CEDA/Demography Lab
Frequently Asked Questions

Carl Mason
carlm@demog.berkeley.edu

Fall 2012 version 3.14

Contents

Chapter 1

The Gnome Desktop

1.1 How can I change the focus behavior

focus behavior refers to the condition that determines which window recieves input from the keyboard. The standard rule is "click to focus" which means that the window on which you last *clicked* holds the focus until you click somewhere else.

It will save litterally minutes (over a your life time) if you select instead what is called *sloppy* of *focus follows mouse* under which behavior, the window in which the mouse currently resides gets the focus.

This allows you type into partially obscured windows and generally works better with X-style select and past functions.

Under Gnome, the focus behavior is set via the `[System]`→`[Preferences]`→`[Windows]` menu.

1.2 How can I map a keystroke to a command

`[System]`→`[Preferences]`→`[Keyboard shortcuts]`

is for doing this. Click on the function of interest and then, when prompted, type a keyboard combination to asign to that function.

1.3 That thing that used to be on my task bar thingy disappeared

The "aplets" that live on your various "panels" can be easily created and accidently destroyed. The two most useful applets are the `Workspace Switcher` applet which shows your "virtual desktops" and allows you to switch between them; and the `window list` applet which shows your running applications. To (re)create these or other panel applets, LEFT BUTTON on the panel to which you wish to add an applet. Then scroll to select the applet of interest. `Window`

`List` and `Workplace Switcher` are near the bottom. Click the [`Add`] to add the applet. There are lots and lots of gizmos and gimcracks *besides* `Workspace Switcher` and `window list` that you can add this way.

1.4 How can I add buttons to launch applications

If the application in question is in a menu, you can simply drag and drop the thing from the menu to a panel. Just select the item in the menu; click and hold [RIGHT BUTTON] ; then drag it to the panel wherein it belongs.

If the item is not in a menu – or if you just like doing things the hard way, you can add a "launcher" by clicking [LEFT BUTTON] over the panel to which you wish to add the launcher and selecting `[Panel→Add to panel→launcher]`.

The resulting dialog box allows you to specify the command that launches the program and whether or not it should be run in a terminal window.

1.5 How can I move windows when the title bar is obscured

If the title bar is **not** obscured then just click and hold the [RIGHT BUTTON] over the title bar. If the title bar **is** obscured then of course you cannot do the this. In this case your best option is to make sure that the window you wish to move has the focus, then hit [ALT] + [F7] . Click the [LEFT BUTTON] to release the window in a new location.

You can also move a window by clicking and dragging the [SHIFT] + [LEFT BUTTON] on its representation in the virtual desktop and dragging it.

1.6 How can I create a panel at the top of the screen

The desktop in the current release of Fedora has a panel at the top of the screen. If you need to recreate it click the [RIGHT BUTTON] on an existing panel, then select `[New Panel→menu panel]`. The new panel might appear along a side instead of the top, if it does, then click [LEFT BUTTON] and select `[properties]` to change its "orientation" to something to your liking.

Chapter 2

Printing

2.1 How can I print "n-up"

Applications such as acroread or openoffice allow you to print "n-up" within their native print menus. If you need to print "n-up" outside of one of those fancy modern programs, you can use either `xpp` (See 2.2, or `a2ps`. Typically one prints two (portrait) pages on single (landscape) page, however more pages per sheet are possible – for those who believe that squinting can save the planet.

If the file is ASCII, PostScript, or pdf, you can simply type

```
@:> a2ps -2 -g -P printer filename
```

The `-g` flag causes a fancy border to be drawn around each page the `-2` flag indicates that 2 pages per sheet is sufficient

For those who cannot be bothered with typing commands, there is also a GUI alternative, see 2.2.

2.2 How can print in fancy ways without remembering (knowing) complicated command line options

`xpp` is your best option. `xpp` offers you menus and buttons that allow you to change the printer features. For example you can print n-up, or choose toner saving modes and perhaps do other morally virtuous printer related things.

To launch type:

```
@:> xpp filename
```

2.3 What should I do if the printer runs out of paper

Add paper – there boxes of reams in the Xerox room. Why not bring as many reams as you can carry without endangering your health and save other's the trouble.

2.4 What should I do if the printer runs out of toner

Install a new toner cartridge – or if you're too chicken at least tell a grown up that something is wrong. Toner cartridges live in the Xerox room. They should be labeled as to which printers the will work in – if not check with Monique. Either way send Monique email informing her that you have used a toner cartridge so that another can be ordered.

2.5 Printing from macs and Windows

Instructions for setting up your computer to print to demography printers when connected via AirBears is at `http://lab.demog.berkeley.edu` Under `[Documentation]`→`[Printing from Portables]`.

Chapter 3

Tricks in R

3.1 How can I import data from other programs into R

Several data file formats can be imported directly in to R via the `foreign` library. If the sort of data you want to import is not on the list below, then there is still a good chance that you will be able to use `stat-transfer` to convert it to one that is recognized by R.

Stat-transfer can convert just about any kind of file into a SAS transport format file or an ASCII delimited file, either of which can be read by R. Some information – such as SAS informats and labels – will be lost in this process, as there are no equivalent structures in R.

To read ASCII delimited files, the easiest method is `read.table()`

To read data from a tab delimitted ASCII file called 'truth.csv' with the first row of the file containing variable names the following should suffice:

```
> truth ←read.table(file='truth.csv',      ;
  sep="",header=T,
  quote="",comment.char="",as.is=T)
```

Note the following gotchas with `read.table()`:

- **sep=""** tells R to trat TAB as a delimitter. Other choices are of course possible.
- **quote=""** tells R not to treat ' or " as quotations (quotations in a data set would only be necessary if some of your data elements contain the delimtter character). It is much more likely that you have a data element like `O'Conner` in your dataset.
- **comment.char=""** tells R that there are no comments in this dataset – Just in case one of your data elements contains a "#".

- **as.is=T** tells R not to convert all character strings into factors – which is the default behavior. You might want R to do this conversion, but if you are not expecting it, it is confusing.

To read and write non-ASCII types of files, use the `foreign` library. To access it you must enter:

```
> library(foreign)
```

To see how it works type:

```
> help(package=foreign)
```

Currently, the the `foreign` library supports the following file types

```
S3 read functions       Read an S3 Binary File
read.dta                Read Stata binary files
read.epiinfo            Read Epi Info data files
read.mtp                Read a Minitab Portable Worksheet
read.spss               Read an SPSS data file
read.ssd                obtain a data frame from a SAS permanent
                        dataset, via read.xport
read.xport              Read a SAS XPORT format library
write.dta               Write files in Stata binary format
```

All of these functions expect to be passed a `file` parameter as in:

```
> importantdata ←read.dta(file='/data/commons/filename.dta')
```

> Be patient, `foreign` library functions can take a long time to run. A moderate 17MB stata file can take in excess of 20 minutes to be converted and read in. Unless you get an error message, get coffee

Like Stata, R insists on holding all objects in core memory, this makes it a poor choice for data sets that are larger than 500 MB. Since tapinos has 10GB of RAM, large datasets can be processed in R and Stata, but you will surely want to do such things in batch mode.

3.1.1 How can I import a SAS data file into R

First convert your sas7bdat (or whatever type of SAS file) to "SAS transport format". This can be done in SAS **or** with `stat-transfer`. See Section 4.4.2 for more details, but roughly speaking:

@:> `st`

> `cp filename.sas7bdat sasx filename.xpt`

Import your SAS *transport-format* file into R:

> `library(foreign)`

> `objectname` ←`read.xport(file='/path/to/filename.xpt')`

`objectname` will be a dataframe.

3.1.2 How can I import Stata,SPSS,excel or some other kind of file

Some files can be imported directly using methods contained in the `foreign` library, these include SPSS, Stata, Epiinfo and minitab (see the help pages on package foreign for details). For other types, you may need to convert the file into a SAS transport or other convenient type. `Stat-transfer` (See Section 4.4.2) is great for this. Once your file is in say SAS transport format (See Section 3.1.1 for instructions on importing it).

3.2 Graphics tricks

3.2.1 How can I draw a population pyramid

Below is a chunk of R code that will draw a pretty nice population pyramid, please adapt it and improve it.

```
#########################################################################
## Sat Mar  9 20:21:19 PST 2002
## Carlm's pretty good population pyramid function.
#########################################################################


poppyr<-function(age.male,age.female,agecats=c(0,1,4,seq(5,100,5))){
  ##age.male is a vector of ages of males, age.females is a vector of the
  ## mean batting averages of the New York Yankees, agecats is a
  ## vector of break points of ages -- defaults to a pretty standard
  ## one
  ##
```

```
  ## Will draw a pretty good but not quite perfect population pyramid

  male.ac<-table(cut(age.male,breaks=agecats))
  female.ac<-table(cut(age.female,breaks=agecats))


  ## 5/13/04 Megan Heller reports that some alternative 'figs'
  ## values can improve the appearance of the population pyramid
  ## by eliminating the space between the sexes. Here are
  ## Megan's suggested values:
  #  ##this one looks best in terminal
  #  split.screen(figs=rbind(c(0,.58,0,1),c(.43,1,0,1)))
 ## this one looks best in png (png files are the easiest to
  # import into MSWord -- not that you'd ever want to do that.
  # split.screen(figs=rbind(c(0,.58,0,1),c(.38,1,0,1)))


  split.screen(figs=rbind(c(0,.57,0,1),c(.45,1,0,1)))
  screen(1)


  barplot(female.ac,horiz=T,names=paste(agecats[-length(agecats)]),
          xlim=c(max(female.ac)*1.1,0),col="#FF9900")
  title("Female")

  screen(2)
  barplot(male.ac,horiz=T,axisnames=F,
          xlim=c(0,max(male.ac)*1.1),col="#0000FF")
  title("Male")

  close.screen(all=T)
}



age.male<-rnorm(10000)*50
age.female<-rnorm(10000)*50
poppyr(age.male,age.female)
mtext(side=3,line=3,text="Population Pyramid",cex=1.5)
```

3.2.2 How can I create a plot with two y-axes

You might want to do this if you have two vectors of data which have very different scales but which are both thought to be related to a third vector. In

other words, you have two y-vectors which you would like to plot against a single x-vector.
The trick is to use the `par(new=T)` command, which tells R **not** to reinitialize the plotting device with subsequent high level commands (such as `plot()` which would ordinarily cause this to happen. Instead, R will superimpose subsequent plots on top of what is already displayed on the current device. You will most likely also want to use `axes=F` argument to **one** of the plot() commands. This argument supresses the drawing of axes.

```
################################################
## this will create a scatter plot displaying
## two sets of points, one corresponding to the
## y axis on the left and one to the y axis on the
## right
################################################

dev.off()   ## start with a new graphics device
# X11() or postscript()
plot(x<-rnorm(100),y<-rnorm(100))
z<-rnorm(100)*250
par(new=T)   ## Tell R not to reinitialize graphic device
             ## for subsequent plots
plot(x,z,col='blue',axes=F)
axis(side=4,col.axis='blue')

par(new=F)
```

NOTE: in the example, the two y-vectors are plotted against **the same x-vector** if the x-vectors were merely similar it would be necessary to use the `xlim=` argument to both `plot()` commands in order to prevent very misleading results. The second `plot()` command does not know anything about the first. It plots using the same graphics device but it does not care where the first plot thought the x-values were. To verify that the x-axes are identical, you can show the second x-axis by typing: `axis(side=1,line=1)`

3.2.3 How can I make the text for the y-axis come out horizontal rather than parallel to the axis

(Thanks to tmiller) There should be an easy way. Meanwhile, here is a hard way that works:

```
# Set margins on yaxis side large enough (10.1?) to accomodate ‘‘foo’’
par(mar=c(5.1,10.1,4.1,2.1))
par(las=1) # print axis numbers horizontally
# Don’t print y-axis text label
plot (x,y, ylab=" )
# Print y-axis text label horizontally
mtext(‘‘foo’’,side=2,line=4)
```

3.3 LaTeX and R

3.3.1 How can I make a latex table from a matrix in R

`xtable` is a library for writing LaTeX or HTML files from R objects.
Load the library:

```
> library(xtable)
```

Create an object of type “xtable”.

```
> object.4.export ←xtable(object-name,
  label=’will appear as label in output
  object’, caption=’will appear as
  caption in output’, digits=3)
```

Print the object to a file:

```
> print.table(x=object.4.export,
  type=’latex’, file=’table42.tex’)
```

Inside your latex file add:

```
\input{table42}
```

The `xtable()` function has special tricks for several objects that it knows about. For example:

```
> lmout ←lm( NetHumanWorth   hatsize +
  shoesize, data=census)
```

```
> cooltab ←xtable(lmout,
  caption=’Regression Results’,
  label=’tab:truth’)
```

To see a list of objects that `xtable()` knows about type:

```
> methods(xtable)
```

3.3.2 How can I make an HTML table from a matrix in R

See Section 3.3.1 for a description of how to make latex tables from R objects. the `xtable()` function can be instructed to create HTML instead of latex. Also see Section 7.0.4 for a description of how to translate latex into very nice HTML.

Chapter 4

File Conversion

4.1 What is the difference between "raster" and "vector" image files

Files such as jpeg, gif, bmp, pnm, png, tiff, ppm and lots of others. What distinguishes these file formats from say PostScript, is that the information they contain is where to put a bunch of dots. PostScript, WMF, CGM and others are "Vector" formats. They contain information on lines and shapes which the computer then converts into points when rendering it on the screen.

4.2 Vector image files: Postscript and pdf

4.2.1 How can I convert Postscript to pdf

`ps2pdf13` runs on the linux machines. There are 3 versions of `ps2pdf`. Ps2pdf14 produces output compatible with Acrobat 4 and later. If you need something compatible with more ancient versions check the man page for ps2pdf.

```
@:> ps2pdf14 filename.ps
```

Each of the above commands produces a file called `filename.pdf`. NOTE: if your postscript file is the output of `dvips` it will look **much** better if `dvips` was given the `-Pcms` option. BUT more to the point, you can convert `.dvi` to `.pdf` directly. See(7.0.3).

4.2.2 How can I convert several postscript files into one .pdf

On a linux machine, this should work:

```
@:> cat first.ps second.ps third.ps | ps2pdf14 - all.pdf
```

4.2.3 How can I convert Postscript files to graphic: .gif,.jpg...

There are at least three good ways of doing this:

1. Use `ghostscript` to convert to .png then use `xv` to convert to your chosen format.

```
@:> gs -sDEVICE=ppm -sOutputFile=filename.ppm filenmame.ps(or pdf)
@:> xv filename.ppm
```

2. Use ImageMagic. To read in, edit if you like, and save as some other format.

```
@:> display filename.ps
```

3. Use `gimp`. Gimp is a huge and complex graphics manipulation package that runs on linux machines. It has the capacity to edit just about any kind of image file – including postscript files. To convert using gimp just load the file and save it as something else.

```
@:> gimp filename.ps
```

Both gimp and ImageMagic can be used to edit and convert among graphic types also.

4.2.4 How can I convert Postscript files to eps with tiff preview

Generally this is useful for including a postscript file in MSWord.
Use `epstool`:

```
@:> epstool -t4 -ofilename.eps filename.ps
```

Then in Word: `[Insert]`→`[Picture]`→`[From File]`. The low resolution .tiff preview will appear when editing in Word, the high resolution postscript file will appear in the printout.

4.2.5 How can I convert LaTeX (tex or dvi) files to Postscript

Use `dvips` after running `latex` of course, on the .dvi file:

```
@:> dvips -Pcms -ofilename.ps filename.dvi
```

4.2.6 How can I convert LaTeX to pdf

- First convert to postscript (See Section 4.2.5) then to pdf (See Section 4.2.1.
- Or try pdflatex: **@:>** `pdflatex filename`

  where filename is the `.tex` version of the file.

4.3 Raster (aka "graphics") files

4.3.1 How can I convert one sort of raster file to another

netpbm is a package containing something like 200 programs for converting or modifying raster image files. In most cases the name of the program tells you what they do. For example `pstopnm` converts a postscript file into a pnm file. The documentation can be found at `http://netpbm.sourceforge.net/doc` Version 10-6 is installed for the linux workstations.
Note that netpbm does not have a graphic user interface. These are utility programs that work from the command line. If you want to edit graphics files see 4.3.2

4.3.2 How can I edit graphics files

There are lots of choices for editing graphics files. The most comprehensive and therefore complicated is "gnome image manipulation program" or gimp. Gimp can edit just about any file format including postscript and it can do just about anything to it. It is not easy to use but it is very cool.
To launch gimp either find it in the menu or type **@:>** `gimp`

Lots of documentation is available at `http://www.gimp.org/tutorials.html`.
A somewhat simpler graphics editing program is ***ImageMagic*** to launch it type

@:> `display filename`

documentation is available at `http://www.imagemagick.org`.

4.4 Data file formats

4.4.1 How can I convert binhex to useful

The `hexbin` program (on linux) will convert macintosh files to something readable. Use the `-3` flag and look for the `some-file-name.`*data*.

@:> `hexbin -3 filename`

NOTE: `some-file-name` need bear no relation whatsoever to `filename`. `hexbin -3` will give its decoded file(s) names which are encoded in the binhex file itself. The binhex file can be named anything. You probably got it as an

email attachment and saved it under some arbitrary name. For this reason it might be smart to unpack your binhex files in a nearly empty directory.

4.4.2 How can I convert among statistical file formats

The program ***stat-transfer*** is the most useful for converting between SAS, STATA, R, Splus, SPSS, and just about any other data file format. Even Microsoft formats can be dealt with.
To launch stat-transfer type:

```
@:> st
```

then from the stat-transfer prompt (>) you can generally convert files using the `cp` as

```
> cp filename.ext1 filename.ext2
```

where ext1 and ext2 represent the more or less standard file suffixes. To see which "standard" file suffixes mean what to the current version of stat-transfer type:

```
> ?formats
```

`http://lab.demog.berkeley.edu/Refs/stat-trans.unixman.pdf` for more detailed instructions.
NOTE if you happen to need to convert files that are over 2GB in size, `st` will probably fail. A special version of `st`, called `stl` exists on our system. It is not a production version, so it probably will fail in ways that `st` will not, but for large files it's worth a try.

4.4.3 How can I convert an MSword file to ASCII text

```
@:> antiword filename | less
```

4.4.4 How can I print an MSword file without running MSword

```
@:> antiword -p letter filename | lpr
```

Chapter 5

Scanning documents and images

5.1 Where is the scanner

There are currently two scanners one is Most likely in the basement lab connected to `census` the other is probably in the library connected to `logit`. The scanner in the basement is a "document" with a 20 page "auto document feeder". The scanner in the attic is a "flat bed" scanner. The forer is suitable for scanning one or two sided documents (probably into .pdf format) that latter is useful for scanning images one at a time.

5.2 How can I scan a whole stack of pages into a single .pdf file

The Avision AV220 scanner in the basement lab is ideal for this task. Assuming that your document is monochrome and contains writing on no more than two sides of each page, then you can simply:

1. **verify that the Avision AV220 is turned on**
2. **Get a shell on `census`** (unless the Avision AV220 is connected to a different machine). To get a `shell` either logon to census **or `ssh`** to census from another machine.
3. **Run `scan2pdf`.** `scan2pdf` is a locally written program that calls several other programs which: (1) scans your document, (2) convert each page to postscript, (3) converts the postscript into a single .pdf file, and (4) deletes all the intermediate junk.

So, once you have a shell on `census` just type:

```
@:> scan2pdf -o document.pdf
```

where `document.pdf` is the name of the pdf file that you would like to create from the scanned images. `scan2pdf` will prompt you as to whether you like 1 or 2 sided scanning.

> When you're finished, why not turn off the scanner and thereby save the world.

5.2.1 Suppose I want my scanned pages to be something other than pdf...

Before going down this road ask yourself it it is posible to either convert a .pdf into what you want or to use .pdf instead. It's going to be a lot simpler if you can. See Section 4 on file format conversion for more information.
Well if you must scan into something other that .pdf, then you'll need to use the `scanadf` command. The `scanadf` command produces a file for each page in your stack of documents. Those files will be in `.pnm` format, which though seldom used, is easily converted into just about anything via a *pnmto-whatever* command. See Section 4.3.1 for information on the netpbm pakage, which allows you to convert .pnm files to all sorts of other types.

> NOTE: (See Section 5.2 if your goal is a pdf file).

Using scanadf

To scan a 13 page stack of documents into pnm format (a good choice)

```
@:> mkdir scannedpages
```

```
@:> cd scannedpages
```

```
@:> scanadf -o page%02d.pnm --source='ADF Duplex'
    --mode 'Lineart' -y 279 -x 216 --resolution 300
```

The result will be 13 files called 'page01.pnm - page13.pnm' in the scannedpages directory.
Note that the "-y 279" refers to the length of the page(s) being scanned. 297 conresponds to 11.5 inches. "–mode 'Lineart"' is good choice for black and white documents. There a lot of options that you can experiment with, all of which are explained in the man page for `scanadf`.

5.3 How can I scan an image

Once you have located the flatbed scanner (look in the attic) and politely made sure that no one else is using it, you can get started. The scanner makes very little noise when not in use, and has no on/off switch. If `logit` is on, then the scanner is probably on as well – especially if it is plugged in.
Get a terminal window on `logit` (either by logging in or `ssh`'ing from another machine) and type

@:> `xsane`

For tediously detailed instructions, see 5.3.1

5.3.1 Step by step instructions for using the scanner

1. Get a terminal window on `logit`.

   > NOTE: you do not need to be directly logged on to the workstation that drives the scanner in order to use the scanner. You can operate the scanner by remotely logging in (See Section 6.2)

2. Put a document on the scanner's glass window. (Lift up on front of the document feeder). The document should be face down. It does not matter where on the glass you put it. If you prefer to work with right-side-up images, then make sure the top of the document is closest to the wall.

3. Launch the scanner interface program `xsane` by typing the command:

   @:> `xsane`

   The result should be the main xsane dialog window and the preview window shown in shown in figure 5.1

4. Select the proper scan mode. The default is Lineart which is wrong for most things. The scan resolution is set by the slider bar on the main window just below the output filename. Resolution is given in dots per inch (dpi). As you change the dpi setting, a box near the bottom of the window displays the size of the resulting image file. For images that will be used on computer screens, anything above 75 dpi is wasted space.

5. click `Acquire Preview` (on the Preview Window) to see the image of what's on the scanner. The clarity of the preview image is not affected by the setting in the main window. it is just there for cropping and fiddling.

6. To crop the image, click and drag the LEFT BUTTON to create a dotted line box around the part you want. The display in the main window showing the size of the image in KB and in cm will change as you adjust the size of the image to acquire.

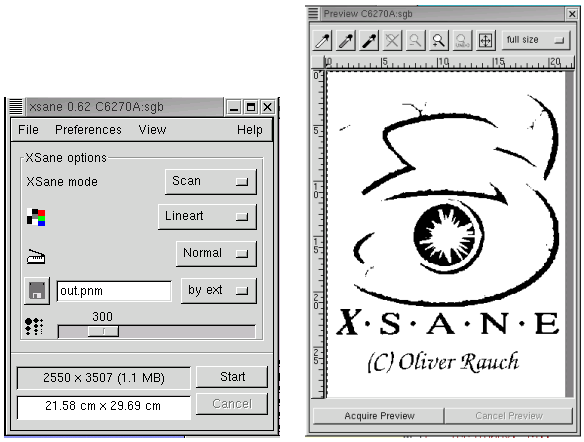

Figure 5.1: xsane main dialog and preview windows

7. If you are happy with what you have in the preview window. Then on the main dialog box (not the preview window) set the filename of the image file that you are about to create. The default is `out.pnm`. Then push `Start`.

   You can determine the format of the image file via the `by ext` pull-down menu. The default type ***.pnm*** is easily converted to other types – but if you know what you want, and it's in the menu well...

8. When the image is scanned, and the `out.pnm` or whatever you decided to call it is complete, you can use either `xv` or (`gimp`) to convert (or edit) it to a more familiar or exotic format. `xv` is easiest (by far) to use, but it has no editing features. `gimp` is a much more sophisticated image manipulation program which can edit, crop, morph, and whatever the file. The commands for launching either are:

```
@:> xv filename @:> gimp filename
```

5.3.2 Photographs

If you are scanning a photograph and understand such things as *custom gamma tables* and *color matrices* you will want to explore options which become available when you choose `color` (as opposed to `Lineart` or `Grayscale`. Other knobs and dials are available under the `[View]` menu.

5.4 How can I convert the graphic images I have scanned into text (OCR) ?

The process by which graphic images of written text are transformed into editable text files is called "optical character recognition" or OCR. We do not have an OCR program running locally, but don't despair. There is a site on the web that will do this for you and Gene Hammel says it works pretty well. Here is Gene's description:

> Subject: Re: ocr
>
> http://docmorph.nlm.nih.gov/docmorph/
>
> Contains a good deal of information. Basically, a user must register and have a password, then can log in and do all kinds of graphic and text file conversions (except from pdf to anything else). You upload the file to be transformed, you are informed when it is ready to download, and that's it. It all takes place on the WEB page, in interaction with the file browsing abilities of your own OS.
>
> I had scanned 18pp of typed text at 600 dpi. These were *.pnm files. Each was about 1 MB. I read the instructions on the docmorph page, then sent and received each file separately.
>
> First you click on a link to upload a file. You can type in the path or browse for it, then click submit. After it uploads you get a message saying to wait; it takes 12 seconds to process a page. Then a link appears that allows you to download the file, which appears in your WEB browser. I then saved this page as *.txt; the file naming is automatic. Then I clicked the other link, to upload a new file, and went through the process again.
>
> There were no difficulties whatever. The OCR seems to be perfect although I have yet to read through it. The txt files are about 3 KB each. The only problem is tables; these do not preserve their original format, and I need to inquire of their tech support how to do this.
>
> It might have been simpler if I had catenated all the pnm files and uploaded just one file for conversion. But I do not know what happens to a set of pnm files when catenated, and it might be tricky to get them in the right order if one used a regexp in the cat command line so as to avoid a lot of tedious typing. Something like cat file[0001..0055].pnm >> allfiles.pnm might work.
>
> In short, I do recommend it.

Chapter 6

Remote, Portable and Wireless Access

6.1 How should I connect the the Demography Lab from afar

This used to be complicated, but now it is simple: use FreeNX. To do so you will need to install a free client application on your machine and do a little configuring. Once you have it set up, you will be able to connect easily and efficiently and do just about everything from afar that you can do from 2232 Piedmont.
FreeNX is explained in `http://lab.demog.berkeley.edu/LabWiki`.

6.2 How can I just just get a remote Demography Lab login shell

The Demography Lab is accessible only via encrypted connection. Telnet, and `rlogin` are not encrypted so you cannot use those. If your personal computer runs linux or mac OSX, `ssh` is probably installed and usable, just open ("Terminal" on a mac) and shell type

```
@:> ssh demog.berkeley.edu
```

NOTE that this is also the command you would use to connect to a server from within the local network – except instead of demog.berkeley.edu you would type the name of the server e.g. `tapinos`, or `coale`.
If you are running Windows, below are two other options.

> NOTE: firewall software often inteferes with ssh in diabolical ways. If things don't work as advertized, try disabling your firewall

- mindterm Using your java enabled browser, click on `http://lab.demog.berkeley.edu/Mindterm`. This java application provides encrypted "ssh" connection to any site. It is very handy when you are away from home and only have a browser to work with.

- Secure Shell

  Several options exist for running ssh from windows. All of them require some installation. Most people (who need a secure shell under windows) use PuTTY.
  `http://www.chiark.greenend.org.uk/~sgtatham/putty/`.

6.3 How can I get my ethernet card equipped portable onto the network

There are two network switches aka "laptop landing zones" in the building. One is in the basement lab and the other is in the Library. If your machine has an ethernet card (which could be built in or PCMCIA) you should be able to connect simply by:

1. leaving all network parameters set to their default values. If you need to reset these values – the key idea is to allow the DHCP server to assign all of the ipaddresses, network masks & etc.

2. Connecting your machine to an unused port on one of the above mentioned switches via a "category 5" patch cable. (It looks like a telephone wire on steroids). On a good day there will be such a wire sticking out of a port on the switch.

BUT WAIT: Campus security policy makes connecting your portable just a little bit tougher. In order for the above procedure to work, you must send email to troubledemog.berkeley.edu telling us who your are and the "mac address" of your ethernet card. IF you have no clue about what a mac address is or how to find it see 6.7.

6.4 How can I get my 802.11 equipped portable onto the network

AirBears. AirBears is run by IST, it is available from all over campus to anyone with a CalNet ID.

6.5 How can I transfer files between my home computer and the Demography Lab

6.5.1 DropBox

DropBox is an application that synchronizes files and directories between machines. You can configure it to do so between your portable machine and the Demography servers. Just find
`[Applications]`→`[Internet]`→`[DropBox]`.
TWO IMPORTANT CONSIDERATIONS If you decide to setup DropBox:

1. **Do not put the DropBox folder in it's default location**. During the configuration process make sure that DropBox puts it's folder in `/data/commons/your-userid` otherwise it will quickly use up your home directory quota.

2. It is a good idea to configure DropBox to only share **some** of your stuff. There is no need to access you collection of Woody Allen movies on the Demography server. Just share the part of your DropBox volume that you do science with.

6.5.2 sftp

If you have secure shell installed on your home computer, you can use `sftp`. The `sftp` program functions somewhat like `ftp` – which we no longer use as it is not encrypted.
If your home machine runs Linux, you would type:

@:> `sftp userid@demog.berkeley.edu`

after logging in, you can move around and find stuff using `cd` and `ls`. To move a file to the machine you are sitting in front of, type `get filename` to move a file to the Demography Lab network, type `put filename`.
Under MSWindows, the ssh which you can download for free (as long as you affiliated with UCB) see 6.2, has sftp application with a reportedly "easy to use" interface.
NOTE: `sftp` is picky about what it sees on STDOUT. If your `.bashrc` echo's a message, it is possible that `sftp` will hang after taking your password.

rsync to transfer a lot of files

One option for transferring/synchronizing entire directories is `rsync`. Rsync is free and available on many platforms. It is much more powerful than it is simple to use so be careful. Treat it as you would a circular saw...with all the safety guards removed. Read the man page carefully before experimenting. Your best friend, where `rsync` is concerned, is the `---dry-run` argument. Also, note that in order to use `rsync` from outside of the department, you will

need to tell rsync to operate over an encrypted connections. This is done via the `-e ssh` argument.
Here's a simple example of how to move all the files from a directory called `dissertation` in your Demography Lab home directory, to directory called `backup/dissertation` on your home computer.
Assuming your home computer runs Linux, of course:

```
@:> rsync -uva -e ssh userid@demog.berkeley.edu:dissertation ./backup
```

6.6 How can I print from my portable

Instructions for setting up your portable computer to print to Demography printers are found in `http://lab.demog.berkeley.edu` under the `[Documentation]` link.

printing from windows machines

6.7 How can I find my "mac address"

The "mac address" also goes by the names "ethernet address", "hardware address", "station address", "ethernet id" and "physical address" and perhaps there are others. It is a 12 digit hexadecimal number generally shown with a ":" separating each pair of digits. Since it's hexadecimal, digits include the letters A-F. For example: "00:0C:76:00:4A:FA".
In most cases the mac address will be printed somewhere on you network card. If not, your computer can tell you. Under Linux the command is `/sbin/ifconfig`. The numbers you want will be labeled "HWaddress".
Under Windows, if you can find a "command prompt", you can type: `ipconfig /all`. The answer will be reported as "physical address". NOTE that there may be more than one such *physical address* so make sure you get the one associated with the wireless card. (Thanks to Sarah Staveteig for this)
Here is a helpful site with more instructions on how to find your mac address under various OS's: this `http://www-dcn.fnal.gov/DCG-Docs/mac`.
Be especially vigilant if you have two or more network devices–as most portable computers do. Each such device has its own unique mac address. Make sure the mac address you give us is associated with the device that you plan to connect to the LAN with.

Chapter 7

LaTeX

7.0.1 How can I learn to use LaTeX

A good place to start is with the *Not so short guide* available at `http://www.demog.berkeley.edu/Refs/lshort.pdf`. There are also several pretty good books on using LaTeX. Unfortunately the tend to be rather expensive. There should be a few copies of *LaTeX: A Document Preparation System* by Leslie Lamport lying about the lab.

7.0.2 Are there any slick tools for editing LaTeX documents

Emacs is great way edit LaTeX documents, if you like emacs. By visiting a buffer/file with a `.tex` suffix, emacs should put you in "LaTeX" mode all sorts of good things will follow therefrom. Documentation on Auctex can be found at `http://www.demog.berkeley.edu/Refs/auc-tex.ps`
For those not inclined toward emacs, there are other choices such as LyX `http://www.lyx.org/` or on macs, TeXshop
`http://pages.uoregon.edu/koch/texshop/`

7.0.3 Can LaTeX documents be converted into pdf

yes `pdflatex` can do this.

@:> `pdflatex filename(.tex)` .

But embedded graphics are lost, so if your document is just text this is quick and easy. But...
If your document has embedded graphics then the two step process below works very well:

@:> `latex filename|tex` **@:>** `dvips -Pcms filename|dvi`

@:> `ps2pdf13 filename.ps`

It is also very effective in almost all cases to convert the `.dvi` file to pdf using:

@:> `dvipdf filename.dvi` .

7.0.4 What's the best way to put LaTeX documents on the web

There are at least two pretty good ways of doing this:

`htlatex filename` takes `filename.tex` and creates a single filename.html – with potentially lots of other files on which that single html file relies, e.g. graphic images converted to .png format.
The command

`latex2html filename.tex` will create a subdirectory called `filename` containing a bunch of HTML files linked together. Just give that directory name as the URL. Because all the filenames and links are machine generated, it is not fruitful to attempt to edit the HTML files. The right approach is to edit the latex files if you need to change something.
`{\use{hyperref}` is a nice package to include in documents that are bound for the web.

7.0.5 How might one create a table in LaTeX –without losing one's mind

There are two pretty good choices:

1. If the data for your table happens to reside in R, then you can use the `xtable()` function to output a chunk of LaTeX for inclusion. See 3.3.1 for details

2. If you prefer to create tables in a spreadsheet program, then you can use `gnumeric`. See 7.0.5

How can I use gnumeric to make tables in LaTeX

A reasonable approach for making tables in LaTeX is with `gnumeric`.
Gnumeric is nice spreadsheet program which can read `.xls` files and can save them as `.tex` files.
It is of course, not as simple as one might wish. Gnumeric uses the `longtable` package so in order to modify what `gnumeric` does, you will to come to terms with `longtable`.
Here is a rough procedure for getting a table into a LaTeX document. If your tables aren't too complicated and you are not too picky, it'll probably work. But if the wheels come off, you'll want to have a look at the documentation – there's lots of LaTeX docs at `www/Refs/LaTeX-DOCS`.

Create your table in gnumeric you could do this by importing an `.xls` or `.sxc` file or you can start from scratch or whatever–it's a spreadsheet program. Resist the temptation to get fancy, colors and fonts and such will be lost in the next step. Make sure each table is in a separate tab. It would be smart **not** to include the title of your table in the `gnumeric` version. That goes in `\caption` later.

save the file as LaTeX Save each table to a file. Use `[file]`→`[save as]` and select "LaTeX 2e."

Follow the directions in the file you just created The section at the top of the file that `gnumeric` spit out informs you of all the the `\usepackage` commands that you'll need to insert in your "including" document. That is in the file that you want the tables to show up in ultimately.

All of the `\usepackage` and `\newlength` commands come between the `\documentclass` and the `begin{document}` command. That is the go in the "preamble."

Edit the file that `gnumeric` wrote The file that `gnumeric` creates includes a "float" it is the moral equivalent of something inside a:

```
\begin{table}[htbp]

\end{table}
```

You'll surely want to add a `\caption` and a `\label` to your table. To do so, you need to edit the file that `gnumeric` wrote.

Look for the command that looks like:

```
\begin{longtable}[c]{%
        b{\gnumericColA}%
        b{\gnumericColB}%
        b{\gnumericColC}%
        b{\gnumericColD}%
        }
```

and make it look like this:

```
\begin{longtable}[c]{%
        b{\gnumericColA}%
        b{\gnumericColB}%
        b{\gnumericColC}%
        b{\gnumericColD}%
        }\caption{Fascinating Table Showing Dramatic Result}\\
\label{tab:fascinating}
```

NOTE the \\ at the end of the `\caption`.

use `input` to include your table . The `\input` directive will act as though the input'ed file is part of the including document. NOTE that since the input'ed file contains a table-like float, the table counter will be updated, and the `listoftables` directive will recognize it as a table. Also, since it's a float, the table will appear where LaTeX thinks it should – which might not be right where the `\inpt` directive resides.

Chapter 8

Reading and Writing USB devices, CDROMs, Floppies, and DVDs

8.1 What is "mounting" and why should I care?

The most convenient way to access the stuff on your floppy or CDROM, is to make the contents of the entire disk part of the filesystem[1] The process of joining a device to the filesystem is known as *mounting* and can be accomplished manually via the `mount` command. But generally, the workstation will mount it automatically or nearly automatically and put something on your desktop for you to click on.

> What you cannot rely on the machine to do for you is **unmounting**. See Section 8.3 for the why and how of this.

So, you stick your CDROM or floppy or USB device into the appropriate slot and then look for evidence of your new device.
On a good day, your workstation will automatically mount it and there by join it to the filesystem at the approrpriate ***mount point***[2]. It will also *probably*

[1] the filesystem can be thought of as the collection of all of the directories to which you can cd. Although this tree of directories behaves as though it lives entirely on the local hard disk, it is in fact distributed across many machines which share their local disks via a protocol known as NFS.

[2] The ***mount point*** associated with the newly mounted device is the place in the filesystem where the CD or floppy has been grafted onto the filesystem 'tree'. In other words, if the floppy drive is mounted on `/media/floppy` then, in a terminal window, you can cd /media/floppy and see and touch all of the files on that disk. Table reffig:mount1 shows the auto-mount points for each device

pop up a `nautilus` (file manager) window in which you can putz around and click and drag and whatnot.

8.1.1 What if it's a bad day and my device does not appear magically on the desktop

Well if you've been bad, and you device does not appear as promised you can try the following things:

- Look at the task bar for an icon like this:  if you LEFT BUTTON on it you should see a menu that will allow you to "mount" or "unmount" the device.
- Take a look under the `[System]`→`[Prefences]`→`[Hardware]`→`[Removeable Drives and Media]` for further options.

Though mounting a device is simple and automatic, it is not the only way to access data on floppy disks. See Section 8.6 for an alternative.

8.2 Do I have to "mount" floppies, zips and CDROM?

Yes (except for floppies[3]) – if you want to see what is on them, but the workstation will do it automatically for you.

8.3 Do I also have to "un-mount" removable media?

YES YES YES. **especially if you just wrote something to it**. Unix often "buffers" writes in order to be more efficient. If you just yank a the media out of the drive – or disconnect a USB device, who knows what's been written and whether files have been closed.
The easiest way to un-mount a device is to click left button on the  **icon (on your taskbar) then select `[unmount]` or `[eject]`.**
If your are working with the device through the filemanager (nautilus), then should be an unmount or "eject" button within nautilus as well.

[3] See Section 8.6

NOTE that unmounting will fail if you are accessing the media either through an application (including `nautilus`) or as the current working directory of a shell. You must `cd` out of the media and close all applications that might be accessing the media **before** you can unmount it.

Figure 8.1: Auto mount points

device	**mount point**	**to access contents**	**retreival procedure**
cdrom	/media/<name-of-cdrom>	cd /media/<name-of-cdrom>	eject cdrom **or** taskbar icon
floppy	/media/floppy	cd /media/floppy	umount /media/floppy **or** taskbar icon **then** push the button
usb device	/media/disk	cd /media/disk	umount /media/disk **or** taskbar icon **BEFORE disconnecting**

8.4 How might one (re)format a floppy disks?

In the simplest case where you have a standard 1.44MB floppy and you want to put a basic DOS filesystem on it (so you can read the disk on a windows machine) just put the disk in the drive and type:

@:> `formatfd`

This command is locally aliased to run:
`/bin/fdformat /dev/fd0H1440; mformat a:`
read on to find out what this means.
If you only put tar files on floppy diskettes then the "high level" format described below is irrelevant. It neither helps nor hurts to create it. You **must**, however, be sure that your diskette has a low-level format. Instructions for creating a low-level format are given below. But if you wanna save time... just run **@:>** `formatfd` like everyone else. Then

@:> `tar -cvf /dev/fd0 ./path/to/files`

8.5 What is the general theory of formatting a floppy diskette?

Floppy disks have 2 levels of formatting: a "low level" format which determines how the little ones and zeroes are stored on the diskette. This has to do with tracks and sectors and stuff like that. Once the low level format is

done, diskette controllers handle it and it becomes very uninteresting to end users. The only thing you need to know is that if you have a raw diskette, you have to make sure this low level format is created. In almost all cases it will be pretty clear if the low level format is faulty, because just about nothing will work.
The second "high level" format refers to the filesystem. The filesystem takes care of file names, permissions, directory structures and that sort of thing. For most purposes, you will want your floppy disk to have a DOS or "FAT" filesystem on it. These can be read both by Windows and Linux machines. If you only use Linux machines, you may wish to format your diskettes as *ext2* but you will probably be the only person on the planet who does this.
If you wish to put an ext2 (linux) filesystem on your floppy disk, use the command:

```
@:> /sbin/mke2fs /dev/fd0
```

8.5.1 low-level format

Use the `fdformat` command to install a low level format on a raw diskette. You need to tell the fdformat command what kind of diskette you have. The way this is done is to specify the appropriate device driver. In the box above, the device driver is `/dev/fd0H1440` which more or less translates to *floppy disk 0 High density 1440 KB*. All machines around here have 1 or fewer diskette drives so the "/dev/fd0" part never changes. Where you need to give some thought is to the "H1440" part. 1440 KB is very much the standard but it is possible to find some "H720" diskettes. If you have anything else, I would toss it. Diskettes are cheap.

8.5.2 High level format

To put a standard DOS (FAT) filesystem on your diskette use the command:
`@:> mformat a:`
Note that `mformat` is part of a suite of tools known collectively as mtools. See Section 8.6.

8.6 How can I access diskettes with DOS-like commands?

The `mtools` suite of programs allows you to operate on diskettes, just like you used to – if you are over 30[4].
mtools contain a version of just about all of the DOS commands that can be used to fiddle with floppies just stick an 'm' in front of the DOS command as in `mcopy, mdir, mcd, mdel` and `mdeltree`. All of these exist and work as

[4]If you're under 30 DOS stands for Disk Operating System – it was widely in use in the 1980s

you *might not* expect with `a:` instead of `/dev/fd0`. Consult the man page (`@:>` `man mtools`) for a complete list.

8.7 What steps are involved in writing a CDROM?

A growing subset of machines (including nearly all the machines in the basement lab) have (re)writable CDROM nd DVD drives. Obviously, writing to a CD will not work well unless you are using one of these machines.
Writing CDs is a 2 step process (or a 3 step process if you include going to the store (or Rm 101B) to buy a blank CD-(W)R disk). First a disk image must be created. This is file that contains the image of what will eventually be on the CD. Since you can only write once to CDs you cannot simply add files one at a time. The entire filesystem including the bookkeeping parts that you don't generally realize are there must be assembled before the CD can be written. A filesystem image assembled in this way is usually stored in a file with a ".iso" suffix. The second step is to transfer this filesystem image to the CD.
You need only be vaguely aware of these two steps if you choose to make your CDs with `nautilus` file manager application – aka "the easy way".

8.7.1 How might one write a CD/DVD the easy way

1. Assemble all the files that you want to write to CD or DVD into one or a few directories – which include **only** stuff that you want to copy to the CD/DVD

2. Open a `nautilus` window either by clicking on  or by selecting: `[Applications]`→`[System Tools]`→`[file Browser]`

3. From the `nautilus [Places]` menu select `[CD /DVD creator]`. This will produce another `nautilus` window.

4. Navigate, in the original `nautilus` window to the directory containing the stuff you want to copy to CD and drag it over to the `CD/DVD Creator` window.

5. When everything that you want to copy to CD is in the `CD/DVD Creator` window, hit the hit the `Write to Disk` button.

6. You should now be prompted to insert a blank or re-writable disk. It *should* be enough to stick a disk in the drive and close it, but often it takes the machine upwards of 30 seconds to figure out that yout you have complied with its request. Consequently, the "Insert rewriteable or blank disk" dialogue box sometimes appears and reappears and reappears – you might need to hit `OK` seventeen or eighteen times.

Gotcha: If you are using previously written re-writeable media, it will be necessary to erase the disk before writing. If you follow the order of things given above, you will be prompted to erase anything that is already written you your disk. If, however, you do not follow the above order and insert the disk that you wish to re-write **before** you hit the `Write to disk` button, then confusion may follow.
The reason is that when you insert a nonblank disk into the drive, the machine will probably assume that you want to read it and it will therefore attempt to mount it for you. You cannot write to a disk that is mounted for reading.

8.7.2 How might one create a CD/DVD the "hard" way

The hard way isn't really all that hard and is useful if you want to make sure that certain nonstandard features are included in your CD/DVD – or if you just prefer typing to clicking. Follow the directions below for first creating the disk image file and then for transferring it to CD.

How do I Create a disk image

To create the disk image, use the `mkisofs` command. It works best if all of the files of interest reside beneath a single top level directory, but this is not essential[5]. Suppose for example the you wanted to make a CD of `/data/commons/user/Dissertation` and all it's subdirectories. The command would be:

```
@:> mkisofs -iso-level 3 -L -allow-multidot -o
    /72hours/<cd-image.iso> /data/commons/user/Dissertation
```

`cd-image.iso` is the filesystem image that you want to make. The `.iso` suffix is optional, but cheap so why not do it.

[5] there is a man page on `mkisofs` which describes all the options

The <iso-level 3> arguement tells misofs to conform to the ISO-9660 level 3 standard. This allows files names to be 31 characters long; directories to be nested 8 levels deep; and full path names to be as long as 255 characters. If you plan to use the CD on an ancient machine, you may wish to leave out this argument–that will result in filenames no longer than 8 (upper case) characters.
Other arguments to mkisofs that can affect the way filenames appear and which you might wish not to use if you plan to read the CD on an ancient system include:

- -L allows filenames to begin with '.'. The default behavior is to change a leading '.' to '_'
- -allow-lowercase allows filenames to include both upper and lower case letters. The default is to convert filenames to upper case
- -allow-multidot allows more than one '.' in a filename

the mkisofs man page warns that all of these options violate the ISO9660 standard but "happen to work on many systems".

NOTE: The /other/72hours directory is a large enough to hold many CD/DVD image files, but as the name suggests, data in that directory will be erased without notice in 72 hours. In other words, it's scratch space. You may substitute something meaningful for <cd-image>.

How might I transfer the disk image to the CD/DVD?

Once your disk image is made, you can use wodim to transfer it to CD – Or you can just RIGHT BUTTON on the .iso file in a nautilus window and select Write to Disk.
The macho way is to use the wodim command. To transfer a cd image called <cd-image.iso> that is in the current working directory – probably /other/72hours. Don't forget to put your **blank** CD/DVD disk in the drive before running wodim.

@:> `wodim -v -data <cd-image.iso>`

Once the transfer operation is complete, you can inspect the CD's content by mounting it. Just eject the CD and then push it back in.
NOTE: if you are re-using re-writeable media, you must "blank" or erase the disk before attempting to write to it. See 8.7.2 for instructions.

If the above command fails, it is probably because wodim has the wrong cd/dvd device set as default. you can get wodim to tell you what devices exist by following command:

@:> `wodim --devices`

Generally the result of the above command will indicate the correct argument as something like "dev=/dev/scd0".

@:> `wodim dev=/dev/scd0`

will then solve the problem.

How can use `wodim` to erase a cd/dvd?

A rewriteable cd/dvd must be "blanked" before new stuff can be written to it. CDs are not like floppies or USB drives, you cannot simply add files to an already written CD. The entire disk must be erased in order for new content to be written.
To blank a disk:

1. put the disk in the appropriate drive **but do not close the drive door**
2. execute the following command:

   @:> `wodim blank=fast`

   On a good day, the drive should now close by itself and the blanking process should start.[6]

On a not so good day, `wodim` will gag on the above command. To make `wodim` try harder you can add the `-force` flag. Ultimately, the following command will do the trick, but it can take quite a while to complete:

@:> `wodim -force blank=all`

[6] If you blank a lot of disks it might be easier to set your hardware options so that the machine does not automatically try to mount a disk as soon as the drive door closes.
To do this go to `[System]`→`[Preference]`→`[Hardware]`→`[Removable drives and media]` and clear the "mount removable media when inserted" check box.

Chapter 9

Office Applications

9.0.3 What kind of word processing applications are available

Your best (long run) option for word processing is LaTeX – See `www.demog.berkeley.edu/Ownersmanual` for and impassioned diatribe regarding the virtues of logically oriented text processing.
If you lack sufficient idealism to pursue LaTeX – or if you just prefer something will lower startup costs – the OpenOffice suite is a good options. OpenOffice (OO) is an open-source alternative to the Microsoft Office software. OO is supported by Sun which directly supports StarOffice – we generally use StarOffice here (since the campus buys it) but I'll refer to it as OO anywya. OO works just about the same way as MS with a few exceptions that might annoy you at first – if you are accustomed to the MS way and are too rigid to consider alternatives.
OO and MS office software are sufficiently similar that one can easily go back and forth between them. OO can both read and write all of MS's secret proprietary formats and even has its own version of the infernal talking paper clip.
It is also possible to run MS Office applications under Linux. We do this with cxoffice See 10. Since MS applications run on Linux only under duress, you can expect some quirks and glitches. Of course, you expect that with any MS software.
It is less frustrating to work with the OpenOffice applications under Linux. OpenOffice is also available for Windows and Mac for free.

Equation editors are quite different

If one moves back and forth between OO and MS, one must be a bit careful about equations. OO can display equations created in MS but it is best **not** to try to edit them. MS does not understand OO's equations at all.

9.0.4 What spreadsheet programs are available

OpenOffice also has a spreadsheet that compares well to MS Excel. As with the word processors there are slight differences, but OO can do everything that MS can and once you get used to the small differences it is not difficult to move back and forth between them.

9.0.5 What happens when I run OpenOffice for the first time

The **first** time you run open office, a screen will popup offering to allow you to "install" OpenOffice. All the default suggestions are fine so if you're brave you can stop reading here and click away. More details for the more timid are below.
What happens first will depend on whether you have used previous versions of OpenOffice before. If you have you may be prompted to update and old installation – resist this temptation and "install" in a new directory.
Eventually the following steps happen:

1. You will be asked early on to read through a bunch of "important" information and then click `next` to go on. (you can decide for yourself).
2. You will be asked to read and accept the license agreement
3. You will be asked to enter "User Data" such as your name and email address.
4. You will give you a choice of installing either the "workstation" or the "local" version. **"Workstation"** is what you want to choose. It is the default and it just installs about 1.2MB of "dot file" that configuration files for you personal preferences and what not. Choose the default and move on. You may be asked to replace files in `.kde2` – go ahead and accept.
5. You should then be asked permission to create a new directory with a long name that includes "OpenOffice.org1.1.2" (the numbers might be higher by now). Say yes and everything should finish up nicely.
6. Then you hit the `install`. Some lights flash and then you hit the `complete` and you're done... with the installation

After the setup program exits, you can launch any of the various OpenOffice applications as described in 9.0.6

9.0.6 How do I launch OpenOffice Applications

There are several ways of launching OpenOffice applications.

from the command line If you wish to open an existing document – either one produced in OpenOffice or in some Microsoft application, you can type:

```
@:> soffice filename
```

where filename includes the typical file suffix such as `.doc` or `.xls`. `soffice` will choose the appropriate application to open you document with.

If you wish to start a new document, then you can save keystrokes by launching the particular application that you want to use. The application command names are shown in Table 9.1

Word Processor	`swriter`
Spreadsheet	`scalc`
Presentation	`simpress`
Drawing	`sdraw`
Equation editor	`smath`

Table 9.1: OpenOffice Applications

From the menu or panel the OO applications are in the menus under. `[Applications]`→`[Office]`. They each have their own nifty post-literate symbol on the panel.

From the nautilus filemanager The filemanager, lives on the desktop it's called "userid's Home". It shows you all the files and directories and if you click on an appropriate document it will open it with something. If the file in question is a native OO document, then click and be happy. If, however, it is an MS `.doc` file then there are lots of choices as to how it should be opened and you will need to use the RIGHT BUTTON to select which application you want.

Chapter 10

Running Windows Applications under Cxoffice

10.1 What is Cross-over Office and why should I care?

Crossover Office (or cxoffice) is commercial grade WINE implementation. Its purpose is to allow windows addicts to run their favorite applications under Linux.
It works amazingly well – considering the challenge. In other words, it works, but there are a few bugs.
From the `[RedHat]`→`[X Windows Applications]` you can launch Word, Excel, or Powerpoint. In nearly all respects these program behave, under Crossover Office, exactly as the do under Windows. With just a few additional idiosyncrasies and bugs thrown in.

10.2 What applications run under CXoffice?

Currently the following windows applications can be run from your Linux desktop:

- MS Word 97
- MS Excel 97
- MS PowerPoint 97
- Census CD 2000
- Census CD 1990
- Census CD 1980

More will be added as we gain experience with this system.

10.3 What are some of the bugs that we know about?

10.3.1 Printing fails

If MS Word et. al. simply will not print – and will not give you any sort of error message, the following procedure is likely to help. But it is kind of radical. You should **not** need to make a habit of this.

1. **Close all MS Applications** or at least make sure that your work is saved.

2. **delete your entire .cxoffice directory** This is the directory that stores your personal configuration. If you have changed preferences and such, that will be lost. If you hate that prospect, skip this step, and maybe redo the whole procedure if printing still fails.

   Be extremely careful with this command – if you screwup bad things could happen.

   @:> `cd; /bin/rm -r .cxoffice`

3. **Reset CXoffice** Either find it on the menu or else at the Unix prompt type:

   @:> `/other/cxoffice/bin/cxreset`

   When it asks permission to proceed, say yes.

4. **Verify that PRINTER and LPDEST are not set** `PRINTER` and `LPDEST` are environment variables that determine your default printer. Their presence will screwup CXoffice. To find out if you have these variables set type: **@:>** `echo $PRINTER` **@:>** `echo $LPDEST`

   If the name of a printer is returned from either of these commands, edit your `.tcshrc` file and delete them. See the *Owner's Manual* if the phrase "edit your .tcshrc file" intimidates you.

   If your `.tcshrc` file did set PRINTER or LPDEST, then you will need to logoff and log back in before you continue.

5. **Verify that Wine-Postscript-driver is your default printer** Start word and try printing. Your default printer should be "Wine-Postscript-driver". The first time you print, you **might see an error message** regarding memory. It can be ignored.

If the above procedure worked, then printing to the "wine-postscript-driver" will launch a printing application which will give you lots of choices as to where to send your print job and how to print it. **The first time you see this kprinter application** you must change the "printer driver" setting from

LPD to CUPS. This only needs to be done once (unless you do that radical reset procedure in 10.3.1.
Hereafter, always select the "Wine-Postscript-driver". And **never** try to print MS docs to status. (See 10.3.2).

10.3.2 The printer status fails to print MS Word files

status is the printer in the library. If you try to print an MS Word file to it, you will get a line or two of incomprehensible junk instead of your file. NOTE that for best results you should print to the "wine-postscript-driver" rather than any familiar printer directly. The "wine-postscript-driver" launches a GUI application that gives you more control over your print jobs. Printing to status will not work from there either however.

Workarounds include:

- printing to a different printer
- printing to a file then running the command:
  @:> `ps2ps filename - | lpr -Pstatus`

10.3.3 Equation Editor is fragile

The equation editor under cxoffice is prone to crashing but the crashes are not random.

Workaround : when using the mouse click crisply. Do **not** allow your heavy finger to rest with the mouse button depressed as you select and drag mathematical symbols into your equation. If you do equation editor will die. If not, it will function acceptably

10.3.4 CensusCD 1980 and 1990 fail to export text/dbf files

CensusCD 2000 works properly, but the 1980 and 1990 applications generate lots of errors when you try to create a flat ascii or dbf file of the data. Mapping works fine – it is only the data exporting function that fails. In most cases, it writes the file before failing for what that's worth. But you will need to use [RedHat]→[CrossOverOffice]→[Reset Crossover Office] to recover control of your desktop after each failure.

Workaround There is no good work around, you can just write and clean up as noted above, or you can go to the library in Haas or the lab in 64 Barrows to use these applications on native windows machines.

Chapter 11

Protection of Personal Data

11.1 Is it ok to store personal/financial data on the computer

Senate Bill 1386, Assembly Bill 700 and common sense dictate that care must be taken with personal financial data stored on computers – especially computers connected to the Internet.
By directive of the Chair, there shall be no systematic collection or storage of *protected Information* on Demography Department Computers. By **Protected Information** we mean "protected information" as it is defined in the above noted Senate and Assembly Bills. This includes an individual's first and last name in combination with any of the following:

- social security number
- driver's license number
- financial account number
- credit card number
- a password enabling access to financial accounts

Staff, faculty and students are therefore instructed to remove any and all such information from any database system or file on any networked device in the Demography Department.
By Dec 31, 2005 (or when your account is set up) you will be asked to sign a statement promising to comply with this directive.
The chair also advises that when called upon by outside agencies to provide social security numbers for Demographers, polite resistance is a good strategy. It is often the case that even though the important looking form *asks* for a social security number, none is actually *required*.

Sensitive data such as grades and exams which need to be accessed on computers should either be stored on removable media **only** or be encrypted using a public-key encryption system such as GnuPG with an adequate keysize. If you choose to encrypt files there are choices to be made about how that should be done.

11.2 How can I encrypt a sensitive file

The program `gpg` (GnuPG) is the best option for encrypting files. For details on how to use it's many features check the man page:

@:> `man gpg`

Briefly, gpg has two modes of encryption:

- The simplest encryption option is called a "symmetric cipher". This is simple to use yet it can only be broken by guessing. So if there is any chance that you might lose or forget you pass-phrase, than your symmetrically ciphered data would be lost.

  This is not such a big deal if you have also stored that data on a CDrom, under your bed.

- A more complicated, but in some ways safer method is to encrypt with multiple "keys" including at least one other trusted person's public key in order that they may be decrypted should you lose your private key or be hit by a bus.

If you need encrypt with multiple keys, contact carlm and we'll figure something out. If you just want to do the easy thing see 11.3

11.3 How do I just do that easy thing you mentioned

To "symmetrically cipher" a file simply type:

@:> `gpg -c filename`

you will be prompted twice for a pass-phrase – it can contain spaces and it can be long.
You will probably see a message warning you about "insecure memory". Don't worry about it unless you like to worry.
When complete you will have a file called `filename.gpg` as well as your original **unencrypted** file.
Obviously it would be rather boneheaded to leave the unencrypted file in place once you are sure that the encryption worked. right?
To decrypt the file you just encrypted type:

@:> `gpg -d filename.gpg > filename.decrypted`

You don't need to call the resulting file `filename.decrypted` any filename will do. But existing files by the same name will be **overwritten**.
Before deleting the unencrypted file you may wish to check that this process really worked. Here' how:

```
@:> cmp filename filename.decrypted
```

`cmp` returns nothing if the two files are identical. If they are not – it tells you so.
Now, you can safely erase the unencrypted file – if you can remember the password.

L

Chapter 12

Disk Usage

12.1 How much disk space can I use on demography system

The short answer is "lots" – but there is a more complicated answer. The demography system maintains 3 distinct types of disk space for distinct types of uses:

home directories Space in home directories is the most limited and is to be used for the highest value files. These include programs that you write, correspondence that is not too old, preference files for software that you use and the text for your dissertation. These are files that were expensive to produce and would be extraordinarily painful to reproduce. The reason this space is limited is that we back it up very carefully and very frequently. If we backed up all of our disks as carefully as we backup home directories we would do nothing but load and unload tapes – and of course we would make even more mistakes than we already do.

data directories Space in data directories such as `/data/commons` is less limited. This is a good place to store things like data sets. Data directories are for large files which can be relatively easily reproduced – by say downloading them again. Data directories are backed up weekly. Since data sets are not supposed to change – it doesn't matter that backup copy is a week old.

temporary directories Temporary directories are for temporary files (duhhh). Programming in SAS or STATA means writing lots of temporary files. By default SAS writes temp files to /72hours (aka /Sastemp). STATA on the other hand writes those files to its current directory (so watch out for STATA).

These intermediate results / temporary files belong in `/72hours` where they are not backed up an are deleted after 72 hours of disuse.

If you are using large data files, and you politely store them in a compressed state (See 12.4 for advice on this) temporary directories like `/72hours` are generally good places to uncompress.

NOTE: `/72hours` are not **networked**. That is, `/72hours` on `coale` is not the same as `/72hours` on `tapinos` or `census` or whatever. They are all different, and to move stuff between them you need to use a program like `sftp`.

Directory	Soft limit	Hard limit	Grace period	Backed up
home directory ~ or /hdir/0/username	350MB	500MB	7 days	daily
data directories /data/commons	1.5GB	7GB	28 days	weekly
temporary directories: /Sastemp /other/72hours	politeness	whole disk	deleted after 72 hours	never

Table 12.1: Quotas and backup times

12.2 How are disk use quotas enforced

The quota system is designed to keep users from "carelessly" consuming huge amounts of disk space. Of course this is far more complicated than simply refusing to do anything for users when their allotment of space is used up. That would be mean. Instead when a user exceeds her *soft* limit for a particular filesystem, the quota system begins a count down that lasts a period of days. While that countdown is in progress, the user can operate as if nothing were wrong – but she will get warnings when she either logs onto a server **or** reads her email.
When the countdown (or "grace period") is complete, the system then imposes the only penalty that gets the users attention: no more disk writes until disk use falls below the soft limit. Read access is still permitted, and so is file removal.
There is also, a *hard* limit over which a user's disk use can never go. This is designed to stop terrorist processes from filling up entire disks.
We set the hard limits, soft limits and grace periods independently for each filesystem. Home directories, `/data/commons`, `/Sastemp` and `/other/72hours` all live on different filesystems and therefore have different *soft limits hard limits* and *grace periods*.

12.3 checking and managing disk consumption

If you are the sort of person who prefers to shape her own destiny, then it behooves you to periodically monitor the growth of that garbage heap which is your home directory. Since your stored email as well as your **email inbox** count against your Home directory quota, things can get quickly out of hand.

12.3.1 Are you over quota?

The command for determining your quota status is `quota`, the command

@:> `quota -v`

shows your disk use on all relevant filesystems. If you are over your soft limit, it tells you how much time remains before you will be punished.

12.3.2 Finding big ugly useless files

A good place to start your search for trash is:

@:> `findtrash`

which is an alias that executes: `du -sh .[a-zA-Z]* * |egrep -e '[0-9]M' | sort -n`. which (obviously) produces a listing of directories (including ".dot" directories) sorted by space consumed. Note this alias takes a minute to run
For more precise tasks, the command `du` (for "disk use") will produce a directory by directory listing of amount of space used in each of directory. With a `-k` flag it will give you the data in Kilobytes, on linux machines the `-h` flag gives you file and directory sizes in "human" readable units.

@:> `du -k |less`

Will give you the output page by page.

> NOTE: files with names like `core` or `core.nnnn` where `nnnn` are digits, are core dump files. These are the result of programs crashing. Unless you know what to do with these files, you can safely delete them – and you should as they tend to be **huge**.

Once you locate a directory with lots of stuff in it, you can use `ls` and `sort` to find the big files. The command

@:> `ls -s | sort -n`

prints each file precede by a number indicating it's size, the `sort -n` order the output numerically.

12.3.3 Sources of disk pollution

email pollution If you send lots of attachments back and forth in email – and particularly if those attachments have MS Word documents in them, then you can fill your disk quota quite rapidly. Because pine, by default, saves your outgoing messages with attachments, each time you save a message and respond, you might be storing two copies of the attachment. Word files tend to have a very low value to weight ratio.

browser detritus Browsers are another source of crap. Because they "cache" pages in order for you to be able to backup quickly, they too can fill your disk. Within each browser, you should be able to click on something labeled "clear disk cache".

.cxoffice Crossover office – which runs all of the Microsoft applications tends to store a whole lot of stuff in your `.cxoffice` directory. Since the directory starts with a ".dot" it is not obvious. You can delete this directory with impunity. If you switch to OpenOffice (See 9), it won't come back. If you run cxoffice, the directory will be recreated.

.Trash filemanagers (those pointy clicky applications that spare you the indignity of remembering `cd` and `ls`, don't actually "delete" files but rather move them to your `.Trash` folder. `.Trash` folders can exist in locations other than your home directory e.g. in `/data/commons`. You can erase these directories with impunity.

12.4 How can I compress files

Compressing files can save you a lot of space. Data files in particular can sometimes compress by as much as 90%. There are several different programs/algorithms for compressing files. Three common ones are:

1. standard UNIX compress
2. gnu zip (aka gzip)
3. bzip2

bzip2 is the fastest and most powerful, but it is not yet standard so if you need to uncompress your data later on an Amiga, bzip2 might be a bad choice. GNU zip, `gzip` offers a good compromise between universality and power. `compress` is not much used anymore.
To gnu compress a file type the following:

@:> `gzip` *`filename`*

The result will be a file called `filename.gz`. The original file will not be harmed – you must remember to remove `filename` yourself, otherwise file compression will just have resulted in additional disk usage.
There are two ways to uncompress a file – one deletes the compressed file and one does not.
To uncompress leaving the compressed file in tact, use `zcat`:

@:> `zcat` *`filename.gz > newfile`*

This will create a file called `newfile` leaving filename.gz in tact. This is a useful thing to do if your datafile is so large that it takes a long time to compress and you are uncompressing onto a temporary directory (See 12.1 for details).
To uncompress and remove the compressed file at the same time type:

@:> `gunzip` *`filename.gz`*

12.5 How can I compress a whole directory

A good way to compress an entire directory and all it's subdirectories is with the `tar` command. This creates a single "tape archive" file which can later be "un-tarred" to reproduce the original directory structure. The `tar` command has lots and lots of options – so check out the man page ok?
Here is a simple example:

@:> `tar -czvf dissertation.tgz ./dissertation`

Here `./dissertation` is a directory holding lots of files and perhaps many subdirectories as well. The original directory is not changed by this action. So if you goal is to save space, you will need to erase the uncompressed original. `dissertation.tgz` is a compressed `tar` file which can be moved, mailed, renamed, or erased just like any other file. But unlike any other file, it can also be *untarred* to reproduce the original directory – perhaps in a different place, on a different computer. Here is the command to *untar*:

@:> `tar -zxvf` *`dissertation.tgz`*

The file `dissertation.tgz` is not destroyed or changed in this process. However, untarring **can overwrite preexisting files**. The command above will recreate ./dissertation exactly as it was when you tarred it. If, therefore, you `tar -x ...` in the same directory as you tar -c'ed earlier **and** in between time, you improved but did not rename, some files in `dissertation` then the untarring will overwrite the new improved files with the old files of the same name. This is all fine of course, As long as you expect it.

12.6 backups are not archives nor are they complete backups

The purpose of tape backups is to allow us to recover from nuclear attack, earthquake, typhoon or firestorm. It is vital that users realize that system

backups are worthless for reproducing scientific results from a project more than a few days old. Backups are designed to allow us to recreate the filesystem as it (mostly) was on a very recent day. If a disk fails, we need to be able to replace the disk and its contents as of yesterday and that is the goal of system backups.

12.6.1 backups are not complete backups

In order to concentrate resources on backing up the **most critical** files, we employ several strategies of which a good scientist should be aware:

1. **We do not backup really large files**. Files larger than 500MB are not backed up – ever. They can't be anything but data or intermediate results. Either way, a serious scientist will be able to reproduce either easily.

2. **We do not backup .dta files** Stata .dta files are assumed to be intermediate work so we don't back them up. If you wish to follow the unhealthy practice of storing your raw data as .dta files, then you should compress them (and of course make a CD of it). We **do** backup .dta.gz files – as long as they are smaller than 500MB.

Besides saving resources, the system we have in place encourages good programming practice, reproducibility of results and thus good science. Knowing as you do now that large data files are vulnerable to hardware failure and terrorist attack, you have no doubt resolved to follow the practices outlined in Demog 213:

1. Store your programs in your home directory; store your data in `/data/commons`

2. Make sure your programs can always take you from raw data to your current results. Obviously practices such as creating new variables in a GUI and then creating other newer variables based on those variables – all without ever writing the code down in a way that allows it to be rerun – are right out.

3. Make an off line copy of your raw data (unless you just got if from IPUMS or equivalent) (See 8.7 for instructions on writing CDs).

4. Make an archive of your entire project as soon as it is complete. Burn it to a CD; send a copy to your mom.

12.6.2 Backups are NOT ARCHIVES

Reproducing results that you generated 2 years ago is a task for which system backups will be completely useless. In order to be able to reproduce your work you must make an archive of your project at the time it

is complete. That archive may contain files from all over the filesystem and, if you're clever, it will contain some notes on how you did what you did so you (or your RA or your biographer) can do it again after you are dead.
Archives are the responsibility of the only person who cares about and understands the work you do.
There are several choices regarding archiving. For projects requiring less that 750MB, CDROMs are a good choice. DVD's hold about 5GB and are just as easy to write as CDs. See 8 for instructions on how to use the various removable media.