Rich's lesson module checklist

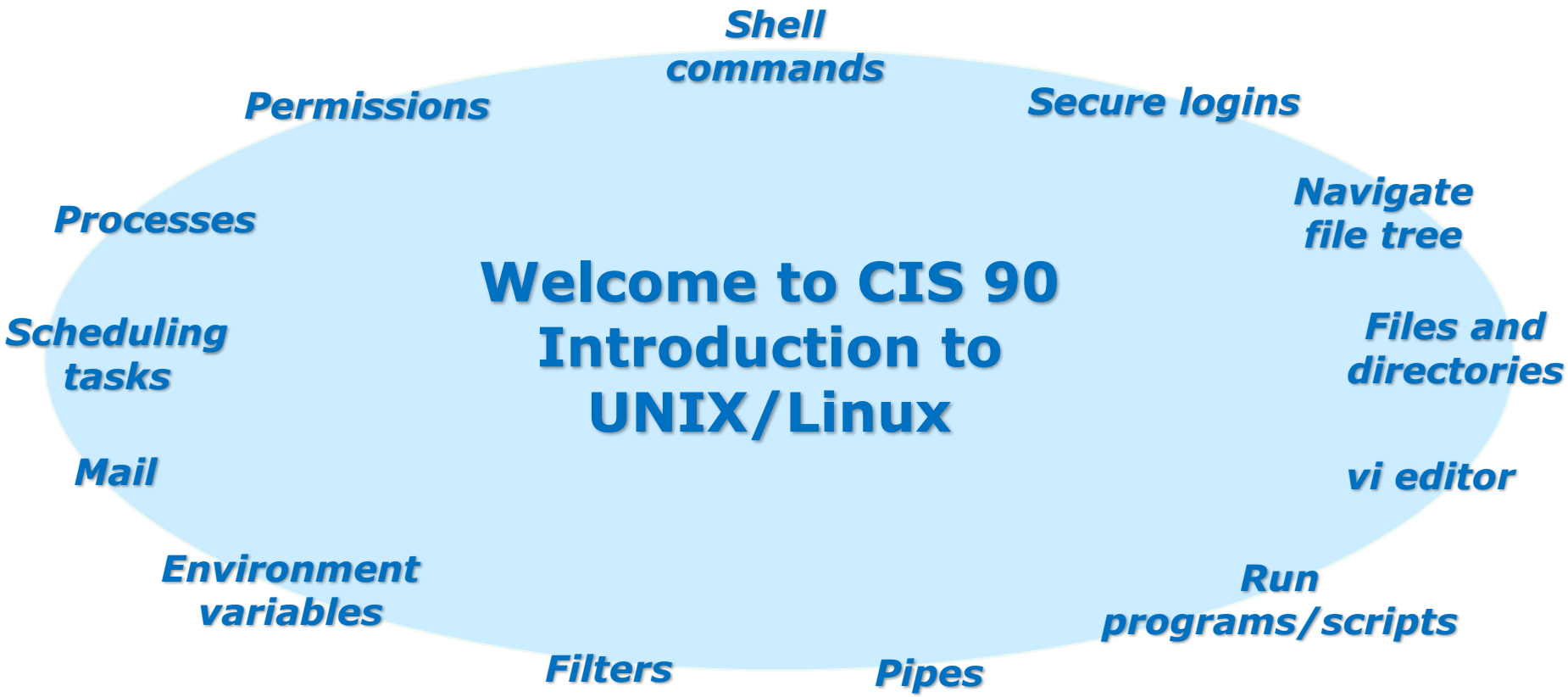
- ❑ Slides and lab posted
- ❑ WB converted from PowerPoint
- ❑ Print out agenda slide and annotate page numbers

- ❑ Flash cards
- ❑ Page numbers
- ❑ 1st minute quiz
- ❑ Web Calendar summary
- ❑ Web book pages
- ❑ Commands

- ❑ Lab 5 and X2 tested
- ❑ Put sonnet6 & bigfile in depot

- ❑ Real Test 1 configures on canvas
- ❑ Real Test 1 systems scheduled access and shutdown
- ❑ Practice Test 1 systems shutdown scheduled

- ❑ 9V backup battery for microphone
- ❑ Backup slides, CCC info, handouts on flash drive
- ❑ Key card for classroom door



Student Learner Outcomes

1. Navigate and manage the UNIX/Linux file system by viewing, copying, moving, renaming, creating, and removing files and directories.
2. Use the UNIX features of file redirection and pipelines to control the flow of data to and from various commands.
3. With the aid of online manual pages, execute UNIX system commands from either a keyboard or a shell script using correct command syntax.

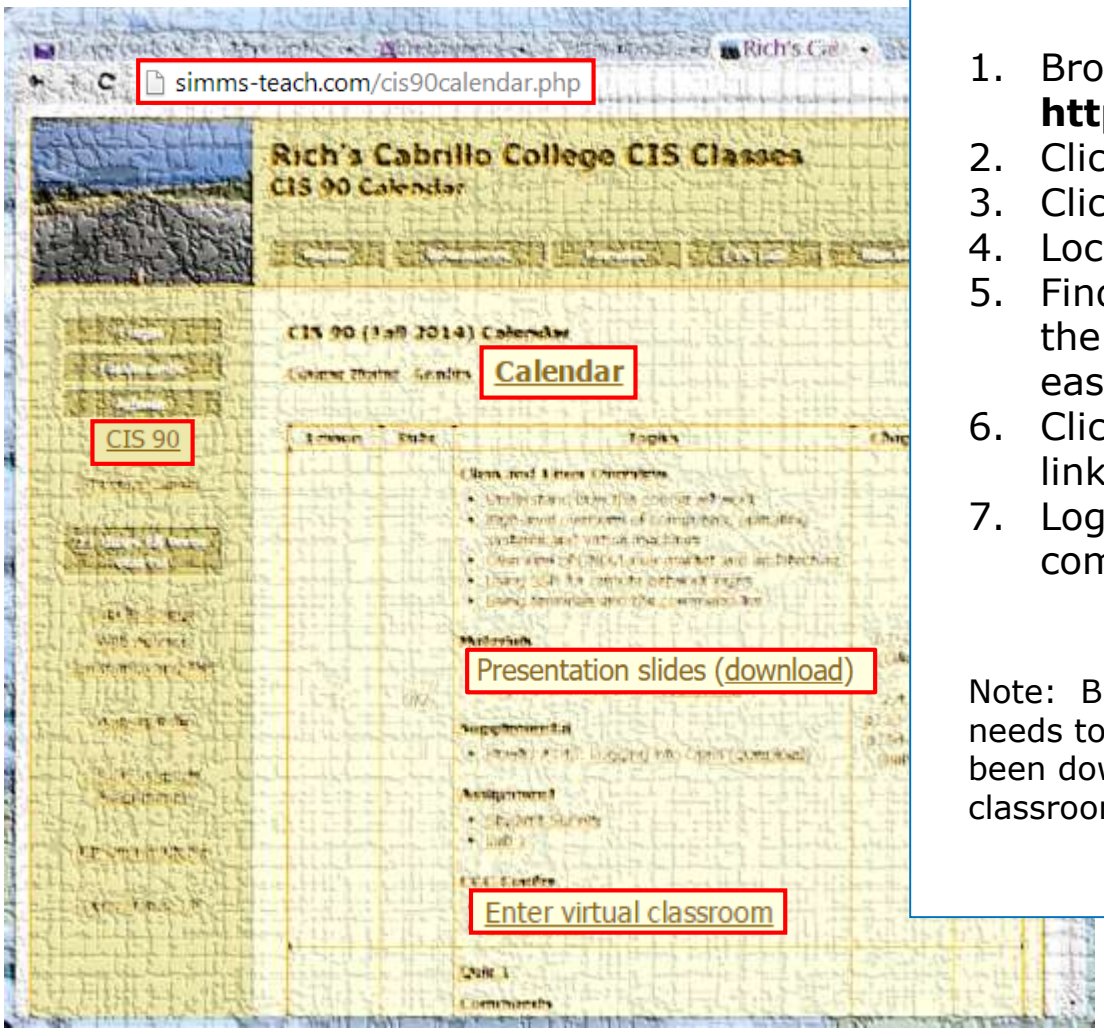
Introductions and Credits

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/)

Student checklist for attending class

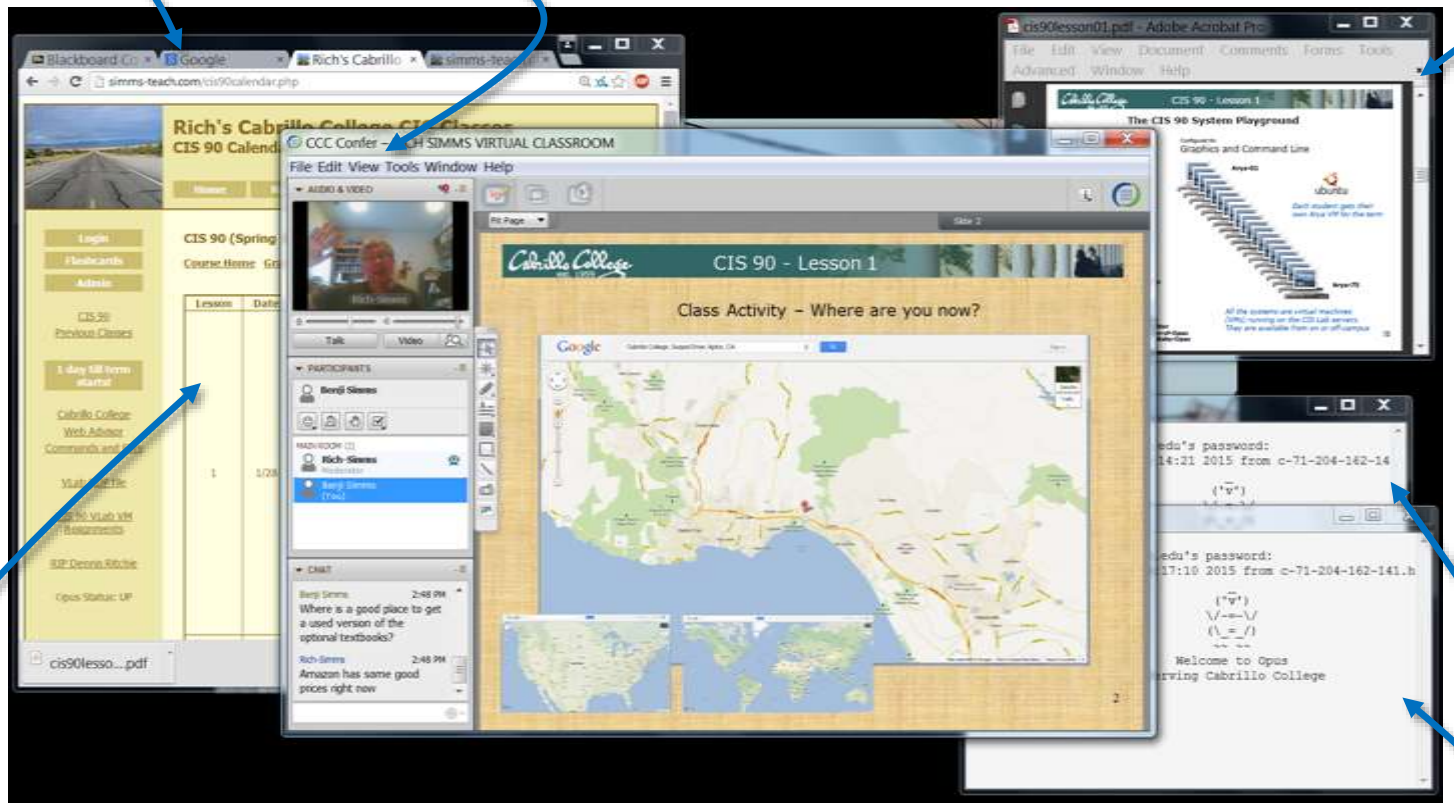


1. Browse to: **http://simms-teach.com**
2. Click the **CIS 90** link.
3. Click the **Calendar** link.
4. Locate today's lesson.
5. Find the **Presentation slides** for the lesson and **download** for easier viewing.
6. Click the **Enter virtual classroom** link to join CCC Confer.
7. Log into Opus with Putty or ssh command.

Note: Blackboard Collaborate Launcher only needs to be installed once. It has already been downloaded and installed on the classroom PC's.

Student checklist for suggested screen layout

- ❑ *Google*
- ❑ *CCC Confer*
- ❑ *Downloaded PDF of Lesson Slides*



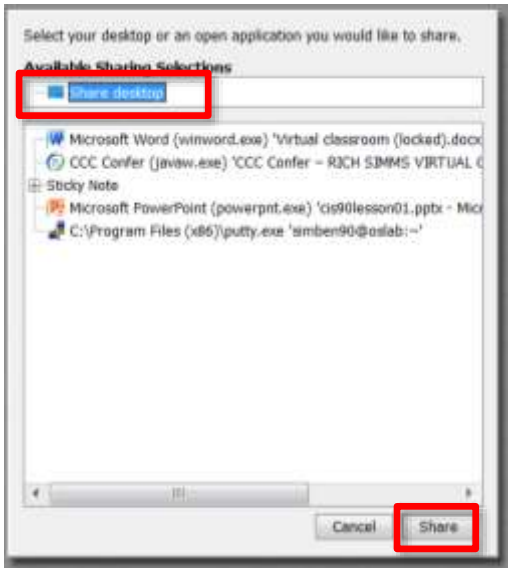
- ❑ *CIS 90 website Calendar page*
- ❑ *One or more login sessions to Opus*

Student checklist for sharing desktop with classmates

1) Instructor gives you sharing privileges

2) Click overlapping rectangles icon. If white "Start Sharing" text is present then click it as well.

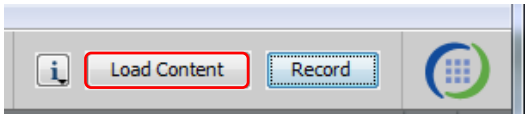
3) Click OK button.



4) Select "Share desktop" and click Share button.

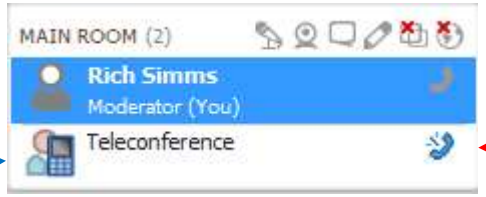
Rich's CCC Confer checklist - setup

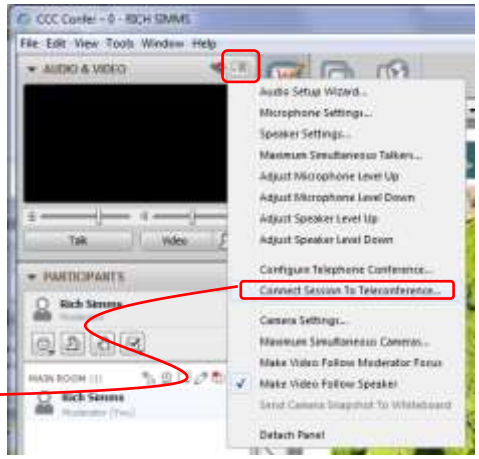
[ ] Preload White Board



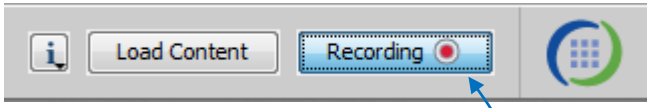
[ ] Connect session to Teleconference

Session now connected to teleconference





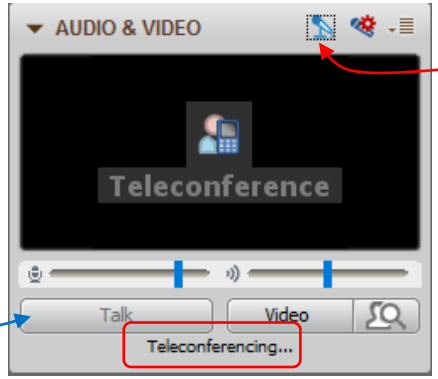
[ ] Is recording on?



Red dot means recording

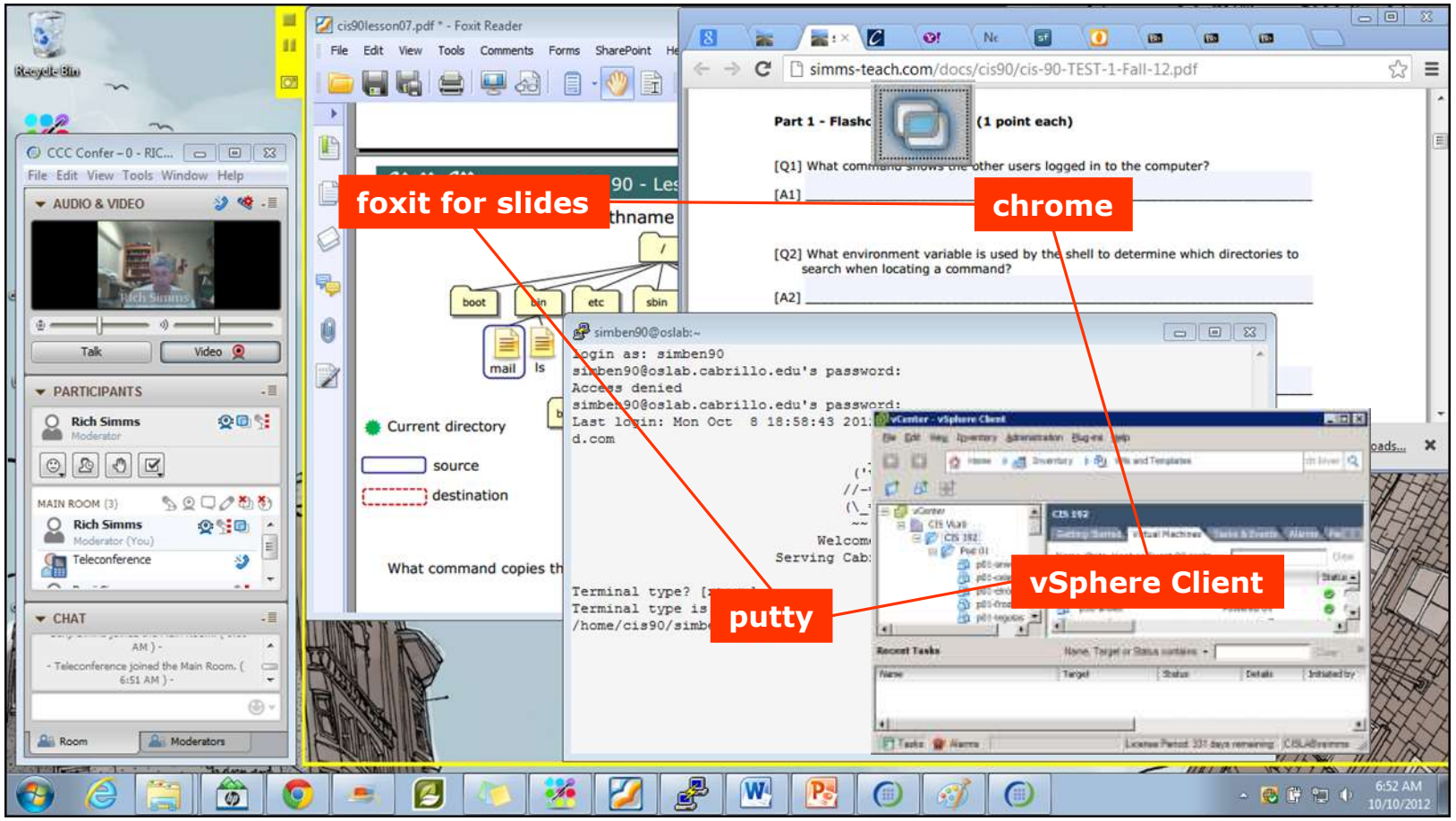
[ ] Use teleconferencing, not mic

Should be grayed out



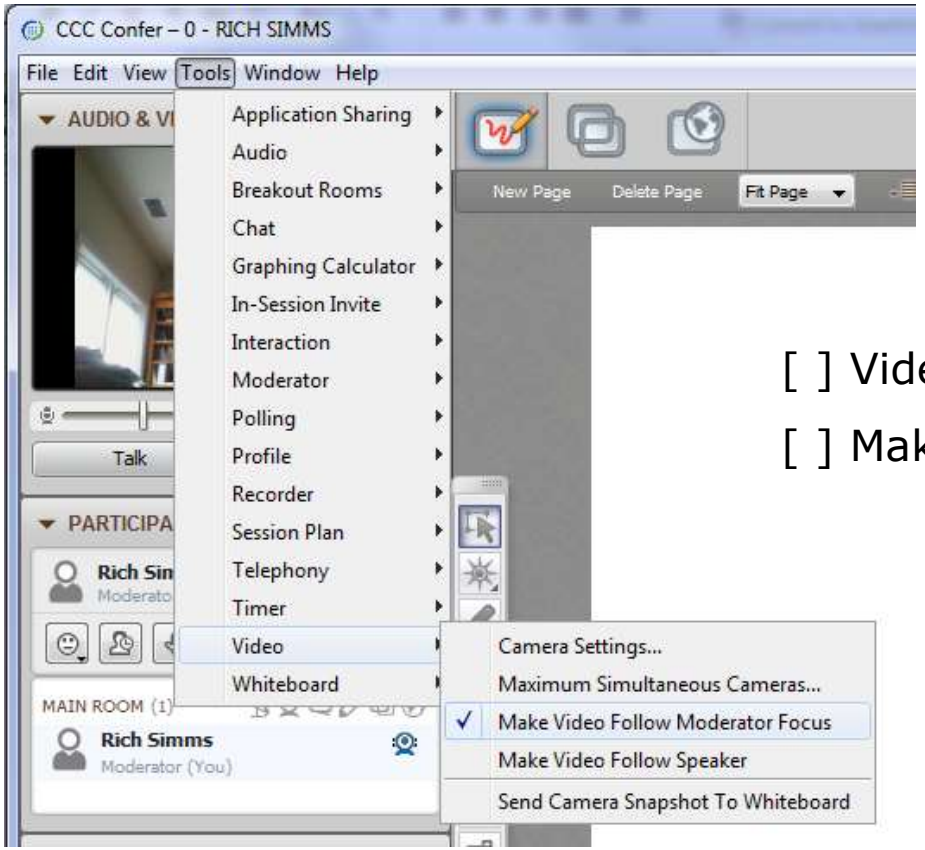
Should change from phone handset icon to little Microphone icon and the Teleconferencing … message displayed

Rich's CCC Confer checklist - screen layout



[ ] layout and share apps

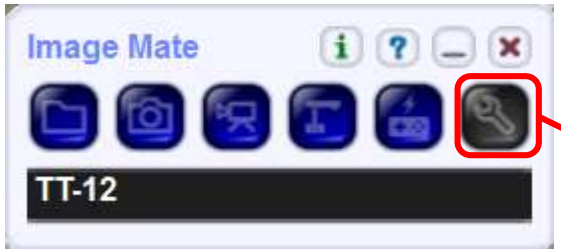
Rich's CCC Confer checklist - webcam setup

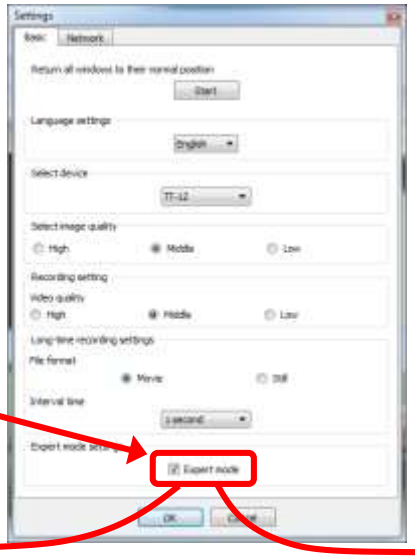


[ ] Video (webcam)

[ ] Make Video Follow Moderator Focus

Rich's CCC Confer checklist - Elmo





The "rotate image" button is necessary if you use both the side table and the white board.

Quite interesting that they consider you to be an "expert" in order to use this button!

Elmo rotated down to view side table

Elmo rotated up to view white board

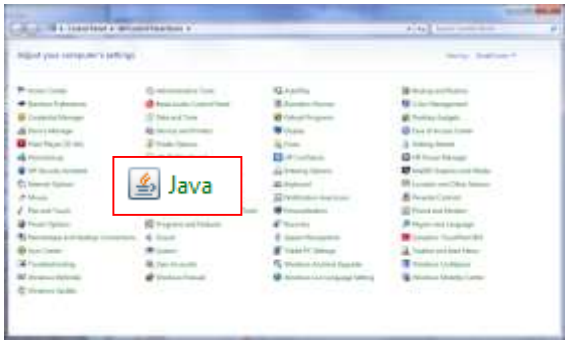
Run and share the Image Mate program just as you would any other app with CCC Confer

Rich's CCC Confer checklist - universal fixes

Universal Fix for CCC Confer:

1) Shrink (500 MB) and delete Java cache
2) Uninstall and reinstall latest Java runtime
3) http://www.cccconfer.org/support/technicalSupport.aspx

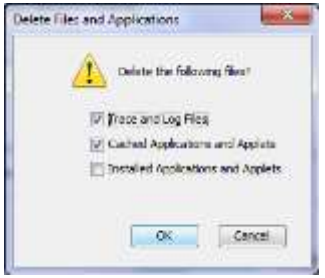
Control Panel (small icons)



General Tab > Settings…

500MB cache size

Delete these



Google Java download

Start

Sound Check

*Students that dial-in should mute their line using *6 to prevent unintended noises distracting the web conference.*

*Instructor can use *96 to mute all student lines.*

CIS 90 - Lesson 6

Instructor: **Rich Simms**
Dial-in: **888-886-3951**
Passcode: **136690**


Jacob

Ethan

Amr


Becka


Brenda


Nikki


Brad

Tyler

Justin

Nick

Cody

Miles


Carrie

Danny

Steven


Wes


Jade


Brandon


Alan


Bryanda

Max


Nicole

Email me (risimms@cabrillo.edu) a relatively current photo of your face for 3 points extra credit

First Minute Quiz

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

No Quiz today ... test instead

For credit email answers to:

risimms@cabrillo.edu

within the **first few minutes of class**

Managing Files

Objectives	Agenda
• Be able to create, copy, move, remove and link files	• Questions • Housekeeping • Managing files • Creating directories • Creating regular files • Listing files • Copying files • Moving Files • Removing files • Linking files • Assignment • Wrap up • Test #1

Questions

Questions?

Lesson material?

Labs? Tests?

How this course works?

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

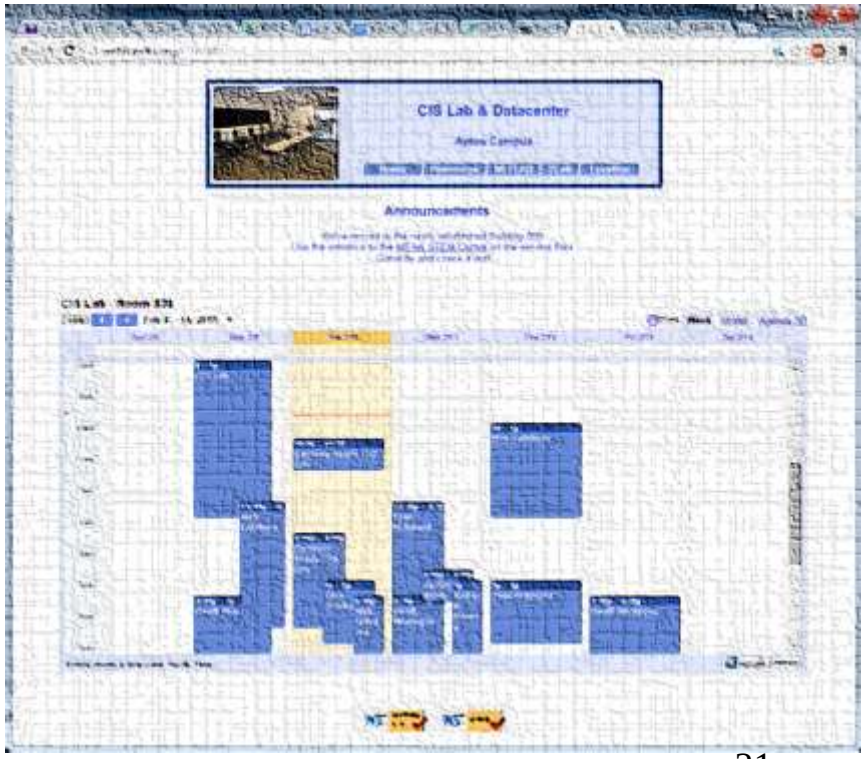
Who questions much, shall learn much, and retain much.
- Francis Bacon

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.*

Got stuck or having trouble getting started in this course?

If you would like some additional come over to the CIS Lab. There are student lab assistants and instructors there to help you.



Takashi, Melissa, Sam and Andrew are CIS 90 Alumni.

Chris is a STEM tutor and can help you Mondays 3-9 and Wednesdays 4-9.

Mike Matera is the other Linux instructor.

I'm in there Mondays 10:00-12:30.

Housekeeping

No labs due today

Test 1 will become available at 11:00 AM today

- Open book, open notes, open computer.
- You must work alone and not help or receive help from others.
- Online timed 60 minute test using Canvas
- Online "archive watching" students that work can take it later today but it must be completed by 11:59 PM.
- Practice test systems shutdown 30 minutes before real test starts!

Next week:

- Quiz 5
- Lab 5 is due

Test 1 Instructions

HONOR CODE:

This test is open book, open notes, and open computer. HOWEVER, you must work alone. You may not discuss the test questions or answers with others during the test. You may not ask or receive assistance from anyone other than the instructor when doing this test. Likewise you may not give any assistance to anyone taking the test.

INSTRUCTIONS:

Every question on the test was designed to be answered using one of the systems below.

1. oslab.cis.cabrillo.edu (port 2220) - This server is named Opus internally.
2. sun-hwa-vii.cis.cabrillo.edu (port 22)
3. son-of-opus.simms-teach.com (port 2220)
4. arya-xx (port 22) - Select xx for your own Arya.

Each question begins with [*system name*] so you know which system you should be logged into to answer the question.

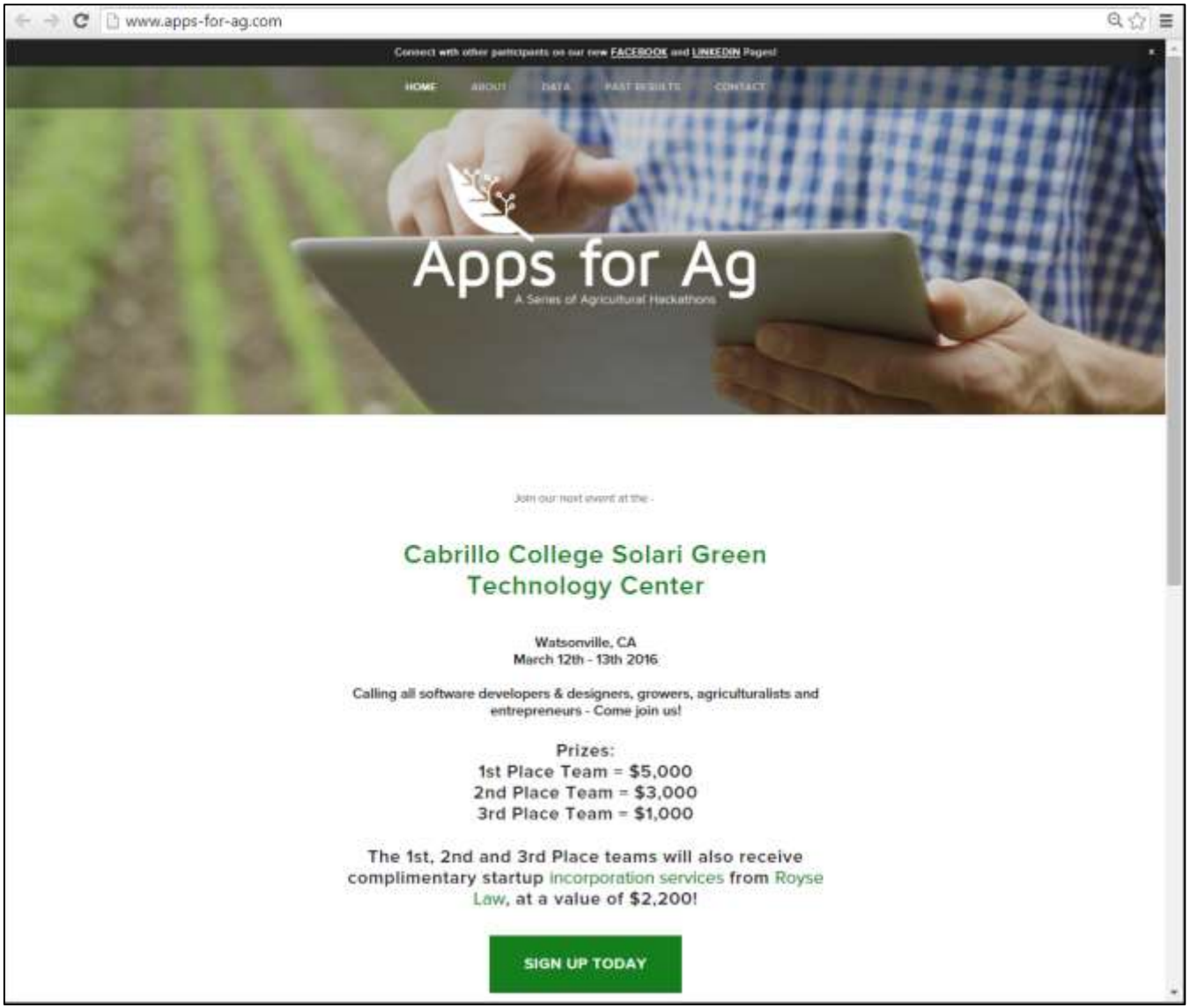
All systems are accessible using ssh from opus. For sun-hwa-vii and son-of-opus login using your original opus credentials. For arya, use the generic cis90 account.

IF YOU GET STUCK on a question you can ask the instructor for the answer and forfeit the points. The instructor will be available during class and be online between 8-10 PM in the evening for online or long distance students.

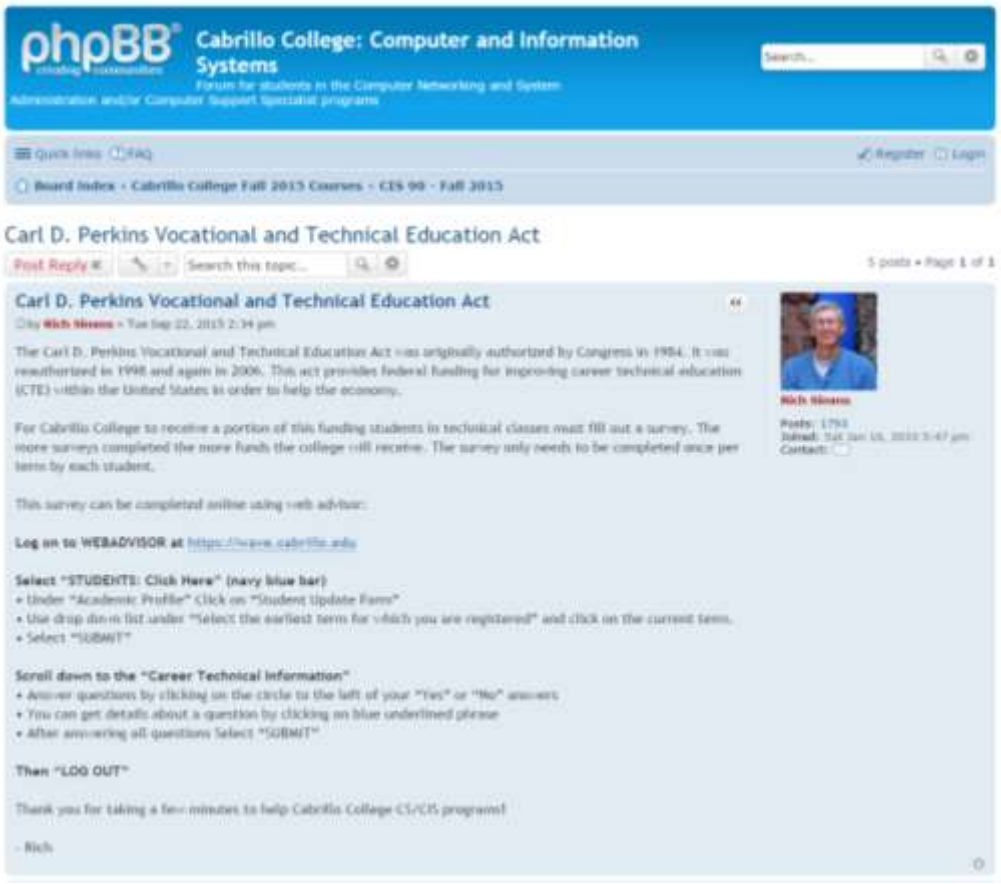
Please KEEP YOUR ANSWERS TO A SINGLE LINE ONLY !!

This test must be completed in one sitting. The submittal will be made automatically when the time is up. If you submit early by accident you will not be able to re-enter and continue. If that happens don't panic! Just email the instructor any remaining answers before the time is up.

http://www.apps-for-ag.com/



Don't Forget -- Perkins/VTEA Survey

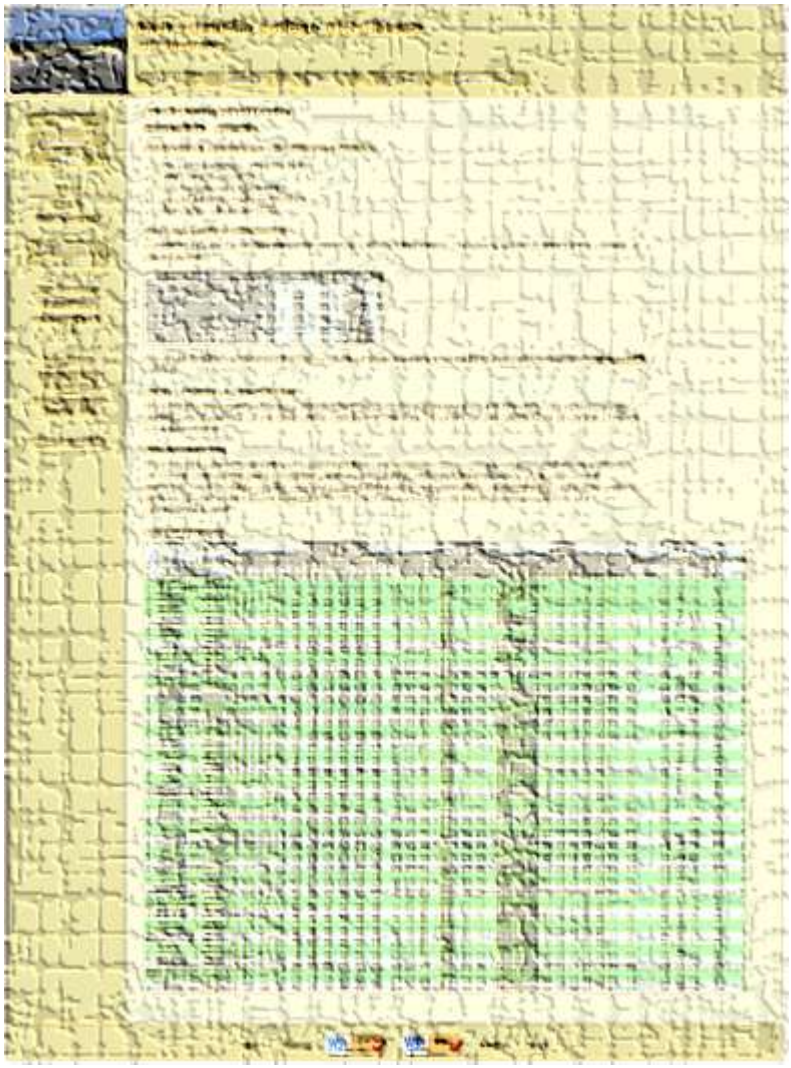


http://oslab.cis.cabrillo.edu/forum/viewtopic.php?f=118&t=3976

This is an important source of funding for Cabrillo College.

Send me an email stating you completed this survey for ***three points extra credit!***

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



GRADES

Be sure and check your progress on the Grades page as the course continues on.

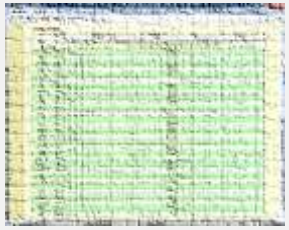
Send me a student survey if you haven't already to get your LOR secret code name.

Where to find your grades

Send me your survey to get your LOR code name.

The CIS 90 website Grades page

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



Points that could have been earned:

4 quizzes:	12 points
4 labs:	120 points
1 forum quarter:	20 points
Total:	**152 points**

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

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

Or check on Opus

checkgrades *codename*

(where codename is your LOR codename)

Written by Jesse Warren a past CIS 90 Alumnus

grades *codename*

(where codename is your LOR codename)

Written by Sam Tindell a past CIS 90 Alumnus. Try his tips, schedule and forums scripts as well!

Managing Files

Lesson 6 commands for your toolbox:

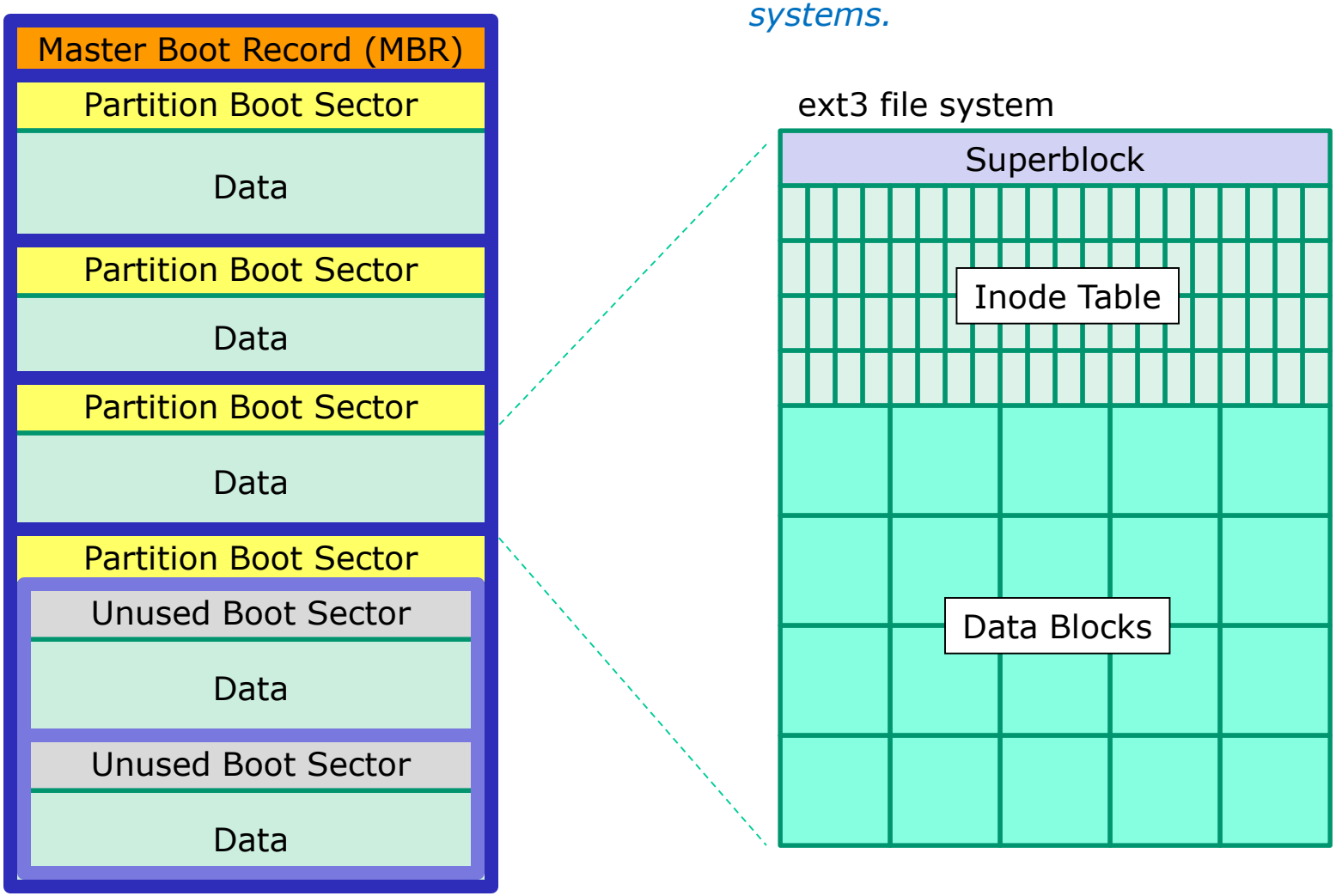
touch	- make a file (or update the timestamp)
mkdir	- make a directory
cp	- copy a file
mv	- move or rename a file
rmdir	- remove a directory
rm	- remove a file
ln	- create a link
tree	- visual list a directory

Redirecting stdout:

> filename	- redirecting stdout to create/empty a file

File Systems
Linux

The hard drive is partitioned and the data areas can be formatted as a file system. Linux typically uses ext2, ext3 and ext4 file systems. Windows uses FAT32 and NTFS file systems.



UNIX Files
The three elements of a file

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

filename

+

```
/home/cis90/simben/Poems $ ls -li twister
102625 -rw-r--r-- 1 simben90 cis90 151 Jul 20  2001 twister
```

inode number — *inode information*

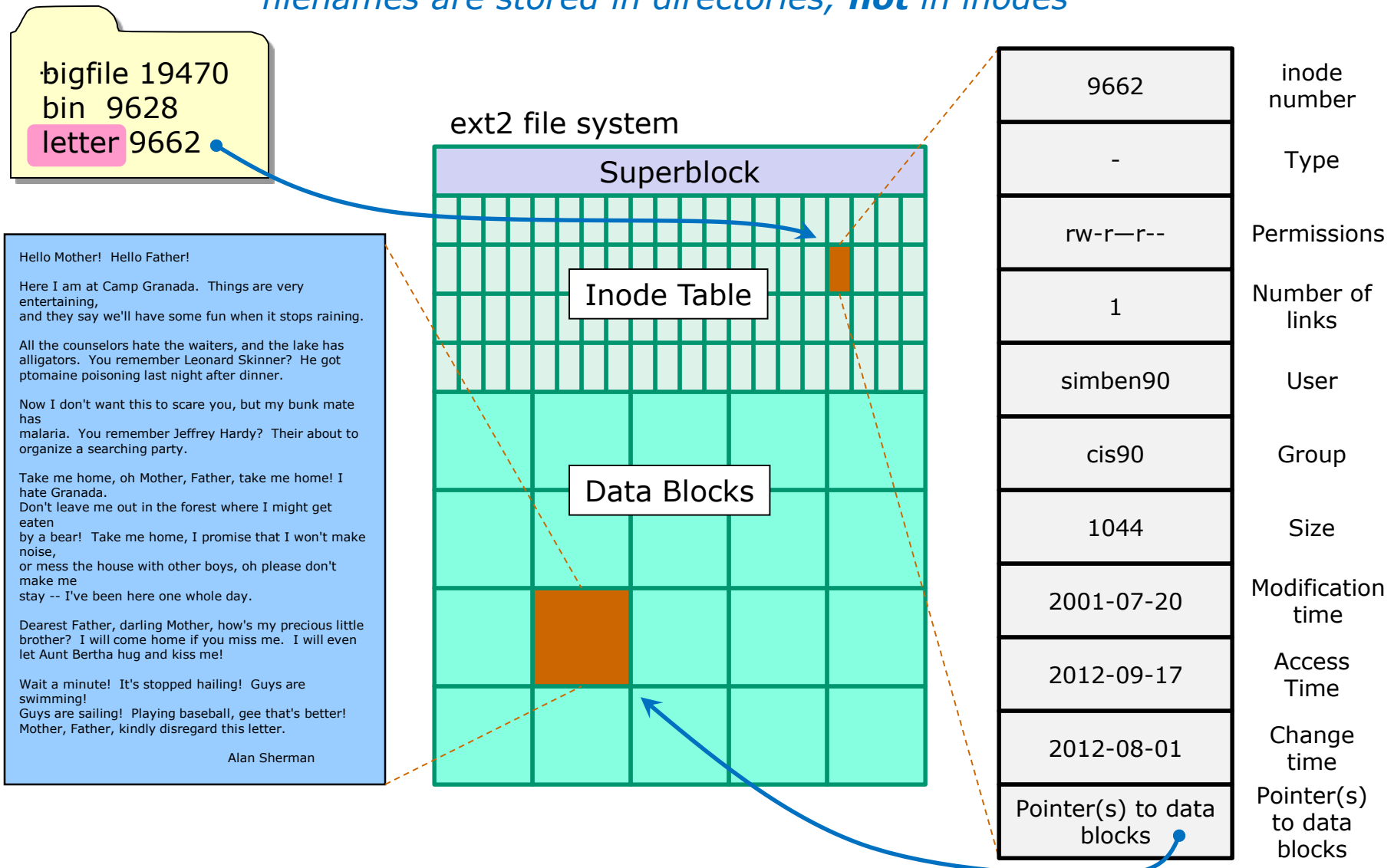
inode

+

```
/home/cis90/simben/Poems $ cat twister
A tutor who tooted the flute,
tried to tutor two tooters to toot.
Said the two to the tutor,
"is it harder to toot?  Or to
tutor two tooters to toot?"
```

data

*filenames are stored in directories, **not** in inodes*




```
/home/cis90/simben $ ls -il letter
9662 -rw-r--r--. 1 simben90 cis90 1044 Jul 20  2001 letter
```

Creating Directories

Creating Directories

Command syntax:

mkdir <*new-directory-name*>

- creates an empty directory(s)
- options: -p (to create nested directories)

Remember, everything in Unix is a file … even directories!

Creating Directories

The mkdir command

mkdir *<new-name>*

Create a new directory named island

```
/home/cis90/simben $ ls -l island
ls: island: No such file or directory

/home/cis90/simben $ mkdir island
/home/cis90/simben $ ls -ld island
drwxrwxr-x 2 simben90 cis90 4096 Mar 18 06:43 island
```

*Note: Use the **d** option on the **ls** command to list information about the directory itself rather than directory contents*

The basic file type is a directory

The file owner is a simben90

The file size is 4096 bytes

Creating Directories

The mkdir command

Create multiple directories at once

```
/home/cis90/simben $ mkdir redhat debian slackware
/home/cis90/simben $
/home/cis90/simben $
/home/cis90/simben $
/home/cis90/simben $
/home/cis90/simben $
/home/cis90/simben $ ls -ld  redhat/ debian/  slackware/
drwxrwxr-x 2 simben90 cis90 4096 Mar 17 09:36 debian/
drwxrwxr-x 2 simben90 cis90 4096 Mar 17 09:36 redhat/
drwxrwxr-x 2 simben90 cis90 4096 Mar 17 09:36 slackware/
```

*Note: Use the **d** option on the **ls** command to list information about the directories themselves rather than their contents*

Column 1 of the long listing shows the basic file type is a "d" for directory

Creating Directories

The mkdir command

Create nested directories (one directory inside another)

```
/home/cis90/simben $ mkdir africa/ghana
mkdir: cannot create directory `africa/ghana': No such file
  or directory

/home/cis90/simben $ mkdir -p africa/ghana
/home/cis90/simben $ ls africa
ghana
```

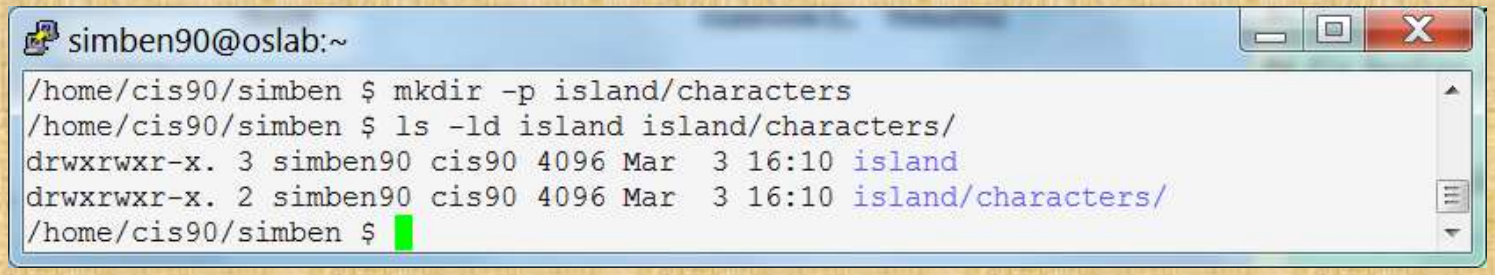
*Need to use the **p** option to create new parent directories as needed*

Activity

In your home directory create a directory named *characters* inside a directory named *island* then list both new directories:

```
mkdir -p island/characters

ls -ld island island/characters/
```



Creating Regular Files

Creating Files

Command syntax:

touch <*new-filename*>

- creates an empty ordinary file(s), or if the file already exists, it updates the time stamp.

echo "*string***" >** <*new-filename*>

- Creates or overwrites a text file

Creating Files

The touch command

touch *<new-name>*

Creates one or more empty regular files, or if the file already exists, it updates the time stamp.

```
/home/cis90/simben $ ls -l sawyer
ls: sawyer: No such file or directory

/home/cis90/simben $ touch sawyer
/home/cis90/simben $ ls -l sawyer
-rw-rw-r-- 1 simben90 cis90 0 Mar 18 06:34 sawyer
```

The file type is a regular file

The file owner is simben90

The file size is 0 bytes (an empty file)

Creating Files
The touch command

Multiple files can be created with one command

```
/home/cis90/simben $ ls -l a b c
ls: a: No such file or directory
ls: b: No such file or directory
ls: c: No such file or directory

/home/cis90/simben $ touch a b c
/home/cis90/simben $ ls -l a b c
-rw-rw-r-- 1 simben90 cis90 0 Mar 17 09:27 a
-rw-rw-r-- 1 simben90 cis90 0 Mar 17 09:27 b
-rw-rw-r-- 1 simben90 cis90 0 Mar 17 09:27 c
```

multiple arguments allowed

Column 1 of the long listing shows the basic file type is a "-" for regular file

Creating Files

The touch command

The "last modified" timestamp is updated if the file already exists

```
/home/cis90/simben $ ls -l sawyer
-rw-rw-r-- 1 simben90 cis90 0 Mar 18 06:34 sawyer
```

Wait a few minutes then touch the file to update the timestamp

```
/home/cis90/simben $ touch sawyer
/home/cis90/simben $ ls -l sawyer
-rw-rw-r-- 1 simben90 cis90 0 Mar 18 06:40 sawyer
```

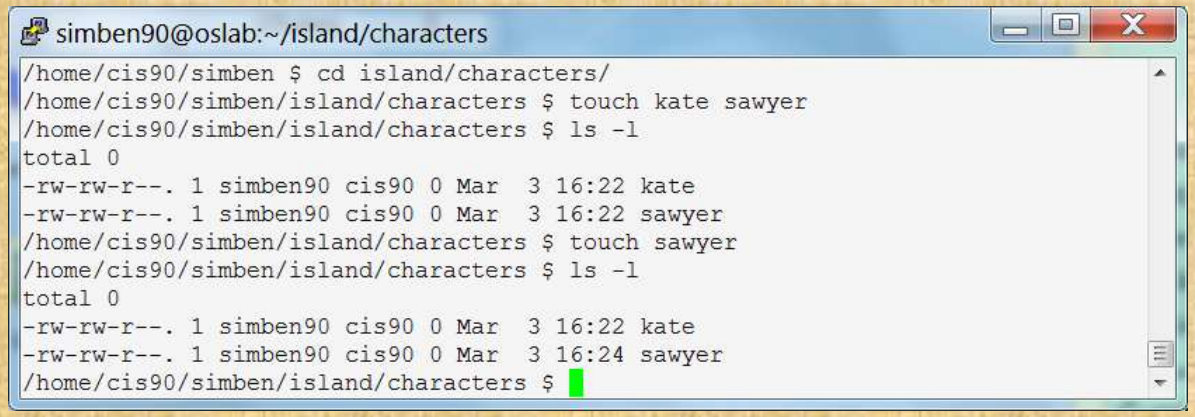
Activity

In the directory named *characters* create 2 new files:

```
cd island/characters
touch kate sawyer
ls -l
```

wait a minute or two

```
touch sawyer
ls -l
```



Creating Files

Redirection to stdout

echo "string" > newfile Creates or overwrites a text file

Creating a file named accra and adding some text to it

```
/home/cis90/simben $ cd africa
/home/cis90/simben/africa $ ls
ghana
/home/cis90/simben/africa $ cd ghana
/home/cis90/simben/africa/ghana $ echo Population 1,658,937 > accra
/home/cis90/simben/africa/ghana $ cat accra
Population 1,658,937
```

Output of the echo command is redirected from the screen to a file named accra

Creating Files

Redirection to stdout


```
/home/cis90/simben/africa/ghana $ cat accra
Population 1,658,937
/home/cis90/simben/africa/ghana $ > accra
/home/cis90/simben/africa/ghana $ cat accra
/home/cis90/simben/africa/ghana $
```

The redirection character > will create the file named if that file does not exist. ***If the file does exist it will be emptied without warning!***

Activity

- In the directory named *characters* create a new file:

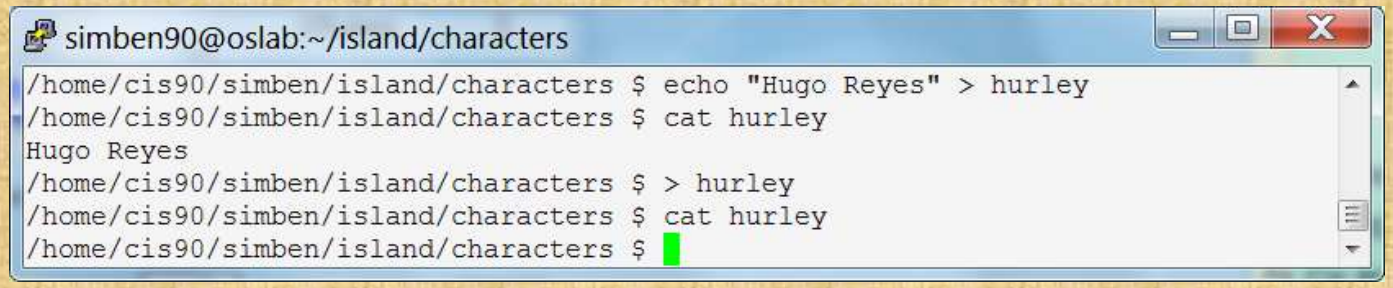
```
echo "Hugo Reyes" > hurley
```

- Print the new file with:

```
cat hurley
```

- Empty the file *hurley*

```
> hurley
cat hurley
```



Listing Files

Listing Files & Directories

Short listing

```
/home/cis90/simben $ ls island
characters
```

Short recursive listing

```
/home/cis90/simben $ ls -R island
island:
characters

island/characters:
hurley  kate  sawyer
```

Listing Files & Directories

Long listing

```
/home/cis90/simben $ ls -l island
total 4
drwxrwxr-x. 2 simben90 cis90 4096 Mar  3 16:53 characters
```

Long recursive listing

```
/home/cis90/simben $ ls -lR island
island/:
total 4
drwxrwxr-x. 2 simben90 cis90 4096 Mar  3 16:53 characters

island/characters:
total 0
-rw-rw-r--. 1 simben90 cis90 0 Mar  3 16:53 hurley
-rw-rw-r--. 1 simben90 cis90 0 Mar  3 16:22 kate
-rw-rw-r--. 1 simben90 cis90 0 Mar  3 16:24 sawyer
```

Listing Files & Directories

Making a directory tree diagram

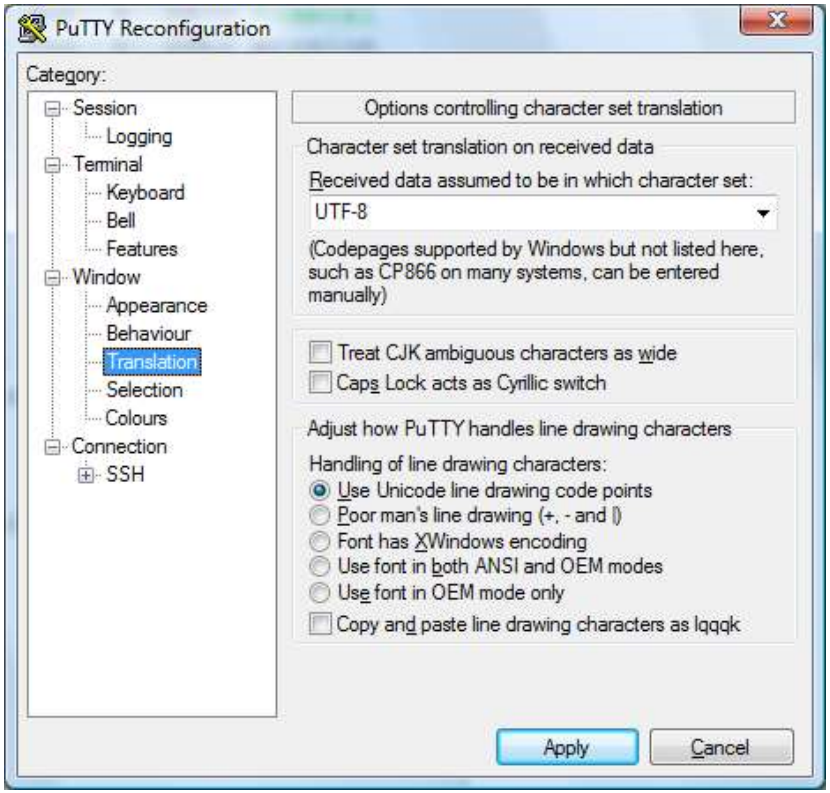
```
/home/cis90/simben $ tree island
island
`-- characters
    |-- hurley
    |-- kate
    `-- sawyer

1 directory, 3 files
/home/cis90/simben $
```

Putty must be configured to use the UTF-8 translation to show line drawing characters

Managing the UNIX/Linux File System

Putty may need to be configured UTF-8 for tree command



Activity

- Return to your home directory with:

```
cd
```

- Do a long listing of the *island* directory with:

```
ls island
```

- Do a long recursive listing of the *island* directory with:

```
ls -lR island
```

- Make tree diagram of the *island* directory with:

```
tree island
```

Copying Files

Copying files
The **cp** command

Command syntax:

cp *<source file> <target file>*

cp *<source file> <target directory>*

cp *<source file> <target directory>/<target file>*

cp *<source file> <source file> <target directory>*

options: **-i -r**

i = warn before overwriting target files
r = recursive (copies all source sub-directories)

*Where: <source file> <target file> <target directory> are **absolute** or **relative** pathnames*

Copying files

Copy one file to another

cp *<source file> <target file>*

```
/home/cis90/simben $ cd
/home/cis90/simben $ cd island/characters/
/home/cis90/simben/island/characters $ ls
hurley  kate  sawyer
/home/cis90/simben/island/characters $ echo "Hugo Reyes" > hurley
```

Make a copy of the hurley file

```
/home/cis90/simben/island/characters $ cp hurley hurley.bak
/home/cis90/simben/island/characters $ ls
hurley  hurley.bak  kate  sawyer
```

Copying files

Copy multiple files to a directory

cp *<source file> <source file> <target directory>*

```
/home/cis90/simben/island/characters $ ls
hurley  hurley.bak  kate  sawyer
```

Make a new directory called backup

```
/home/cis90/simben/island/characters $ mkdir backup
```

Copy three files of the four files to the new directory

```
/home/cis90/simben/island/characters $ cp hurley kate sawyer backup/
/home/cis90/simben/island/characters $ ls backup
hurley  kate  sawyer
```

Copying files

Copy multiple files to a directory

cp *<source file> <source file> <target directory>*

*While parsing the shell expands * to hurley hurley.bak kate sawyer*

Copy all files to the new directory

```
/home/cis90/simben/island/characters $ cp * backup/
cp: omitting directory `backup'
```

*Although * matches backup, it is not included in the copy*

List the four files in the new directory

```
/home/cis90/simben/island/characters $ ls backup/
hurley  hurley.bak  kate  sawyer
```

Note: copying a file to an existing file will overwrite that file without warning!

Copy files

The i (interactive) option to warn about overwrites

```
/home/cis90/simben/island/characters $ ls h*
hurley  hurley.bak
/home/cis90/simben/island/characters $ cp -i hurley hurley.bak
cp: overwrite `hurley.bak'? yes
/home/cis90/simben/island/characters $
```

*The **i** option provides some interaction with the user before overwriting a file*

Copying files

The r (recursive) option to copy an entire tree branch

```
/home/cis90/simben/island/characters $ cd ..
/home/cis90/simben/island $ ls
characters
```

This directory does not exist yet

```
/home/cis90/simben/island $ cp -r characters players
/home/cis90/simben/island $ ls -R players
players:
backup  hurley  hurley.bak  kate  sawyer

players/backup:
hurley  hurley.bak  kate  sawyer
/home/cis90/simben/island $
```

A recursive copy will copy everything in a directory (including all files and nested subdirectories) to another directory

Class Exercise

- Change to your *island* directory using an absolute path

```
cd /home/cis90/simben/island/characters/
```

Use your own username

- Make a backup copy of *kate*

```
cp kate kate2
```

- Copy *hurley* and overwrite *kate* using interactive mode

```
cp -i hurley kate
cat kate
```

(Respond with **yes** to overwrite)

- Restore *kate* from the backup copy

```
cp kate2 kate
cat kate
```

Moving Files

Moving Files

The **mv** command

Command syntax:

mv *<source file> <target file>*

mv *<source file> <target directory>*

mv *<source file> <target directory>**/**<target file>*

mv *<source file> <source file> <target directory>*

options: **-i**
i = warn before overwriting

*Where: <source file> <target file> <target directory> are **absolute** or **relative** pathnames*

Moving Files

Renaming a file with the **mv** command

mv *<original name> <new name>*

This is how you rename files in UNIX/Linux!

oops … typo!

```
/home/cis90/simben $ touch iPhone iPad ProLiant Pavilion Powerege

/home/cis90/simben $ mv Powerege PowerEdge

/home/cis90/simben $ ls iP* P[ra]* Pow*
iPad  iPhone  Pavilion  PowerEdge  ProLiant
```

typo fixed by renaming file

successfully renamed

Moving Files

Moving a file into a directory

mv *<source file> <target directory>*

```
/home/cis90/simben $ mkdir Apple HP Dell
```
Make some new directories

```
/home/cis90/simben $ mv iPhone Apple/
/home/cis90/simben $ mv iPad Apple/
```
Move one file at a time into one of the new directories

```
/home/cis90/simben $ ls Apple
iPad  iPhone
```
List the new directory the files were moved into

Moving Files

Moving multiple files into a directory

mv *<source file> <source file> <target directory>*

```
/home/cis90/simben $ mv ProLiant Pavilion PowerEdge HP/
```

Moving multiple files at once into a directory

Moving Files

The **mv** command

Listing the contents of multiple directories to verify file moves

```
/home/cis90/simben $ ls Apple HP Dell
Apple:
iPad  iPhone

Dell:
PowerEdge

HP:
Pavilion  ProLiant
```

```
/home/cis90/simben $ tree Apple HP Dell
Apple
|-- iPad
`-- iPhone
HP
|-- Pavilion
`-- ProLiant
Dell
`-- PowerEdge

0 directories, 5 files
```

Class Exercise

- Change to your *island* directory using an relative path

```
cd
cd island/characters/
```

- Rename *kate* to *katherine*

```
mv kate katherine
cat katherine
```

- Create a new file named *jin* and rename it to be hidden

```
touch jin
mv jin .jin
```

(verify with `ls` and `ls -a`)

Removing Files

Removing Files

The **rm** and **rmdir** commands

Removing files:

rm <*file-pathname*> **...**

The … (ellipses) mean you can specify more than one filename per command

options: **-i -r -f**

i = prompt before remove
r = recursive (delete subdirectories)
f = force (never prompt)

rmdir <*directory-pathname*> **...**

Directories must be empty for this to work

Removing Files

The **rm** and **rmdir** commands

Remove a file:

```
/home/cis90/simben $ touch junk1 junk2 junk3 junk4
/home/cis90/simben $ ls junk*
junk1  junk2  junk3  junk4

/home/cis90/simben $ rm junk1
/home/cis90/simben $ ls junk*
junk2  junk3  junk4
```

Create four test files

Remove one of them

Note: the file is removed without warning!

Removing Files

Using the i option to interactively remove multiple files

Remove one or more files interactively:

```
/home/cis90/simben $ rm -i junk*
rm: remove regular empty file `junk2'? y
rm: remove regular empty file `junk3'? n
rm: remove regular empty file `junk4'? n

/home/cis90/simben $ ls junk*
junk3  junk4
```

Remove just the junk2 file

Verify it was removed

Removing Files
The **rmdir** command

Use **rmdir** to remove a directory

```
/home/cis90/simben $ mkdir junkdir1   Make a test directory
/home/cis90/simben $ touch junkdir1/junk6   Put a test file in new directory

/home/cis90/simben $ rmdir junkdir1   Try to remove non-empty directory
rmdir: junkdir1: Directory not empty

/home/cis90/simben $ rm junkdir1/junk6   Remove file in directory
/home/cis90/simben $ rmdir junkdir1   Remove empty directory
/home/cis90/simben $
```

Directories must be empty to be removed by rmdir

Class Exercise

- Change to your home directory

```
cd
```

- Create some test files

```
touch junk1  junk2  junk3  junk4
ls junk*
```

- Remove one

```
rm junk1
ls junk*
```

- Remove the others

```
rm junk[234]
ls junk*
```

linking files

Linking files

The **ln** command

Command syntax:

ln <*existing-name*> <*new-name*>

options: -s

s = symbolic link (like Windows shortcut)

With UNIX there are hard and soft (symbolic) links

Linking files
Hard links

Creating a "hard" link

ln *<existing-name> <new-name>*

```
/home/cis90/simben $ echo "Chocolate Licorice Taffy Jelly Beans" > sweets
/home/cis90/simben $ cat sweets
Chocolate Licorice Taffy Jelly Beans

/home/cis90/simben $ ln sweets dulces
/home/cis90/simben $ ls -il sweets dulces
100176 -rw-rw-r-- 2 simben90 cis90 37 Mar 14 09:29 dulces
100176 -rw-rw-r-- 2 simben90 cis90 37 Mar 14 09:29 sweets
```

Hard link dulces to sweets

same inode

number of hard linked files

*Hard links allows **multiple** filenames for the **same** file. The link count on a long listing tells you how many names the file has.*

Linking files
Hard links

Creating a "hard" link

ln *<existing-name> <new-name>*

```
/home/cis90/simben $ ln sweets candy
/home/cis90/simben $ ls -il sweets dulces candy
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 candy
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 dulces
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 sweets
```

Hard link candy to dulces

same inode

number of hard linked files

```
/home/cis90/simben $ ln sweets bonbons
/home/cis90/simben $ ls -il sweets dulces candy bonbons
100176 -rw-rw-r-- 4 simben90 cis90 37 Mar 14 09:29 bonbons
100176 -rw-rw-r-- 4 simben90 cis90 37 Mar 14 09:29 candy
100176 -rw-rw-r-- 4 simben90 cis90 37 Mar 14 09:29 dulces
100176 -rw-rw-r-- 4 simben90 cis90 37 Mar 14 09:29 sweets
```

Hard link bonbons to sweets

same inode

number of hard linked files

Linking files
Hard links

The . and .. directories are hard links!

```
/home/cis90/simben $ ls -ldi . /home/cis90/simben
98306 drwxr-xr-x 10 simben90 cis90 4096 Mar 14 09:41 .
98306 drwxr-xr-x 10 simben90 cis90 4096 Mar 14 09:41 /home/cis90/simben
```

same inode

number of hard linked files
(includes the . file and .. files in sub-directories)

```
/home/cis90/simben $ ls -ldi .. /home/cis90/
2395394 drwxr-x--- 42 rsimms cis90 4096 Mar  6 08:17 ..
2395394 drwxr-x--- 42 rsimms cis90 4096 Mar  6 08:17 /home/cis90/
```

same inode

number of hard linked files
(includes the . file and .. files in sub-directories)

*Hard links allows **multiple** filenames for the **same** file.*

Note the hidden . and .. files different filenames for the same directories

Linking files
Hard links

Creating a "hard" link

ln *<existing-name> <new-name>*

```
/home/cis90/simben $ rm sweets
/home/cis90/simben $ ls -il sweets dulces candy bonbons
ls: sweets: No such file or directory
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 bonbons
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 candy
100176 -rw-rw-r-- 3 simben90 cis90 37 Mar 14 09:29 dulces
```

same inode

number of hard linked files

Removing one of the hard linked files will not delete any of the other hard links, it will just decrement the number of hard links shown in a long listing

Linking Files
Symbolic "Soft" Links

Creating a "soft" (symbolic) link

ln -s *<existing-name> <new-name>*

The s option for a symbolic link

Creating a symbolic link to the Apache configuration file

```
/home/cis90/simben $ ln -s /etc/httpd/conf/httpd.conf apache

/home/cis90/simben $ ls -li apache /etc/httpd/conf/httpd.conf
 100172 lrwxrwxrwx 1 simben90 cis90    26 Mar 14 09:13 apache -> /etc/httpd/conf/httpd.conf
1280166 -rw-r--r-- 1 root     root  33776 Feb 29 18:45 /etc/httpd/conf/httpd.conf
```

l for symbolic link, - for regular file

Different inodes

Symbolic links are like Windows shortcuts. They are two separate files and it is possible to break the links when the target files get renamed.

Linking Files
Symbolic "Soft" Links

```
/home/cis90/simben $ ls -li apache /etc/httpd/conf/httpd.conf
 100172 lrwxrwxrwx 1 simben90 cis90    26 Mar 14 09:13 apache -> /etc/httpd/conf/httpd.conf
1280166 -rw-r--r-- 1 root     root  33776 Feb 29 18:45 /etc/httpd/conf/httpd.conf


/home/cis90/simben $ head -n 5 apache
#
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.2/> for detailed information.
# In particular, see

/home/cis90/simben $ head -n 5 /etc/httpd/conf/httpd.conf
#
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See <URL:http://httpd.apache.org/docs/2.2/> for detailed information.
# In particular, see
```

From Benji's home directory, he can now refer to the Apache configuration file using either apache or /etc/httpd/conf/httpd.conf

Class Exercise

- Create a file named candy using:

```
> candy
```

- Create a hard link to candy named sweets using:

```
ln candy sweets
```

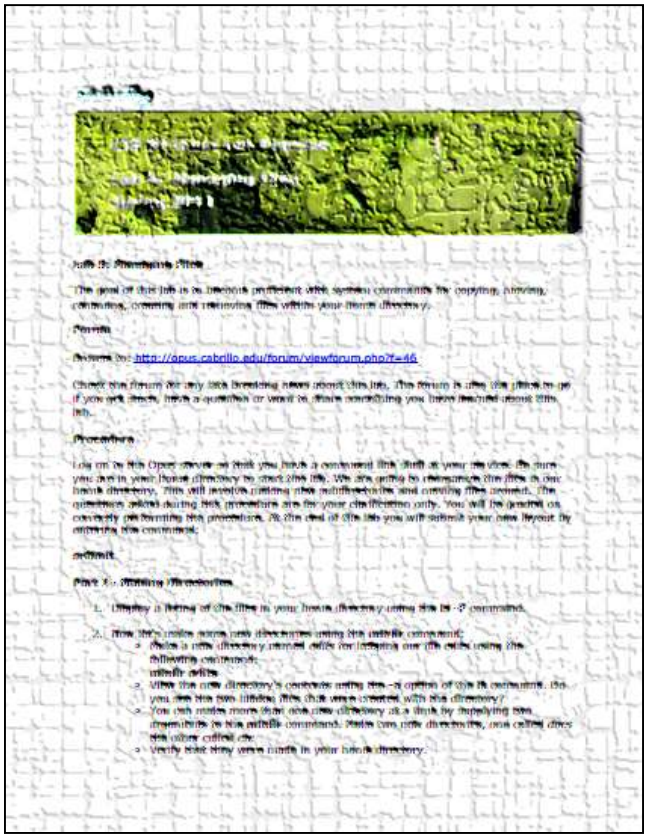
- Create a soft link to candy named dulces using:

```
ln -s candy dulces
```

- List them using:

```
ls -li candy sweets dulces
```


Assignment



In this lab you will reorganize your home directory

Be careful. For this lab, the slower you go the sooner you will be done!


Wrap up

New commands:

cp	copy files
ln	link files
mkdir	make directory
mv	move or rename files
rm	remove files
rmdir	remove directory
touch	make/modify a file
tree	draw file tree branch

Redirection:

>	redirects stdout

Next Class

Assignment: Check Calendar Page on web site to see what is due next week:
http://simms-teach.com/cis90calendar.php

Lab 5 due

Quiz questions for next class:

- What command is used to rename a file?
- If two files are hard linked do they have the same or different inode numbers?
- What option for the rm command provides confirmation when deleting files?

Test 1

Notes to instructor

[ ] Schedule end of practice test on Canvas *[T-30]*

[ ] Shutdown practice test systems

```
cp /etc/nologin.bak /etc/nologin
```
[at job T-31]

```
shutdown -P +10 "Practice test period ending."
```
[at job T-40]

[ ] Schedule real test to display in Canvas *[T-0 till splashdown]*

[ ] Remove password on real test on Canvas *[T-0]*

[ ] Send email on Opus to students

```
/home/rsimms/cis90/test01/q29/mail-q29-T1 2 q
```
[at job T-0]

[ ] Allow logins on real test systems

```
rm /etc/nologin
```
[at job T-0]

[ ] Shutdown real test systems

```
cp /etc/nologin.bak /etc/nologin
```
[at splashdown-1]

```
shutdown -P +10 "Test period ending."
```
[at splashdown-10]


Test 1

Backup

More Examples

Practice Tasks

For use on Opus

Task 1: Create a new directory named *birds* in your home directory. In that new directory create a sub-directory named *Antarctica*. Copy the *penguin* file from the */home/cis90/depot* directory to the new *Antarctica* directory. View the last line of the *penguin* file. Recursively remove the *birds* directory when finished.

```
/home/cis90/simben $ cd
/home/cis90/simben $ mkdir -p birds/Antarctica
/home/cis90/simben $ cp ../depot/penguin birds/Antarctica/
/home/cis90/simben $ tail -n1 birds/Antarctica/penguin
and envy your plumed pride.
/home/cis90/simben $ head -n1 birds/Antarctica/penguin
Magellanic Penguin
/home/cis90/simben $ rm -rf birds/
/home/cis90/simben $
```

Performing Task 1 from the home directory using relative pathnames only.

Practice Tasks

For use on Opus

Task 1: Create a new directory named *birds* in your home directory. In that new directory create a sub-directory named *Antarctica*. Copy the *penguin* file from the */home/cis90/depot* directory to the new *Antarctica* directory. View the last line of the *penguin* file. Recursively remove the *birds* directory when finished.

```
/home/cis90/simben $ cd
/home/cis90/simben $ mkdir birds
/home/cis90/simben $ cd birds
/home/cis90/simben/birds $ mkdir Antarctica
/home/cis90/simben/birds $ cd Antarctica
/home/cis90/simben/birds/Antarctica $ cp /home/cis90/depot/penguin .
/home/cis90/simben/birds/Antarctica $ tail -n1 penguin
and envy your plumed pride.
/home/cis90/simben/birds/Antarctica $ cd
/home/cis90/simben $ rm -rf /home/cis90/simben/birds/
/home/cis90/simben $
```

Performing Task 1 by changing directories and using a mix of relative and absolute pathnames.

Practice Tasks

For use on Opus

Task 1: Create a new directory named *birds* in your home directory. In that new directory create a sub-directory named *Antarctica*. Copy the *penguin* file from the */home/cis90/depot* directory to the new *Antarctica* directory. View the last line of the *penguin* file. Recursively remove the *birds* directory when finished.

```
/home/cis90/depot $ cd /home/cis90/depot/
/home/cis90/depot $ ls penguin
penguin
/home/cis90/depot $ mkdir -p ~/birds/Antarctica
/home/cis90/depot $ cp penguin ~/birds/Antarctica/
/home/cis90/depot $ tail -n1 ~/birds/Antarctica/penguin
and envy your plumed pride.
/home/cis90/depot $ rm -rf ~/birds
/home/cis90/depot $
```

Performing Task 1 from the /home/cis90/depot directory and using the ~ for the home directory.

Practice Tasks

For use on Opus

Task 1: Create a new directory named *birds* in your home directory. In that new directory create a sub-directory named *Antarctica*. Copy the penguin file from the /home/cis90/depot directory to the new *Antarctica* directory. View the last line of the *penguin* file. Recursively remove the *birds* directory when finished.

```
/home/cis90/depot $ cd /home/cis90/depot/
/home/cis90/depot $ ls penguin
penguin
/home/cis90/depot $ mkdir -p ../simben/birds/Antarctica
/home/cis90/depot $ cp penguin ../simben/birds/Antarctica/
/home/cis90/depot $ tail -n1 /home/cis90/simben/birds/Antarctica/penguin
and envy your plumed pride.
/home/cis90/depot $ rm -rf /home/cis90/simben/birds/
/home/cis90/depot $
```

Performing Task 1 from the /home/cis90/depot directory and using relative and absolute pathnames.