Introduction to X and UNIX Tutorial *

University of Waterloo

Version 1.1

Peter A. Buhr and Caroline Kierstead

December 11, 1998

*For Further Information: http://www.math.uwaterloo.ca/mfcf/faq/index.html

Contents

1 Introduction

When first confronted with a UNIX system with an X interface, many beginners find themselves floundering in a sea of details. The major reason for this difficulty is that most aspects of UNIX are aimed at providing a productive environment for advanced users. This design bias can be seen in many areas:

- The UNIX command syntax is very terse.
- Documentation is designed to make it easy to look up details on commands that are already understood.
- UNIX software offers a great deal of functionality, but does little “hand-holding”. For example, prompts for input are usually very terse since verbose prompts are often annoying to advanced users.

Once you are past your apprenticeship on UNIX, you should find it a productive system on which to interact with a computer. The goal of this tutorial is to get over the initial hurdles so that you will be off and running on your own.

This tutorial is designed to give a very basic introduction to the X environment and the UNIX file structure and command interface. The tutorial is divided into several sections, each dealing with a major concept. By working through the exercises in each section, basic mechanisms are introduced and can be practiced. While you can work through the tutorial at your own pace, try to complete an entire section at a time to maintain continuity. The tutorial is not intended as a complete instructional guide. Additional manuals on X, “An X-window User’s Guide”, and UNIX, “An Introduction to UNIX”, are sold in MC1052 (the IST Computer Help and Information Place). As well, many books are available on these two topics. Finally, X/UNIX consultants are also available (see Section 7.1).

Throughout the tutorial, the following symbols are used:

$\Rightarrow$ This symbol indicates that you are to perform the action marked by the arrow.

$\bowtie$ This symbol indicates that the section explains a concept that may be unfamiliar even if you have some previous experience using a computer (e.g., Windows). Make sure you understand this concept before advancing in the tutorial.

2 MFCF Undergraduate Computing Environment

Eventually you will use most of the MFCF undergraduate computing environment, so it is reasonable to look at it in some detail now.

A computing environment is composed of various functions, one or more of which are combined within a single machine. The division of functions within the Math Faculty instructional computing environment (often called the ”undergrad” environment, although MFCF does have a single machine with that name) are:

- displays
- windowing
- servers
- CPU
- files
  - home directories
  - applications
- world wide web (WWW)
- mail
- news

Several approaches are used in the undergrad environment to distribute this functionality. In all cases, a user’s files are stored remotely on a separate file server. There are also separate servers providing WWW, mail, and news service. All systems are connected by a local area network (ethernet and fast ethernet).

The conventional approach for providing computing functionality uses a PC or Mac, merging the display, windowing, and CPU services into a desktop unit. The ”thin client” approach uses a display on the desktop, with possibly some services such as window management and WWW browsers as well. Servers for windowing, files, WWW, mail, and news are accessed remotely. When you log onto a terminal in the Unix environment, one of the many machines that constitute the teaching environment is selected for you, in a way that tries to distribute the load equally among the appropriate machines.

It is important to note that the Undergrad environment comprises multiple, incompatible architectures. The PCs run Polaris Win95, Macs run MacJanet MacOS, and the various Unix systems may run SunOS, multiple versions of Solaris, or other operating systems. So programs compiled on one system may not run on a different system (e.g., error message "bad magic number"). Even data files generated on the different architectures may not work correctly (big endian versus little endian problems). Character files are portable between Unix systems, but there may be problems when transferring them between Unix and non-Unix systems, e.g., a PC (due to the difference between Unix and Microsoft newline conventions). In such cases, you will have to add or delete extra carriage returns as necessary.

Not all the Unix systems provide the same services/commands, called packages, due to licensing or space or computation restrictions. To determine which servers provide which packages enter:

```
hostselect +h
```

To find out where a particular package can be run enter:

hostselect *packagename*

Detailed information about all of the MFCF computers is available online (man mfcf-machines, see Section 7.1 for what this means). More information about the instructional environment is also available online starting with man mfcf-course-environment.

2.1 MFCF Computing Policy

Anyone using an MFCF computer is advised to read the following policy statement, first issued by the Mathematics Faculty Computing Facility in June 1984, and last updated in September 1997.

The Mathematics Faculty Computing Facility (MFCF) provides computing resources on request to students and faculty within the Mathematics Faculty, and to some extent to members of the wider community. The provision of a computer account by MFCF carries with it certain conditions and responsibilities which this policy statement endeavours to explain. Misuse of accounts or violation of the delineated conditions may result in the termination of the accounts, or in cases of more serious infractions - the submission of the case to the appropriate Disciplinary Advisory Committee of the Faculty. (See the section of the University Undergraduate Calendar headed Academic Regulations and Student Discipline. In particular note items II (1), (4) and (8)).

1. All computing resources (e.g. computer accounts, printer access, network access) are provided for the express purpose for which they were requested. Use of the resources for any other purpose is considered inappropriate. In this context students provided with resources for carrying out work related to a course or courses or project or thesis should confine the use of the resources to activities related to this work. It is inappropriate to use resources for personal use.

2. Computing resources are to be used only by the person or group to whom the resources are given. This implies that it is inappropriate to share an account with others (e.g., your best friend!) or to obtain illicitly access to another's account.

3. Unethical conduct using a computer account is inappropriate. (See UW Policy 33, Ethical Behaviour.) For example, mailing messages which the recipient considers obscene, abusive, or offensive is inappropriate, as is the display in public terminal areas of images that others find offensive.

4. Attempting to attain unauthorized access to another's directories or files or in any way attempting to 'break' system security is deemed unacceptable behaviour, even if it is for purposes of browsing only and not for acquiring data. If a user is able to circumvent the system security, it should be considered an obligation to inform MFCF of this possible breach (so that MFCF may take steps to prevent further occurrences).

5. Monopolizing system resources will be deemed improper behaviour and a misuse of the resources. Examples are locking a public terminal for more than 15 minutes, playing computer games, or attempting to defeat the normal deletion of temporary files (e.g. from /tmp and /usr/tmp).

6. Public computing areas (most of which are terminal rooms) are to be treated with the same respect that would be given to a study area. In general, disruptive or destructive behaviour is inappropriate. Examples of such are

loud or extended conversations, the display of images found offensive by others, consumption of food or drink, or listening to music without headphones.

The following conditions (restrictions), originally issued by the Mathematics Faculty Computing Facility in July 1987, apply to all (Unix) student accounts.

1. Undergraduate students are allocated resources for academic purposes based upon the courses being taken. Additional disk quota may be requested by instructors for students registered in particular courses.

   Students are granted access to laser printers for academic purposes only. In general, all printing must be paid for. Printer pages can be obtained by purchasing a CAS account from IST at the Computer Help and Information Place in MC1052. Graduate students who are registered for a thesis or an essay receive a quota dependent on the requirements. man mfcf-class-resources shows current printer (and disk space) allocations for all courses.

2. No tape jobs are performed for student accounts.

3. Requests for restoration of files after a student account has been deleted, or after the end of term, will normally not be granted.

4. Monopolizing printing resources will be deemed improper behaviour and a misuse of the resources, e.g. printing large amounts of documentation, such as manuals or course notes, on the self-serve printers intended for the results of assignments. Instructors should arrange that documentation intended to be used in print form will be available for purchase.

5. Internet access is not usually a curriculum requirement, however it is recognized that responsible use of the Internet can enhance the academic environment. While the Faculty would like to provide continued access to the Internet, this will only be possible with the continued cooperation of students. Students must be professionally responsible in the use of this technology, and should be aware of current pressures on us all for economic restraint, political awareness, and financial accountability.

   Posting articles that violate the guidelines in the netiquette document found in the news.announce.newusers newsgroup is irresponsible. An example of such irresponsible use is posting ads for pyramid money making schemes.

3 Getting Started

3.1 Userid

You must have a UNIX account before you can sign on to any UNIX system. Undergraduate students in the Faculty of Mathematics automatically receive an account on the MFCF Undergraduate Environment which will normally last for their entire undergraduate career. Students from other faculties who are taking Math or CS courses will also get a Unix account for the duration of that course.

Students taking specific courses may get an account on a PC (e.g., in the Cygnus Lab) or a Macintosh. Cygnus passwords are set by the Consultant, and are NOT the same as a Unix password. Therefore, to prevent two trips, it is to your advantage to tell the Consultant if you will also need a Cygnus (PC lab) password. Information about passwords for the Macintosh labs is given out in class.

Graduate students in the Faculty of Mathematics automatically receive an account on the machine barrow and the machine mercator. Graduate students in Computer Science receive an additional account on the machine neumann. Other UNIX machines are used for various types of research—people wishing to do research on one of these machines must request an account from the appropriate research group.

Each UNIX account is given a unique name called a *userid*. In most cases, you will be given a single userid for your entire career at the University of Waterloo.

A userid normally consists of your initials followed by your last name, as shown on your student id card, truncated to 8 characters in length. For example, *John William Riemannson* has userid jwrieman. Occasionally, a userid may contain a number after the initials if there are multiple users that generate the same userid.

NOTE: the uwuserid that is generated by the Registrar's office is usually the same as the userid given to you by MFCF. However, in rare cases, they do not match up, and this can cause problems setting up printer and terminal server accounts. If you find that you have this problem, please send mail to admin_sys@math.

To determine your userid, you can use help userid. On a terminal, enter help as the userid and nothing (just hit $< RETURN >$) for the password. You will then be prompted for your student number. Once you have typed that your account information will be displayed.

To enable your account, go to the MFCF Consultant in MC3008 with your student card. The Consultant will give you a password for your account. This password does not expire, but you should change it as soon as possible for security reasons. Wait approximately an hour for the password to be transmitted to all MFCF computers on the network before trying to sign on.

3.2 Password

During the hour wait, decide on a new password to protect your account from unauthorized access by others. Each time you sign on, you are required to enter your password to ensure that only you can access your account.

A password is constructed using the following rules:

- must be longer than 6 characters (but only the first 8 characters are significant), and use at least 5 different characters
- the characters may be letters (upper or lower case), digits, punctuation and space
- if entirely of letters, at least 2 letters must be another case (i.e. mixed upper and lower case letters) and at least 1 of the letters in another case must be inside the password (i.e. neither the first nor the last character)
- not be a sequence of characters that appears in the local UNIX dictionary or its reverse

The best sequence of characters for a password is one that is neither a word nor a name in any language. The sequence of characters should have meaning only to you, which makes it very difficult to guess or read while being typed. One approach that often produces memorable but effective passwords is to use the first letter of each word in a phrase, alternating the case.

Do *not* write your password down—if you forget it, a system administrator can always reset it to something that you do know. Do *not* tell anyone your password. Note that allowing anyone else to use your account on an MFCF administered system, even a friend or project partner, is a violation of MFCF computing policy. This policy may seem inconvenient, but there are very good reasons for it. Users who need shared access to files or other resources can arrange such access by means other than exchanging passwords. *NEVER* reply to mail requesting your password. A system administrator never needs to know your password. If you receive such a request, do not reply, but forward the mail to your local system administrator, e.g. admin_sys@math. *Do* change your password once in a while. Whenever you type in your password, try to ensure that no one else can obtain it as you type it in.

3.3 Logging On

After an hour's wait and armed with your password, you are ready to sign on, or login to the MFCF computer system. The following rooms contain terminals for general undergraduate course work (see man mfcf-terminal-rooms for a complete listing):

MC2017, MC2067, MC3003, MC3009 The systems in these rooms are Macintoshes.

MC3004, MC3005, MC3027 The systems in these rooms are PCs.

The Macintoshes and PCs can only be used by those registered in a course that requires them. Your instructor will let you know if this is the case.

MC3006, MC3007, MC3008A, MC3018, MC3022, and MC6080 The terminals in these rooms are high-resolution X terminals, and have either colour or black and white displays. (If the screen is completely grey, turn the terminal off and on to reboot it.)

X terminals have a display similar to that in Figure 1.

Please do not use terminals in other rooms. Please do not bring food or drinks into the terminal rooms. Violations of these rules may result in your account being disabled for at least 24 hours.

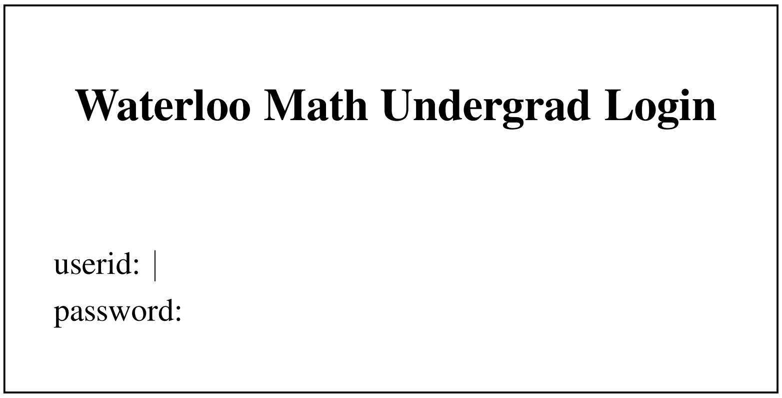

Figure 1: X Login Window

- ⇒ Enter your *userid* after the login: or userid: prompt.
- ⇒ Press $< Return >$ after entering your userid.
- ⇒ You are now prompted for your password. If this is the first time you have signed on, enter the password given to you by the Consultant. Otherwise, your password is the same as that from last term. NOTE: your password is not displayed on the screen as you type it nor does the cursor move. This is a security precaution.
- ⇒ Press $< Return >$ after entering your password.

If you make a mistake in either of the previous steps, you are prompted for your userid and password again. If the login continues to fail, you can query your account to see if it is active by entering the userid help (just hit $< Return >$ for the password) and answer the prompts. If your account does not appear, go back to the MFCF Consultant in MC3008 and explain your problem.

You may skip to Section 5 if you are not using an X terminal.

4 Basic X Environment

Various windows now cover the majority of your screen. See Figure 2 for an approximate picture of the screen. In the centre of the screen is something that looks like an "I", which is one of the many shapes for the "cursor".

- ⇒ Move the mouse The cursor moves in a corresponding fashion across the screen.
- ⇒ Move the cursor all over the screen, especially over some of the windows that appear on the screen. Notice that as the cursor is moved around the screen, it changes shape depending on its location.
- ⇒ Move the cursor over the grey background area (the "root window"); it turns into a "X".

The cursor is the primary way of pointing and selecting things on an X terminal. Input, such as typing, or clicking the mouse buttons, is directed to the window under the cursor. The cursor changes shape over certain windows to make it clear that the result of input differs depending on the cursor location.

If this is your first login to your account, a window labelled "First Login" is present in the centre of the screen; if this is not your first login, go to the next paragraph.

- ⇒ Move the cursor onto the "First Login" window and follow any instructions by typing at the keyboard.
- ⇒ You are prompted for personal information, which can be accessed by other users (see Section 7.2). Provide as much information as you feel comfortable with; you are not required to provide any personal information. (See section 7.2 to understand more about how this information is used and how to change it.)
- ⇒ You are also prompted to change from the initial password given to you by the Consultant to the new password you selected. You are asked to type in the initial password and then you are prompted twice for your new password. The double prompting is to avoid spelling mistakes. Remember that UNIX is case-sensitive! NOTE: you will not be able to see the passwords that you have typed in for security reasons. Your password can be changed at any time by using the passwd command (see Section 7.1). NOTE: it normally takes at least an hour for your new password to circulate to all of the MFCF undergraduate computers.

When you have finished entering all of the required information, the "First Login" window disappears. Unless you re-initialize your home directory, the "First Login" window does not appear again.

Figure 3 shows an approximate picture of the screen for a normal login. A large Important ... window with system news is in the centre of the screen. Some of the news is not immediately relevant, but you should always check the system news for any important announcements (especially machine down times).

⇒ Move the cursor to the Important ... window.
⇒ Read the information presented in the window and follow any instructions. Press the space bar to continue and "Q" or "q" at any time to quit.
⇒ Once you have finished reading this news, the Important ... window is terminated automatically.

Two large windows, TopXterm and BottomXterm, are now present in the middle of the screen; these windows are located on a CPU server that currently has a low load. The machine name is in parenthesis after the window name. A CPU server is simply a machine that is intended for resource consumptive computing, e.g. compiling programs or formatting text.

There is also a partially hidden window in the bottom left corner, called the console window. The console window is special: it displays any X error messages, and when terminated, the entire session is terminated (i.e., all programs are stopped, the windows go away, and the original login window reappears). In the top left-hand corner of the screen is an analog clock and below that is the icon manager.

How a window looks and behaves when input is directed to it depends on the program that created the window. The large windows are controlled by a program called "xterm", which simulates a tty terminal running a command shell (see Section 5). In effect, there are three tty terminals on the X-terminal screen at this point.

4.1 Icon Manager

Multiple large windows can quickly clutter the screen. To remove clutter, it is possible to temporarily shrink a window into a small image that occupies much less space; this small image is called an *icon* and may take any shape. The action of transforming a window into an icon and subsequently transforming the icon back into a window is called *iconifying* and *de-iconifying*, respectively.

An icon manager provides a convenient mechanism to manage the space on the screen through iconifying and de-iconifying windows, and it organizes the icons themselves together into a single window. Figure 4 describes the parts making up the icon manager's window, and the icon manager's window is normally located below the clock. In general, it does not make sense to iconify the icon manager's window.

⇒ Move the cursor to the first window listed in the icon manager's window.
⇒ Click the left mouse button. A small square with an "X" appears next to the name and the window that corresponds to that name is iconified.
⇒ Iconify each window in the list.
⇒ Move the cursor to the list entry named TopXterm.
⇒ Click the left mouse button again. The "X" disappears and the window is de-iconified and reappears on the screen.
⇒ Move the cursor into the TopXterm window and notice that the corresponding entry in the icon manager window is highlighted with a dark border.
⇒ Move the cursor in and out of the TopXterm window and the watch the icon manager window. As well, a shaded rectangle appears at the top of the TopXterm window and its border is highlighted, too. Both forms of highlighting indicate the active window, i.e., the window that is currently receiving input from the mouse or keyboard.

4.2 Parts of a Window

See Figure 5 for a description of the parts making up a window. By default, application windows are surrounded by a "frame" with a titlebar at the top and a special border around the window. The titlebar contains the window's name, a rectangle that is highlighted when the window is active, and function boxes known as "titlebuttons" at the left and right edges of the titlebar.

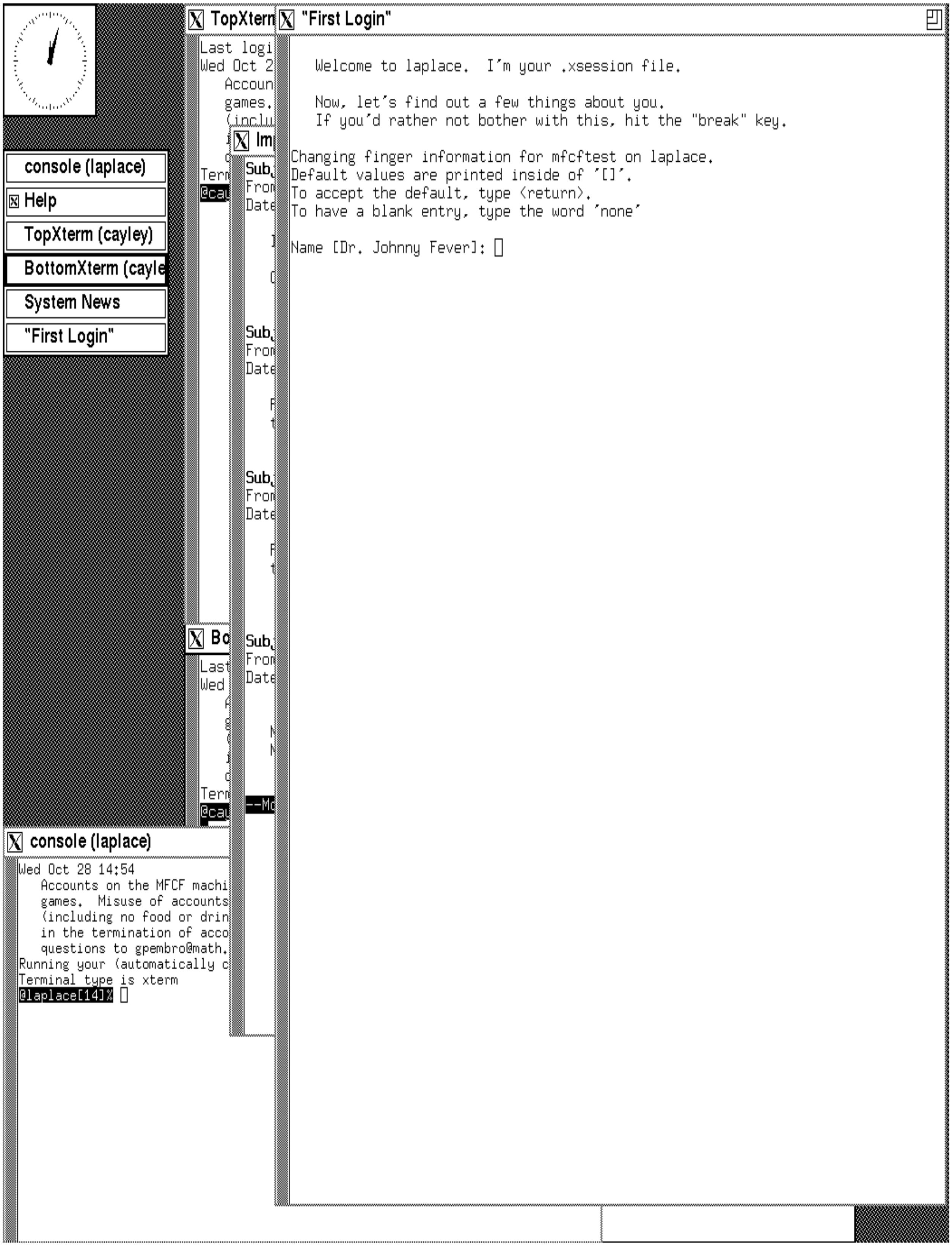

Figure 2: Initial X Environment

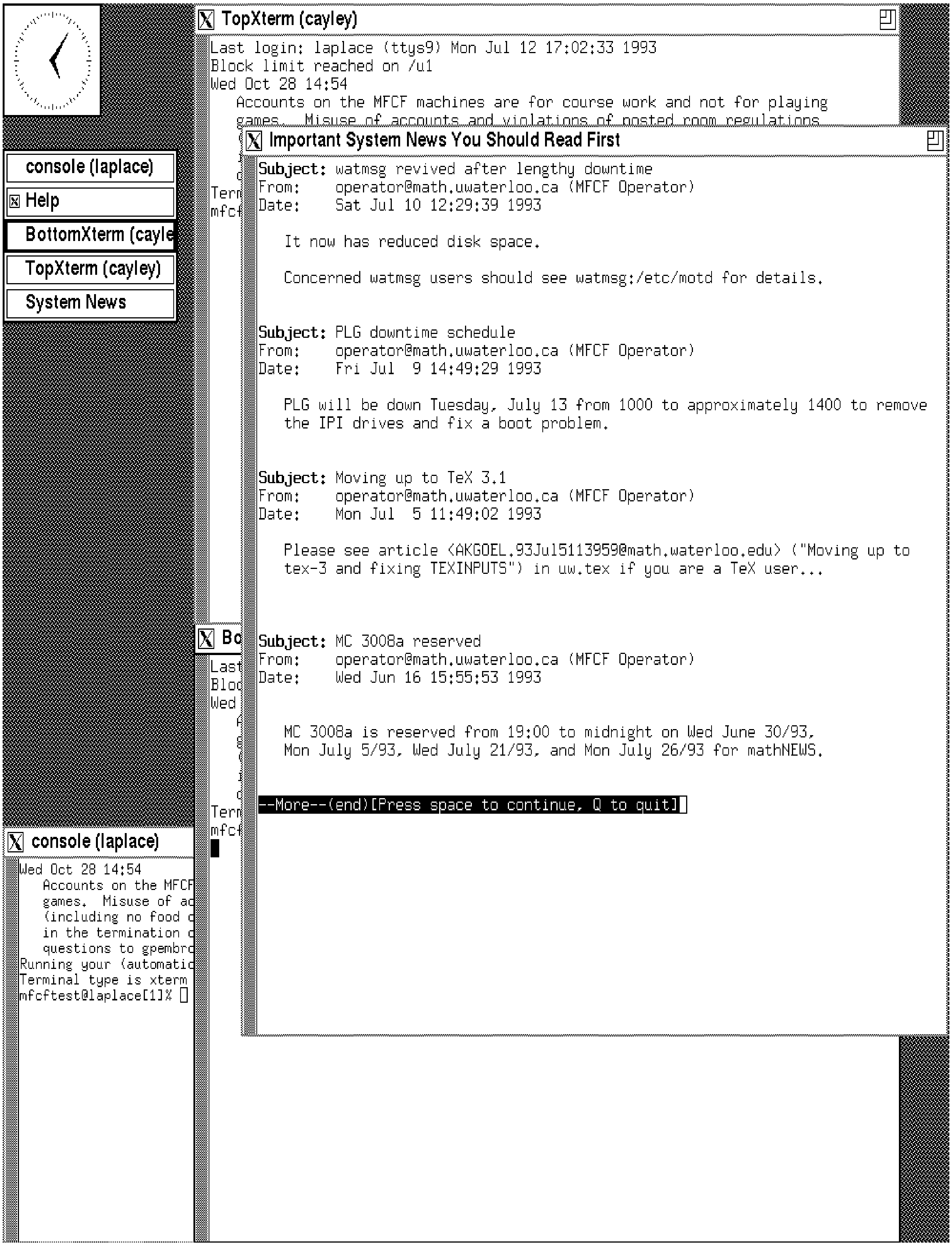

Figure 3: Subsequent X Environment

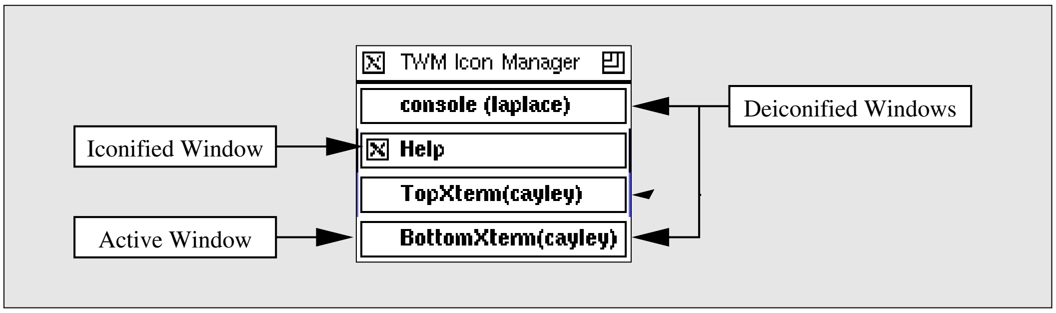

Figure 4: Icon Manager

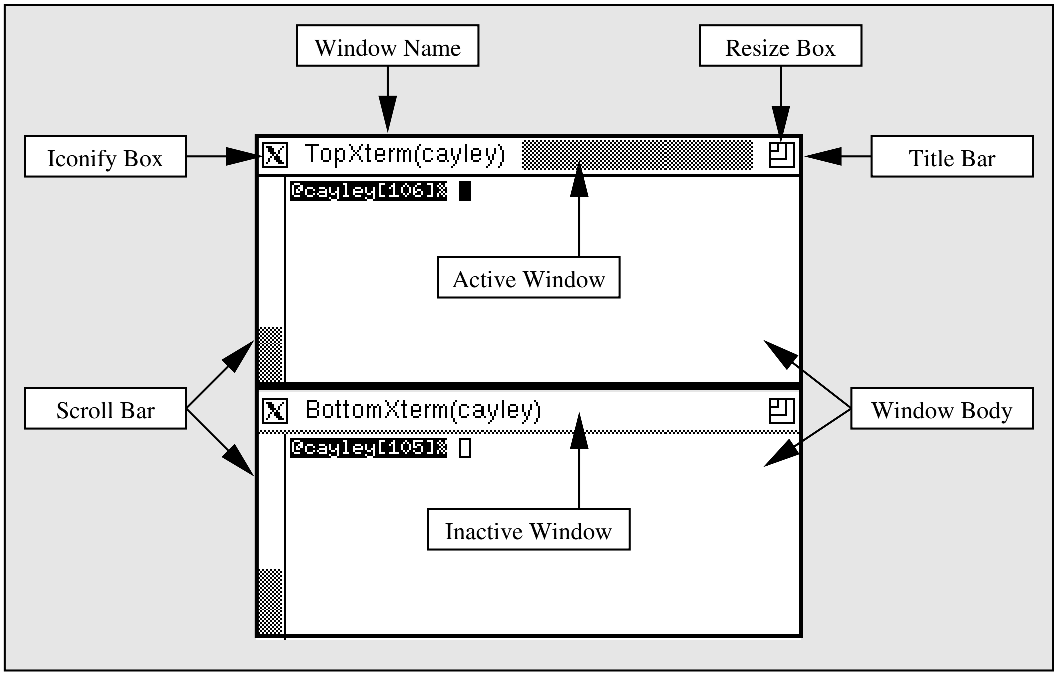

Figure 5: Window Parts

4.3 Iconifying and Deiconifying a Window

As seen already, a window can be iconfied and de-iconified through the Icon Manager; here are some alternative ways. A window can be iconified directly by pressing a mouse button while the cursor is located in the button marked by an "X" in the top, left-hand corner of a window's titlebar.

⇒ Move the cursor to the iconify box of the TopXterm window.
⇒ Click with any mouse button. The window is iconified.
⇒ Move the cursor to the window named TopXterm in the Icon Manager list.
⇒ Deiconify the window.

4.4 Resizing a Window

Resizing requires a new action with the mouse: keeping a button pressed while moving the cursor is called "dragging". Windows are resized by pressing the right titlebutton (described below), which resembles a group of nested squares, dragging the cursor beyond any existing edge of the window to enlarge the window, and releasing the cursor when the outline of the window is the desired size. Once an edge is "grabbed" by crossing over it, the window can also be made smaller by crossing back towards the middle of the window. By moving the cursor through a corner, it is possible to grab two edges and resize both simultaneously.

⇒ Move the cursor to the resize box of the TopXterm window.
⇒ Click a mouse button and drag on the window outline to resize the window to make it fill the top right half of the screen.
⇒ Let go of the mouse button when the window is the correct size.
⇒ Perform the same actions but make the window slightly smaller.

4.5 Moving a Window

Windows are moved by pressing the left mouse button while the cursor is in the title region, dragging a window outline to a new location, and then releasing the button when the outline is in the desired position.

⇒ Move the TopXterm window so it is in the top-right half of the screen.

4.6 Raising and Lowering a Window

When using overlapping windows, one window may obscure all or part of another window (like a piece of paper completely or partially covering other pieces of paper on a desk top). Raising a window means to move the partially obscured window in front of (or to the top of) a pile of windows. Clicking the left mouse button in the title or highlight region of a window *raises* it without moving it. Lowering a window means to obscured a window in back of (or to the bottom of) a pile of windows. Clicking the right mouse button in the title or highlight region of a window *lowers* it without moving it.

⇒ Deiconify the BottomXterm window.
⇒ Move it so that it partially covers the TopXterm window.
⇒ Click the left mouse button in the title bar of TopXterm.
⇒ Click the right mouse button in the title bar of TopXterm.

4.7 Scrolling a Window

The narrow column along the left-hand side of a window is called the "scroll bar". The xterm program keeps a limited history of what has been displayed. With the cursor positioned on the scroll bar, the left mouse button shifts the displayed contents forward in fixed increments, and the right mouse button shifts the displayed contents backward in fixed increments. The shaded portion shows the relative position of what is currently displayed. The middle button allows scrolling what is displayed up and down as long as the button is pressed. If the portion of the window being displayed is not at the bottom, any output causes the window to jump to the end and be displayed. Since input is echoed, the same is true if you type anything.

Since none of the windows currently contain enough information to actually scroll text, you will have to practise using the scroll bars at a later time.

4.8 Cutting and Pasting Between Windows

It is possible to copy text from one window and insert it into another window as if it had been typed there, called "cutting and pasting" text. Position the cursor in the window body of an xterm window and drag the cursor up or down while pressing the left mouse button. Everything between the beginning of the drag and the current cursor position is highlighted. When you release the button, all the highlighted text is copied. Moving the cursor to a new location and pressing the middle button pastes this text as if you had typed it. Clicking the right button grabs all the text from the top left corner to the cursor position.[1] Two fast clicks ("double-clicking") selects just a single word. A triple-click selects a whole line. To undo a selection, click again until nothing is selected. Text that has been cut can be pasted into any window.

⇒ Deiconify the BottomXterm window and make it the active window.
⇒ Enter the command who am i on the command line in BottomXterm. Do *not* press $< Return >$ yet.
⇒ Cut the text who am i.
⇒ Paste it into TopXterm.

[1] Do not attempt to cut a whole screenful of information as xterm may lose some of the information.

⇒ Press $< Return >$.
⇒ Move the cursor back to BottomXterm.
⇒ Press $< Return >$.
⇒ Iconify BottomXterm.

4.9 Window Manager

A window manager is the program that controls the "root window" and the format of all windows that appear on the X-terminal screen. twm, Tab Window Manager,[2] is the window manager shown in the previous sections. Some alternate window managers are awm, dwim, gwm, mwm (like Microsoft Windows), tvtwm (like twm but with a "Virtual Desktop" feature that makes the root window of the screen larger than the physical display), and uwm. Selecting a window manager is a matter of personal preference. Some window managers are arguably better than others offering more features and user customization capabilities. Spend some time learning twm before trying out other window managers.

4.9.1 twm Operations

twm provides several basic functions that can be accessed through a number of pop-up menus. By positioning the cursor onto the grey background (root window), you can bring up menus by clicking the mouse buttons.

⇒ Move the cursor to the root window.
⇒ Press the left mouse button, examine the pop-up window, and then release the mouse button.
⇒ Do the same for the middle and right mouse buttons.

See Figure 6 for a list of some of the available menus.

By keeping a button pressed and dragging the cursor down the menu list, each item is highlighted when the cursor appears over it. If a button is released while an item is highlighted, the menu item is selected and a corresponding action takes place. Every menu has a Help item describing that menu.

The left mouse button pops-up a Create Windows menu to start a new xterm window; the specific items control where the new window is created and where the shell for the xterm is started. This operation is available through a command in other xterm windows but is most conveniently done through the window manager menu. The middle button pops-up an Applications menu, which creates a window running the selected application. These applications are available through commands in xterm windows (discussed later), so this menu is not normally used (in fact these commands may not work correctly so avoid using this menu). The right button pops-up a Window Ops menu that provides a number of window manipulation items, e.g., resizing a window, moving a window, etc. These window operations are available directly from each window or the icon manager, so this menu is not normally used.

4.9.2 Starting an xterm Window

Creating an additional xterm window through the twm menu is convenient and necessary if all the other xterm windows are busy performing other operations.

⇒ Move the cursor onto the root window.
⇒ Press the left mouse button to bring up the Create Windows menu.
⇒ Look at the title for window TopXterm to determine your home computer (the name in parenthesis).
⇒ Drag the cursor down the list of options to the machine with the same name. Upon releasing the mouse button, a window outline appears, attached to the cursor.
⇒ Move the window outline to any location on the screen.
⇒ Press a mouse button to attach it to this location.
⇒ Move the cursor into the new window.
⇒ Enter the command exit which terminates the shell and xterm window.

[2]Portions ©1988 Evans & Sutherland Computer Corporation; portions ©1989 Hewlett-Packard Company and the Massachusetts Institute of Technology; see X(1) for a full statement of rights and permissions.

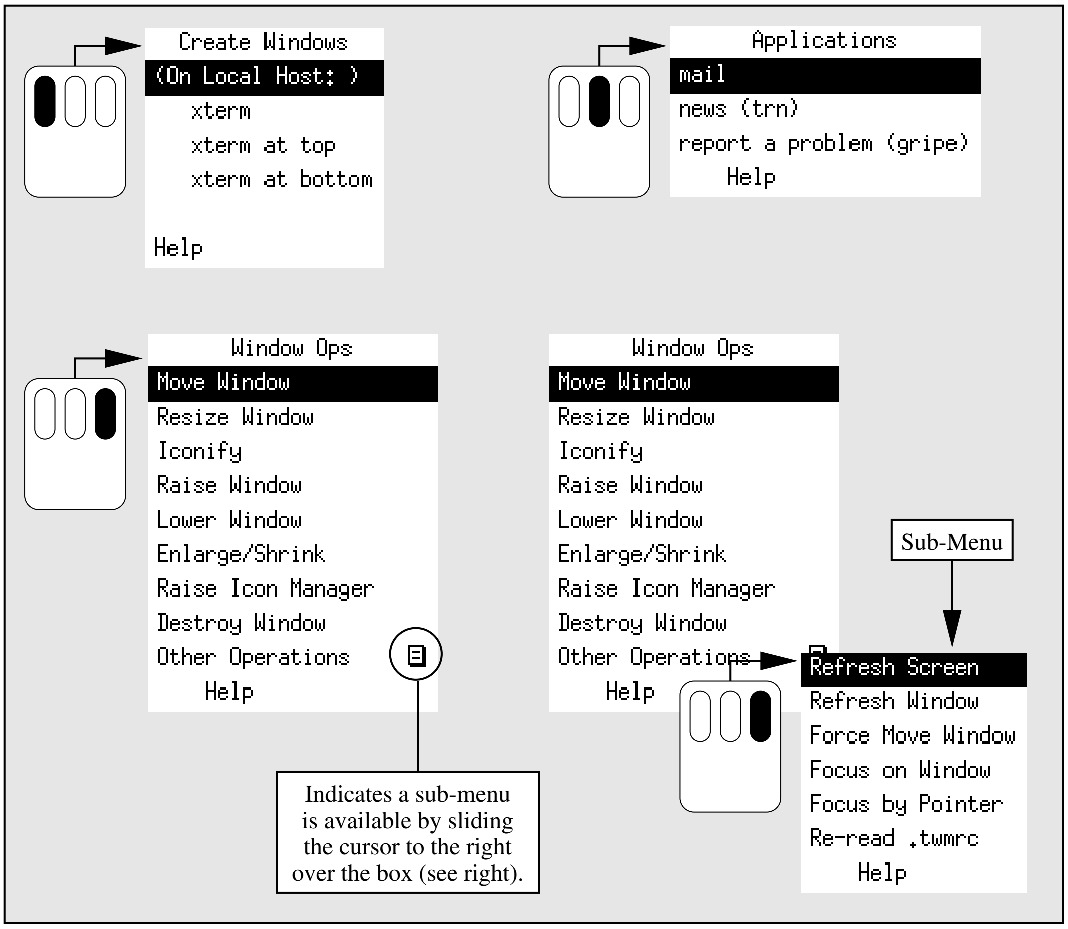

Figure 6: Menus

4.9.3 Window Manager Options

Most window managers provide a large number of options to control the behaviour and look of the windows they create (called "look and feel"). The file that controls which window manager is invoked at startup is called .xsession. This file is executed by X after signing on to an X-terminal. The file is fully commented indicating what commands perform which specific functions. By modifying the .xsession file, you can change what window manager, windows and utilities are normally started up. Once a window manager is started, it usually looks for a file whose name is of the form .*rc, where * is the window manager name, like .twmrc, for window-manager option settings. For twm this file contains specific option settings, like adjusting the size of the window border or the fonts used for printing text in the windows. There are too many options to discuss here (see man twm).

4.10 X Utilities

There are many X utilities available. Some useful utilities are:

xbiff++ displays a little image of a mailbox. When there is no new mail, the flag on the mailbox is down. When mail arrives, the flag goes up and the mailbox beeps (or optionally plays a sound).

xcalc is a scientific calculator desktop accessory that can emulate a TI-30 or an HP-10C.

xcalendar is a simple interactive calendar program with a notebook capability.

xclock displays the time in analog or digital form. The time is continuously updated at a frequency that may be specified by the user.

xload displays a periodically updating histogram of the system load average.

For the best performance, run X utilities from the console window. If you are not actively using a utility, you should terminate it. Utilities that are iconified can still use up CPU and memory on both the server and client computers.

4.11 Making Use of a Large Screen

High-resolution bit-mapped displays in conjunction with a powerful CPU to drive them have produced a new style of interacting with a computer. This style involves displaying more information directly on the screen or making that information quickly accessible in pull down or pop up menus. Having more information directly accessible speeds up a user's work process because the likelihood that the required information or action that the user needs is on the screen is increased. This eliminates typing, having to remember information, and waiting for the information to be found and redisplayed on the screen. Basically, the more information that can be displayed on the terminal about a problem, the easier it is to comprehend the problem and form a solution.

Here are some tips on using a large-screen display to get the most benefit from it.

- Make large windows with text as small as you can comfortably read. This style maximizes the amount of information in the window. It is not unusual to see a person working in a 24x80 window in the middle of a large screen; this person should use one of the tty terminals as they are wasting the capabilities of the large screen.
- When working on a specific problem, use a single large window and tile that window, if possible, with smaller panes of information. An editor like emacs tiles a window with multiple panes. Tiling allows multiple panes of information to be simultaneously visible without the need to explicitly open and move windows with the mouse because tiling systems automatically manage the panes within a window.
- Use overlapping windows for large changes in context but do not overlap windows. For example, when changing from one computer to another to work on a different problem, iconify the current window and de-iconify the window containing the new work. Do not waste screen space with parts of windows that you are not using, just because you might need to access those windows sometime in the future. In the long run, you are reducing your overall productivity by using less screen space just so you can access other windows more quickly. Having to iconify and de-iconify to look at other windows is not a time consuming operation.
- Do not waste valuable screen space with clocks, CPU utilization graphs or other screen garbage. These programs waste memory and CPU time on both the client and server computers, while providing information that can be obtained on demand whenever it is needed (e.g., UNIX commands date, top and ruptime). They also occupy valuable screen space, while providing no benefit in solving the problem at hand.

5 Command Shell

Each of the xterm windows runs a program called a "shell", which accepts UNIX commands and executes them. (A tty terminal is the equivalent of one fixed sized xterm window.) The default shell program is csh or the *C shell*. In the C shell, the prompt is denoted by a % sign. When typing a UNIX command, you can backspace over the command, insert and delete characters. A command is *not* executed until $< Return >$ is pressed.

Try entering the following commands in one of the xterm windows, pressing $< Return >$ after each:

⇒ date
⇒ whoami
⇒ w

These commands print the current date, your userid, and the userids of others users logged on, respectively. NOTE: UNIX is case-sensitive, so the command Date is not the same as date.

Many commands in UNIX have options, specified by a hyphen and then a character, which can change how the command entered operates.

⇒ Enter the command: w -s

A short form of the previous list of users is printed. Unfortunately, there is no standardization for option syntax and names. This lack of uniformity is confusing and forces users to constantly look up each command before using it; this problem is one of the major failings of the UNIX system.

5.1 Logging Off

You can logoff by issuing the command logout or exit. Now is a good time to sign off and sign back on again to test what you have learned about start-up and to try your new password.

⇒ If you are using an X terminal, de-inconify the "console" window and move the cursor to it.
⇒ Enter logout or exit after the command prompt. On an X-window terminal all the windows should disappear and the initial login window appears. It takes about 30 seconds for the initial login window to appear so be patient.
⇒ Sign on again.

5.2 Hierarchical File System

Files are containers for data stored on secondary storage (e.g., tape or disk). UNIX organizes names for these files in a hierarchical manner: directories form the hierarchical structure and the leaves are files. Figure 7 shows the basic structure of the UNIX file system, where indentation represents the different levels in the hierarchy. There are actually many directories and files not shown; you should know about these basic directories.

```
/           root of the local file system
    alex        root of the internet file system
    software    all MFCF software is organized under this directory
    usr
        dict        local dictionary
        bin     more UNIX commands
        lib     system libraries
        include     system include files, .h files
        ucb     Berkeley UNIX commands
    bin     basic UNIX commands
    lib     system libraries
    tmp     system temporary files
    u1      user files
    u2      user files
        ...
        jfdoe       student home directory
            .cshrc, .emacs, .login, ...       student's files
            cs246
                assign1
                    q1, q2, q3
        ...
    u3      user files
    u4      user files
    u5      user files
    u       magic directory combining what is under u1-u5
```

Figure 7: Basic UNIX File Hierarchy

alex – The root of the internet file hierarchy. It is possible to walk through this directory to other computers that are linked to the internet.

software – MFCF organizes all of its executable, include and library files for the entire system under this directory.

usr, etc, bin, lib – The location of most of the UNIX commands and system include and library files.

tmp – Temporary files can be created under this directory. The editor vi creates a backup versions of the file being edited under this directory. Many other commands create temporary files under this directory during their execution. It is not uncommon for this directory to fill with temporary files, and hence, prevent the execution of commands that create temporary files under this directory. If you explicitly create a temporary file under this directory, make sure to delete it as soon as possible. Also, files with names beginning with a # are automatically removed within a week or two.

u1, u2, u3, u4, u5 – All the user files are distributed evenly across these directories. The UNIX convention is that each userid has a directory with the same name as the userid under one of the user directories. For example, userid jfdoe has a directory jfdoe under which more files and directories can be created for that userid. Although the userid is truncated to 8 characters, the directory name is not. So from our earlier example, *John William Riemannson* who has userid jwrieman will not have a directory called jwrieman, it will be jwriemannson. The directory for a particular user is called that user's "home" directory.

u A "magic" directory that contains all of the users under the individual user directories. Therefore, each user's home directory appears under two directories. This means that users do not have to remember which user directory their home directory is located under, they just look under directory u.

A file in the hierarchy can be referenced in one of two ways: with an absolute pathname or a relative pathname. An absolute pathname is a list of all the directories from the root to the file separated with the character "/". Absolute pathnames always start with the character "/", which indicates the hierarchy root. For example, the absolute pathname /u/jfdoe/cs246/assign1/q1 denotes the file q1; the name /u2/jfdoe/cs246/assign1/q1 denotes the same file.

A relative pathname requires the notion of a starting location other than the root of the hierarchy; this location is called the "current directory". When you sign on, the current directory is set to your home directory; any file name that does not start with a "/" is automatically prefixed with the current directory. For example, if user jfdoe signs on and specifies the file name cs246/assign1/q1, then the actual file that is used is /u/jfdoe/cs246/assign1/q1. There are commands to set and reset the value of the current directory. Finally, there are two special directory names dot, ".", and dot dot, "..". Dot is the name of the current directory, so ./cs246/assign1/q1 is the same as /u/jfdoe/cs246/assign1/q1. Dot dot is the name of the directory above the current directory, i.e., the parent directory, so ../jfdoe/cs246/assign1/q1 is the same as /u/jfdoe/cs246/assign1/q1. In summary, dot dot is used to move upwards in the directory hierarchy and "/" is used to move downwards. There is one more special directory name, "~", which is name of your home directory, so ~/cs246/assign1/q1 is the same as /u/jfdoe/cs246/assign1/q1.

5.2.1 Pattern Matching

The shell supports pattern matching of file names to reduce typing lists of file names. Pattern matching is provided through special characters. (Different shells and commands support slightly different forms and syntax for their pattern expressions so always check the documentation.) The special characters are *, ?, [, {. (To enter a file name containing one of these special characters, it must be enclosed in quotes "".)

The only pattern matching character discussed here is the *, called a wild card, which matches 0 or more characters. If the current directory is /u/jfdoe/cs246/assign1, then the file name q*, matches file names q1, q2, q3. Multiple wild cards can appear in a pattern expression, e.g., the expression q*a*.c matches any file name that starts with the letter q followed by 0 or more characters, followed by the letter a, followed by 0 or more characters, and terminating with the two characters ".c".

5.2.2 Current Directory

The command pwd prints the current directory.

⇒ Enter the command: pwd

5.2.3 Changing Directory

The command cd changes the current directory.

cd *[new-directory-path]*

where the directory path is optional. Optional arguments are shown within square brackets. This does not mean that you actually enter the brackets when you specify the option. The argument to cd must be a directory pathname and not a file pathname.

For example, the command cd .. moves you up one directory level. The command cd with no options moves you to your home directory (same as cd ~). cd ~/bin moves to the bin directory contained in your home directory. If the specified path does not exist, cd fails and the current directory is not changed.

⇒ Move to the root directory: cd /
⇒ Verify the current directory: pwd
⇒ Move back to your home directory: cd ~
⇒ Verify the current directory: pwd

5.2.4 Making Directories

The command mkdir creates a new directory in the hierarchy at the current directory location.

mkdir *directory-name-list*

⇒ Enter the command: mkdir cs246
⇒ Move to directory cs246.
⇒ Enter the command: mkdir assign1 assign2

A directory (or file) name can be composed of any character sequence and be up to 256 characters in length. However, putting special characters, like blanks, a minus sign, or pattern matching characters, into a name can cause many problems when using the name in shell commands. Two special characters that are used often in names are the underscore and dot, e.g., my_file or my.file. Try to make names meaningful so it is unnecessary to view the contents of the file to know what information it contains.

5.2.5 Listing Directory

The ls command lists the directories and files at each level in the hierarchy.

ls [-al] [*directory*]

The -a option lists *all* files, including those that begin with a dot. The -l option generates the *long* listing for each file: mode, number of links, owner, size in bytes, last modification time and file name. If no directory is given, the current directory is assumed.

⇒ Enter the command: ls
⇒ Enter the command: ls -a
⇒ Enter the command: ls -al
⇒ Enter the command: ls a*

The command lc lists the directories and files at each level in the hierarchy also; however, the information is divided by whether or not the object in the list is a directory or a file. All "hidden" files (those prefixed by "**.**") are also listed.

⇒ Enter the command: lc

5.2.6 Copying Directories

The cp command with the -r option is used to copy directories.

cp [-ir] *source-directory target-directory*

The -r option recursively copies the contents of a directory and its subdirectories to the target directory. The -i option prompts for verification if a target directory is being replaced and you must reply with "y" or "n" for each file or directory.

⇒ Move back to your home directory.
⇒ Enter the command: cp -r cs246 cs241
⇒ Verify the creation: ls cs241

5.2.7 Moving Directories

Moving a directory is accomplished by renaming the directory. The basic format of the command is:

mv *source-directory target-directory*

⇒ Rename the directory cs241 to cs240.
⇒ Verify the renaming: ls

5.2.8 Removing Directories

There are two different commands available to delete directories. The rmdir command removes only empty directories, i.e., directories with no directories or files under them. The rm command with the -r option can be used to remove non-empty directories.

rm [-ir] *directory-list*

The -r option recursively deletes the contents of a directory, its subdirectories, and the directory itself. The -i option prompts for verification for each directory/file that is being removed. UNIX does not give you a second chance to recover deleted files, so you must be very careful when using rm.

⇒ Enter the command: rm -ir cs240
⇒ Verify the removal: ls

5.3 File Management

As you perform course work, a large number of files are created, moved and deleted. Each file takes up space and each userid has a restriction on the amount of space that can be used. The command diskquota prints out how much space your account currently has allocated as well as the total amount of file space you are allowed, for example:

```
jfdoe@cayley[80]% diskquota
jfdoe (105):
Kbytes    Used   Capacity
 3600     2460   68%          hooke.math:/home
```

which indicates a current usage of 2460Kb, with an quota limit of 3600Kb. **You are warned when the average disk usage is exceeded; pay attention to these warnings!**

⇒ Enter the command: diskquota
⇒ Notice your current usage and quota limitations.

5.3.1 ⋈ Creating an Empty File

For now, an empty file is used to demonstrate file management utilities. One way to create an empty file is with the touch command:

touch *file-list*

which creates an empty file if the specified *file-name* does not exist. It updates the last modified time to the current time. An empty file is useful for testing programs. Updating the last modified time is useful for configuration management programs like make (not discussed in this document).

⇒ Move to the directory assign1
⇒ Enter the command: touch q1
⇒ Verify the file creation: ls

5.3.2 Copying Files

The copy command is used to make a copy of a file or to copy a number of files under a directory. The general format of the copy command is:

cp [-i] *source-file target-file*
cp [-i] *source-file-list directory*

The -i option prompts for verification if a target file is being replaced.

⇒ Enter the command: cp q1 q2
⇒ Copy the file q2 to the file q4.
⇒ Verify the copying: ls

5.3.3 Moving/Renaming Files

The move command is used to move a file or a number of files under a directory. The general format of the move command is:

mv [-i] *source-file target-file*
mv [-i] *source-file-list target-directory*

The -i option prompts for verification if a target file is being replaced.

⇒ Rename the file q4 to q3
⇒ Verify the renaming: ls

5.3.4 Removing Files

The remove command removes a number of files. The basic format of the remove command is:

rm [-i] *file-list*

⇒ Enter the command: rm -i q3
⇒ Verify the removal: ls

5.3.5 Viewing Files

A file's contents can be examine by using the commands more or cat. more shows the contents one screen at a time whereas cat shows the contents in one continuous stream.

⇒ Move to your home directory
⇒ Enter the command: more .cshrc
⇒ Press the space bar to move to the next screen of input.
⇒ Press "q" to quit more.
⇒ Enter the command: more .x*
⇒ Enter the command: cat .x*

5.3.6 Printing Files

The print command is used to print data on the various available printers. The general format of the print command is:

lpr [-P *printer-name*] *options file-list*

For example:

lpr -Fp .cshrc

The default undergraduate printer is a collection of self-serve Postscript laser printers.

In general, all printing must be paid for. Printer pages can be obtained by purchasing a CAS account from IST at the Computer Help and Information Place in MC1052. Graduate students who are registered for a thesis or an essay receive a quota dependent on the requirements. The lpquota command reveals your current printer quota.

After you submit something for printing, go to MC3016 where you'll find the output side of several printers. Pick an available output slot, pass your student card through the small card reader to the left of the slot, and then wait for your output. There's a monitor on the opposite wall that displays the state of each individual printer queue. There's also a terminal that can be used to remove a submission from the print queue, query your quota, and several other things.

To determine if your output has printed, it is possible to view a printer's job queue. The general format of the printer query command is:

lpq -P *printer-name*

which provides a list of all of the jobs waiting to print on the specified printer. Each job is assigned a unique number. If you do not specify a printer name, the queue for your default printer (usually ljp_3016) is displayed.

⇒ Enter the command: lpq

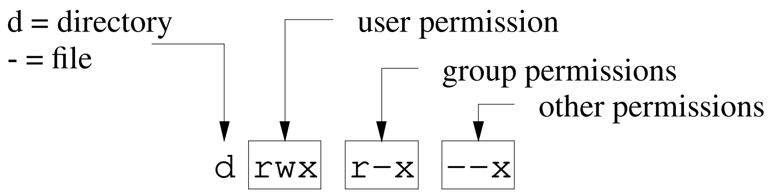

Figure 8: File Permissions

To remove a job from a printer's job queue, issue the command:

lprm -P *printer-name job-number*

You may have to issue this command on the same machine that you were on when you sent the job. You can also remove jobs from the queue by using the terminal in the printing room (MC3016).

If no arguments are specified, lprm deletes the currently active job if it is owned by the user who invoked lprm. For more information on these commands, read the man pages.

5.3.7 File Permissions

The UNIX file structure supports 3 levels of security on each file or directory: user, group and other, where "user" is the userid of the creator of the file, "group" is an arbitrary name associated with a number of userids (not normally used), and "other" is any other userid. At each level of security, a directory or file can have several permissions: read, write, and execute (or search). Readable and writable have obvious meaning. Executable for directories means that the directory can be searched by certain system operations but not read in general; executable for files means that the file can be executed as a command.

⇒ Enter the command: ls -l

which lists all files and directories in the current directory and the permissions associated with each file. The first column is the permission information for each file and Figure 8 shows the organization of the information. For example, drwxr-x---, indicates a directory in which the user has read, write and execute permissions, the group has only read and execute permissions and others have no permissions at all. In general, students should not allow other users to read or write their files.

The permissions on a file or directory are set when the file or directory is created.[3] The default permissions on a file are rw-r-----, which means owner has read/write permission, while group and other have only read permission. The default permissions on a directory are rwxr-x--x, which means owner has read/write/execute, while group and other have only read and execute. Normally, a home directory is set up with permissions drwx--x--x, which means that others cannot look at your account to find out what files you have, but they can read your files if you give out the file names.

Permissions can be added or removed from any of the 3 security levels by using the chmod command. The basic format of the chmod command is:

chmod *mode-list file-list*

mode-list has the form *security-level operator permission*, where the security levels are denoted by the characters u for you the user, g for group and o for other users, the operator is + to add the permission and - to subtract the permission, and the permissions are denoted by the characters r for readable, w for writable and x for executable. The elements of the *mode-list* are separated by commas. For example, to remove read and write permissions from security levels group and other for file xyz can be specified in the following long and short forms:

```
chmod g-r,o-r,g-w,o-w xyz
chmod go-rw xyz
```

5.3.8 ⋈ Restoring Files

If you accidentally delete a file, there are two options for restoring an older version of the file. First, under the directory .snapshot in your home directory, are short-term copies for changed files. See man mfcf-undergrad-home for a detailed

[3] The "umask" setting in the .cshrc file controls this. See man csh for more information.

explanation of this backup service.

Backups of older files are saved on tape. If an old version of a file is not under .snapshot, it may be possible to restore it from a backup tape using the restore command. Files from previous terms are generally not restored unless it can be demonstrated that they are needed for a current course. Restores from tape can require extensive Operator involvement, so they should not be considered as an archive for personal files. The MFCF Operators will notify you via e-mail if there is a problem with a restore. If a restore from tape is completed successfully, you will receive a canned message from the restore software.

5.4 History

For every command issued in the C shell, the number in the prompt goes up by one. This occurs because the C shell keeps a list of the most recent set of commands issued from the terminal, say the last 20. You have the facility to edit and re-issue any of the commands in the history list. This makes it easy for you to repeat all or parts of a command without a lot of typing.

⇒ Enter the command: history

This displays the current history list. To re-run command N in the list, type "!N"; "!!" re-runs the last command. To re-run the last command starting with the string "xyz", use the command "!xyz".

⇒ Re-run your last command.
⇒ Try re-running your last more command using both the command number and the string search capabilities.

5.5 Aliasing Commands

It is possible to create your own set of commands. These commands may be nicknames for commands you use frequently, or a variation of a command you use often. The alternate name is called an alias and is introduced with the alias command, as in:

alias *new-name command [command-option-list]*

⇒ Enter the command

```
% alias d date
```

⇒ Now issue the command d. As you can see, it is a short-hand way to invoke the date command.

Aliases are useful for setting command options for particular commands, as in:

```
% alias cp cp -i
% alias mv mv -i
% alias rm rm -i
```

which always uses the -i option on the commands cp, mv and rm commands, respectively, so you are prompted whenever an existing file is about to be destroyed. To specify a sequence of commands to be executed, place the list of commands (separated by semi-colons) between quotation marks, as in:

```
% alias off "clear; logout"
```

which clears the screen before logging off. An alias can always be overridden by putting the command name in quotes, as in:

```
% "rm" -r xyz
```

which does not add the -i option, and hence, does not prompt for each file that is deleted.

If you enter the alias command on your command line, it takes effect for the current login session. The next time you login, the alias is gone. There are two options for making aliases permanent for each login session.

1. Insert the alias commands in your .cshrc file. If you look through this file, there are instructions about where to place the alias commands.

2. Place the list of alias commands in a file called .aliases in your home directory. The .cshrc file processes this file by doing source .aliases.

5.6 ⋈ Input/Output Re-direction

It is often very convenient to redirect the input or output of UNIX commands to/from sources other than the keyboard and window. The shell provides the redirection operators < for redirecting input and > for redirecting output.

- ⇒ Enter the command: ls -l > xyz
- ⇒ Look at file xyz using the more command.
- ⇒ Enter the command: cat xyz xyz xyz > yyy
- ⇒ Look at file yyy using the more command.
- ⇒ Remove files xyz and yyy

Normally, error messages are not redirected because of their importance. To redirect *all* output, use the redirection operator >&.

The pipe operator | is used to pass the output of one command to the next command as input without having to create an intermediate file.

- ⇒ Enter the command: ls -al | more
- ⇒ Enter the command: rwho | more

The output from the ls command is "piped" into the more command as input.

5.7 ⋈ Jobs

5.7.1 Running Jobs

Each command executed by the shell constitutes a "job". Jobs can be run either in the foreground or in the background. If a job is run in the foreground, you must wait until it finishes execution before you can issue another command unless you either suspend or terminate it. If a job is run in the background, you can keep on executing commands in the shell. To run a job in the background, append an & at the *end* of the command.

The following is an alternative way to start an xterm window.

- ⇒ Move the cursor to the TopXterm window.
- ⇒ Enter the command "xterm &". A window outline appears, attached to the cursor. Also, a message like "[1] 10012" appears; make a copy of this information as it is used later. The number in brackets is the job number and the second number is the UNIX process identifier; both values are discussed shortly.
- ⇒ Move the window outline to any location on the screen.
- ⇒ Press a mouse button to attach it to this location.

When running jobs in the background, there can be problems with output and input. The output from background jobs is sent to the shell's window, which usually interferes with other work. Redirecting the output from the background command into a file is the best way to handle this problem, as in:

```
cc myprogram.c -o myprogram >& comperrs &
```

This compiles a C program in the background and redirects any compilation error messages to file comperrs.

Background jobs may also need input, which may be supplied by a file on the command line or using input redirection, as in:

```
myprogram < input >& output &
```

The command myprogram normally gets its input from the keyboard, but now it gets input from file input and writes its output into file output (including any error messages).

5.7.2 Listing Jobs

The command jobs lists all of the jobs currently running in the background for the current terminal session. Each job in the list is numbered.

- ⇒ Move the cursor to the TopXterm window.
- ⇒ Enter the command jobs

If a job has not yet finished, it is listed along with its current status.

A list of (almost) all jobs currently owned by you can be generated by the command ps.

⇒ Enter the command ps -ux to see all of the jobs you are running. Note the process identifier (also known as a PID) assigned by UNIX to each job.

Note that the process identifier for the xterm is not in the list because you do not own that process; it is owned by the system, but you can still destroy it.

5.7.3 Stopping Jobs

Foreground Jobs There are two ways to stop jobs running in the foreground. Using ^Z *suspends* the job. To type ^Z, hold down the key labeled $< Control >$ (or $< Ctrl >$) and press the z key. The $< Ctrl >$ key is usually found somewhere to the left of the "a" key. The $< Ctrl >$ key is like a shift key: it does nothing itself; instead it modifies the meaning of other keys that are pressed at the same time. The job is not terminated, only stopped. A suspended job still uses resources so do not use this mechanism unless you intend to restart the job. If you have suspended a job, you can use the jobs command to find its job number.

⇒ Move the cursor to the TopXterm window.
⇒ Enter the command xterm and create a new xterm window.
⇒ Move the cursor back to the TopXterm window.
⇒ Use the key sequence ^Z to suspend the job.
⇒ Move the cursor back to the new xterm window and enter $< Return >$. Nothing happens because the xterm is stopped.
⇒ Use the command jobs to find the job number of the stopped xterm.

^C issues an interrupt signal and *terminates* a job. If you need to kill a job running in the foreground, use ^C.

⇒ Move the cursor to the TopXterm window.
⇒ Enter the command xterm and create a new xterm window
⇒ Move the cursor back to the TopXterm window.
⇒ Use the key sequence ^C to kill the job.
⇒ Use the command jobs to verify that you still have a stopped job.

Background Jobs The command kill %1 terminates the background job numbered 1.

```
kill %1
```

⇒ Use the kill command to kill the stopped xterm.
⇒ Use the command jobs to verify that the job is terminated.

A job can also be killed by specifying its PID, as in:

```
kill 10012
```

⇒ Use the kill command to terminate the xterm created at the start of this section.

When logging off, if you receive a message that there are stopped jobs and you are still logged on, use the jobs command to find the background jobs and kill them.

6 ⋈ Electronic Communication

Computers are connected into large networks that allow different forms of communication between them.

mail is point to point communication between specific users.

news is a large body of information, organized into a hierarchy based on particular topics, that is selectively read by particular users.

mail is like writing a letter that is delivered faster than the post office. news is like a public bulletin board, organized by topics. In all cases, it is the responsibility of a user not to misuse these facilities or ALL users suffer.

6.1 Rules for Sending Messages (Netiquette)

Electronic messages lack voice inflections and body language. Because of this, it is easy for messages to be misinterpreted. Humour, especially sarcasm, is very difficult to convey and can be misunderstood. To avoid misunderstandings, symbols that represent facial expressions (turned 90°) are used to indicate inflection. Some common examples of ways to show feelings are:

:-) humour

:-(sadness or disappointment

;-) sarcasm

8-0 amazement or excitement

***'s** can be used for emphasis, as in: a *real* computer has an address space greater than 64K ;-)

capitals can also be used for emphasis, as in: a *REAL* computer has an address space greater than 64K ;-)

Here are some general guidelines for when you send messages:

1. Your message reflects upon you—be proud of it.
2. Think about what you write (do not make a fool of yourself).
3. Do not flame i.e. post an offensive, abusive, or inflammatory message.
4. *Be civil! Never use offensive language.*
5. If you are angry about a message, always wait 12 hours before replying.
6. Be brief—a message will not be read if it is long.
7. Do not waste people's time (avoid repetition and verbosity. REPLY BY MAIL wherever possible).
8. Use the appropriate news group.
9. Do not become a mail or news addict.
10. Do not flame.
11. *Never* flame about spelling or grammar.
12. Do not overdo signatures.
13. *Do not flame!*

6.2 ⋈ Electronic Mail

It is convenient to be able to send mail to someone electronically. One example would be requesting help or an appointment from the lab coordinator. You will be introduced to the mail utility pine. pine can be used on all of the MFCF undergrad machines. Once you have become familiar with pine, you may wish to experiment with other mail utilities such as elm.

All mail addressed to any one of the Math undergrad collection of machines actually goes to undergrad.math.

6.2.1 Starting pine

⇒ Start up pine by issuing the command pine

You will be presented with the main menu. The commands available to you are displayed in a block at the bottom of the screen. To enter a specific command, press the key indicated. For example, pressing "c" allows you to compose a mail message.

If this is the first time you have ever used pine, you will be shown a message that pine is creating a directory and folders for your mail. All mail that you send will be stored in the file/folder "sent-mail". Every time that the month changes, pine will ask you on start-up if you wish to move your previous month's "sent-mail" to a dated file. You can choose to do so or not. However, you should periodically delete "sent-mail" since it will otherwise take up a great deal of space. For the same reason, you should delete all mail messages you no longer need. If you need to keep a copy of a mail message, you should save it to a file and print it.

6.2.2 Sending Mail in Pine

⇒ Press the "c" key.

You will be presented with a blank header area and a blank message body. The cursor is automatically placed in the first field of the message header.[4] Pressing the $< Return >$ key or using the "down" cursor key will allow you to move to the next field.

The To: field indicates to whom the message is to be sent. As many userids as you wish, separated by commas, may be entered there.

⇒ For now, enter your own userid as the recipient.

The Cc: field indicates to whom a carbon-copy of the message is to be sent.

⇒ No information is supplied here, so move to the next field.

The Attchmnt: field contains the list of files which will be sent along with the message.

⇒ Move onto the next field.

The Subject: field contains a text string indicating the subject of the message. NOTE: the subject line is not required but is *strongly* recommended since it allows the person receiving mail to prioritize the reading of the mail.

⇒ Enter a suitable message and press $< Return >$.
⇒ Press $< Return >$ again to move into the body of the mail message. (Note how the commands available to you have changed.)
⇒ Type in your message.
⇒ When you are done, exit the mail message by pressing ^X (this is equivalent to the ^X shown in the command area).
⇒ You will be prompted to check if you really wish to send the message. Enter "y" to indicate "yes".
⇒ You will be placed back to the point in the mailer where you issued the compose command.

6.2.3 Viewing Mail in Pine

In order to view mail, use the Mail Index command.

⇒ Go to the Mail Index.

You will be presented with a list of all the mail you have and its current status.[5] Each piece of mail is given a number. The message whose number is currently highlighted is the one your commands will affect. You can use Prev Msg, Next Msg or the cursor keys to move around in the list.

As new mail arrives, your index will be updated.

⇒ When your mail message arrives, make it the current piece of mail.
⇒ Press the "v" key to view the mail.

6.2.4 Answering Mail in Pine

When replying to a mail message, it is highly recommended that you include the message you are replying to. This makes your response more meaningful to the recipient. Of course, you should edit out the unnecessary parts.

⇒ Return to the Mail Index.

---

[4] Use the "help" command when the cursor is located on the field you want information about to receive detailed information on the purpose of the field

[5] N indicates un-read mail, D indicates mail to be deleted when pine is terminated, and S indicates saved mail.

6.2.5 Deleting Mail in Pine

Mail is not actually deleted until you leave pine. Until then, it is only *marked for deletion*. This mark can be removed by using the undelete command, “u”. This command can be found under the OTHER CMDS command.

⇒ Delete the high-lighted message by pressing the ”d” key.
⇒ Press “o” to bring up the OTHER CMDS menu.
⇒ Undelete your current mail message.
⇒ Press “o” to toggle back to your other list of commands.
⇒ Delete the message again.

6.2.6 Quitting pine

⇒ Press ”q” to quit the mail-handler. You will be prompted for two things.
⇒ Enter “y” or press $< Return >$ to answer yes to the question “Really quit pine? (y/n) [y]”.
⇒ Enter “y” or press $< Return >$ to answer yes to the question “Expunge the # deleted messages from “inbox”? (y/n) [y]”.

6.3 ⋈ Electronic News

There are a large number of electronic bulletin boards available for your use. The ones that will be most important to you are the ones dealing with your course and new users. NOTE: it is your responsibility to read the system and course news groups at least once a day until the end of the term.

6.3.1 Course Newsgroups

Computer science course newsgroups are named in the following way: uw.cs.csNNN where NNN consists of the course number. For example, CS 246 course has the newsgroup uw.cs.cs246. The course newsgroup is for:

- asking questions about course material
- asking questions about assignment guidelines
- asking questions about assignments
- discussing assignment answers (without giving answers away)

A course newsgroup is *not* for:

- the discussion of other Computer Science topics, e.g. programming languages (post to uw.lang)
- to express opinions about the quality of course material (mail directly to the instructor or the course supervisor)
- to express an opinion about other students (talk directly to the individual)
- to express an opinion about the instructor (talk directly to the instructor or the course supervisor)
- testing your ability to post news items (use uw.test instead)

Because of abuses of course newsgroups in the past, failure to follow the above restrictions on posting news items can result in the loss of computing privileges.

6.3.2 Reading News

The News software program is invoked by the command trn. NOTE: At any point in *trn*, you can press ”h” to determine which commands are available to you at this time.

When you first invoke *trn*, you should already have been added to the subscription list for your course and the newsgroup for new users, news.announce.newusers. If not, you should subscribe to them. You *must* remain subscribed to course newsgroups for the duration of the term. If you set up your environment so that the system news window does not come up automatically, you should be sure to subscribe to uw.mfcf.system, so that you are always aware of system information that might affect you.

6.3.3 Posting News

In order to post to the newsgroup, there are two commands available to you, each of which performs essentially the same function. The commands to post are Pnews and postnews. If you want to practice posting an article, please post only to the newsgroup uw.test. This newsgroup is specifically for this purpose. It is strongly advised that you follow the rules of “netiquette” when posting.

7 Finding Information

7.1 About UNIX Commands

To obtain help in UNIX, use the man command which retrieves manual pages. Since the documentation is on-line, it is often more up-to-date than printed manuals.

⇒ Enter the command “man who”.

NOTE: The information appears a screen at a time and can be advanced by pressing the space bar and stopped by typing “q”.

⇒ Enter the command “man man”.

The manual information on the who command and the man command are displayed, respectively. The manual entry page called software contains an extensive list of software that is available on different UNIX systems on campus.

The manual documentation (known as the “man pages”) is broken into sections numbered 1 through 8, with additional sections labeled with letters.

1 – UNIX commands
2 – system calls and errors
3 – functions in UNIX support library (for programs written in C or FORTRAN)
4 – special files and hardware support
5 – file formats
6 – games
7 – miscellaneous useful information
8 – system maintenance and operation
i – information

In some cases, it is necessary to know about this manual structure because there are commands and functions with the same name in different sections. For example, there is a sleep command in section 1, and a sleep function in section 3. To get the correct manual entry, put the section number as a command option.

⇒ Enter the command: man 1 sleep
⇒ Enter the command: man 3 sleep

There is no easy way to discover the section containing a particular command or function. One technique is to look in the manual entry under the heading SEE ALSO, which contains references to other manual pages with related information. For example, the sleep command provides sleep(3) under the heading SEE ALSO, which indicates there is another command or function called sleep in section 3 of the manual.

If you do not know how a command is spelled, it is possible to search for a general keyword that describes the action it performs. The command apropos is the same as man -k.

⇒ Enter the command: man -k copy
⇒ Enter the command: apropos edit

In both cases, many commands are listed and you have to find the one that performs the desired task.

If you need information about software on the undergraduate machines, see the Consultants in MC3008. Note: the Consultants are there to help you understand the hardware and the software, NOT to help on assignments or research projects.

7.2 About People

The personal information that was entered at the first login session is used by other users to locate you (like a telephone book). This personal information is called “finger” information (let your fingers do the walking!).

If you do not know a person’s userid, the command uwdir can be used to obtain a list of possible people on different computers. For example, to find the userid of the CS 200 Level Lab Coordinator, who has the first name “Caroline”:

⇒ Enter the command uwdir Caroline

The response is a list of all people who have the name “Caroline” as part of the information listed with the Registrar, including their department and email address. Staff and Faculty are also included in the list, although information about them comes from the Telephone Services database.

Now that you have a userid and computer, you can use the finger command to find out more about the person by printing their finger information.

⇒ Enter the command finger ctkierst@undergrad.math.uwaterloo.ca

From this information, it is possible to see if the person is currently logged on, if he or she has received (and read) any e-mail recently, his or her office number (if applicable) and possibly a phone number. Sometimes other information is also included. For example, issuing the command finger operator shows the operator’s work schedule, and when output is filed. Finger may be disabled occasionally if the system load climbs too high. Also note that finger may not always be accurate. For example, if someone is only logged on for a few minutes, their finger information may not be updated. If the person did not log out correctly during their last session, this could also cause finger to report incorrect information (e.g., by reporting that they are still logged on when they are not). Also, finger may be disabled on some other systems on campus.

You can change your finger information at any time using the command chfn. The command prompts for information exactly like the first login session.

NOTE: Registrar information can be marked as private by an individual, in which case the uwinfo is flagged as private and will not be used for mail forwarding. Therefore, mail to userid@uwaterloo.ca will not work if userid is flagged as private in the Registrar’s database. In such a case, the full email address, e.g., userid@undergrad.math.uwaterloo.ca, must be used. Also, see Section 3.1 for information on potential conflicts between the uwuserid given by the Registrar and the userid given by MFCF.

7.3 About Courses

Many courses have files listing frequently asked questions (FAQ) about the course; ask your instructor if there is an FAQ file for the course. It is often fastest to first scan through the FAQ file because it contains answers to a large number of course specific questions.

7.4 About Signing on from Home

To access the MFCF systems from home requires a terminal with a modem, as well as an account on a campus terminal server. You can obtain information about terminal settings and phone numbers providing access to the MFCF computers by reading man communications or asking the MFCF Consultant. An account on a campus terminal server can be purchased from the IST Customer Support Centre in MC1052.

7.5 About Problems

If you find a problem with any of the standard software or hardware, you may report it with the gripe command (see man gripe for details), or use the web page version of the gripe command:

http:*//www.math.uwaterloo.ca/mfcf/problem-report.html*

Please be accurate and concise in the description of the problem!

Before posting a problem it is often worth while to check the MFCF consultant’s FAQ (Frequently Asked Questions) web page, which is a good place to look for information on many subjects. It is located at:

http:*//www.math.uwaterloo.ca/mfcf/faq/index.html*

Please check here to see if your problem is addressed, before you report it elsewhere.

Finally, problems relating to operations, e.g., printing, terminal rooms (e.g., room comfort) may be reported to operator@math.

7.6 Netscape, Mosaic

A World-Wide-Web browser is a distributed document delivery service. It allows a user to access various types of data residing on multiple hosts in a seamless fashion. This capability is accomplished by presenting the user with a hierarchical arrangement of documents and by using a client-server communication model. MFCF provides two browsers: netscape and xmosaic. See the man page on netscape and xmosaic for more information. Note, both browsers are resource intensive so they may only be run on specific machines in the Unix environment. In particular, browsers are set up to run remotely on a group of machines called the zenoes, which are located in the MFCF machine room. When you invoke a browser from a Unix terminal, one of these systems should automatically be selected for you, and the browser started there.

Index