Linux and Statistical Computing

Rodney Sparapani, PhD
Division of Biostatistics
Medical College of Wisconsin

June 4, 2015

Training Outline

New Hardware and Software

What is Linux?

Linux Resources

Brief History of Statistical Computing

Installing Software

SAS and Spreadsheets

Using X over the LAN/Internet

Transferring Files Through the Firewall

Keyboard Shortcuts

Emacs

If all else fails, read the instructions.
- Donald Knuth, a renowned computer scientist

New Linux Cluster runnning CentOS 7.1 (RHEL 7.1 clon

login to gouda.biostat.mcw.edu

Same username and password as
MCW email/MCWCorp/VPN/WiFi/D2L/etc.

Call the HelpDesk for password problems

CommVault backup and le recovery via the HelpDesk

PCs are not backed up: so save les on shared drives

Shares: no more Biostat domain or logging on period!

gouda: Master server

14TB of disk space and 128GB of RAM (typo xed)

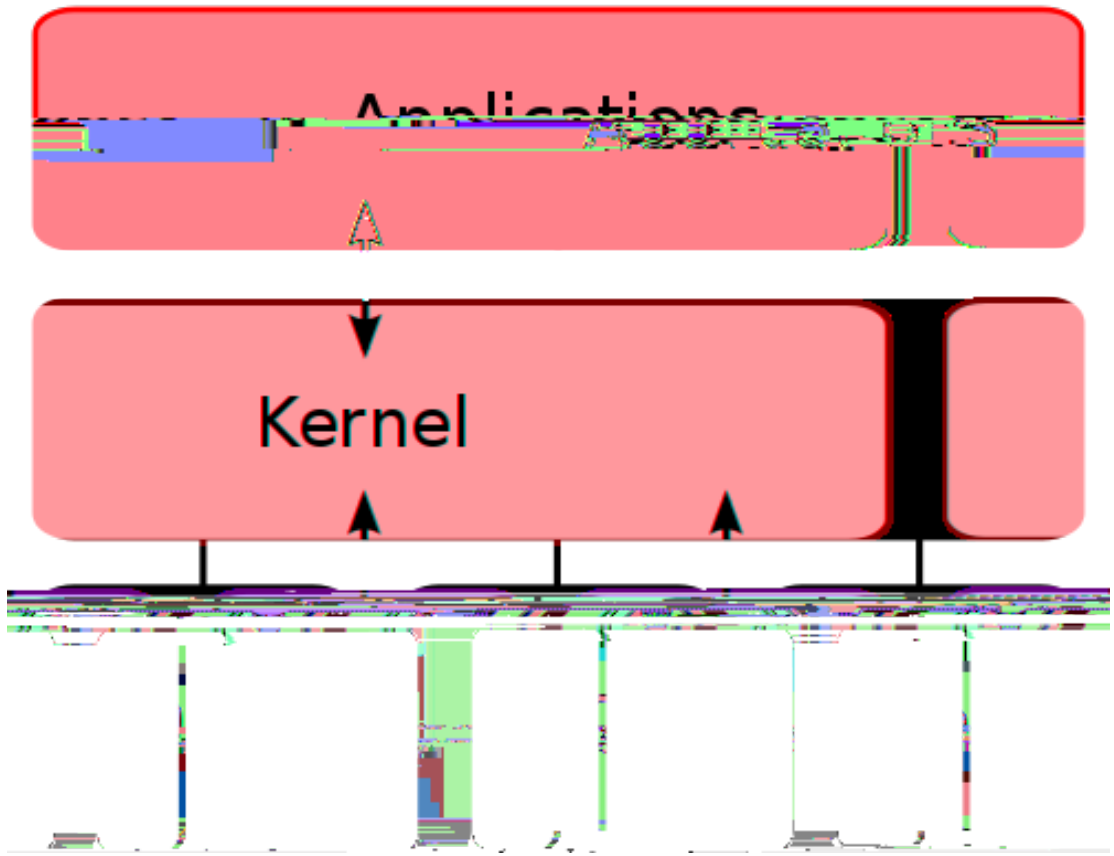
2 CPUs X 8 cores X 2 threads = 32 processes

cheddar and colby: Slaves ASAP

8 CPUs X 8 cores X 2 threads = 128 processes

Software Toolbox

What is Linux?



Linux Resources

man command, i.e. man man

info command, i.e. info info

http://www.mcw.edu/PCOR/Education.htm

The Linux Documentation Project http://www.tldp.org

The Linux Information Project http://www.linfo.org

The Stack Exchange http://stackexchange.com/sites

Wikipedia http://en.wikipedia.org

Just Google it

This presentation will be online

Computer Committee will create an online Linux FAQ

And a column in Datum: Ask the Cheese Wiz

So send us your questions and contributions

A Brief History of UNIX®

1969: AT&T Bell Labs starts work on UNIX

1970: open source UNIX provided for small fee

1972-3: Bell Labs develops C, re-writes UNIX in C

1973-8: DARPA invents TCP/IP network protocol

1978: University of California releases Berkeley Software Distribution (BSD) UNIX

1981-3: ARPANET goes TCP/IP (Internet)

1987: MIT/DEC release the X Window System

1992: AT&T files lawsuit preventing Free BSD

1994: Free BSD released; free, open source UNIX

1998-2000: SSL/SSH/VPN for Internet security

2010-3: Oracle buys Sun kills Sun Ray

A Brief History of GNU Linux

1984: Stallman creates GNU (GNU is Not Unix)
"complete, UNIX-compatible software system"
GNU General Public License (GPL)

1991-2: Linux kernel created/GPLed by Linus Torvalds

1994: Red Hat Commercial Linux is released

1997-9: GNOME GUI and package installers like yum
GNU Compiler Collection (GCC): C/C++/ FORTRAN

2002: Red Hat Enterprise Linux (RHEL) 2.1 is released

2003: Fedora project debuts, desktop/laptop-friendly

2006: CentOS, an RHEL clone, debuts

2009: X2Go 3.0 is released

2014: Red Hat funds CentOS' development
RHEL/CentOS 7.1 released based on Fedora 19

A Brief History of S and R

Late 70s: S, an intepreted, object-oriented statistical programming language developed by Bell Labs

1980: Bell Labs develops C++

Early 80s: the S language was licensed by AT&T for educational and commercial purposes

1997: GNU R software for UNIX/Linux, Windows & Mac
Comprehensive R Archive Network (CRAN)

2001: The Bioconductor Project launches to develop free software R packages for bioinformatics

2005: Rcpp package, seamless R and C++ integration

2014: R breaks into top 20 most popular languages
currently 12th on TIOBE Index (SAS is 24th)

2015: CRAN currently has 6700 R packages
The Bioconductor Project reaches 900 packages

A Brief History of SAS

1966-8: Anthony Barr develops SAS language

1968: Barr and James Goodnight develop ANOVA and multiple regression procedures for SAS

1973: John Sall joins the project

1976: SAS Institute is incorporated by Barr, Goodnight and Sall

1988: Modern SAS era begins
SAS v. 6 re-written in C for portability, adds support for UNIX, X, etc.

1993: SAS for Windows appears

1996: My GPL SAS macro library RASmacro begins

1999: SAS v. 8 released with support for Linux

2013: SAS v. 9.4 released

What are packages?

Package: a frequently overused term in free software

- Linux packages: binary distributions of free software
- Such as Extra Packages for Enterprise Linux (EPEL) https://fedoraproject.org/wiki/EPEL
- R packages available on CRAN http://cran.r-project.org
  - StatLib http://lib.stat.cmu.edu/R/CRAN
- More confusing: some R packages on EPEL like qtl
- LaTeX packages like beamer, graphicx, etc.
- Emacs Lisp Package Archive https://elpa.gnu.org
  - where AUCTeX can be found for example

Installing software

```
sudo yum install emacs # superuser only
compenv # print compiler environmental variables
mc ~/local/src/emacs
wget http://ftp.gnu.org/gnu/emacs/emacs-24.5.tar.gz
tar xzf emacs-24.5.tar.gz
cd emacs-24.5
# with GNU autotools
configure --prefix=~/local # configure --help
nohup make >& all.txt &
make install
```

Installing R packages
http://community.amstat.org/wisconsinchapter/blogs/rodney-sparapani/2013/04/05/installing-r-and-bioconductor-tips-updated-with-rgraphviz-info

SAS and spreadsheets

Use the Comma Separated Value format, i.e. .csv

Standard le format used with FORTRAN since late 60s

Use PROC IMPORT to read in

Use %_cimport SAS macro when PROC IMPORT fails

See the documentation at
/usr/local/sasmacro/_cimport.sas

Use PROC EXPORT to create

Using X over the LAN/Internet

The X Window System AKA X protocol is backwards

The X server is your PC and the client is the server

Fonts come from your PC: xlsfonts to list them

At Work or Home: use X2Go for X acceleration
via NX compression/caching of X data

MATE Desktop vs. Single Application

From Home: I use a single application due to latency

/usr/bin/xterm or /usr/local/bin/emacs

Transferring Files Through the Firewall From the Command Line

```
rho% mkdir ~/.ssh; chmod 700 ~/.ssh; ssh-keygen

rho% cd ~/.ssh; cp id_rsa.pub authorized_keys
gouda% scp -r USER@rho:.ssh .
rho% cd FROM
rho% tar cf FILE.tar FILES-OR-DIRECTORIES
rho% gzip FILE.tar
gouda% scp rho:FROM/FILE.tar.gz TO
gouda% cd TO; tar xzf FILE.tar.gz
```

Standard Keyboard Shortcuts

GNOME Human Interface Guidelines (HIG)
http://developer.gnome.org/hig-book/3.0/input-keyboard.html.en#standard-shortcuts

IBM Common User Access (CUA)

Cut	Sh-Delete
Copy	C-Insert
Paste	Sh-Insert

Emacs and Modes

gouda:~:$ emacs & # & runs in the background\verb
User init le: ~/.emacs
Global init le:
/usr/local/share/emacs/site-lisp/default.el
Debug inits: emacs --debug-init &
Start without inits: emacs --no-init-file &
List command line options: emacs --help
Emacs is a Lisp interpreter (.el) and byte-compiler (.elc)
Modes installed on gouda (written in Lisp)

Emacs and ESS on the web

http://www.mcw.edu/pcor/education/sas/xemacs.htm
http://ess.r-project.org
http://blog.revolutionanalytics.com/2014/03/emacs-ess-and-r-for-zombies.html
http://www.damtp.cam.ac.uk/user/eglen/ess11/index.html

Emacs Command Keys

Modi er Keys

C-KEY means hold down the Control key while pressing another KEY. For example, C-x means hold down Control while pressing x.

Sh-KEY means hold down the Shift key while pressing another KEY.

M-KEY means hold down the Meta key while pressing another KEY. On PC (Mac) keyboards, the Meta key is usually the Alt (Option) key. If you don't have a Meta key, you can press Esc, release, and then press KEY.

Execute an emacs command: M-x COMMAND Enter

M-x list-packages Enter

M-x list-fontsets Enter

C-u M-x list-fontsets Enter (C-u is called pre x arg)

Emacs Common Commands

Getting out of Trouble

- Cancel current command: C-g
- Exit Emacs: C-x C-c

File Commands

- Open a file or directory: C-x C-f
- Open a file/URL in the cursor: right mouse button
  M-x ffap-file-at-point
- Save a file: C-x C-s
- Refresh a file: F2 (ESS-g 0 G 0

Emacs Text Commands

Undo changes: C-x u

Cut region: Sh-Delete (CUA)
- Delete (local shortcut)
- C-Delete (local shortcut)

Copy region: C-Insert (CUA)
- Insert (local shortcut)
- M-w

Paste region: Sh-Insert (CUA)
- Middle mouse button
- C-y (yank command)

Select whole buffer as region: C-x h

Cut a rectangle of text: C-x r k
- M-x kill-rectangle

Emacs Text Commands (cont.)

Paste a Cut rectangle of text: C-x r y
M-x yank-rectangle

Fill paragraph: M-q

Emacs Search and Replace

Search forward: C-s (Return stops search)

Search backward: C-r (Return ...)

Search forward w/ wildcards: M-C-s (Return ...)

Search backward w/ wildcards: M-C-r (Return ...)

Query-replace: M-%
y for replace, n no replace, ! replace all, Return ...

Emacs Buffer Commands

Switch to the *shell* buffer: F8 (ESS)

Send Control character: C-q C-KEY

Split window for two views above/below: C-x 2

Unsplit window: C-x 1

Split window for two views left/right: C-x 3

Close a buffer: C-x k

List all buffers: C-x C-b

vi emulation on: M-Esc (local shortcut)

vi emulation toggle off/on: C-z
(from vi command mode to emacs and back)

New emacs window (frame): C-menu (local shortcut)
M-x make-frame-command

Emacs Cursor Commands

Move to beginning of line: C-a
Home

Move to end of line: C-e
End

Beginning of le: C-Home

End of le: C-End

Page up: PageUp
M-v

Page down: PageDown
C-v

Forward word: M-f

Backward word: M-b

Emacs Help Commands

Emacs tutorial: C-h t or F1 t
(F1 is a short-cut for C-h)

Emacs manuals: F1 i

Search for command: F1 a TEXT

Help for a key: F1 k KEY

Help for an Emacs variable: F1 v VARIABLE

Help for an Emacs function: F1 f COMMAND

Help for an Emacs mode: F1 m

Help for all keys currently available: F1 b

Help from a man page: M-x man

Help from info: M-x info

Emacs ESS[SAS] Commands

Batch submit a .sas program: F3

Switch to the .sas buffer: F4

Switch to the .log, refresh and search for errors: F5

Switch to the .lst and refresh: F6

Switch to the .txt and refresh: F7

Open a SAS dataset with PROC FSVIEW: F9

View a GSASFILE graph: F12

Emacs ESS[R] Commands

Start R: M-x R

Submit whole buffer: C-c C-b

Submit active region: C-c C-r

Submit current paragraph: C-c C-p

Submit current line: C-Enter

Retrieve previous typed command line: C-UpArrow
(in *R* and *shell* buffers)

Retrieve next typed command line: C-DownArrow

Assignment: < (generates <-)

Less than: << (generates <)