

## **My Checklist**

- Slides
- WB
- Flash cards
- Properties
- Page numbers
- 1<sup>st</sup> minute quiz
- Web Calendar summary
- Web book pages
- Commands
- Lab tested
- Microsoft and VMware webstore accounts made
- CIS Lab schedule published
- Census done
- cis90-students alias in /etc/aliases + newaliases command
- Welcome ready for mailing
- Lab 3 historical events ready for mailing
- 9V backup battery for microphone
- Backup slides, CCC info, handouts on flash drive

## Student checklist

- 1) Browse to the CIS 90 website Calendar page
  - <http://simms-teach.com>
  - Click CIS 90 link on left panel
  - Click Calendar link near top of content area
  - Locate today's lesson on the Calendar
- 2) Download the presentation slides for today's lesson for easier viewing
- 3) Click Enter virtual classroom to join CCC Confer session
- 4) Connect to Opus using Putty or ssh command



Instructor: **Rich Simms**

Dial-in: **888-450-4821**

Passcode: **761867**



Buzz



Carlos



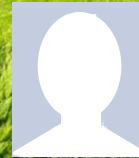
Cody



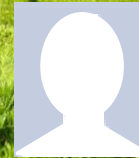
Elijah



Emily



Enrique C.



Enrique R.



Jon M.



Jon W.



Jordan



Joseph



JJ



Kiernan



Maria



Mathew



Mike C.



Michael F.



Mike M.



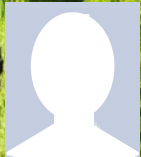
Miles



Nick L.



Nicholas T.



Patrick



Rebecca



Ricardo



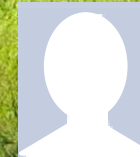
Robert



Ruth



Steve



Tess



Tim



Trevor



Troy



# Introductions and Credits



Jim Griffin

- Created this Linux course
- Created Opus and the CIS VLab
- Jim's site: <http://cabrillo.edu/~jgriffin/>



Rich Simms

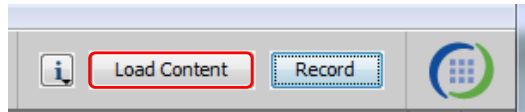
- HP Alumnus
- Started teaching this course in 2008 when Jim went on sabbatical
- Rich's site: <http://simms-teach.com>

And thanks to:

- John Govsky for many teaching best practices: e.g. the First Minute quizzes, the online forum, and the point grading system (<http://teacherjohn.com/>)

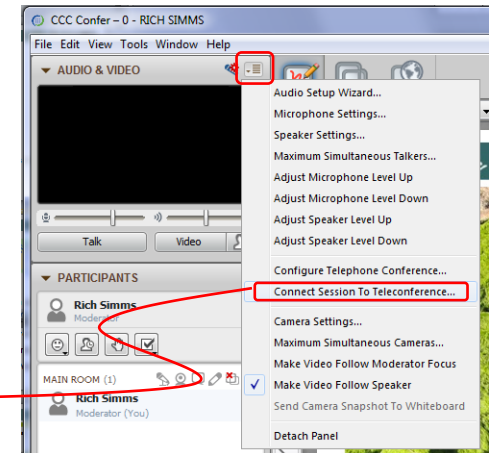
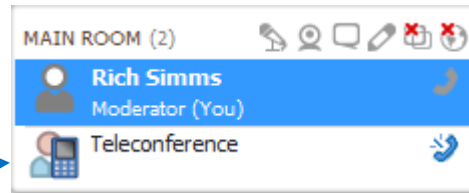


## [ ] Preload White Board with *cis\*lesson??\*-WB*

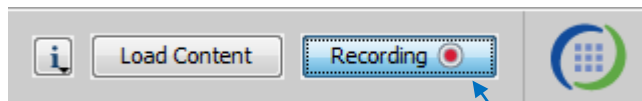


## [ ] Connect session to Teleconference

*Session now connected to teleconference*



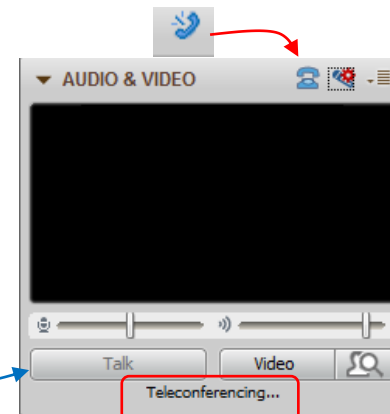
## [ ] Is recording on?



*Red dot means recording*

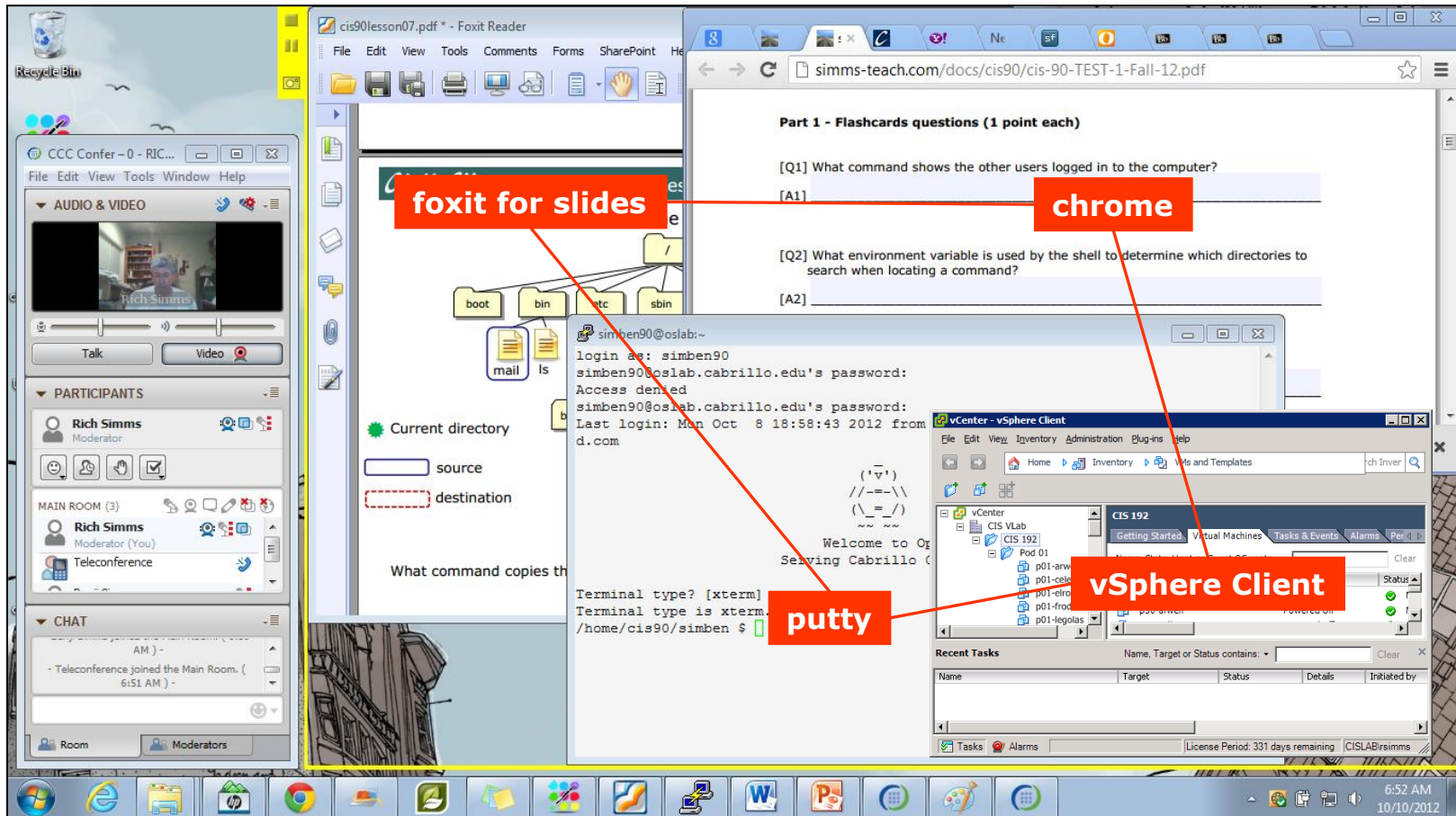
## [ ] Use teleconferencing, not mic

*Should be greyed out*





## [ ] layout and share apps

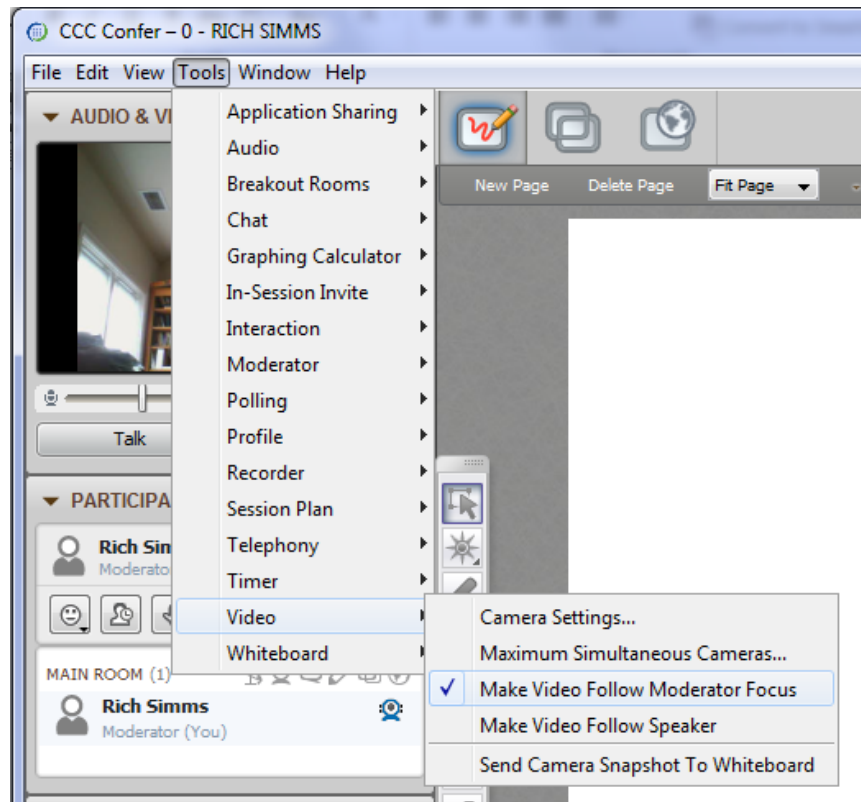




[ ] Video (webcam) optional

[ ] Follow moderator

[ ] Double-click on postages stamps



## Universal Fix for CCC Confer:

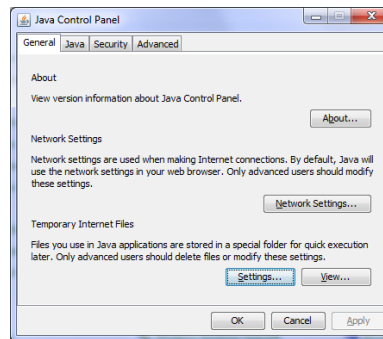
- 1) Shrink (500 MB) and delete Java cache
- 2) Uninstall and reinstall latest Java runtime



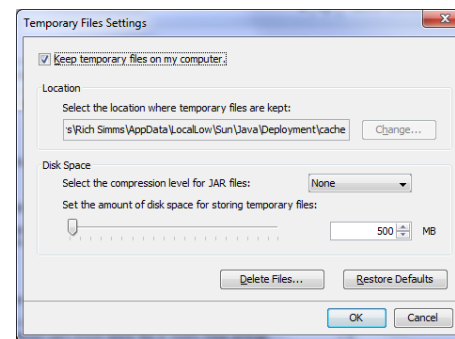
Control Panel (small icons)



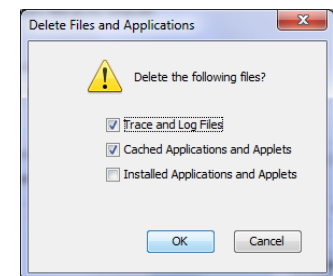
General Tab > Settings...



500MB cache size



Delete these



## Google Java download





## First Minute Quiz

Please answer these questions **in the order** shown:

Use CCC Confer White Board

**email answers to: [risimms@cabrillo.edu](mailto:risimms@cabrillo.edu)**

**(answers must be emailed within the first few minutes of class for credit)**

## Electronic Mail

Objectives	Agenda
<ul style="list-style-type: none"><li>• Learn how to use the UNIX communication tools write and mail.</li><li>• Overview on end-to-end email.</li></ul>	<ul style="list-style-type: none"><li>• Quiz</li><li>• Questions from last week</li><li>• Mini review</li><li>• Housekeeping</li><li>• Write</li><li>• Basic Mail</li><li>• More on Mail</li><li>• End-to-end email</li><li>• Other MUAs, MTAs, DA and AAs</li><li>• Wrap up</li></ul>

## Class Activity

```
( 'v' )  
//--\\  
( \_ = \_ / )  
~~  ~~
```

```
Welcome to Opus  
Serving Cabrillo College
```

If you haven't already,  
log into Opus

# Questions



# Questions?

Lesson material?

Labs? Tests?

How this course works?

- Graded work in home directories
- Answers in /home/cis90/answers

*Who questions much, shall learn much, and retain much.*

- Francis Bacon

*If you don't ask, you don't get.*

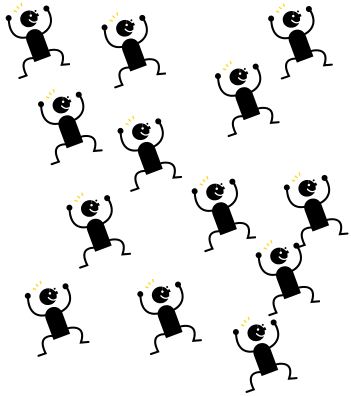
- Mahatma Gandhi

Chinese  
Proverb

他問一個問題，五分鐘是個傻子，他不問一個問題仍然是一個傻瓜永遠。

*He who asks a question is a fool for five minutes; he who does not ask a question remains a fool forever.*

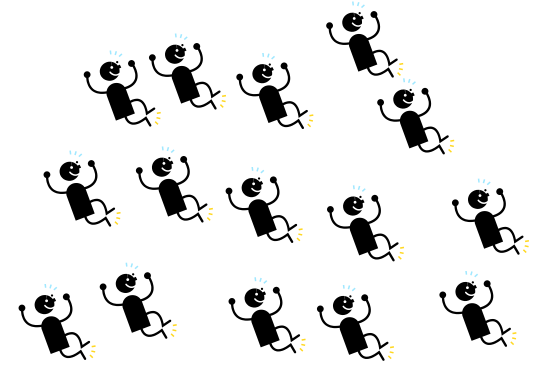
## Quiz 1 Results



No quiz  
submitted= 13



Submitted, but answers  
not in order= 3



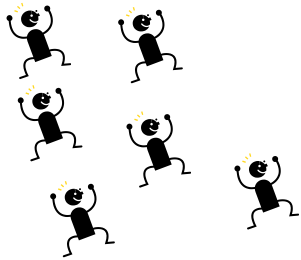
Submitted and answers in  
order = 15

Number of incorrect answers for each question

- 0 "1) What is the lowest level, inner-most component of a UNIX/Linux Operating System called?"
- 0 "2) What command shows the other users logged in to the computer?"
- 1 x "3) What part of UNIX/Linux is both a user interface and a programming language?"

To see answers use: **cat /home/cis90/answers/quiz01** on Opus

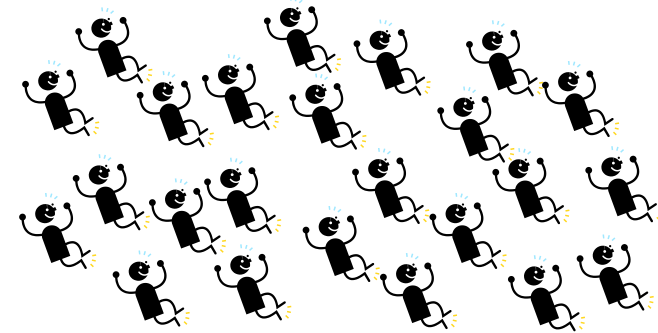
## Lab 1 Results



Number not  
submitted = 6



Number of submittals made  
after the deadline = 1



Number of submittals made  
before deadline = 24

### Number of incorrect answers for each question

0		"1) On Opus, what is the output of the hostname command?"
2	xx	"2) What is your EXACT shell prompt on Opus?"
5	xxxxxx	"3) What is your EXACT shell prompt on Son-of-Opus?"
3	xxxxx	"4) What is the name and version of the distro running on Kakenya?"
2	xx	"5) What is the output of the date command on Doc?"
7	xxxxxxx	"6) Log into Son-of-Opus, Razia and your personal Arwen. Which is running the sh (Bourne) shell?"
0		"7) On Catalina, which two users logged in from Sun-Hwa?"
2	xx	"8) Log into Doc and Son-of-Opus, are your UIDs the same in both sessions?"
0		"9) Start two concurrent login sessions on Kakenya. Does exiting one session exit you from the other?"
0		"10) Start two concurrent login sessions on Opus. What's the output of the tty command in each session?"
3	xxx	"11) Stay logged in twice on Opus. What distinguishes one session from the other in the who output?"
0		"12) Log into Razia, when (date and time) did Tybalt log in?"

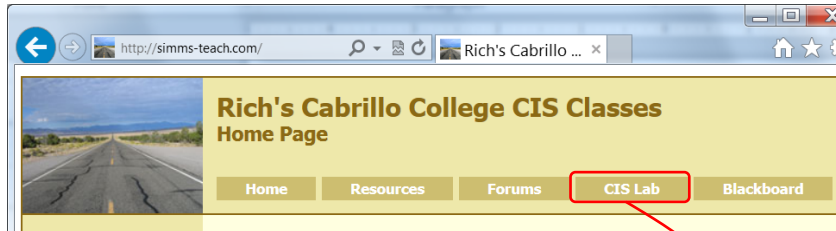
### Extra credit: Number of incorrect answers for each question

11	xxxxxxxxxxxx	"13) Log into Thabiti (as cis90), is Linux or UNIX installed?"
11	xxxxxxxxxxxx	"14) What is the name of the kernel (not the distro) running on Baby-Opus?"
19	xxxxxxxxxxxxxxxxxxxx	"15) What does the tty command output on your Arwen's tty5 virtual console?"

To see answers use: **cat /home/cis90/answers/lab01** on Opus  
To see graded lab use: **cat lab01.graded** on Opus

# For those who didn't submit Lab 1

*If you would like some help getting started, come see us in the CIS Lab.*



*Leandro and Geoff are both CIS 90 Alumni. Michael is the other Linux instructor.*







# Housekeeping



## Lab 2 due tonight

- Use **history -a** before using **submit**.
  - ❖ If you neglect to do this the history snapshot you send me to grade will not have the latest commands you issued.
- Submit as many times as you wish up to 11:59PM
- No credit for late work. Submit what you have for partial credit if you run out of time.
- You can optionally use the **verify** command to see what you submitted for grading.
  - ❖ To grade, I will check your submitted history to see if you used all the commands asked for in Lab 2 as well as your answers to the three questions.

## Forum

The screenshot shows a web browser window with the URL `oslab.cishawks.net/forum/viewforum.php?f=93&sid=4f90a29022aeab31bf623a55cf7a6b51`. The forum header is blue with the phpBB logo and the text "Cabrillo College: Computer and Information Systems". A search bar is on the right. On the left, there are navigation links for "Board", "CIS 90", "Forum rules", "NEWTOPICS", "ANNOUNCEMENTS", and "TOPICS". A blue box with a white background and a blue border contains the following text:

- Next week is the 1<sup>st</sup> five post deadline! (worth 20 points)
- Only your posts in the **CIS 90** forum will earn points
- Make sure your username is your **full first** and **last** name, separated by a space, so you get credit for your posts

Below the list, the text reads: *Email the instructor for username changes or to reset your password*.

At the bottom of the forum, there is a table of topics:

Topic	Replies	Views	Author	Date
Using step for step to transfer files	1	8	by Robert Lemon	Tue Feb 11, 2014 11:17 am
Using virtualbox for fun and education	3	46	by Robert Lemon	Tue Feb 11, 2014 11:15 am
Microsoft and VMware academic webstores	0	8	by Rich Simms	

## Grades posted on website

<http://simms-teach.com/cis90grades.php>

*Grades for Quiz 1, Lab 1 and extra credit have been posted on the CIS 90 website.*

*Send me your survey to get your LOR code name.*

At the end of the term I'll add up all your points and assign you a grade using this table:

Current Progress

Code Name	Grading Choice	Quizzes & Tests												Forum				Labs										Project	Extra Credit	Total	Grade
		Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8	Q9	Q10	T1	T2	T3	F1	F2	F3	F4	L1	L2	L3	L4	L5	L6	L7	L8	L9				
Max Points		3	3	3	3	3	3	3	3	3	30	30	30	20	20	20	20	30	30	30	30	30	30	30	30	30	30	60	90	560	
adaldrida	grade	0																28											1		
alatar	grade																	26											2		
anborn	grade	3																30											3		
aragorn	grade	0																28											3		
arwen	grade	3																28											2		
beregond	grade																														
bilbo	grade	2																24													
celebrian	grade																	30											6		
dori	grade	3																28											1		
dwalin	grade	3																29											2		
elrond	grade																												3		
eomer	grade	3																29											3		
faramir	grade	3																28											2		
frodo	grade																	29											6		
gimli	P/NP																														
gwaihir	grade	3																30											6		
loeth	grade	3																28											5		
legolas	grade	0																30											2		
marhari	grade																	26													
orome	grade																	30													
pallando	grade																														
pippen	grade	3																30													
quickbeam	grade	3																28											6		
rian	grade																														
samwise	grade	3																26											3		
shadowfax	grade																	0													
strider	grade	3																30											3		
theoden	grade																	27											3		
treebeard	grade	3																30											6		
tulkas	grade																	30											2		
ulmo	grade	3																													

Percentage	Total Points	Letter Grade	Pass/No Pass
90% or higher	504 or higher	A	Pass
80% to 89.9%	448 to 503	B	Pass
70% to 79.9%	392 to 447	C	Pass
60% to 69.9%	336 to 391	D	No pass
0% to 59.9%	0 to 335	F	No pass




## Extra Credit

SS  
SS

se. Another 90 points is available from **extra credit** assignments. Students c  
erall progress on the chart below. Contact the instructor by email with any que

				Forum				Labs										Final
Q	T1	T2	T3	F1	F2	F3	F4	L1	L2	L3	L4	L5	L6	L7	L8	L9	L10	Project
3	30	30	30	20	20	20	20	30	30	30	30	30	30	30	30	30	30	60
								30										
								30										

*Note the caps  
on extra credit.*



### Rich's Cabrillo College CIS Classes

#### CIS 90 Extra Credit

[Home](#)
[Resources](#)
[Forums](#)
[CIS Lab](#)
[CTC](#)

[Login](#)
[Flashcards](#)
[Admin](#)

[CIS 90](#)  
[Previous Classes](#)

95 days till term  
ends!

[Cabrillo College](#)  
[Web Advisor](#)  
[CCC Confer](#)  
[Static IPs](#)  
[Quick Ref](#)  
[VM Repairs](#)  
[GAH!](#)

### CIS 90 Extra Credit


[Course Home](#) [Grades](#)

#### General Options

Any combination of the following can be done to earn extra credit up to the maximum amount shown on the Grades page:

- **Web site content review** - The first person to email the instructor pointing out an error or typo on this website will get one point of extra credit per content error found. This includes any errors found on the instructor's downloaded materials that have been covered in class. It does not include lesson PowerPoints or Labs that have not yet been covered in class but are pre-published on the website. (Up to 20 points total)
- **Develop new Howtos** - Investigate and develop a Howto on a new topic area you are interested in. At the Instructor's discretion and your permission, these Howtos will be published on this web site on the Resources page. Make a proposal first to the instructor on the topic area and to determine the amount of extra credit. Submittals must follow the format of the instructor's Howtos on the Resources web page and be web publishable. (Up to 20 points per Howto)
- **Optional activities in lab assignments** - Some of the lab assignments will have optional activities that can be worked for extra credit.
- **Lab assignments** - Some courses may have one or more extra credit labs. Check the Calendar web page. (Point amount varies)

## Extra Credit Howtos



### Rich's Cabrillo College CIS Classes Resources

[Home](#)
[Resources](#)
[Forums](#)
[CIS Lab](#)
[CTC](#)

[Login](#)
[Flashcards](#)
[Admin](#)

[CIS 90](#)
[CIS 192](#)
[Previous Classes](#)

101 days till term ends!

[Cabrillo College](#)
[Static IPs](#)

#### Links

##### Instructors

- [Programming Master Ed](#)
- [Linux Master Jim](#)
- [Web Master John](#)
- [Network Master Gerlinde](#)
- [Network Master Rick](#)

##### Getting Linux

- [Linux ISOs](#)
- [Kernels](#)
- [RPMs](#)

##### Tools and Software

- [Apache](#)
- [Bastille](#)
- [cygwin](#)
- [DIAG](#)
- [diagnostics](#)
- [DOS boot disks](#)
- [John the Ripper](#)
- [MSDN Academic Alliance](#)
- [Netfilter](#)
- [Putty SSH Tools](#)
- [Tripwire](#)
- [VMware Server](#)
- [Wireshark](#)

##### Standards

- [IETF \(RFCs\)](#)
- [IEEE](#)

##### Clubs

- [GNU Linux Users Group](#)

##### Departments

- [CNSA](#)
- [CIS](#)
- [CS](#)

##### Crib Sheets

- [Ollie Wright \(CIS 90\)](#)

##### Documentation

- [TLDP](#)
- [LINFO](#)
- [Commands](#)
- [Summary](#)
- [vi summary](#)


##### Howtos

- [email](#)
- [DNS](#)
- [Ethernet \(NIC drivers\)](#)
- [NIS](#)
- [PPP](#)
- [NFS](#)


##### Student Howtos

- [Marc Romansky \(Accessing VMware remotely via Linksys Router\)](#)
- [Marc Romansky \(Accessing VMware with PuTTY\)](#)
- [Marcos Valdebenito \(VirtualBox\)](#)
- [Michael Wicherski \(Permissions\)](#)
- [Michael Wicherski \(/bin/mail\)](#)

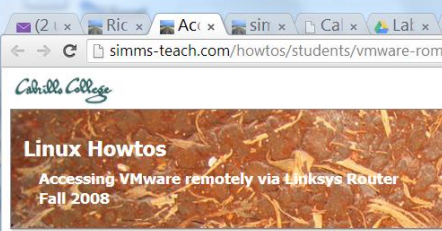
If you have a strong interest in a topic write a Howto on it to share what you've learned and earn some extra credit at the same time



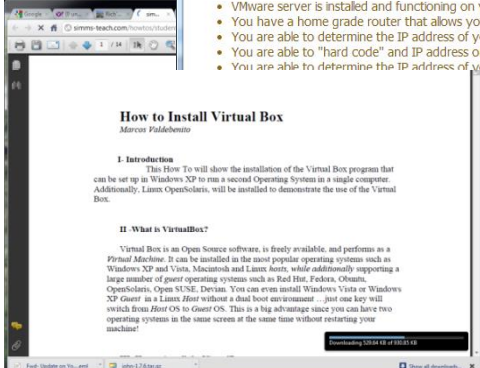
**Wifi Penetration**  
Wireless Communication and Computer/Network Forensics



**Understanding /bin/mail :  
Lost Student versus /bin/mail**

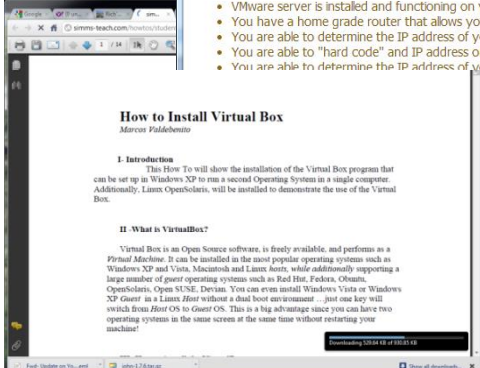


**Linux Howtos**  
Accessing VMware remotely via Linksys Router  
Fall 2008



**Assumptions**

- VMware server is installed and functioning on your home system.
- You have a home grade router that allows you to configure port forward
- You are able to determine the IP address of your home based system
- You are able to "hard code" and IP address on your VMware host system
- You are able to determine the IP address of your Virtual Machine



**How to Install Virtual Box**  
Marcos Valdebenito

**I. Introduction**  
This How To will show the installation of the Virtual Box program that can be set up in Windows XP to run a second Operating System in a single computer. Additionally, Linux OpenSUSE, will be installed to demonstrate the use of the Virtual Box.

**II. What is VirtualBox?**  
Virtual Box is an Open Source software, is freely available, and performs as a *Virtual Machine*. It can be installed in the most popular operating systems such as Windows XP and Vista, Macintosh and Linux hosts, while additionally supporting a large number of guest operating systems such as Red Hat, Fedora, Ubuntu, OpenSUSE, Open SUSE, Debian. You can even install Windows Vista or Windows XP Guest in a Linux host without a dual boot environment... just one key will switch from Host OS to Guest OS. This is a big advantage since you can have two operating systems in the same screen at the same time without restarting your machine!

## Software for CIS students

Rich's Cabrillo College CIS Classes Resources

Home Resources Forums CIS Lab Blackboard

Login  
Flashcards  
Admin

CIS 90  
CIS 192  
Previous Classes

100 days till term ends!

Cabrillo College  
Web Advisor  
Commands and Files

VLab RDP file  
CIS 90 VLab VM Assignments  
CIS 192 VLab Pod Assignments  
RIP Dennis Ritchie

**Links**

**Instructors**

- Linux Master Jim
- Programming Master Ed
- Network Master Gerlinde
- Network Master Rick
- Web Master John
- Windows Master Gary

**Clubs**

- GNU Linux Users Group

**Departments**

- CNSA
- CIS
- CS

**Crib Sheets**

- Ollie Wright (CIS 90)

**Documentation**

- TLDP
- LINFO
- UNIX Rosetta Stone

**Getting Linux/UNIX**

- Linux ISOs
- Kernels
- RPMs (rpmfind)
- RPMs (phone)
- OpenSolaris

**Tools and Software**

- Apache
- Bastille
- CoRD
- cygwin
- DOS boot disks
- Dynamips/Dynagen
- John the Ripper
- Netfilter
- Putty SSH Tools
- Quagga routing suite
- Tripwire
- Wireshark

**Commands**

- Practical
- Summary
- Useful
- vi summary
- vi cheat sheet

**Howtos**

- HowtoForge
- email
- DNS
- Ethernet (NIC drivers)
- NFS
- NIS
- PPP
- Putty SSH Keys
- sed

**Student Howtos**

- Logging into Opus from a Mac by Laura Sreckovic
- Install and DualBoot into Microsoft Windows 7 and Linux Ubuntu by Richie Fou
- Making an ethernet cable by Michael George
- Home VM access via Linksys router by Marc Romansky
- Putty to VMs by Marc Romansky
- Installing VirtualBox by Marcos Valdebenito
- Linux Permissions by Michael Wicherski

**e-academy sites for CIS students**

- MSDN Academic Alliance
- VMware e-academy

**Virtualization**

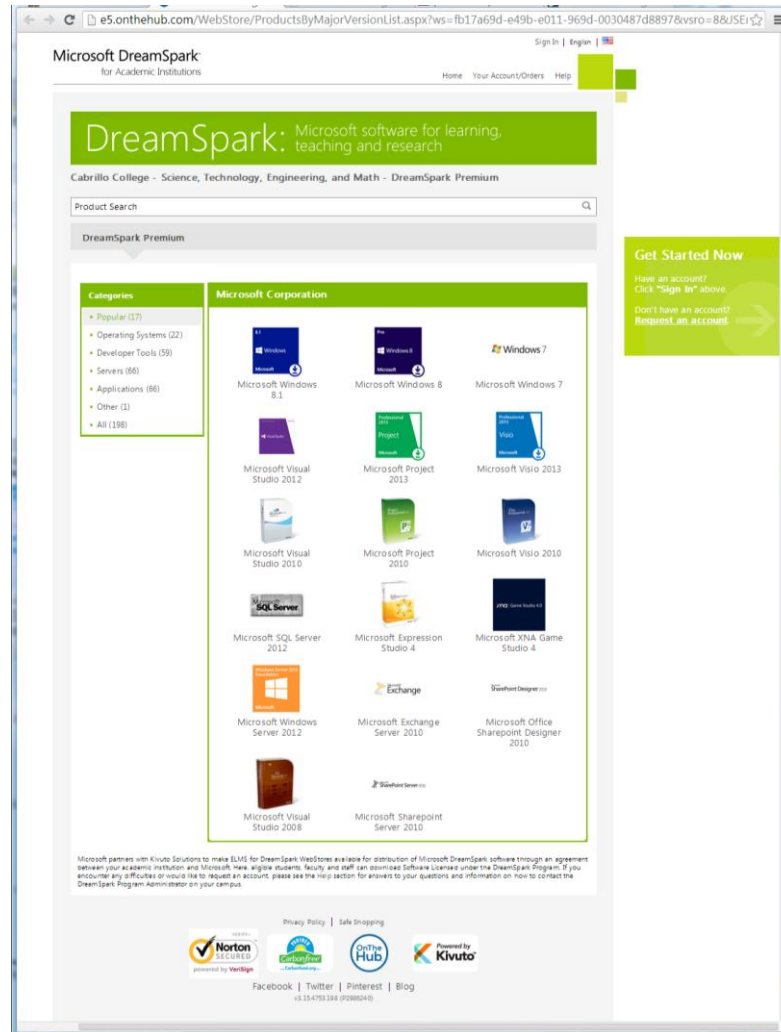
- VirtualBox
- VMware ESXi and vSphere client



*How to obtain Microsoft and VMware software for academic use*



# MSDN for CIS Students



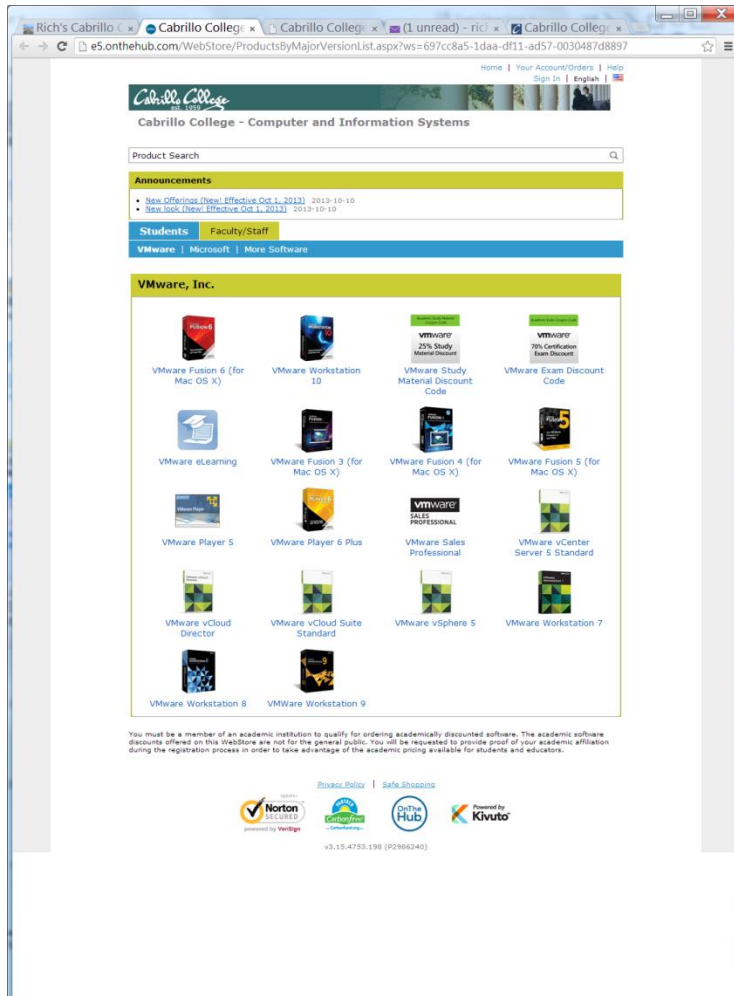
*Accounts for students enrolled in CIS 90 have been created using your WebAdvisor email addresses.*

*Link is on website Resources page in Tools and Software section*

*Happy downloading!*



# VMware Software for CIS Students



*Accounts for students enrolled in CIS 90 have been created using your WebAdvisor email addresses.*

*Link is on website Resources page in Tools and Software section*

*Happy downloading!*

# Spring Cybersession 2014 Online Classes (March 24th–May 24th)

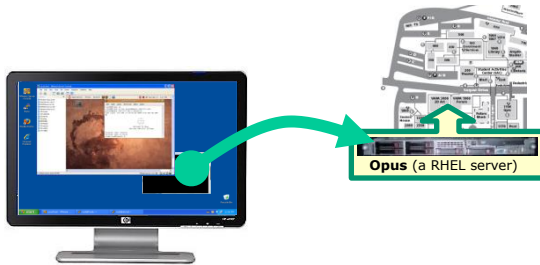
<http://www.cabrillo.edu/services/disted/online.html>

The screenshot shows a web browser window displaying the Cabrillo College website. The browser's address bar shows the URL [www.cabrillo.edu/services/disted/online.html](http://www.cabrillo.edu/services/disted/online.html). The website has a dark blue header with the Cabrillo College logo and navigation links: Campus Life, Learning Resources, Student Services, and Directories A-Z. Below the header is a horizontal menu with links: ABOUT, ACADEMICS/CAREERS, ADMISSIONS, CLASS SCHEDULES, REGISTRATION, WEBADVISOR, and a search bar labeled "Most sought pages ...".

The main content area is divided into two columns. The left column is titled "DISTANCE EDUCATION" and features a Blackboard Login section with a "Bb" logo and links for "Trouble logging in?", "Blackboard Orientation", and "Browser check, Java, and popup troubleshooting". Below this is an "iTunesU" section and a "Resources for Students" list including "How to Apply", "Online Orientation to Cabrillo", "Online Advising", "WebAdvisor", "Adding a Course and Waitlists", and "Dropping a Course".

The right column features a large photograph of a brick building with white columns. Below the photo is the heading "Distance Education Schedule". The text below the heading states: "REGISTRATION IS NOW OPEN for Spring 2014 Cybersession (8 weeks) begins Monday, March 24, 2014. See the class list below." It then asks, "Is an on-campus orientation, prerequisite, or assessment required?" and provides instructions to check the "Schedule of Classes" and click on the instructor's name in the list of classes below. It also mentions visiting the "Assessment Center's webpage" for the testing schedule. A link for "Need more information? -or- Ready to enroll? Visit Resources for Students for links to IMPORTANT information." is provided. At the bottom of the right column, the heading "Spring Cybersession 2014 Online Classes (March 24th-May 24th)" is displayed, followed by a dropdown menu labeled "Courses and Descriptions by Department".

# Subtle Stuff



Putty to:  
***rsimms@oslab.cis.cabrillo.edu***  
vs ***oslab.cis.cabrillo.edu***

Basic options for your PuTTY session

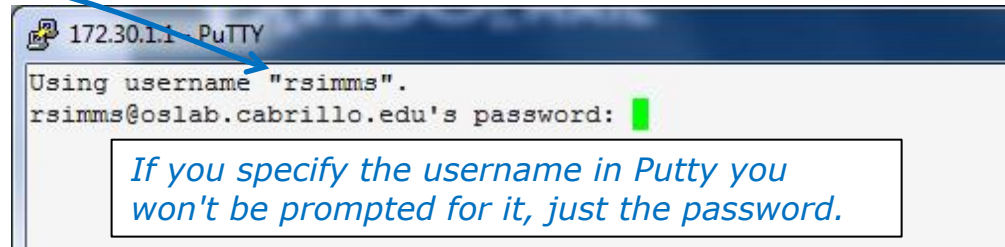
Specify the destination you want to connect to

Host Name (or IP address) Port

rsimms@oslab.cis.cabrillo.edu 2220

Connection type:

☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial



Basic options for your PuTTY session

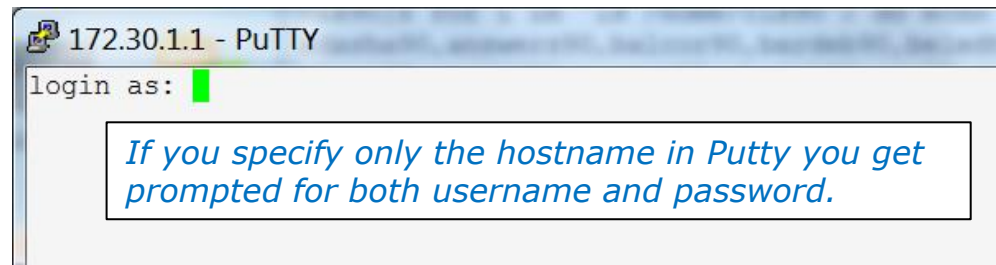
Specify the destination you want to connect to

Host Name (or IP address) Port

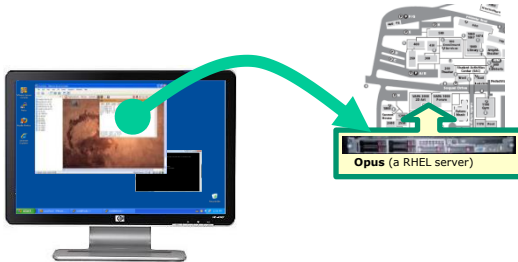
oslab.cis.cabrillo.edu 2220

Connection type:

☐ Raw ☐ Telnet ☐ Rlogin ☒ SSH ☐ Serial



*Tip: Use the Putty "Saved Sessions" for your Opus connection. Then you don't have to type in the username, hostname and port number each time you connect to Opus.*



***ssh thabiti vs ssh cis90@thabiti***

```
simben90@oslab:~
/home/cis90/simben $ ssh thabiti
Password:
Password:
Password:
Permission denied (gssapi-keyex,gssapi-with-mic,publickey,keyboard-interactive).
/home/cis90/simben $
```

*Benji is logged in as simben90 on Opus uses  
ssh command to log into Thabiti*



```
simben90@oslab:~
/home/cis90/simben $ ssh cis90@thabiti
Password:
Last login: Tue Sep 10 23:50:55 2013 from cataline.cis.ca
Oracle Corporation      SunOS 5.11      11.1      September 2012
cis90@thabiti:~$
```

*Benji is logged in as simben90 on Opus uses  
ssh command to log into Thabiti*




*If you don't specify the username on the **ssh** command it will use the username you are currently logged in as. This account may not exist on the remote system!*

## type and man caveats

**Usually**, to find the location of a command on your path, use the **type** command:

```
/home/cis90/simben $ type hostname  
hostname is /bin/hostname
```

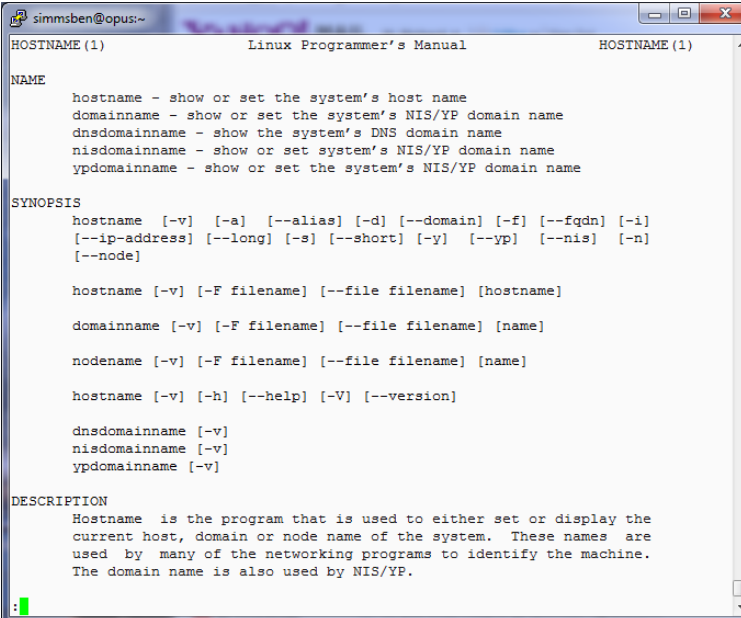
 *The hostname program  
file is in the /bin directory*



## type and man caveats

**Usually**, to find the manual page for a command, use the **man** command:

```
/home/cis90/simben $ man hostname
```



```
simmsben@opus:~
HOSTNAME(1)                                Linux Programmer's Manual            HOSTNAME(1)

NAME
hostname - show or set the system's host name
domainname - show or set the system's NIS/YP domain name
dnsdomainname - show the system's DNS domain name
nisdomainname - show or set system's NIS/YP domain name
ypdomainname - show or set the system's NIS/YP domain name

SYNOPSIS
hostname [-v] [-a] [--alias] [-d] [--domain] [-f] [--fqdn] [-i]
[--ip-address] [--long] [-s] [--short] [-y] [--yp] [--nis] [-n]
[--node]

hostname [-v] [-F filename] [--file filename] [hostname]
domainname [-v] [-F filename] [--file filename] [name]
nodename [-v] [-F filename] [--file filename] [name]
hostname [-v] [-h] [--help] [-V] [--version]

dnsdomainname [-v]
nisdomainname [-v]
ypdomainname [-v]

DESCRIPTION
Hostname is the program that is used to either set or display the
current host, domain or node name of the system. These names are
used by many of the networking programs to identify the machine.
The domain name is also used by NIS/YP.
```

## Command Review

*However,*

*sometimes you may get something different than expected with the **type** and **man** commands*

## type and man caveats

```
/home/cis90/simmsben $ type ls  
ls is aliased to `ls --color=tty`
```

*If the command is an alias (which we will learn about later) the type command by default doesn't show where the command resides on the path*

```
/home/cis90/simmsben $ type -a ls  
ls is aliased to `ls --color=tty`  
ls is /bin/ls
```

*To get around that use the **-a** option*

 *The ls program file resides in the /bin directory*

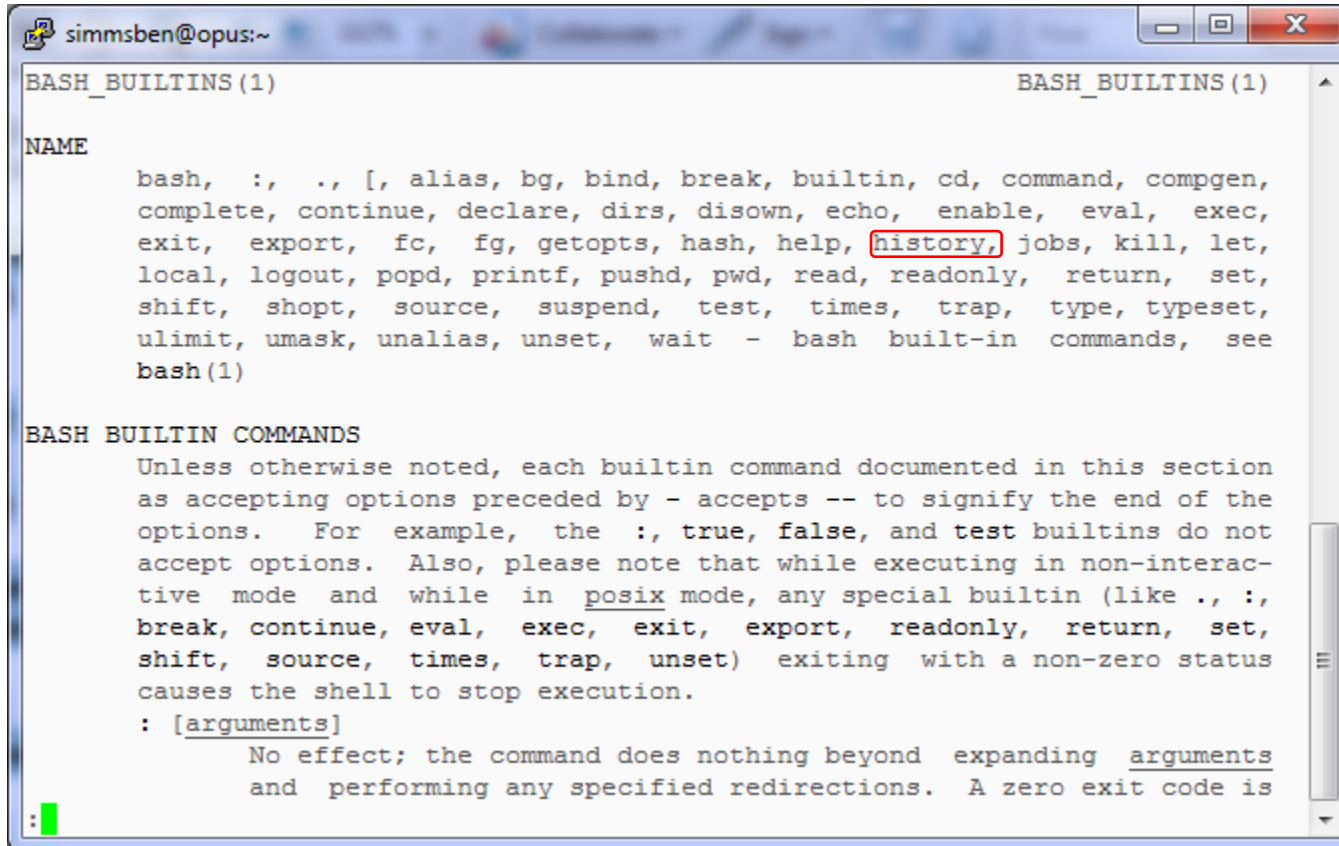
*The **ls** command is aliased, use the **-a** option on the **type** command to find where the command resides on the path*

## type and man caveats

```
/home/cis90/simben $ type history
history is a shell builtin
```

*The **history** command is built into the shell and does not have its own program file*

```
/home/cis90/simben $ man history
```



The screenshot shows a terminal window titled 'simmsben@opus:~'. The first command executed is 'type history', which outputs 'history is a shell builtin'. The second command is 'man history', which displays the 'BASH\_BUILTINS(1)' manual page. In the manual page, the word 'history' is highlighted with a red box in the list of builtins. Below the list, the 'BASH BUILTIN COMMANDS' section is visible, starting with a description of how builtin commands are documented and followed by the entry for the ':' command.

```
BASH_BUILTINS(1)
NAME
    bash, :, ., [, alias, bg, bind, break, builtin, cd, command, compgen,
    complete, continue, declare, dirs, disown, echo, enable, eval, exec,
    exit, export, fc, fg, getopts, hash, help, history, jobs, kill, let,
    local, logout, popd, printf, pushd, pwd, read, readonly, return, set,
    shift, shopt, source, suspend, test, times, trap, type, typeset,
    ulimit, umask, unalias, unset, wait - bash built-in commands, see
    bash(1)

BASH BUILTIN COMMANDS
    Unless otherwise noted, each builtin command documented in this section
    as accepting options preceded by - accepts -- to signify the end of the
    options. For example, the :, true, false, and test builtins do not
    accept options. Also, please note that while executing in non-interac-
    tive mode and while in posix mode, any special builtin (like ., :,
    break, continue, eval, exec, exit, export, readonly, return, set,
    shift, source, times, trap, unset) exiting with a non-zero status
    causes the shell to stop execution.

    : [arguments]
        No effect; the command does nothing beyond expanding arguments
        and performing any specified redirections. A zero exit code is
```

*The **history** command does not have its own man page either!*

*... but it is included in the man page for bash builtins*

# Mini Review

## Expectation Check

Skills you should be comfortable performing

- Navigating the course website: [simms-teach.com](http://simms-teach.com)
- Entering the CCC Confer Virtual Classroom
- Reviewing video archives of the lessons
- Downloading and searching lessons PDFs
- Checking your current grade status
- Checking when assignments are due
- Checking when quizzes and tests will be held
- Finding the answers for graded labs and quizzes
  
- Logging into Opus from home or school using SSH
- Logging into Arwen or other VMs from Opus using SSH
- Using Arwen's graphical desktop via VLab
- Changing Virtual (TTY) Terminals on Arwen
- Reading and making forum posts
  
- Parsing any shell command
- Getting documentation on any command
- Identify the four key components of the UNIX/Linux architecture
- Identify the six steps the shell does for every command
- Temporarily change your shell prompt
- Set and show values of shell variables

*If you have any questions on these skills, post a question on the forum!*



### Expectation Check

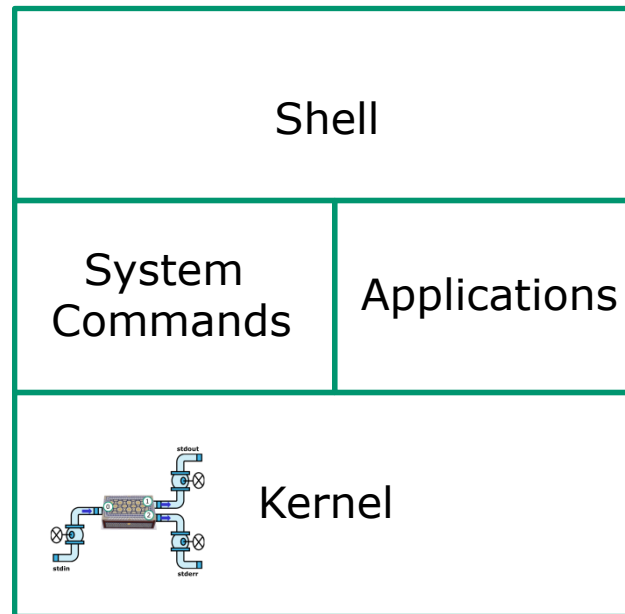
Commands you should understand and be comfortable using

Lesson/Lab 1		Lesson/Lab 2	
Commands	Files & Directories	Commands	Files & Directories
cal clear date exit history hostname id ps ssh uname tty who who am i	/etc/issue /etc/*-release	apropos banner bash bc cat cd echo env file finger info file ls passwd set type man whatis	/bin /usr/bin /sbin /usr/sbin /etc/passwd /etc/shadow

*If you have any questions on these commands, post a question on the forum!*

# Key components of the Linux/UNIX architecture

*Users interact with the shell to run commands*



*Commands such as ls, cal, date, tty, id, who, etc.*

*Web servers, dhcp servers, word processors, etc.*



*The kernel manages processes, memory, file system, and the network stack and interacts with all the hardware components*

The **ssh** command is used to login as *username* on a remote UNIX/Linux system named *hostname* via a firewall *port* as follows:

**ssh -p *port* *username*@*hostname***

For example:

```
ssh -p 2220 simben90@son-of-opus.simms-teach.com
```

```
ssh -p 22 cis90@thabiti.cishawks.net
```

Typing shortcuts:

- If the port is 22, then it does not need to be specified
- If the username is the same on the remote system it can be left off
- If domain suffixes are automatically added they can be left off

For example Benji could use either command below to log into Doc from Opus:

```
ssh -p 22 simben90@doc.cishawks.net
```

```
ssh doc
```

## My favorite ice cream shop



Source: [http://attractions.uptake.com/blog/files/2008/10/dsc\\_0002.jpg](http://attractions.uptake.com/blog/files/2008/10/dsc_0002.jpg)

## Activity

1) What command could be used on Opus to log into this remote system:

**hostname:** sun-hwa-iii.cishawks.net

**username:** *same as your Opus username*

**port:** 22

*Write your command in the chat window*

2) Log into sun-hwa-iii and run the **icecream** command.

*Copy the output of that command into the chat window.*

## Practice Test Questions

On Sun-Hwa-III, is the **icecream** command on your path?

*Write your answer in the chat window*



## Practice Test Questions

On Sun-Hwa-III, is the **icecream** command on your path? If so what directory is it in?

*If the shell can find it when you run it then it is on your path!*

```
[simben90@sun-hwa-iii ~]$ icecream  
first=Benji  
uid=1201  
index=1  
Benji gets Marianne's Banana ice cream today!
```

*Use the **type** command to find the first directory on your path containing the command*

```
[simben90@sun-hwa-iii ~]$ type icecream  
icecream is /usr/local/sbin/icecream  
[simben90@sun-hwa-iii ~]$
```

**Answer:** YES, the **icecream** command is in the `/usr/local/sbin` directory

## Practice Test Questions

On Sun-Hwa-III, what kind of a file is the **icecream** command?

*Write your answer in the chat window*

## Practice Test Questions

On Sun-Hwa-III, what kind of file is the **icecream** command?

*Use the **file** command to probe and get extended file type information*

```
[simben90@sun-hwa-iii ~]$ file /usr/local/sbin/icecream  
/usr/local/sbin/icecream: Bourne-Again shell script, ASCII text  
executable  
[simben90@sun-hwa-iii ~]$
```

**Answer:** BASH shell script

## Practice Test Questions

On Sun-Hwa-III, how many directories does the shell have to search to locate the **icecream** command on your path?

*Write your answer in the chat window*

## Practice Test Questions

On Sun-Hwa-III, how many directories does the shell have to search to locate the **icecream** command on your path?

*Echo the PATH environment variable to see the order of the directories on the path*

```
[simben90@sun-hwa-iii ~]$ echo $PATH  
/usr/local/bin:/bin:/usr/bin:/usr/local/sbin:/usr/sbin:  
/home/cis90/simben/.local/bin:/home/cis90/simben/bin
```

*Note the colon character : is used to delimit one directory from the next*

**Answer:** Four

- 1) /usr/local/bin
- 2) /bin
- 3) /usr/bin
- 4) /usr/local/sbin ←

## Practice Test Questions

Is **icecream** a standard UNIX command?

*Write your answer in the chat window*



## Practice Test Questions

On Sun-Hwa-III, is **icecream** a standard UNIX command?

*Use the **man** command to see if there is any documentation on **icecream***

```
[simben90@sun-hwa-iii ~]$ man icecream  
No manual entry for icecream
```

**Answer:** NO

## Practice Test Questions

Is Sun-Hwa-iii a Linux or UNIX system?

*Write your answer in the chat window*

## Practice Test Questions

Is Sun-Hwa-iii a Linux or UNIX system?

*Use the **uname** command to show the name of the kernel*

```
[simben90@sun-hwa-iii ~]$ uname  
Linux
```

**Answer:** Linux

## Practice Test Questions

What distro has been installed on Sun-Hwa-iii?

*Write your answer in the chat window*

## Practice Test Questions

What distro has been installed on Sun-Hwa-iii?

*Use **cat /etc/issue** or **cat /etc/\*-release**  
to show the distro*

```
[simben90@sun-hwa-iii ~]$ cat /etc/issue  
Fedora release 17 (Beefy Miracle)  
Kernel \r on an \m (\l)
```

```
[simben90@sun-hwa-iii ~]$
```

**Answer:** Fedora 17

# More Review (variables)



# Environment Variables

Use `$` for the “value” of a variable

Analogy: Each variable is a named location. The contents of any location is the “value” of that variable.

```
$ echo $LOGNAME  
simmsben
```

```
$ echo HOME  
HOME
```

```
$ echo $HOME  
/home/cis90/simmsben
```

```
$ echo $SHELL  
/bin/bash
```

```
$ echo $HOSTNAME  
opus.cabrillo.edu
```



# Make your own shell variables

*Imagine creating a new variable for use as the fan speed in your car*



```
$ echo $FAN
```

*Initially it's not defined so if echoed it has a null value*

```
$ FAN=HI
$ echo $FAN
HI
$ echo "The fan is set to: " $FAN
The fan is set to: HI
$ FAN=LO
```

*Create a variable named FAN and set the value to "HI"*

```
$ echo "The fan is set to: " $FAN
The fan is set to: LO
```

*Now set the FAN variable to "LO"*

## Activity

```
/home/cis90/simben $ weather=rain
/home/cis90/simben $ country=Spain
/home/cis90/simben $ location="the plain"
/home/cis90/simben $ echo The $weather in $country stays mainly in $location
The rain in Spain stays mainly in the plain
/home/cis90/simben $
```

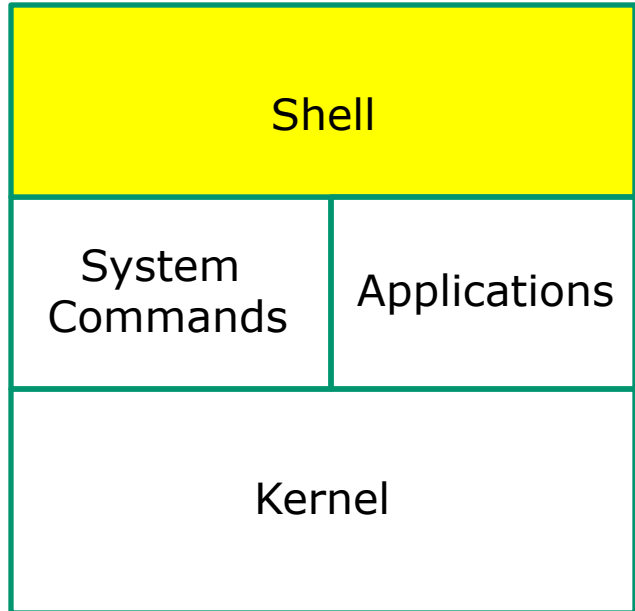
When **echo** is loaded into memory and starts to run:

- 1) How many arguments does it receive from the bash shell?
- 2) Does **echo** see "\$weather" or "rain" as one of the arguments it receives?

*Write your answers in the chat window*

# More Review (shell)

## The Shell



- Allows users to interact with the computer via a **"command line"**.
- **Prompts** for a command, parses the command, finds the right program and gets that program executed.
- Is called a **"shell"** because it hides the underlying operating system.
- Multiple shell programs are available: **sh** (Bourne shell), **bash** ("bourne-again" shell), **cs** (C shell), **ksh** (Korn shell).
- The shell is a **user interface** and a **programming language** (scripts).
- GNOME and KDE desktops could be called **graphical shells**



# Command Syntax

*Shell prints  
this to prompt  
user to enter a  
command*

*Shell parses this command line*



## Examples

**Options** modify the  
behavior of the command

**Arguments** are what the  
command works upon

**Redirection** is  
covered later in  
the course

```

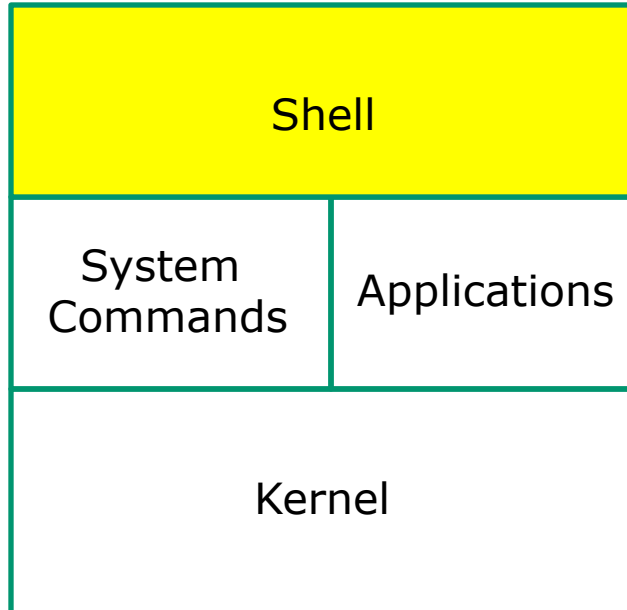
/home/cis90/simben $
/home/cis90/simben $ ls
/home/cis90/simben $ ls -l
/home/cis90/simben $ ls -l -t
/home/cis90/simben $ ls -li Poems/
/home/cis90/simben $ ls -a Poems/ bin/
/home/cis90/simben $ ls -d Poems/ bin/ > mylist
  
```

**Spaces (blanks)** are used to separate the command,  
options and arguments. Additional blanks are ignored.





# The six steps of the Shell



- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat





## Life of the Shell

### Example:

```
/home/cis90/simben $ ls -lt proposal1 proposal2
-rw-r--r--. 1 simben90 cis90 1074 Aug 26 2003 proposal1
-rw-r--r--. 1 simben90 cis90 2175 Jul 20 2001 proposal2
/home/cis90/simben $
```

#### Shell Steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat

*Lets take a deep dive into how a command gets executed.*

***Note it is always a team effort by both the shell and the command.***



## Life of the Shell

### Shell Steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat

# 1) Prompt user for a command

Example:

*The shell begins by outputting the prompt  
(which is based on the PS1 variable)*

```
/home/cis90/simben $ ls -lt proposal1 proposal2
```

*Then you type the command*

FYI, you can mimic outputting the prompt yourself with these commands:

```
/home/cis90/simben $ echo $PS1 to show value of PS1 variable
```

```
$PWD $
```

```
/home/cis90/simben $ echo $PWD $ echo the output of the  
previous command
```

```
/home/cis90/simben $ was output by the echo command above
```

```
/home/cis90/simben $ echo my prompt is: $PWD $  
my prompt is: /home/cis90/simben $
```



## Life of the Shell

### Shell Steps

- 1) Prompt
- 2) **Parse**
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat

## 2) Parse command user typed

Example:

```
ls -lt proposal1 proposal2
```

- Command = ls
- 2 Options = l, t
- 2 Arguments = proposal1, proposal2
- Redirection = NA

*The shell uses the command syntax rules to break down the command line into options, arguments and redirection.*

*Parsing includes expanding variables and properly handling any metacharacters.*

*The shell doesn't actually distinguish between options and arguments. To the shell it is just another argument comprised of a string of text separated by blanks. We will distinguish between options and arguments to better understand command syntax and how it controls what commands do.*



## Life of the Shell

### Shell Steps

- 1) Prompt
- 2) Parse
- 3) **Search**
- 4) Execute
- 5) Nap
- 6) Repeat

# 3) Search for program on the path

**ls** -lt proposal1 proposal2

*Use this command to see the path directories (separated by ':'s) on your path*

```
/home/cis90/simben $ echo $PATH
/usr/lib/qt-3.3/bin:/usr/local/bin:/bin:/usr/bin:
/usr/local/sbin:/usr/sbin:/sbin:
/home/cis90/simben/../../bin:/home/cis90/simben/bin:.
```

*The shell will search each directory in order for an **ls** command*

```
/usr/lib/qt-3.3/bin no
/usr/local/bin no
/bin YES! - it was found in the /bin directory
/usr/bin
/usr/local/sbin
/usr/sbin
/sbin
/home/cis90/simben/../../bin
/home/cis90/simben/bin
.
```

*Note: If the shell cannot find the command on the path it will output "command not found"*

*Try mimicking what the shell does to search for ls:*

```
/home/cis90/simben $ ls /usr/lib/qt-3.3/bin/ls
ls: cannot access /usr/lib/qt-3.3/bin/ls: No
such file or directory
```

```
/home/cis90/simben $ ls /usr/local/bin/ls
ls: cannot access /usr/local/bin/ls: No such
file or directory
```

```
/home/cis90/simben $ ls /bin/ls
/bin/ls
```



## Life of the Shell

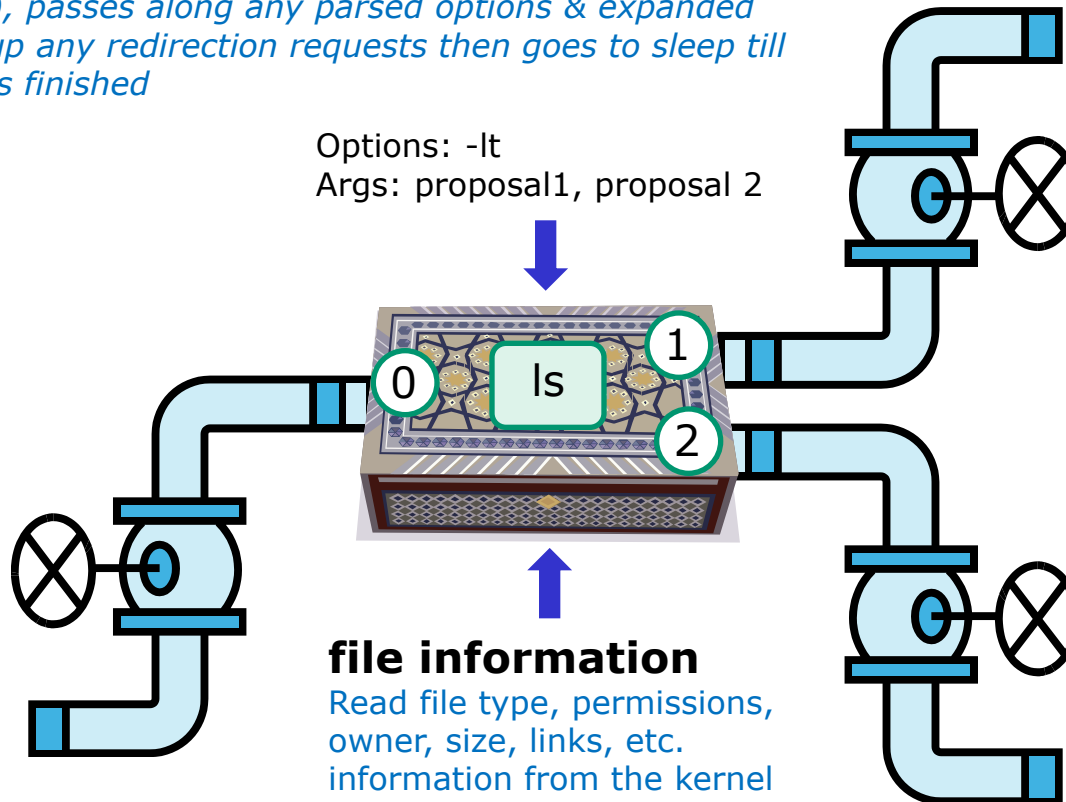
### Shell Steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) **Execute**
- 5) Nap
- 6) Repeat

## 4) Execute the command

```
ls -lt proposal1 proposal2
```

*Invokes the kernel to load the program into memory (which becomes a process), passes along any parsed options & expanded arguments, hooks up any redirection requests then goes to sleep till the new process has finished*





## Life of the Shell

### Shell Steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) **Nap**
- 6) Repeat

# 5) Nap while the command (process) runs to completion

(The shell, itself a loaded process, goes into the sleep state and waits till the command process is finished)

```
/home/cis90/simben $ ls -lt proposal1 proposal2
-rw-r--r--. 1 simben90 cis90 1074 Aug 26 2003 proposal1
-rw-r--r--. 1 simben90 cis90 2175 Jul 20 2001 proposal2
```





## Life of the Shell

**6) And do it all over  
again ... go to step 1**

### Shell Steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat

## Practice Test Questions

Knowing the steps the shell performs, which of the two processes shown below is “taking a nap”?

```
/home/cis90/simben $ ps
  PID TTY          TIME CMD
 21559 pts/0    00:00:00 bash
 22012 pts/0    00:00:00 ps
```

### Shell's steps

- 1) Prompt
- 2) Parse
- 3) Search
- 4) Execute
- 5) Nap
- 6) Repeat

## Practice Test Questions

Knowing the steps the shell performs, which of the two processes shown below is “taking a nap”?

```
/home/cis90/simben $ ps
  PID TTY          TIME CMD
 21559 pts/0        00:00:00 bash
 22012 pts/0        00:00:00 ps
```

Shell's steps  
1) Prompt  
2) Parse  
3) Search  
4) Execute  
5) Nap  
6) Repeat

**Answer:** bash (the shell) is sleeping

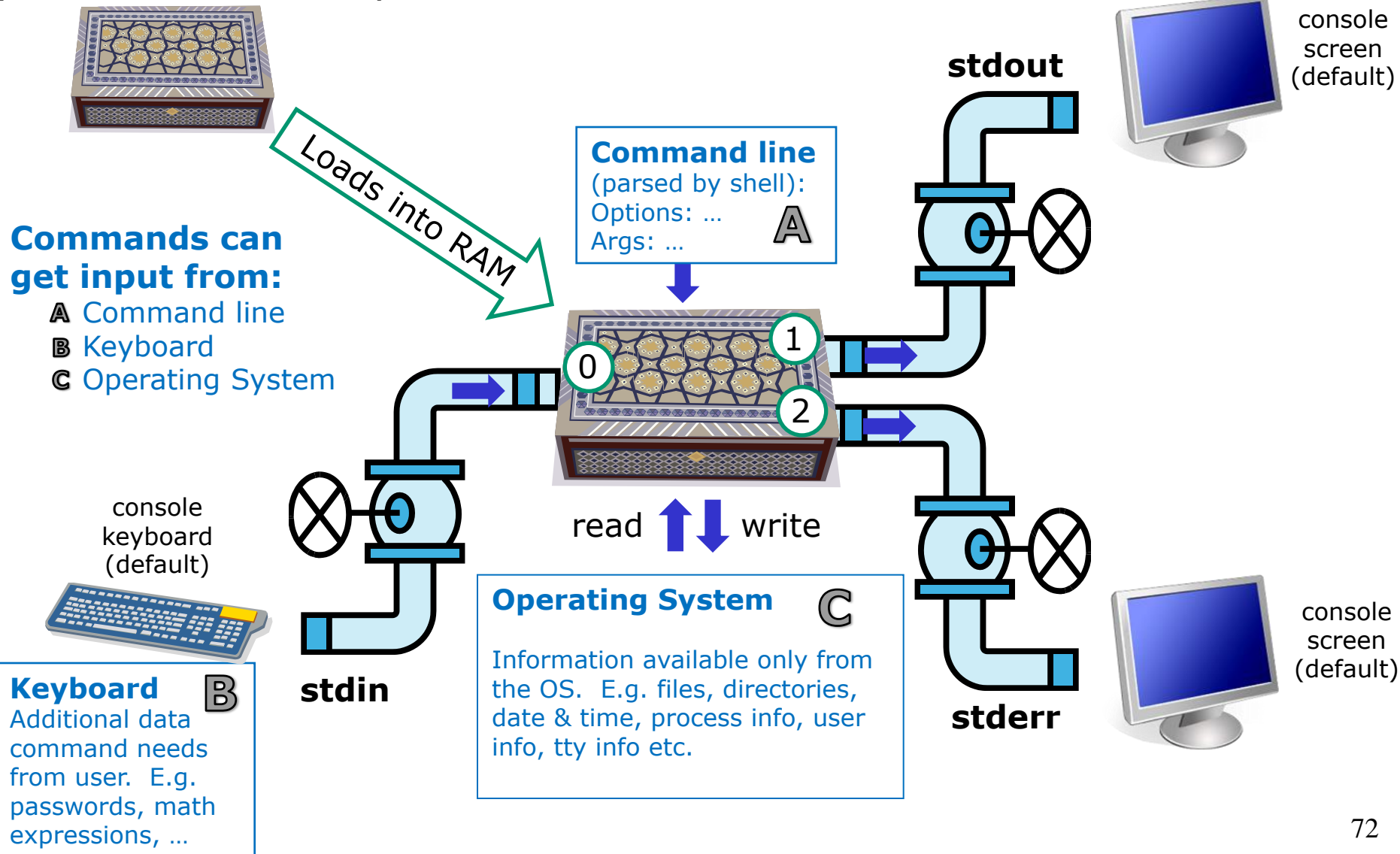
```
/home/cis90/simben $ ps -l
F S   UID     PID   PPID  C  PRI  NI ADDR SZ WCHAN  TTY          TIME CMD
0 S   1001  21559 21558  0   80   0  -  1275 -          pts/0        00:00:00 bash
0 R   1001  22013 21559  0   80   0  -  1213 -          pts/0        00:00:00 ps
```

*Status column, R=running, S=sleeping*

# More Review (inputs)

Program  
(a file on the hard drive)

## Inputs to commands

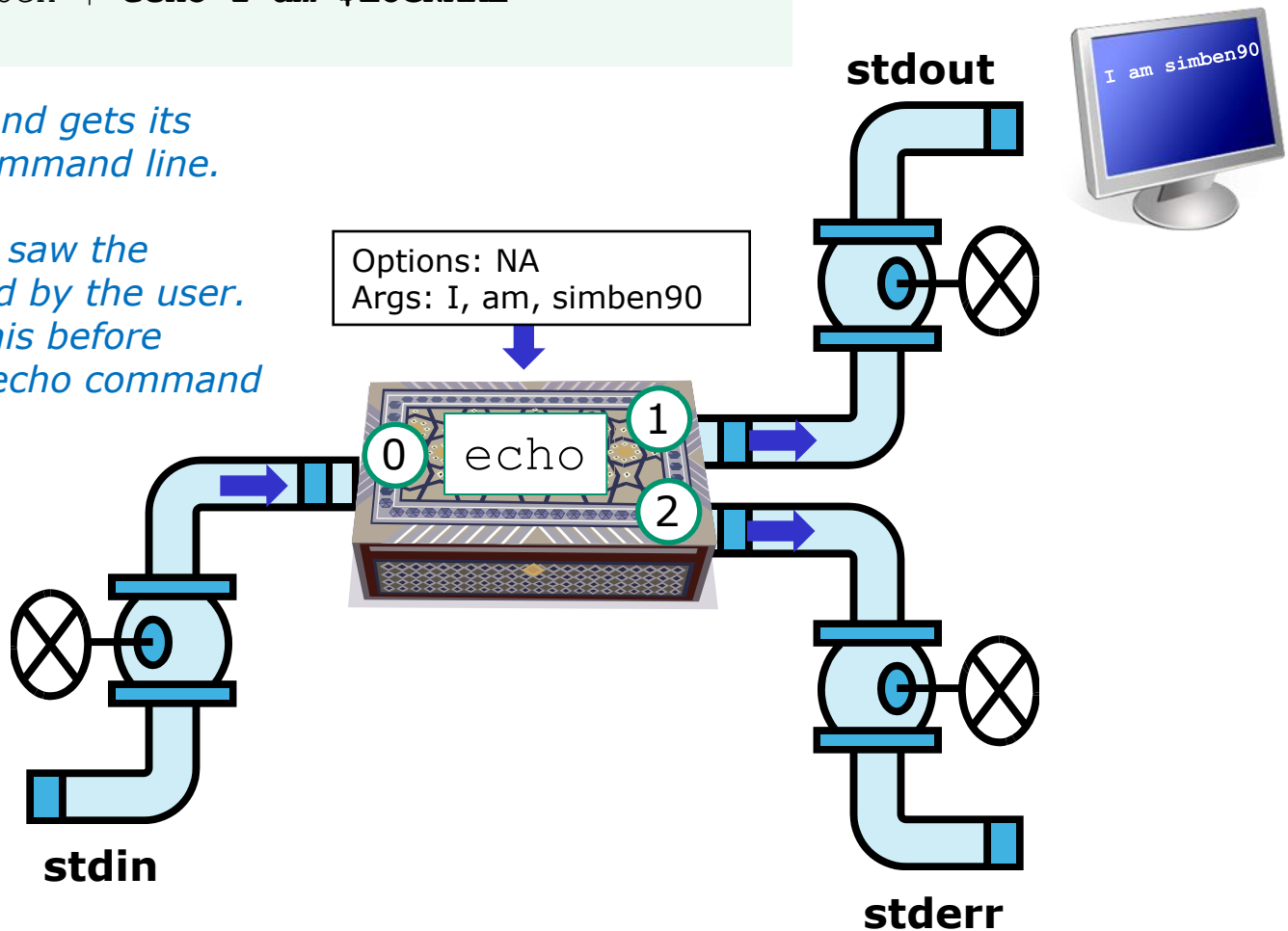


## echo gets input from the command line

```
/home/cis90/simben $ echo I am $LOGNAME
I am simben90
```

The **echo** command gets its input from the command line.

Note: *echo* never saw the "\$LOGNAME" typed by the user. *bash* expanded this before passing it to the *echo* command

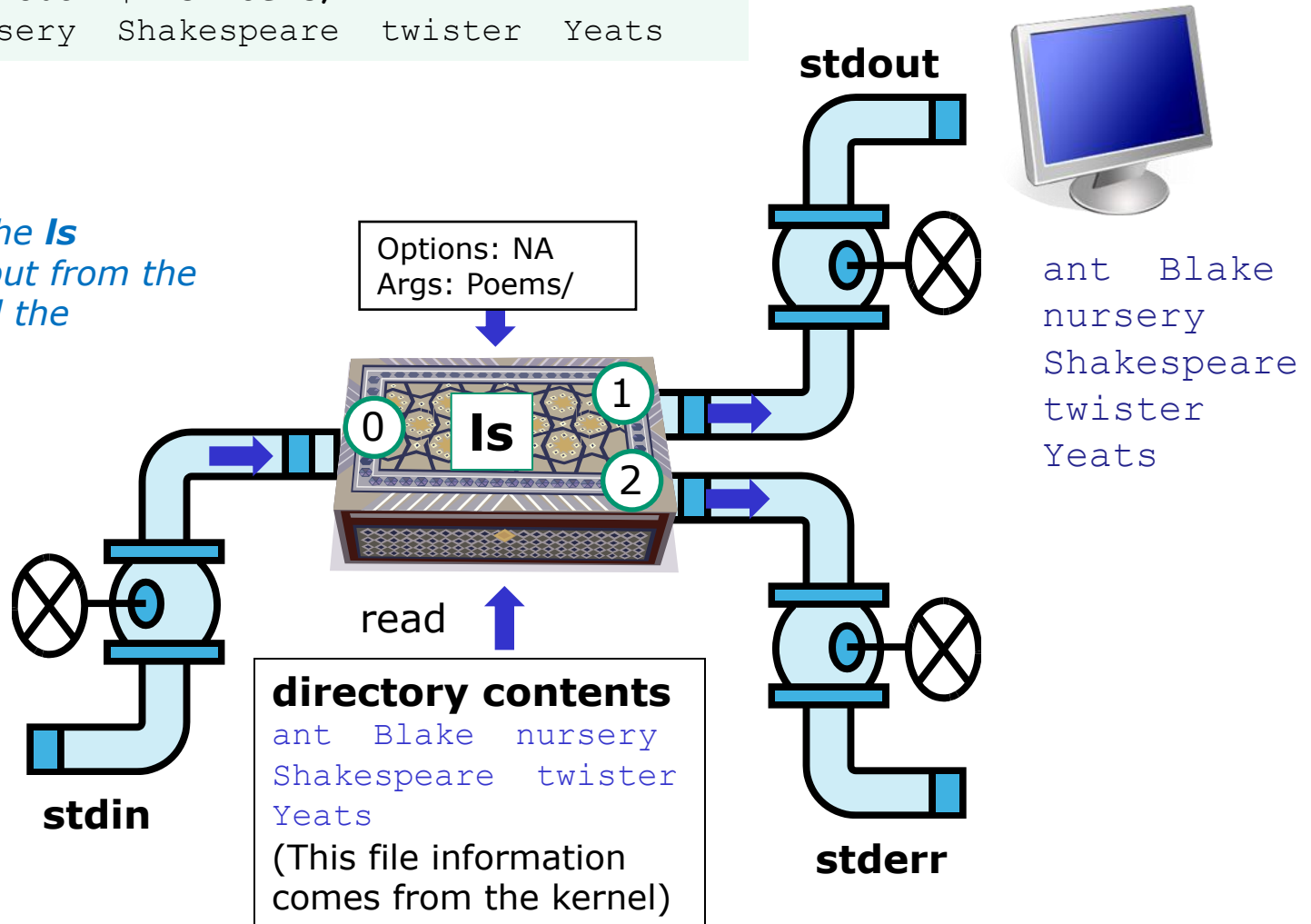


*stdin and stderr were never used for this command*

## This ls command got input from the OS

```
/home/cis90/simmsben $ ls Poems/
ant Blake nursery Shakespeare twister Yeats
```

*In this example, the **ls** command gets input from the command line and the operating system*



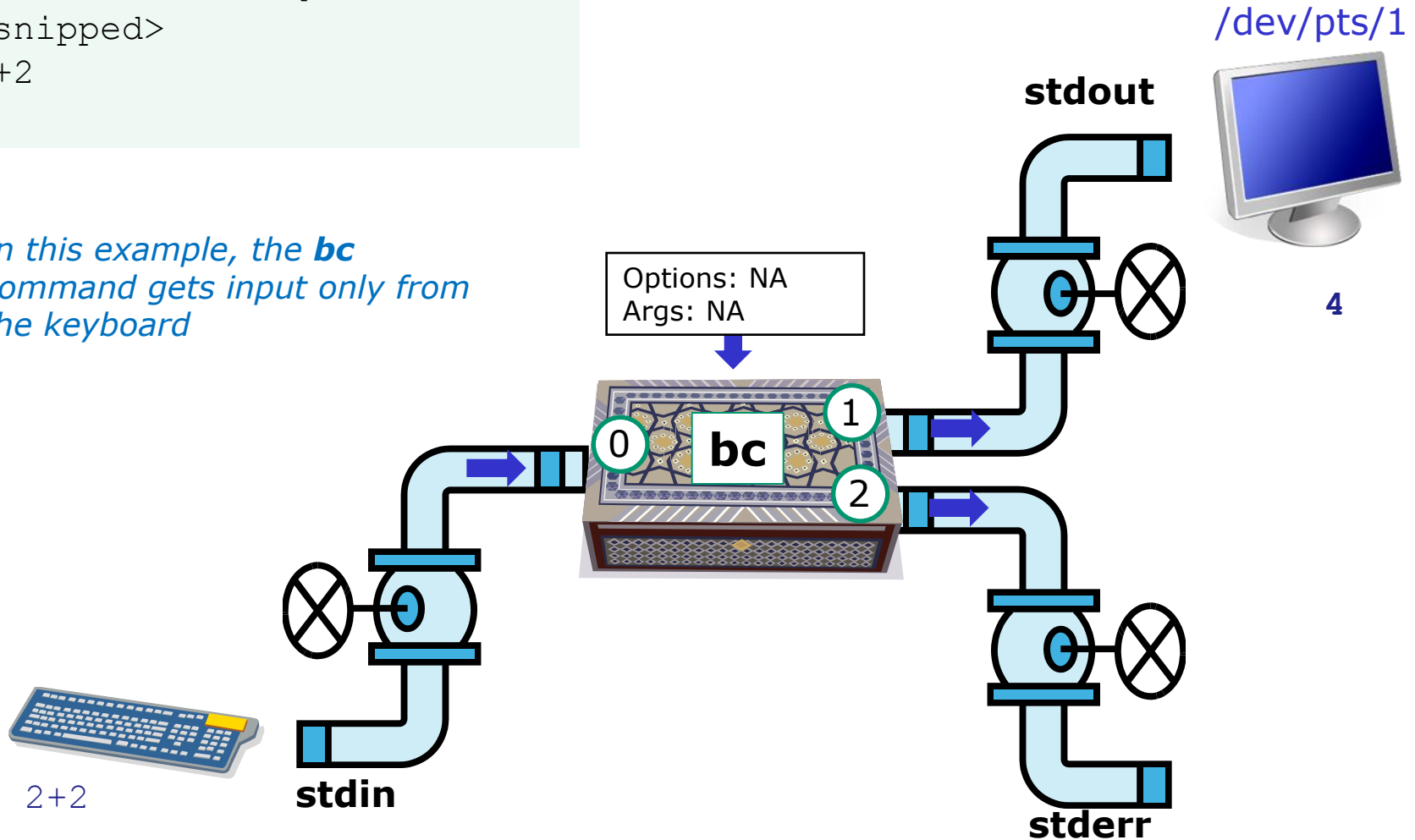
*stdin and stderr were never used for this command*



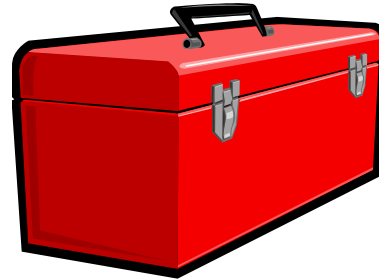
This bc command gets input from the keyboard

```
[rsimms@nosmo ~]$ bc
<snipped>
2+2
4
```

*In this example, the **bc** command gets input only from the keyboard*



*stderr was never used for this command*



# More commands for your toolbox

## Introducing some new commands for this lesson

<b>write</b>	<i>"chat" with another user by writing to their terminal</i>
<b>mesg</b>	<i>enable/disable writes to your terminal</i>
<b>mail</b>	<i>send and read email</i>

# Write Command

Use the write command to chat with another user



```
simben90@oslab:~
/home/cis90/simben $ write milhom90
What's up?

Message from milhom90@oslab.cishawks.net on pts/1 at 09:30 ...
Not much ... want to run around and bark for awhile?
Sure, meet you in the park in 5 mins
Ok
EOF
/home/cis90/simben $ ^C
/home/cis90/simben $
```

```
milhom90@oslab:~
Message from simben90@oslab.cishawks.net on pts/0 at 09:30 ...
What's up?
write simben90
Not much ... want to run around and bark for awhile?
Sure, meet you in the park in 5 mins
Ok
/home/cis90/milhom $ EOF
/home/cis90/milhom $
```

# write command

send a message to another user

**write** *username [ttyname]*

- Use *ttyname* only if there are multiple logins by the target username
- The receiver sees:

Message from *yourname@yourhost* on *yourtty* at *hh:mm* ...

- Each line you type gets sent to the other user's terminal
- To end sending messages type Ctrl-D (Hold down Ctrl and tap D key)
  - The receiver will see an EOF (end of file) at the end
- If the receiver wants to reply then they must use the **write** command as well
- Use **mesg n** (to block incoming messages)
- Use **mesg y** (to allow incoming messages)

# write command

send a message to another user

```
/home/cis90/simben $ type write  
write is /usr/bin/write
```

*The write command resides in the /usr/bin directory*

```
/home/cis90/simben $ file /usr/bin/write  
/usr/bin/write: setgid ELF 32-bit LSB shared object, Intel 80386,  
version 1 (SYSV), dynamically linked (uses shared libs), for  
GNU/Linux 2.6.18, stripped
```

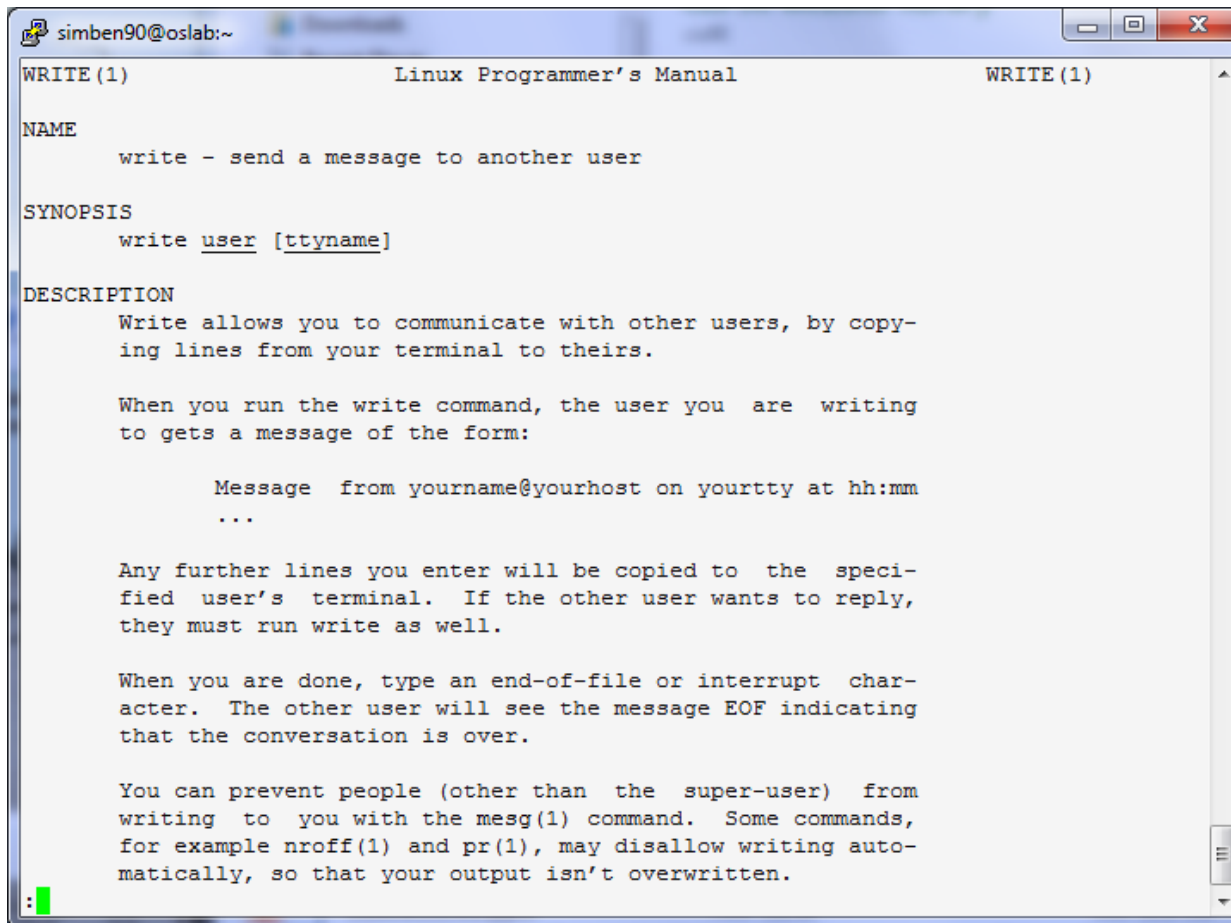
*The write command is a binary executable*



# write command

send a message to another user

```
/home/cis90/simben $ man write
```



The screenshot shows a terminal window titled 'simben90@oslab:~'. The terminal displays the output of the 'man write' command. The output is formatted as a man page with sections: NAME, SYNOPSIS, and DESCRIPTION. The NAME section states 'write - send a message to another user'. The SYNOPSIS section shows 'write user [ttyname]'. The DESCRIPTION section explains that 'write' allows communication with other users by copying lines from the terminal to theirs. It also shows an example of a message format: 'Message from yourname@yourhost on yourtty at hh:mm' followed by three dots. The terminal window has a standard Linux desktop environment with window controls at the top.

```
simben90@oslab:~  
WRITE (1)                                Linux Programmer's Manual                                WRITE (1)  
  
NAME  
    write - send a message to another user  
  
SYNOPSIS  
    write user [ttyname]  
  
DESCRIPTION  
    Write allows you to communicate with other users, by copy-  
    ing lines from your terminal to theirs.  
  
    When you run the write command, the user you are writing  
    to gets a message of the form:  
  
        Message from yourname@yourhost on yourtty at hh:mm  
        ...  
  
    Any further lines you enter will be copied to the speci-  
    fied user's terminal. If the other user wants to reply,  
    they must run write as well.  
  
    When you are done, type an end-of-file or interrupt char-  
    acter. The other user will see the message EOF indicating  
    that the conversation is over.  
  
    You can prevent people (other than the super-user) from  
    writing to you with the mesg(1) command. Some commands,  
    for example nroff(1) and pr(1), may disallow writing auto-  
    matically, so that your output isn't overwritten.
```

Use the **man** command to review how the write command works.

# write command

simben90 writes to milhom90



*Benji, uses the **who** command to see the current users logged into Opus. He sees his friend Homer is logged in twice.*

```
/home/cis90/simben $ who
srelau98 pts/0      2012-09-11 06:36 (anice-34-27-241-136.wanadoo.fr)
simben90 pts/1      2012-09-11 06:47 (42-15-94-107.dsl.com)
alvdes98 pts/2      2012-09-11 07:49 (c-25-14-136-111.comcast.net)
milhom90 pts/3      2012-09-11 08:03 (42-15-94-107.dsl.com)
milhom90 pts/4      2012-09-11 08:09 (42-15-94-107.dsl.com)
```



*Homer, ever curious, uses the **tty** command to see what terminal device he is currently using*

```
/home/cis90/milhom $ tty
/dev/pts/4
/home/cis90/milhom $
```

# write command

simben90 writes to milhom90



```
/home/cis90/simben $ write milhom90
```

*1) Benji enters this*

```
write: milhom90 is logged in more than once; writing to pts/4
```



```
/home/cis90/milhom $
```

```
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...
```

*2) Homer sees this appear on his terminal*

# write command

simben90 writes to milhom90



```
/home/cis90/simben $ write milhom90  
write: milhom90 is logged in more than once; writing to pts/4  
What do you think of the new CentOS distro?
```

*1) Benji enters this*



```
/home/cis90/milhom $  
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...  
What do you think of the new CentOS distro?
```

*2) Homer sees this appear on his terminal*

# write command

simben90 writes to milhom90



```
/home/cis90/milhom $
```

```
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...  
What do you think of the new CentOS distro?
```

```
write simben90
```

*1) Homer enters this*



```
/home/cis90/simben $ write milhom90
```

```
write: milhom90 is logged in more than once; writing to pts/4  
What do you think of the new CentOS distro?
```

```
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...
```

*2) and Benji sees this appear on his terminal*

# write command

simben90 writes to milhom90



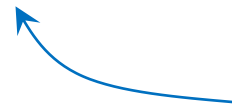
```
/home/cis90/milhom $
```

```
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...
```

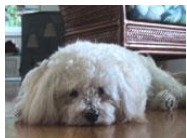
```
What do you think of the new CentOS distro?
```

```
write simben90
```

```
What's with the periods on the long listing permissions?
```



*1) Homer enters this*



```
/home/cis90/simben $ write milhom90
```

```
write: milhom90 is logged in more than once; writing to pts/4
```

```
What do you think of the new CentOS distro?
```

```
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...
```

```
What's with the periods on the long listing permissions?
```



*2) and Benji sees this appear on his terminal*

# write command

## simben90 writes to milhom90



```
/home/cis90/simben $ write milhom90
write: milhom90 is logged in more than once; writing to pts/4
What do you think of the new CentOS distro?
```

```
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...
What's with the periods on the long listing permissions?
```

```
I think it's SELinux
```

*1) Benji enters this*



```
/home/cis90/milhom $
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...
What do you think of the new CentOS distro?
write simben90
What's with the periods on the long listing permissions?
I think it's SELinux
```

*2) Homer sees this appear on his terminal*



# write command

## simben90 writes to milhom90



```
/home/cis90/milhom $  
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...  
What do you think of the new CentOS distro?  
write simben90  
What's with the periods on the long listing permissions?  
I think it's SELinux  
Talk to you later, I'm going to bark a little and take a nap
```

*1) Homer enters this*



```
/home/cis90/simben $ write milhom90  
write: milhom90 is logged in more than once; writing to pts/4  
What do you think of the new CentOS distro?  
  
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...  
What's with the periods on the long listing permissions?  
I think it's SELinux  
Talk to you later, I'm going to bark a little and take a nap
```

*2) and Benji sees this appear on his terminal*

# write command

## simben90 writes to milhom90



```
/home/cis90/milhom $  
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...  
What do you think of the new CentOS distro?  
write simben90  
What's with the periods on the long listing permissions?  
I think it's SELinux  
Talk to you later, I'm going to bark a little and take a nap  
Ctrl-D ← 1) Homer issues a Ctrl-D (holds down Ctrl  
key, then taps D key)  
/home/cis90/milhom $
```



```
/home/cis90/simben $ write milhom90  
write: milhom90 is logged in more than once; writing to pts/4  
What do you think of the new CentOS distro?  
  
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...  
What's with the periods on the long listing permissions?  
I think it's SELinux  
Talk to you later, I'm going to bark a little and take a nap  
EOF ← 2) and Benji sees this appear on his terminal
```

# write command

## simben90 writes to milhom90



```
/home/cis90/simben $ write milhom90
write: milhom90 is logged in more than once; writing to pts/4
What do you think of the new CentOS distro?
```

```
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...
What's with the periods on the long listing permissions?
I think it's SELinux
Talk to you later, I'm going to bark a little and take a nap
EOF
```

bye ← 1) Benji enters this

---



```
/home/cis90/milhom $
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...
What do you think of the new CentOS distro?
write simben90
```

```
What's with the periods on the long listing permissions?
I think it's SELinux
```

```
Talk to you later, I'm going to bark a little and take a nap
```

```
/home/cis90/milhom $ bye ← 2) Homer sees this written to his terminal
```

# write command

## simben90 writes to milhom90



```
/home/cis90/simben $ write milhom90
write: milhom90 is logged in more than once; writing to pts/4
What do you think of the new CentOS distro?
```

```
Message from milhom90@oslab.cabrillo.edu on pts/4 at 09:55 ...
What's with the periods on the long listing permissions?
I think it's SELinux
Talk to you later, I'm going to bark a little and take a nap
EOF
bye
Ctrl-D
```

*1) Benji issues a Ctrl-D (holds down Ctrl key, then taps D key)*

```
/home/cis90/simben $
```



```
/home/cis90/milhom $
Message from simben90@oslab.cabrillo.edu on pts/1 at 09:52 ...
What do you think of the new CentOS distro?
write simben90
What's with the periods on the long listing permissions?
I think it's SELinux
Talk to you later, I'm going to bark a little and take a nap
/home/cis90/milhom $ bye
EOF
```

*2) and Homer sees this appear on his terminal*

# mesg command

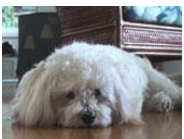
**mesg y** enables and **mesg n** disables writes to your terminal



```
/home/cis90/milhom $ mesg n
```



*1) Homer disables writes to his terminal so he can take his nap*



```
/home/cis90/simben $ write milhom90  
write: milhom90 has messages disabled
```

*2) Benji discovers that Homer is no longer accepting messages*

# who command

The -T option shows who is writeable

*The -T option shows users messages status*

```
/home/cis90/simben $ who -T
srelau98 + pts/0          2012-09-11 06:36 (anice-34-27-241-136.wanadoo.fr)
simben90 + pts/1          2012-09-11 06:47 (42-15-94-107.dsl.com)
alvdes98 + pts/2          2012-09-11 07:49 (c-25-14-136-111.comcast.net)
milhom90 - pts/3          2012-09-11 08:03 (42-15-94-107.dsl.com)
milhom90 - pts/4          2012-09-11 08:09 (42-15-94-107.dsl.com)
```

*+ indicate writes to this user are enabled and - indicates writes to this user are blocked*

```
/home/cis90/simben $ ls -l /dev/pts*
total 0
crw--w----. 1 srelau98 tty 136, 0 Sep 11 08:15 0
crw--w----. 1 simben90 tty 136, 1 Sep 11 08:25 1
crw--w----. 1 alvdes98 tty 136, 2 Sep 11 08:25 2
crw-- --. 1 milhom90 tty 136, 3 Sep 11 08:19 3
crw-- --. 1 milhom90 tty 136, 4 Sep 11 08:19 4
c----- . 1 root root 5, 2 Jul 30 21:25 ptmx
```

*We will learn about file wildcards and permissions later.*

*This is a just a preview showing that write permission is removed from /dev/pts/3 and /dev/pts/4 for the tty group.*

## Class Exercise

write and mesg

- Students, please login to Opus using your own accounts
- Rich, run the pairs script to pair up all the CIS 90 students.
- Students, use the write command to "chat" with your pair mate. e.g. **write** *username*
- Students, ask your pair mate for their real first name and put that in the chat window.
- End the chat session with Ctrl-D

Note to Rich:

Run the pairs script in your cis90/misc/uhist directory

# Sending Mail



# UNIX mail

## Sending messages

**mail** *recipient1 recipient2 ... recipientn*

*The mail command can be used to send an email to one or more recipients.*

*Each argument designates a recipient specified by a username (in /etc/passwd), a normal email address, or an alias (in /etc/aliases).*

### Examples:

**mail** **rsimms**     *username as argument*

**mail** **simben90 calmic90 gutemi90**     *multiple usernames as arguments*

**mail** **richsimms@yahoo.com lefnic90**     *regular email address and  
username as arguments*

**mail** **\$LOGNAME**     *your username, specified using a variable, as argument*

**mail** **cis90-students**     *an alias (used as a distribution list)  
for all CIS 90 students*

# UNIX mail

## Sending messages

```
/home/cis90/simben $ type mail  
mail is /bin/mail
```

*The mail program is on the path and in the /bin directory.*

```
/home/cis90/simben $ file /bin/mail  
/bin/mail: symbolic link to `mailx'
```

*It is a "symbolic link" (we learn about these later) to the mailx program.*

```
/home/cis90/simben $ type mailx  
mailx is /bin/mailx
```

*The mailx program file is also in the /bin directory.*

```
/home/cis90/simben $ file /bin/mailx  
/bin/mailx: ELF 32-bit LSB executable, Intel 80386, version 1  
(SYSV), dynamically linked (uses shared libs), for GNU/Linux  
2.6.18, stripped
```

*The mailx program is a binary executable.*

# UNIX mail

## Sending messages

*As an example, Benji sends an email to Homer (a user on Opus) and Rich (using his Yahoo email address)*

**Homer**  
(milhom90)



**Rich**  
(richsimms@yahoo.com)



**Benji**  
(simben90)

```
/home/cis90/simben $ mail milhom90 richsimms@yahoo.com
```

```
Subject: Where is the old bone
```

```
I can't find my old bone. Let me know if you see it.
```

```
Thanks,
```

```
Benji
```

```
.
```

```
EOT
```

```
/home/cis90/simben $
```

*Use Ctrl-D or a single period to end the message (End Of Text)*

*Recipients can be Opus users (just specify their username) or regular email addresses.*

## Class Exercise

### UNIX mail

- Login to Opus
- Send me a message

```
/home/cis90/simben $ mail rsimms  
Subject: Hello  
This mail program is pretty crazy!  
.  
/home/cis90/simben $
```

## *Notes to Rich*



[ ] - Send out Welcome letter

Use script in /cis90/misc/uhist directory

cp list-full list

mail-welcome

[ ] - Test cis90-students alias

# Reading Mail

# UNIX mail

## Sending messages

### **mail**

*To read mail, enter the mail command with no arguments. The mail command has its own mini-shell with its own set of mail oriented commands.*

# UNIX Mail

## Reading messages



**Homer**  
(milhom90)

```
/home/cis90/milhom $
```

```
You have new mail in /var/spool/mail/milhom90
```

*Homer notices he has received new mail and runs the mail command to see what has arrived*

```
/home/cis90/milhom $ mail
```

```
Heirloom Mail version 12.4 7/29/08. Type ? for help.
```

```
"/var/spool/mail/milhom90": 1 message 1 new
```

```
>N 1 Benji Simms Tue Sep 11 12:59 22/830 "Where is the old bone"
```

```
& 1
```

*He types 1 to read message 1*

```
Message 1:
```

```
From simben90@oslab.cabrillo.edu Tue Sep 11 12:59:27 2012
```

```
Return-Path: <simben90@oslab.cabrillo.edu>
```

```
From: Benji Simms <simben90@oslab.cabrillo.edu>
```

```
Date: Tue, 11 Sep 2012 12:59:27 -0700
```

```
To: richsimms@yahoo.com, milhom90@oslab.cabrillo.edu
```

```
Subject: Where is the old bone
```

```
User-Agent: Heirloom mailx 12.4 7/29/08
```

```
Content-Type: text/plain; charset=us-ascii
```

```
Status: R
```

```
I can't find my old bone. Let me know if you see it.
```

```
Thanks,
```

```
Benji
```

*The N signifies a new message*

*The & is the mail prompt*

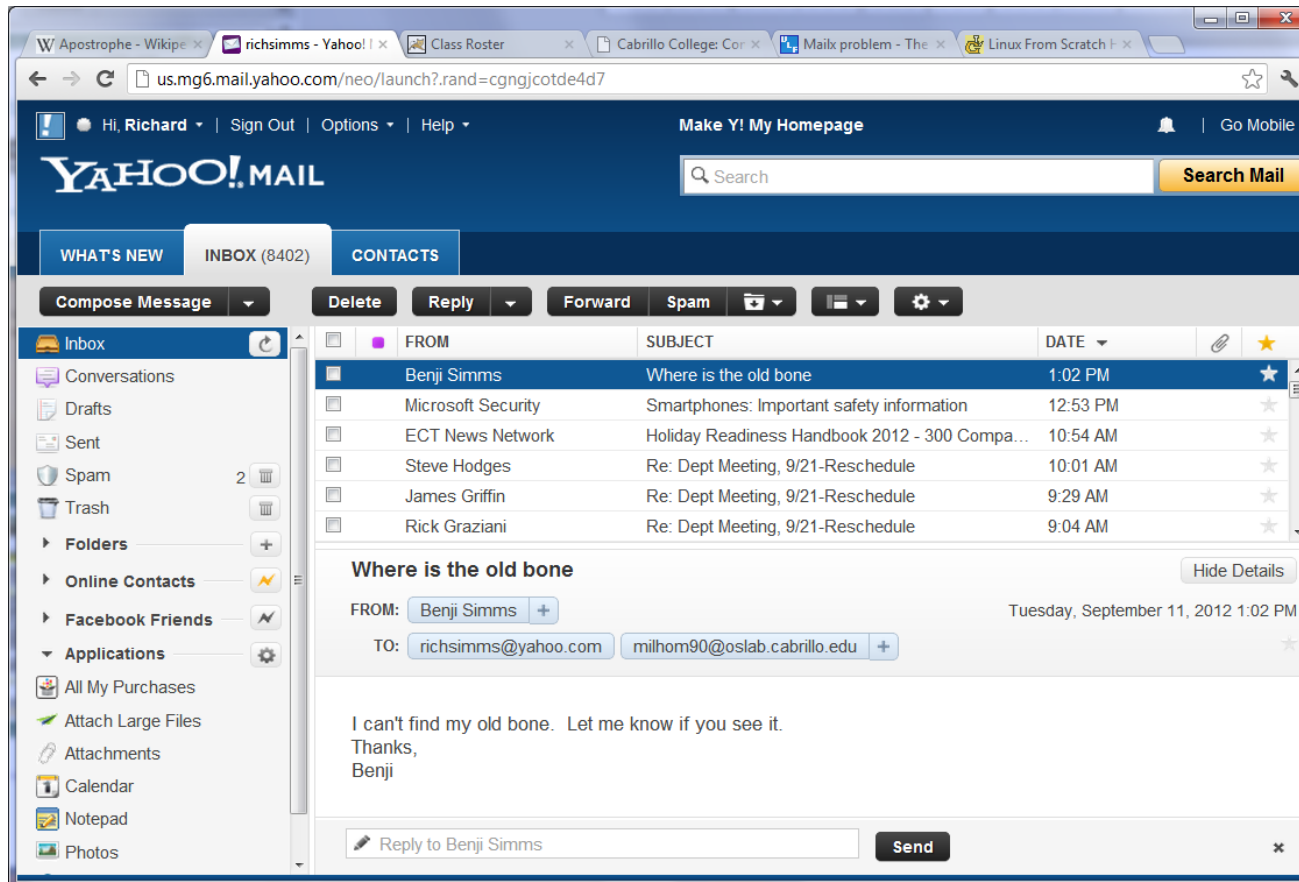


# UNIX mail

## Reading messages sent from UNIX mail



**Rich**  
(richsimms@yahoo.com)



*Rich reads the  
email from Benji  
using Yahoo mail  
(a mail user agent)*

## Class Exercise

### UNIX mail

- Read your own mail by typing the **mail** command by itself
- Use the **p** command followed by the number of the message to print a message.
  - p 1**
  - p 2**     *Or just type the number of the message.*
- Use the **q** command to exit

*Tip: You can just hit the Enter key by itself to read the next unread message.*

# Replying to Mail

# UNIX Mail

## Replying to messages



**Homer**  
(milhom90)

*< continued from above >*

I can't find my old bone. Let me know if you see it.  
Thanks,  
Benji

& **r 1**  
To: milhom90@oslab.cabrillo.edu richsimms@yahoo.com  
simben90@oslab.cabrillo.edu  
Subject: Re: Where is the old bone

Benji Simms <simben90@oslab.cabrillo.edu> wrote:

> I can't find my old bone. Let me know if you see it.  
> Thanks,  
> Benji

**I think its under the sink**  
**- Homer**

.

EOT

&

*After reading the message  
from Benji, Homer replies  
with the mail **r** command  
(for reply to all).*

# UNIX Mail

## Benji gets the reply from Homer



**Benji**  
(simben90)

```
You have mail in /var/spool/mail/simben90
/home/cis90/simben $ mail
Heirloom Mail version 12.4 7/29/08.  Type ? for help.
"/var/spool/mail/simben90": 1 message 1 unread
>U 1 Homer Miller          Tue Sep 11 13:35  30/1096  "Re: Where is the old bone"
& 1
Message 1:
From milhom90@oslab.cabrillo.edu  Tue Sep 11 13:35:30 2012
Return-Path: <milhom90@oslab.cabrillo.edu>
From: Homer Miller <milhom90@oslab.cabrillo.edu>
Date: Tue, 11 Sep 2012 13:35:30 -0700
To: simben90@oslab.cabrillo.edu, richsimms@yahoo.com,
    milhom90@oslab.cabrillo.edu
Subject: Re: Where is the old bone
User-Agent: Heirloom mailx 12.4 7/29/08
Content-Type: text/plain; charset=us-ascii
Status: RO

Benji Simms <simben90@oslab.cabrillo.edu> wrote:

> I can't find my old bone.  Let me know if you see it.
> Thanks,
> Benji
I think its under the sink
- Homer
```

*Benji notices he  
has new mail  
which he reads  
using the mail  
command (with no  
arguments) and  
then typing the  
message number  
he wants to read*



**Rich**  
(richsimms@yahoo.com)

## UNIX Mail

The screenshot shows a web browser window with multiple tabs. The active tab is the Yahoo! Mail inbox for user richsimms. The inbox list shows several emails, with the top one from Homer Miller titled 'Re: Where is the old bone' at 1:38 PM. Below the inbox list, the details of this email are shown. It is a reply to a message from Benji Simms. The email body contains a quoted message from Benji Simms asking for help finding an 'old bone'.

FROM	SUBJECT	DATE
Homer Miller	Re: Where is the old bone	1:38 PM
Benji Simms	Where is the old bone	1:02 PM
Microsoft Security	Smartphones: Important safety information	12:53 PM
ECT News Network	Holiday Readiness Handbook 2012 - 300 Compa...	10:54 AM
Steve Hodnes	Re: Dent Meeting 9/21-Reschedule	10:01 AM

**Re: Where is the old bone**

FROM: Homer Miller

TO: simben90@oslab.cabrillo.edu, richsimms@yahoo.com, milhom90@oslab.cabrillo.edu

Tuesday, September 11, 2012 1:38 PM

Benji Simms <simben90@oslab.cabrillo.edu> wrote:

> I can't find my old bone. Let me know if you see it.  
 > Thanks,  
 > Benji  
 I think its under the sink  
 - Homer

*Since Homer  
replied to all,  
Rich also gets a  
copy*

## Class Exercise

### UNIX mail

- Use **ls /home/cis90** to see all CIS 90 home directories (add "90" to get the usernames) or the **who** command and send an email to three other CIS 90 students (your choice) in one message.

Hint: use **mail** *user1 user2 user3*

- Reply to any emails you get (run **mail** and use **r** command)



# Saving Mail to a Folder



# UNIX Mail

## Saving messages

```
/home/cis90/simben $ mail ← Benji checks for new mail
Heirloom Mail version 12.4 7/29/08. Type ? for help.
"/var/spool/mail/simben90": 1 message 1 new
>N 1 Homer Miller Tue Sep 11 21:04 21/830 "Salsa"
& 1 ← Prints the first (and only) message
Message 1:
From milhom90@oslab.cabrillo.edu Tue Sep 11 21:04:16 2012
Return-Path: <milhom90@oslab.cabrillo.edu>
From: Homer Miller <milhom90@oslab.cabrillo.edu>
Date: Tue, 11 Sep 2012 21:04:16 -0700
To: simben90@oslab.cabrillo.edu
Subject: Salsa
User-Agent: Heirloom mailx 12.4 7/29/08
Content-Type: text/plain; charset=us-ascii
Status: R

Don't forget, salsa class tonight at the Palomar
- Homer


& s 1 archives ← Saves this message to a folder named "archives"
"archives" [New file] 23/851
& q
```

# Browsing a mail folder

# UNIX mail

## Browse mail folders using the -f option

*use the f option to specify a mail file*



```
/home/cis90/simben $ mail -f archives
Heirloom Mail version 12.4 7/29/08.  Type ? for help.
"archives": 5 messages 4 new
   1 Homer Miller      Tue Sep 11 21:04  22/841  "Salsa"
>N  2 Homer Miller      Tue Sep 11 21:25  20/790  "Hola"
   N  3 Rich Simms       Tue Sep 11 21:58  20/752  "Treasure"
     4 Rich Simms       Tue Sep 11 22:01  21/798  "Lab Hours on Monday"
   N  5 Rich Simms       Tue Sep 11 22:01  20/796  "Where were you last
summer?"
&
```

*Opening a mail folder named archives which has some saved messages*

# Forwarding Mail

# mail commands

## Forwarding a message with ~m

```
rsimms@opus:~$ mail
Mail version 8.1 6/6/93.  Type ? for help.
"/var/spool/mail/rsimms": 5 messages 1 unread
>U  1  jimg@opus.cabrillo.e  Sun Jun 22 13:53  22/836  "Hot days and servers"
    2  simmsmar@opus.cabril  Thu Jul 24 12:28  19/739  "Don't forget to bring"
    3  simmsben@opus.cabril  Thu Jul 24 12:27  17/708  "Nisene Hike"
    4  rsimms@opus.cabrillo  Thu Jul 24 12:33  21/819  "Re: Hot days and serv"
    5  roddyduk@opus.cabril  Thu Jul 24 15:41  19/702  "Salsa"
& m simmsben
Subject: re: Salsa
Hi Benji,

Did you see this:
~m5
Interpolating: 5
(continue)

Later,

- Rich
.
Cc:
&
```

*This is how  
you forward  
message 5*

*Tip: Use this to  
send the event  
you get from  
the instructor  
to others for  
Lab 3*

```
simmsben@opus:~$ mail
/home/cis90/simmsben $ mail
Mail version 8.1 6/6/93.  Type ? for help.
"/var/spool/mail/simmsben": 1 message 1 new
>N  1  rsimms@opus.cabrillo  Thu Jul 24 18:51  33/935  "re: Salsa"
& p 1
Message 1:
From rsimms@opus.cabrillo.edu  Thu Jul 24 18:51:55 2008
Date: Thu, 24 Jul 2008 18:51:55 -0700
From: Rich Simms <rsimms@opus.cabrillo.edu>
To: simmsben@opus.cabrillo.edu
Subject: re: Salsa

Hi Benji,

Did you see this:

From roddyduk@opus.cabrillo.edu  Thu Jul 24 15:41:35 2008
Date: Thu, 24 Jul 2008 15:41:35 -0700
From: Duke Roddy <roddyduk@opus.cabrillo.edu>
To: rsimms@opus.cabrillo.edu
Subject: Salsa

You and Elizabeth coming to the Palomar this Friday?
Let me know,
- Duke

Later,

- Rich
&
```

# mail commands

## Alternate ways to forward a message

*There is an easier way to forward a message with the latest version of mailx!*

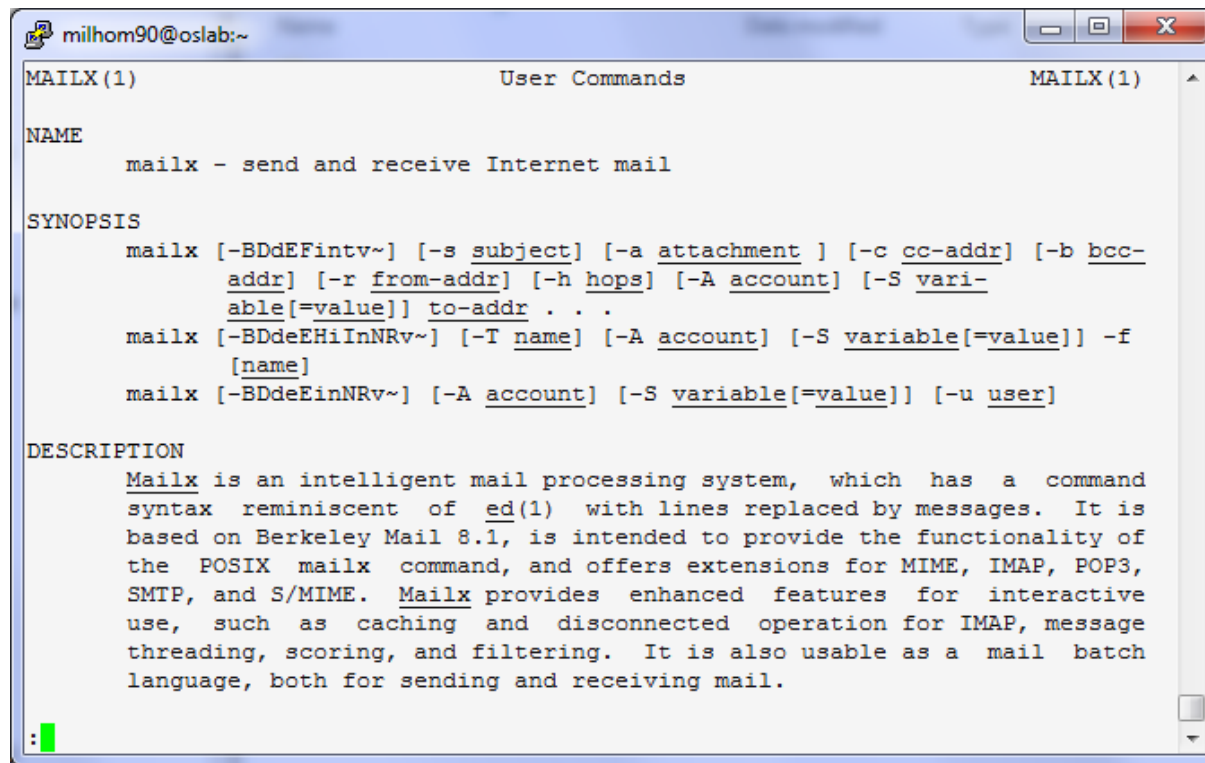
*I wonder who will be the first person to find out how its done and post the solution to the forum?*



# Mail Documentation

# man page for mail

```
/home/cis90/milhom $ man mail
```



```

MAILX(1)                                User Commands                                MAILX(1)

NAME
    mailx - send and receive Internet mail

SYNOPSIS
    mailx [-BDdEFintv~] [-s subject] [-a attachment] [-c cc-addr] [-b bcc-
        addr] [-r from-addr] [-h hops] [-A account] [-S vari-
        able[=value]] to-addr . . .
    mailx [-BDdeEHInNRv~] [-T name] [-A account] [-S variable[=value]] -f
        [name]
    mailx [-BDdeEinNRv~] [-A account] [-S variable[=value]] [-u user]

DESCRIPTION
    Mailx is an intelligent mail processing system, which has a command
    syntax reminiscent of ed(1) with lines replaced by messages. It is
    based on Berkeley Mail 8.1, is intended to provide the functionality of
    the POSIX mailx command, and offers extensions for MIME, IMAP, POP3,
    SMTP, and S/MIME. Mailx provides enhanced features for interactive
    use, such as caching and disconnected operation for IMAP, message
    threading, scoring, and filtering. It is also usable as a mail batch
    language, both for sending and receiving mail.

:
  
```

*In the bash shell, use the man command for extensive documentation on mail*



# Mail ? command

& ?

	mail commands	
type <message list>		type messages
next		goto and type next message
from <message list>		give head lines of messages
headers		print out active message headers
delete <message list>		delete messages
undelete <message list>		undelete messages
save <message list> folder		append messages to folder and mark as saved
copy <message list> folder		append messages to folder without marking them
write <message list> file		append message texts to file, save attachments
preserve <message list>		keep incoming messages in mailbox even if saved
Reply <message list>		reply to message senders
reply <message list>		reply to message senders and all recipients
mail addresses		mail to specific recipients
file folder		change to another folder
quit		quit and apply changes to folder
xit		quit and discard changes made to folder
!		shell escape
cd <directory>		chdir to directory or home if none given
list		list names of all available commands

A <message list> consists of integers, ranges of same, or other criteria separated by spaces. If omitted, mail uses the last message typed.

&

*Use the ? command to see a short list of common mail commands*

# Listing messages (headers)

# mail h (headers) command

e.g. list my current folder)

```
rsimms@oslab:~/cis90/misc/uhist
& h
> 1 Rich Simms      Fri Feb 19 10:50  17/659  "Test"
   2 Rich Simms      Wed Apr 28 15:52  24/721  "another get well mess"
   3 Jim Griffin     Sat May  1 14:11  28/1131  "Re: Get well soon"
   4 Christopher Botos Wed Sep  1 21:44 152/10825 "Re: Cabrillo CIS 90 u"
   5 Jason Hamil     Wed Sep  1 21:48 191/9909  "RE: Cabrillo CIS 90 u"
   6 Laura Pirkle    Wed Sep  1 22:46 217/9590  "Re: Cabrillo CIS 90 u"
   7 Adriana Plastina Wed Sep  1 22:58 1028/77247 "picture of my face f"
   8 Saulius Zilis   Wed Sep  1 23:12  34/2112  "Re: Cabrillo CIS 90 u"
   9 dennis anti     Thu Sep  2 00:22 178/9983  "Re: Cabrillo CIS 90 u"
  10 francisco cardenas Thu Sep  2 15:15 3166/192496
  11 Jennifer Parrish Tue Sep  7 22:59 3288/201881 "Re: Cabrillo CIS 90"
  12 Rudy Perez      Wed Sep  8 13:15  46/2182  "ccconfer class listin"
  13 francisco cardenas Wed Sep  8 13:15  47/2356  "quiz"
  14 James Garibay    Wed Sep  8 13:32 3153/191560
  15 Jim Griffin     Tue Aug 17 20:20  22/1016  "Opus mail"
  16 Rudy Perez      Thu Sep  2 17:17 2529/192676 "student survey"
  17 Rich Simms      Tue Sep 14 20:26  88/7804  "Re: Saulius"
  18 Mike Delfin     Wed Sep 15 15:06  15/634  "Re: Welcome"
  19 Mike Delfin     Wed Sep 15 15:08  17/636  "Re: Welcome"
& █
```

*Use the h command to show messages the current folder*

# mail h (headers) command

e.g. list my current folder)

*N = New message, a U = Unread message*

```
simben90@oslab:~  
& h  
N 1 Homer Miller      Tue Sep 11 21:25  20/790  "Hola"  
N 2 Rich Simms        Tue Sep 11 21:58  20/752  "Treasure"  
> 3 Rich Simms        Tue Sep 11 22:01  20/788  "Lab Hours on Monday"  
N 4 Rich Simms        Tue Sep 11 22:01  20/796  "Where were you last summer?"  
&
```

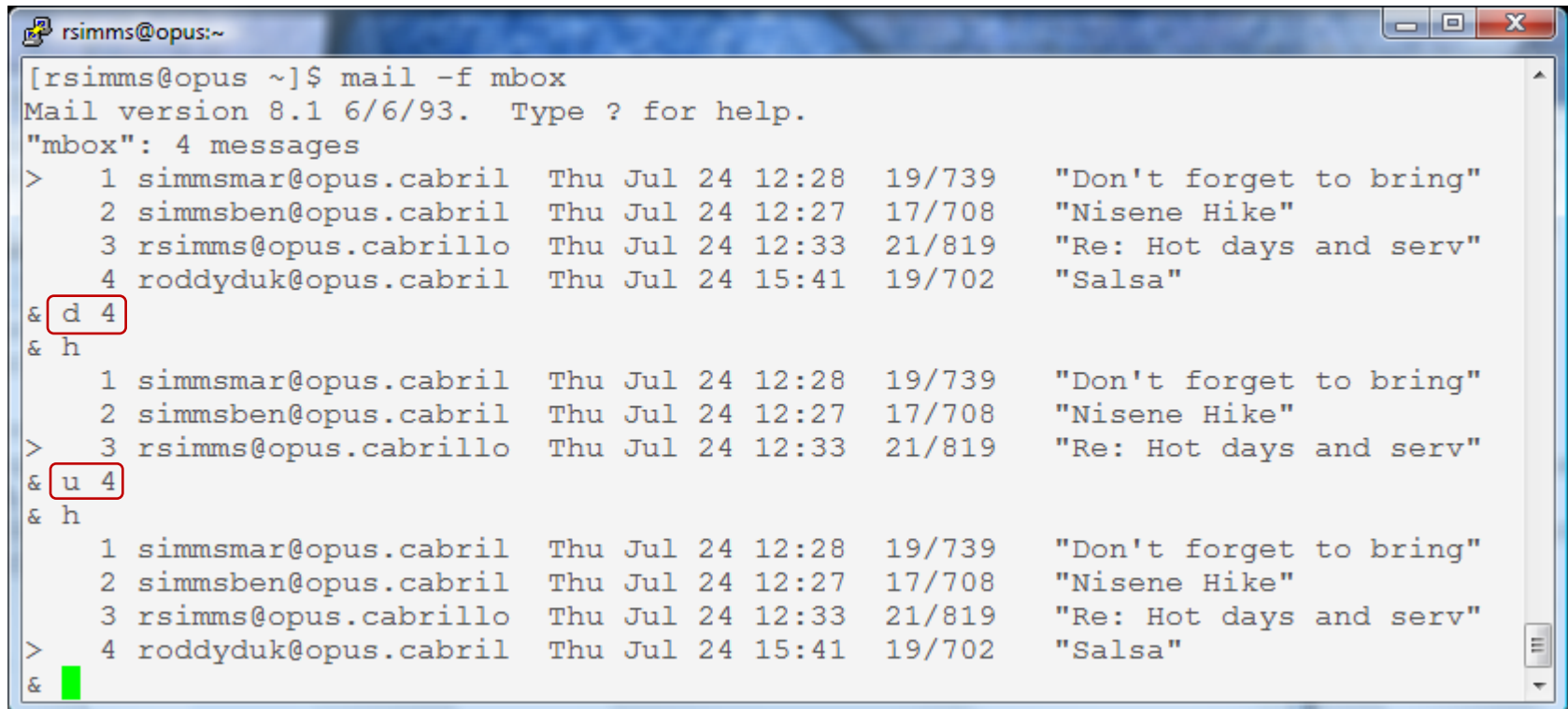
*& is mail prompt for next command*

*> points to the current message (last one printed)*

# Deleting Messages

# mail commands

## (d)elelete and (u)ndelete



```
[rsimms@opus ~]$ mail -f mbox
Mail version 8.1 6/6/93.  Type ? for help.
"mbox": 4 messages
>  1 simmsmar@opus.cabrill  Thu Jul 24 12:28  19/739  "Don't forget to bring"
   2 simmsben@opus.cabrill  Thu Jul 24 12:27  17/708  "Nisene Hike"
   3 rsimms@opus.cabrillo   Thu Jul 24 12:33  21/819  "Re: Hot days and serv"
   4 roddyduk@opus.cabrill  Thu Jul 24 15:41  19/702  "Salsa"
& d 4
& h
   1 simmsmar@opus.cabrill  Thu Jul 24 12:28  19/739  "Don't forget to bring"
   2 simmsben@opus.cabrill  Thu Jul 24 12:27  17/708  "Nisene Hike"
>  3 rsimms@opus.cabrillo   Thu Jul 24 12:33  21/819  "Re: Hot days and serv"
& u 4
& h
   1 simmsmar@opus.cabrill  Thu Jul 24 12:28  19/739  "Don't forget to bring"
   2 simmsben@opus.cabrill  Thu Jul 24 12:27  17/708  "Nisene Hike"
   3 rsimms@opus.cabrillo   Thu Jul 24 12:33  21/819  "Re: Hot days and serv"
>  4 roddyduk@opus.cabrill  Thu Jul 24 15:41  19/702  "Salsa"
&
```

*Messages can be deleted (and undeleted)*

# Mail files

# UNIX mail

## The dead.letter mail file

```
/home/cis90/simben $ mail bogus
Subject: Dead stuff
I doubt you will get this because you don't exist!
.
EOT
You have mail in /var/spool/mail/simben90
/home/cis90/simben $ /home/cis90/simben/dead.letter... Saved message in
/home/cis90/simben/dead.letter

/home/cis90/simben $ mail -f dead.letter
Heirloom Mail version 12.4 7/29/08.  Type ? for help.
"dead.letter": 1 message
> 1 To bogus          Tue Sep 17 10:04  18/562  "Dead s"
& d 1
& q
"dead.letter" complete
/home/cis90/simben $
```

*Undeliverable mail is placed in your dead.letter file. You can cat this file or open it with the mail command*



# UNIX mail

The mail folders are ascii text files

```
/home/cis90/simben $ ls
```

archives	empty	Lab2.1	Miscellaneous	proposal2	text.err
bigfile	Hidden	letter	mission	proposal3	text.fxd
bin	lab01.graded	log	Poems	small_town	timecal
dead.letter	Lab2.0	mbox	proposal1	spellk	what_am_i

```
/home/cis90/simben $ ls /var/mail/simben90
```

```
/var/mail/simben90
```

1 & 4: User's can create there own mail folder files, giving them any name they like, such as archives and mbox

```
/home/cis90/simben $ file archives dead.letter mbox /var/spool/mail/simben90
```

```
1) archives: ASCII mail text
2) dead.letter: ASCII mail text
3) mbox: ASCII mail text
4) /var/spool/mail/simben90: ASCII mail text
```

*Mail files are text files that you can **cat** or open with **mail -f***

2) All undeliverable messages go into a user's dead.letter file

3) All incoming new messages are initially placed in the /var/mail/<username> file

# UNIX mail

## The mail folders are ascii text files

*Mail files are ASCII text files. You can cat them out or open with the mail command.*

```
/home/cis90/simben $ cat archives
From milhom90@oslab.cishawks.net Mon Sep 16 18:52:53 2013
Return-Path: <milhom90@oslab.cishawks.net>
Received: from oslab.cishawks.net (localhost [127.0.0.1])
    by oslab.cabrillo.edu (8.14.4/8.14.4) with ESMTP id r8H1q rmw008499
    for <simben90@oslab.cishawks.net>; Mon, 16 Sep 2013 18:52 :53 -0700
Received: (from milhom90@localhost)
    by oslab.cishawks.net (8.14.4/8.14.4/Submit) id
    for simben90; Mon, 16 Sep 2013 18:52:53 -0700
From: Homer Miller <milhom90@oslab.cishawks.net>
Message-Id: <201309170152.r8H1qrJZ008497@oslab.cishawks
Date: Mon, 16 Sep 2013 18:52:53 -0700
To: simben90@oslab.cishawks.net
Subject: Fwd: Hot Potato
User-Agent: Heirloom mailx 12.4 7/29/08
MIME-Version: 1.0
Content-Type: text/plain; charset=us-ascii
Content-Transfer-Encoding: 7bit
Status: O

----- Original Message -----
From: Rich Simms <rsimms@oslab.cishawks.net>
Date: Sun, 15 Sep 2013 15:41:49 -0700
To: milhom90@oslab.cishawks.net
Subject: Hot Potato
```

You got it ... forward it on! - Rich

```
/home/cis90/simben $
```

```
/home/cis90/simben $ mail -f archives
Heirloom Mail version 12.4 7/29/08. Type ? for help.
"archives": 1 message 1 unread
>U 1 Homer Miller Mon Sep 16 18:52 28/1002 "Fwd: H"
& 1
Message 1:
From milhom90@oslab.cishawks.net Mon Sep 16 18:52:53 2013
Return-Path: <milhom90@oslab.cishawks.net>
From: Homer Miller <milhom90@oslab.cishawks.net>
Date: Mon, 16 Sep 2013 18:52:53 -0700
To: simben90@oslab.cishawks.net
Subject: Fwd: Hot Potato
User-Agent: Heirloom mailx 12.4 7/29/08
Content-Type: text/plain; charset=us-ascii
Status: RO
```

```
----- Original Message -----
From: Rich Simms <rsimms@oslab.cishawks.net>
Date: Sun, 15 Sep 2013 15:41:49 -0700
To: milhom90@oslab.cishawks.net
Subject: Hot Potato
```

You got it ... forward it on! - Rich

```
& q
"archives" complete
/home/cis90/simben $
```

## Class Exercise

### UNIX mail

- Send yourself several test messages with different subjects:

**mail \$LOGNAME**

**mail \$LOGNAME**

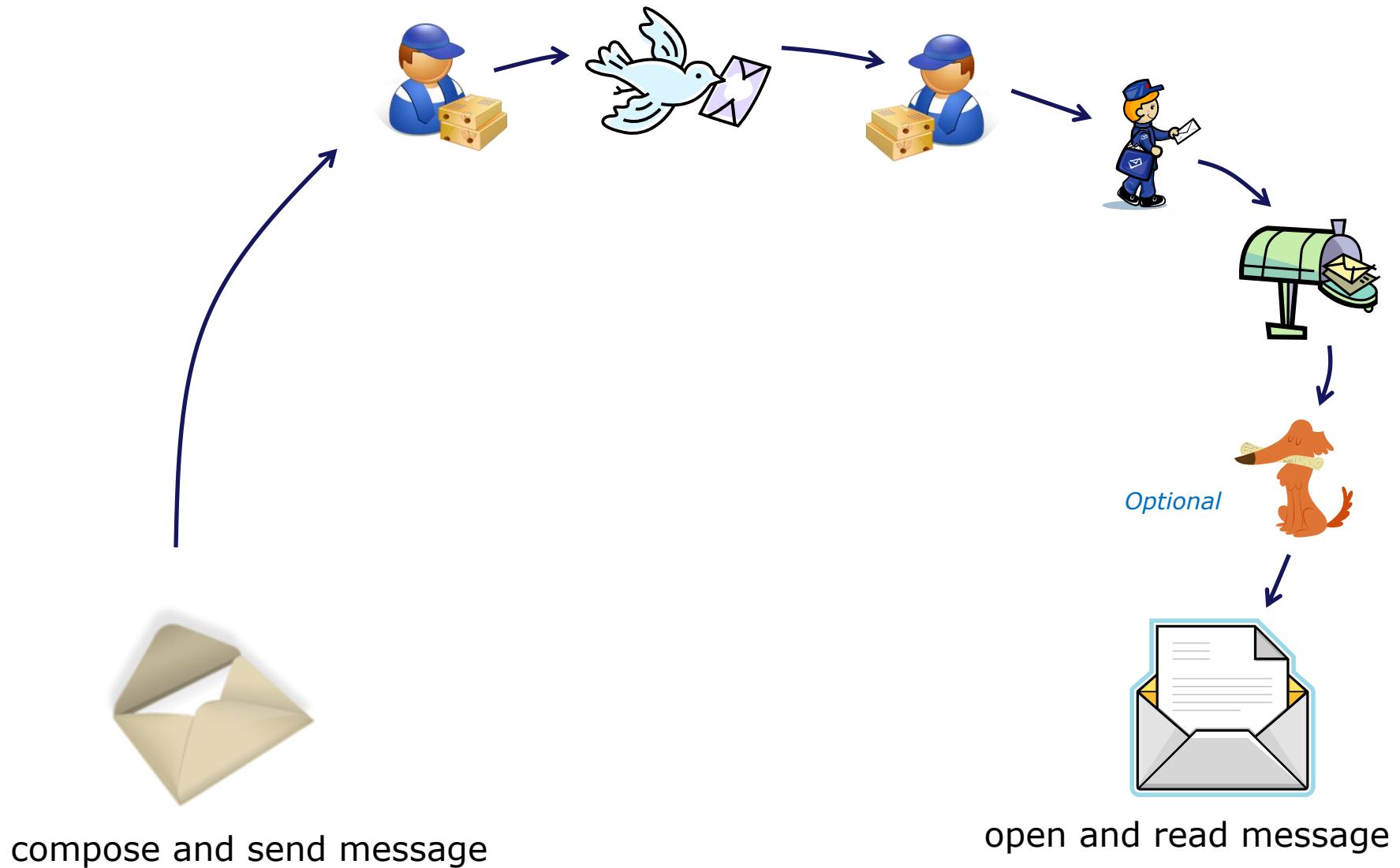
- Now read your mail

**mail**

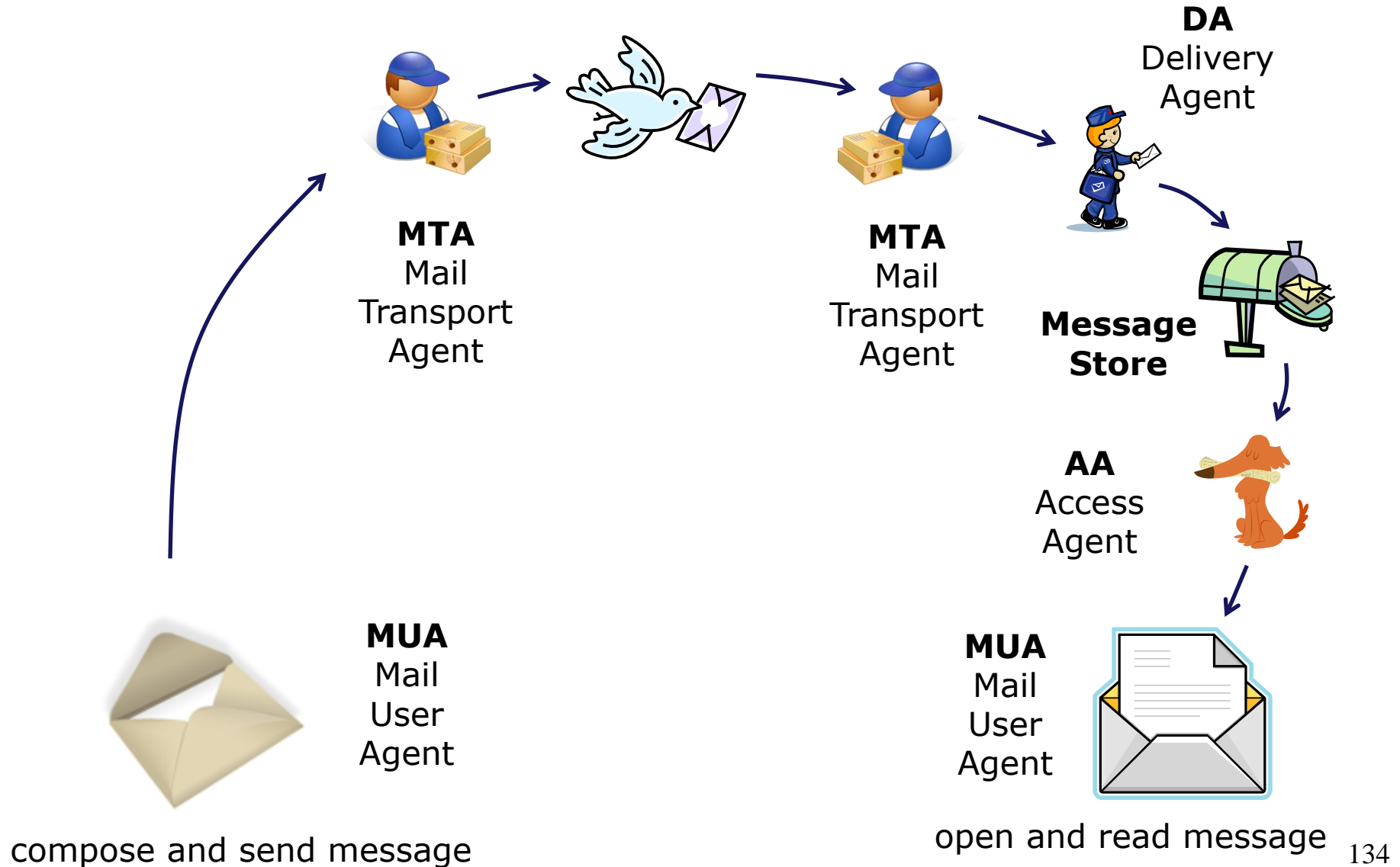
- Use the **h** command to list the message headers
- Read all your messages using **p** command
- Use the **d** command to delete one of the messages
- Use the **s** command to save one message to a folder named archives
- Use **q** to quit mail
- Read the mail in your archives with **mail -f archives**
- Use **q** to quit mail

# end-to-end email

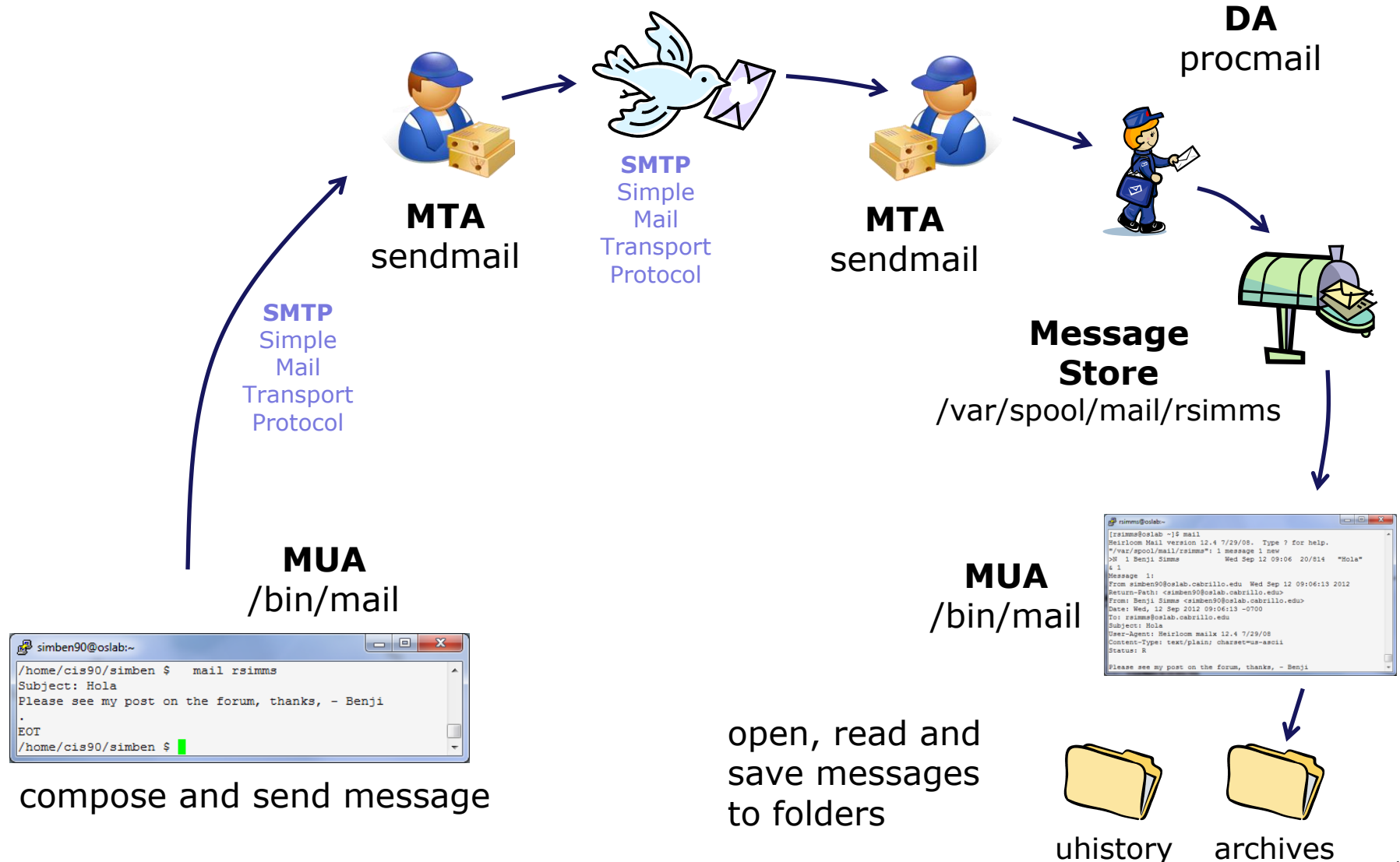
## end-to-end email



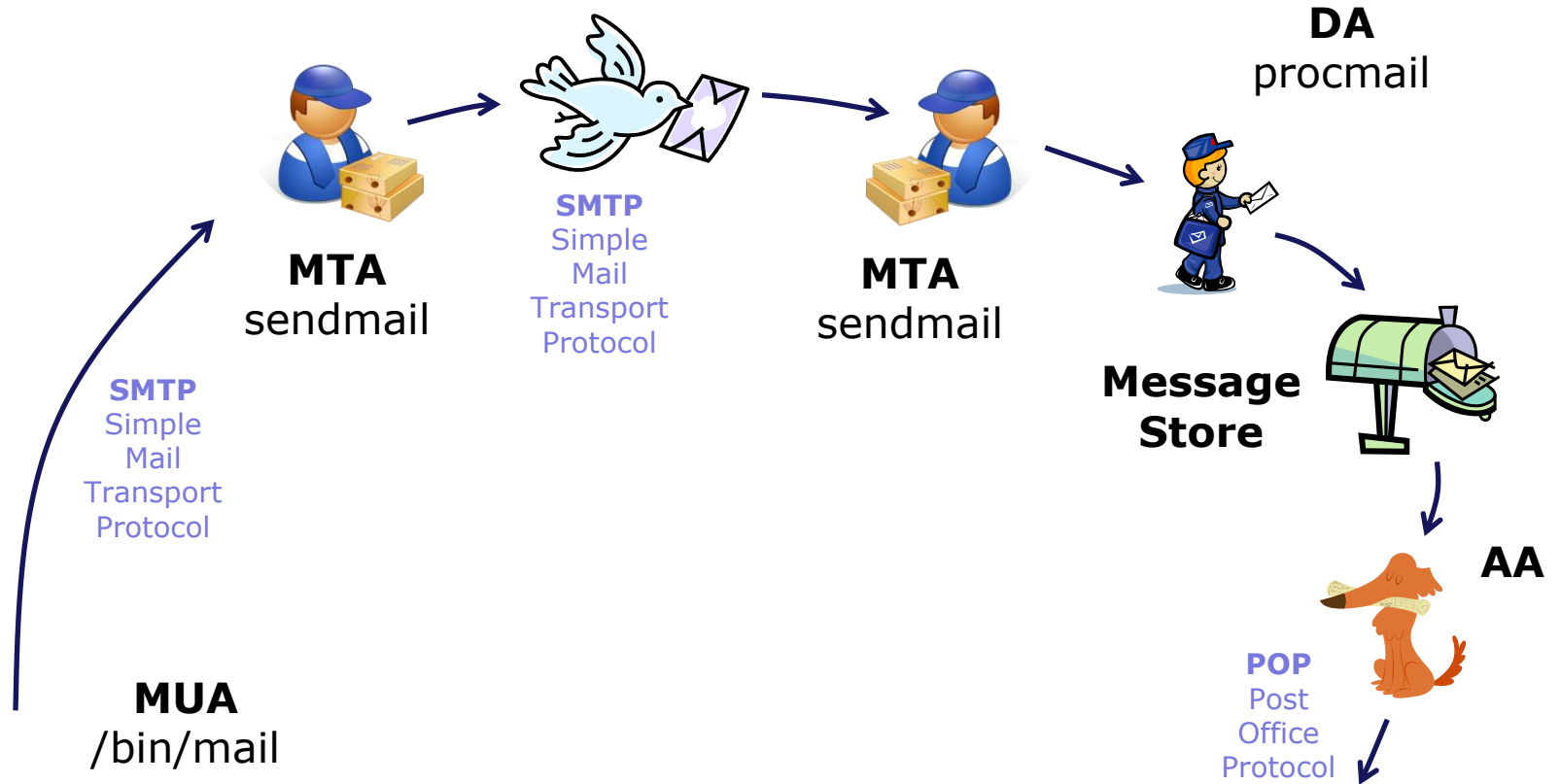
## end-to-end email



# end-to-end email: example Implementation



# end-to-end email: example Implementation



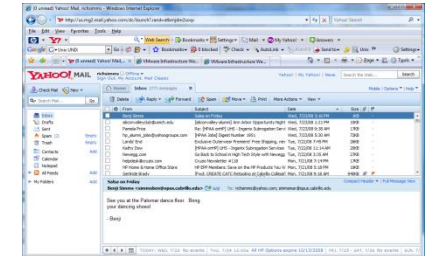
```

simmsben@opus:~
/home/cis90/simmsben $ mail simmsmar richsimms@yahoo.com
Subject: Salsa on Friday
See you at the Palomar dance floor. Bring
your dancing shoes!

- Benji
Cc:
/home/cis90/simmsben $
  
```

compose and send message

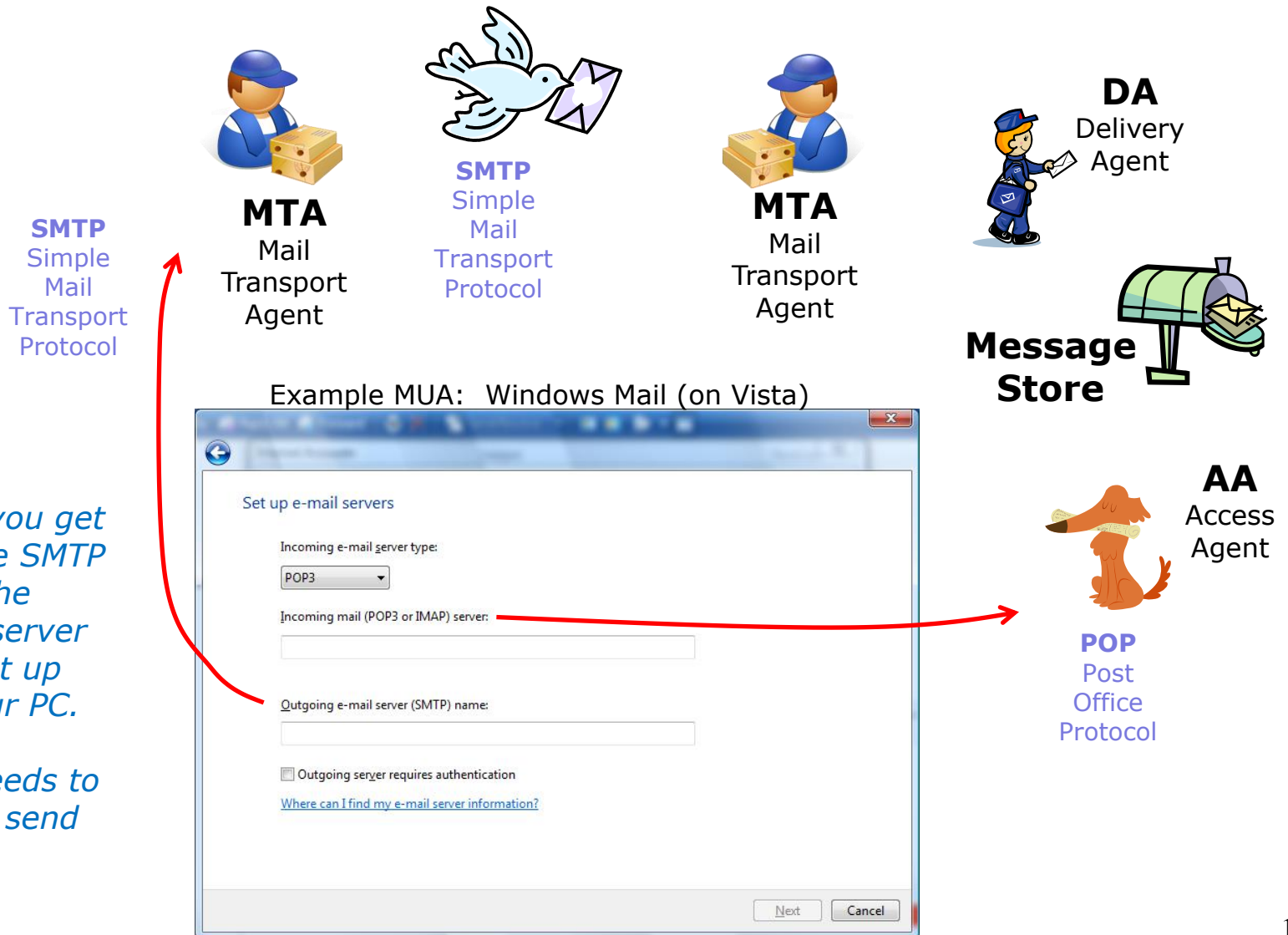
**MUA**  
Yahoo Mail



open and read message 136



# end-to-end email: configuring your MUA (Mail User Agent)



# Other MUAs MTAs, DAs, AAs

## end-to-end email

some of the many players

**MTA**



sendmail, Exim, Microsoft Exchange, Postfix

**DA**



/bin/mail, procmail, smrsh

**AA**



imapd, spop

**MUA**



/bin/mail, pine, elm, Outlook, gmail, Evolution, Yahoo Mail

# Lab 3

## *Notes to Rich*



[ ] - Send out UNIX historical events for Lab 3  
use mail-lab03 script in /cis90/misc/uhist directory

## Lab 3 - Start early and check your Opus email every day!

You will receive another mail message from me that describes a UNIX historical event for a particular year from 1968 to 2003. Save this message to a mail file called *uhistory*.

The objective of this lab is to exchange and collect all the individual events that were sent to each student using UNIX mail.

Start by sending an email to your other classmates with your event and ask them to send you their events. Each time you get a UNIX event that you haven't already saved, save it to your *uhistory* mail file. See how many dates you can accumulate. Can you get all 18?

Rules:

- Do this lab on Opus using `/bin/mail` (the **mail** command).
- When someone asks you for the date that you received, you must send it to them with the subject being the year of the event, e.g. 1972. The email message must contain the complete text of the event for that year.
- Each email saved in *uhistory* must be for a single event/year.

If you receive an email that is missing the event or does not have the year as the subject, reply to the sender and ask them to resend a corrected version.

When you get all the UNIX event messages saved in your *uhistory* mailbox you should have up to 18 messages, each with a different date for the Subject field. Delete any duplicate dates you may have.

*Lab 3 (and all future labs) must be done on Opus*

## Tips for Lab 3

*Start this lab early in the week and check your mail daily to collect all messages*

- Use the **s** command in mail to save a message to your uhistory file
- use **mail -f uhistory** to review your collection
- Use the **d** command in mail to delete duplicates in your uhistory file

*Watch for more tips on the forum*

# Wrap up



## New commands:

### mail

```
type <message list>
next
from <message list>
headers
delete <message list>
undelete <message list>
save <message list> folder
copy <message list> folder
write <message list> file
preserve <message list>
Reply <message list>
reply <message list>
mail addresses
file folder
quit
xit
!
cd <directory>
list
```

### - UNIX mail

```
type messages
goto and type next message
give head lines of messages
print out active message headers
delete messages
undelete messages
append messages to folder and mark as saved
append messages to folder without marking them
append message texts to file, save attachments
keep incoming messages in mailbox even if saved
reply to message senders
reply to message senders and all recipients
mail to specific recipients
change to another folder
quit and apply changes to folder
quit and discard changes made to folder
shell escape
chdir to directory or home if none given
list names of all available commands
```

A <message list> consists of integers, ranges of same, or other criteria separated by spaces. If omitted, mail uses the last message typed.

### mesg

- Enable or disable writes to your terminal

### write

- Write message to another user

## New Files and Directories:

/var/mail

- Message store for mail

/var/mail/*username*

- Incoming mailbox for *username*

## Next Class

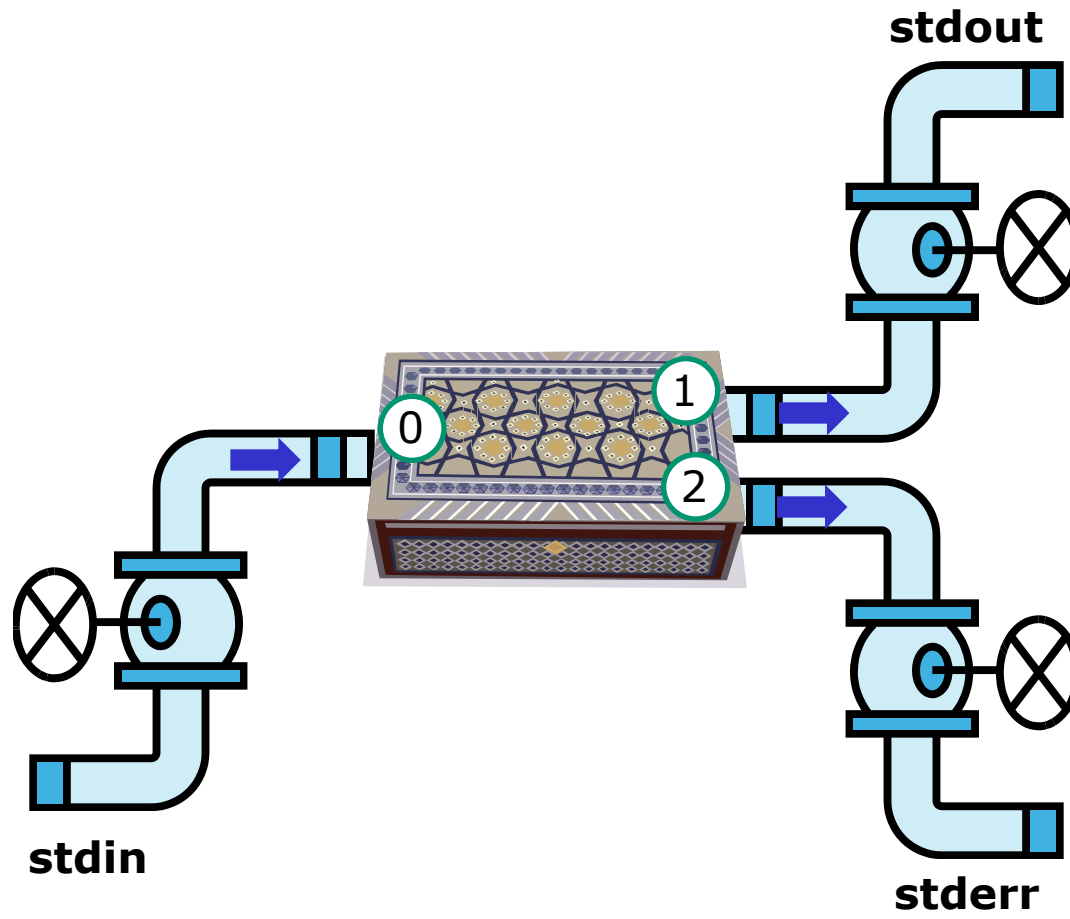
Assignment: Check Calendar Page on web site to see what is due next week.

**1st five forum posts  
and Lab 3**

Quiz questions for next class:

- What command can you use to "chat" with another user?
- How do you forward a message with /bin/mail?
- What is the dead.letter folder?

# Backup



## Practice Test Questions

What is simben90's uid (user ID) on Opus?

## Practice Test Questions

What is simben90's uid (user ID) on Opus?

*Benji's uid is 1201*

```
/home/cis90/simben $ id simben90  
uid=1201(simben90) gid=190(cis90) groups=190(cis90),100(users)  
/home/cis90/simben $
```

## Practice Test Questions

What day of the week was Sept 11, 2001?

## Practice Test Questions

What day of the week was Sept 11, 2001?

*It was a Tuesday*

```
/home/cis90/simben $ cal 9 2001
    September 2001
Su Mo Tu We Th Fr Sa
                1
 2  3  4  5  6  7  8
 9 10 11 12 13 14 15
16 17 18 19 20 21 22
23 24 25 26 27 28 29
30
/home/cis90/simben $
```



## Practice Test Questions

Where (what directory) does the program file for the **ps** command reside?

## Practice Test Questions

Where (what directory) does the program file for the **ps** command reside?

```
/home/cis90/simben $ type ps  
ps is /bin/ps
```

*It's in the /bin directory*



## Practice Test Questions

Parse the following command line. What is the command? How many options and how many arguments are there? What are the options and arguments?

```
ls -l /boot/grub/
```

## Practice Test Questions

Parse the following command line. What is the command? How many options and how many arguments are there? What are the options and arguments?

```
ls -l /boot/grub/
```

*Command: ls*

*One option: -l (for long listing)*

*One argument: /boot/grub*

## Practice Test Questions

Parse the following command line. What is the command? How many options and how many arguments are there? What are the options and arguments?

```
echo "1 2 3" four 5 six
```

## Practice Test Questions

Parse the following command line. What is the command? How many options and how many arguments are there? What are the options and arguments?

```
echo "1 2 3" four 5 six
```

*Command: echo*

*No options*

*4 arguments:*

- *"1 2 3"*
- *four*
- *5*
- *six*

## Practice Test Questions

Which program gave you this error message?

```
/home/cis90/simben $ uname-x  
-bash: uname-x: command not found  
/home/cis90/simben $
```

## Practice Test Questions

Which program gave you this error message?

```
/home/cis90/simben $ uname-x  
-bash: uname-x: command not found  
/home/cis90/simben $
```

*It was the bash program. bash is the shell we are using and it could not find a command named uname-x on the path*



## Practice Test Questions

Which program gave you this error message?

```
/home/cis90/simben $ uname -x  
uname: invalid option -- 'x'  
Try `uname --help' for more information.  
/home/cis90/simben $
```

## Practice Test Questions

Which program gave you this error message?

```
/home/cis90/simben $ uname -x  
uname: invalid option -- 'x'  
Try `uname --help' for more information.  
/home/cis90/simben $
```

*It was the uname program. The uname program was loaded into memory. It started to handle its options and discovered an unknown option. It printed the error message and aborted.*

## Practice Test Questions

What terminal device are you using?

## Practice Test Questions

What terminal device are you using?

Use the tty command to find out:

```
/home/cis90/simben $ tty  
/dev/pts/0  
/home/cis90/simben $
```

## Practice Test Questions

What type of terminal are you using?

## Practice Test Questions

What type of terminal are you using?

Use the **echo \$TERM** command to find out:

```
/home/cis90/simben $ echo $TERM  
xterm
```

*This user's terminal type is xterm*

## Practice Test Questions

What directories make up your path?

## Practice Test Questions

What directories make up your path?

*Use echo \$PATH to find out:*

```
/home/cis90/simben $ echo $PATH  
/usr/lib/qt-3.3/bin:/usr/local/bin:/bin:/usr/bin:  
/usr/local/sbin:/usr/sbin:/sbin:/home/cis90/simben/../bin:  
/home/cis90/simben/bin:.
```

*/usr/lib/qt-3.3/bin  
/usr/local/bin  
/bin  
/usr/bin  
/usr/local/sbin  
/usr/sbin  
/sbin  
/home/cis90/simben/../bin  
/home/cis90/simben/bin  
.*

*There are 10 directories specified on  
this user's path*



## Practice Test Questions

Are the **yum**, **useradd**, and **yell** commands on your path?

## Practice Test Questions

Are the **yum**, **useradd**, and **yell** commands on your path?

```
/home/cis90/simben $ type yum    Yes, on path  
yum is /usr/bin/yum
```

```
/home/cis90/simben $ type useradd  Yes, on path  
useradd is hashed (/usr/sbin/useradd)
```

```
/home/cis90/simben $ type yell    No, not on path  
-bash: type: yell: not found
```

*Note: "is hashed" means bash has previously searched the path and run this command. The location of the command has been saved in the hash table to speed up subsequent searches.*

## Practice Test Questions

What is the name of the environment variable that defines your shell prompt?

## Practice Test Questions

What is the name of the environment variable that defines your shell prompt?

*It's PS1*

```
/home/cis90/simben $ echo $PS1  
$PWD $
```

```
/home/cis90/simben $ echo "The PWD variable =" $PWD  
The PWD variable = /home/cis90/simben  
/home/cis90/simben $
```

*Both PS1 and PS2 are environment variables*

## Practice Test Questions

How do you change the shell prompt to `"Enter next command: "` ?

## Practice Test Questions

How do you change the shell prompt to "Enter next command: " ?

*Set PS1 to new value using "=" sign*

```
/home/cis90/simben $  
/home/cis90/simben $ PS1="Enter next command: "  
Enter next command:  
Enter next command: echo $PWD  
/home/cis90/simben  
Enter next command: echo $PS1  
Enter next command:  
Enter next command:
```

## Practice Test Questions

How do you restore the original shell prompt so it displays the current directory followed by a \$ and a blank?

## Practice Test Questions

How do you change the shell prompt to "Enter next command: " then change it back again?

*To restore the original prompt use:*

```
Enter next command: PS1='$PWD $ '
/home/cis90/simben $
```