen, usually by clicking other buttons. Generally, the location of the cursor on-screen signals to the computer what pop-up menu to open.

Menu items that are unavailable or *inactive* will appear fainter in color and outline. Options that have not been “grayed out” are *active*, that is, they will be executed when you select them.

**Window Control Menu**

All windows have a pull-down *window control menu* on the frame that lists functions for moving, resizing, and iconifying the window. This menu is anchored to a single window “button” left of the title bar. On the Sun Unix machines, this button is a square with a bar in it. On Windows, it is a small picture icon.

To “pull down” the menu, select the button with the left mouse button. A box opens that lists window-manipulation functions. To perform any of them, either select the function or type its underlined letter.

**Window Control Functions**

<table>
<thead>
<tr>
<th>choose . . .</th>
<th>to . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>restore</td>
<td>restore a window from an icon to maximum (full-screen) size.</td>
</tr>
<tr>
<td>move</td>
<td>move the window to another location on the screen.</td>
</tr>
<tr>
<td>size</td>
<td>size the window to make it larger or smaller.</td>
</tr>
<tr>
<td>minimize</td>
<td>minimize the window’s size (with the application still running in it) to an icon.</td>
</tr>
<tr>
<td>maximize</td>
<td>maximize the window’s size (application still running) to fill the entire screen.</td>
</tr>
<tr>
<td>lower</td>
<td>lower or “shuffle” a window to the back, or behind all windows</td>
</tr>
<tr>
<td>close</td>
<td>close the window and the application running in it.</td>
</tr>
</tbody>
</table>
The Unix Root and Application Menus

To see the two main pop-up menus you use to bring up applications and perform basic session functions on the Sun Solaris machines, move the cursor into the gray background. The cursor changes to an $X$. Hold down the third mouse button, and a menu called the **Root Menu** pops up with a list of functions on it. Release the button, and the menu disappears.

Now, hold down the middle mouse button, and the **Application Menu** pops up (see below). The arrows on the **Application Menu** indicate that there is more than one application available in some categories. These tools are listed in *submenus* (also called *cascading menus* because one menu drops off another in cascade fashion). Submenus can branch to other submenus, and in some software applications, the cascading goes on for several levels.

To move into a submenu for a category of tools, hold down the middle mouse button in the root window to bring up the **Application Menu** and pull the pointer down the list to the menu option you want. Then pull to the right, as the arrow indicates, and a submenu appears. Drag up or down this submenu to the application you want and then release. This action selects the application and brings it up in its own window.

Not all applications are available from the **Application Menu**. Many must be launched from the command line (type **add** to see a list).
**The Start Menu on Windows**

Users of previous versions of Windows may be familiar with the Program Manager, which organized applications into groups or folders. Windows uses groups (folders with icons that are double-clicked to launch the program), but it also puts all groups on the **Start** menu with the individual programs listed on cascading menus.

- **Programs**: lists all the programs that you can run on your computer.
- **Documents**: lists the last 15 documents you opened. Selecting one will open it in the application in which the document was created.
- **Settings**: displays the settings of the Control Panel, Taskbar, and Printers.
- **Find**: locates files and folders on your computer, a Search function.
- **Help**: brings up index of Help Topics.
- **Run**: opens a window in which you specify by name a program to run.
- **Shut Down**: provides options for a graceful shutdown of the machine.

**Linux/GNOME Program Menus**

Programs and applications are easy to find on the Linux platform and display from the top menu bar:

- Programs
- Favorites
- Settings
- Desktop

or from the GNOME footprint in the lower left of the screen.

**Keyboard Control**

Some users do not like to take their hands off the keyboard and prefer key commands to do the operations of menus and windows. Some of the main key combinations on the Unix workstations are listed below:

<table>
<thead>
<tr>
<th>Sun X/Motif Window Menu</th>
<th>Key Combinations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Open</td>
<td>Meta Spacebar</td>
</tr>
<tr>
<td>Restore</td>
<td>Alt F5, arrow keys to position</td>
</tr>
<tr>
<td>Move</td>
<td>Alt F7, arrow keys to position</td>
</tr>
<tr>
<td>Size</td>
<td>Alt F8, arrow keys to position</td>
</tr>
<tr>
<td>Minimize</td>
<td>Alt F9, arrow keys to position</td>
</tr>
<tr>
<td>Maximize</td>
<td>Alt F10, arrow keys to position</td>
</tr>
<tr>
<td>Lower</td>
<td>Alt F3, arrow keys to position</td>
</tr>
<tr>
<td>Close</td>
<td>Alt F4, arrow keys to position</td>
</tr>
</tbody>
</table>
### General Controls

<table>
<thead>
<tr>
<th>Action</th>
<th>Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete left</td>
<td>Ctrl 8 or Ctrl h</td>
</tr>
<tr>
<td>Delete right</td>
<td>Ctrl d</td>
</tr>
<tr>
<td>Delete Line</td>
<td>Ctrl w or Ctrl u</td>
</tr>
<tr>
<td>Backspace (no delete)</td>
<td>Ctrl b</td>
</tr>
<tr>
<td>Up Arrow</td>
<td>F11 or Ctrl p</td>
</tr>
<tr>
<td>Beginning of line</td>
<td>Ctrl a</td>
</tr>
<tr>
<td>Enter</td>
<td>Ctrl j, Ctrl c, Ctrl m</td>
</tr>
<tr>
<td>Clear</td>
<td>Ctrl l</td>
</tr>
</tbody>
</table>

Keyboard shortcuts are handled differently on Windows. Double-click **My Computer** → **Control Panel** → **Accessibility Options** to bring up a panel with labeled tabs for devices you can control. Select the **Mouse** tab and check “Use MouseKeys” to enable the numeric keypad (the panel of keys at the far right of the keyboard) to manipulate your mouse. Once the numeric keypad is set to function as mouse keys, use the arrow keys to move the cursor vertically and horizontally, and the **Home**, **End**, **PgUp**, and **PgDn** keys to move it diagonally. Once the cursor is positioned, type 5 on the numeric keypad to click the mouse.

With MouseKeys enabled on the numeric keypad, use arrow keys and **Home**, **End**, **PgUp** and **PgDn** to position cursor and 5 to select.

<table>
<thead>
<tr>
<th>Action</th>
<th>Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>Click mouse (left button)</td>
<td>5 on numeric keypad</td>
</tr>
<tr>
<td>Double-click</td>
<td>+ on numeric keypad</td>
</tr>
<tr>
<td>Right-click</td>
<td>– on numeric keypad</td>
</tr>
<tr>
<td>Both mouse buttons</td>
<td>* on numeric keypad, then 5 or +</td>
</tr>
<tr>
<td>Hold left mouse button</td>
<td>Ins on numeric keypad</td>
</tr>
<tr>
<td>Release left mouse button</td>
<td>Del on numeric keypad</td>
</tr>
<tr>
<td>Resume standard clicking</td>
<td>/ on numeric keypad</td>
</tr>
<tr>
<td>Switch between windows</td>
<td>Alt Tab</td>
</tr>
<tr>
<td>Same, but in order launched</td>
<td>Alt Esc</td>
</tr>
<tr>
<td>Open a window menu</td>
<td>Alt Spacebar</td>
</tr>
<tr>
<td>Open <strong>Start</strong> menu</td>
<td>Ctrl Esc</td>
</tr>
<tr>
<td>Copy image of desktop</td>
<td>Print Screen</td>
</tr>
<tr>
<td>Copy image of window</td>
<td>Alt Print Screen</td>
</tr>
<tr>
<td>Select All</td>
<td>Ctrl a</td>
</tr>
<tr>
<td>Simple Cut</td>
<td>Ctrl x</td>
</tr>
<tr>
<td>Simple Copy</td>
<td>Ctrl c</td>
</tr>
<tr>
<td>Simple Paste</td>
<td>Ctrl v</td>
</tr>
</tbody>
</table>

The workstation numeric keypad is enabled as mouse control keys by turning on MouseKeys via the Accessibility option on the Control Panel.

Accessibility options are not available on student lab computers with high security settings, which turn off most of the Control Panel and wipe the C: drive at logout. Accessibility Options are available only to machines with low security settings, such as office computers and special accessibility workstations.
The appearance of windows on the user interface, or of screens within a screen, is created by X Windows and the fvwm2 Window Manager on the Sun Solaris machines, and Microsoft Windows or Linux/GNOME on the Dell computers. Windows are the modus operandi for displaying and running applications and for communicating with the operating system. Learning to manipulate windows helps you work more effectively with the programs running inside them.

**X, fvwm2, MS Windows, and GNOME**

The X Window System, Version 11, was developed at MIT and is now distributed by X.org. Also called X Windows, or X11, this software is similar to Microsoft Windows. However, when users manipulate the windows (e.g., resize and shuffle them), they are not working directly with X but with a window manager. On the campus Unix machines, the window manager is fvwm2. fvwm2 is Unix’s standard graphical user interface and is responsible for making the windows look and operate in the same way. Together, X and fvwm2 create the “look and feel” of Eos/Unity on the Unix platform.

The interface on the Dell computers is Microsoft Windows, except for some engineering labs that run Red Hat Linux and GNOME.

**Selecting a Window**

You can only write data to one window at a time. In order for a window to receive data or input, it must be “selected,” that is, it must be identified to the system as the active window. To select a window, point anywhere inside the window or on its frame and click the left mouse button. The active window then comes forward to the front of the screen.

**Grabbing and Moving a Window**

You can move a window to another location on the screen by pointing to the top of the window frame (the cursor turns into an arrow) and holding down the left mouse button. This action “grabs” the window by the frame. Slide the mouse, and the arrow on the screen drags a “ghost” or “rubber band” outline of the window until you stop and release the button to position the window in a new place.
Or, select **Move** on the window menu and use the cursor keys to position the window.

**Sizing a Window**

To change the size of a window, point into the content area of a window and slowly move the cursor toward any part of the frame until the cursor changes to an arrow similar to those in the table below. When the cursor changes to one of these arrows, press and hold the left mouse button, drag the window to the desired size, and release the button. Or select **Size** from the window menu and use the cursor keys to size.

Where you grab the window frame determines how the window will be sized when you drag the mouse. Dragging shrinks or stretches the window from that point (see following table).

<table>
<thead>
<tr>
<th>To stretch or shrink the window...</th>
<th>Point and drag from...</th>
</tr>
</thead>
<tbody>
<tr>
<td>vertically from the . . .</td>
<td></td>
</tr>
<tr>
<td>top</td>
<td>top of the frame, above the title bar</td>
</tr>
<tr>
<td>bottom</td>
<td>bottom of the frame</td>
</tr>
<tr>
<td>horizontally from the . . .</td>
<td></td>
</tr>
<tr>
<td>left</td>
<td>left side of the frame</td>
</tr>
<tr>
<td>right</td>
<td>right side of the frame</td>
</tr>
<tr>
<td>diagonally from the . . .</td>
<td></td>
</tr>
<tr>
<td>top left</td>
<td>upper left corner of the frame</td>
</tr>
<tr>
<td>top right</td>
<td>upper right corner of the frame</td>
</tr>
<tr>
<td>bottom left corner</td>
<td>lower left corner of the frame</td>
</tr>
<tr>
<td>bottom right</td>
<td>lower right corner of the frame</td>
</tr>
</tbody>
</table>

**Iconifying (Minimizing) a Window**

Sometimes as you work, you want to shrink or move windows out of the way without closing down the application altogether. To do this, you turn the windows into **icons**. Icons (like the one in the margin) are small graphical representations of objects, applications, or windows arranged on the root window. Iconifying a window does not close the application in it; for example, an iconified program will continue to execute.

To **iconify** or **minimize** a window, point to and click the minimize button, which is one of two window buttons located in the upper right corner of the window frame (see figure below). The icon button is the left button of the two, a square with a dot or dashed line in it. When you click this icon button, the window quickly shrinks to an icon. You can also **Minimize** from the window menu.
To move the icon to another part of the screen, point to it and then press and hold the left mouse button. Drag the icon to the part of the screen where you want it, and then release the mouse button to position it there.

**Maximizing a Window**

The other button in the upper right corner, a square with a smaller square in it, is a maximize button. Clicking it will enlarge the window and its application to fill the whole screen. You can also select **Maximize** on the window menu brought up by right-clicking a minimized window icon.

**Restoring a Window**

To **restore** a maximized window to normal size, either click the maximize button again, or select the **Restore** option on the window menu. To restore an icon (change it back into a window), point to the icon and double-click the left mouse button.

**Scrolling a Window**

When you fill up a window with information, the contents move or **scroll** up and out of sight so that you have room at the bottom to add more information. Although these contents disappear from the window, they are not gone altogether. You can scroll down to bring them in view again.

On the left side of the window is a **scroll bar**, which allows you to control what part of the information appears in the window and to move quickly forward or back through it. The scroll bar has an arrow at each end and a white column or **slider** in the middle. The slider moves up and down as the information scrolls.

You can move the slider by clicking on the arrow buttons or by grabbing and dragging the slider. You can click the up or down arrow to scroll a little at a time, or hold them down for continuous scrolling. Or, you can grab the slider and drag it up or down to scroll more quickly.
Shuffling Windows

When several windows overlap on a screen, you can bring one of them to the front simply by clicking on it. This is how you shuffle windows, laying one on another as though you were shuffling cards. The selected window on top is the window that shows whatever you type. The windows behind the selected window are inactive.

**Special to Sun workstations:** You can also shuffle windows using the **Shuffle Up** or **Shuffle Down** options on the **Root Menu**, which is brought up in the gray background (root) window by holding down the right mouse button. Or, you can **lower** or send the window to the back by selecting **Lower** on the window menu, an option NOT on the Windows window menu.

**Special to Windows workstations:** Shuffling or arranging windows on the Windows interface is generally done from the **Task Bar** at the bottom of the screen. Minimized applications are arrayed there, and right-clicking in the bar’s gray background will bring up a menu that allows you to do a number of things to the positioning and layering of windows.

Closing a Window

To close an application and the window in which it is running, open the window menu and drag to **Close**.

Remember that the application closes with the window! If you do not wish the application to close, iconify the window to temporarily move the window out of way.

**Special to Windows and Linux workstations:** To the right of the minimize and maximize buttons at the top of the window is a button with an X in it. Clicking this X will close the window and its application. This is much faster way to close windows than using the window menu.

Also, on Windows computers, you can close multiple windows at the same time. If you hold down the **Shift** key when you click on the **Close** button (or when you select **Close** from a menu), Windows closes the current window as well as any **My Computer** windows or “parent” windows that are open.
The first thing that users should do when they log in for the first time is change their assigned passwords to passwords they choose themselves. Users should continue to change their passwords regularly, at least once a semester. The password is a lock on the account, keeping the user’s resources and information secure.

Users should use the Web page below so that their Eos/Unity passwords are synchronized with Netware Directory Services (NDS), which Windows 2000 uses. This synchronization allows them to move among all Eos/Unity platforms, Unix, Linux, and Windows. The password change page is http://www.ncsu.edu/password/

Change your password at http://www.ncsu.edu/password/
(The Unix passwd command is no longer used.)
Password changes take up to an hour to take effect. The Kerberos database, which handles user authentication, propagates at the top of every hour, a process that takes 10-15 minutes. (If you get the message, “Database locked. Try later.” Kerberos is going through its update.)

Follow these guidelines in choosing a password:

1. Choose a name 5-8 characters long that you can remember but no one else can guess. Do not write it down but make sure to memorize it. If you forget it, you will have to contact a system administrator to reset your password back to the original.

2. Do not use names of family members, your username, words commonly associated with you that can easily be guessed, or words that can be found in the dictionary.

3. Use some combination of letters and numbers. Because passwords are case-sensitive, you can also use a combination of uppercase and lowercase letters.

Forgot your password? If you forget your password, you must contact a system administrator at the Help Desk in 208 Hillsborough Building on Hillsborough Street, or in 200 Page, if you are in the College of Engineering. Bring your NCSU ID! System administrators cannot find out your old password for you. Rather, they reset your password to a new one, which you must then change to a password of your choosing.

Computing resources on campus are reserved for fee-paying users only, and your account is for your use only. You may not share it with friends or family members. Sharing your password and account is a serious policy violation, so please read the Computer Use Policies chapter carefully (see also http://www.ncsu.edu/it/rulesregs/).
GETTING HELP (help)


Manual Guide to Eos and Unity Computing in bookstores and online at http://www.eos.ncsu.edu/guide/

Course E115 Introduction to Computing Environments http://courses.ncsu.edu/e115/common/

E-mail help@ncsu.edu, eoshelp@ncsu.edu

Walk-in 8-5 M-F, 208 Hillsborough Bldg. (all, bring ID)
8-5 M-F, 200 Page Hall (engineering only, bring ID)

Phone/Fax 515-HELP (515-4357); Fax: 515-3787 (all)
515-2458; Fax: 515-7463 (engineering only)

In Xterm Type help at prompt

Zephyr zwrite -i help

An excellent resource is the help database at http://help.ncsu.edu/. It was built with questions sent to help@ncsu.edu and has an easy-to-use interface for searching for answers to frequently-asked questions (FAQ). The figure on the opposite page is the home page for the “help” site.

The Remedy Action Request System is a problem-tracking application in wide use in industry and other universities, including other UNC system schools.

Computing questions mailed to help@ncsu.edu enter the Remedy tracking system and are reviewed by consultants at Computing Services. The user who sends the mail first receives an automatic reply from “The Help Desk” to let him or her know that the mail was received and where and how to check on the call (http://help.ncsu.edu/callinfo.html).

If a call cannot be handled by the consultants, or it is intended for another computing group on campus, e.g., ITECS in the College of Engineering or Administrative Computing Services (ACS), then the call is placed in the work queue of that group. Users or clients of these computing groups and services on campus can mail directly into that workgroup in Remedy with the workgroup@help.ncsu.edu address, e.g., acs@help.ncsu.edu.
Help!

A database of answers to frequently asked computing questions at NC State.

To search the database, type your question in the form below, and then click the [Get Answers] button.

Enter your question:

Limit search to at most 10 documents returned

Get Answers  Undo Changes

Note: If you already have submitted to our Computing Consultants a question that was not in our database, you were given a Call ID Number. You can use this tracking number to determine the status of your call.

Quick Navigator

Find the status of a call  Go!

FAQ Help Database
http://help.ncsu.edu/
LOG OUT AND SHUT DOWN

When users are ready to quit work, they must initialize a logout or shut-down procedure that puts their files away, shuts down the programs running, and frees up space and memory for someone else to work.

It is very important to log out. If you leave the workstation without logging out, then the next person at the workstation has full access to your account, to read and send email, change or delete files, alter directory permissions, etc. Most people will not vandalize your account should they find it open, just as they would not vandalize your house if they found the door unlocked. Still, failing to log out leaves your account fully exposed, and that is a security risk you don’t want to take.

Before you log out or shut down the computer, make sure that you have saved any work you have done in the applications. It is a good practice to close down all applications individually to make sure that there is no unsaved work open that could be lost during logout.

Windows Shut-Down Procedure

There are three ways to log off a Windows machine. The safest way is to double-click the Logoff Windows icon on the desktop and select the OK button in the window that tells you, “You are about to close all running applications and end your Windows session.” It prompts you to go back and save your files if you have not already, so you can Cancel the logout if you need to do that.

If you click OK, all processes are killed, and you get no other chance to save. If you happen to leave without selecting OK, the session will terminate in two minutes anyway, killing all processes running so that your account is put away and protected from access by another user.

You can also select Ctrl+Alt+Delete, a custom logout procedure developed by ITD (see Logging In). The screen goes blank at first, and then a small window appears with some logout options, including access to the Task Manager. Choose Logoff...

Finally, you can logout via the Start menu, perhaps not the obvious place to look, but think of it as starting up the logout procedure. Select Shut Down to bring up the following menu. Wait until the login screen comes up before you leave.

The shortcoming of the last two procedures is that if a file is unsaved or a process is hung, the logout will not proceed, as it does for the Logoff
Windows option. You will get a message asking you to Wait, End Task, or Cancel the process. If you have already left the workstation, you cannot respond, and your account is left open. For this reason, you should wait until the login screen comes up before leaving.

**Solaris Shut-Down Procedure**

Logging out of a Sun Solaris workstation can be done two ways, from the Root Menu or from the command line.

**To logout from the Root Menu:** Move the cursor into the root window (the gray background of the screen) and hold down the right (third) mouse button. When the pop-up Root Menu appears, drag to the Logout option on the list and release the mouse button.

**To logout from the command line:** In any Xterm window on the screen, type logout (also on Linux workstations).

If you log out correctly, your windows will disappear. However, because you can make errors in running the logout program (you can accidentally drag past the word “Logout” on the menu or mistype logout at the command line), make sure that the opening login screen comes up before you leave the workstation. Always logout, but do not turn off the power to your system unit or monitor.

**Linux Shut-Down Procedure**

Both the Desktop and GNOME footprint menus have a logout function you select with the mouse and OK.
THE EOS/UNITY NETWORK AND SOFTWARE
Software is layered and integrated in such a way that it is sometimes difficult to see where one piece of software ends and another begins. Users are generally most familiar with the application software they use for specific work activities, e.g., word processing, e-mail, spreadsheets. However, in addition to the application software, there are a number of other programs that manage the way the system behaves, define what users can do, and provide services to clients and users.

**Operating Systems**

An operating system (OS) is the basic set of programmed instructions that tells a computer how to work. It is the intermediary between users and the hardware, performing operations according to user commands or input. The users’ interaction with the operating system can take place via a command interpreter (or shell), which runs in a terminal window on the screen, e.g., the Unix C shell in an Xterm window or Windows MS-DOS command prompt. Or, users can interact with the OS through a graphical interface, e.g., XWindows/fvwm2 on Solaris or GNOME on Linux.

The shell surrounds the **kernel**, which is the core programming that is responsible for machine-level operation and connection to hardware devices. The kernel cannot be accessed or modified by the routine user.

There are a variety of Unix OSes, and in the past, Eos/Unity supported IBM AIX, HP/UX, Digital Unix, and others, in addition to Sun Solaris. However, in 1999, multi-platform Unix gave way to a single Unix (Solaris) so that development work could be done to make Microsoft Windows a full realm client. In addition, Y2K issues made it necessary to phase out older and lesser used technologies.

Red Hat Linux followed in 2000 and was installed in Leazar and Withers labs on Eos workstations and taught in E115 to entering freshman engineering students. Eos labs ran Windows NT 4.0 in the 2001-02 year. Windows 2000 was introduced in 2002 and continues to run on nearly all Eos and Unity Windows machines.

For Windows, and now Red Hat Linux, system administrators have layered on top of their standard installations a number of customizations.
that allow them to behave like their Solaris counterpart and run the
system services that access campus file space, utilities, and servers that
make up the Eos/Unity system environment.

**System Services**

The system has layers of software that provide services to users and make
the system work efficiently. Though critical for running the system, they
are not always obvious to the user. These include:

- **Hesiod.** The Hesiod names server provides information
  about users, where their home directories are located, and the
  access privileges they have (to see the Hesiod record of a user,
  `hes userid` or `hes uid number`).

- **Kerberos.** The Kerberos authentication system is for system
  security. It checks passwords at login and verifies that users
  are who they claim to be. Authenticated users are granted
  Kerberos `tickets`, which allow them to access network utilities
  and work on the system.

**Andrew File System (AFS)**

AFS manages files in a distributed computing environment, assigning
storage locations and handling their distribution. AFS allows users to log
in to any workstation and access their files, granting them `tokens` to work
and making the system appear to the user to be a single homogeneous
computer system rather than the heterogeneous multi-computer network
it really is.

A distributed file system like AFS joins the file systems of individual
machines. As a result, users can get information from any place on the
network as easily as they would from a local disk. AFS hides the
existence of the underlying network and makes it easy for people to share
information and work together on the same files, no matter where those
files are located or where users may log in. However, AFS makes it just
as easy to prohibit other users from accessing files, thus ensuring network
security and individual privacy.

Users never really need to know the physical location of their files since
AFS does all the work of finding, storing, and managing them over
multiple servers. AFS also relies on the local workstation for the
processing and computation of retrieved files and software, which does
not slow AFS performance and improves overall network efficiency.

Using AFS effectively requires a familiarity with only a couple of AFS
commands on Unix, or the AFS menu and check boxes on Windows.
These are described in a later chapter, *AFS File-Sharing Commands.*
Activation of System Software at Login

All of these software pieces are brought into play at login. In a typical login session, the user enters his/her userID and password. The Hesiod database authorizes the user to use the workstation. The Kerberos database authenticates the user’s password and gives him/her a ticket file to access system services. AFS tokens are granted for the user’s home AFS cell (Eos or Unity), and the user’s home directory is attached. Then the session is configured, and applications are made available.

There are some differences in the Unix/Linux and Windows login, of course, and the general breakdown of each is outlined below.

What happens at login on Sun Solaris and Dell Linux workstations:

1. User enters userid and password.
2. Hesiod authorizes user to use the workstation.
3. Kerberos checks the password and provides the authenticated user with session tickets.
4. The user’s account is added to a password file.
5. AFS gets tokens for the user’s home cell (Eos or Unity).
6. The user’s home directory attaches.
7. System and user files configure the session. Mail, Zephyr, fvwm2, aliases, and other processes run.
8. The console, xterm, and window managers (fvwm2, GNOME) are started giving access to applications.

What happens at login on Windows 2000 workstations:

Windows 2000 environments uses key technologies from Novell, including NDS (Novell Directory Services), ZENWorks for Desktops, the Novell Application Launcher, and Norton Ghost and Boot Control (part of the Free DOS Project) to provide image restore capabilities.

1. User enters userid and password.
2. The user’s Novell Directory Services (NDS) password is checked.
3. The user is authenticated to NDS.
4. The user’s roaming profile is copied from the network.
5. The Novell Application Launcher (NAL) is started, containing Windows 2000 applications.
6. The Wolfcall (formerly KAUTH) application runs.
7. Kerberos provides the authenticated user with session tickets.
8. AFS gets tokens for the user’s home cell (Eos or Unity).

9. The user’s home directory is mapped to the K: drive.

The K: drive on Windows 2000 is directly mapped (shortcut) to the AFS space that is referred to on Unix/Linux as the user’s “home directory.” This is the individual network file space allocated to each user for his/her data and work, and it resides in the AFS file system in /afs/eos/users/ or /afs/unity/users/.
The applications that can be launched from the Unix Application Menu, the Windows Novell Application Launcher (NAL), and the Linux Realm Kit Application Launcher (RKAL) are briefly described in this chapter. These menus change as software is acquired or eliminated, so users may see some differences in what is displayed on their screens and what is described in this manual. Platforms, versions, descriptions, instructions for use, and documentation are also provided in Appendix C and at http://www.eos.ncsu.edu/software/.

**Solaris (Unix) Application Menu**

On Solaris, the applications most often requested by users can be brought up from the Application Menu (hold down the second mouse button in the gray “root” background to bring it up). The rest must be brought up from the command line. Type `add` at the prompt for a list of applications and launch instructions.

**Note:** If the following software runs on Eos/Unity platforms other than Solaris, those have been added in parentheses. Many of the applications are large and take some time to launch, so don’t keep executing them thinking that the launch has not worked. This will only cause multiple copies to launch causing delay and problems.

**New Terminal Window (xterm)**

The terminal window, or Xterm, is a terminal emulator program for the X Window System. A user can have many Xterms running at once on the same display, each of which provides independent input and output for the process running in it (normally a shell). It provides DEC VT102 terminal emulation for programs that cannot use the X Window system directly, and permits cutting and pasting to other windows. (Linux)

**E-Mail (webmail.ncsu.edu)**

Web Mail (webmail.ncsu.edu) is the principal electronic mail program on the system. It uses IMAP (or Internet Message Access Protocol), which stores email messages on university IMAP servers so that they can be accessed from any Web browser or compatible mail client on the Web. IMAP accounts are 30MB of file space for message storage, which is in addition to the user’s Eos/Unity 50MB quota. (Linux, Windows, and any Web browser anywhere. Requires login to Web page.)
Netscape Web Browser (netscape)

The World Wide Web is a distributed client-server information retrieval system. It organizes information on the Internet and links it together as hypertext documents (see World Wide Web). Netscape is a popular Web browser and the default browser on the Eos/Unity system. (Linux, Windows)

Print Manager (nxpr)

The nxpr Network X Print Manager (authored by former NCSU student, Lance Lovette) is designed to help users manage and conserve their printing. It allows the user to print 2-4 pages per sheet of paper and keeps track of the user’s print quota (see Printing). (Linux)

Eos/Unity Policies (policy)

Selecting this item will launch a Netscape browser (it may take a little time to load). It will open at pages describing the university’s policies on computer use. Users are responsible for following the policies and practices that have been put in place to protect people, equipment, and the university. These policies have been included in this manual (Computer Use Policies), but the Web pages hold the most current information and should be consulted as the best reference for these guidelines.

Help (help)

Typing help on the command line launches a Netscape browser to the Web address, http://help.ncsu.edu/. The FAQ or Help Database is the first resource you should try if you have a question or problem. It was created from the information gathered from Remedy, the problem-tracking software used at NCSU. Enter your question in the field provided and select the Get Answers button. If the question has been asked/answered before, generally on help@ncsu.edu, then it will probably be in the database. (see Help).

NCSU Libraries

Selecting this item (or type library in an Xterm) will launch Netscape and take you to the NCSU Libraries home page, http://www.lib.ncsu.edu. This page has links to online catalogs, indexes, databases, and the Triangle Research Libraries Network (TRLN) collections at Duke, NCSU, and UNC-CH.

Word Processing

There are three powerful word processing/publishing packages on Unix. Corel WordPerfect is the easiest to use and a full-function word processor (only on Solaris, see WordPerfect). Adobe FrameMaker is a comprehensive package for publishing long works in print or on the
Interleaf, which has been on the system for more than 10 years, is no longer being sold or supported. Users are urged to move to Framemaker, which has been upgraded to 7.0 on both Solaris and Windows.

Web. It offers book-building capability and supports exporting to HTML, XML, and PDF. FrameMaker is also available on the Windows platform in Eos labs.

StarOffice is available on Sun, Linux, and Windows workstations and can be launched from the command line (add staroffice).

Editors

<table>
<thead>
<tr>
<th>Editors</th>
</tr>
</thead>
<tbody>
<tr>
<td>NEdit</td>
</tr>
<tr>
<td>Visual SlickEdit</td>
</tr>
<tr>
<td>asWedit HTML Editor</td>
</tr>
</tbody>
</table>

Math and Statistics

Math and Statistics branches to a submenu listing several popular applications in math and statistics.

- **NExS Scientific and Engineering Spreadsheet**
  The Network Extensible Spreadsheet is an interactive spreadsheet designed specifically for the X Windows environment. It is similar to Microsoft Excel and is capable of monitoring several X applications at once and updating the relevant mathematical, matrix, string and statistical data from each (as well as any resulting calculations). It supports matrix, vector, and fourier transformation operations, as well as the arithmetic, Boolean, and logic functions available in the C programming language. (Linux)

- **Maple Numeric and Symbolic Computation**
  Maple Maple is a comprehensive computer system for advanced mathematics. It includes facilities for interactive algebra, calculus, discrete mathematics, graphics, numerical computation and many other areas of mathematics. It also provides a unique environment for rapid development of mathematical programs using its vast library of built-in functions and operations. Maple includes palettes, context-sensitive menus, and the input mode, Standard Math notation, that allow entry and manipulation of mathematical expressions without detailed knowledge of Maple syntax. (Linux, Windows)

- **MATLAB Computation and Visualization**
  The MATLAB numerical matrix manipulation package is a technical computing environment for numeric computation and visualization. The name “MATLAB” stands for matrix laboratory. As a high-performance language for technical
computing, MATLAB integrates computation, visualization, and programming in an easy-to-use environment where problems and solutions are expressed in familiar mathematical notation. MATLAB is an interactive system whose basic data element is an array that does not require dimensioning. This allows users to solve many technical computing problems, especially those with matrix and vector formulations, in a fraction of the time it would take to write a program in a scalar noninteractive language such as C or Fortran. MATLAB includes the Simulink extension and many add-on “toolbox” applications. (Linux, Windows)

- **SAS Applications System**
  SAS is an integrated applications system for data processing, statistics and analysis. It is a powerful programming language and a collection of ready-to-use programs called *procedures* for accessing, managing, analyzing, and presenting data. SAS programs may be used in operations research (models of distribution networks, resource allocation problems, scheduling, production systems), report writing and graphics, business forecasting and decision support, project management, and applications development. (Windows)

**CAD and Plotting**

A few of the several computer-aided design (CAD) and plotting tools on Eos/Unity are available on the Application Menu.

- **CADRA** The CADRA family of CAD/CAM products includes CADRA Design Drafting, a fast and highly productive mechanical design documentation tool; CADRA NC, a comprehensive 2 through 5 axis numerical-control programming application; and CADRA integration with SolidWorks, an integrated drawing production system and a 3D solid modeler. (SolidWorks is on the Eos/Unity system on the Windows platform.) (Windows)

- **Pro/ENGINEER** is a powerful tool for engineering design and an industry leader among CAD programs. Its parametric solid-modeling methodologies capture design intent and support feature-based modeling of parts and the subsequent combination of parts in assembly. Its fully associative architecture delivers a comprehensive suite of solutions for the development process, from a product’s conceptual design and simulation through manufacturing. (Windows)

- **Tecplot Interactive Plotting** is versatile and powerful interactive plotting software with extensive 2-D and 3-D capabilities for visualizing data from analyses, simulations, and experiments. With Tecplot, you can create XY plots,
contour plots, vector plots, mesh plots, carpet plots, 3D stream ribbons, isosurfaces, light-source-shaded surfaces, etc. You can also visualize complex data defined in one, two, or three dimensions, or on its original non-rectangular grid (e.g., multi-block, curvilinear, triangular, quadrilateral, 8-node bricks, and tetrahedral), preserving its original variation of grid resolution and retaining exact grid boundaries. (Windows)

**Graphics**

The three graphics programs available from this menu are **gimp**, **xpaint** and **xv**. The **Gimp** is a popular freeware package that has excellent capability and is comparable to Photoshop. **xpaint** is a color image editing tool which features most standard paint program options. **xv** is an interactive image manipulation program for the X Window System. Be aware that sophisticated graphics capability are also available in the commercial word processing and publishing packages listed above, as well as in the CAD software. (All on Linux)

**Miscellaneous**

The Miscellaneous category includes programs that are useful desktop tools to have available while you work. The first three—Calculator, Clock, and Rolodex—are self-explanatory. **XTRACS** is a course-scheduling program, written by former student Lance Lovette, that downloads TRACS listings nightly and allows students to arrange and graph their class schedules (see **XTRACS**).

**Additional Unix Applications on the “Add” List**

For additional applications on the Solaris platform, type **add** in the Xterm window at the prompt. You will see a screen like the one below and must press the spacebar several times to see the whole list. You cannot scroll back up the list and so must press **q** to stop and repeat the command to view the list again (application versions below are not current for 2003).
Novell Application Launcher

On Windows, the most-used applications reside in the **Novell Application Launcher**, called the **NAL** (see below).

---

The applications displayed here are from the spring 2002 NAL, so the applications could change in the 2002-03 year.
The rest are launched from the **Start** menu. Programs that users add themselves are appended under Windows Explorer. There are also useful applications under **Accessories**.

**Microsoft Office XP Applications**

The popular Microsoft Office applications are in the NAL in the **Office Applications** folder, including Word for word processing, Excel for spreadsheets, PowerPoint for presentations, and Access for database creation and management (also Photo Editor and Query).

**Other NAL Applications**

There may be more or different applications available in the NAL, but the ones available at the time of writing are listed and described briefly below. Additional Windows applications that are also available on the Sun Solaris platform are Maple, Matlab, Netscape, Pro/Engineer, SAS, and Visual SlickEdit, described earlier in this chapter.

A more complete list of software applications is in Appendix C and http://www.eos.ncsu.edu/software
http://www.ncsu.edu/it/essentials/software/

**Acrobat** by Adobe is used for publishing documents online in PDF format. It enables users to create and share documents across platforms while maintaining the documents’ original look and feel. Also on Solaris (**add acrobat**).

**ArcView** enables the user to visualize, explore, query and analyze geographic data spatially. ArcView is made by Environmental Systems Research Institute (ESRI), a leader in geographic information system (GIS) software. Also on Solaris (**add arcview**).

**AutoCAD** is an industry standard design and drafting package for the creation and manipulation of 2-D and 3-D line drawings and images.

**Java Development Toolkit (JDK)** contains software and tools for developers to compile, debug, and run applets and applications written in the Java programming language. Also on Solaris and Linux (**add jdk**).

**JMP** from SAS Institute, is a statistics analysis application that displays information in an easy-to-understand graphical environment. Data tables are presented clearly in spreadsheet form and dynamically linked to related graphics and tables. JMP offers six statistical analysis platforms, including a three-dimensional spin plot and capabilities for performing univariate statistics, analysis of variance and multiple regression, nonlinear fitting, multivariate analysis, and nonparametric tests.

**Macromedia Web Suite: Dreamweaver** is a visual editor for creating Web sites and pages. It permits site and page creation, updating, and
management through a WYSIWYG interface that supports Cascading Style Sheets, templates, layers, rollover images, ftp, etc. **Fireworks** allows you to create, edit, and animate Web graphics using a complete set of bitmap and vector tools. **Flash** is software for designing and delivering low-bandwidth animations, presentations, and Web sites.

**Norton AntiVirus** is virus detection and removal software available to NCSU faculty, staff, and students free of charge for their campus and home computers. The information and download pages are:

http://www.ncsu.edu/it/software/nav.html
http://www.ncsu.edu/it/antivirus/

**Photoshop** from Adobe Systems, Inc., is the premier and de facto standard for digital image enhancement, photo retouching, and image compositing. Photoshop allows the user to create original artwork, generate realistic or interpretive textures and backgrounds, correct color, retouch and composite scanned images, and prepare professional-quality separations and output for print or the Web. Also on Solaris but in the earlier 3.0 version (add photoshop).

**SolidWorks** is mechanical design automation software that allows designers to easily sketch out ideas, experiment with features and dimensions, and produce models and detailed drawings. SolidWorks is one of the industry leaders in CAD and 3-D solid modeling.

**XWin32** is StarNet’s X terminal application, which allows Windows users to connect to Linux/Unix servers on a network and run the applications from those servers on their Windows desktops (similar to Hummingbird’s eXceed).

There are additional applications available in the NAL and in other NDS folders that contain specific college applications (and therefore may not be available in all Eos and Unity labs). Right-click the applications icon and select **Properties** to get product description, help, and contact information for the application. Plus new applications are being added, including accessibility software (**JAWS** and **Zoomtext**).

If you have difficulty running an application, right-click its icon and select Verify. Then try opening it again.

**Software Documentation**

Documentation to support the software is often online inside the program, generally through the Help system. Also, check the software pages at [http://www.eos.ncsu.edu/software/](http://www.eos.ncsu.edu/software/) and the new documentation site, [http://manuals.eos.ncsu.edu/](http://manuals.eos.ncsu.edu/). The latter site was added this past year to provide users with access to user guides and documentation for the software applications on the system. The manuals are in both HTML
and PDF format, and a Unity login is required to access the site due to licensing restrictions. Print manuals can sometimes be ordered from the NCSU Bookstore, and D.H. Hill library also has copies of some manuals. In addition, the Web is a growing resource for software documentation, tutorials, guides, and FAQs.

**Commercial and Non-Commercial Software**

Commercial software is purchased and licensed by NCSU upon recommendation by faculty for use in classes and research. For these packages, NCSU offers support and maintenance.

Non-commercial software is not covered by an explicit contractual agreement with NCSU. It includes freeware, shareware, public-domain software, and software that is freely available for non-commercial or academic use. It may or may not be licensed, copyrighted, or in other ways protected by law. These packages are installed on the Eos/Unity system by NCSU staff, but in general are not distributed or supported by NCSU. Also, no upgrades or new versions are guaranteed. Many of these programs are excellent, but you must use them at your own risk.

**Tables of Application Software**

The software packages listed in the tables at the end of this chapter make up Eos/Unity’s main suite of software. The third-party vendor software, the so-called “commercial” packages, are listed in the first two tables, for Solaris and Windows. The third table lists public-domain, locally written, and system service applications for Solaris and Linux. There are many other public-domain, freeware, and shareware programs not listed in this table that users will want to explore themselves.

The tables place each package in a general category describing its primary function. However, many of the packages have other integrated functions that make them more versatile than the category may suggest. For this reason, an annotated list that describes more fully what each application does appears in Appendix C. The best source of information about Eos/Unity software is online at [http://www.eos.ncsu.edu/software/](http://www.eos.ncsu.edu/software/). At this site there are individual information pages for each package, which tell what workstations run the program, the commands to execute, the tutorials and documentation available, useful Web sites, etc.
<table>
<thead>
<tr>
<th>Software</th>
<th>Writing/Publishing</th>
<th>Analysis</th>
<th>Programming</th>
<th>Graphics</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access (Microsoft)</td>
<td>database mgmt</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Acrobat (Adobe)</td>
<td>PDF publishing</td>
<td></td>
<td></td>
<td></td>
<td>PDF writer</td>
</tr>
<tr>
<td>ADAMS (MDI)</td>
<td>mech sys simulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AMPL/CPLEX (ILOG)</td>
<td>optimization modeling</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ansys (Ansys)</td>
<td>finite element analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arcview (ESRI)</td>
<td>geographic info system</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Arena (Rockwell)</td>
<td>modeling/simulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Aspen (AspenTech)</td>
<td>chemical simulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AutoCAD (Autodesk)</td>
<td>CAD</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Automod (BrooksPRI)</td>
<td>modeling/simulation</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CADRA (Softech)</td>
<td>CAD</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Dreamweaver (Macromedia)</td>
<td>web publishing</td>
<td></td>
<td></td>
<td></td>
<td>HTML editor</td>
</tr>
<tr>
<td>Excel (Microsoft)</td>
<td>spreadsheet</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FactoryCAD (EDS)</td>
<td>AutoCAD extensions</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fireworks (Macromedia)</td>
<td>web graphics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Flash (Macromedia)</td>
<td>web graphics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>FrameMaker (Adobe)</td>
<td>print/web publishing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>GNU C/C++ (GNU)</td>
<td>C/C++ compiler</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JDK (Sun)</td>
<td>java development</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>JMP (SAS)</td>
<td>statistical analysis</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Lindo /Lingo (Lindo)</td>
<td>optimization</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Maple (Waterloo)</td>
<td>advanced mathematics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Mathcad (Mathsoft)</td>
<td>advanced mathematics</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Matlab (Mathworks)</td>
<td>math computation</td>
<td>language</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NAGf90 (NagWare)</td>
<td>Fortran compiler</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Norton AntiVirus</td>
<td>antivirus</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PhotoShop (Adobe)</td>
<td>photo/image dev</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Powerpoint (Microsoft)</td>
<td>presentations</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Primavera (Primavera)</td>
<td>project planner</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ProEngineer (PTC)</td>
<td>solid modeling, CAD</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Smalltalk (VisualAge)</td>
<td>OOP (object lang)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SAS (SAS)</td>
<td>statistical applications</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>SolidWorks (SolidWorks)</td>
<td>solid modeling</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>StarOffice (Sun)</td>
<td>ftp client</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tecplot (Amtec)</td>
<td>2D 3D plotting</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>TKSolver (UTS)</td>
<td>equation solver</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Visual SlickEdit (Slickedit)</td>
<td>program editor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Visio (Microsoft)</td>
<td>technical diagramming</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Word (Microsoft)</td>
<td>word processing</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XWin32 (Starnet)</td>
<td>X app server</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Software</td>
<td>Writing/Publishing</td>
<td>Analysis</td>
<td>Programming</td>
<td>Graphics, CAD</td>
<td>Other</td>
</tr>
<tr>
<td>------------------------------</td>
<td>--------------------------</td>
<td>---------------------</td>
<td>----------------------</td>
<td>---------------</td>
<td>---------------------</td>
</tr>
<tr>
<td>1. Acrobat (Adobe)</td>
<td>PDF publishing</td>
<td></td>
<td></td>
<td></td>
<td>PDF writer</td>
</tr>
<tr>
<td>2. ADAMS (MDI)</td>
<td></td>
<td>mech sys simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3. AMPL/CPLEX (ILOG)*</td>
<td></td>
<td>optimization modeling</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4. ANSYS (Ansys)</td>
<td></td>
<td>finite element analysis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5. ARC/INFO (ESRI)</td>
<td></td>
<td>geographic info system</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6. ArcView (ESRI)</td>
<td></td>
<td>geographic info system</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7. Cadence (Cadence)</td>
<td></td>
<td>circuit design, CAD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8. CADRA (Softech)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>CAD</td>
</tr>
<tr>
<td>9. Data Explorer (IBM)</td>
<td></td>
<td>visualization/analysis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10. FrameMaker (Adobe)</td>
<td>word processor</td>
<td></td>
<td></td>
<td></td>
<td>graphics/drawing</td>
</tr>
<tr>
<td>11. HSPICE (Avant)</td>
<td></td>
<td>circuit simulation</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12. Illustrator (Adobe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>graphics/drawing</td>
</tr>
<tr>
<td>13. JDK (Sun) *</td>
<td></td>
<td>Java development</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14. LSF (Platform)</td>
<td></td>
<td>batch processing</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15. Maple (Waterloo) *</td>
<td></td>
<td>advanced mathematics</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16. Matlab (Mathworks) *</td>
<td></td>
<td>math computation</td>
<td></td>
<td></td>
<td>language</td>
</tr>
<tr>
<td>17. NAG95 (NagWare)*</td>
<td></td>
<td></td>
<td>Fortran compiler</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18. Netscape (Netscape)*</td>
<td></td>
<td></td>
<td>Web browser</td>
<td></td>
<td></td>
</tr>
<tr>
<td>19. NExS (GreyTrout) *</td>
<td></td>
<td>spreadsheet</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20. Opnet (Opnet)</td>
<td></td>
<td>network modeling</td>
<td></td>
<td></td>
<td>graphics/drawing</td>
</tr>
<tr>
<td>21. Photoshop (Adobe)</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22. Pro/ENGINEER</td>
<td></td>
<td>solid modeling, CAD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23. Pro/MECHANICA</td>
<td></td>
<td>solid modeling, CAD</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24. Quantic (Quantic)</td>
<td></td>
<td>electronic signal anal.</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25. SAS (SAS)</td>
<td></td>
<td>statistical applications</td>
<td></td>
<td></td>
<td>language</td>
</tr>
<tr>
<td>26. Silvaco (Silvaco)</td>
<td></td>
<td>circuit design</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27. S-Plus (MathSoft) *</td>
<td></td>
<td>data analysis</td>
<td></td>
<td></td>
<td>language</td>
</tr>
<tr>
<td>28. StarOffice (Sun)*</td>
<td>word processing</td>
<td>spreadsheet, db</td>
<td></td>
<td>presentation</td>
<td>office apps</td>
</tr>
<tr>
<td>29. Synopsys (Synopsys)</td>
<td></td>
<td>digital circuit synthesis</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30. Tecplot (Amtec)*</td>
<td></td>
<td>2D 3D plotting</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31. Together Control Center (Togethersoft) Win/Linux only</td>
<td></td>
<td>software engineering</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32. Visual Slickedit (Slickedit)*</td>
<td>text editor</td>
<td></td>
<td>programming editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>33. WordPerfect (Corel)</td>
<td>word processor</td>
<td></td>
<td></td>
<td></td>
<td>graphics/drawing</td>
</tr>
</tbody>
</table>
### Table 3: Selected Non-Commercial Software on Solaris and Linux

<table>
<thead>
<tr>
<th>PD Software</th>
<th>Writing/Publishing</th>
<th>Analysis</th>
<th>Programming</th>
<th>Graphics, CAD</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Acrobat Reader</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>PDF reader</td>
</tr>
<tr>
<td>aswedit</td>
<td>html editor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Common Lisp</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Diffpack</td>
<td></td>
<td></td>
<td></td>
<td>partial diff. eq. libraries</td>
<td></td>
</tr>
<tr>
<td>elm</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>emacs</td>
<td>gnu emacs editor</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>g++</td>
<td>gnu C++ compiler</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>gcc</td>
<td>gnu C compiler</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ghostscript, view</td>
<td></td>
<td></td>
<td></td>
<td>PostScript viewer</td>
<td></td>
</tr>
<tr>
<td>gimp</td>
<td></td>
<td></td>
<td>graphics/drawing</td>
<td>PostScript viewer</td>
<td></td>
</tr>
<tr>
<td>gnu software</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>various gnu programs</td>
</tr>
<tr>
<td>LaTeX</td>
<td>text formatter</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>lynx</td>
<td></td>
<td></td>
<td></td>
<td>text browser</td>
<td></td>
</tr>
<tr>
<td>mathtools</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>mutt</td>
<td></td>
<td></td>
<td></td>
<td>electronic mail</td>
<td></td>
</tr>
<tr>
<td>nedit</td>
<td>text editor</td>
<td></td>
<td>editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Netscape</td>
<td></td>
<td></td>
<td></td>
<td>web browser</td>
<td></td>
</tr>
<tr>
<td>perl</td>
<td></td>
<td></td>
<td>language</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pgplot</td>
<td></td>
<td></td>
<td></td>
<td>plotter</td>
<td></td>
</tr>
<tr>
<td>pico</td>
<td>text editor</td>
<td></td>
<td>editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>pine</td>
<td></td>
<td></td>
<td>electronic mail</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Print Manager nxpr</td>
<td></td>
<td></td>
<td></td>
<td>print manager</td>
<td></td>
</tr>
<tr>
<td>psutils</td>
<td></td>
<td></td>
<td></td>
<td>PostScript utilities</td>
<td>PostScript utilities</td>
</tr>
<tr>
<td>rcstools</td>
<td></td>
<td></td>
<td>revision control</td>
<td></td>
<td></td>
</tr>
<tr>
<td>ssh</td>
<td></td>
<td></td>
<td></td>
<td>secure shell</td>
<td></td>
</tr>
<tr>
<td>Tcl/Tk</td>
<td>script language</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>vi</td>
<td></td>
<td></td>
<td>editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>vim</td>
<td></td>
<td></td>
<td>editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xanim</td>
<td></td>
<td></td>
<td></td>
<td>animation program</td>
<td></td>
</tr>
<tr>
<td>xemacs</td>
<td></td>
<td></td>
<td>editor</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xfig</td>
<td></td>
<td></td>
<td>drawing/graphics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xpaint</td>
<td></td>
<td></td>
<td>drawing/graphics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xrologex</td>
<td></td>
<td></td>
<td></td>
<td>rologex program</td>
<td></td>
</tr>
<tr>
<td>xtracs</td>
<td></td>
<td></td>
<td>TRACS class scheduler</td>
<td></td>
<td></td>
</tr>
<tr>
<td>xv</td>
<td></td>
<td></td>
<td>drawing/graphics</td>
<td></td>
<td></td>
</tr>
<tr>
<td>zephyr</td>
<td></td>
<td></td>
<td></td>
<td>message system</td>
<td></td>
</tr>
</tbody>
</table>
**RUNNING APPLICATIONS**

Launching and running applications are different on the three platforms (Unix, Linux, and Windows) and are described below.

### Launching Applications on Solaris (add)

The **Application Menu**, which is brought up by holding down the middle mouse button in the gray background, does not list all of the available software on the Sun workstation. There are a number of applications that do not appear on the menu that can be brought up from the command line. (Application Menu software can also be brought up by command.)

Most of the applications listed in the previous Table 2 require two commands to launch. The first command is `add` followed by the name of the software package you want to use. This command sets up the software on your workstation and runs any startup programs that are part of the package. To launch the program after it is added usually requires a single command, which is sometimes followed by the name of the file you want to create or work on in the program. This second command “executes” the program and brings it into a window on the screen.

If you do not remember the commands, type `add` at the shell prompt (eos%, unity%, etc.) to list the programs and the commands to execute them. When a program is added, a message usually appears telling you what command to use to execute (bring up) the program.

### Background and Foreground Processes

Programs running on your computer are called **processes**, and they exist in one of three states: background, foreground, or stopped. When an ampersand (&) is typed after the executing command, it tells the system to run the program in the **background**. The shell prompt is returned and you can continue to type commands in the terminal window.

If you do not type the ampersand, the program still runs, but you are not able to type commands in the terminal window you used to launch the program. In this situation, the process is running in the **foreground**, and it has read and write access to the controlling terminal. The shell must wait until the processing is finished before returning the prompt.

If you decide you want to background a process that has already been launched in the foreground, type **CONTROL z** (hold down the **CONTROL**
key and type z). This command puts the process in a *stopped* or suspended state and gives you back the prompt. Type **bg** to background the process. To bring a stopped process to the foreground, type **fg**.

Solaris and Linux assign a unique reference number, called a **process identification (PID) number**, to each process running. Users refer to the PID when they want to affect a process in some way, such as to terminate or **kill** it. To see what processes are running on your workstation, use the **ps** (or **-ef**) command if you are on a Sun. To kill a process running, type **kill** followed by the PID number assigned to the process. Sometimes a command or process can also be aborted by typing **CONTROL c**.

**Note:** When launching an application, whether from the command line or the Application Menu, you often must wait awhile before it appears on-screen (more than a minute at times). Be patient. Do not keep launching it again and again. If you do, you will eventually get multiple copies of the program running on your machine. You do not need more than one copy of a program taking up memory and slowing down the processing of the machine.

### Launching Applications from Novell and Realm Kit Launchers

The applications on the Windows and Linux platforms are accessible in an Application Launcher and appear as icons. Double-clicking an icon will launch the program.

On Windows, the applications are located in the **Novell Application Launcher** (NAL). They also can be launched from the **Programs** submenu on the **Start** menu. Most of these programs appear in Table 1 in the previous chapter. However, the applications you are able to see and launch may vary depending on the college you are in.

Applications and processes are identified and monitored on Windows by the **Task Manager**, which is brought up by right-clicking in the gray area of the task bar at the bottom of the screen. Selecting the **Applications** tab shows you what applications are running. The **Processes** tab shows all background and foreground programs running and provides a full accounting of session activity. To “kill” an application or process, select it and then select the **End Task** button.

The Linux platform has a similar launcher called the **Realm Kit Application Launcher** (RKAL), from which applications can be executed by double-clicking icons. A **Programs** submenu can also be brought up from the GNOME footprint icon, similar to the **Start** menu on Windows. The handling of processes is identical to the way they are handled on Solaris (see previous section). However, a Windows-style task bar appears at the bottom of the screen and displays an application panel that shows what is running. Tables 2 and 3 in the previous chapter show most of the applications available on Linux.
**Network Overview**

**client workstation:** a user’s computer that relies on server computers for its resources and services.

**server:** a resource-sharing computer that shares its files and provides particular support services to other computers on a network.

**cluster:** a group of client workstations connected off the same subnet and usually located close to each other.

**Fiber Distributed Data Interface (FDDI):** a standard for fiber-optics network technology. The ncstate.net network infrastructure supports a gigabit backbone (see also http://comtech.ncsu.edu/)

**Ethernet:** a 10-1000 Mbps (million bits per second) local-area network (LAN) connecting to the backbone.

**TCP/IP (Transmission Control Protocol/Internet Protocol):** the formal rules (protocols) that the Internet uses to support such services as file transfer and mail.

**router:** a dedicated computer that links, translates, and moves data in units called “packets” over networks.

*Client/server* describes the way components interact with each other. In the client/server model of interaction, a computer program at one site sends a request to a program at another site and awaits a response. The requestor is called the *client*, and the respondent is called the *server*.

The terms *client* and *server* can also be applied to hardware. Client workstations on Eos/Unity are usually computers for individual use that are grouped together in laboratory or department *clusters*. These clusters of workstations are managed and served by network server computers, which provide the software and files that a user needs to do work.

The client/server model bridges the two most familiar but disparate modes of computing, the standalone personal computer and the large time-sharing mainframe. When resources are downloaded into it, the workstation looks and behaves much like a personal computer. However, because it has no real capability without these resources, which come from a computer at another location, it also feels like a terminal hooked to a mainframe.

More than 1,500 client workstations are available to students, all connected to a high-speed *FDDI* data network by many *Ethernet* local-area subnets and several *TCP/IP routers*. These subnets minimize the load on the FDDI network. The local hard disks in the workstations, which are able to store most of the software and files used during a session at the workstation, ease traffic loads on both subnet and main network alike. The system also links users to other campus computers and to national and international networks.

Over fifty high-speed laser printers are connected to the network to handle the printing requirements of a cluster of users. There is no printer directly connected to a workstation. Instead, printing is “remote,” that is, it is a shared resource generally located in a cluster where a number of users can access it easily.

Servers are specialized in what they do for network clients. There are fours classes of servers on the Eos/Unity network: (1) *system servers*, which handle user authentication, name services, batch processing, mail, message notification, software licenses, and printing; (2) *file servers*, which manage the transfer and storage of user files; (3) *binary servers*, which manage the operating system and software files, and (4) *database servers*, which manage the various AFS databases on the system.
The diagram below enlarges one piece of the campus network, the Leazar lab subnet with some sample servers.

The diagrams on the following pages are more than three years old but give some idea of the campus network topology. More current information and diagrams are at http://comtech.ncsu.edu/networking/, including the following:

NCSU’s very fast, high-capacity digital network infrastructure, ncstate.net, supports a diverse, multiplatform, high-bandwidth distributed computing environment for the campus. From freshmen to administrators to advanced researchers, the NCSU community benefits from an excellent networking infrastructure and connections to global internets:

- Port/pillow switched ethernet connection in residence halls and fraternity/sorority houses
- 70% of 19,030 ports on campus with CAT 5 or CAT 6 wiring
- Building access via switches using resilient 2xGigE links
- Fiber-based 100 Mbps or 1000 ethernet connecting 85% of NCSU buildings to backbone
- Gigabit backbone with core switches connected via 4xGigE links
- DWDM-based 2.4 Gbps fiber ring access to NCREN and 290 Mbps of commodity Internet bandwidth
- 622 Mbps Internet2 (Abilene) link

The campus has 12 multimedia distance education classrooms capable of bi-directional, interactive analog audio/video transmission. 6 of these also support IP-based teleclassing. 3 high-quality external channels (analog or MPEG2) provide connectivity to NCREN and UNC system communities; 2 commodity channels allow for global gateway capabilities. The university’s growing nomadic computing system (VLAN) currently has more than 800 ports for users to plug in laptops, be assigned dynamic IP addresses using DHCP and authenticate to the campus network. The nomadic wireless system, which utilizes strong authentication protection, is currently in its testing phase.
The Eos/Unity Lab Network at North Carolina State University
Next Generation IP-Only Network and the Engineering Hub of the Legacy IPX/Appletalk Network

To itgw1, cmdfgw1, and smdfgw4 routers in Legacy IPX/Appletalk network (see next page)
The Eos/Unity Lab Network at North Carolina State University
Information Technology, Main Campus, and Centennial Campus Hubs in IPX/Appletalk Legacy Network

Campus Gigaswitch

David Clark Labs
Kilgore
Scott
N. Gardner
Nelson

D.H. Hill

North Hall

119A North Residence Hall

Textiles

Research 1
Research 2
Research 3
Corporate
ARRC
EGRG

cmdfgw1

Harris
Pullen
Schaub
University Graphics
Administrative Services
Western Studios
McKimmon
Gardner
Williams
Cox
Bureau of Mines
Broughton
Mann
Student Center

itgw1

Weaver Labs

122 Weaver
144 Weaver

Grinnells

1105 Grinnells

Dabney

315 Dabney
718 Dabney

G100 Harrelson
247 Harrelson
269 Harrelson

Jordan

1111 Jordan
2105 Jordan
6124 Jordan

100Mbps FDDI Fiber backbone
800Mbps Ether-Channel backbone
May 1999
WORKING WITH FILES AND DIRECTORIES
**Files and Directories**

All information in a computer system is stored in files. Programs and software are collections of files, and the information that users create using these programs is stored in files also. Files are further organized into directories, which typically group like files together. Directory names can be used to identify what kind of files the directory includes, such as, the program to which they belong or the user who owns them.

**Directories**

A directory is really a file also. However, instead of storing information or data as ordinary files do, these directory files point or link to other files, including other directories. (Directories are folders on Windows.)

The AFS file management software on Eos/Unity builds a tree-like hierarchy of directories for storing all software and user files on the system. It is this shared file space that makes the Eos/Unity system unique. Each user and every piece of software is somewhere represented as directories on the tree. AFS stores these user and software directories on network server machines and then retrieves them for the user, no matter when or where the user is logged in, or on which realm platform.

For example, each user who logs in to the system is automatically placed in a unique part of the tree with 50 MB of file space to use. That area of the tree is the user’s home directory where the files the user creates will be collected and stored. The home directory, identified by the user’s username, defines a branch in the tree that belongs to that user alone (e.g., /afs/unity/users/j/jqpublic/, see also Working Directories and Paths).

Technically, everything that branches from a directory is called a subdirectory. However, users tend to use the word “directory” not only for the topmost root directory, but also for any main directory of files. In other words, a directory usually refers to a main branch on the directory tree (a base or starting place), and a subdirectory to a more specific collection of files that branches from that base. A subdirectory is also called a child directory to the parent directory just above it in the hierarchy. Every directory, except the topmost root directory, has one parent, and directories can be nested to any depth.

Users’ home directories and files are not stored locally on the workstation drive but out on a networked server, which AFS finds and accesses for
them. User files are kept on these servers so that they can be kept secure and backed up nightly for the best protection of user data.

**Lockers**

*Locker* is another name for a directory or folder and is a term that originated at MIT in the Athena Project. Although the terms *directory* and *locker* are used interchangeably, the term *locker* usually means the collection of a main directory and all the subdirectories under it. Often, it is a separate AFS volume with its own quota and access control list. Lockers are generally accessible via the *add* and *attach* commands and are associated with a purpose, application, user, etc.

Users often call their home directories their “lockers.” Likewise, software is stored in distribution directories that are also called lockers. Professors may have “WolfWare lockers” or “course lockers,” that is, file space for their courses that their students can also access. Other kinds of lockers also exist for research and projects. In other words, lockers are storage directories in the file system (often publicly accessible) that are identified by their particular content, purpose, or creator.

**File and Directory Naming**

Operating systems have conventions that users need to adhere to in naming their files and directories. File names can be up to 256 characters in length, but obviously, users will want their file names to be relatively short, easy to remember, and accurately descriptive of the file contents.

Unix file-naming, like command entry, is case sensitive. As a result, the three names--file, *FILE*, and *File*--would represent three different files, not one.

Although most characters will be accepted in a file name, many should be avoided since they can cause problems or confusion. Some characters have already been assigned special meaning. For example, you should not use the “slash” character (/), which is the symbol for a directory. Also, spaces in file and directory names can cause you trouble.

The period should be used with caution since it has its own special meaning in a file name. The period is used to create an *extension*. These additional characters are frequently added to the end of a file name to identify it further. For example, the extension may identify *file.c* as C code or *file.txt* as an ASCII text file.

Also, if you use the period as the first character of a file name, you create a hidden file or *dotfile*. Dotfiles are “hidden” files that will not appear in ordinary directory listings (see *Customizing*).

**Note:** Do not give files in a directory the same name, and make sure that you follow a system that helps you remember file names.
WORKING DIRECTORIES AND PATHS

cell: an independently administered site running AFS consisting of a collection of server and client machines belonging to that cell.

The Unix command to find out what directory you are working in:

```
pwd
```

To see the global AFS file space and its independently administered sites, called “cells”:

On Unix and Linux:

```
ls /afs
```

On Windows:

```
My Computer -> Afs (J:)
```

To see the contents of your personal home directory in the AFS tree:

On Unix and Linux:

```
ls ~
```

On Windows:

```
My Computer -> Afs (K:)
```

You can get to your home directory (K:) via J: drive as well. K: is just a shortcut through AFS file space to your home directory.

All directories (folders) and files on Eos and Unity branch from the /afs directory, which also encompasses other sites in the world that run AFS. Two of these AFS cells, Eos (cell name eos.ncsu.edu) and Unity (cell name unity.ncsu.edu) branch from this global collection of cells into their own large but local file space (see figures at the end of the chapter).

The unity.ncsu.edu and eos.ncsu.edu cells under /afs branch again into many directories, one of which is the location of user home directories on Unity and Eos. The /users directory contains subdirectories from /a to /z in which the individual home directories are grouped alphabetically by username. The location of a file or directory in the file system tree (the drives and directories that contain a file) is called the path.

When you are working in your home directory, you are several levels down from the top of the AFS tree. If you are on a Unix or Linux workstation, you can find out exactly where you are by typing the Unix `pwd` command (path of working directory) at the prompt in an Xterm window. The system responds by writing the pathname of the directory. For example, if the user xzuser were in her home directory and typed the `pwd` command, the following pathname would be displayed.

```
% pwd
/afs/unity.ncsu.edu/users/x/xzuser
```

On the Windows platform, you access your home directory via My Computer and the K: drive. Double-clicking the K: drive icon will show you all the files in your home directory.

The path is not explicitly shown unless you set your My Computer options to display the full path in your title bar. Select View -> Options -> View, and then check the appropriate box.
You generally work in one directory or folder at a time. Whatever directory you are working in at the moment is called the working or current directory. A default directory is the directory that the system puts you in automatically.

For example, when you log in to a Unix workstation, you are placed “by default” in your home directory. Type ls at the prompt to see the files in your home directory. If you are in another directory, type ls ~ (or cd or cd ~ to take you “home”). The tilde (~) is the symbol for your home directory so you don’t have to type the full path to it. On Windows, you are placed by default on the “desktop” and can view the contents of your home directory by double-clicking the K: drive icon.

To illustrate, the home directory for Xy Z. User (xzuser) is in /afs/unity/users/x/ Remember that user directories are alphabetized by username, not last name, so xzuser is in the /x directory, not /u. Users can only get into their own home directories. Anyone who tries to access or look at someone else’s home directory will get a “Permission denied” message, unless different permissions have been set for the directory by its owner.

The path to one’s home directory originates in the root directory, the first slash. Other slashes join the names of subsequent subdirectories until the working directory is reached. This full or absolute pathname of a directory always starts at the root directory. A relative pathname begins in the working or current directory.

The root directory, /afs, can be reached with the command, cd /afs On Windows, this directory is the Afs (J:) drive (see figure on the previous page). The contents of the /afs directory are also displayed on the following pages, both as an Afs (J:) folder and as a Unix ls listing.

If you are granted access to other parts of the AFS tree, you will be able to move through and look at other directories, but only via the NCSU cells eos and eos.ncsu.edu and/or unity and unity.ncsu.edu. You may even be able to read and write in other directories if explicit permissions are set to grant you that access. For example, you might be granted access to a subdirectory under /afs/eos/lockers/ in order to write to one of the engineering lockers that are kept there.

The whole campus is interconnected through AFS, making possible a lot of common access and file-sharing. The key to getting where you want to go in AFS is the pathname of the desired location (i.e., file or folder), and, when you get there, the proper permission to access it (see AFS File-Sharing Commands). AFS pre-dates the World Wide Web but works in much the same way, that is, know the pathname of what you want and have the authorization to access it.

A pathname is simply a map that shows you how to get to a file through levels of nested subdirectories. It is like a route charted through a maze of streets that takes you to the address you want to reach.
% ls /afs  (same as J: drive on Windows)
adlw.nih.gov  eos  palo_alto.hpl.hp.com
andrew.cmu.edu  eos.ncsu.edu  pitt.edu
anl.gov  ethz.ch  psc.edu
athena.mit.edu  fnal.gov  pub.nsa.hp.com
bcc.ac.uk  gr.osf.org  rel-eng.athena.mit.edu
bp  grand.central.org  ri.osf.org
bp.ncsu.edu  graphics.cornell.edu  rose–hulman.edu
bstars.com  hrzone.th–darmstadt.de  rpi.edu
bu.edu  iastate.edu  rrz.uni–koeln.de
cards.com  ibm.uk  rus.uni–stuttgart.de
catt  inel.gov  sfc.keio.ac.jp
catt.ncsu.edu  ipp–garching.mpge.de  sipb.mit.edu
ce.cmu.edu  ir.stanford.edu  spc.uchicago.edu
cern.ch  isi.edu  stars.com
cgc  jrc.flinders.edu.au  stars.reston.unisys.com
cgc.ncsu.edu  lrz–muenchen.de  syzygy
cheme.cmu.edu  lsu.umich.edu  syzygy.ncsu.edu
ciesin.org  math.lsa.umich.edu  test.alw.nih.gov
citi.umich.edu  me.cmu.edu  theory.cornell.edu
cbm.cc.cmu.edu  media–lab.mit.edu  titech.ac.jp
cmf.nrl.navy.mil  msc.cornell.edu  transarc.com
cmu.edu  mtxinu.com  tx

cs.arizona.edu  nada.kth.se  tx.ncsu.edu
cs.brown.edu  ncat.edu  ucop.edu
cs.cmu.edu  nce  umich.edu
cs.cornell.edu  nce_cnc  uni–freiburg.de

cs.unc.edu  nce_psc  unity

cs.washington.edu  ncsa.uiuc.edu  unity.ncsu.edu

cs.wisc.edu  nd.edu  urz.uni–heidelberg.de

css.cs.utah.edu  nersc.gov  vfl.paramax.com
ctd.ornl.gov  net.mit.edu  wam.umd.edu
citr.se.ibm.com  next.mit.edu  zorglub

dmsv.med.umich.edu  northstar.dartmouth.edu  zorglub.ncsu.edu
dsg.stanford.edu  nsf–centers.edu

ece.cmu.edu  others.chalmers.se

% ls /afs/unity.ncsu.edu
adm  cds.afs.proxy  info  public  system
admin  contrib  lockers  sadm  users
bso  dist  project  source

% ls /afs/unity.ncsu.edu/users
a  b  c  d  e  f  g  h  i  j  k  l  m  n  o  p  q  r  s  t  u  v  w  x  y  z

% ls /afs/unity.ncsu.edu/x
xaruss  xdzhang  xiaoyun  xlo  xpsj  xvillaf  xyyu
xatondeu  xerxes  xie  xlu2  xqian  wang  xzhang2
xbj  xzhzh  xilin  xman  xqiao  xx cui  xxzhang3
xdai  xiang  xlin1  xmin  xsu  xxma  xzhou1
xdai1  xiaofeng  xliu  xmricharxu  xylem  xzhou2  xzuser

% ls /afs/unity.ncsu.edu/x/xzuser  (same as user’s K: drive on Windows)
/afs/unity.ncsu.edu/users/x/xzuser: Permission denied

75
WORKING WITH DIRECTORIES AND FOLDERS

In your AFS directories and folders, you will create a hierarchy of files and subdirectories to organize your work. If you create a directory on a UNIX or Linux workstation, it will appear as a folder on Windows, and vice versa. Although they perform the same function, directories and folders are created and manipulated differently on each platform.

UNIX Directory Commands

All UNIX commands are typed in a terminal (Xterm) window on the command line after the shell prompt. Check the path (pwd) to make sure you are creating, listing, or removing directories where you intend.

Make Directory (mkdir)

The mkdir command creates a directory. You type the command mkdir followed by a space and the name you want to give to the directory. For example, if you are in your home directory and want to make a subdirectory called homework, you would type

```
mkdir homework
```

UNIX creates a new subdirectory named homework and places it one level below the working directory (~/homework). It will appear as the homework folder on your K: drive on Windows.

Change Directory (cd)

The cd command changes or moves you into another directory. When you “cd” to a directory, you change your current working directory to the one you specify. For example, after creating the homework subdirectory in your home directory, you change into that directory by typing

```
cd homework
```

The subdirectory homework is now your working directory, and the files you create will reside in this subdirectory.

To change back to your home directory (the parent directory), which is one level up, simply type cd followed by a space and two periods, i.e.,

```
cd
```
To list the files in a directory:

```
ls
```

Some options to add to `ls`

- `-a` lists files including hidden files
- `-l` lists the mode, number of links, owner, size in bytes, and time of last modification for each file
- `-t` sorts by time when files were modified, with most recent first
- `-F` identifies type of file
  / beside directories
  * beside executable files, etc.

Options can be combined as in

```
ls -altF
```

To remove a directory

```
rmdir directory
```

Note: Sometimes `rmdir` will not remove a directory with a trailing slash.

Right: `rmdir birds`
Wrong: `rmdir birds/

**Attach and Add Directory (attach, add)**

The `add` and `attach` commands simplify directory access, especially when accessing directories that are not the user’s own.
Technically, these commands allow users to attach a remote file system to a directory hierarchy on their workstations and then map the path to it to a shorter name space. They create a link from the longer AFS pathname to the shorter one and also obtain AFS tokens that allow access to the files in that locker. As a result, instead of typing the long pathname (beginning with /afs) to change into the directory you want, you attach or add the locker you want and then change directories into it using the short pathname, /ncsu.

Add and attach do much the same thing. In fact, add calls the attach command but adds the bin directory to your path for that locker. As a rule, you attach user and course lockers because you are not likely to want to execute any files in those directories. For example, to attach the locker e115, where information about the E115 course is kept, you type attach e115, and then change into that directory with the command cd /ncsu/e115. However, because you DO want to run all necessary programs in a software locker, you add software, e.g., add nекс or add framemaker.

However, the act of attaching/adding a locker does not necessarily mean that you can use the files in it. The e115 locker permits you to read and copy its files; however, you cannot change or delete them. Other lockers you cannot access at all. Permissions to use a locker must be set by its owner or administrator (see AFS File-Sharing Commands). As a result, you are only likely to attach/add lockers that you know have been set up for you to use. These lockers may be for software, courses, research, or general World Wide Web access. Or, they may be the lockers of your friends and classmates who have specifically given you permission to access their files.

So, what really happens when you add a locker, for example, add gnu?

1. add creates a symbolic link for the locker in /ncsu, that is, /ncsu/gnu -> /afs/eos.ncsu.edu/contrib/gnu by means of the attach program. (To find out the path to the gnu locker before adding it, type hes gnu. After you add and cd into the gnu locker, type pwd to find out the path.)

2. It updates your PATH and MANPATH environment variables to put all of the locker’s executables in your directory path (at the beginning) and all of the locker’s UNIX manual pages in the path where the man family of commands looks for them (see The C Shell and Environment Variables and Man Pages).

3. It executes commands found in a file called .environment located in the root directory of the locker. This file is generally used to display information about the locker’s contents and to configure any additional environment parameters as necessary.
Specifying Directories

The following table lists some of the shorthand ways to specify directories rather than typing their complete pathnames.

<table>
<thead>
<tr>
<th>Use</th>
<th>To Indicate</th>
</tr>
</thead>
<tbody>
<tr>
<td>.</td>
<td>the current working directory</td>
</tr>
<tr>
<td>..</td>
<td>the parent directory directly above the current</td>
</tr>
<tr>
<td></td>
<td>working directory</td>
</tr>
<tr>
<td>../hwk1</td>
<td>the file \textit{hwk1} in the directory above the</td>
</tr>
<tr>
<td></td>
<td>current directory</td>
</tr>
<tr>
<td>~</td>
<td>your home directory</td>
</tr>
<tr>
<td>~/hwk1</td>
<td>the file \textit{hwk1} in your home directory</td>
</tr>
</tbody>
</table>

Windows Directory Manipulation

Windows replaces the MS-DOS environment entirely. As a result, it does not promote the use of the Command Prompt interface for directory and file manipulation. All directory control is mouse-manipulated, or “point and shoot.”

Make Folder

A directory or folder in Windows has its own special icon. To see the hierarchy of folders on a disk drive, you would select Start $\rightarrow$ Programs $\rightarrow$ Windows Explorer. The figure at the right is what you will see. Directories are represented by a yellow folder icon. To create a new folder, select File $\rightarrow$ New $\rightarrow$ Folder (see below).
Manipulating Folders

Changing into another folder is as simple as double-clicking it. When the folder opens, its contents are displayed, so no separate action is needed to list files. However, there are different ways to list the contents of a folder via the View menu. You can view files as large or small icons, or see the "Details" of their creation (time and date) and a full description (see below). To delete a folder, simply drag its icon to the Recycle Bin.

Likewise, there is no need to add or attach programs and locker space in order to use them. You can simply move through drives and folders as necessary to find them, and double-click the program icon to launch it (or select from a menu).

Finding Folders with Windows Explorer

Windows Explorer (Start → Programs → Windows Explorer) is a file manager program that hierarchically lists everything on your computer and the various drives and file systems to which it is connected. Under Tools, you can select Find to search through the system to find the files or folders you need (see figure at left). Or select Go to... to enter a path to the folder location you want, rather than double-clicking through folders.
**UNIX COMMAND OPTIONS AND ARGUMENTS**

Commands tell the system what actions to perform. However, if the user wants to modify those actions and specify that the action be performed in a particular way on a certain file, then the user must add that information to the command as options and arguments.

The portion of a command that names the files(s) or entity to be affected by the command is called an argument. For example, all of the Unix directory commands in the previous chapter specified and affected a directory file, which was the argument in the command syntax.

It is also common to add options after a command. Options are generally single characters or a whole word preceded by a hyphen. The option specifies a variation in the basic command, telling it to do something in a specific way. A full command consists of a command, followed by zero or more options, followed by zero or more arguments.

For example, if you type the command `ls` to list the files in a directory, the system will display a multi-column list of file and subdirectory names only. However, if you want more information about these files, e.g., when they were created or how big they are (View –> Details in Windows), then you would add the `-l` option to the command to specify a “long” listing. If you want hidden files (dotfiles) listed as well, you would add the option `-a` for “all.”

Additional options and arguments can be added on. For example, if jqpublic is in his home directory and wants a long listing of all files in his `~/homework` subdirectory in the order they were created with the most recent first, he adds a `t` (for time) to the `-al`, as in:

```
ls -alt homework
```

His directory listing would look something like the following:

<table>
<thead>
<tr>
<th>access permission</th>
<th>links</th>
<th>user</th>
<th>size</th>
<th>date last modified</th>
<th>time last modified</th>
<th>file name</th>
</tr>
</thead>
<tbody>
<tr>
<td>-rw–rw–r–</td>
<td>1</td>
<td>jqpublic</td>
<td>8900</td>
<td>June 19</td>
<td>08:23</td>
<td>hwk1</td>
</tr>
<tr>
<td>-rw–rx–r–x</td>
<td>1</td>
<td>jqpublic</td>
<td>1298</td>
<td>May 8</td>
<td>10.03</td>
<td>hwk2</td>
</tr>
</tbody>
</table>
There are many ways to manipulate and work with files in Unix, Linux, and Windows. Once again, Unix relies on command control and Windows on mouse and menu options. Linux with its GNOME interface makes use of both.

**Unix File Commands**

A command is simply a program that the computer runs. The following are the most frequently used Unix commands for working with files. These and additional commands are listed and defined in *Appendix B*.

**Display File Contents (more, less)**

The `more` command displays the contents of a file one screenful of text at a time in a terminal window. It normally pauses after each screenful, printing **More** at the bottom of the screen. If the user presses the spacebar, another screenful is displayed. The user can also move forward one page at a time by pressing the `f` key, and backward by pressing the `b` key. The `q` key quits or closes the file, the `h` key brings up a help screen, and the RETURN key scrolls the text line by line. When `more` reaches the file’s end, it exits and returns the prompt.

The `less` program is similar to `more`. However, instead of printing **More** at the bottom of each screen, `less` tells the percentage of the file that has been displayed up to that point, giving you some idea about how much of the file is left to page through. The `less` program also does not exit when you reach the end of the file; you must type q to exit.

**Concatenate Files (cat)**

The `cat` command (short for *concatenate*, to link together) will also display the contents of a file on your screen. This command dumps the contents of a file in one burst onto the screen, so if the file is very long, it will scroll past faster than you can read it (see margin for control sequences to manage this).

Used with other operators, `cat` will combine or merge the contents of two or more files in another file (see commands in the margin). In addition to these commands, the `cat` command can also be used to read input directly from the keyboard to create a new file, i.e.,

```
cat > filename
```
or to append what is typed to the contents of an already existing file, i.e.,

```
cat >> filename
```

When finished with direct keyboard input, press **CONTROL d** to quit.

### Copy Files (cp)

The **cp** command copies the contents of one file to another. The file name and contents remain unchanged in the *source* file, and a copy of them is placed in the *target* file. The target file may have the same name or a new one, but the contents of the two files will be identical.

Say, for example, you want to copy the file *hwk1* to the file *hwk1a* (**cp hwk1 hwk1a**). If *hwk1a* does not exist, it will be created and will contain the contents of the *hwk1* file. If it does exist, its contents will be overwritten by the contents of the *hwk1* file. However, if *hwk1a* is a directory, the system will know that, and will place a copy of the *hwk1* file in that directory, *hwk1a/hwk1*. The file is the same in both places.

If you want to copy files from one directory to another, you must specify a path. For example, the command **cp** *hwknew/hwk1 hwkold/hwk1* will copy the contents of the file *hwk1* in the *hwknew* directory to a file called *hwk1* in the *hwkold* directory (or just **cp hwknew/hwk1 hwkold** to place the file *hwk1* in the *hwkold* directory). To copy a complete directory with all its files, add the **-r** option, e.g., **cp -r directory1 directory2**.

### Move and Rename Files (mv)

The **mv** command is used both for moving and renaming files. The **mv** command moves the contents of the source file to a target file and then erases the source file. The effect is essentially that of renaming the original file. For example, **mv hwk1 hwk1old** will give the *hwk1* file the new name *hwk1old*. The contents of the *hwk1old* file are the same as they were in *hwk1*; only the name of the file is different.

The **mv** command also moves files as well as renames them. If you want your *hwk1* file moved from the *hwknew* directory to the *hwkold* directory, you issue the command **mv hwknew/hwk1 hwkold**. The system knows that *hwkold* is a directory and will not try to rename the *hwk1* file *hwkold*. It knows to move the *hwk1* file into the *hwkold* directory and to preserve the file name *hwk1*.

### Remove Files (rm)

To remove files from your account, use the **rm** command. This command will permanently remove the files from your directory space and cannot be undone. For more information about **rm** and its options, **man rm**.

On the Eos/Unity system, the **rm** command is “aliased” to **rm -i** (-i for interactive) so that **rm** will prompt the user for confirmation before
removing any write-protected files. This safeguard is good to maintain. However, if you want to remove files without prompting, you can unalias \texttt{rm} with \texttt{\textasciitilde rm}, e.g., type \texttt{\textasciitilde rm file}, or \texttt{\textasciitilde rm} * to remove all the files in the directory you are working in. Use \texttt{\textasciitilde rm} with great care!

**Windows File Manipulation**

Files, like folders, are opened in Windows by double-clicking them, or right-clicking them and selecting **Open** from the menu that pops up. If a file has been created with a Unix application, it may not open on Windows if the same application does not run under Windows, and vice versa. You will need to become familiar with what runs on all platforms.

**Display File Contents**

When you open a file (or **document**, as it’s generally called in Windows), it comes up in the program it was created in so you can continue working on it. Launching the program as well as the file may take some time, so you may wish to preview the file first before opening it in the program. For a read-only look at the file, select **Quick View** from the **File** menu, or from the shortcut menu that pops up when you right-click the document icon. Select the left-most button of the **Quick View** window if you want to go ahead and launch the document in its native application.

Documents that you have worked on most recently are listed for quick selection on the **Documents** submenu of the **Start** menu. An identifying icon tells you what application or program the file was created in, e.g., Excel, Word, Netscape HTML, etc.

**Copy and Paste Files**

Copying a file creates a duplicate of it for placement elsewhere, e.g., in the same folder, on a floppy disk, or in another file system or folder.

Simply select the file icon and do one of the following:

- Choose **Copy** from the **Edit** menu; then select where you want it go and **Edit –> Paste**.
- Right-click the icon and select **Copy** from the pop-up menu; then select where you want it go and right-click and **Paste**.
- Choose the **copy** icon from the toolbar (check **View –> Toolbar** if you want the toolbar displayed); then select where you want it go and select the paste icon (see following figure).
- “drag and drop” it with the right mouse button, and select **Copy Here** from the pop-up menu (see following figure). Or hold down the Ctrl key and drag with the left mouse button.

If you change your mind, select **Undo** copy or paste from the **Edit** menu.
Cut and Move Files

Moving files follows the same procedures as above, except that you Cut or Move as needed. “Cutting” removes the file from its original location, so follow immediately with a paste command so you don’t lose it. Or, drag and drop with the left mouse button to move a file. Select Undo from the Edit menu if you need to cancel an operation.

Rename Files

You can rename a file by selecting the file or object, then selecting the name area (a dashed line appears around the name). Type a new name and press Enter. Or, select an object or file and right-click to pop up a menu and choose Rename.

Remove/Delete Files

Removing or deleting files follows the same procedures as Cut, Copy and Move, except that you select Delete as needed instead. Select Undo from the Edit menu if you need to cancel a delete operation.

You can also drag a file to the Recycle Bin to delete it. On some systems, you must also empty the Recycle Bin to rid all deleted files from your disk space. Select the Recycle Bin icon (it sits on your Desktop) and then right-click to bring up a pop-up menu and choose Empty Recycle Bin.

File Properties

To find out more about a file—what type it is, what settings are in place, etc.—select it and then select Properties from the File menu or the pop-up menu brought up by right-clicking the file.

Linux File Manipulation

The Linux Realm Kit has been designed to give users the standard Unix interface and, with GNOME, something akin to the file manipulation of Windows. As a result, users will be able to adapt and apply both the Unix and Windows file-manipulation functions described above on Linux.
A few of the following suggestions will save you time and make working on Eos/Unity easier. Most are just available on UNIX and Linux.

**Copying and Pasting between Windows**

One of the most useful functions is the ability to copy and paste information from one windowed application to another. On UNIX, all you have to do is point, hold down the left mouse button, and drag over the information you want to copy to select it. Then point into the application where you want the information to be copied and click the middle mouse button to place it there.

You cannot do this quite as easily on Windows. You must use the standard copy and paste methods (see *Manipulating Files*), or use the keyboard shortcuts: **Ctrl a** (select all) **Ctrl c** (copy), **Ctrl x** (cut), **Ctrl v** (paste).

**Completing Command Entry without Typing**

You can also use the **Tab** key to save you time in typing commands, pathnames, or data. For example, if you issue a command that affects the file *software_list*, you do not need to type all the characters in this long filename. You can simply type the first characters in the name, until it is distinguished from all other files in the directory, and then press the **Tab** key to let the system finish the typing for you. (N/A on Windows)

For example, if *software_list* is the only file that begins with the letter *s* in the directory, then the **s** and **Tab** keys are all you would have to type to enter this filename on the command line, e.g.,

```%
more s[Tab]
```

This shortcut is also very useful when typing in long pathnames, which on Eos/Unity is something you have to get used to doing frequently. You can simply type a part of a directory’s name and then press the **Tab** key.

**Repeating Commands without Typing**

To repeat a command that you have typed previously in a session, press the up-arrow cursor key until you find the command again. To execute the command after you have found it, simply press the **RETURN** key. If you want to modify the command before executing it, use the right and
left cursor keys to position the cursor and then type in or delete characters as appropriate. (N/A on Windows)

The history command also displays the last commands you typed in a session up to whatever number is specified in the shell environment (type set to see the setting for savehist; see The C Shell chapter). If the savehist variable is set to 100, then your last 100 commands have been recorded and can be displayed with the history command. (N/A on Windows)

The commands are numbered in the order you typed them. If you want to execute the last command you typed, type two exclamation marks (!!) at the prompt, or use the up-arrow key as described above.

% !!

If you want to execute an earlier command in the list, type an exclamation mark! (sometimes called a “bang” or a “shriek”) followed by the line number of the command, line 34 for example.

% !34

**Aliases (alias)**

Often, you find yourself typing the same commands, filenames, and paths over and over. Or, you find that you confuse UNIX commands with the commands of another operating system you know better. If you wish, you can create substitutions or aliases for these files, paths, and commands, using something shorter and easier to remember. (N/A on Windows) For example, if you want to use the shorter command md instead of mkdir to make a directory, type,

% alias md mkdir

The command alias is followed by the substitution you want to use and then the original path, filename, or command (or a combination of these) that you want the alias to replace. If you have aliases that you want to use all the time, put them in your ~/.myschrc file (see Customization).

To take out an alias, use the unalias command. For example, to undo the md alias you created for the UNIX mkdir command, type

% unalias md

**Useful Key Combinations**

Individual keys and key combinations will perform particular operations and functions that can help you or save you time. When you see instructions that tell you to press some kind of “CTRL+character” combination, it means to hold down the CONTROL key (or the Ctrl key
Frequently-used key combinations:

CTRL z  Suspend a process
CTRL c  Cancel or abort a process

Wildcards and Metacharacters

Wildcards are special characters designed to work in place of other characters and strings of characters in command arguments. They save you time when you are trying to work with a number of files at once.

The asterisk * (called a star) can be used in place of any string of characters, even a string with no characters in it. For example, if you want to list (command ls) only the files in a directory with the extension .doc, then you can type

    ls *.doc

You can use the wildcard with any command that operates on files: copy, remove, print, etc. Thus, you do not have to work with files individually but can manage them as groups. The * by itself stands for all files. As a result, if you type rm *, you will remove all the files in a directory. Be careful with wildcards, and use the -i option with commands like rm. This option asks the user for confirmation before removing each file.

The question mark ? represents a single character. For example, typing

    ls essay?

lists all the files you named essay followed by a single character, such as,

    essay1 essay2 essay3 essay4

Or, you can use a series of question marks to stand for two characters or more. The command ls essay?? picks up essay10 essay11, etc.

Redirection of Input and Output

The keyboard and terminal screen are UNIX’s standard input and output devices, respectively. UNIX expects input from the keyboard and always sends output to the screen, unless told to do otherwise. Redirection is the process of changing the source and destination of input and output.

To UNIX, your screen is just another file. When a command generates output, the shell writes it to a standard output (abbreviated stdout) file, which puts the data on your screen. However, you can redirect this output if you wish, sending it to another file instead. For example, if you want a directory listing sent to a file rather than to your screen, you might type

    ls -al > myfile

where myfile is the name of the file that receives the directory-listing output. The right-angle bracket (>) tells the shell to perform the
Redirection. If the file *myfile* does not exist, it will be created. If it does exist, its contents will be overwritten by the new data. To append the data to the contents of a file rather than overwriting them, use two right-angle brackets instead of one. The following command places the directory listing at the end of any data already in the *myfile* file.

```bash
% ls -al >> myfile
```

Redirecting input is a similar procedure. Using the left angle bracket (<), you can specify input from a source other than the standard keyboard input device. For example, if you want to use the contents of the *myfile* file as input to the *Elm* mail program, you type

```bash
% elm jqpublic < myfile
```

This command uses the contents of the *myfile* file as input to *elm*, which sends these contents as a mail message to the user, jqpublic.

To merge the contents of files together, that is, concatenate them, use the *cat* command and redirection. For example, *cat* *file1* *file2* > *file3* will merge *file1* and *file2* into a new file called *file3*. The right angle bracket tells the system to send the result of the concatenation to *file3* (see *Manipulating Files* for more on *cat*). (N/A on Windows)

### Pipes and Pipelines

The process of *piping* sends output to a command or a series of commands. A series of commands is called a *pipeline*. You pipe data from one command to another by entering the commands on a single line with a vertical bar between them. In this way, the output of one command becomes the input to another command. (N/A on Windows)

For example, when you enter a command and its output is too large to fit on one screen, you may want to cancel the output (type *CONTROL c*) and type the command again, this time “piping” the output through *more* so it will be displayed in a format that you can move through more easily.

Say you have a directory with many files in it, so many that they scroll off the screen before you can read them when you do a directory listing. You can pipe the *ls* command through *more* instead, e.g.,

```bash
% ls -l | more
```

Or, perhaps you would like to know who is logged on in Park Shops. You can run the *zwho* command to find out who is currently on Eos/Unity, and then pipe it to the *grep* command, which searches for a particular pattern (in this case, *ps*, the abbreviation used for Park Shops machine names).

```bash
% zwho | grep ps
```

<table>
<thead>
<tr>
<th>jjoine</th>
<th>Jeffrey Allen Joines</th>
<th>c50010–114ps.ie.ncsu.edu</th>
<th>May 20 07:43</th>
</tr>
</thead>
<tbody>
<tr>
<td>kay</td>
<td>Michael Kay</td>
<td>c50029–208ps.ie.ncsu.edu</td>
<td>May 20 09:51</td>
</tr>
<tr>
<td>yslee</td>
<td>Yuan-shin Lee</td>
<td>c50046–206ps.ie.ncsu.edu</td>
<td>May 25 15:37</td>
</tr>
</tbody>
</table>

Note: Redirecting and piping are similar. However, redirection sends input and output to a file, whereas piping sends output to a command.

Pipe: an interprocess data channel. A *pipeline* is a sequence of one or more shell commands separated by a pipe symbol.
Information is available online to explain how to use most of the commands you encounter on the system. In fact, all UNIX commands are described in the online documentation called the UNIX Man(ual) Pages. A separate man page exists for every command. The man command followed by the name of a UNIX command will bring up that command’s man page in a more display format.

Most man pages provide the following information:

- **Name**: a simple definition of the command.
- **Syntax**: the correct way to type the command and its options and arguments.
- **Description**: a longer explanation of how to use the command and in what situations.
- **Options**: the options (characters or terms preceded by a hyphen) that may be combined with the command and what effect they have.
- **Restrictions**: known limitations on the use of the command.
- **See Also**: a list of cross-references to other commands that are related to or can shed light on the use of the command.

You may notice that the commands have a numerical argument, such as `zwrite(1)` or `chmod(2)`. This number refers to the type of command the argument is (see list in margin).

Most of the commands the general user will be referencing will be of the (1) User Command variety, so this is the default. Some commands, such as `chmod`, have multiple usages; `chmod` has man pages set up for `chmod(1)`, `chmod(2)` and `chmod(3f)`.

To look up a set of commands of a particular type, use man with the option `-k` for `keyword`, which lets you to specify a keyword to search for.

For more information on the man command, a logical place to look is `man man`, which displays the UNIX manual page on man itself.
**AFS File-Sharing Commands**

The Andrew File System (AFS) is a distributed file system that stores, organizes, and sets access to directories and files. The AFS guide opens with this introduction to working in AFS: “A distributed file system joins the file systems of individual machines. Files are stored (distributed) on different machines in the computer network. Information stored anywhere on the network is as accessible as information stored on your machine’s local disk. The AFS distributed file system hides the existence of the underlying network. Working in AFS is like working on the file system on your local machine, only you can access many more files.”

With AFS, Eos/Unity users do not need to know the physical location of the files or software they use. They simply request the files they need, and AFS finds and retrieves them automatically. AFS uses a client/server model whereby client machines access files stored on server machines, rather than from the local disk. In this model, files are downloaded to the workstation for work and computation and then stored out on server machines until needed again. This reliance on the power of the client workstations for computation means that the number of users on the AFS file system does not slow its performance appreciably, making Eos/Unity an efficient and widely scalable computing environment.

AFS also makes is easy for groups of people to share information and work together on the same files, no matter where they are located on the system. However AFS makes it just as easy to prohibit other users from accessing files. AFS uses both password authentication and special access control lists (ACLs) to ensure that the environment is a secure but flexible place for users to work, both privately and collectively.

**The Root Directory and /afs Cells**

Like Unix, AFS has its own terminology and uses a hierarchical tree-like file structure. Users will see the AFS root directory `/afs` in their file and directory pathnames (type `pwd`). The directories under `/afs` are cells. Cells are independently administered sites running AFS and consist of a collection of file servers and client workstations defined as belonging to that cell. A computer can belong to only one cell at a time. A cell’s administrators determine how workstations and servers are configured for that cell and how much storage space is available to each user. However, each cell can also connect with the file space of other cells running AFS. The result is a huge file space that allows file-sharing with people in every cell using AFS, whether these cells are located across the hall or across the country (see also Working Directories and Paths).
Partition, Volume, Quota, and Mount Point

The storage disks on a server are sectioned into disk partitions. These partitions are further divided into volumes, which are “containers” or sets of related files and directories. Most AFS cells use one volume for each user’s home directory. Volumes have a size limit, or quota, assigned by the system administrator. A volume’s quota, measured in 1 kilobyte (1024 bytes) blocks, determines the storage space allowed in the volume. Currently, individual volumes for users in both the Eos and Unity cells are set at 50 megabytes. When users exceed that quota, they will receive error messages. As a result, users should check their quotas often with the quota command (a system-wide alias for fs listquota).

Each user’s home directory (volume) is located on a one-gigabyte disk partition with many other users. The quota command shows both percent of volume and percent of partition used, e.g.,

<table>
<thead>
<tr>
<th>Volume Name</th>
<th>Quota</th>
<th>Used</th>
<th>% Used</th>
<th>Partition</th>
</tr>
</thead>
<tbody>
<tr>
<td>users.m.mcdaniel</td>
<td>50000</td>
<td>25000</td>
<td>50%</td>
<td>98%</td>
</tr>
</tbody>
</table>

The Used and % Used of volume are all that affect your account. The Partition does not affect your account, even if it is as high as 98%, which on a one-gigabyte partition still leaves 20 megabytes of disk space to use.

Access to a volume is provided through a mount point, which indicates the physical storage location of the volume. Your own volume resides on one of many file servers, and the mount point is the pointer that AFS uses to find and retrieve it for you.

Sharing Information (fs)

AFS makes it easy for users to share the information in their files and directories with others. **You should never give anyone the password to your account in order to share information!** AFS uses access control lists (ACLs, pronounced “ackles”) to determine who accesses information in AFS file space, as well as what actions they can perform.

Unix uses mode bits and the chmod command to determine the access and actions that can be performed on individual files (see man chmod). These “rights” appear in the left column when you do a ls -l of a directory.

<table>
<thead>
<tr>
<th>access permissions</th>
<th>links</th>
<th>user</th>
<th>size</th>
<th>date last modified</th>
<th>time last modified</th>
<th>file name</th>
</tr>
</thead>
<tbody>
<tr>
<td>–rwxr–xr–x</td>
<td>1</td>
<td>jqpublic</td>
<td>1298</td>
<td>May 8</td>
<td>10.03</td>
<td>hwk2</td>
</tr>
</tbody>
</table>

AFS augments this standard Unix file-protection mechanism with additional access control at the directory level. An ACL exists for each directory in the file system, and it specifies what actions different users
can perform on that directory and its files. Three Unix user `rwx` mode
bits—read, write, and execute—protect each file, but seven AFS access
rights protect the directory in which the file resides. The result is that
individual file permissions do not mean very much in AFS-defined
directory space (except for the execute bit `x`). Rather, it is through AFS
commands (fs and pts) that users assign and manage access to directories,
not file-by-file using Unix and chmod.

- Access Control List (ACL)

You set permissions for directory access in the access control list (ACL).
A directory’s ACL is a list of users and groups and the rights they have
to access and use the files in that directory, specifically, r—Read, l—Look,
i—Insert, d—Delete, w—Write, k—Lock, and a—Administer. The owner of
a directory (and anyone who has administer rights) can set and
manipulate the ACLs for a directory.

Access  Meaning

<table>
<thead>
<tr>
<th></th>
<th>r</th>
<th>read (and copy) the contents of files in the directory.</th>
</tr>
</thead>
<tbody>
<tr>
<td>l</td>
<td>look (not read access). Can list (ls) directory and look at its ACL. You must have l access to use other access rights, e.g., to read you must have rl.</td>
<td></td>
</tr>
<tr>
<td>i</td>
<td>insert files or subdirectories (create new files, move existing ones).</td>
<td></td>
</tr>
<tr>
<td>d</td>
<td>delete files or subdirectories from the directory.</td>
<td></td>
</tr>
<tr>
<td>w</td>
<td>write or edit the contents of files in the directory.</td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>lock. Sets an advisory lock on a file (not used often).</td>
<td></td>
</tr>
<tr>
<td>a</td>
<td>administer or change permissions in the ACL. Owner has administer rights.</td>
<td></td>
</tr>
</tbody>
</table>

Aliases have also been set up for common levels of access, i.e., read, write
and administer. These can be used in place of the letter abbreviations.

<table>
<thead>
<tr>
<th>Alias</th>
<th>Access</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>read</td>
<td>rl</td>
<td>read and look</td>
</tr>
<tr>
<td>write</td>
<td>rlidwk</td>
<td>all rights but administer</td>
</tr>
<tr>
<td>all</td>
<td>rlidwka</td>
<td>full owner’s permissions including right to administer. Be careful giving all rights. Use write instead.</td>
</tr>
<tr>
<td>none</td>
<td></td>
<td>remove all rights, e.g., fa sa directory username none</td>
</tr>
</tbody>
</table>

Once again, individual files do not have access permissions associated
with them. Using a file depends on the permissions that have been set for
the directory the file is in. As a result, if a file is moved to a directory
where the access permissions are different, the file will inherit those
new settings.

To make a file an executable:
chmod +x filename

To list all fs commands:
fs help
• Listing ACL Permissions

There are two general types of AFS commands: file server (fs) commands and directory protection commands (pts). To look at the access rights on your home directory (type cd to return to your home directory), type the file server command, fs la (file server listacl). By default, this command shows you the access control list for the current (.) directory, e.g.,

```
eos% fs la
Access list for . is
Normal rights:
    system:administrators rlidwka
    system:anyuser l
    mcdaniel rlidwka
```

You can also specify a path to a directory, e.g., `fs la ~/.elm` checks the access to your ~/.elm directory. The output above tells you that system administrators have full rights to administer your directory, just as you do as owner of the directory (your username would replace mcdaniel): r—Read, l—Look, i—Insert, d—Delete, w—Write, k—Lock, and a—Administer. Everyone else (system:anyuser) can “look,” that is, they can show a directory listing (ls) of your files and check the ACLs on your directories with the fs la command, but they cannot do anything else to your files (“look” does not mean “read”!). Only you and the system administrators are allowed to change these settings.

There are very few people with system administrator privileges, and they are carefully screened, full-time employees of the university computing staff. It is necessary for them to have access rights in order to assist you if you have problems with your account. It is not a good idea to change or remove the administrators’ permissions on your directories, although you can.

• Setting ACL Permissions

To grant someone access to a directory of yours, you must set access to it with the fs sa command (file server setacl). Use the following command syntax to set new access rights on a directory:

```
fs sa directory username access
```

where directory is the name of or path to the directory to which access is being granted, username is the login name of the person to whom you are granting access, and access is the permission being granted to username.

For example, if jqpublic wants to give joeuser full access rights to his ~/bin directory (except for administer rights), he would type the following at the prompt (if his current working directory is ~/bin).

```
fs sa . joeuser rlidwk  (or fs sa ~/bin joeuser rlidwk)
```
Or, he could use the alias `write` for `rlidwk`:

```
fs sa . joeuser write
```

To take away or remove these rights, jipublic would use the command

```
fs sa . joeuser none
```

Fewer rights can be given than these. If jipublic wants joeuser to be able to read and copy his files but nothing else, he would set `rl` permission on the directory (you have to grant `l` access before any of the other settings will work), e.g.,

```
fs sa . joeuser rl
```

Or, he could use the alias `read` for `rl`:

```
fs sa . joeuser read
```

**WARNING:** Remember that when you grant access to a directory, you also grant access to all new subdirectories created under it. Because new subdirectories inherit the ACL of the parent directory, be careful when you grant access to your home directory.

### Sharing Information with Groups (pts)

AFS permits only 20 users or groups for each directory. As a result, if you want to grant `write` access to a directory to a class of 25 students, you could not do so unless you put them in a group.

You will find two system-defined groups in every ACL you list. The group `system:administrators` contains a list of the computing staff who are authorized to administer your cell (eos.ncsu.edu, unity.ncsu.edu, etc.). The other is `system:anyuser`, which is a group through which you can grant or deny access to everyone. In other words, it is a designation for general public access. In most cases, you do not want to change the default access: `l` for `lookup`. However, in some cases, as in a World-Wide Web directory, you would change `l` to `rl` to make your files readable to everyone accessing them on the Web.

An ACL `group` is a defined list of individual users that you can place on the ACLs of your directories. Instead of adding and removing individuals separately, you can add or remove them as a group.

When you create a group, you automatically become its owner. You create a group with the `pts creategroup` command (or `pts cg`):

```
pts creategroup owner:group
```

where `owner` is your own username and `group` is a name you make up for the group. Most groups have these two parts: the part before the colon tells who owns the group, and the part after is the group’s name.

To nullify someone’s permissions on a directory:

```
fs sa dir username none
```

If you accidentally take away your own access rights:

```
attach yourusername
cd /ncsu/yourusername
fs sa . yourusername all
```

To create a group:

```
pts cg owner:group
```

To add a user to a group:

```
pts ad username owner:group
```

To remove a user:

```
pts rem username owner:group
```

To check the membership of a group:

```
pts m owner:group
```

To delete a group:

```
pts del owner:group
```

95
Groups that you encounter that do not have an owner “prefix” are special groups created by system administrators. All of the groups you create must have an owner prefix and a colon before the group name.

To add a member to the group, type

```
pts adduser username owner:group
```

(or `pts ad`), where `username` is the Eos/Unity username of the person you want to add, and `owner:group` is the name of the group you have created. This command places that user in the group. There is no restriction on the number of members in a group. To check who is in the group, type

```
pts membership owner:group
```

(or `pts m`). Other `pts` commands include `pts removeuser username owner:group` to remove a user from a group, and `pts delete owner:group` to delete a group. To get a full listing of `pts` commands, type `pts help`.

For example, with the following commands, jqpublic makes a `classproj` directory that she and three classmates (moe, larry, and curly) can all work in together. Jqpublic creates the group `jqpublic:projgroup` and adds the usernames, moe, larry, and curly, to it. She then adds this group to the ACL of the `classproj` directory with the `fs sa` command and gives the group “write” access (rlidwk).

```
mkdir classproj
cd classproj
pts creategroup jqpublic:projgroup
pts adduser moe jqpublic:projgroup
pts adduser larry jqpublic:projgroup
pts adduser curly jqpublic:projgroup
```

Or, to add all three users in one command:

```
pts adduser -user moe larry curly -group jqpublic:projgroup
```

You grant access for a group the same way you would for an individual:

```
fs sa . jqpublic:projgroup write
```

To check the membership of the group:

```
pts members jqpublic:projgroup
```

Members of `jqpublic:classproj` (id: -1234) are:

moe
larry
curly
**AFS on Windows**

The Windows workstations have only recently been brought in as full AFS clients, and the way they interface to the campus file system is different. Once again, nothing is command-line driven on Windows. AFS control is available from the main **File** menu of most files and folders on the system.

If you have used Windows computers before, you will not have seen **AFS** on the **File** menu. It is a special customization that was done to make the Windows platform fit into the Eos/Unity AFS infrastructure so that files could be accessed, moved, backed up, and shared easily.

The AFS information about files and directories (folders) can be brought up and changed via menus and dialog boxes on Windows. For example, the following figure shows the K: drive (home directory) of the user selected. Also selected is **File -> AFS** to show the various AFS options.

![AFS menu on Windows](image)

**Quota**

You can check your quota, or how much disk space you have on your K: drive (home directory), by selecting **Volume/Partition**.

![Quota dialog box](image)

To check quota or disk space on the Windows machines, select a folder or drive and right-click to bring up a pop-up menu:

**AFS -> Volume/Partition**

Wolfcall is the newest AFS client on Windows, see information at [http://www.eos.ncsu.edu/wolfcall/](http://www.eos.ncsu.edu/wolfcall/)
You can do this on any folder or drive to see how much disk space has been allocated and used. It is the same as typing the `quota` command on the Unix and Linux workstations, or `fs lsq` (`fs listquota` or `fs quota`).

**Access Control Lists**

You select **Access Control List** to change accesses to your directories. (Remember that in AFS, you set permissions on directories, not on individual files.) For example, to check the permissions on the `www` subdirectory in mcdaniel’s home directory (K: drive), select that folder and **File -> AFS -> Access Control Lists...** (or right-click the folder and select AFS options on the pop-up menu). The following dialog box appears and you can point and click to Add/Remove people to the ACLs and grant or take away their access permissions.

Select **Help** to bring up additional information about Access Control Lists. A Help system is available on the AFS client for Windows and can be brought up by selecting **Contents** or **Index**, depending on how you wish to search the help documentation (see figure on next page).

**AFS from the Command Prompt**

It is also possible to use the `fs` commands from the MS-DOS command prompt, available from **Start -> Programs -> Command Prompt**, or the **Application Launcher**. In the window that comes up at the `C:\WINNT\...` prompt, you can change to one of the AFS drives (e.g., K: or J:) and run your AFS commands from there. You simply type `K:` or `J:` and press **Enter** to change to that drive.
Use the `dir` command to list the contents of directories (for more help with MS-DOS commands, go to `Start -> Help` and search on MS-DOS). The AFS documentation (see figure above) has a chapter on Command Prompt Commands.

**AFS Glossary**

For more assistance with AFS, which defines much of the computing infrastructure at NCSU, consult the following glossary as well as the help resources mentioned in this chapter.

**access control list**: A list associated with an AFS directory that specifies what actions a user or group is permitted to perform on the directory and its files.

**acl**: See `access control list`.

**acl entry**: An entry on an ACL that pairs one user or group of users with specific AFS access permissions. An entry can be normal, granting the user or group the specific permissions, or negative, denying the user or group the specific permissions.

**afs**: See **Andrew File System**.

**afs uid**: An identification number assigned to each AFS user and group. It is guaranteed to be unique.

**Andrew File System**: A file service that joins the local file systems of several File Server machines. Files are stored (distributed) on different machines in the computer network but are accessible from all machines.

**authenticated**: The state of a principal whose identity has been verified by AFS.

**authentication**: Verification that a user or process is presenting a valid identity. Authentication involves certifying that a password provided by the user is correct.

**cache manager**: The portion of an AFS Client machine that communicates with AFS server processes by translating file requests made locally into remote procedure calls.
It stores the requested files in a cache on the local disk, from which it makes the files available to local users.

**cell**: An administratively independent site using AFS and consisting of a set of File Server machines and client machines. A machine can belong to only one cell at a time.

**file server**: A type of machine in AFS used to store files and transfer requested files to client machines.

**foreign cell**: An AFS cell other than the one to which the local (client) machine belongs. The local machine’s cell is referred to as the local cell.

**local cell**: The cell to which the local client machine belongs. Even though a user can authenticate in a foreign cell or fetch files from it, the identity of the local cell remains the same throughout a logon session.

**mount point**: A special type of directory that connects a location in the AFS filespace with a volume. A mount point looks like a standard directory; listing the directory shows the contents of the volume. Each mount point corresponds to a single volume.

**network drive**: A connection to the hard drive of a remote computer, allowing you to access shared files and directories. You can establish a network drive connection to a directory in the AFS filespace.

**partition**: A logical section of a disk in a computer.

**password**: A unique, user–defined string of characters that validate the user’s system identity. The user must enter the password to become authenticated.

**quota**: The size limit of a volume assigned by the system administrator and measured in kilobyte blocks.

**token**: A set of data that is granted after a user authenticates to AFS. A token is used by the Cache Manager when requesting services from AFS servers. A token has an associated lifetime and expires after a set period of time. If your token expires, you no longer have authenticated access to AFS. The standard token lifetime is 25 hours.

**unauthenticated**: The state of a principal whose identity is not known to AFS.

**username**: The name a user types in when authenticating that uniquely identifies the user in the local cell. It is mapped to the user’s AFS UID.

**volume**: A ”container” that keeps a set of related files and directories together on a disk partition that is specific to AFS.

**volume location server**: An AFS server process that maintains the Volume Location Database, which records location and other status information about all volumes in the cell.
SELECTED APPLICATIONS AND SERVICES
PRINTING

Most applications place a **Print** option on the **File** pull-down menu so that when users need to print their work, they just select **Print**. If there is a **Print...** option on the menu, then it can be selected to adjust the default print settings, e.g., number of copies, range of pages, etc. The ellipsis “...” after a menu option means that some kind of dialog box will come up with options and settings to select.

**Printing Facilities**

There is no printer connected directly to the individual workstations in the labs. Rather, printing is a remote network service on Eos and Unity. Every subnet has at least one printer that handles the printing requirements for a cluster of users. The Labs chapter lists the names and locations of all of the Eos and Unity lab printers.

WolfCopy Centers manage and maintain the public printers. Also, the operators in the labs are responsible for on-site management of printing facilities. Contact these people to help you. The printers are complex and expensive pieces of equipment that users should not try to fix themselves. To report problems, send e-mail or call **help@ncsu.edu** (515-HELP) or **wolfcopy@ncsu.edu** (515–2131).

**Print Quota (lpquota)**

When you are issued a computer account, you will need to go to a WolfCopy center on campus and purchase print quota before you can print on the printers in any of the labs. All platforms use the same print quota debit system. The centers are open 8:00 a.m.–5:00 p.m. for immediate additions to your print quota. Printing is $0.06/page, and the minimum quota purchase is $5.00 for about 83 pages.

After 5:00 p.m., you can still purchase print quota, but it will not be effective until the following day. Your print quota will be charged 6 cents a page, even if your quota drops below $0.00. Please check it often.

**Checking print quota on Unix and Linux:** You can check how much you have used of your print quota by typing **lpquota -q** at the command prompt in an XTerm window to list all your charges, or **lpquota -l** for a complete log of all the print jobs you have sent, including the time, date, number of pages, and the printer used. Or, use the web site, [http://print.ncsu.edu](http://print.ncsu.edu).
Printing Files (lpr)

Although most printing can be done from inside applications, there are times when you may need to print from the command line or without having to open the application. Most applications let you “print to a file,” which means that you can send your output to a file rather than to a printer. Then you can print the file whenever you want without having to open the application.

On Unix, to print a file that has been formatted for printing, generally a PostScript file (.ps), type `lpr` (line print) followed by the file name or the path to the file, i.e., `lpr filename` or `lpr dir/subdir/filename`. On Windows, right-click the file and select Print on the menu.

You can now browse to select a text, .ps or .pdf file and specify the printer to send it to from http://print.ncsu.edu, which is an easier cross-platform solution. You can also send the file to a printer from anywhere with an Internet connection, not just from a lab.

Previewing Files Before Printing

Before sending a job to the printer, check your file carefully to make sure that it is ready to print so that you do not waste your print quota. Many programs have a Print Preview capability, generally available from the File -> Print... menu. If you have a PostScript (.ps) or file that you want to look at before printing, you can view it with one of the PostScript viewers on the system, e.g., `gv` or `ghostview`. On Windows, you can also right-click a file and select Quick View from the menu (if available).

Printer Selection

The workstation you use will automatically route your files to a particular printer, usually a local one in your office or lab. However, sometimes, you can route print jobs to other printers by either (1) selecting the printer you want from the Printer options available inside the application, or (2) printing to a file and then sending the file to the printer you specify. The latter is done by adding `-P` after the `lpr` command followed immediately (no space) by the name of the printer you want to use (see printer names in Labs) and the specific print-formatted file you want to print.

For example, if you want to print to a printer in D.H. Hill library, you would need to specify that printer’s name, e.g., `lpr -Pdhl-2413-1`, and then the file you want to print, e.g., `filename.ps`. Be careful not to confuse the letter “I” and the number “1” (dhl ← letter, 2413-1 ← number).

On Unix, the default printer is set in the PRINTER environment variable (see The C Shell and Environment Variables). To find out your default printer, type `printenv PRINTER`. To change your default printer, type `setenv PRINTER printer`.

To print a file, usually with a .txt (text) or .ps (PostScript) extension:

```
lpr file.ps
```

PostScript: a page description language that describes the appearance of text, graphics, and images on printed or displayed pages.

PostScript and PDF files can be viewed in Ghostview programs.

To send a file to a specific printer:

```
lpr -Px file.ps
```

where `x` is the printer name

To find out your default printer:

```
printenv PRINTER
```

To change your default printer:

```
setenv PRINTER printer
```
typing `setenv PRINTER` followed by the name of the printer you want to use. When you want to specify a printer other than the default, use the `-P` option (see above) to override the default setting in your environment.

On Windows, the default printers are listed in **My Computer –> Printers**, but because of the high-security settings on the lab-installed workstations, users cannot add printers or change configurations.

Once again, you can find printers and perform many print functions from http://print.ncsu.edu.

### Print Queue and Removing Jobs (lpq, lprm)

If other people have also sent jobs to the printer you are using, you may have to wait awhile before your output comes out. If you want to see where your job has been placed in the list of jobs to be printed, or **print queue**, use the Unix `lpq` command.

Sometimes you send a file to a printer and then decide suddenly that you do not want to print it after all. Use the `lpq` command to list the print jobs in the queue. All jobs are numbered. To remove a job from the queue, type `lprm` followed by the number of the job (you can only remove your own jobs). If you have been quick enough to catch it, the `lprm` command will remove the file from the print queue and keep it from printing. The command `lprm -` removes all of your print jobs from the queue.

On Windows, print jobs are queued and deleted via **My Computer –> Printers**. Double-click the icon of the printer to see the queue, and then select your job and **Document –> Cancel**.

### Using the Network X Print Manager (nxpr)

NXpr (written by former NCSU student, Lance Lovette) is an interface to `lpr`, the Unix line printer command. It was created to help users conserve their print quota. It allows you to print files with smaller font sizes and several pages to a single sheet of paper so you can preview your output before printing the final job (also on Linux but not Windows).

Online help is provided inside the program. To open the Help index, select **Help** at the top right of the NXpr window. Then select **Index** to bring up a scrollable dialog box on the functions and topics in NXpr.

### Printing Selected Text from the Root Menu

Use the **Print Selected Text** option on the **Root Menu** to print selections of text you highlight on the screen. Drag over the text with the left mouse button to highlight it and then select **Print Selected Text**.

### Color Printing

The color printer **lez-100-color1** is available for use in 100 Leazar Hall. To print to this printer, you would first print to a PostScript (.ps) file or
Portable Document Format (.pdf) file; do not print to Encapsulated PostScript (EPS). Send the file to the printer with the Unix command:

```
lpr -Plez-100-color1 filename.ps
```

`lpr` is the print command, `-P` specifies output to a particular printer, and `lez-100-color1` is that printer. A color printer is also in the D.H. Hill Library Unity lab named `dhl-2413-color1`. Other color printers are listed at http://print.ncsu.edu.

### Printing Screens

To capture screens and windows for printing, use the `xv` application on Solaris and Linux. Bring it up from the Application Menu (Graphics -> `xv`), or by typing `xv` on the command line. Select the `Grab` button and follow the instructions provided to capture a window, area, or whole screen, and then save to a file for printing.

On Windows, to copy the entire screen as it appears on your monitor, press the **Print Screen** key on the keyboard. To copy only the active window, press the **Alt** and **Print Screen** keys at the same time.

To paste the image into a document, select **Edit -> Paste** in the application and then **Save** and **Print**.
Advice for Conserving and Sharing Print Resources

1. Accustom yourself to working on-screen rather than on paper. Read and edit online, and print final copies only. Send and share files via e-mail and AFS file-sharing commands rather than making paper copies.

2. Check files with preview utilities, editors and viewers before printing.

3. Only print pages that you have changed, not the whole document. This will also keep you and others from having to wait long for output.

4. Check your print settings carefully before you print. For example, if you have just changed a setting to print three copies of a document, make sure you change the print setting back to one before printing again.

5. Use the **NXpr Print Manager** and the **Print Selected Text** utility on the **Application** and **Root** menus in order to print only what you need and no more. NXpr allows you to print 2-4 pages per sheet of paper and also keeps track of your print quota. **Print Selected Text** prints only what you have highlighted on-screen (but only in ASCII text).

6. Do not use the printer as a copier. Copying costs as little as 3 cents a page; printing is 6 cents a page.

7. Remember how to cancel print jobs. Type **lpq** to show the print queue, find your print job, and then remove it with the command **lprm job#**. On Windows, select **My Computer -> Printers -> Document -> Cancel**. Or, use the facility for checking and removing jobs at http://print.ncsu.edu.

8. Turn off header and banner pages.

9. Be considerate of other users. Break up big print jobs into several shorter jobs so you do not monopolize the printer.

10. If you are not in a hurry, choose times when the lab is not busy to print large documents.


**Electronic Mail**

Electronic mail, or *e-mail*, is used to send and receive mail to/from anyone on Eos/Unity, or on other campus, national or international networks to which Eos/Unity is connected via the Internet.

***IMAP Mail Access and Programs***

(From [http://www.ncsu.edu/imap/](http://www.ncsu.edu/imap/)) IMAP is the mail protocol NCSU uses for all accounts created in or after Fall 2000. Nearly all students are now on IMAP. It has several key features that are different from the previously used POP protocol. The most notable of these is Webmail, a portal much like Hotmail or Yahoo mail that allows users to access their mail from any up-to-date browser on any personal computer with Internet access. Since your mail stays on the mail server rather than being downloaded to your Unity space, IMAP is both location and platform independent, so you can get to it from anywhere.

Programs like Netscape Mail can be set up to read mail from the IMAP server, but using Webmail is the easiest alternative. All users are allotted a free IMAP quota of 30 MB. This is in addition to the user’s Unity account quota of 50MB. For a fee, users can request additional IMAP quota up to 250MB.

Webmail ([http://webmail.ncsu.edu/](http://webmail.ncsu.edu/)) provides the following features:

- You can read and send email, including most attachments, from any supported browser on the Web.
- You can sort and store your email in separate folders (called mailboxes in Webmail) for easier organization.
- Your connection to the Webmail server is encrypted, so your password and the content of your messages will remain secure as long as the computer you are using is secure.
- You can create an address book to organize your email addresses or import an address book from another source.
- You can create and include a signature file for more personalized messaging.
- Webmail works best when used with newer versions of Netscape and Microsoft Internet Explorer. It is recommended that you use Netscape 4.6 or higher or IE 5.0 or higher.

**New!** You no longer have to add “eos” or “unity” to your address, e.g., jqpublic@unity.ncsu.edu. All users can now use their Unity ID followed by @ncsu.edu for their mail address, e.g., jqpublic@ncsu.edu.

**New!** Squirrel Mail is the new IMAP Webmail beginning fall 2003.

Standard mail access is through:

- [http://webmail.ncsu.edu/](http://webmail.ncsu.edu/)
- Help is at:
  - [http://www.ncsu.edu/it/essentials/learn_more/a_z_tutorials/webmail/](http://www.ncsu.edu/it/essentials/learn_more/a_z_tutorials/webmail/)

Users can pay to augment their IMAP mail quota:

- [http://www2.ncsu.edu/cc/pub/cc_forms/imapquota.html](http://www2.ncsu.edu/cc/pub/cc_forms/imapquota.html)
Mail in Windows labs: The lab configuration is set up for IMAP only and uses Netscape IMAP Mail (Messenger) and Webmail.

Mail in Solaris/Linux labs: If you are on IMAP and want to read mail on Solaris/Linux platforms, use Netscape and http://webmail.ncsu.edu/, or the pine or mutt text-based (line-mode) programs.

Note to POP Users

NCSU is changing from POP (Post Office Protocol) to IMAP (Internet Message Access Protocol). Most users want IMAP-type capability, which allows them to send/receive mail and manipulate their stored messages from multiple computers and locations. They also want to use a Web browser for their access rather than the location-dependent and proprietary mail programs that POP uses. POP works best if the user stores messages in one place only and uses one mail program. If you are a POP user (likely faculty, staff, or graduate student), it is very important to pay attention to the changes that are being made. IMAP and POP are completely incompatible, and you have to be very careful moving between platforms.

Campus users already on POP switch to IMAP voluntarily when they are ready to change (see http://www.ncsu.edu/imap/howmove.html). Once the change from POP to IMAP is made by the appropriate system personnel, users cannot use the POP mail programs they used previously, nor can they switch back to POP. It is an all-or-nothing changeover. Because IMAP is completely incompatible with POP, it has caused problems for users who have not fully understood what a move to IMAP entails. Please read the Web pages at http://www.ncsu.edu/imap/.

Mail in Windows labs: The Windows lab configuration is IMAP only. However, if you are on POP and want to read mail in the Windows labs, you can make a remote-access connection, log in, and use a command-line program like elm.

Mail in Unix labs: POP mail in Solaris/Linux labs is read with elm.

Preferred Mail Addresses

Many users have mail accounts through AOL, HotMail, and others. Your mail can be sent to these other mail providers if you prefer. However, remember that you will be on campus for a long time, and you may wish to consider the benefits of a free and long-term campus mail account. You get more quota and can send and receive larger mail attachments than you can with a service provider. Plus, you receive local support.

NCSU supports only one mail address per user, which is the Unity mail address unless the user specifies an alternate preferred mail address (not...
a forward). In other words, users can use either the Unity address (userid@unity.ncsu.edu, soon to be userid@ncsu.edu for everyone) or an alternate one, but you cannot set it so that mail sent to the Unity ID forwards to the alternate address.

Students set their preferred email addresses at:

https://www.acs.ncsu.edu/reg_records/tracs_lk/trc_frm.html

Faculty and staff set their preferred email addresses at:

https://ncsweb1.fis.ncsu.edu/dirupdate/

**Forwarding Mail**

The Unix mail .forward file mechanism is not supported on Eos/Unity and will not work. However, campus ITD offers short-term e-mail forwarding to students. Send your request to accounts@ncsu.edu with the complete Internet address to which you want your e-mail directed. Write to this same address to turn off mail forwarding. Forwarding is available to everyone, but is limited to four months for undergraduates. Faculty and staff request forwarding in the same way but can have mail forwarded for longer periods of time. **Important! Mail cannot be sent to two addresses. If your mail is forwarded, you can no longer access it via the mail services and software on Unity.**

**Using Mail**

In the window of your mail program, type the username(s) of the person(s) you want to write on the **To:** line. When you write to more than one person, add usernames with a comma between them (jqpublic, iamuser, uruser). You can add a space after the comma or leave it out. Write a clear description of the message on the **Subject:** line, and then type your message in the text buffer at the bottom of the window.

To send mail to someone on Eos/Unity from a campus computer, all you need to type on the **To:** line is the person’s username, e.g., jqpublic. However, someone not on campus who writes to jqpublic must use the whole address with hostname, which is jqpublic@ncsu.edu. **Beginning this year, you do not have to add “eos” or “unity” to the mail address, e.g., jqpublic@unity.ncsu.edu. All users can use their Unity ID followed by @ncsu.edu for their rmail address.**

If you want to write someone who is not on Eos/Unity, e.g., someone in another university, company, or country, you need to use that person’s username and complete hostname. To write you back, they also need to use your complete address.

To automatically affix a signature to your mail messages, that is, a body of text information about yourself (name, address, telephone, etc.), create
a `.signature` file in your home directory. Some programs are set up to read `.signature` files automatically and to append any information found in them to the mail messages you create.

For help with campus mail:

- `http://www.ncsu.edu/it/antivirus/`

**Server setup for IMAP:**

The IMAP Server (Incoming) host is:

`userid.mail.ncsu.edu`

where `userid` is your Unity login ID, e.g.,

`jqpublic.mail.ncsu.edu`

The SMTP Server (Outgoing) host is the SMTP server provided by your ISP.

The on-campus SMTP server is smtp.ncsu.edu.

Get anti-virus software at:

`http://www.ncsu.edu/it/antivirus/`

---

**Outgoing Mail Relay**

The preferred name to use for the on-campus SMTP relay is smtp.ncsu.edu. Off-campus users will need to change their outgoing SMTP server to their local ISP’s SMTP server.

**Viruses and Spam**

Most viruses find their way to your account via e-mail and e-mail attachments. NCSU purchases Symantec’s Norton AntiVirus software and makes it available at `http://www.ncsu.edu/it/antivirus/` for download by students, faculty, and staff to their on-campus and home computers. Users can install either a managed version, which automatically updates virus definitions, or the unmanaged version, which requires the user to initiate updates. Virus information can be found at `http://symantec.com/`.

Spam is unwanted “junk” e-mail that we all receive. It is difficult for the campus to filter mail for all, so individuals need to do what they can to protect themselves. The following advice may help:

- Do not become a spammer yourself by broadcasting mail, see Policies.
- Do not reply to spammers. Report abuse to `http://spam.abuse.net/`.
- Never “unsubscribe” to spam mail. Offering you the opportunity to unsubscribe is a ruse to see if your mail account is alive and therefore valuable to other spam lists the spammer may sell to.
- Don’t post your e-mail address on Web sites.
- Keep a Hotmail or Yahoo e-mail account for personal subscriptions, newsgroups, etc. Use your Unity e-mail for academic purposes only.

---

110
WolfWare Course Lockers

WolfWare Web-based Course Management was conceived and built by computing staff in the Department of Computer Science, COE Information Technology and Engineering Computer Services (ITECS), and PAMS. Help and feedback came from Registration and Records (R&R), CALS, CHASS, and the 1999-2000 beta users. WolfWare was released for general campus use in fall 2000 and is now developed and supported by NCSU’s Computing Services/Information Technology Division and Learning Technology Service.

What is WolfWare?

WolfWare is a Web-based course-management environment that simplifies the development and delivery of online course content. As a campus-wide system of automatic course-locker creation, WolfWare replaces the several manual systems used to house online content, some of which pre-date the Web. Faculty and students expect to focus on course content, not computing, so WolfWare works to keep computing infrastructure in the background while making it do more to support the teaching of both on-campus and distance learners.

Built within the Eos/Unity computing environment, WolfWare leverages AFS to scale a common system of uniform course-locker creation and management across all courses taught at NCSU. WolfWare lockers share a common directory and file hierarchy that facilitates the automatic placement of information into them and the delivery of course content out. The Web browser is the graphical interface to both the instructor side of WolfWare, where courses are set up and administered, and the student side, where students access their courses’ online content and tools.

WolfWare uses course-catalog and Registration and Records information to automatically create a boilerplate home page for every course so that all courses have a presence on the Web. This URL is the initial point of entry to the course locker with links to every active section. Like a classroom, it is the location of class collaboration and interaction online. Class administration and support facilities are pushed out to faculty and students through these course URLs, saving users the effort of hunting through the campus Web for the information they need.

WolfWare automatically generates rolls and class mailing lists and also provides for discussion forums, a secure homework-submission space, a

WolfWare course lockers are located at:
http://courses.ncsu.edu/

The project site is:
http://wolfware.ncsu.edu/

Help can be found at:
http://wolfware.ncsu.edu/help/

See if the course you are teaching or taking is in WolfWare:
http://courses.ncsu.edu/
place to pick up graded assignments, and secure (WRAP-ed) Web space. However, determining who can access the class Web is up to the instructor. Other valuable features include a development “PREP” area, where instructors prepare classes for upcoming semesters, and a flexible “PARK” area, where each course is archived for possible use by the instructor in a later semester.

WolfWare is platform and browser independent. Faculty can also use any standard content-creation tools to create and modify Web pages and place them within WolfWare’s framework: Dreamweaver, Front Page, WebCT, text editors, etc. WolfWare is not itself a content-creation tool but rather a course-management tool to support and deliver the content that the instructor assembles.

Using WolfWare

WolfWare course lockers are listed at http://courses.ncsu.edu/ Students access the locker contents and tools through the Links column of the course boilerplate page at http://courses.ncsu.edu/xxxxyy/. At this time, there are five links possible: Home Page, Message Board, Submit Assignments, Retrieve Assignments, and WebAssign. To access/use any of these, the student locates his/her class in the Course-Section column and then selects the appropriate link from the Links column (see the E115 boilerplate page below).

The URLs for courses and course sections follow:

Course location: http://courses.ncsu.edu/e115/
Web site for all course sections: http://courses.ncsu.edu/e115/common/
Section Web site (lecture): http://courses.ncsu.edu/e115/lec/001/
Section Web site (lab): http://courses.ncsu.edu/e115/lab/001/
Section secure space for both lecture and lab (requires login): http://courses.ncsu.edu/e115/lec/001/wrap/ (lec or lab)

If you select the Message Board, Submit Assignments, or Retrieve Assignments link, you will have to authenticate to secure Web space since access is restricted. Enter your Unity name and password. Home pages may also be restricted, but that is left up to the instructor, who may place course content in WolfWare space or at another location altogether. Most course content is delivered through the Home Page link. The other links (Message Board, Submit/Retrieve Assignments, WebAssign, etc.) take you to other tools or utilities the course uses.

WolfWare lockers are set up automatically with downloaded rolls from Registration and Records. Professors also receive the preferred email addresses of all the students in their classes and use automatically generated class mail lists to communicate with them, e.g., e115-001@wolfware.ncsu.edu.

You may not write to the class e-mail address unless the instructor sets it up for you to. Default class mail is one way, from the instructor to the class. Professors are not able to change your email in WolfWare since the data is downloaded to them. If you want to make any changes to your preferred email address, you must do so through:

https://www.acs.ncsu.edu/reg_records/tracs_lk/trc_frm.html

You should consider the benefits of using your Unity email address, username@unity.ncsu.edu, rather than that of a service provider. You will have it for four years as well as four months after graduation. Service is available anywhere, anytime via http://webmail.ncsu.edu/. Your mail is backed up nightly on university mail servers; plus, your quota and the size of mail attachments you can send/receive are larger than most service providers provide.

**Message Board**

WolfWare uses Message Board for class discussion. The instructor sets up discussion Forums, and students participate in the online discussion by posting topics and replying to posts within these forums. Put simply, faculty create Forums, and students create Topics, although faculty may define other ways to use the tool.

For example, in the History 270 class below, the instructor has set up Forums to discuss both a subject area, History of the Middle East, and a specific class, the May 29 Lecture. To participate in the May 29 Lecture forum, students select it, and on the Web page that comes up can post a
new topic, use the **Search Topics** to search and locate information by key word, or select a topic link to enter the discussion.

<table>
<thead>
<tr>
<th>Forum</th>
<th>Posts</th>
<th>Last Post</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>History of the Modern M.E.</strong>&lt;br&gt;This is the discussion forum for the internet course on the history of the Modern Middle East</td>
<td>156</td>
<td>Wed Jun 27 12:46</td>
</tr>
<tr>
<td><strong>May 29 Lecture</strong>&lt;br&gt;Cultural Encounters</td>
<td>26</td>
<td>Fri Jun 1 21:15</td>
</tr>
</tbody>
</table>

Select Forum link to bring up list of Forum Topics.

<table>
<thead>
<tr>
<th>Topic</th>
<th>Originator</th>
<th>Replies</th>
<th>Last Post</th>
</tr>
</thead>
<tbody>
<tr>
<td>Syrian Women and the Vote in 1920</td>
<td>userid1</td>
<td>4</td>
<td>Fri Jun 1 21:15</td>
</tr>
<tr>
<td>Balance of Power</td>
<td>userid2</td>
<td>5</td>
<td>Fri Jun 1 21:04</td>
</tr>
<tr>
<td>Islamic History Science</td>
<td>userid3</td>
<td>3</td>
<td>Fri Jun 1 17:39</td>
</tr>
</tbody>
</table>

Select **Post New Topic** to create a new topic. Or, select a topic link to go to a new page where the topic is discussed and a **Post a Reply** option is offered.

**Submit Assignments**

**Submit Assignments** is a tool for the secure submission of student files into the course locker. Assignments are added when the instructor uses the Sudmit Admin tool to name the assignment, set inclusive time/dates for assignment submission, and identifies the users who can submit. When the instructor sets up an assignment for submission, it will appear as a link on the boilerplate course page in the correct section’s **Links** area.

Students select this link to go to the submission page (the system lets through only those students enrolled in the class and able to submit). Once there, students see a list of assignments they can submit to. If the assignment cannot be submitted to, it will say CLOSED. If open, the
student will be able to select it and go to a page like the one below. A **Browse** button appears that allows the student to locate the file on their local or network drives. Once the file is selected (it must appear in the field beside the Browse button to be selected), the student selects **Upload File** to transfer the file. Any type of file can be submitted to the locker, but a folder or multiple files would have to be zipped or tarred first before submission as a single file.

The file is uploaded to the secure *submitted* directory that is automatically created by WolfWare when the locker is generated.

The AFS location of the student’s submitted file is:

`/afs/eos/courses/xxx/yyyy/lec/zzz/submitted/assignment/userid/file`

The Submit tool creates the assignment directory (e.g., *Homework3*), and all students who submit files to this assignment will have directories created for them inside this directory, e.g.,

`/afs/eos/courses/ww/ww101/lec/001/submitted/Homework3/mcdaniel/webprog.doc`

Students cannot delete files once submitted, but they can overwrite them by submitting a file of the same name. If the upload is successful, students are returned to the page showing a listing of all the files they have uploaded, the date of the upload, and the size of the file as saved in the locker (see above). If the instructor chooses to grade and return assignments through WolfWare, students can pick up their returned work via the **Retrieve Assignments** link on the course boilerplate page.
**WORLD WIDE WEB**

The Internet is a worldwide complex of computer networks that universities, companies, and governments use to exchange information. The World Wide Web, called the “Web,” is an Internet client-server system for distributed information retrieval that organizes the information on the Internet and links it together as *hypertext* documents.

Users, or clients, run *web browsers* to display these hyperlinked documents on their computers. Pointing a mouse and selecting a hyperlink (highlighted or underlined text and images) retrieves the document that is referenced by that link. Even if the referenced document is on a computer on the other side of the world, the browser will be able to retrieve and display it on the client computer. These documents are written in *HTML*, or *hypertext markup language*, and are “served” to the Web from computers that have been specially set up as web servers.

The Web has become a very important medium for instruction and communication at NCSU. After e-mail, the most frequently used application on Eos/Unity is the web browser.

**The Web Browser**

A browser is a graphical application that allows you to navigate the Web and access its resources. *Netscape Navigator* is the principal supported web browser on Eos/Unity. On Solaris/Linux, it can be brought up from the Application Menu or from the command prompt by typing `netscape &`. On Windows, Netscape is in the Application Launcher and can be launched by double-clicking the icon for it. The Windows platform also has Internet Explorer (IE) available from the Start –> Programs menu. IE is not on Solaris or Linux.

Both browsers automatically open to the NCSU home page when launched (see following figure). The web page that comes up automatically can be changed to something else you prefer via Edit –> Preferences. A home page is a starting place for accessing a body of related hyperlinked information. The university home page connects you to many online campus information resources: university catalogs, handbooks, calendars, phonebooks, maps, and more.

Most of the campus pages are best viewed with current browsers, so if you are running Netscape or IE from a home computer, you should make sure that the version of your browser is a recent one.
Using the left mouse button, click on words, phrases, and images that “link” to other web pages of information.

If you click on the hyperlinked text For Students on the home page, you will access an index page that lists principal campus information links of interest to students. Selecting Computing Resources will take you to other computing links. Computing Essentials is the home page for campus computing, http://www.ncsu.edu/it/essentials/, where you can find many resources to help you, including links to the College of Engineering Eos web site, http://www.eos.ncsu.edu (see following figure of these home pages). At the top of the browser, you will find the web address for the page, the Uniform Resource Locator, or URL.

If you know the URL of a web page you want to look up, you can simply delete the address in the Location window, type in the address you have, and press RETURN to go to that URL address.

Creating an HTML document goes beyond the scope of this manual, although a list of HTML tags is provided at the end of this chapter. There are many guides to help you make web pages. Recently added to the software suite on Eos/Unity is the popular Windows program, Dreamweaver, by Macromedia. It is a graphical web editor and site manager for developing pages for the web. Also available are Macromedia Fireworks for web graphics, and Flash for web animation. However, after you have created web pages with these tools or others, you will need to know where to put them in order to make them accessible on the Web. The remainder of this chapter explains this process, with information provided by ITD.

URL: Uniform Resource Locator. A URL is text used for identifying and addressing an item in a computer network. A URL provides location information for World Wide Web documents. Try out the Macromedia web suite on the Windows platform: Dreamweaver, Fireworks, Flash. For creating and working with PDF files, use the full Adobe Acrobat, also on the Windows lab platform.
The Computing Essentials web site for general Unity computing information.
http://www.ncsu.edu/it/essentials

The Eos Computing web site for both general and engineering-specific computing information.
http://www.eos.ncsu.edu
Publishing Your Web Pages

You may publish your own web pages on the www4 server provided by NC State Computing Services. To get on the Web, you create a web page (also known as a home page) that is coded in HTML. Then, you set the access rights on the page’s directory in AFS so that it is world-readable.

Computing Services has an electronic guide that covers HTML and access rights. This guide is on the Web at the following URL:

http://www.ncsu.edu/it/pub/orientation/webservs/

Creating a ~/www Directory and Setting Access Permissions

Because your Unity/Eos account allows only you, the owner, to look at and read the files inside it, you must specifically grant the rest of the world access to your files in order to publish your HTML documents.

To give the world access to your HTML files, do the following while in your home directory (~). (Type cd at the prompt to return home.)

1. Grant lookup permissions to your home directory (fs sa . system:anyuser l).
2. Create a subdirectory in your home directory called www (mkdir www).
3. Change directories to your www directory (cd www).
4. Grant lookup and read permissions to your www directory (fs sa . system:anyuser rl).
5. Create your HTML files in the www directory, or move them there.

Below are step-by-step instructions on how to do these things.

1. Grant lookup permissions to your home directory.

First, you are going to alter the AFS file permissions for your home directory so that system:anyuser (which is everyone in the world) has lookup (l) access (see also AFS File-sharing Commands). Think of your directories as boxes inside each other. Before others can lookup and read what is in your www box, they have to be able to lookup or open what is in your home directory box first. Be aware that when you grant lookup access to system:anyuser, anyone will be able to see the file and directory names in your home directory. They won’t be able to read the files or move into subdirectories, just see the names.

Here is the syntax for the fs command:

    fs sa directory [access list entries]

And here is what you should type:

    cd
    fs sa . system:anyuser l

**Explanation:** The cd command returns you to your home directory from wherever you may be on the system. The fs stands for file server; the sa
stands for set access. The period means the current directory (the one you are in). The name system:anyuser means any person, and the $l$ means that anyone can lookup file and directory names.

2. Create a subdirectory in your home directory called www.

Create a directory called www (mkdir www). All of your HTML documents that you want to let others view will reside in this directory. An automatic procedure runs nightly to find www subdirectories in users’ home directories and to automatically post their contents to the Web. This allows people to put their materials on whenever they wish without involving system personnel.

3. Change directories to your www directory.

Move into the www directory by typing the command cd www.

4. Grant lookup and read permissions to your www directory

Once inside the www directory, you are ready to change its permissions to grant read access to system:anyuser.

```
fs sa . system:anyuser rl
```

**Important**: Never give system:anyuser more rights than read and lookup. You must grant lookup privileges for read to work (or any other rights), and you must have system:anyuser $l$ granted in your home directory.

5. Create your HTML files in the www directory, or move them there.

Use the mv command to move your HTML files into the www directory. The syntax for the move command is:

```
mv [what] [where]
```

For example, to move the file test.html from your home directory to the www subdirectory, type the following while in your home directory.

```
mv test.html www
```

**Warning**: If the destination directory does not exist (in this case, the www directory), the mv command will rename the file www (see also Common File Commands).

**Shorthand Access to Your URL**

All users are on the www4 server, so the http://www4.ncsu.edu/ part of the URL is the same for everyone. The URL (address) for your page follows this syntax and viewable from the web browser:

```
file:/afs/unity.ncsu.edu/users/x/userid/www/
or file:/afs/unity.ncsu.edu/users/j/jqpublic/www/
```
Most people do not want such a long URL address. It is for this reason that the above procedure was created. If everyone has the same path to their web files, the procedure can be automated and an alias created. The shorthand way to reference your pages on the www4 server is:

http://www4.ncsu.edu/~jqpublic

The ~ stands in place of the full directory path, assuming that your web pages are in the www subdirectory in your home directory and that the initial page to display is index.html.

Web servers are set up to display an index.html file automatically if one exists in the directory. If an index.html is not present in the directory, then you must specify the file explicitly in the URL, for example, http://www4.ncsu.edu/~jqpublic/home.html

Moving Documents to/from Home and Non-Realm Computers

If the files you want to publish are not on computers within the AFS Eos/Unity realm, you must put them into “live space” via the ftp server (see File Transfer Protocol). Use the Netscape browser to make the ftp connection and download/upload your files to AFS space. Open URL:

ftp://username@ftp.ncsu.edu

Supply your password when prompted. To upload a file, choose File –> Upload File from the menu. You download by just clicking on the file.

Responsibility for Web Pages

In creating documents for the Web, remember that your information is being served from university computers. Both your username and your URLs bear the university’s name. For this reason, please make sure that what you present in your pages is appropriate, since the content and use of your pages are subject to university policy (see Policies). The intent of the university in providing these facilities and encouraging student use of the Web is educational. Please keep your activities within that arena so that these facilities remain available to others in the future with minimal need for supervision and restriction.

Creating HTML Documents

Although this guide does not offer a full explanation of how to write HTML documents, a list of HTML tags follows that will help you as a reference after you learn the basics.

See these web sites for additional help in creating web pages:

http://www.ncsu.edu/it/edu/html_trng/html_intro.html
A document is generally set up on a page with these minimal elements:

```html
<HTML>
<HEAD>
<TITLE> </TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

The whole document is placed inside the `<HTML></HTML>` tags. Make sure that there is a clear, descriptive title between the `<TITLE></TITLE>` tags, which is placed inside `<HEAD></HEAD>`. Between the `<BODY>` and `</BODY>` tags, you create your document. (It does not make any difference if you type tags in uppercase or lowercase letters.) Also, make sure to “sign” and date your pages, that is, put your name on your pages and the date when they were created and/or updated.

For help in creating pages, you may want to explore the following public-domain lockers:

- add www
- add imagetools
- add java

Some HTML tags:

- `<address></address>` Address or signature for a document, italicized
- `<a href=" ">` Hyperlinked reference, e.g., `<a href="reffile.html">link</a>`
- `<b>` Bold text
- `<blockquote></blockquote>` Quoted passage
- `<body></body>` Document body
- `<br>` Line break, single spacing
- `<cite></cite>` Name or title of cited work
- `<code></code>` Short words or phrases of source code
- `<dd>` Definition of a term in a definition list
- `<dl></dl>` Definition list or glossary
- `<dt>` Term in a definition list
- `<em></em>` Emphasized phrase
- `<h1></h1>` Heading, level 1
- `<h2></h2>` Heading, level 2
- `<h3></h3>` Heading, level 3
- `<h4></h4>` Heading, level 4
- `<h5></h5>` Heading, level 5
- `<h6></h6>` Heading, level 6
- `<head></head>` Document head
- `<hr>` Horizontal ruled line
- `<html></html>` HyperText Markup Language Document
- `<i>` Italic text
- `<img>` Link to image, e.g., `<img src="file.gif">`
<li> List item, used with ordered <ol>, unordered <ul>, or menu <menu>
<menu></menu> Menu list
<ol></ol> Ordered or numbered list
<p> Paragraph, double spacing
<pre></pre> Preformatted text
<strong></strong> Strong emphasis
<title></title> Title of document (Important! Every document needs one.)
<ul></ul> Unordered or unnumbered list, with bullets
SSH AND PuTTY

To connect to other computers on the Internet, you typically use ftp, telnet, or ssh. These protocols permit authorized users to log in to a multi-user computer from another computer over a network. ftp permits you to transfer files between your computer and another but not to view files or run programs on the other machine. telnet and ssh allow you to view files and run programs remotely but not to transfer files. As a result, you need to determine whether your task is to move files (ftp) or work on them (telnet/ssh) in order to choose the appropriate connecting protocol.

Telnet

Telnet is the Internet standard protocol for remote login over TCP/IP, and many users have experience using it. However, because it is an older protocol with minimal security, telnet has been turned off on most NCSU servers. You will need to use the ssh methods described below to connect to campus servers.

Secure Shell SSH

ssh is a Unix shell program for logging in and executing commands on a remote computer. It provides secure encrypted communications between hosts over a network and substitutes for telnet on NCSU systems. With ssh, users must prove their identity to the remote machine using some authentication method. On Eos/Unity, users authenticate with their Unity IDs to campus ssh servers. Once authenticated, ssh permits users to log in from remote systems and run programs on Eos/Unity computers. To connect to a remote-access computer:

```
ssh node.site.domain
```

For remote connection, use PuTTY, an icon in the NAL of the Windows platform. Or, Start -> Run -> putty login.ncsu.edu

SSH on Windows via PuTTY

Recommended remote access on Windows is through PuTTY, a free ssh client and X-term terminal emulator for 32-bit Windows systems. PuTTY launches a terminal session whereby your computer emulates a VT100 terminal to another machine so that you can enter and receive data from it. PuTTY can be launched from the NAL Unity Applications.

telnet is being turned off across campus. Use ssh.

terminal: an electronic or electromechanical device for entering data into a computer or communications system and displaying the data received.
The PuTTY Telnet/SSH client is installed on Eos/Unity Windows computers so that users can connect to and run Solaris and Linux applications from the Windows platform. PuTTY can also be downloaded to computers not on Eos/Unity, including home computers. With PuTTY, users can access and work in command-line (text) mode with Solaris/Linux applications.

The X-Win32 application for Windows, also available in the NAL Engineering Applications and for download, makes it possible for users to run Solaris/Linux applications in full graphical mode on the Windows platform. Users work through the GUI interface just like they were sitting at a workstation.

Users with a valid Unity ID can download PuTTY and XWin32 to their home or non-Eos computers in order to run campus applications remotely. NCSU users can access applications through the remote-access servers, remote.eos.ncsu.edu and remote-linux.eos.ncsu.edu. Those who connect to these resources are asked to run only one application at a time to minimize the load on the servers.

The window below is a login to one of the Eos remote-access servers, which appear as icons if you download them. Note the message for use of these resources.

---

You have logged in to NCSU College of Engineering remote-access servers, which provide computing resources to faculty and students working remotely. These facilities are shared and limited, so users are asked to run only one application at a time, if possible (no additional browsers, games, etc.). Please use LSF rather than these servers for batch processing and other long jobs (see http://www.eos.ncsu.edu/software/lsf/).

How you use these servers affects other users logged in to them as well, so keep to a minimum the compute-intensive applications and jobs you run in a session. For more information on remote access, see http://www.eos.ncsu.edu/remoteaccess/.

Contact: eoshelp@eos.ncsu.edu

---

For the latest Eos/Unity news, information on reporting system problems and getting help with the system, type "sysnews". For the Eos/Unity usage policy, type "policy".

Be sure to change your password at least once a semester. Choose a password with a mix of upper and lowercase characters, numbers, or other special characters. Password changes may not take effect for up to 30 minutes after requesting the change.

---

Download PuTTY and X-Win32 for home and office use:
http://www.eos.ncsu.edu/remoteaccess

For more information on PuTTY and X-Win32:
http://www.eos.ncsu.edu/software/putty/
http://www.eos.ncsu.edu/software/xwin32/

For remote-access information, www.eos.ncsu.edu/remoteaccess
File Transfer Protocol (ftp)

To ftp to another computer:

    ftp ftp.node.site.domain

On Windows, go to Start → Run and then ftp hostname. Or use SmartFTP in the labs.

Use ls, cd, pwd, etc., to manipulate files and directories. ? for help, * as wildcard, etc.

Type quit to exit.

To transfer a file:

    get file:
    put file

To transfer multiple files:

    mget file1 file2 file3
    mput file1 file2 file3

To ftp multiple files without prompting:

    prompt
    mget * (or mput *)

To transfer a binary file:

    binary
    Then, get or put file.

To return to text mode:

    ascii

File Transfer Protocol, or ftp, enables you to connect to another computer on the Internet and transfer files and data to and from it. This service is sometimes called anonymous ftp when server access is set to permit anyone to log in, usually using the login name “anonymous.” Generally speaking, the password is your e-mail address, which allows site administrators to contact you if they need to.

Ftp sites permit this kind of login because their purpose is to make information, software, and documents available to connecting users. As a result, they try to make it as easy as possible for you to get and put materials back and forth. There are also many ftp programs available for download from the web from http://download.com and other sites. Some programs like Dreamweaver have ftp built in.

The drawback of using ftp is that it only permits you to browse directories and transfer files. Unlike telnet/ssh, you cannot read anything on the host computer or run applications there.

Ftp-ing to/from Your Eos/Unity Account

To ftp to your account and get a file, type

    ftp ftp.ncsu.edu

When you connect, you will get something like the following message:

    unity% ftp ftp.ncsu.edu
    Connected to ftps.unity.ncsu.edu.
    220 North Carolina State University - Unity FTP Server
    Name (ftp.ncsu.edu:mcdaniel): mcdaniel
    331 Password required for mcdaniel.
    Password:
    230 User mcdaniel logged in.
    ftp>

To log in, use your username and password. The pwd command at the ftp> prompt will show you that you are in your home directory. Change to the directory you want and get file. Type bye to exit ftp.

If you are ftp-ing from an Eos/Unity workstation to another computer on the Internet where you do not have an account, you generally login as anonymous or guest and use your e-mail address as your password.
Once in, you get an \texttt{ftp}> prompt and can \texttt{ls} and \texttt{cd} to directories until you find what you want. You then \texttt{get} or \texttt{put} as appropriate. This process is fairly standard for all public ftp sites.

When you are \texttt{getting} and \texttt{putting} files, make sure that you are in the correct directories (both for sending and receiving), and then type \texttt{put file} to move the file from the local to the remote system, or \texttt{get file} to move the file from the remote system to your local file space. Multiple files can be transferred by adding an “m” in front of \texttt{get} and \texttt{put}, e.g., \texttt{mput *.doc} will move all local files that have the .doc extension to the remote site.

When you get a message that so many bytes have been transferred, then you know that ftp has been successful. Type \texttt{quit} at the \texttt{ftp>} prompt to return to your current system.

If the file you are sending or receiving is a binary file, that is, anything other than plain text, put ftp into \texttt{binary} mode before you \texttt{get} or \texttt{put}. To do this, type \texttt{binary} or \texttt{bin} (or, \texttt{set file type binary}). To return to text mode after you have made your binary transfer, use the command, \texttt{ascii}.

\section*{Ftp-ing via a Web Browser}

You can also use the ftp capabilities of your browser by opening the URL:

\texttt{ftp://userid@ftp.ncsu.edu}

Use your Unity ID for \texttt{userid}. The ftp connection is made to the host machine, \texttt{ftp.ncsu.edu}, and you type in your password when prompted. Netscape displays your home directory in the browser window, and you select \texttt{File –> Save} (possibly \texttt{Upload}) to transfer a file from the computer you are on to your Eos/Unity file space. To download a file from your Eos/Unity space, select the file and \texttt{File –> Copy} (or \texttt{Save}).

\section*{Moving from ftp to Secure File Transfer}

File transfer between computers is commonly done through ftp. It is such a long-time standard that many people do not realize that it is inherently insecure. Your password is passed unencrypted over this connection, so if you have not done so already, you should move to a secure (ssh) ftp client or service for transferring files.

The campus is gradually implementing SFTP and ssh-based clients. Smartftp, an SSL-based client, replaces WS_FTP in the Unity labs. The College of Engineering has two programs it recommends and makes available for download by users of Windows computers: WinSCP and F-Secure. More information about downloading and using ssh-based software is available from http://www.eos.ncsu.edu/remoteaccess.
ResNet and Remote Access

From
www.ncsu.edu/it/essentials/connections/off_campus/internetbody.html and http://www.ncsu.edu/resnet/

Students, faculty, and staff can access the campus network if they have an Eos/Unity account and a computer with a modem and appropriate communications software. Because so few dialup lines are available, users are encouraged to have an Internet Service Provider to provide them with remote access support. Dormitory residents can make a direct connection to the campus network via ResNet without going through a modem/dialup connection.

NC State ResNet

ResNet is the residential computer network service provided to students living on campus. When equipped with a 10/100BaseT Ethernet adapter, student-owned computers in the dorms can connect directly to the University’s computer network and to the Internet. For questions about ResNet, first consult the Web site at http://www.ncsu.edu/resnet/. If you still need help, e-mail resnet@ncsu.edu or help@ncsu.edu, or call ITD at 515-HELP (515-4357). NC State’s Communication Technologies also has information for students at http://comtech.ncsu.edu/students/.

Who Can Connect. All residence halls and houses on Fraternity Court. Beginning with the 2001-02 academic year, all students living in the residence halls received ResNet service. Road Runner cable modem service is available for residents of E.S. King Village for $20 per month.

Who Cannot Connect. University Towers provides residents with 10BaseT Ethernet service and a connection to the Internet similar to ResNet via their “University Towers Connect” program. UT has contracted with CampusLink to provide Ethernet service, so contact University Towers (755-1943) or CampusLink (800-962-4772).

What ResNet Gives You. ResNet lets you connect your personal computer, when equipped with a 10BaseT Ethernet adapter, directly to NCSU’s data communications network, and through the university network, to the Internet. It enables you to perform many tasks from your dorm room: exchange e-mail, browse and publish information on the
Web, access campus information and the library, submit assignments to professors, conduct research for papers, participate in electronic groups (mailing lists and news groups), work in programs in text mode, and access your Eos/Unity network file space via applications like ssh and ftp.

**What It Does Not Give You.** The Windows and X/Motif application software in the Eos/Unity environment with graphical user interfaces (such as Pro/ENGINEER, AutoCAD, MATLAB, etc.) cannot be accessed from a computer connected to ResNet services.

**How to Sign Up.** All subscriptions to and cancellations of ResNet service are conducted online through either University Housing’s Assignment process or the Greek Court Communication Technologies/ResNet services application process. Before signing up, read the ResNet terms and conditions at [http://www.ncsu.edu/resnet/policy/](http://www.ncsu.edu/resnet/policy/). Then sign up online at [http://www.ncsu.edu/resnet/app/](http://www.ncsu.edu/resnet/app/).

**What it Costs.** The fee for ResNet service is $80 per semester. Beginning with the 2001-02 academic year, ResNet will be provided automatically for all students living in University Housing’s residence halls. All ResNet data jacks will be active in students’ rooms. This change does not affect Greek Court. Students living in the residence halls will be billed for ResNet at the same time they are billed for their on-campus housing, with the charge showing up on their student account.

**Recommended Computers:** [http://www.ncsu.edu/it/compspecs/](http://www.ncsu.edu/it/compspecs/) and [http://www.ncsu.edu/resnet/info/minimum.html](http://www.ncsu.edu/resnet/info/minimum.html) provide specifications for purchasing computers to bring to NCSU. Residence halls and Greek Court houses are wired with 10/100BaseT (also known as twisted pair) Ethernet cable. All adapters must have a 10BaseT or 10/100BaseT Ethernet connection, see [http://www.ncsu.edu/resnet/help/](http://www.ncsu.edu/resnet/help/). Purchase only an Ethernet adapter that has a 10BaseT or 10/100BaseT connection!

**Remote Access to Eos/Unity**

**Off-Campus Internet Protocol (IP) Connections**

Modem access is provided by free of charge, but be warned that this service tends to be very busy. Configure your computer’s communication/modem software to dial 515-6320. Modem access via Internet Service Providers is quicker and better.

**On or Off-Campus telnet and ftp Hosts**

Telnet is a text-only Internet protocol for remote login. Most computers have telnet software included among the basic software applications. It acts as a terminal emulator that you can use to access and work in your Eos/Unity file space. This is useful when working from a computer that is not a realm workstation, either on campus or off.
However, telnet is an older protocol and is not secure. For this reason, the campus recommends downloading a secure shell client (ssh) like PuTTY to run from home (see http://www.eos.ncsu.edu/remoteaccess/). For more, see SSH and PuTTY.

FTP is another text-only Internet protocol. If you are not using an Eos/Unity workstation, you can use ftp to transfer files from your computer to your Unity file space. ftp to the host machine, ftp.ncsu.edu (see File Transfer Protocol). Like telnet, ftp is not secure and passes passwords in the clear, so look at the secure clients available for download from http://www.eos.ncsu.edu/remoteaccess.

**Dialup Services**

NC State University students, faculty, and staff can get telephone access to the campus computing network if they have an Unity account, a computer with Internet capability, and a modem. Unity accounts are automatically generated for all NCSU students, faculty and staff, and all account owners must abide by current account policies (see Policies, http://www.ncsu.edu/it/rulesregs/remote-access/)

**Service Providers**

Computing Services supports a free dialup service, but because the demand for this service has increased beyond what University resources can provide, phone lines are often busy. Callers may have to wait a long time to gain phone access to the system.

**NCSU Dialup Services.** To use the campus dialup services, first make sure your computer is Internet capable, then create a dialup connection to 515-6320. When your computer connects with the campus dialup servers, login to the system with your Unity username and password.

**Commercial Internet Service Providers (ISPs).** To obtain more reliable dialup access to the Internet and the campus computing network, most students, staff, and faculty subscribe to commercial Internet Service Providers (ISP). While Computing Services does not endorse a particular ISP, it has gathered a list of local ISPs in Choosing an Internet Service Provider (http://www.ncsu.edu/cc-consult/ISPs.html), which also provides a checklist of features to consider when subscribing to a commercial service.

**Types of Dialup Connections and Software**

**Internet Protocol (IP) connections.** Point-to-Point Protocol (PPP) client software gives your computer a temporary Internet address, which allows the use of Graphical User Interface (GUI) applications, such as
Netscape or Internet Explorer. Most new computers come already equipped with a modem that can handle PPP connections. At the very least, a 56K modem or better is strongly recommended.

**Line-mode connections (also called text-mode or TTY).** Line-mode modem connections provide basic network services—e-mail, ftp, telnet and Web access—but in text mode, not graphical formats. For example, e-mail is done with pine (IMAP), elm (POP), and Web browsing in lynx. Users need a computer, telephone line, modem with appropriate cables, and communications software that will emulate a VT100 series terminal or better. The phone number is 515-3980.

### What Can I Do on Dialup?

Anything you can type at an eos% or unity% prompt and get a response that does not require additional windows to pop up or non-ASCII graphics, you can also do on a dialup machine. This includes sending and receiving mail, reading and posting news, editing, compiling and running programs, using lynx (line-mode Web browser), communicating with other users, and ssh-ing and ftp-ing to other sites.

Many users become very proficient at working from home using these and other applications. Some of the principal Eos/Unity software packages (see [http://www.eos.ncsu.edu/software/](http://www.eos.ncsu.edu/software/)) also have line-mode interfaces that users can access. See also the last section of this chapter for use of X-Win32 to display applications in graphical mode from Eos remote-access servers.

### What Can’t I Do on Dialup?

Anything that requires X-Windows or graphical screen displays will not work over dialup (Maple, Matlab, FrameMaker, NExS, etc.). There are alternatives available for some applications. You can use command-line options (such as the -ttymode option on Zephyr), or applications other than those you typically use at the workstation (e.g., editing with vi or pico instead of NEdit). With applications you use a lot, you may wish to buy a copy for your machine at home and then transfer files between it and campus computers.

Because there are too few dialup lines available for the large number of users demanding access, people are not permitted to play games. Similarly, running very large and intricate programs that take up large amounts of CPU time are not recommended on dialup. In addition, some dialup machines restrict the number of processes users may have at any one time. For batch jobs, look at using Load-Sharing Facility (LSF) [http://www.eos.ncsu.edu/software/lsf/](http://www.eos.ncsu.edu/software/lsf/)

### Reading Mail from Dialup Machines

Campus users are being moved from the POP mail protocol to IMAP, but POP users can still use the line-mode program elm. IMAP users can use Large and/or GUI (graphical user interface) applications with non-ASCII input/output cannot run on dialup machines. Games are not allowed on dialup machines. Use the Load-Sharing Facility (LSF) for remote batch processing, see [http://www.eos.ncsu.edu/software/lsf/](http://www.eos.ncsu.edu/software/lsf/)

515-3980
Consult the following URLs for guidance on accessing mail from home if you are not yet getting your mail through IMAP: http://webmail.ncsu.edu/ and http://www.ncsu.edu/imap/

**Compiling and Running Programs on Dialup**

Every computer language has a method of compiling programs from the prompt. As long as it does not require an X window to pop up, you should be able to compile and run programs without difficulty.

**Editing Files on Dialup**

The editor recommended for use over dialup is **vi**. The documentation for vi is too extensive to reproduce here, but some documentation is available in **man vi**. A table of vi commands appears in Appendix B. Also add **editors** and **cd /ncsu/editors/bin/** for other editors you might use. See also **Common Text Editors**.

**Communicating Directly with Other Users**

Because the **Zephyr** system will not work in quite the same way as it does on a standard workstation, you will have to use it in TTY terminal mode. To start up Zephyr over dialup, type:

```
/usr/athena/etc/zwgc -ttymode
```

(do not put this command in the background with &)

The **-ttymode** option will redirect incoming zephyrs to you in an all-text format that the dialup machine can handle (see also **man zwgc**).

**Principal NCSU Server Addresses**

Dialup server (all Eos and Unity accounts):

**login.ncsu.edu**

This address names a large pool of dialup machines that handle access for NCSU students, staff, and faculty. The **login.ncsu.edu** name is a hostname alias that points to this pool and automatically finds a dialup connection on one of the machines.

ftp server (all Eos and Unity accounts): **ftp.ncsu.edu**

ftp server (College of Engineering only): **ftp.eos.ncsu.edu**

**Eos Remote-Access Services**

The College of Engineering is developing more remote-access services to support its student-owned computing and distance education initiatives. Software and information are available from:
A high-speed connection through an Internet Service Provider (DSL or cable modem rather than a dialup modem) is recommended with some types of connections (file transfer) and essential in others (XWin32). To protect the privacy of those who use these services, all technologies provided use secure, encrypted communication. Also, open-source technologies are used whenever possible to implement remote access.

**Running Windows from Home**

Most users run Windows on their home computers but need access to their AFS file space and to Solaris/Linux resources on campus (no access to campus Windows resources is yet available). Because of fundamental differences in the operating systems, remote access from Windows to Solaris/Linux can be problematic. Also, the X Window System, used in both realm Solaris and Linux, is not in any distribution of Windows.

To make a Windows computer access Unix-based resources remotely requires the installation of several programs.

- Secure file transfer requires WinSCP or F-Secure.
- Terminal access for command-line execution of commands and applications remotely requires PuTTY or F-Secure.
- Running an application in graphical mode requires X-Win32 running with PuTTY or F-Secure.
- Native AFS access requires WolfCall, Kerberos for Windows, and OpenAFS.
- Batch processing through LSF requires PuTTY or F-Secure and connecting to separate lsf.ncsu.edu batch servers.

**Secure File Transfer**

Two file-transfer packages are available for download: WinSCP and F-Secure. Conventional ftp software is not recommended because it is not secure. For file transfer only, WinSCP is easier to install and use. However, if you also want telnet-type connection through Secure SHell terminal access, then you may wish to consider F-Secure.

It is important to note that you do not connect through ftp.eos.ncsu.edu or ftp.ncsu.edu with this software but to the ssh-configured servers, `remote.eos.ncsu.edu` and `remote-linux.eos.ncsu.edu`.

**Running Applications Remotely**

To run applications remotely from a Windows computer also requires connecting to the remote-access servers: remote.eos.ncsu.edu to run
Solaris applications and remote-linux.eos.ncsu.edu to run Linux applications. The applications you connect to actually run on these remote computers, not on your Windows machine, so such things as printing locally from the application will not work. Also, the speed at which the application runs is determined by the capacity of the remote server and the load it is carrying. Remote-access servers are shared resources.

For command-line execution of applications like pine, elm, pico, etc. and to execute Unix and AFS commands at an eos% prompt requires the installation of either F-Secure or PuTTY. To run an application in full graphical mode (one application at a time only) requires X-Win32 to be downloaded each time you make a connection, see also
http://www.eos.ncsu.edu/software/putty
http://www.eos.ncsu.edu/software/xwin32

Native AFS Access (Windows 2000/XP only)

WolfCall (http://www.eos.ncsu.edu/wolfcall) is a replacement for KAUTH. WolfCall makes it possible for you to map your home directory (K: drive) and AFS root directory (J: drive) on your home computer so that you can work directly with files in your campus AFS space without going through file transfer or terminal access. You open, save, copy, and delete files just like they were on your local machine, or, like you were sitting in an Eos lab.

Batch Processing

Remote batch processing is the same as working from a campus machine except that you need to download an SSH client, such as F-Secure or PuTTY, to make a secure connection to the lsf.ncsu.edu batch servers and the application you need (see http://www.eos.ncsu.edu/software/lsf/).

Consult http://www.eos.ncsu.edu/remoteaccess for more information and access to the resources described above, plus more.

Important! Don’t forget to maintain current anti-virus software on your home computer. Norton Anti-Virus software with regular anti-virus updates is free for download to all NCSU faculty, staff and students,

http://www.ncsu.edu/antivirus/
UNIX-ONLY APPLICATIONS AND CUSTOMIZATION
Zephyr is a set of command-driven utilities for message transport and delivery that are executed from an Xterm window.

To get rid of a Zephyr, just click inside the Zephyr window. Do not select Close from the menu of a Zephyr window! This action will shut down the program, and you will not be able to send or receive messages.

To restart the program:
```
/usr/athena/bin/zwgc
```

To run Zephyr on Windows, select XWin32 and log in. Then type the command above at the prompt.

For help using Zephyr:
```
man zephyr
```
or http://web.mit.edu/olh/Zephyr

To locate a user with Zephyr:
```
zlocate username
```
To locate more than one user:
```
zlocate username username
```

The **Zephyr** service supports the quick exchange of messages in real time among users logged on the system. A **Zephyr windowgram**, called a **Zephyr** or **Zephyrgram**, is the simplest and quickest way to send a message to another user or a group of users. Its advantage over electronic mail is that it is faster and immediately noticeable to the receiver. Unlike e-mail, however, Zephyr does not save, store, or retrieve its messages. Zephyr is best used for the same kind of immediate and unrecorded communication that you use Instant Messenger for.

The **Zephyr Windowgram Client (zwgc)**, which is automatically started for you at login, is the program that displays Zephyr notices. As soon as you log in, Zephyr makes you “locatable,” that is, others can find when and where you are logged on. You receive Zephyr messages through subscription. All users are subscribed to two kinds of messages: **personal** messages and **operational** messages. The former permits you to send and receive personal messages. The latter is used by system administrators to alert you to system changes, problems, and news.

To get rid of Zephyr messages that appear in the upper left corner of your screen, just click on them with any mouse button. Do not **Close** from the window menu or you will kill Zephyr and have to restart it (see margin).

**Locating a User (zlocate)**

To exchange Zephyr messages, you must be logged on to the system. To see if someone is on the system and where they are working, use the **zlocate** command followed by the person’s username, e.g.,
```
% zlocate jqpublic
```

If the person is logged in, the system will deliver you a message telling you the workstation(s) the user is on (it is possible to be logged in to more than one workstation). It also tells the time and date of login and the machine’s location, e.g.,
```
c00201-100lez.eos.ncsu.edu :0.0 Mon May 10 09:23:21 2002
```

If the user is not logged in or is not subscribing to messages, the system will respond with the message,
```
Hidden or not logged-in.
```
If you want to find out if several users are logged in, you can type more than one name after the `zlocate` command, e.g., `zlocate moe larry curly` (one space and no commas between usernames).

**Writing a Zephyr Message (zwrite)**

Zephyrwrite, or `zwrite`, is the program that allows you to compose and send a zephyrgram. In an Xterm window, type `zwrite` followed by the username(s) of the person(s) who will receive the message, e.g.,

```shell
% zwrite jqpublic
```

If the user is not logged in, then s/he cannot receive Zephyr messages. Thus, you would need to write to the person using e-mail, which holds mail messages and delivers them to users when they log in.

If the user is not logged in, you will get the following message:

`jqpublic: Not logged in or not subscribing to messages`

If the person is logged in, you will be instructed to:

- Type your message now. End with control-D or a dot on a line by itself.

After this message, write the message you want to send. When you have completed the message, press `RETURN` to move down one line. Then, either type a period on a line by itself and press the `RETURN` key, or type `CONTROL d` (hold down the `CONTROL` key and type `d`). This action immediately sends the message to the recipient(s). When a message is sent, the sender receives the following notice in the Xterm:

`jqpublic: Message sent`

Zephyr is designed to help the user prepare and send short messages quickly. It is not designed with the flexibility and functionality of a word processor or mail program. For example, there is no word wrap in Zephyr; as a result, after you type a line of text, you must press the `RETURN` key to move to the next line. If you do not do this, the text will automatically wrap to the next line on your screen but not on the screen of the person(s) receiving the message. The line they see will be very long, probably with words cropped off.

Also, the cursor keys do not work in Zephyr. On some workstations, you may be able to backspace or delete, but it stops at the beginning of the line you are on. As a result, you cannot move up to other lines in the text or even backspace to get to them. You just need to type very carefully, or cut and paste from an editor.

The following figure is a sample Zephyrgram sent by the sender to herself. It shows how the sender writes and sends a message and, also, what the message looks like when it appears on the screen of the receiver.
The term *authentic* in the *zwgc* window means that the sender field was verified by Kerberos authentication. If authentication information was not verified by Zephyr, then the message is *unauthentic* (although this is not necessarily anything the user needs to be concerned about).

### Changing the Format of a Zephyrgram

Zephyr also lets the sender specify different font types and sizes. The table on the opposite page shows a list of commands that format the text of a zephyrgram. The command on the left produces the effect on the right. The figure that follows is a sample screen showing what the sender typed to get the formatting that appears in the receiver’s *zwgc* window.

The spacing of the formatting command is important. To get a boldface line of text, you must first type @b followed by the text you want boldfaced inside parentheses. There must not be a space added to the command. For example, you should type @b(This is bold.) if you want the text *This is bold.* to appear on the receiver’s screen.

If you want more than a single formatting instruction to apply—for instance, if you want bold italics—you add those additional format instructions to the command with a left parenthesis separating them, e.g., @b(@i(This is bold italics.) produces *This is bold italics*. Three format instructions for centered bold italics would look like @b(@i(@c(This is centered bold italics.) Remember to use lowercase letters with an @ sign preceding each and a left parenthesis following.
Format commands are followed by text in parentheses, except when specifying font or color.

<table>
<thead>
<tr>
<th>Format Command</th>
<th>Effects of the Command</th>
</tr>
</thead>
<tbody>
<tr>
<td>@b or @bold(text)</td>
<td>Bold</td>
</tr>
<tr>
<td>@i or @italic(text)</td>
<td>Italics</td>
</tr>
<tr>
<td>@roman(text)</td>
<td>Nullifies the @italic and/or the @bold command</td>
</tr>
<tr>
<td>@l or @left(text)</td>
<td>Left-aligns text</td>
</tr>
<tr>
<td>@c or @center(text)</td>
<td>Centers text</td>
</tr>
<tr>
<td>@r or @right(text)</td>
<td>Right-aligns text</td>
</tr>
<tr>
<td>@small(text)</td>
<td>Puts text in small type size</td>
</tr>
<tr>
<td>@medium(text)</td>
<td>Puts text in medium type size</td>
</tr>
<tr>
<td>@large(text)</td>
<td>Puts text in large type size</td>
</tr>
<tr>
<td>@huge(text)</td>
<td>Puts text in type larger and bolder than large</td>
</tr>
<tr>
<td>@beep(text)</td>
<td>Beeps on arrival at the receiver’s workstation</td>
</tr>
<tr>
<td>@font(fontname)text</td>
<td>Changes to a specified font type, e.g., Courier, Times, Helvetica</td>
</tr>
<tr>
<td>@color(colorname)text</td>
<td>Sets the color (type showrgb</td>
</tr>
</tbody>
</table>

```
Authentic: Personal message at 21:41:55 on Wed May 12
The sender’s workstation is elvis.ece.nmsu.edu
This is small. This is italic.
This is bold. This is italic. This nullifies @b and @i.
```

```
Type your message now. End with control-B or a dot on a line by itself.
@small(This is small.)
@medium(This is medium.)
@large(This is large.)
@l(This is left-aligned.)
@r(This is right-aligned.)
@c(This is centered.)
@b(This is bold.)
@i(This is italics.)
@b@i(This is bold italics.)
@roman(This nullifies @b and @i.)
```

```
mcdaniel: Message sent
```

```
cos²
```
**Copying and Pasting**

You can copy from a Zephyr window and paste into another window, but not in the usual way (that is, selecting the text with MB1 and pasting with MB2, as described in Helpful Shortcuts). As soon as you click MB1 in a Zephyr window, it disappears. Instead, hold down the **SHIFT** key and drag over the text with MB1 to select it (the text changes to white on a black field). Once selected, release the **SHIFT** key and copy the text into another window in the usual way: point at the place in the window where you want the text to be and click MB2.

To copy something into a Zephyr message, select what you want to copy (MB1) and paste it into the message by pointing and clicking MB2 in the Xterm window after the following line:

Type your message now. End with control–D or a dot on a line by itself.

Once the text is pasted in and you have added anything else you want to say, press **RETURN**. Then, either type a period on a line by itself and press **RETURN** again, or hold down the **CONTROL** key and type **d**.

**Subscribing to Other Messages (zctl)**

In addition to the personal and operational messages you receive automatically, you may also want to subscribe to other **classes** of Zephyr messages. These classes are general categories of Zephyr messages that you can receive only if you are subscribed to them. Under these general message categories is a second group of messages called **instances**, which you also specify in your subscription to a message class. The third piece of information that subscription requires is the **recipient**, the subscriber who receives the information. The default recipient is your username.

For example, the personal messages to which you are automatically subscribed come from the class called **message** and the instance called **personal**. The system messages you receive from the operations staff belong to the class **operations** and the instance **message**. In both cases, the recipient is your username.

The act of subscribing to information sources on a network is a common one, both locally on the Eos/Unity system and more widely on the Internet. Although a great deal of information gets to you whether you ask for it or not, an even greater amount remains that you must ask for specifically. Subscribing to network information sources works very much like a subscription to a magazine or newspaper (although most of the services you can subscribe to on the net are free!).

In general, to subscribe to a service requires the simple act of writing a single line of instructions in a particular syntax (a line of code) and
sending it out via a mail or messaging service to a computer. The computer reads the code and automatically sets you up to receive the information automatically. In other words, you do not write a person and ask to subscribe to a resource. Subscribing is all automated.

The process of subscribing in Zephyr works in the same fashion. The format for subscribing to a Zephyr message class and instance is the command `zctl` (for Zephyr Control) followed by `sub` (to subscribe for the current session only) or `add` (for a regular subscription). These two words are then followed by specific `class`, `instance`, and `recipient` arguments. One space separates each word in the command syntax.

**The Help Instance**

Most Zephyr messages that you will be interested in subscribing to are in the general class called `message`, with the `instance` describing the particular type of message you want to receive. One particular instance you may want to subscribe to is the `help instance`. If you are subscribed to “instance help,” you can broadcast a question or help request to the whole Eos/Unity network. Users who are also subscribed to this instance will get your message, and chances are, someone will write you back an answer. The operators in the labs are generally subscribed to this instance and will pick up most of the questions that users ask.

If you want to receive all help messages exchanged on the system, use the wildcard character `*` as the recipient instead of your username. This gives you access to all messages that the help instance sends out, which is directed to a large audience of users, not just individuals. To become a regular subscriber to the help instance, type at the prompt, `zctl add message help *`. To write to help, use the command `zwrite -i help`.

When you subscribe, a `.zephyr.subs` file is automatically created in your home directory to log your subscriptions. To drop your subscription to an instance, you would use `unsubscribe` if you have subscribed with the `sub` command, and `delete` if you have subscribed with `add` (see margin).

**Locating Groups of Users (znol)**

Like `zlocate`, the `znol` command helps you locate people on the network. Whereas the `zlocate` command lets you add multiple usernames as arguments (e.g., `zlocate moe larry curly`), `znol` reads a `.anyone` file that you create and place in your home directory. In this file is a list of the usernames of people you want to locate often. When you run `znol`, the program reads the file of usernames and lets you know when and where each person is logged in. A `.anyone` file lists usernames one to a line, e.g.,

```none
moe
larry
curly
```

To subscribe temporarily to a Zephyr message class and instance (for the current session only):

`zctl sub class instance recipient`

To unsubscribe:

`zctl unsub class instance recipient`

To subscribe regularly to a Zephyr message class and instance:

`zctl add class instance recipient`

To unsubscribe:

`zctl delete class instance recipient`

**Note:** If you are the recipient of the message, you do not have to type anything for “recipient” since it defaults to your username.

To subscribe to the help instance on Eos:

`zctl add message help *`

To write to help:

`zwrite -i help`

To locate users whose usernames are listed one to a line in your `.anyone` file:

`znol`
Blocking Messages (zaway, zctl hide)

If you want to discourage users from sending you messages while you are working, you can use either the zaway or the zctl hide commands to let them know that you are not available for Zephyr exchanges. These commands will not keep messages from coming in; however, they will send a notice to the sender that you are not responding to messages.

- The command zaway returns a Zephyrgram to anyone sending you a message, saying,

I’m sorry, but I am currently away from the terminal and am not able to receive your message.

Or, zaway will substitute for this message any message you have written and placed in a .away file in your home directory. To kill the zaway process (and get the prompt back), type Control c in the Xterm window where you typed zaway.

(Note: If you type zaway and then zwrite yourself, you will not get either the zaway or zwrite message.)

- The command zctl hide hides you from anyone using the zlocate command to find you, returning to the sender the message:

Hidden or not logged in.

To “unhide,” type zctl unhide.

To block out all Zephyr messages, except those sent out from the system administrators, and to hide from the zlocate command, use the command zctl set exposure none. To become “locatable” and receive messages again, type zctl set exposure realm-visible.

<table>
<thead>
<tr>
<th>Command to Engage</th>
<th>Command to Disable</th>
<th>Effects on zlocate</th>
<th>Effects on zwrite</th>
</tr>
</thead>
<tbody>
<tr>
<td>zaway</td>
<td>Control c</td>
<td>user can be located</td>
<td>user receives zephyrs; sender receives zway message</td>
</tr>
<tr>
<td>zway &amp;</td>
<td>kill process#</td>
<td></td>
<td></td>
</tr>
<tr>
<td>zctl hide</td>
<td>zctl unhide</td>
<td>user cannot be located</td>
<td>user receives zephyrs</td>
</tr>
<tr>
<td>zctl set exposure none</td>
<td>zctl set exposure realm-visible</td>
<td>user cannot be located</td>
<td>user does not receive zephyrs</td>
</tr>
</tbody>
</table>
COMMON TEXT EDITORS

In every software application you use, you are creating and using files of information: text, graphics, charts, equations, tables, data sets, etc. In an editor, you are specifically creating information in ASCII text, either for documents or computer programs.

On Eos/Unity, there are several excellent editors you can use to create computer programs and write documents (but not with all the functions of a word processor). Some are graphical X applications, such as NEdit and Visual Slickedit, which can only be used on the workstations. In addition, there are line-mode editors that can be used for dialing into the system and in ssh sessions: vi and pico. Finally, there is a specialized HTML editor called asWedit, which is designed to produce and correct files written in Hypertext Markup Language for delivery as documents on the web. Other editors can be found in the editors locker: add editors.

NEdit

NEdit is probably the easiest editor to use for writing programs and papers, particularly the latter. It can be brought up from the main Application Menu (Editors -> NEdit), or by typing nedit & on the command line (also on Linux but not Windows). It opens with the simple screen layout below.

From NEdit’s Help pages: NEdit is a standard GUI (Graphical User Interface) style text editor for programs and plain-text files. Commands

ASCII: American Standard Code for Information Interchange, a standard computer code used to facilitate the exchange of information on various types of data-processing equipment.

For other editors:

add editors
cd /ncsu/editors/bin

nedit &

Also on the
Application Menu –> Editors
are available from the pull-down menus (File, Edit, Search, etc.). All dialog boxes have Cancel buttons, and the Undo command in the Edit menu can reverse any modifications that you make. NEdit does not change the file you are editing until you tell it to Save.

Editing an Existing File

To open an existing file, choose Open... from the File menu. Select the file you want in the pop-up dialog that appears and click OK. You may open any number of files at the same time. Each file will appear in its own window. Opening another file rather than re-typing the NEdit command and running additional copies of NEdit will give you quick access to all of the files you have open via the Windows menu, and will ensure that you do not accidentally open the same file twice. NEdit has no “main” window. It remains running as long as at least one editor window is open.

Creating a New File

If you already have an empty (Untitled) window displayed, just begin typing in the window. To create a new Untitled window, choose New from the File menu. To give the file a name and save its contents to the disk, choose Save or Save As... from the File menu.

Backup Files

NEdit maintains periodic backups of the file you are editing so that you can recover the file in the event of a problem such as a system crash, network failure, or X server crash. These files are saved under the name ~filename (on Solaris), where filename is the name of the file you were editing. If an NEdit process is killed, some of these backup files may remain in your directory.

Selecting Text

To select text for copying, deleting, or replacing, press the left mouse button with the pointer at one end of the text you want to select and drag it to the other end. The text will become highlighted. To select a whole word, double-click (click twice quickly in succession). Double-clicking and then dragging the mouse will select a number of words. Similarly, you can select a whole line or a number of lines by triple-clicking or triple-clicking and dragging. Clicking four times selects the entire contents of the window. After releasing the mouse button, you can still adjust a selection by holding down the Shift key and dragging on either end of the selection. To delete the selected text, press delete or backspace. To replace it, begin typing.

Cut and Paste

The easiest way to copy and move text around in your file or between windows, is to use the clipboard, an imaginary area that temporarily stores text and data. The Cut command under Edit removes the selected
text from your file and places it in the clipboard. Once text is in the clipboard, the **Paste** command will copy it to the insert position in the current window. For example, to move some text from one place to another, select it by dragging the mouse over it, choose **Cut** to remove it, click the pointer to move the insert point where you want the text inserted, then choose **Paste** to insert it. **Copy** copies text to the clipboard without deleting it from your file. You can also use the clipboard to transfer text to and from other Motif programs.

**Finding and Replacing Text**

The **Search** menu has a number of commands for finding and replacing text. The **Find...** and **Replace...** commands present dialogs for entering text for searching and replacing. These dialogs also allow you to choose whether you want the search to be case-sensitive, or whether to use the standard UNIX pattern matching characters (regular expressions). Searches begin at the current text insertion position.

**Find Same** and **Replace Same** repeat the last find or replace command without prompting for search strings. To selectively replace text, use the two commands in combination: **Find Same**, then **Replace Same**, if the highlighted string should be replaced, or **Find Same** again to go to the next string.

**Find Selection** searches for the text contained in the current primary selection. The selected text does not have to be in the current editor window. For example, if the word “dog” appears somewhere in a window on your screen, and you want to find it in the file you are editing, select the word “dog” by dragging the mouse across it, switch to your NEdit window and choose **Find Selection** from the **Search** menu.

**Searching Backwards**

Holding down the **SHIFT** key while choosing any of the search or replace commands from the menu (or using the keyboard shortcut) will search in the reverse direction.

**Selective Replacement**

To replace only some occurrences of a string within a file, choose **Replace...** from the **Search** menu, enter the string to search for and the string to substitute, and finish by pressing the **Find** button. When the first occurrence is highlighted, use either **Replace Same** (\^T) to replace it, or **Find Same** (\^G) to move to the next occurrence without replacing it, and continue in such a manner through all occurrences of interest.

To replace all occurrences of a string within some range of text, select the range, choose **Replace...** from the **Search** menu, type the string to search for and the string to substitute, and press the **R. in Selection** button in the dialog. Note that selecting text in the **Replace...** dialog will unselect the text in the window.
Spell Checking

The way NEdit checks the spelling in a document is to run the UNIX shell command `spell` on the file. When you select Shell at the top of the window, a pull-down menu opens with `spell` as one of the menu items. Select it to run the program. The program will produce a list of words that the program does not find in its dictionary. You can then use this list to find and correct the misspelled words in the document.

Consult Help for additional information, including features for programming, shell commands and filters, regular expressions, etc.

Visual SlickEdit  Programmer’s Editor

Visual SlickEdit is one of the most highly rated programmer’s editors in the industry. Very powerful but easy-to-use, it is multi-platform, highly extensible, and fast. Its Slick-C language and multi-platform graphical development environment simplify program enhancements and enable rapid additions of such editor features as SmartPaste, API apprentice, difference editing, code block selection, and macro recording. It includes a code beautifier for C++/Java, graphical 3-way merge, hex editing, source-code tagging, spell checking, and sophisticated hyperlinked Help programs for learning and using SlickEdit. (On Linux and Windows)

Suggestion for conserving file space when using SlickEdit: By default, Visual SlickEdit makes a full-sized backup copy of every file that you open/edit/save using the program. The file copies are "hidden" in the directory `~/.vslick/backup`, and over time, slowly eat up your disk space. To fix this, go to Tools -> Configuration -> File Options, click on the Backup tab and uncheck the Make backup files option box. Cleaning out this directory and changing the backup option will save your quota.
**vi**

**Vi** is one of the line-mode text editors on Solaris and Linux. Type **vi filename** on the command line at the system prompt in order to access the editor. The filename is either the name of the file you want to edit (one that already exists) or the name you want to give to a newly created file. If you do not provide a filename, vi will open a new file, but it will remain unnamed until you save it.

There are two different modes for adding text to a file. The first is to append the text. Appending text means that you add characters after the cursor location. To enable the append mode, type **a** and begin typing. If you repeat the same text sequence several times by typing **na** (where **n** is the total number of times the sequence repeats), vi will automatically copy what you type for a total of **n** times after you hit **RETURN** or exit the append mode. To end the append mode, hit the **ESC** (escape) key.

The second way to edit text in vi is to use the insert mode. Inserting text means that you add characters before the cursor location. To enable the insert mode, type **i** and begin typing. As with the append mode, if you type a number before the **i**, vi will copy what you type that many times. To end the insert mode, hit the **ESC** key.

There are several ways to delete. To delete a character under the cursor, just type **x**. To delete a word, place the cursor at the beginning of the word to be deleted and type **dw**. Make sure your cursor is at the beginning of the word because vi will delete the word from where the cursor is until the next space. Typing a **D** (uppercase D) will delete the text from where the cursor is to the end of the line. To delete the whole line, type **dd**.

Vi also allows several different ways to save and exit files. To save what you have written into a file, type **:w** (the colon is very important, so don’t forget it). This will write what you have written to a file and still allow you to continue typing. To save the file and quit vi, type **:wq**. To quit if nothing has changed in the file, type **:q**. If you want to abandon the file completely and not save changes, type **:q!**.

Many users prefer **vim** (**vi** improved), which is on both Solaris and Linux platforms:

- **add editors**
- **vim**

**vim** allows multiscreen editing, more flexible insert/command mode handling, better C indentation and much more.

For help with **vi** and **vim**:

- **man vi**
- **man vim**
For additional help with vi, type `man vi`. Also, a more complete list of commands is available in Appendix B.

**Pico**

Pico is another line-mode editor on Solaris and Linux. To bring it up, type `pico filename`.

Pico is a simple, easy-to-use text editor with a layout similar to the pine mailer. The status line at the top of the window shows pico’s version, the current file being edited, and whether or not there are outstanding modifications that have not been saved. The third line from the bottom is used to report informational messages and for additional command input. The bottom two lines list the available editing commands.

Each character typed is automatically inserted into the buffer at the current cursor position. Editing commands and cursor movements are given to pico by typing special control-key sequences. A caret ( ^ ) is used to denote the CONTROL key. Hold down the CONTROL key and type the letter indicated. For example, the CTRL-X key combination is written as ^X and is the command for exiting the program. ^G brings up Help.

**asWedit HTML Editor**

asWedit is specially designed as an editor for creating and modifying files written in HTML, or Hypertext Markup Language, the language used to create documents for the web. It is particularly useful for flagging and fixing errors you might make in composing HTML files.

The Help program is an online hypertext system with linked words that lead to other topics. These links are framed by thick vertical bars. Selecting and double-clicking on the links (or pressing RETURN) takes you the topic’s page. asWedit also checks spelling (Tools –> Spell).
Another popular editor on Solaris and Linux is GNU emacs. The main commands are:

Get help: C-h (Hold down CTRL and press h)
Undo changes: C-x u
Exit Emacs: C-x C-c
Get a tutorial: C-h t
Use Info to read docs: C-h i
Activate menubar: F10 or ESC

See also http://www.gnu.org/software/emacs/
and
http://www.gnu.org/software/emacs/emacs-faq.text
XTRACS, written by former NCSU student, Lance Lovette, is an online course-scheduling program for the NCSU TRACS registration system. This program enables a student to prepare an up-to-date course schedule, although it cannot officially enroll the student in classes. XTRACS gets the complete and official NCSU course listings from a file that is updated nightly by TRACS. Students can find, select, and arrange courses on a schedule graph, and then print the graph to use when phoning in a completed schedule to TRACS. (On Linux, but not Windows)

You can bring up the XTRACS program from the main Application Menu under Miscellaneous. Or, type at the prompt

\texttt{xtracs &}

You will be asked which semester’s courses you want to load. Select the session you want, and then click the OK button. A message will pop up telling to wait for the course listings to be loaded.

When the loading is done, a schedule window will open with all of the academic department abbreviations listed in a column on the left. Selecting any one of these will list that department’s courses by course number in the second column. Selecting any course number in the second column will list all sections of that course in the third column, showing full course title, section information, and instructors’ names. If a class is open, you can add it to your schedule by selecting it and clicking the Add button. Dropping the section is just as easy; select and click Drop.

A graph window also opens when you bring up XTRACS. (If the graph does not come up, you can bring it up yourself from the Graph menu.) Whenever you make a section selection, the course section will appear simultaneously on the schedule graph so you can see how it fits into your schedule. If you try to add a course that is scheduled for the same time as another class on your graph, the program will alert you to the conflict.

If you want to schedule activities other than classes, such as work, use Schedule \texttt{\rightarrow} Block Time. If you want XTRACS to build a schedule for you, use the Generate Schedules option under Schedule.

The following figure shows a sample course schedule as it appears in the schedule window and on the course graph. The help facility inside the program gives complete instructions for using the program. Click Help to bring up the Help Index and Tutorial. Select the Tutorial to go through the procedure of filling out a course schedule.
When the **Add** button is selected, the highlighted course is added to the schedule at the bottom of the XTRACS window and on the Course Graph.
**Corel WordPerfect**

WordPerfect 8 does not run under Solaris 8, so the default version for 2003-04 is version 7. The content in this chapter is derived from [www.ncsu.edu/it/essentials/learn_more/a_z_tutorials/wordperfect/](http://www.ncsu.edu/it/essentials/learn_more/a_z_tutorials/wordperfect/)

Corel WordPerfect for Unix is the most easy-to-use word-processing program in the Eos/Unity environment. With it, you can easily type, format, and print professional-looking documents. In addition to all of the standard word-processing functions, WordPerfect for Unix includes sophisticated tools for making tables, graphics, charts, and equations. It supports mail merge, sorting, and the insertion of comments, bookmarks, sound, and spreadsheet/database links. Also integrated with the program is the “Internet Publisher,” which permits you to save your documents in HTML and publish them to the Web. (Not on Linux or Windows)

Since WordPerfect is a popular PC and Mac program, you can move among these three platforms with ease. You can save your documents in Eos/Unity file space or on IBM-formatted diskettes for use on other computers. As for printing, remember that before you can print from WordPerfect in an Eos or Unity lab, you must have print quota. Print quotas can be purchased from the WolfCopy Centers on campus.

**Starting WordPerfect**

WordPerfect may be started from the Eos/Unity main Application Menu under Word Processing. WordPerfect may also be started from your Xterm window. At your eos% or unity% prompt, type

```
add wp
```

This command adds the WordPerfect software to your environment.

At the next eos% or unity% prompt, type

```
xwp &
```

to start the WordPerfect program.

**Creating a Document**

When WordPerfect starts, two windows will appear: the program window and the document window. The program window is the smaller window that opens in the upper right corner of your screen and is used
to open document windows and the File Manager, to customize and set program “Preferences,” and to exit. The other larger window is a New Document window where you begin typing and editing a new document.

The blinking insertion point, or cursor, shows where text will be entered. You may change the position of the cursor after you have entered text by moving the mouse pointer (it will change from an arrow to a highlighted I shape) and clicking on the new position.

All new documents default to a layout using an 8.5” x 11” page with 1-inch margins, single spacing, and Courier font. As you will see below, these formats can all be changed easily.

To work on an existing document, go to the File menu and select Open. A window will pop up listing your files (on the left) and your directories (on the right). Select the name of the document you want to open by clicking on it; then click the Open button.

Using the Pull-down Menus and Power Bar

At the top of the document screen, there are two bars which enable you to perform different functions on your document, such as saving and printing, or changing the font or line spacing. The menu bar lists the name of pull-down menus, File, Edit, View, etc. To familiarize yourself with the functions available to you from the pull-down menus in the menu bar, click on each menu heading. The pull-down menus will stay open, and you can select a function by clicking on it.
The button bar, also known as the *Power Bar*, has icons for frequently used functions. To find out the function of each of the buttons on the Power Bar, place your mouse pointer over the button (don’t select, just point). A description of the button’s function pops up in a yellow box.

You may change the font, line spacing, and margins of your document either before you start typing or after you have entered text by selecting these options from the Power Bar or from the **Format** pull-down menu. Notice the position of the blinking cursor before selecting any layout options. Any change you make only affects text that follows the insertion point. To change the appearance of portions of text you have already written, highlight the text you wish to change and then select the desired function. To highlight text, hold down the left mouse button (MB1) and drag the cursor over the text and then release the mouse button. Do not type the **BACKSPACE** or **DEL** keys or the highlighted text will be deleted!

One of the prime advantages of a word processor is its ability to cut and paste text, tables, graphics, etc. Highlight what you wish to cut or copy and select **Cut** or **Copy** from the Power Bar, the pull-down **Edit** menu, or the pop-up menu selected with the third mouse button (MB3). Position the cursor where you want to insert and select **Paste** from one of these same bars or menus. Or, simply paste what you have highlighted by positioning the cursor and then clicking the middle mouse button (MB2). You can paste inside the same or another WordPerfect document, or into any other windowed application on your screen.

**Saving Your Document**

Be sure to save your document often! If anything goes wrong while you are logged on, you could easily lose the work you have done on your document since it was last saved. To save your document, either select the **Save** or **Save As** option from the **File** pull-down menu, or click on the save button on the Power Bar.

The first time you save your document, a pop-up window will appear on your screen. At the top left of the window is the **Filename/Current Selection** field, where you will type the name of your new document (the default name is *Document1*). At the top right of the window is the name of your **Current Directory** (most often your home directory), which shows where the new file will be saved. A **Directory List** is also displayed, and you can save your document to a subdirectory in the list by double-clicking on it.

The **Save Options** field at the bottom of the **Save As** window is also important to note. If you plan to work on your document in another word processor or an earlier version of WordPerfect, you can save your document in another format. For more information about saving documents, click on the **Help** button in the lower right corner of the **Save As** window. To save your document, click the **OK** button at the bottom left of the window.
Saving WordPerfect Files to a Floppy Disk

WordPerfect cannot save your document directly onto a floppy disk. You will first need to save the document in your file space and then use `mtools` to copy it to disk (see Backing Up and Restoring Files). Mtools commands must be typed at your `eos%` or `unity%` prompt, but you will not need to exit WordPerfect. Just bring up an Xterm window. The disk you insert into the floppy drive (drive `a:`) must be IBM formatted.

To copy a document from your Eos/Unity file space onto a floppy disk, type the following command at your prompt:

```
mcopy document.wp a:
```

To copy a file from your floppy disk to your Eos/Unity file space, type:

```
mcopy a:document.wp
```

`a:` is the drive the floppy disk is in, and it must be placed before the name of the file, or the computer will look for the file in your Eos/Unity file space. The period tells the computer to copy the file to your current working directory.

Saving in HTML with Internet Publisher

You may also save your WordPerfect documents in HTML, or HyperText Markup Language, the language of World Wide Web pages. On the File menu is the Internet Publisher, which opens the window below:

You can either select the first button to begin creating a Web document in HTML, or format and save existing WordPerfect documents in HTML to publish on the Web. You can look at the HTML documents saved in your home directory using a Web browser and the `file:/` URL, e.g., `file:/afs/unity/users/j/jqpublic/file.html` (or `file:/afs/eos...`).
**ExpressDocs Templates**

Templates that have already been designed and formatted are available from File -> ExpressDocs.

If you are not able to bring them up, select Preferences in the other program window and click the Files icon. Select ExpressDocs, and for the Default Directory, enter the path `/afs/bp/dist/wp70/wpexpdocs/`. Click OK. Then select ExpressDocs from the document File menu to bring up the following list of templates to choose from.

![ExpressDocs Templates](image)

**Printing Your Document**

Before you can print, you must have a print quota. To check your print quota, type `lpquota -q`. To print a document:

1. Select Print from the File menu, or click on the print button on the Power Bar. A pop-up window will appear.
2. Select the number of copies and which pages to print from the pop-up window and then click the OK button.

If you wish to print the entire document without choosing options, select Print Document from the File menu.

WordPerfect does not send the document directly to the printer. Rather, it prints to a PostScript file. You must then send the file to the printer with the `lpr` command typed at the command prompt.

The process is this: When you OK to print in the Print window, another smaller Destination Filename window pops up with the path to your print file (in your home directory). The print file has the same name as the WordPerfect file but with the .ps extension (or change the name or path as desired, but keep the .ps extension). Click OK to save your document as a PostScript file.
In an Xterm window at the prompt, type `lpr document.ps` to print your document on the local printer (see *Printing*).

**Tools**

WordPerfect has many useful tools to support your writing activities. It comes with a utility that underlines all misspelled words as you type them so that you will see to fix them as you go along. A standard Speller program is also available from *Tools*, as is a *Thesaurus* and the *Grammatik* grammar-checking program. WordPerfect 8 also has commenting, outlining, graphics, and equation-editing capabilities, among many other features.

**Getting Help**

WordPerfect contains its own online help. Open the *Help* menu at the top right of the document window and select *Contents* or “*How Do I...*” to get further information about performing word-processing tasks.

**Quitting WordPerfect**

To quit WordPerfect, go to the program window and select *Exit* from the *Program* pull-down menu.
All user volumes (home directories), including student volumes, are backed up nightly, both online and to tape. However, because every user’s disk space is a limited resource, users will need to store files on floppy or zip disks to get them out of their home directories and free up space. Good practice also dictates that files that are important to you should be backed up on a removable storage medium for safe keeping.

Preserving files that are very important (an end-of-the-semester project, dissertations, important research, etc.) is ultimately the responsibility of the users who create them. For this reason, users should get in the habit of regularly copying their files to disk. All workstations come with floppy and or zip drives for this purpose. On Solaris and Linux, there is a collection of tools called mtools for making backups. However, if you do lose a file, follow the procedure provided in this chapter to restore it.

**Backing Up Using MS-DOS Formatted Disks**

The recommended way to back up files on Eos/Unity is on MS-DOS formatted diskettes. This is particularly helpful to those people who have PCs or PC-compatible computers at home. It also means that the files can be read by something other than a UNIX workstation.

The process of backing up on a floppy disk involves two steps: formatting the diskette (if it is not pre-formatted) and writing data to it.

**Formatting a Floppy Disk (fdformat, mformat)**

To format a floppy diskette for MS-DOS at a UNIX workstation, insert a 3.5-inch diskette into the floppy drive. Then, at the prompt, type:

```
fdformat
```

By default, `fdformat` uses the configured capacity of the drive to format the diskette. A 3.5-inch high-density (HD) drive uses diskettes with a formatted capacity of 1.44 megabytes (MB), so a density option would not have to be specified (all Eos/Unity workstations have HD drives).

However, a density option must be specified when using a diskette with a lower capacity than the drive’s default, for example, a double-sided...
(DS) double-density (DD) diskette with a 720 kilobyte (KB) capacity. If you want to back up on a diskette with this capacity, you would need to run `fdformat` with the `-D` option, or, `fdformat -D` (see `man fdformat`).

After formatting the disk, you must prepare it to work with `mtools`, a public-domain collection of programs that allows UNIX systems to read, write, and manipulate files on an MS-DOS file system (typically a diskette). The `mformat` command adds a minimal MS-DOS file system to a diskette that has already been formatted by a UNIX low-level format, in this case, `fdformat`. To add the MS-DOS file system to the diskette:

```
mformat a:
```

**Writing Data to and from a Floppy Disk (mdir, mcopy)**

The following instructions tell how to copy from a UNIX directory on Eos/Unity to an MS-DOS formatted diskette in the floppy drive, and then from the diskette to the UNIX directory.

**Copying from UNIX to disk:**

With the floppy diskette in the drive, first change into the directory from which you wish to copy a file, and type:

```
mcopy unixfile a:
```

where `unixfile` is the name of the file on Eos/Unity that you want copied to diskette. This action copies the file using the same name it had on UNIX. If you want to give the file a different name on the diskette, then you type `mcopy unixfile a:dosfile`, where `dosfile` is the new name of the file.

To list the contents of the diskette in drive a:, type:

```
mdir
```

The command `mdir` reads the a: drive by default (to specify another drive, type the drive name after `mdir`, e.g., `mdir z:`).

To illustrate, if you `mcopy` the file `oddsends` to drive a: and then do a directory listing on that drive with `mdir`, you will see:

```
Directory for A:/

ODDSENDS            2080    5–19–01  11:33a
1 File(s)     1455104 bytes free
```

**Copying from disk to UNIX:**

With the floppy diskette in the drive, first change into the directory where you want the file to be copied. After you check the contents of the diskette in drive a: with the `mdir` command to find the file you want to copy, type:
In this case, you specify a new name for the “destination” file (*unixfile*). Or, you can copy the file with the same name to the current directory, e.g.,

```
mcopy a:oddsends
```

Make sure that there is no space between the `a:` and the source file. However, be sure to put a space between the source and destination files.

You can also use the wildcard `*` to copy files from one place to another. For example, if you want to copy all of the files in your current directory to the diskette in drive `a:`, type `mcopy * a:`, or just `mcopy *` (since the default is the `a:` drive). Or, if you want to copy to disk only those files in the directory with the `.html` extension, type `mcopy *.*.html a:`.

**Note:** If you are copying ASCII files from MS-DOS to UNIX, or vice versa (such as files created with `nedit` or `vi`), you may have trouble with linefeeds. UNIX and DOS have different ways of marking the end of a line in a text file. To get around the problem, add the `-t` option to `mcopy`:

```
mcopy -t a:dosfile eosfile
```

**Man Pages on Mtools**

Consult the man page on `mtools` commands as you need them, e.g., `man mdir`, `man mcopy`, etc. The man page on `mtools` itself will give you some idea of the commands in this tool set, including,

- `mattrib` – change MS-DOS file attribute flags
- `mcd` – change MS-DOS directory
- `mcopy` – copy MS-DOS files to/from UNIX
- `mdel` – delete an MS-DOS file
- `mdir` – display an MS-DOS directory
- `mformat` – add an MS-DOS filesystem to a low-level formatted diskette
- `mlabel` – make an MS-DOS volume label
- `mmd` – make an MS-DOS subdirectory
- `mrd` – remove an MS-DOS subdirectory
- `mread` – low-level read (copy) an MS-DOS file to UNIX
- `mren` – rename an existing MS-DOS file
- `mtype` – display contents of an MS-DOS file
- `mwrite` – low-level write (copy) a UNIX file to MS-DOS

**Problems and Error Messages**

If one of the `mtools` gives a message like:

```
.mcwd file is out of date
```

it is because `mtools` stores the path to the directory being worked in on a floppy in the `.mcwd` (current working directory) file in your home
directory. When you switch floppies, and this file contains a path that does not exist on the new floppy, `mtools` gets confused and gives you the “out of date” error message. The solution is to remove the `.mcwd` file.

```
rm ~/.mcwd
```

**Restoring Lost Files** *(from NC State Help Database document 1332)*

All user volumes (home directories) are backed up every 24 hours. Part of the backup process is the generation of backup volumes, or online backup copies. Backup volumes are generally created shortly after midnight, and they exist, unaltered, until the next time a backup volume is created. Backups are made from a snapshot of your files as they exist on or around midnight of that day. For example, files restored on 5/17/01 (Thursday) are as they existed when you quit working on Wednesday.

In order to restore a deleted file, the file must have been copied to backup at some time, since it is not possible to restore a file that was never backed up. In other words, if you delete a new file that you just created, you will not be able to restore it because it has not yet been copied to backup.

**Restoring Your Own Lost Files** *(fs examine, fs whichcell)*

In order to restore a deleted file yourself, the most recent backup volume must have the file in it, and you must restore it before the next backup volume is made (generally in the early morning hours). Otherwise, the user volume without the file will be on the next day’s backup.

**Step One: Getting Started**

Before you can do a restore, you must know:

- name of the file(s) you want restored
- location of the file(s) in your home directory
- name of your user volume
- name of your AFS cell

1. To identify your user volume name:
   
   `fs examine ~`

   which will output something like:

   ```
   Volume status for vid = 537008297 named users.n.nsj
   The name of this user volume is users.n.nsj.
   ```

2. To find out in which AFS cell your volume resides:

   ```
   fs whichcell ~
   ```
fs whichcell ~

which will output something like:

File /ncsu/nsj lives in cell ‘eos.ncsu.edu’

The cell in which this user volume resides is eos.ncsu.edu.

Step Two: Mounting Your Backup Volume

Use the cd command with no arguments to move to your home directory.

Next, “mount” your backup volume with the following command:

fs mkmount -dir mountpoint directory -vol volume.backup -cell cell

e.g.:

fs mkmount -dir backup -vol users.n.nsj.backup -cell eos.ncsu.edu

Step Three: Retrieve Your Files

After you have mounted your backup volume, cd into the mountpoint directory, or ~/backup You are now in the “root” of your home directory as it was at the time the backup volume was made. Find the file(s) you are looking for, and copy them out of the backup directory into your home directory, e.g.:

cp lostfile.doc ~

Repeat this step for each file you wish to restore, then cd back to your home directory.

Step Four: Unmount Your Backup Volume

To unmount the mountpoint directory:

fs rmmount -dir mountpoint directory

e.g.:

fs rmmount -dir backup

Requesting a Restore

If you are not successful doing your own restores, and the file you lost is very important to you, write to help@ncsu.edu and request a restore. Restoring files from backup is a labor-intensive and time-consuming process. As a result, system administrators are limited to performing only two restores per person. Make sure to provide your login name, the name of the file, the directory in which it resided, and the date when the file was last known to be on the system.
A staff member will attempt to locate a suitable copy of the file(s) and place it in a temporary directory away from your home file space. The path to the restored file will be e-mailed to you. You should examine the restored file, and if it is what you need, copy it from the temporary directory back to your home directory (and to a floppy disk!).

After approximately 10 days, the temporary restored copy will be removed from the system.

Please note that backups are only kept for 28 days. The computing staff cannot fulfill restore requests for data and files more than 28 days old.
Unix is composed of two main layers, a kernel and a shell. The *kernel* is the internal layer or core program that manages the computer system. Few users ever communicate directly with the kernel. Instead, users send instructions to the *shell*, an outer layer of software that interprets and executes commands, passing on to the kernel any further processing that may be required. For example, a user may give the `lpr` command to print a file, which the shell interprets and executes. However, the actual connection and transmission to the printer itself is handled by the kernel.

When you login, the operating system starts a C shell for you. The characters you type are entered into a *command-line buffer*. When you press the *RETURN* key at the end of a line, the operating system accepts the contents of the buffer as the command and then interprets it. If it cannot tell what the command means, then it delivers an error message back to the user. For example, if you type `copy` instead of `cp` when you try to copy a file, you will get the message

```
% copy: Command not found
```

A command to the shell, or *shell command*, actually tells the shell to run a program. The `cp` command tells the shell to run the copy program.

The user enters instructions and commands to the shell at the *shell prompt*. The percent character (%) in the prompt is not an arbitrary character. It stands for the particular kind of shell the system is running, in this case, a *C shell*, or `tsch`, the enhanced version of the Berkeley Unix C shell. If the symbol were a dollar sign ($), then it would mean that a *Bourne shell* is running.

More than one shell can run at a time. Each time the user opens an XTerm terminal window, a new shell is created to accept user input.

Your environment is determined by the behavior of the shell and other programs that interact with it. The shell has a facility for storing data in variables. There are two categories of variables: environment variables and shell variables. *Environment variables* are “global,” that is, the values or settings for these variables are the same throughout all the shells you create. *Shell variables* are specific to each shell; in other words, each window has its own shell variables that can be set.
Environment Variables (printenv, setenv)

Environment variables contain information about your working environment. These are set by system administrators but can be changed by the individual user. Environment variables determine the default operation of all shells and are also passed on to application programs. The list of environment variables is fairly lengthy and includes settings for a default printer, a default editor, a default path, etc. To list them, type the command `printenv`.

The format for setting the values of variables is the command `setenv` followed by the variable name in caps and then the value the variable should receive. Some values must be placed in quotation marks, particularly those that have embedded spaces.

Shell Variables (set)

Shell variables work like environment variables, except for the following differences.

- A set of shell variables is used by a single C shell only.
- Shell variables do not propagate to new shells or other programs.
- Shell variable names are written in lowercase.
- Shell variables are displayed, set, and changed with the `set` command.

Generally, shell and environment variables are independent, although they may look and behave similarly, both in name and purpose. Shell variables may also obtain their initial values from environment variables. However, once values are set for a shell, they will not be affected by changes in environment variables. Setting shell variables is a way of overriding environment settings in a shell.

To change a value for a variable, use the `set` command followed by the variable name (in lowercase), an equals sign (=), and then the value to be assigned to the variable.

For example, to replace `eos%` or `unity%` with `eos/unity%`

```bash
set prompt=eos/unity%
```

Or, to show the directory where you are working as a prompt:

```bash
set prompt="%/ 
```

Place the new `set prompt` command in your `.mycshrc` file to have it come up with every window (see Customizing).
CUSTOMIZING

Upon login, users will see the standard or default screen configuration, which is the same for all users (the look of this standard screen interface appears in The User Interface chapter). This standardization has its benefits. Because the standard configuration makes the software work in the same way across all platforms, many users never change it. In fact, the environment has been set up so that special customizing by the user is not required. Defaults are accepted standards that apply in most situations and work for most users.

However, as you come to use the system more, you may want applications to come up automatically rather than bringing them up from menus or the command line. You may also find that you like other colors and font sizes better. Or, you may simply want a more personalized screen setup with special graphics, windows, and icons that you select and arrange. Customizing the workstation is a way of improving and personalizing the workstation environment according to your own standards and giving you choice and flexibility in how you like to work.

Customizing the workstation is also instructive because it shows the user how to modify files that control how the computer works. Everything about the screen interface is set in one system file or another: the color of the background, the size of the windows, the prompt that comes up, etc. To modify this configuration requires rewriting existing configuration files or creating new files that modify or add to the configuration process.

CAUTION: Customizing may slow down your login and affect other things you do at the workstation. To safeguard against unexpected side effects, customize a little at a time to assess how your system is affected by the changes. Back up the original files, preferably in a separate directory. Also, make sure that you put the correct commands in the appropriate file. Use care when you work with the files described below. An error in writing the commands may adversely affect or crash your system.

ESCAPE: If one of your customized configuration files crashes your workstation or prevents you from logging in, select Session Type—> Default Environment or Repair Dotfiles on the opening login screen. This action will start a session that bypasses your customization files. Once in, you can correct the problem in your files.

Configuration Files

Your configuration files are dotfiles. Dotfiles are special files that are present in directories but not obvious. When you list the files in your
home directory with the `ls` command, the dotfiles that the system administration has already put there (.cshrc and .login) will not be listed with the rest of the directory’s contents. The period or “dot” before them indicates that they are hidden files, set off from other files because they contain special information. Dotfiles are only listed if you use the argument `-a` after the `ls` command: `ls -a`.

The .cshrc and .login files at the top of the list in the table below contain default settings and definitions set by the system administration. To change the default configuration, you do not alter these files themselves. Rather, you create some of the other files on this table to tell the system what to add, override, or adjust in these files. The table indicates the level of difficulty in modifying specific configuration files.

It may not be necessary to create all of the files on this list. You may find that ample customization can be done in one or two files only. The two files most often created to adjust the system configuration are the personal dotfiles .mycshrc and .mylogin. All personal configuration files are created and modified in a text editor of your choosing (see Text Editors) and should be placed in your home directory.

<table>
<thead>
<tr>
<th>Configuration Files</th>
<th>Contents</th>
<th>Ease of Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>.cshrc</td>
<td>default system C-shell run commands</td>
<td>do not change</td>
</tr>
<tr>
<td>.fvwm2rc</td>
<td>f virtual window manager run commands</td>
<td>medium</td>
</tr>
<tr>
<td>.login</td>
<td>default system instructions executed at login</td>
<td>do not change</td>
</tr>
<tr>
<td>.logout</td>
<td>user-created instructions executed at logout</td>
<td>easy</td>
</tr>
<tr>
<td>.mycshrc</td>
<td>user-created C-shell run commands</td>
<td>easy</td>
</tr>
<tr>
<td>.mylogin</td>
<td>user-created instructions executed at login</td>
<td>easy</td>
</tr>
<tr>
<td>.Xdefaults</td>
<td>Xsession resource file</td>
<td>medium</td>
</tr>
<tr>
<td>.Xstartup</td>
<td>user-created instructions executed at Xsession startup</td>
<td>easy</td>
</tr>
<tr>
<td>.xsession</td>
<td>default driver for workstation environment</td>
<td>difficult</td>
</tr>
</tbody>
</table>

**Levels of Session Customization**

Users customize to a greater or lesser extent depending on their needs. The preceding table gives some indication of the degree of difficulty the user encounters in working in each file (see Session Configuration for how these files are called during session startup). Users generally fall into one of three groups when it comes to defining their user environment.
Common customization files to create:

- logout
- .mycshrc
- .mylogin
- .Xdefaults
- .Xstartup

Files for substantial reconfiguration:

- .cshrc
- .login
- .fvwm2rc
- .xsession

Do not put commands in these files that require user input. No window is defined during shell startup in which the user could respond.

1. Default Users (no system experience required)
   These users accept without change the system configuration defaults set by the system administration. They will rely on the .cshrc and .login files already configured for them and placed in their home directories.

2. Customizing Users (moderate experience required)
   These users adjust the standard defaults–revise settings or start processes–but keep the standard order of events and basic session configuration. They are likely to create one or more of the files .logout, .mycshrc, .mylogin, .Xdefaults, or .Xstartup to handle their customization requirements.

3. Reconfiguring Users (substantial experience required)
   These users extensively revise the system defaults and the structure and sequence of the session. In addition to creating lengthier and more complex files than those of the customizing user, they might also create and/or adjust the files .cshrc, .login, .fvwm2rc, and .xsession.

Writing Configuration Files

This chapter addresses the needs of the customizing user principally; reconfiguring the workstation environment goes beyond the scope of this manual. The following guidelines should help you with placing commands and settings in the appropriate configuration file.

- Place in the .mycshrc file all statements that set environmental variables (command `setenv`) and statements that set shell variables (command `set`). Put statements that set aliases (command `alias`) or change the standard path (command `set path`) in this file also.

- Place all statements that set colors, appearances, and other attributes of X applications in the .Xdefaults files. This file is only read in an X session.

- Place all non-X applications and statements that attach lockers (command `attach`) in the .mylogin file, which is only read during a TTY session (remote or dialup session, see following chapter).

- Place all commands that start up new windows and X applications in the .Xstartup file. Be sure to place an ampersand (`&`) after the application name in this file so that the program runs in the background.

- Place all commands appropriate for a remote or dialup TTY session in the .mycshrc or .mylogin files. A TTY session does not run .Xdefaults or .Xstartup (see following chapter).
Most customization files are plain text files that list a valid command on each line. The following sections offer guidelines for constructing and modifying the four easiest dotfiles to create: .mycshrc, .Xdefaults, .mylogin, and .Xstartup. Follow closely the procedures recommended for modifying these files, and pay particular attention to the cautions and warnings issued.

**CAUTION:** All users must use care in creating and altering any of these dotfiles. In general, do not delete or copy over any dotfiles you find in a directory. Do not copy over, rename, or delete .login and .cshrc, and do not copy them to your .mylogin or .mycshrc files. Reconfiguring users must be certain that they know what they are doing before altering system files. Most users should leave the files .cshrc, .login and .xsession alone.

### The .mycshrc File

The .mycshrc (my c-shell run commands) file is read each time a new shell (Xterm window) is created. As a result, you would place in this file the commands you want executed each time an Xterm window is opened.

You typically modify environment and shell variables in .mycshrc. The following command changes the shell prompt to *You rang?*: 

```bash
set prompt='You rang? '
```

Single quotes are placed around the new prompt in this statement because it contains blank spaces. The quote marks will not appear when the prompt comes up. If you want some space between the prompt and the commands you enter, add a space or two before the closing quote.

If you want the prompt to always show the path you are in:

```bash
set prompt="/%/ %"
```

Another command often found in this file is the alias command, which changes a default specification to something shorter or easier to remember. The following command creates a shorter alias, the word *eos*, for the longer string, `cd /afs/eos/www/eos/`. Once this line is in the .mycshrc file, all the user needs to type to change into this directory is *eos* (see The .mylogin File section that follows).

```bash
alias eos cd /afs/eos/www/eos/
```

Or, suppose that you have come to Unix from a MS-DOS environment and are accustomed to using the command *dir* instead of the Unix *ls* to show a directory listing. The following alias in the .mycshrc file will permit you to substitute the DOS *dir* command for its Unix equivalent.

```bash
alias dir ls
```

A sample .mycshrc file containing these commands appears below. An Xterm window is also displayed to show what these commands produce.
You will be able to see the changes you make to your `.mycshrc` file when you bring up a new Xterm window (but be sure to debug the file also, see last section).

---

**The `.Xdefaults` File**

The `.Xdefaults` file is only read in an X Windows session, not when you login remotely. This dotfile holds the resources for programs on the system that use X output, keyboard focus, and other information. The system’s standard `.Xdefaults` file is located in `/usr/lib/X11/xdm`. This file is read during X session setup and modified by the `.Xdefaults` file in your home directory, if you have one.

X-based applications read the `.Xdefaults` file during startup and use the appropriate resource specifications to customize the appearance or characteristics of their windows and icons (size, shape, color, etc.).

The format of a typical specification in the `.Xdefaults` file is:

```
name*resource:  value
```

The format of a typical specification in this file is `name*resource: value`. The `name` of the specific X application to be affected is written first. It is followed by the particular `resource` within the application that you want to change or set a value for. The `value` or specification for the resource is written last, generally after a colon. If no application name is given, the system assumes that you want all X applications with that resource to receive the value you specify.

`Resource` is the name for a broad category of items, from fonts and colors to windows and buttons, essentially anything in an application that can be modified. The asterisk (`*`) that you see in many specifications is a delimiter similar to a wildcard. It indicates that the specification applies to names and applications that have not been listed.
The first specification in the `.Xdefaults` file in the next figure makes “wheat” (value) the color of the background (resource) in all X applications (name=*). This specification will change the default background color (which is a dark blue) to a lighter wheat-colored background in all applications. The color must be written exactly as it is in the showrgb palette.

The second line changes the size of the screen font, that is, the size of the letters and characters that appear in an xterm window. Xterm is the name of the application affected; the font is the resource to be altered; and the font size (10x20) is the value the resource will be given, in this case 10 pixels by 20 pixels.

The third line modifies the color of the mouse cursor inside one application, xterm, to show how specific you can get with your customizing. In all resource specifications, you must use the precise X terms, spellings, spacing, and case if they are to work. If you want a pale turquoise cursor, then you must write it as paleturquoise [sic].

You will not see the results of additions or changes you make to your `.Xdefaults` file until you logout and login again. Therefore, be sure to debug the file before you logout (see last section in this chapter). Or, merge the new changes with the command

    xrdb -merge .Xdefaults

### The `.mylogin` File

The commands in the `.mylogin` file are executed each time you login, whether you are logging in directly from a workstation or remotely.

However, commands that call up X windows resources and applications should not be placed in the `.mylogin` file because a remote machine generally does not have the software to handle the graphical interface of

<table>
<thead>
<tr>
<th></th>
<th>File</th>
<th>Edit</th>
<th>Search</th>
<th>Preferences</th>
<th>Shell</th>
<th>Macro</th>
<th>Windows</th>
<th>Help</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>background:</em></td>
<td>wheat</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>xterm*font:</td>
<td>10x20</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>XTerm*cursorColor:</td>
<td>paleturquoise</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To list the available colors:

    showrgb | more

To view colors:

    add goodies
xcoloresel

To see a list of available fonts:

    xlsfonts | more

For more X resources and help using them:

    man X
X Windows. As a result, you are likely to put in .mylogin only those commands that both a local and remote login would be able to execute.

The command generally placed in this file is the attach command. For example, if you frequently use the resources in the imagetools locker, you may want to add a command to .mylogin that attaches this locker each time you login.

If you have other lockers you want to attach, you do not have to write a separate command for each of them but can simply add them to the line.

**You will not see the results of additions or changes you make to your .mylogin file until you logout and login again. Therefore, be sure to debug the file before you logout (see last section).**

---

### The .Xstartup File

The .Xstartup file is only read in an X session when you login directly. It will not be read when you login remotely. The .Xstartup file starts up X Windows and applications, such as your Xclock and Xterm applications. It also changes the attributes of these windows, their size, placement on the screen, etc. In addition, commands to bring in screen graphics and images, such as those that appear in the root window, are usually placed in this file.

The following figure shows a sample .Xstartup file at the top and the screen configuration it produces below it. Its commands are a little more complex than the commands listed in the .mylogin and .mycshrc files discussed above. Not only does it list commands to be executed during session startup, but it also specifies the screen coordinates where the applications or their icons should appear, how long the system should wait before bringing them up, the path(s) to specific files, etc.

The first command in the sample .Xstartup file brings an image into the root window of the screen. It specifies the command (**xloadimage**), followed by an option that tells the system where to put the image (**-onroot**), ending with the absolute path to the moon.xbm.Z bitmap image.
Do not forget to add the ampersand (&) after the path to background the process.

The .Z added to the file name means that the file is compressed. The xloadimage command will automatically uncompress the file before displaying it, so you do not have to run the uncompres command.
If you want to view an image before you put it in your .Xstartup file, you can use the xloadimage command followed by the image you want to display to see it inside a window.

The next line of the .Xstartup file opens up the Xclock application and also sizes and places it on the screen. The application is given a geometry (option -geometry) of 100x100 pixels size and a screen location (780+42).

The more applications you bring up, the longer the session startup takes. Also, if you have a number of applications coming up at once, you can have initialization problems. The sleep command listed after each application in this file tells the system to wait a certain amount of time (in this case, 5 seconds) before launching the next application. This keeps the applications from competing for resources (memory, cache, etc.) when they come up.

Big applications may need a longer sleep (15 seconds or so) to get started. Although the sleep command appears to be adding time to the session startup, it is in fact reducing the total startup time by making the system work more efficiently in bringing up each application.

**fvwm2 Window Manager**

The default window manager on Eos and Unity is fvwm2, F(?) Virtual Window Manager (version 2) for X11. In 2002, the default window manager was changed from the Motif Window Manager (mwm) to fvwm2 to minimize memory consumption, provide a 3D look to window frames, and a virtual desktop. Substantial information is available on the man page for fvwm2.

```
add wm
man fvwm2
```

When you log in, the default.xsession checks to see if you have the WINDOW_MANAGER variable set. If you have not changed the default window manager to something else, it will find WINDOW_MANAGER=/usr/fvwm/bin/fvwm2 (setenv will show you your settings).

If you wish to customize the fvwm2 window manager, copy the system.fvwm2rc file to your home directory as .fvwm2rc.

```
cp /usr/fvwm/share/fvwm/system.fvwm2rc ~/.fvwm2rc
```

Then edit the file in a text editor. When you finish your changes, select **Restart fvwm2** from the Root Menu (hold down third mouse button in root window) to see the modifications.

To restore the system default, remove the .fvwm2rc file from your home directory.
If you wish to change your window manager to something besides fvwm2, add the following line to your ~/.mycshrc file:

```
setenv WINDOW_MANAGER /afs/bp/contrib/wm/bin/name_of_wm
```

The following command will add the window manager locker, and you can view and try out window managers:

```
add wm
cd /ncsu/wm/bin
```

### Debugging and Fixing Configuration Files

The `source` shell command can be used to debug configuration files. This command runs a program that checks the commands and syntax in the file to verify that they are correct. This debugging routine can aid the user who does not wish to logout and login to view changes in a configuration file.

For more help on customizing, consult the man page on X (`man X`).

To debug a dotfile:

```
source filename
```

For additional information on customizing:

```
man X
```

To fix dotfiles:

```
init_dotfiles
```
SESSION CONFIGURATION

The Eos/Unity system administration puts the configuration source files `.cshrc` and `.login` in your home directory. These files (which should not be deleted, changed, or copied) transfer control to the system files `cshrc` and `login` in `/usr/athena/lib/skeleton` to create the default configuration of the user’s session at the workstation.

The file `.login` is read only once when you login to the system. The file `.cshrc` is read whenever you activate a shell (tcsh), that is, whenever you open an Xterm window. However, before bringing up the default configuration, the system checks your home directory for any additional configuration files you might have created and executes the customizing commands in those files as well. Thus, customizing is an “add-on” procedure, adding steps to the session startup that modify the system defaults.

Users login either directly or remotely. Thus, there are two kinds of sessions, each created by a different sequence of events.

- **The direct-access X Windows session.** Users who login directly to the network use a campus workstation running X Windows. This session supports a window manager, simultaneous activity in multiple windows, and many sophisticated support and applications programs.

- **The remote access TTY (Terminal Teletype) session.** Users who access the network remotely generally log in from home via dialup, ssh/puTTY, or telnet. This terminal session is generally not able to run X Windows. The user usually runs activities one after the other rather than simultaneously.

In both sessions, the system reads the system and personal configuration files in a specific order. There are two kinds of variables set by these files:

- **environment variables** are general settings that stay the same for all the shells you create and get passed on to programs and application software;

- **shell variables** are specific to each shell, that is, each shell may have its own specific configuration that overrides the environment settings in that particular window.
In addition, these files handle device setup and characteristics as well as the initialization of processes that make the workstation or terminal run properly. Specific user-defined activities are also begun and terminated in both sessions during session startup and termination, respectively. The following files are the personal configuration files that are called during session configuration (if they exist).

- **The ~/.xsession File**
  The ~/.xsession file contains system driver commands for the Eos workstation environment, such as the startup message in the console window and calls to the window manager. To create a ~/.xsession file, the user copies the system’s .xsession file to his/her home directory and then adjusts it there.

- **The ~/.cshrc File**
  The ~/.cshrc (c-sh ell r un commands) file is read each time a new shell (Xterm window) is created. Because this file contains complex default system commands, modification of this file is not recommended. Users should place modifications in a ~/.mycshrc file, which ~/.cshrc calls.

- **The ~/.mycshrc File**
  The ~/.mycshrc (personal my c-sh ell r un commands) file contains user-created commands for execution and is read each time a C shell (Xterm window) is created.

- **The ~/.Xdefaults File**
  The ~/.Xdefaults file is only read during an X Windows session, not in a remote TTY session. X-based applications read the file during startup and use the appropriate resource specifications to customize the appearance or characteristics of windows and icons (e.g., size, shape, color, decoration).

- **The ~/.fvwm2rc File**
  During initialization, fvwm searches for the configuration file .fvwm2rc in ~/.fvwm or your home directory. If it fails to find this file, it looks for system.fvwm2rc, which is in /afs/bp/contrib/wm/configs/system.fvwm2rc. To customize, copy system.fvwm2rc to your home directory and call it .fvwm2rc. Then configure it using the man page to help you (add wm, man fvwm2).

- **The ~/.login File**
  The ~/.login file is only read during a remote TTY login. Because this file contains standard default system commands, modification is not recommended. Users should place modifications in a ~/.mylogin file, which is called by the ~/.login file.

- **The ~/.mylogin File**
  The ~/.mylogin file is executed at login, whether a user is at a
workstation or connected remotely. However, since a remote machine generally cannot handle the graphical interface of X Windows, a ~/.mylogin file should contain only those commands that both a local and remote login can execute.

- **The ~/.Xstartup File**
  The ~/.Xstartup file is only read at the beginning of an X Windows session, not over a remote connection. ~/.Xstartup typically contains commands to start X resources, windows, and applications, although it will execute virtually any command that can run from the shell prompt.

- **The ~/.logout File**
  The ~/.logout file is read at logout and contains commands the user wishes executed immediately before logout.

---

**The X Windows Session Configuration**

Xsession, which is invoked by the X Display Manager (xdm), resides in /usr/lib/X11/xdm and begins a standard X Windows session. The system looks to see if the user has a ~/.xsession file and will use it if he/she does. If there is not a user file ~/.xsession, the system uses its own .xsession file (in /usr/lib/X11/xdm).

The .xsession file is an executable shell script that starts up an X11 session for the user. The script contains a number of X11 commands that configure the layout of the screen and windows, call files and X applications, and define the sequence of events in the session, i.e., setup, startup, and termination.

The .xsession file also calls ~/.cshrc, which is placed in the user’s home directory by system administrators (every user has a ~/.cshrc file). The ~/.cshrc file sets environment and shell variables, aliases, file protection, prompts, paths, etc., and then calls and uses additional settings in the system’s /usr/athena/lib/skeleton/cshrc source file.

The system’s cshrc file then calls the user’s ~/.mycshrc file, if there is one, for any adjustments or customization the user wants made. The user’s ~/.mycshrc overrides any system defaults or settings and is read each time a new C shell is created.

The X session also calls the user-defined ~/.Xdefaults, ~/.mylogin, and ~/.Xstartup files, if they exist. These files generally adjust X default settings and start up specific X applications, respectively. In addition, the X session starts up the Window Manager (fvwm2) using the system’s system.fvwm2rc file or the user’s ~/.fvwm2rc file, if he/she has one.

**The X session does not make use of the ~/.login file in the user’s home directory.** That file is only used in a remote session. However, the
# Configuration Sequence for an X Windows Session

## Session Startup

### Environment and Shell Setup

- `$SYSTEM/Xsession in /usr/lib/X11/xdm` runs setup files to configure environment.
- `%SESSION/.xsession in /usr/lib/X11/xdm` starts Window Manager (fvwm), making custom adjustments if ~/.fvwm2rc exists, otherwise using system.fvwm2rc.
- `$SYSTEM/.xsession in /usr/lib/X11/xdm` starts initial Xterm.
- `$skeleton/cshrc in /usr/athena/lib/skeleton` is started, calling ~/.cshrc, cshrc, and ~/.mycshrc each time.
- `.cshrc` is executed.
- `.mycshrc` is executed.
- `.Xdefaults` is executed.
- `.fvwm2rc` is executed.
- `.logout` is executed.
- `.mylogin` is executed.
- `.Xstartup` is executed.
- `.logout` is run.
- `.Xsession` runs custom settings from ~/.Xdefaults, such as the pathname for the resource description file (~.fvwm2rc), keyboard focusing policy, etc.
- `.Xsession` runs custom X activities specified in this file.
- `.Xsession` runs this file’s custom logout procedures.

## Session Termination
session does call the user-defined ~/.mylogin file, if the user has one, and executes any commands in this file.

Finally, the X session shuts down the session, running the ~/.logout file first, if there is one, and then running the system default logout and cleanup procedures.

**The TTY Session Configuration**

A remote terminal (TTY) session occurs when you login to the Eos/Unity network via a PC and modem, telnet, dialup0, etc. The TTY session events proceed as they are depicted in figure to the right. The environment and shell variables are set first, with any user customization done in the ~/.mycshrc file. The settings in this latter file override any system settings. These system and user cshrc files are read each time a new C shell (Xterm) is created.

The user’s ~/.login file is called next. This file is placed in the user’s home directory by system administrators. This file is only called during remote login, never in an X session, and is only read once, not each time a C shell is created. The ~/.mylogin file, if there is one, customizes the login with any user-defined adjustments and additions. The TTY session cannot read X files, such as ~/.Xdefaults or ~/.Xstartup.

In session termination, the TTY session uses system-defined logout and cleanup procedures, unless there is a ~/.logout file, which it will run before activating the system’s standard procedures for logging out.
### Configuration Sequence for a TTY Session

<table>
<thead>
<tr>
<th>Session Setup</th>
<th>System Configuration Files</th>
<th>Personal Configuration Files</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Environment and Shell Setup</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>~/.cshrc</td>
<td>The user’s ~/.cshrc calls the system’s cshrc. These files set up the default shell and environment configuration.</td>
<td>~/.mycshrc</td>
</tr>
<tr>
<td>$skeleton/cshrc in /usr/athena/lib/skeleton</td>
<td>System cshrc calls ~/.mycshrc for custom adjustments to the default environment and shell configuration.</td>
<td></td>
</tr>
<tr>
<td>Shell setup is done each time a new C shell (Xterm) is started, calling ~/.cshrc, cshrc, and ~/.mycshrc each time.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Device Setup</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>~/.login</td>
<td>~/.login runs remote-access TTY activities. It calls the system login file for other initialization procedures, such as setting device characteristics (VT100 emulation).</td>
<td></td>
</tr>
<tr>
<td>$skeleton/login in /usr/athena/lib/skeleton</td>
<td></td>
<td></td>
</tr>
<tr>
<td>The .login files are run once a session, so the setup they do is done one time only. The system login file calls the ~/.mylogin.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Session Startup</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Runs standard startup activities (checks mail, etc.)</td>
<td>~/.mylogin</td>
<td></td>
</tr>
<tr>
<td>~/.mylogin runs the custom non-X startup activities specified in this file.</td>
<td></td>
<td></td>
</tr>
<tr>
<td><em>Session Termination</em></td>
<td></td>
<td></td>
</tr>
<tr>
<td>The system runs standard logout activities.</td>
<td>~/.logout</td>
<td></td>
</tr>
<tr>
<td>Runs custom logout activities and procedures (non-X).</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
MANAGING YOUR ACCOUNT

This manual has attempted to guide and enhance your use of the Eos/Unity computing system at NCSU. In many parts of this manual are procedures recommended to you as ways to manage and protect your work. This chapter summarizes some of those recommendations and supplies additional instructions to help you manage your account.

Regular Checks

Regularly check your disk and print quota and directory permissions:

- `quota` Checks your home directory disk quota
- `fs lq` Shows quota usage of current volume
- `lpquota -q` Checks your print quota and charges
- `lpquota -l` Lists the jobs you have printed in the year
- `fs la ~` Checks permissions on your home directory
- `fs la .` Checks permissions on your current directory

Managing Your Disk Quota

When you have used up your 50 MB disk quota, you will start getting error messages, such as not being able to save the file you are working on or not being able to access an application. If after checking your quota you find that it is filled, go through your directories and get rid of files you do not need (`rm filename`). The chapter Backing Up and Restoring Files tells how to move files that you want to keep but are no longer using onto floppy disks; you can copy them back later as needed.

Run the `du` program to see what is taking up room in your account. You can go through and remove backup files, except for anything you are working on currently. Get rid of large files, duplicate files, mp3s, images, etc. Executable files created from computer programs take up a lot of space and should be removed if they are not currently in use. You can always recreate an executable by recompiling the program. Just remember to keep the original program. Also, get rid of old mail messages, particularly attachments.

You can also compress files to save room with the `compress` command (the files produced have `.Z` extensions). The command `uncompress` takes them out of compressed mode so that you can use them again.
Managing Your Print Quota

WolfCopy Centers (wolfcopy@ncsu.edu) maintain the printers and also handle printing purchases. Conserve money and paper by using the NXpr Print Manager when possible, and also follow the advice for managing printing at the end of the chapter on Printing. Print in small jobs and only the pages you need. Get used to previewing and editing your documents online, and stop unwanted jobs from printing (lpq, lprm job).

Changing Your Password

Change your password regularly at http://www.ncsu.edu/password/. But please remember it! If you forget your password, you must contact a system administrator to reset it. Follow the recommendations for choosing a secure password in the chapter Password Changes.

Protecting Your Account

Do not give out your password to anyone! This is the main thing you can do to protect yourself and your resources. Remember that if someone else has your password, s/he can use your disk space and print quota. That person can also access your mail and Zephyr communications, send them out in your name, give out your password to others, or reset the permissions on your directories. You also leave your files vulnerable to theft and possible corruption.

Do not let anyone talk you into sharing your password. Account sharing is against policy, and if you are the one loaning your account, then you are the one who risks losing it. Also, the equipment purchased with student fee money is enough to support fee-paying users only. If users give out their passwords, then non-paying users will be taking equipment and resources away from the people who are paying for them (see Policies and http://www.ncsu.edu/it/rulesregs/).

Also, do not leave your workstation unattended! Not only does it tie up a station that someone else could be using, but it also leaves your account open and vulnerable.

Sharing Information but not Your Password

Users can grant other people access to specific directories (but not to individual files) by using the fs command. However, because access privileges are inherited from parent to child subdirectories, users need to remember that after changing a directory’s access, all new subdirectories created under it will inherit those settings. For this reason, users should not grant more than lookup access to their home directories, the top level of their lockers. If they do, they grant another user substantial access to all new subdirectories and files they create in their home directories.

To remove a print job from the queue:

 lpq
 lprm jobnumber

See your print quota at http://www.ncsu.edu/lpquota/

To change a password, go to http://www.ncsu.edu/password/

For more on Account Policies:
Student http://www.ncsu.edu/it/rulesregs/caccounts/student-maint-sop.html
Faculty http://www.ncsu.edu/it/rulesregs/caccounts/facstaff-maint-sop.html

To check permissions:

 fs la directoryname

To set permissions:

 fs sa dir username access

where dir is the name of the directory to which access is granted;

where username is the login name of the user to whom access is granted;

where access is the privilege(s) granted to username: read, look, insert, delete, write, lock, or administer (rlidwka).
One strategy for sharing information is to create a subdirectory called `public` under your home directory (mkdir `public`) and then change its `system:anyuser` setting to `rl` (fs sa ~/public system:anyuser rl). You can then use this subdirectory as the place where you copy files (cp `filename` ~/public) that you want to make public to others. The `rl` access gives any user the right to read and copy the files in this directory, but not to change or delete them. Such a directory would cover most circumstances where you want to put files out so others can read and copy them. This keeps you from having to set permissions often or remember who has access to your various directories.

If you create a ~/public directory in this way, make sure that your home directory has lookup access for `system:anyuser` (which would give any user the right to list (ls) the files in your home directory and all new directories created under it, but not to copy, open, delete them, etc.).

The chapter AFS File-Sharing Commands explains more about how to set permissions on directories that you want to share.

Some people may also have access to public lockers where they can place and share files. For organizations and classes, public lockers serve as a workspace for people to share and post information. These public lockers can only be created by staff in Computing Services and the various colleges, including the College of Engineering ITECS office.

### Using Care with `add` and `attach`

When you want to access a public locker or another user’s directory, you can get to that directory with a command like cd /afs/eos.ncsu.edu/info/eos_info or cd /afs/eos.ncsu.edu/users/j/jqpublic. These absolute pathnames are long and hard to remember, though, and ultimately can be avoided if the user first attaches the directory (attach eos_info or attach jqpublic). After an attach, the user can use the abbreviated directory pathname, /ncsu (cd /ncsu/eos_info or cd /ncsu/jqpublic). The /ncsu substitutes for the longer pathname and streamlines directory access.

The `add` command does more than `attach` and is used when you want to access and run application software. Adding a directory allows any startup programs to run that are part of the software in that directory.

**You should not add user directories!** Because of the way `add` works, if you add a user directory that has a `.environment` file in it, that file is “sourced”; in other words, the commands in that file are executed by default. ITECS has received reports of people writing `.environment` scripts that give them access to the files of users who `add` their directories. An unfriendly `.environment` script could re-set the access control list (ACL) on your home directory, or even remove your access to your own home directory. As a result, when you have permission to access another user’s directory, use the `attach` command instead (and regularly check permissions on your home directory).
Your Personal Web Page

You can create your own Web site in ~/www, which is found and posted to the Web automatically as http://www4.ncsu.edu/~username. Follow the instructions in the World Wide Web chapter for creating your Web site. Among the documents you may put there is your resume. Please read the computing policies before you begin developing the space (see Policies).

Finding Other Users

To find people on the Eos/Unity system, you can use one of several commands: hes, whois, and zlocate. In some cases, you search on the username of the person, as in hes username and zlocate. With whois, you must know the user’s full name.

For example, if you want to find out the whole name of the person whose username on the system is mcdaniel, type hes mcdaniel. The Hesiod database returns the following information:

PASSWD: mcdaniel:*:3693:108:Ellen Mcdaniel:/ncsu/mcdaniel:/bin/tcsh
FILSYS: AFS /afs/eos.ncsu.edu/users/m/mcdaniel w /ncsu/mcdaniel
POBOX: POP uni02mh.unity.ncsu.edu mcdaniel

This information shows the user’s name, location of user file space, and the mail server she uses. It also shows what access groups she is in. The userid number is here also. This number rather than the username will display in directory listings and other locations. You can hes a userid number just as you can a username to see who it is, e.g., hes 3693.

whois and zlocate are two other ways to locate people. With whois, you need to know the user’s full name. zlocate user zephyr to find out where someone is logged in.

% whois mcdaniel, ellen

1 matches found for your keyword.

<table>
<thead>
<tr>
<th>Name</th>
<th>Mcdaniel, Dr. Ellen</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uid/Key</td>
<td>3693</td>
</tr>
<tr>
<td>Campus Address</td>
<td>255 Page Hall, Box 7901</td>
</tr>
<tr>
<td>Campus Phone</td>
<td>919–515–8102</td>
</tr>
<tr>
<td>Campus Fax</td>
<td>919–515–7463</td>
</tr>
<tr>
<td>Department</td>
<td>Info Tech &amp; Engineering Computer Services</td>
</tr>
<tr>
<td>Position</td>
<td>Manager, Web And User Services</td>
</tr>
<tr>
<td>E-mail Address</td>
<td><a href="mailto:mcdaniel@ncsu.edu">mcdaniel@ncsu.edu</a></td>
</tr>
</tbody>
</table>

% zlocate mcdaniel

bizarro.eos.ncsu.edu :0.0 Mon Jun 25 08:28:02 2001
More information about users is online on the Web. On the NCSU home page at http://www.ncsu.edu/, the **Directories** link will take you to telephone and mail directories for people and departments. The home page is a good starting place to find out about people, classes, and activities at NCSU, so point your Web browser there and begin exploring.

Keep Telecommunications and Registration and Records updated with your information as well.

Students update their information and set preferred email addresses at:

https://www.acs.ncsu.edu/reg_records/tracs_lk/trc_frm.html

Faculty and staff update their information and set their preferred email addresses at:

https://ncsweb1.fis.ncsu.edu/DirUpdate/
APPENDIX A:
TROUBLESHOOTING
Trouble Shooting Problems (Unix workstations)

Login Problems

<table>
<thead>
<tr>
<th>Screen is dim (black)</th>
<th>Is screen-saver on?</th>
<th>Move mouse or touch keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Is monitor and/or system unit on? (Check indicator lights)</td>
<td>Turn power on (see Hardware Problems)</td>
</tr>
</tbody>
</table>

User cannot login

<table>
<thead>
<tr>
<th>Is username unknown?</th>
<th>Try first &amp; middle initial and first six letters of last name without spaces</th>
</tr>
</thead>
<tbody>
<tr>
<td>Is password unknown?</td>
<td>Try student ID number without spaces or dashes</td>
</tr>
<tr>
<td>Is username and password known?</td>
<td>Possible dotfile problem; login with Default Environ. Not enrolled and/or paid fees</td>
</tr>
</tbody>
</table>

Screen displays >> or >

| Reboot workstation (see Hardware Problems) |

Login screen is obscured with messages

| Click background or Clear button in login box |

General Problems

User receives message “Unable to attach your home directory”

File server may be down; login later Report to operator or help@ncsu.edu

No windows, or problems with windows

| Configuration files recently modified? | Login with Default Env. or Repair Dotfiles option; correct dotfiles |

prompt replaced by >

| Does pwd show user in tmp directory? | Server may be down Wait and login again or type kreset for more tokens |

| Dotfiles missing from home directory? | Log back in with Repair Dotfiles option or type init_dotfiles |

<p>| Dotfiles present in home directory? | Type detach username and logout and login again; or report to operator |</p>
<table>
<thead>
<tr>
<th>Issue</th>
<th>Possible Cause</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prompt disappears from command line</td>
<td>Application running without ampersand?</td>
<td>Use <code>CTRL z</code> to suspend process and return prompt; use <code>bg</code> to move process to background</td>
</tr>
<tr>
<td>Cannot open or save files</td>
<td>Is user’s disk quota full?</td>
<td>Recover disk space by deleting files</td>
</tr>
<tr>
<td></td>
<td>(check with <code>quota</code>)</td>
<td>Check file size with <code>du</code> command</td>
</tr>
<tr>
<td></td>
<td>Is the workstation disk quota full?</td>
<td>Login again; report to operator or <a href="mailto:help@ncsu.edu">help@ncsu.edu</a></td>
</tr>
<tr>
<td></td>
<td>(check with <code>df /tmp</code>)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Is disk space available?</td>
<td>Check console window for messages about possible server problems</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Report to operator or <a href="mailto:help@ncsu.edu">help@ncsu.edu</a></td>
</tr>
<tr>
<td></td>
<td>Does user have tokens?</td>
<td>Reset tokens with <code>kreset</code></td>
</tr>
<tr>
<td></td>
<td>(check with <code>tokens</code>)</td>
<td></td>
</tr>
<tr>
<td>Cannot receive Zephyr messages or be located with <code>zlocate</code></td>
<td>Is user “hidden?”</td>
<td>“Unhide” user with <code>zctl unhide</code> or <code>zctl set exposure realm-visible</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Is zwgc running?</td>
<td>Restart with <code>/usr/athena/etc/zwgc</code></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Note: do not add ampersand (<code>&amp;</code>)</td>
</tr>
<tr>
<td>Application does not appear on screen after launching</td>
<td>Have you waited more than several minutes (some programs take a long time to load). Do you have other large programs running?</td>
<td>See console and <code>sysnews</code> for messages about license or server problems Reduce number of processes running Do not keep launching application! Report to <a href="mailto:help@ncsu.edu">help@ncsu.edu</a></td>
</tr>
<tr>
<td>Cursor disappears or application “locks”</td>
<td></td>
<td>Wait a couple minutes for application to finish process. Otherwise, close window to exit application</td>
</tr>
<tr>
<td>Application does not disappear after closing</td>
<td></td>
<td>List processes (<code>ps -e</code>) and <code>kill</code> process number to shut down application</td>
</tr>
<tr>
<td>Screen is cluttered with ghost images from multiple windows</td>
<td></td>
<td>Select Refresh from the Root Menu to redraw windows</td>
</tr>
<tr>
<td>License expired messages?</td>
<td></td>
<td>Report to <a href="mailto:help@ncsu.edu">help@ncsu.edu</a></td>
</tr>
</tbody>
</table>
## Hardware Problems

<table>
<thead>
<tr>
<th>Problem Description</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power anomaly restarts workstations</td>
<td>Type <code>b</code> at &gt; prompt, wait 3 minutes or <code>boot</code> at “OK” prompt Stop+ a and boot</td>
</tr>
<tr>
<td>Is system unit a Sun?</td>
<td></td>
</tr>
<tr>
<td>Mouse, keyboard, monitor, system unit, or printing hardware problems</td>
<td>Report problem to operator, or send e-mail to <a href="mailto:help@ncsu.edu">help@ncsu.edu</a>, or call 515-HELP</td>
</tr>
<tr>
<td>Printer stops working (e.g. low toner, paper jam, etc.)</td>
<td>Report problem to operator, or call WolfCopy at 515-2131, or send e-mail to <a href="mailto:wolfcopy@ncsu.edu">wolfcopy@ncsu.edu</a></td>
</tr>
</tbody>
</table>

## Print Problems

<table>
<thead>
<tr>
<th>Problem Description</th>
<th>Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td>User cannot print (check with <code>lpquota</code>)</td>
<td>Purchase additional print quota at Wolfcopy Copy Center</td>
</tr>
<tr>
<td>Is print quota empty?</td>
<td></td>
</tr>
<tr>
<td>Print quota is not empty.</td>
<td>Check print queue (<code>lpq</code>) to see if other jobs are ahead of yours;</td>
</tr>
<tr>
<td></td>
<td>Report to operator or <a href="mailto:help@ncsu.edu">help@ncsu.edu</a> if print queue stops</td>
</tr>
<tr>
<td>Are tickets and/or tokens valid?</td>
<td>Reset with <code>kreset</code></td>
</tr>
<tr>
<td>File accidentally sent to printer</td>
<td>Type <code>lp</code> and <code>lprm job#</code> to remove jobs from print queue. Check <a href="http://print.ncsu.edu">http://print.ncsu.edu</a></td>
</tr>
<tr>
<td>Printer stops working (e.g. low toner, paper jam, etc.)</td>
<td>Report problem to operator, or call Wolfcopy at 515-2131, or send e-mail to <a href="mailto:wolfcopy@ncsu.edu">wolfcopy@ncsu.edu</a></td>
</tr>
</tbody>
</table>
APPENDIX B:
SUMMARY OF COMMANDS
## Unix and Local Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>add</td>
<td>displays a list of all the software the user can bring up at the workstation</td>
</tr>
<tr>
<td>add <em>software</em></td>
<td>loads a software package</td>
</tr>
<tr>
<td>alias <em>alias oldname</em></td>
<td>gives an alias or new name (usually shorter) to a command, path, etc. (e.g., alias <em>dir</em> ls lets you type <em>dir</em> for the <em>ls</em> command)</td>
</tr>
<tr>
<td>unalias alias</td>
<td>alias <em>oldname</em></td>
</tr>
<tr>
<td>cat <em>file</em></td>
<td>displays, concatenates, or merges files together, depending on the arguments and operators used with it</td>
</tr>
<tr>
<td>compress <em>file</em></td>
<td>makes file smaller so it takes up less disk space (a .Z is added to these files); uncompress restores file to original size</td>
</tr>
<tr>
<td>cp <em>file</em> <em>newplace</em></td>
<td>copies a file to a new location</td>
</tr>
<tr>
<td>date</td>
<td>prints current date and time</td>
</tr>
<tr>
<td>diff <em>file1</em> <em>file2</em></td>
<td>compares the contents of files or groups of files</td>
</tr>
<tr>
<td>du</td>
<td>summarizes disk usage</td>
</tr>
<tr>
<td>exit</td>
<td>closes a terminal (xterm) window</td>
</tr>
<tr>
<td>find -name <em>file</em></td>
<td>finds named file in a directory hierarchy</td>
</tr>
<tr>
<td>grep <em>string</em></td>
<td>searches a file for a pattern</td>
</tr>
<tr>
<td>head <em>file</em></td>
<td>head displays first ten lines of a file; tail displays last ten lines of a file</td>
</tr>
<tr>
<td>less <em>file</em></td>
<td>displays contents of a file</td>
</tr>
<tr>
<td>ls</td>
<td>lists the contents of a directory</td>
</tr>
<tr>
<td>logout</td>
<td>logs user out</td>
</tr>
<tr>
<td>man <em>command</em></td>
<td>shows Unix manual page, a text file of information, for the named command</td>
</tr>
<tr>
<td>more <em>file</em></td>
<td>displays contents of a file</td>
</tr>
<tr>
<td>mv <em>old</em> <em>new</em></td>
<td>renames a file, preserving the contents but under a new name; moves a file from one location to another (clear paths must be specified)</td>
</tr>
<tr>
<td>mv <em>file</em> <em>newlocation</em></td>
<td>displays the Eos/Unity user policy</td>
</tr>
<tr>
<td>policy</td>
<td>displays the Eos/Unity user policy</td>
</tr>
<tr>
<td>ps aux or ps -e</td>
<td>lists all processes running on the workstation by process ID number (PID); kill <em>pid</em> shuts down the process that the PID number identifies in the argument</td>
</tr>
<tr>
<td>ps aux or ps -e</td>
<td>lists all processes running on the workstation by process ID number (PID); kill <em>pid</em> shuts down the process that the PID number identifies in the argument</td>
</tr>
<tr>
<td>ps aux or ps -e</td>
<td>lists all processes running on the workstation by process ID number (PID); kill <em>pid</em> shuts down the process that the PID number identifies in the argument</td>
</tr>
<tr>
<td>ps aux or ps -e</td>
<td>lists all processes running on the workstation by process ID number (PID); kill <em>pid</em> shuts down the process that the PID number identifies in the argument</td>
</tr>
<tr>
<td>pwd</td>
<td>displays path of working or current directory</td>
</tr>
<tr>
<td>rm <em>file</em></td>
<td>removes a file; file cannot be restored if removed</td>
</tr>
<tr>
<td>sort <em>file</em></td>
<td>sorts file data</td>
</tr>
<tr>
<td>spell <em>file</em></td>
<td>lists all words in a file that the dictionary does not recognize</td>
</tr>
</tbody>
</table>
**Directory Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>add software</code></td>
<td>loads a software package</td>
</tr>
<tr>
<td><code>attach directory</code></td>
<td>accesses another user’s directory, a public locker, or a different file system</td>
</tr>
<tr>
<td><code>cd directory</code></td>
<td>changes directory, moving the user from one directory to another</td>
</tr>
<tr>
<td><code>mkdir directory</code></td>
<td>makes or creates a directory and gives it the name you provide</td>
</tr>
<tr>
<td><code>rmdir directory</code></td>
<td>removes directory (directory must first be empty of all files)</td>
</tr>
</tbody>
</table>

**Printing Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>lpq</code></td>
<td>shows the jobs in the print queue</td>
</tr>
<tr>
<td><code>lpquota</code></td>
<td>gives information about user’s print quota and print jobs</td>
</tr>
<tr>
<td><code>lpr file</code></td>
<td>prints a file</td>
</tr>
<tr>
<td><code>lprm job#</code></td>
<td>removes a print job from the printer queue</td>
</tr>
<tr>
<td><code>nxpr &amp;</code></td>
<td>brings up Print Manager application</td>
</tr>
</tbody>
</table>

**Zephyr Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>zaway</code></td>
<td>notifies others via Zephyr when user is unavailable</td>
</tr>
<tr>
<td><code>zctl options</code></td>
<td>Zephyr control program with many options (see <code>man zctl</code>), e.g., <code>zctl flush_locs</code> to kill old login processes</td>
</tr>
<tr>
<td></td>
<td><code>zctl hide</code> and <code>zctl unhide</code> to hide and unhide from <code>zlocate</code></td>
</tr>
<tr>
<td></td>
<td><code>zctl set exposure none</code> and <code>zctl set exposure realm-visible</code> to block zephyrs</td>
</tr>
<tr>
<td><code>zleave hhmm</code></td>
<td>sends Zephyr at the time specified, e.g., <code>zleave 1530</code> sends a Zephyr at 3:30 p.m.; the <code>cancel</code> option cancels the currently running <code>zleave</code></td>
</tr>
<tr>
<td><code>zlocate username</code></td>
<td>finds a user running Zephyr</td>
</tr>
<tr>
<td><code>znol</code></td>
<td>locates selected users (listed in <code>~/anyone</code>) running Zephyr</td>
</tr>
<tr>
<td><code>zwho</code></td>
<td>locates all users running Zephyr</td>
</tr>
<tr>
<td><code>zwrite username</code></td>
<td>writes to another user via Zephyr</td>
</tr>
</tbody>
</table>

**User Location Commands**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>`finger username</td>
<td><code>or</code>f username`</td>
</tr>
<tr>
<td><code>hes username</code></td>
<td>queries name server for account information on specific user</td>
</tr>
<tr>
<td><code>midas string</code></td>
<td>searches Hesiod database and returns matching usernames and login status (e.g., <code>midas john</code> matches user first or last names, <code>john</code>, <code>johns</code>, <code>johnson</code>, etc.)</td>
</tr>
<tr>
<td><code>whois string</code></td>
<td>searches local Internet phonebook and returns matching names, addresses, etc., of user(s) inquired about</td>
</tr>
<tr>
<td><code>zlocate username</code></td>
<td>finds a user who is using Zephyr</td>
</tr>
<tr>
<td><code>zwho</code></td>
<td>finds all users who are using Zephyr</td>
</tr>
</tbody>
</table>
## AFS Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Function</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>fs la directory</code></td>
<td>displays access control list for a directory</td>
</tr>
<tr>
<td><code>fs sa directory username rights,</code> or <code>fs sa directory owner:groupname rights</code></td>
<td>sets access control list for a directory, specifying who has rights (rldwka)</td>
</tr>
<tr>
<td><code>fs help</code> or <code>fs command -help</code></td>
<td>help on other <code>fs</code> commands</td>
</tr>
<tr>
<td><code>fs whereis</code></td>
<td>list file’s location</td>
</tr>
<tr>
<td><code>quota</code> (an alias for <code>fs listquota</code> or <code>fs lq</code>)</td>
<td>shows user’s disk quota status</td>
</tr>
<tr>
<td><code>pts adduser username -group owner:groupname</code></td>
<td>adds a member to a group of people</td>
</tr>
<tr>
<td><code>pts creategroup owner:groupname</code></td>
<td>creates a group of people</td>
</tr>
<tr>
<td><code>pts delete owner:groupname</code></td>
<td>removes a group of people</td>
</tr>
<tr>
<td><code>pts membership username</code></td>
<td>lists all groups user is a member of</td>
</tr>
<tr>
<td><code>pts removeuser username -group owner:groupname</code></td>
<td>removes a member from a group of people</td>
</tr>
<tr>
<td><code>pts help</code>, or <code>pts command -help</code></td>
<td>help on other <code>pts</code> commands</td>
</tr>
</tbody>
</table>

## Vi Commands, distributed by UNIX WORLD magazine

<table>
<thead>
<tr>
<th>Mode</th>
<th>Change</th>
<th>Delete</th>
<th>Move</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CHANGE</strong></td>
<td>cw</td>
<td>dw</td>
<td>k</td>
<td>u</td>
</tr>
<tr>
<td></td>
<td>cc</td>
<td>dd</td>
<td>j</td>
<td>.</td>
</tr>
<tr>
<td>c</td>
<td>C</td>
<td>D</td>
<td>l</td>
<td>/</td>
</tr>
<tr>
<td>s</td>
<td>s</td>
<td>x</td>
<td>h</td>
<td>?</td>
</tr>
<tr>
<td>S</td>
<td>S</td>
<td>X</td>
<td>0</td>
<td>n</td>
</tr>
<tr>
<td>r</td>
<td>r</td>
<td>xp</td>
<td>$</td>
<td></td>
</tr>
<tr>
<td><strong>vi states</strong></td>
<td>command</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>initial state; cancel partial command with Esc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>enter with a i o c s A I O C S; end with Esc</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>last line enter with : / ? end with RET</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>INSERT</strong></td>
<td>a</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>A</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>i</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>I</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>o</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>O</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>yw</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>EX COMMANDS</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>se nu</td>
<td>set numbers</td>
<td>:w</td>
<td>write buffer</td>
</tr>
<tr>
<td></td>
<td>se nonu</td>
<td>no numbers</td>
<td>:q</td>
<td>quit</td>
</tr>
<tr>
<td></td>
<td>r</td>
<td>read in file</td>
<td>:wq</td>
<td>write and quit</td>
</tr>
<tr>
<td></td>
<td>!</td>
<td>run cmd</td>
<td>:q!</td>
<td>abandon buffer</td>
</tr>
<tr>
<td></td>
<td>se wm=10</td>
<td>wrap words</td>
<td>ZZ</td>
<td>same as :wq</td>
</tr>
<tr>
<td></td>
<td></td>
<td>^Z</td>
<td>suspend vi</td>
<td></td>
</tr>
<tr>
<td><strong>SAVE/QUIT</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

194
## Vi Commands, Compiled by Tim Lowman

### Write, Edit, Quit, and Undo Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>:q</td>
<td>quit vi, unless the buffer has been changed</td>
</tr>
<tr>
<td>:q!</td>
<td>quit vi, without writing the file</td>
</tr>
<tr>
<td>:wq</td>
<td>write the buffer and quit</td>
</tr>
<tr>
<td>:w</td>
<td>write the file</td>
</tr>
<tr>
<td>:w &lt;name&gt;</td>
<td>write file to name</td>
</tr>
<tr>
<td>:x,y w name</td>
<td>write line x to y to file name</td>
</tr>
<tr>
<td>:w!</td>
<td>override protection and write</td>
</tr>
<tr>
<td>:e &lt;name&gt;</td>
<td>edit another file name without leaving vi</td>
</tr>
<tr>
<td>:e!</td>
<td>re-edit a messed-up file</td>
</tr>
<tr>
<td>^L</td>
<td>refresh the screen</td>
</tr>
<tr>
<td>^Z</td>
<td>suspend vi</td>
</tr>
<tr>
<td>^G</td>
<td>tell current file name and line number</td>
</tr>
<tr>
<td>u</td>
<td>undo latest change</td>
</tr>
<tr>
<td>U</td>
<td>undo all changes on a line, not having moved off it</td>
</tr>
</tbody>
</table>

### Append and Delete Commands

<table>
<thead>
<tr>
<th>Meaning (all appends and inserts are ended with Esc key)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Append and Delete Commands</td>
</tr>
<tr>
<td>* a</td>
</tr>
<tr>
<td>* i</td>
</tr>
<tr>
<td>* o</td>
</tr>
<tr>
<td>* O</td>
</tr>
<tr>
<td>* [&quot;&lt;a–z1–9&gt;]&lt;p</td>
</tr>
<tr>
<td>* [&quot;&lt;a–z1–9&gt;]&lt;p</td>
</tr>
<tr>
<td>* .</td>
</tr>
<tr>
<td>* x</td>
</tr>
<tr>
<td>* X</td>
</tr>
<tr>
<td>* d&lt;h,j,k,l&gt;</td>
</tr>
<tr>
<td>* r&lt;char&gt;</td>
</tr>
<tr>
<td>* R</td>
</tr>
<tr>
<td>* s</td>
</tr>
<tr>
<td>– ~</td>
</tr>
<tr>
<td>* J</td>
</tr>
<tr>
<td>– [x,y]&lt;s/pattern&gt;/&lt;replace&gt;/&lt;flag&gt;</td>
</tr>
<tr>
<td>* yy</td>
</tr>
</tbody>
</table>
More information about the software listed in this appendix is on the NCSU Web. Check http://www.eos.ncsu.edu/software/ for updates on new versions, executing commands, links to on- and off-campus support, documentation, and additional packages not listed here (see also http://www.ncsu.edu/it/essentials/software/overview_software.html).

The following abbreviations are used to identify the supported hardware platforms:

- **Sun** Sun Unix workstation running Solaris 8
- **Windows** Intel workstation (usually Dell) running Microsoft Windows 2000
- **Linux** Red Hat Linux 9

On Sun Solaris (Unix) machines, type `add` on the command line to list the software that runs on the workstation you are using.

On Windows, application icons are displayed in the right panel of the Novell Application Launcher (NAL). Special folders containing applications you can run appear in the left panel of the NAL. The Unity Applications folder has applications that are available to all users. Other folders contain applications for a college or department only, e.g., Engineering Applications. If you are in a department or college lab, you will be able to access specialty applications that are not generally available in Unity labs. When a folder is opened, the applications available to you to run will appear as icons in the right panel. If you have trouble running an application, right-click its icon and select Verify; then try again. Applications are also available on the Start -> Programs menu.

On Linux, see http://www.linux.ncsu.edu/ and http://www.linux.ncsu.edu/realmkit/ for information on application software. Software also appears in the Realm Kit Application Launcher (RKAL) and on the GNOME Start menu. Realm Kit Linux is principally supported by ITECS in the College of Engineering and the campus Linux Users Group.

Consult the web addresses above for the most current information.
ADAMS Mechanical System Simulation
Mechanical Dynamics, Inc.

ADAMS is mechanical system simulation software with modeling, analysis, and visualization capabilities. It enables users to produce virtual prototypes, realistically simulating the full-motion behavior of complex mechanical systems on their computers and quickly analyzing multiple design variations until an optimal design is achieved. This reduces the number of costly physical prototypes, improves design quality, and dramatically reduces product development time.

http://www.eos.ncsu.edu/software/adams/

Adobe Acrobat, Reader, and Photoshop
Adobe Systems, Inc.

Adobe Acrobat lets you convert any document, including Microsoft Office documents, to Adobe Portable Document Format (PDF). Anyone can open your document across a broad range of hardware and software, and it will look exactly as you intended, with layout, fonts, links, and images intact.

The Acrobat Reader is a tool for viewing, navigating, and printing PDF documents and can be downloaded for free from the Adobe web site. It launches automatically in your browser, but on Solaris machines can also be launched as a separate application (add acroread).

Adobe Photoshop is the de facto standard for digital image enhancement, photo retouching, and image compositing. It allows you to create original artwork, generate realistic or interpretive textures and backgrounds, correct color, retouch and composite scanned images, and prepare professional-quality separations and output for print or the Web.

http://www.eos.ncsu.edu/software/acrobat/
http://www.eos.ncsu.edu/software/photoshop/

AMPL/CPLEX Optimization Modeling
ILOG, Inc.

AMPL is a comprehensive and powerful algebraic modeling language for linear and nonlinear optimization problems, in discrete or continuous variables. It is also a computer language for describing production, distribution, blending, scheduling and many other kinds of problems known generally as large-scale optimization or mathematical
programming. AMPL’s familiar algebraic notation and interactive command environment are designed to help formulate models, communicate with a wide variety of solvers, and examine solutions.

**CPLEX** delivers high-performance, robust, flexible optimizers for solving linear, mixed-integer and quadratic programming problems in mission-critical resource allocation applications. Most leading end users and software providers in supply chain planning, telecommunication network design, and transportation logistics use ILOG CPLEX.

http://www.eos.ncsu.edu/software/ampl/
http://www.eos.ncsu.edu/software/cplex/

---

**ANSYS Finite Element Analysis**
ANSYS, Inc.

ANSYS is a tool for analyzing finite element engineering problems. Sample applications found in test programs include thermodynamic analysis (heat transfer, cooling analysis, temperature gradients, thermal responses) and structural analysis (torsion computation, vibration, stress analysis on various substances, elasticity). The package includes solid modeling/meshing, 3-D graphics, quality-analysis tools, and more.

http://www.eos.ncsu.edu/software/ansys/

---

**ARC/INFO Geographic Information System Toolkit**
Environmental Systems Research Institute (ESRI)

ARC/INFO is a GIS toolkit with thousands of built-in functions for sharing and processing geographic data, plus optional, fully integrated extensions for performing specialized tasks. ARC/INFO integrates all these data types with tabular DBMS data: vector, raster, photographs, scanned documents, satellite images, CAD drawings, and sound/video.

http://www.eos.ncsu.edu/software/arcinfo/

---

**ArcView Geographic Information System**
Environmental Systems Research Institute (ESRI)

ArcView is a powerful, easy-to-use tool that enables the user to spatially visualize, explore, query and analyze geographic data. ArcView has a useful set of ready-to-use data, and additional geographic data sets are available. ArcView supports various platforms and can be run on both Sun and Windows operating systems.

Platforms: Sun (8.1), Windows (8.3)

To run:
add arcinfo
arc

To run on Sun:
add arcview
arcview32

To run on Windows:
NAL Unity Applications
available from ESRI and from various third parties. The user can use ArcView to access ARC/INFO resources, including vector coverages, map libraries, grids, images and event data.

http://www.eos.ncsu.edu/software/arcview/

### ARENA Simulation System
Rockwell Software, Inc.

The Arena graphics simulation system is a complete and flexible modeling environment with an easy-to-use graphical user interface. It is designed for building computer models that accurately represent an existing or proposed application. Arena integrates all simulation-related functions-animation, input data analysis, model verification, and output analysis-into a single simulation modeling environment, with flexible flow-charting objects to capture systems of all kinds.

http://www.eos.ncsu.edu/software/arena/

### Aspen Plus and Icarus
Aspen Technology, Inc.

Aspen Plus, part of the Aspen Engineering Suite, simulates the behavior of various models involving chemical interaction and vapor-liquid processes. Sample applications found in test programs include calculation of viscosity and thermal conductivity of various substances, determination of parameters for chemical systems, hydrogenation and distillation systems, solids handling, costing and economic evaluation, electrolyte systems, and advanced physical properties.

http://www.eos.ncsu.edu/software/aspen/
http://www.eos.ncsu.edu/software/icarus/

### AutoCAD Computer-Aided Design
Autodesk, Inc.

AutoCAD is an industry standard for computer-aided design, offering exceptional performance, high-quality presentation, smart drawing tools and time-saving reference, raster, and Internet file-sharing capabilities.

http://www.eos.ncsu.edu/software/autocad/
AVR Studio Microcontroller Development
Atmel Corporation

AVR Studio is an Integrated Development Tool for the AVR family of microcontrollers. It enables the user to control execution of programs on the built-in AVR Instruction Set Simulator, or on an AVR In-Circuit Emulator. AVR Studio supports source-level execution of Assembly programs assembled with the AVR Assembler and other assemblers and compilers that support either UBROF or COFF as their object file format.

http://www.eos.ncsu.edu/software/avr/

Cadence Circuit Design Toolkit
Cadence Design Systems, Inc.

Cadence is a collection of digital and analog electronic design and simulation tools. Cadence provides products and solutions for every aspect of electronic design, and is used by the leading semiconductor, computer system, communications equipment, and consumer electronic companies in the world. Cadence is a very powerful and large package and only runs on higher performance workstations.

http://www.eos.ncsu.edu/software/cadence/

CADRA Computer-Aided Design
SofTech, Inc.

The CADRA family of CAD/CAM includes CADRA Design Drafting, a fast and highly productive mechanical design documentation tool; CADRA NC, a comprehensive 2 through 5 axis NC programming application; and CADRA integration with SolidWorks, an integrated 3D solid modeler and drawing production system. CADRA also has an extensive collection of translators and software options for multi-platform and multi-application organizations.

http://www.eos.ncsu.edu/software/cadra/

Dreamweaver Visual HTML Editor
Macromedia, Inc.

Dreamweaver is a professional HTML editor for visually creating and managing Web sites and pages. Dreamweaver provides advanced design
and layout tools and supports Dynamic HTML features, such as animated layers and behaviors. Browser-targeting checks pages for potential problems on all popular platforms and browsers. Dreamweaver makes it possible for users to create Web pages without coding, but you can easily hand-code within the environment if you prefer.

http://www.eos.ncsu.edu/software/dreamweaver/

FactoryCAD Factory Layout Drafting
EDS, Inc.

FactoryCAD is an E-factory layout software tool that allows you to create detailed, intelligent factory models. FactoryCAD provides equipment symbols, block management, optimized layer management tools, and smart factory objects, everything you need to create an entire factory model. Instead of drawing lines, arcs, and circles, smart objects represent floor and overhead conveyors, mezzanines, cranes, operators, material handling containers, etc. These objects allow you to snap together a layout model without wasting time drawing the equipment.

http://www.eos.ncsu.edu/software/factorycad/

Fireworks Web Graphics
Macromedia, Inc.

Fireworks allows you to create, edit, and animate Web graphics using a complete set of bitmap and vector tools. You can also use export controls to optimize your images, give them advanced interactivity, and export them into Macromedia Dreamweaver, Flash, and other HTML editors.

http://www.eos.ncsu.edu/software/fireworks/

Flash Web Animation
Macromedia, Inc.

Flash is software for designing and delivering low-bandwidth animations, presentations, and Web sites. It offers scripting capabilities and server-side connectivity for creating engaging applications, Web interfaces, and courses. Flash also integrates with Macromedia Dreamweaver and Fireworks.

http://www.eos.ncsu.edu/software/flash/
FORTRAN 90 and 95 NAGware Compilers
Numerical Algorithms Group (NAG)

The NAGWare f95 compiler is a full ISO and ANSI standard implementation of the FORTRAN 95 language. The NAGWare f90 compiler, launched in June 1991, was the world's first FORTRAN 90 compiler. The new features of FORTRAN 95 have been added to this well-proven compiler and includes many performance and operations improvements. The NAGWare f95 Compiler compiles both free format FORTRAN 90/95 input and FORTRAN 77 fixed format input.

http://www.eos.ncsu.edu/software/f95/

FrameMaker Word Processor and Publisher
Adobe Systems, Inc.

FrameMaker is a multi-channel publishing package with full word-processing features, graphics and drawing capabilities, equation editor, bookmaking, and other features for creating professional output. FrameMaker includes tools for writing and distributing long, structured, content-rich documents across all major computing platforms, with export to XML. The software supports the entire publishing process, from word processing and layout to electronic distribution.

http://www.eos.ncsu.edu/software/framemaker/

HSPICE Electronic Circuit Simulator
Synopsys, Inc.

HSPICE is an industry-standard cell-design and process-modeling tool with worldwide sign-off standard for circuit simulation. It supports more than 20 customized proprietary models, including Level 28 and many MOSFET, JFET, diode and capacitance models and public-domain models such as BSIM3. It provides superior convergence and optimizes circuits for models and cells, with multi-parameter optimizations in AC, DC and transient simulations. It also provides interpreted Monte Carlo yield-analysis model, worst-case design analysis support, and highly accurate timing models for gate-level simulators.

http://www.eos.ncsu.edu/software/hspice/

Java Development Kit (JDK)
Sun Microsystems, Inc.

Java 2 Platform, Standard Edition (J2SE) provides a complete, secure foundation for building and deploying network-centric enterprise
applications ranging from the PC desktop computer up to the workgroup server. It provides a stable, secure and feature-complete development and deployment environment designed from the ground up for the Web. It provides cross-platform compatibility, safe network delivery, and smartcard to supercomputer scalability. It also provides software developers with a platform for rapid application development.

http://www.eos.ncsu.edu/software/jdk/

JMP Statistical Discovery Software
SAS Institute, Inc.

JMP (pronounced “jump”) is an interactive environment for statistical visualization and exploratory data analysis. JMP presents statistics in an easily understood, graphical environment. Data tables are presented in spreadsheet form and are dynamically linked to related graphs and tables. JMP offers capabilities for performing univariate statistics, analysis of variance and multiple regression, nonlinear fitting, multivariate analysis, and nonparametric tests.

http://www.eos.ncsu.edu/software/jmp/

LINDO/LINGO Optimization
LINDO Systems, Inc.

HyperLINDO is an interactive package for solving linear, quadratic and integer programming problems, evaluate the appropriateness of the results, make minor modifications to the data or parameters, and retest to obtain optimum output. As a general-purpose LP, QP, and IP optimizer, it recognizes general integer variables (not just 0/1), free variables, and bounded variables. LINGO is a language for developing large structured models. It is a general-purpose modeling language and optimizer with a built-in text editor. LINGO recognizes subscripted variables, sets, operations over sets, and general mathematical expressions. Commonly used trigonometric, mathematical, and statistical functions are built in.

http://www.eos.ncsu.edu/software/lindo/
http://www.eos.ncsu.edu/software/lingo/

LSF Load-Sharing Facility
Platform Computing, Inc.

The Load-Sharing Facility (LSF) for Application Resource Management provides background batch processing on Eos/Unity workstations. LSF
is an industry-standard set of integrated products that transforms a distributed network into a single shared computing resource for robust load sharing and batch scheduling.

http://www.eos.ncsu.edu/software/lsf/

**Maple Symbolic Math Computation**
**Waterloo Maple Software**

Maple is a comprehensive computer system for advanced mathematics. It includes facilities for interactive algebra, calculus, discrete mathematics, graphics, numerical computation and many other areas of mathematics. It also provides a unique environment for rapid development of mathematical programs using its vast library of built-in functions and operations.

http://www.eos.ncsu.edu/software/maple/

**Mathcad Calculation and Documentation**
**Mathsoft, Inc.**

Mathcad is an integrated environment for performing and communicating math-related work. Features include math operator display options for equals, partial derivatives, multiplication, assignments, and more; differential algebraic equation solving functions; over 17 arithmetic, 12 vector and matrix operators, and 4 summations and products operators, as well as customized user-defined operator support; and image manipulation functionality (zoom/pan/crop, brightness/contrast, rotate/flip/transpose, etc.)

http://www.eos.ncsu.edu/software/mathcad/

**MATLAB Numerical Matrix Computation**
**The MathWorks, Inc.**

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, prototyping, data analysis, visualization, scientific and engineering graphics, and application development, including Graphical User

To run:
- ssh to lsf.ncsu.edu
- add lsf
- lsfsubmit

Platforms: Sun, Windows, Linux (8.0)

To run on Sun and Linux:
- add maple
- xmaple

Also on the Application Menu

To run on Windows:
- NAL Unity Applications

Platforms: Windows (2001)

To run:
- NAL Engineering Applications

Platforms: Sun, Linux (6.1r12.1), Windows (6.5r13)

To run on Sun and Linux:
- add matlab
- matlab

Also on the Application Menu

To run on Windows:
- NAL Unity Applications

To run on Windows:
- NAL Unity Applications
Interface building. MATLAB features a family of application-specific solutions called toolboxes. Toolboxes are comprehensive collections of MATLAB functions (M-files) that extend the MATLAB environment to solve particular classes of problems.

http://www.eos.ncsu.edu/software/matlab/

**Moldflow Plastics Injection Molding**
Moldflow Corporation

Moldflow allows students in manufacturing classes to quantitatively evaluate the design of plastic parts. It simulates and predicts how molten plastic will flow into a cavity. It can then be used to predict manufacturing defects that can be corrected with proper part redesign.

http://www.eos.ncsu.edu/software/moldflow/

**NExS Scientific and Engineering Spreadsheet**
GreyTrout, Inc.

NExS (Network Extensible Spreadsheet) is an interactive spreadsheet designed specifically for the X Windows environment. It is capable of monitoring several X applications at once and updating the relevant mathematical, matrix, string and statistical data from each. It supports matrix, vector and fourier transformation operations, as well as the arithmetic, Boolean, and logic functions available in C.

http://www.eos.ncsu.edu/software/nexs/

**Norton AntiVirus Corporate Edition**
Symantec Corporation

Symantec’s Norton AntiVirus Solution provides anti-virus protection for individual systems up to large wide area networks. The scalable Symantec System Center management console offers real-time communication with clients and servers from a single point, allowing convenient distribution of new virus definition sets.

http://www.ncsu.edu/antivirus/

**Office**
Microsoft Corporation

The Microsoft Office suite offers a set of tools to create, publish, and analyze data. It includes Access database tool, Excel spreadsheet, PowerPoint presentation graphics, Publisher, and Word.
OPNET Network Management and Modeling
OPNET Technologies Incorporated

OPNET is a comprehensive development environment for modeling communication networks and distributed systems. OPNET analyzes the behavior and performance of modeled systems with tools for model design, discrete-event simulation, and data collection and analysis.

Platforms: Sun (9.2)

To run on Sun:
add opnet modeler

Primavera Project Planner
Primavera

Primavera is high-performance project management software. Its features include unlimited project groups, multiproject resource and task relationship control, critical path scheduling, 31 activity calendars per project, resource and cost management, over 150 predefined tabular and matrix reports and graphics, and a custom report writer.

Platforms: Windows (3.1)

To run:
NAL Engineering Applications

Pro/ENGINEER and Pro/MECHANICA
Parametric Technology Corporation

Pro/ENGINEER Computer-Aided Design software is a powerful tool for engineering design and an industry leader among CAD programs. Its parametric solid modeling methodologies capture design intent and support feature-based modeling of parts and the subsequent combination of parts in assembly creation.

Platforms: Sun, Windows (2001)

To run on Sun:
add proeng proe2001 (Pro/Engineer) cdrs2001 (Conceptual Design)
Also on the Application Menu

To run on Windows:
NAL Engineering Applications

SAS Data Analysis Applications
SAS Institute, Inc.

SAS is an integrated applications system for data access. Its programs may be used in operations research (models of distribution networks, resource

Platforms: Sun (8.2), Windows (9.2)

To run on Sun:
add sas sas
Also on the Application Menu
allocation problems, scheduling, production systems), report writing and graphics, business forecasting and decision support, project management, and applications development, among other things.

http://www.eos.ncsu.edu/software/sas/

---

**Silvaco Integrated Circuit Design**

Silvaco International

Silvaco’s Virtual Wafer Fab is an integrated environment of 1D, 2D and 3D process and device simulators, graphical user interface tools and automation tools. It provides a high-throughput environment for evaluating fabrication and performance of silicon and advanced materials semiconductor technologies.

http://www.eos.ncsu.edu/software/silvaco/

---

**SolidWorks Computer-Aided Design**

SolidWorks Corporation

SolidWorks for CAD and 3D solid modeling is mechanical design automation software for producing models and detailed drawings. Add-ons like CosmosWorks and CosmosMotion extend SolidWorks into structural analysis and motion.

http://www.eos.ncsu.edu/software/solidworks/
http://www.eos.ncsu.edu/software/cosmosworks/
http://www.eos.ncsu.edu/software/cosmosmotion/

---

**S-Plus Advanced Data Analysis**

Insightful, Inc.

S-Plus is an interactive computing environment that provides a full-featured graphical data analysis system and an object-oriented language. It is a programmable tool for data analysis and can be used for exploratory analysis, graphics, statistics, and mathematical computing.

http://www.eos.ncsu.edu/software/splus/

---

**StarOffice Office Application Suite**

Sun Microsystems, Inc.

StarOffice is a fully featured, integrated, and comprehensive office suite you can use to create dynamic documents, analyze data, design
presentations, collaborate with team members, publish Web content, and schedule appointments.

http://www.eos.ncsu.edu/software/staroffice/

---

**Synopsys Digital Circuit Synthesis**  
Synopsys, Inc.

Synopsys is a digital electronic circuit synthesis tool. It transforms a functional (register-transfer level) description of a module into a gate-level netlist. The Synopsys Design Analyzer is the graphic interface to the Synopsys family of synthesis tools.

http://www.eos.ncsu.edu/software/synopsys/

---

**Tecplot Interactive Plotting**  
Amtec Engineering, Inc.

Tecplot is an interactive package for visualizing technical data and creating XY plots, contour plots, vector plots, mesh plots, carpet plots, 3D stream ribbons, isosurfaces, light-source-shaded surfaces, etc. It is able to visualize complex data defined in one, two, or three dimensions or on its original non-rectangular grid.

http://www.eos.ncsu.edu/software/tecplot/

---

**TK Solver Plus Simultaneous Equation Solver**  
Universal Technical Systems, Inc.

TK Solver has a declarative, rule-based programming language that lets you work directly with a mathematical description of a problem without sequencing the operations to be performed by the computer, and without isolating the unknowns on one side of the equation. It recognizes eight classes of objects: rules (or equations), variables, lists of values (numeric, symbolic), functions (built-in, user-defined, or problem-specific), unit conversions, and plot, table and format specifications.

http://www.eos.ncsu.edu/software/tksolver/

---

**Together Control Center**  
TogetherSoft Corporation

Together ControlCenter is a graphical integrated environment for large-scale, collaborative software engineering projects and e-business projects.
solutions development. It supports collaboration with a common language, common diagrams, and common building block components.

http://www.eos.ncsu.edu/software/togethersoft/

---

**Visio Professional**

Microsoft Corporation

Visio Professional is used to create professional-looking diagrams. Drag-and-drop is used to assemble flowcharts, block diagrams, web site maps, floor plans, and basic network diagrams. Existing databases and Visual Studio projects can be reverse engineered to understand database schema and software code.

http://www.eos.ncsu.edu/software/visio/

---

**VisualAge Smalltalk**

IBM, Inc.

VisualAge Smalltalk enables software developers to use object-oriented technology to create highly portable, scaleable, multi-tier business applications. Smalltalk combines an object-oriented environment with visual programming, adding B2B support through a new XML parser, a new level of VisualAge for Java[tm] integration, enhanced WebSphere integration, and the addition of the new Advanced Database feature.

http://www.eos.ncsu.edu/software/smalltalk/

---

**Visual SlickEdit Programmer’s Editor**

SlickEdit, Inc.

Visual SlickEdit is an easy-to-use text editor but comprehensive and flexible code editor for C/C++, C#, Java, HTML, XML, Unicode and other programming languages, file types, and encodings. It has the same GUI on Windows, Linux and UNIX platforms and provides superior code-editing tools for intelligent coding assistance, file/directory tree differencing, and 3-way merge editing.

http://www.eos.ncsu.edu/software/slickedit/

See http://www.eos.ncsu.edu/software/ for additional information and more applications, including Automod, Femlab, Microstation with Geopak, RS Logix, SURFCAM, SmartFTP, Timberline Estimating, Visual Studio, WaterCAD, WordPerfect, X-Win32, etc.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>access control list (ACL)</td>
<td>a list that specifies what access privileges users have to a directory, specifically, the right to lookup, insert, delete, read, write, lock, or administer files in a directory (abbreviated as l, i, d, r, w, k, a, respectively).</td>
</tr>
<tr>
<td>alias</td>
<td>an alternate name or abbreviation (usually short and easy to remember) that substitutes for a pathname, command, list, or expression (usually long and hard to remember).</td>
</tr>
<tr>
<td>Andrew File System (AFS)</td>
<td>a software product of Transarc Corporation that distributes, stores, and joins files on networked computers. This distributed file system software makes it possible for users to access information located on any computer in a network.</td>
</tr>
<tr>
<td>argument</td>
<td>information the shell needs to perform a command, usually the file(s) or entity that will be affected by a command. In a command statement, the argument follows the command, e.g., <code>delete file</code>, where <code>delete</code> is the command and <code>file</code> is the argument.</td>
</tr>
<tr>
<td>ASCII</td>
<td>American Standard Code for Information Interchange, a standard computer code used to facilitate the exchange of information on various types of data-processing equipment. Files are produced in or converted to ASCII code to make them easier to move into or out of software applications running on different machines.</td>
</tr>
<tr>
<td>Athena Project</td>
<td>the joint project of MIT, DEC, and IBM in the 1980s that developed a distributed academic network for the MIT campus.</td>
</tr>
<tr>
<td>ATM</td>
<td>Asynchronous Transfer Mode, or “fast packet.” A method for the dynamic allocation of bandwidth using a fixed-size packet (called a cell).</td>
</tr>
<tr>
<td>authentication</td>
<td>the recognition of a user as having a valid account on the system with legitimate access to its resources, usually determined during login by username and password.</td>
</tr>
<tr>
<td>background process (bg)</td>
<td>a program that runs without interfering with command entry and processing taking place in other windows. The &amp; character added after a command will run that process in the background.</td>
</tr>
<tr>
<td>binary</td>
<td>a number system in which the base is two, each number being expressed in powers of two by using only two digits, 0 and 1. A binary file is a program file built for the computer architecture. A binary server is a server computer on which the files of the operating system reside.</td>
</tr>
<tr>
<td>byte</td>
<td>a unit of measure (8 bits) specifying file size, quota, and disk space.</td>
</tr>
<tr>
<td>C</td>
<td>a general-purpose programming language and a primary language of the Unix operating system.</td>
</tr>
<tr>
<td>cached file</td>
<td>a copy of a file that the cache manager stores on a workstation’s local disk.</td>
</tr>
<tr>
<td>cache manager</td>
<td>a program on a client machine that accesses files stored in AFS. When a user requests a file, the cache manager retrieves it from the appropriate file server and stores or “caches” a copy of it on the client workstation’s local disk for the user to use.</td>
</tr>
<tr>
<td>cache memory</td>
<td>is a mechanism interposed in the memory hierarchy between main memory and the CPU to improve effective memory transfer rates and raise processor speeds. The name refers to the fact that that the mechanism is essentially hidden and appears transparent to the user.</td>
</tr>
<tr>
<td>cell</td>
<td>an independently administered site running AFS and consisting of a collection of file server and client machines defined as belonging to the cell. A machine can belong to only one cell at a time.</td>
</tr>
<tr>
<td>client</td>
<td>a program or machine that performs for a user, requesting files and information from a server program or computer in order to complete its function.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>client/server</td>
<td>a model for distributed network computing that relies on server computers to supply software and services to client computers that request them.</td>
</tr>
<tr>
<td>cluster</td>
<td>a group of client workstations, usually close together, that connect off the same subnet.</td>
</tr>
<tr>
<td>command line interface</td>
<td>a terminal-type means for sending commands to the shell and entering data into the computer. The command line is indicated by a symbol, or prompt (e.g., &gt;, %, $), showing the user where to enter commands. Unlike a graphical user interface, this interface requires that commands be typed in and written in a precise syntax in order for them to be interpreted correctly by the shell.</td>
</tr>
<tr>
<td>console window</td>
<td>the terminal window in which error and processing messages are written.</td>
</tr>
<tr>
<td>C shell (csh)</td>
<td>a Unix shell developed by the University of California at Berkeley.</td>
</tr>
<tr>
<td>cursor</td>
<td>a small symbol, often flashing, that indicates the user’s place on the screen. It is the point-of-insertion indicator showing where data will be entered.</td>
</tr>
<tr>
<td>daemon</td>
<td>a Unix system task that runs as a background process (usually initiated at system boot time) to perform a particular system function. Common Unix daemons control spooled printer output, accept incoming telnet requests, activate time-of-day scheduled tasks, etc.</td>
</tr>
<tr>
<td>dataswitch</td>
<td>a piece of hardware with ports to accept data and then to switch and transfer that data out to other computers and services.</td>
</tr>
<tr>
<td>default</td>
<td>an automatic setting, mode, or action. Usually, there are alternative ways to perform any action in a computer program, but the default action is what will occur automatically if no other alternative is selected.</td>
</tr>
<tr>
<td>detached process</td>
<td>a process that continues to run in the background after the user has logged out. Generally, a detached process is started when a user does not expect the process to finish during a session.</td>
</tr>
<tr>
<td>dialog box</td>
<td>a small window-like box that opens after an operation has been selected. In it, you select options and settings to tailor the operation before it proceeds.</td>
</tr>
<tr>
<td>directory</td>
<td>a special kind of file that points to or contains others files and directories. Directories can be nested to any depth. Some software may refer to directories and subdirectories by other names, such as, folders, lockers, drawers, cabinets, etc.</td>
</tr>
<tr>
<td>distributed file system</td>
<td>a file system that joins together the file systems of individual machines. Files are stored (distributed) on different machines in a computer network but are accessible from all machines.</td>
</tr>
<tr>
<td>domain name system (DNS)</td>
<td>a general-purpose distributed, replicated, data query service chiefly used on Internet for translating hostnames into Internet addresses.</td>
</tr>
<tr>
<td>dotfile</td>
<td>see hidden file.</td>
</tr>
<tr>
<td>editor</td>
<td>a program that allows the user to enter, alter, format and store program and text files.</td>
</tr>
<tr>
<td>environment variables</td>
<td>global values or settings that determine the default operation of all shells and are also passed on to application programs. Environment variables contain information about your working environment. These are set by system administrators but can be changed by the individual user (main commands, printenv and setenv).</td>
</tr>
<tr>
<td>Eos</td>
<td>name of the development project and the distributed AFS-based computing network in the NCSU College of Engineering. The technology is based on the Athena Project technology.</td>
</tr>
<tr>
<td>Term</td>
<td>Definition</td>
</tr>
<tr>
<td>-----------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Ethernet</td>
<td>a network communications protocol developed and originally marketed by Xerox Corporation. This technology is designed to handle the communication procedures of separate devices, such as word processors, personal computers, file and print servers, etc. Ethernet uses a bus technology, that is, it connects all stations on the network through a single channel of coaxial cable called a bus.</td>
</tr>
<tr>
<td>executable</td>
<td>a statement or procedural step in a programming language that calls for processing action by the computer, e.g., performing arithmetic, reading data from an external medium, making a decision, etc. An executable file is a file with its mode is set to executable, making it a file that performs a process rather than simply holding data.</td>
</tr>
<tr>
<td>expression</td>
<td>a series of fundamental elements making up a command statement that a compiler can use to produce a value. Expressions have one or more operands and, usually, one or more operators.</td>
</tr>
<tr>
<td>FDDI</td>
<td>Fiber Distributed Data Interface, a standard for fiber-optics network technology that specifies a 100 Mbs (100 million bits per second) data rate.</td>
</tr>
<tr>
<td>file</td>
<td>a collection of information stored and retrieved under a single name.</td>
</tr>
<tr>
<td>file server</td>
<td>a computer used to store files and transfer requested files to client machines. Also, the AFS <code>fs</code> command stands for &quot;file server.&quot;</td>
</tr>
<tr>
<td>file system</td>
<td>a set of many files organized in a hierarchical tree of directories and subdirectories.</td>
</tr>
<tr>
<td>filter</td>
<td>a command that reformats or removes unwanted data from its input and writes the rest as output.</td>
</tr>
<tr>
<td>ftp</td>
<td>Internet file transfer protocol for transferring files from one computer to another.</td>
</tr>
<tr>
<td>foreground process (fg)</td>
<td>a process that engages the terminal so that it cannot be used for anything else until the foreground process has finished running. The shell must wait for the process to finish before prompting for another command.</td>
</tr>
<tr>
<td>gateway</td>
<td>a device linking two networks that use different protocols. It accepts all packets from each network addressed to the other, buffers them, converts them to the next format, and re-transmits them to the other network.</td>
</tr>
<tr>
<td>GMS</td>
<td>Global Message System.</td>
</tr>
<tr>
<td>gopher</td>
<td>an interface, locator, and browser for the Internet’s file and information resources.</td>
</tr>
<tr>
<td>graphical user interface (GUI)</td>
<td>picture-based software, such as OSF/Motif, that employs window frames, icons, and pointers to interface with files and directories rather than commands. This graphical “front end” or GUI is designed to be easier for the user than issuing commands to the shell via the command line.</td>
</tr>
<tr>
<td>Hesiod</td>
<td>a name coined by Project Athena for the “name-server” services of the network, which keeps track of resources and translates a request for a logical destination to a physical location. Named after the Greek poet, Hesiod.</td>
</tr>
<tr>
<td>hidden file</td>
<td>a file that does not appear in directory listing, unless the user invokes the <code>-a</code> option, i.e., <code>ls -a</code>. Names of hidden files begin with a period, e.g., <code>.mylogin</code>; also called a <code>dotfile</code>.</td>
</tr>
<tr>
<td>home directory (~)</td>
<td>a directory in a file system owned by a single user and used by that person to store files that s/he creates or copies there. The home directory is the directory the user enters upon login. It is represented by the tilde (~) in commands.</td>
</tr>
<tr>
<td>hostname</td>
<td>the unique name by which a computer is known on a network, used to identify it in electronic mail, Usenet news, or other forms of electronic information interchange.</td>
</tr>
</tbody>
</table>
**HTML**  Hypertext Markup Language, a hypertext document format. Built on top of SGML and embedded as “tags” in `.html` files used on the World WideWeb.

**http**  Hypertext Transfer Protocol, the client-server TCP/IP protocol used on the World WideWeb for the exchange of HTML documents. It conventionally uses port 80.

**hypertext**  information written, organized, and presented in an electronic “document” that has words or pictures linked to other documents. Hypertext is a document with embedded links that when selected connect the user to related text, graphics, or sound file.

**icon**  a symbol or small picture on the display screen representing a software application or operation. Typically, a user points to or selects the icon with a pointing device, such as a mouse, to manipulate the program or operation in specific ways.

**init process**  a process that begins execution when the system starts up and is responsible for creating login processes that wait for input from terminals. The init process is owned by the superuser and is controlled by the console.

**Internet**  a worldwide complex of computer networks, communicating at high speeds using the TCP/IP protocol, which universities, companies, and governments use to exchange information, electronic mail, etc.

**job**  a process or group of processes executed with a command, e.g., a print job, which is the process of sending and printing a file.

**Kerberos**  a name coined by Project Athena for the authentication and security services of the network. Kerberos provides workstations and services with encrypted “tickets” to be used when requesting a service on the network. Named after Kerberos in Greek mythology, the three-headed dog that guards the gates of Hades.

**kernel**  the central program and core of the operating system responsible for all machine-level work, including connecting to hardware devices. The kernel cannot be modified by the routine user.

**LAN**  Local Area Network, a data communications network which is geographically limited (typically to a 1 km radius) allowing easy interconnection of terminals, microprocessors, and computers within adjacent buildings. Ethernet and FDDI are examples of standard LANs.

**link**  a directory entry that refers to a file. There are two kinds of links—hard and soft—both using the `ln` command. A hard link is indistinguishable from an original directory entry; it may not span file systems or refer to directories. A soft or symbolic link contains the name of the file to which it is linked; it may span file systems and refer to directories.

**literal string**  a string of characters enclosed in quotation marks, which indicate that the string is to be interpreted literally.

**locker**  a directory, often used to mean the collection of a main directory and all the subdirectories and files under it.

**login or logging in**  the user-initiated process of gaining access to the system. Users identify themselves with their usernames and passwords, and the system verifies their accounts and permits them to access files and services.

**logout or logging out**  the user-initiated process of signing off the system, including shutting down processes, putting away files, and cutting the user’s connection to the system.

**login id**  see `username`.
mail handler  the user interface software for the mail system, which communicates with the Post Office server to access spooled user mail and deliver it to users on their display screens. This program also permits users to write mail message to other users.

man(ual) pages  online reference documentation for Unix, organized by command into individual pieces or pages of explanation. For example, the command, man ls, brings up the man page on the ls command, which is further subdivided into parts covering the command’s name, syntax, description, options, restrictions, etc.

MB1, MB2, MB3  abbreviation for the first, second, and third mouse buttons respectively.

menu  list of options and functions that users can “pull down” or “pop up” in places on the screen; user generally uses a mouse to select, and initialize operations from a menu.

metacharacter  special characters that are neither letters nor numbers but have special meaning either to the shell or the operating system, e.g., > and <, which perform redirection, and |, which “pipes” commands. For a metacharacter to be interpreted literally and not for its special meaning, it must be placed in quotation marks, e.g., ‘<’.

mode bits  a set of access rights associated with a file or directory in the Unix file system, which are shown with the ls -l command. The rights are read, write and execute (r,w,x). AFS combines their effect with AFS access rights in order to determine what type of access someone has to the files.

MOTD  Message of the Day, generally delivered via a Zephyr window.

Motif  graphical user-interface software that defines the appearance and operation of the on-screen windows.

mount point  a special type of directory that connects a location in the AFS file space with a volume. A mount point looks like a standard Unix directory. Listing the directory (ls) shows the contents of the volume.

mouse  a hand-held locator and pointing device connected to the workstation by a cord.

multitasking  able to support the processing of numerous programs and computations at the same time. Programs process concurrently and, thus, more quickly, permitting the easy sharing and movement of data, graphics, and text among windowed applications on the screen.

NFS  a standard protocol developed by Sun Microsystems, which allows a computer to access files over a network as if they were on its local disks (similar to AFS).

node  a connecting point on the backbone network. From nodes, bridges fan out to local ethernets.

operating system (os)  software (programs and data) that initiates the interaction of the electronic and electromechanical components of a computer so that they constitute a useful system for carrying out calculations; a set of instructions that tells a computer how to work. The operating system is the means for processing programs and sharing equipment and computer services among users.

operators  symbols that represent processes to be carried out.

option  an argument that controls how the shell executes a command, e.g., in the command ls -l, the -l is an option that tells the shell to do a special kind of directory listing, that is, a long listing of files.

partition  an area of a computer disk used for storage and further subdivided into volumes.
password  a unique, user-defined string of characters validating the user’s system identity. The
user must correctly enter the password (which does not appear on-screen when it is
typed) in order to be authenticated by the system.

pathname  the location of a file or directory in the system’s hierarchical structure of directories.
The pathname tells the shell where in the directory tree to find a file. Files may be
referred to by absolute pathname (also called full or complete pathname) or relative
pathname. An absolute pathname is the full specification of a path beginning with the
root directory /afs and including all directory levels the system passes through to locate
the file. A relative pathname is the location of the file or directory relative to the
directory in which the user is currently located (the current working directory).

pid  abbreviation for process abbreviation number (see process).

pipe ( | ) used to represent a pipe between two processes in a shell command line.

pipeline  a sequence of one or more shell commands separated by a pipe symbol ( | ). The
standard output of each command is sent as standard input to the next command. Each
command is run as a separate process, but the shell waits for the whole series to finish
before issuing a new prompt.

PostScript  a page description language that codes files for printing on PostScript printers.

process  a job or program running on a computer. Unix assigns each process a unique reference
number, called a process identification number (PID). Users refer to the PID when they
want to affect a process in some way, such as to terminate it. A user identification
number (UID) is also attached to every operating system process. When a user enters
a Unix command, it is executed as a subordinate process, or child process of a parent
process. Every process on the system has a parent except the init process (see init
process). All processes that a user creates will be children of the login shell.

protocol  a set of rules governing the communication and transfer of data between computers.

program  a logical sequence of coded instructions specifying the operations to be performed by
a computer in solving a problem or in processing data; or, a series of operations which
may be used to control the function of an electronic device.

prompt  a symbol, word, or message that the system displays to tell the user that it is ready for
new input or commands. The default shell prompt of the Eos system is eos%.

quota  a limit set by the system administrator on such things as disk storage (measured in
kilobytes) and printing.

redirection  the process of writing output from a command to a file using the right-angle bracket
(>), or of reading input for a command from a file using the left-angle bracket (<).

remote access  connection to the network from outside the established realm of client and server
machines, usually via telephone and modem connection.

RISC  Reduced Instruction Set Computer.

root directory ( / ) the top-level directory in the system’s directory hierarchy, represented by the “forward
slash” symbol ( / ).

router  a dedicated computer that links, translates, and moves data in units called “packets”
over networks.

scrolling  the effect of lines moving on and off the screen or in and out of a window, either
vertically or horizontally, as the user types or manipulates the cursor to move through
a file that has been opened.
server  a resource-sharing computer that shares its files and provides particular (usually specialized) support services to other computers on a network.

shell  a program that control user interactions with the kernel of the system by interpreting and executing commands. Sometimes called a command interpreter.

shell script  a file of shell commands, also known as a shell program or shell procedure. Files having the # character as the first character are interpreted as C shell scripts.

shell variable  a named storage location that contains a value. A value is assigned to a variable by using the set command. Shell variables work like environment variables, except that a set of shell variables is used by a single C shell only, and shell variables do not propagate to new shells or other programs.

standard input  a file that represents where commands will get their input, which is usually from the keyboard. Typed commands are treated as standard input unless the user redirects the standard input to come from elsewhere.

standard output  a file that represents where commands will send their output, which is usually to the terminal screen. When the system writes to the standard output file, the output appears on the screen unless the user redirects the standard output, such as to a file.

subdirectory  a directory that resides in another directory.

superuser  the administrator of the operating system. The superuser has special privileges, including the right to boot (start up) the system and to stop it, and to set up accounts and special user groups.

TCP/IP  Transmission Control Protocol/Internet Protocol, the formal rules (protocol) that the Internet uses to support such services as file transfer and mail.

terminal  the combined keyboard and monitor through which the user communicates with the computer system.

token  a set of data that indicates that a user has been authenticated and is authorized to request files and services on the system.

Unity  the campus-wide implementation of the NCSU College of Engineering’s computing environment, called Eos, a Unix-based network for distributed computing based on technology developed in MIT’s Athena Project.

Unix  a popular and mature operating system (of which there are many versions) that runs on many kinds of computers.

URL  Uniform Resource Locator, a simple extension of the standard filename concept. It can be used to point to files and directories on any machine connected to the network. It can be accessed through several different methods, e.g., ftp, gopher and HTTP.

username  a short string of characters (usually 8) entered at login that uniquely identifies a user. Also called the login id, the username is the first thing typed into the computer and, together with the password, is used to authenticate user accounts on the system.

variable  a symbol whose value can be set.

vi  a full-screen editor available to Unix users.

volume  a container that keeps a set of related files and directories together on a disk partition (specific to AFS), e.g., a user’s home directory.

wildcard  a metacharacter that can be used in place of other characters or words in filename arguments. The asterisk (*) and question mark (?) are wildcards.
<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>word wrap</td>
<td>allows you to type continuously without pressing RETURN to start a new line because the program “wraps” the text to the new line for you.</td>
</tr>
<tr>
<td>working directory</td>
<td>the directory the user is currently working in. Typically, when users log in, they are placed in their home directories, which would be their working directory. The command <strong>pwd</strong> (path of working directory) tells what the user’s working or current directory is.</td>
</tr>
<tr>
<td>World Wide Web</td>
<td>international effort to organize the Internet’s data and information resources into hypertext formats that are linked together and searchable.</td>
</tr>
<tr>
<td>X11</td>
<td>a network protocol and subroutine library used to create graphic images and windows.</td>
</tr>
<tr>
<td>Zephyr</td>
<td>a name coined by Project Athena for the rapid user-notification and message service on the system. Messages are sent from a terminal window with the zwrite command and received and displayed in a window on the receiver’s screen. Named after Zephyr, god of the West Wind and son of Eos and Astraeus in Greek mythology.</td>
</tr>
</tbody>
</table>
INDEX
Symbols

```
!!, 87
?, 88
&, 62, 168, 173
@, 138
*, 88, 160, 170
], 89
>, 88
~, 70, 76, 79, 94, 176

A
Access, 56, 59, 206
access control list (ACL), 91-99, 119, 184
accessibility, 15, 33, 57
Accessories, 29, 56
Acrobat, 56, 59, 60, 61, 198
ADAMS, 59, 60, 198
add, 31, 50, 54, 62, 71, 77, 78, 184, 184, 192, 193, 197
Adobe, 56
alias, 87, 93, 95, 121, 132, 168, 169, 192
AMPL/CPLEX, 60, 198
Andrew File System (AFS), 2, 47, 48, 70, 72, 91-96, 115, 119, 183
ANSYS, 59, 60, 199
Application Menu, 31, 50, 62, 63, 143, 152
ARC/INFO, 60, 199
ArcView, 56, 59, 60, 199
Arena, 59, 200
argument, 81, 90
ASCII, 106, 127, 131, 143, 160
Aspen, 59, 200
asWedit, 52, 143, 148
Athena Project, 1, 2, 8
attach, 71, 77, 78, 95, 168, 172, 184, 193
authentication, 47, 91, 99, 120, 138
AutoCAD, 56, 59, 129, 200
AVR Studio, 201
```

B
background process (bg), 62, 63
backup, 29, 162
binary (bin), 127

C
Cadence, 60, 201
CADRA, 53, 59, 60, 201
Calculator, 54
cat, 82, 82, 89, 192
cd, 73, 76, 78, 94, 193
cell, 72, 92, 100, 161
b.p.ncsu.edu, 2, 91
eos.ncsu.edu, 1, 48, 72, 73, 91
unity.ncsu.edu, 1, 48, 72, 73, 91
cell phones, 13
Centennial Campus, 67
child directory, 70
chmod, 90, 92, 93
client, 17, 64, 91
client-server, 64, 91
Clock, 54
color, 64
College of Engineering (COE), 1, 7, 8, 15, 40, 132, 184, 229
command, option, 103
command interpreter, 25, 46, 164
command prompt, 79
compress, 173, 182, 192
Computing Services, 7
concatenate, 82
configuration files
  .cshrc, 167
  .login, 167, 176, 177, 180
  .logout, 167-169, 178, 180
  .fwmm2rc, 167-169, 177, 178
  .cshrc, 176-178
  .mycshrc, 87, 167-169, 172, 177, 178, 180
  .mylogin, 167-172, 177, 178, 180
  .signature, 110
  .Xdefaults, 167-171, 177, 178, 180
  .xsession, 167-169, 177, 178
  .Xstartup, 167-169, 172, 178, 180
console window, 25
Control Panel, 29, 33
cp, 25, 83, 192
current directory, 73, 182
customization, 3, 29, 46, 71, 166

D
Data Explorer, 60
default, 166
dialup, 130-135, 168, 176
disk
  backup, 158, 182
  drive, 18, 29, 72, 73, 97, 98
  quota, 7, 70, 92, 97, 182, 183
documentation, 57
dotfiles, 22, 71, 166
Dreamweaver, 56, 59, 112, 117, 126, 201
du, 182, 192

E
editors, 50, 52, 143-149
electronic mail, 107-110
elm, 61, 89, 94, 108, 131
emacs, 61, 149
environment variables, 164, 165, 168, 169,
  176, 178, 180
Ethernet, 64, 128
Excel, 52, 56, 59, 84, 206
exit, 192

F
FactoryCAD, 59, 202
FDDI, 64
fdformat, 158, 159
fee, Education and Technology, 2, 6, 183
fg, 63
File Transfer Protocol (ftp), 121, 126-127
find, 192
Fireworks, 59, 202
Flash, 59, 202
foreground process (fg), 62
Fortran, see NAGware
FrameMaker, 51, 59, 60, 131, 203
fs, 91, 92, 93, 95, 98, 119, 161, 183, 194
fs examine, 161
fs la, 94, 182, 183, 194
fs listquota (fs lq), 92, 98, 182, 194
fs rmrmount, 162
fs sa, 94, 95, 119, 183, 194
fs whereis, 47, 92, 194
fs whichcell, 162
ftp, 121, 126, 127, 131, 131, 132-134

G
games, 131
get, 126, 127
ghostview, 61, 103
gimp, 54, 61
GINA, 2
GNOME, 27, 32, 34, 43, 46, 63, 85, 197
gnu, 61
GNU C/C++, 59
Grammatik, 157
graphical user interface (GUI), 25, 46, 171
grep, 89

H
head, 82, 192
help, 40, 51, 192
hes, 185, 193
Hesiod, 47, 48, 185
history, 87, 192
home directory, 70, 72, 76, 97, 119, 126,
  141, 182
HSPICE, 60, 203
HTML, 52, 84, 116, 117, 121, 122, 143,
  148, 152, 155, 201
hypertext, 51, 52, 116, 148

I
IMAP, 50, 108, 131, 132
Information Technology and Engineering Computer Services, 229
init_dotfiles, 175
Internet, 7, 51, 64, 107, 109, 116, 126, 140, 152
Internet Service Provider (ISP), 128, 130

J
Java Development Kit (JDK), 28, 56, 59, 60, 203
JAWS, 57
JMP, 56, 59, 204

K
KAUTH, 3
Kerberos, 2, 3, 39, 47, 48, 138
kernel, 46, 164
keyboard, 19
kill, 63, 192
kreset, 192

L
labs, 14-16, 67
LaTeX, 61
less, 82, 192
LINDO, 59, 204
LINGO, 204
Linux, 3, 14, 17, 23, 24, 27, 32, 34, 38, 43, 46, 50, 60, 61, 63, 76, 85, 102, 108, 197
Linux Users Group, 4
locker, 71, 78, 172, 183, 184
logging in, 132
login, 166, 170, 176
logout, 29, 42, 172, 192
lpq, 104, 106, 183, 193
lpquota, 156, 182, 182, 193
lpr, 156, 193
lprm, 104, 183, 193
ls, 73, 77, 81, 88, 167, 169, 192, 216
LSF Load-Sharing Facility, 60, 204
lynx, 61, 131

M
mail, see electronic mail
man, 78, 81, 83, 90, 132, 136, 148, 160, 175, 192
man page, 81, 90, 160
Maple, 28, 52, 56, 59, 60, 131, 205
Mathcad, 59, 205
Matlab, 28, 52, 56, 59, 60, 129, 131, 205
mcopy, 155, 159, 160
mdir, 159, 159
Message Board, 113
Meta key, 19
metacharacter, 88
mformat, 158, 159
mget, 126
Microsoft Office, 206
midas, 193
MIT, 1, 8, 34
mkdir, 76, 87, 119, 193
mode bits, 92
Moldflow, 206
more, 82, 86, 89, 90, 192
mount point, 92, 100
move, 83
mput, 126, 127
MS Windows, 3, 26
MS-DOS, 46, 79, 98, 158, 169
mtools, 155, 158, 159, 160
mutt, 61, 108
mv, 83, 192
My Computer, 33, 72

N
NAGware Fortran, 28, 59, 60, 203
nedit, 52, 61, 131, 143
Netscape, 51, 56, 60, 61, 84, 108, 110, 116, 127
network, 65, 66, 67
NExS, 52, 60, 131, 206
Norton AntiVirus, 57, 59, 206
Novell Application Launcher (NAL), 3, 50, 55, 63, 197
Netware Directory Services (NDS), 3, 23, 38, 48
nxpr, 51, 61, 193

O

OPNET, 60, 207
options, 81, 90

P

parent directory, 70, 76
partition, 92, 97, 100
password, 6, 20, 38, 47, 48, 91, 92, 100, 183
path, 72, 76, 83, 87, 165, 168, 172, 173
pathname, 86
  absolute, 72, 73, 79, 184
  relative, 73, 79
perl, 61
permissions, 182, 183, 184
Photoshop, 57, 59, 60, 198
pico, 61, 131, 143, 148
pine, 61, 108, 131, 132
pipe, 89
pipeline, 89
policy, 51, 192
  account sharing, 7, 8, 39
  cell phone use, 13
  commercial use, 10
  copying software, 13
dialup, 131
games, 12
general, 8, 51, 183
labs, 12, 14
network use, 12
personal use, 9
ResNet, 13
violations, 10, 12
Web pages, 121
Post Office Protocol (POP), 108, 131
PostScript, 156
PowerPoint, 56, 59, 206
Primavera, 59, 207
Print Manager (nxpr), 50, 51, 61, 104, 106, 183
printenv, 103, 165
printing
  charges, 102
  color, 104
  conserving, 104, 106
  default printer, 165
  environment variable, 103
  PostScript, 103, 104
  preview, 103, 106
  Print Manager (nxpr), 104, 183
  printer locations, 14, 15, 67, 102
  printer selection, 103
  queue, 104
  quota, 7, 51, 102, 104, 106, 152, 156, 182, 183
  removing jobs, 104
  selected text, 104
Pro/ENGINEER, 53, 56, 59, 60, 129, 207
ps, 63, 192
pts, 91, 93, 94, 95, 96, 194
pts adduser, 96, 194
pts creategroup, 95, 194
pts delete, 96, 194
pts help, 96, 194
pts membership, 96, 194
pts removeuser, 96, 194
Publishing Your Web Pages, 119
put, 126, 127
pwd, 72, 76, 91, 192

Q

Quantic, 60
quota, 92, 98, 100, 182, 194
R

realm, 2, 46
Realm Kit Application Launcher (RKAL), 3, 28, 50, 63, 85, 197
Recycle Bin, 80, 85
Red Hat, 3, 14, 27, 34, 46, 197
redirection, 88
Remedy, 40, 51
remote access, 132-134
remove, 83
rename, 83
ResNet, 13, 128, 129
restore, 162
rm, 77, 83, 88, 161, 182, 192
rmdir, 77, 193
Rolodex, 54
root directory, 73
Root Menu, 31, 37, 43, 104, 106
root window, 25
router, 64, 67

S

S-Plus, 60, 208
SAS, 53, 56, 59, 60, 207
server, 17, 64, 91, 100, 116
session configuration, 48, 176
set, 87, 165
setenv, 104, 165, 168
shell, 164, 176, 178, 180
  C shell (csh), 25
  command, 25, 50, 146
  environment, 87
  prompt, 25, 62, 76, 164, 169
  variables, 164, 165, 168, 169, 176, 178, 180
shuffle windows, 37
Silvaco, 60, 208
sleep, 174
SlickEdit, see Visual SlickEdit
Smalltalk, 59, 210
SolidWorks, 57, 59, 208
sort, 192
source, 175
spell, 146, 148, 157, 192
ssh, 61, 124-5, 127, 133-134
StarOffice, 59, 60, 208
Start Menu, 32
Sudmit, 114
Synopsys, 60, 209

T

tail, 82, 192
Task Manager, 42, 63
TCP/IP, 64
Tecplot, 53, 59, 60, 209
telnet, 126, 131, 176, 180
terminal window, 50, 124
ticket, 47
TK Solver, 59, 209
Togethersoft, 60, 209
token, 47, 48, 78, 100
TTY, 132, 176, 177

U

unalias, 87, 192
uncompress, 173, 182, 192
Unix, 2, 14, 17, 23, 24, 34, 38, 43, 50, 60, 61, 62, 76, 82, 86, 90, 102, 108, 158, 159, 160, 164, 169
URL, 120, 121
username, 6, 20, 47, 48, 70, 96, 100, 109, 121, 127, 136, 137, 140, 185

V

vi, 61, 131, 132, 143, 147, 160
vim, 61, 147
Visio, 59, 210
Visual SlickEdit, 28, 52, 56, 59, 60, 146, 210
volume, 92, 97, 100, 158, 161
W
Web Mail, 50, 113
WebCT, 112
whois, 185, 193
wildcard, 88
window
copy and paste, 140
root, 172
Xterm (terminal), 24, 25, 169, 170
Windows, 3, 14, 17, 23, 24, 26, 32, 34, 38, 42, 46, 50, 59, 63, 79, 84, 86, 97, 108
Wolfcall, 134
WolfCopy, 7, 102, 152, 183, 229
WolfWare, 71, 111
Word, 56, 59, 84, 206
word processing, 46, 50, 51, 54, 56, 152
WordPerfect, 51, 60, 152-157, 210
working directory, 70, 73
X
X Windows, 2, 34, 46, 50, 52, 54, 129, 131, 170, 172, 176, 177, 178, 206
X11, 34, 170, 178
xpaint, 54, 61
XTRACS, 54, 61, 150
xv, 54, 61
XWin32, 57, 59, 133, 134, 210
Z
ZEN, 3
Zephyr, 8, 40, 48, 61, 131, 136-142, 183
copy and paste, 140
formatting, 138
help instance, 141
subscription, 136, 140
zaway, 142, 193
zctl, 140
zctl hide, 142
zctl unhide, 142
zleave, 193
zlocate, 136, 141, 142, 185, 193
znol, 141, 193
zwgc, 132, 136, 138
zho, 89, 193
zwrite, 90, 137, 193
zip, 158
Zoomtext, 57
Eos and Unity Support

Help: help@ncsu.edu, http://help.ncsu.edu/, and http://www.ncsu.edu/it/

Printing: For print quota, go to WolfCopy Centers. For problems with print quota purchases or printing hardware, email wolfcopy@ncsu.edu. Send all other printing problems to help@ncsu.edu.

Information Technology Division/Computing Services

Box 7109, 208 Hillsborough Building
(919)515-HELP (4357); http://www.ncsu.edu/it/essentials/tools/helpdesk/helpdesk.html

College of Engineering
Information Technology and Engineering Computer Services

Box 7901, 200 and 213 Page Hall
(919) 515-2458; eoshelp@ncsu.edu
http://www.eos.ncsu.edu/

Thomas K. Miller III, Assoc. Dean, Distance Education and Information Technologies
William G. Scott, Director
Charles Hunt, Assistant Director
Tony Baumann, Computer Consultant, Student-Owned Computing
Marhn Fullmer, Systems Programmer
Gary Gatling, Manager, Eos Labs
Rob Grau, Manager, Help Desk
Diane Harper, Administrative Secretary
Troy Hurteau, Web Programmer
Robbie Little, Software Manager
Kathy Mayberry, Coordinator, Student-Owned Computing
Ellen McDaniel, Manager of User Services and Web Coordinator
Lin Osborne, Systems Programmer
Aaron Peeler, Systems Programmer
Josh Thompson, Systems Programmer
Angela Wu, Applications Analyst/Programmer
Jason Young, Manager, Software and Systems