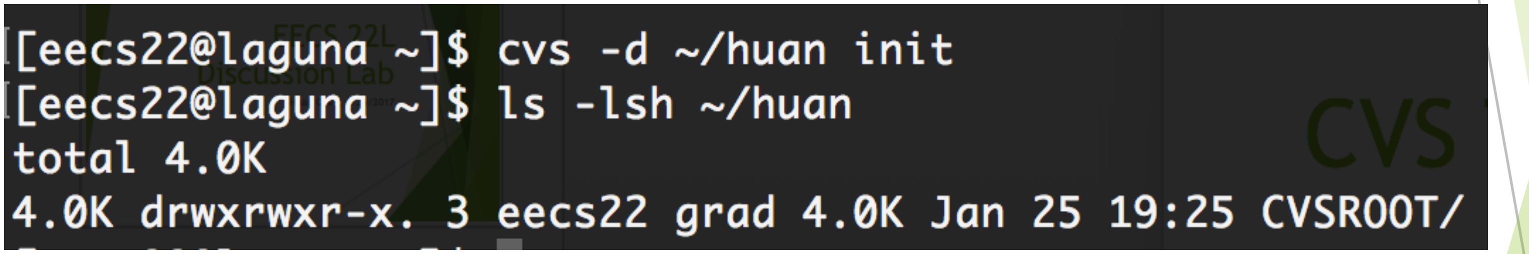
EECS 22L Discussion Lab

Huan Chen, 1/24/2017

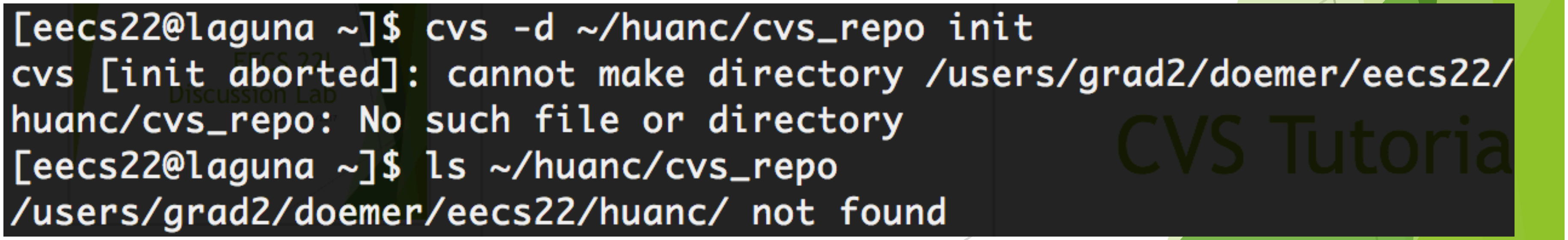
CVS – Create a repository

- Use your team account, create a CVS repository
  - $ cvs –d TARGET_CVS_REPOSITORY_DIR init



```
[eecs22@laguna ~]$ cvs -d ~/huan init
[eecs22@laguna ~]$ ls -lsh ~/huan
total 4.0K
4.0K drwxrwxr-x. 3 eecs22 grad 4.0K Jan 25 19:25 CVSROOT/
```

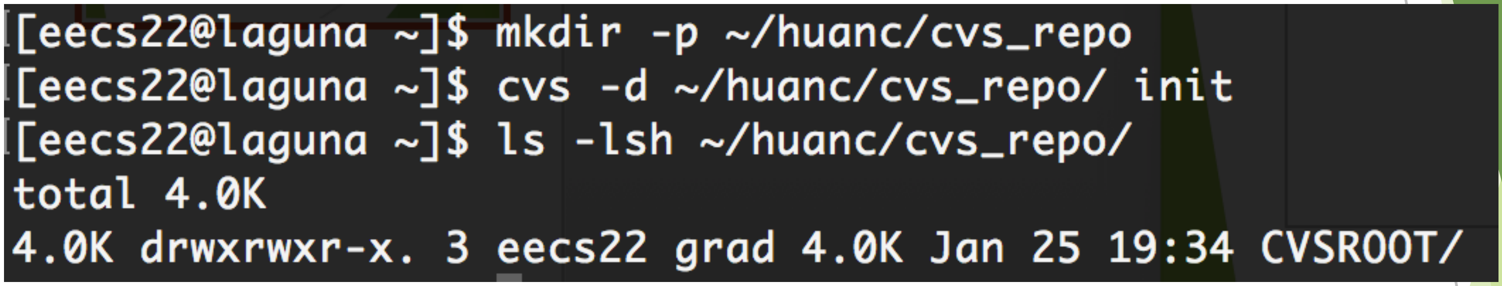
  - if your target directory does not exist, it will be created by cvs, however, if it contains recursive directories, you need to create it first.



```
[eecs22@laguna ~]$ cvs -d ~/huanc/cvs_repo init
cvs [init aborted]: cannot make directory /users/grad2/doemer/eecs22/
huanc/cvs_repo: No such file or directory
[eecs22@laguna ~]$ ls ~/huanc/cvs_repo
/users/grad2/doemer/eecs22/huanc/ not found
```

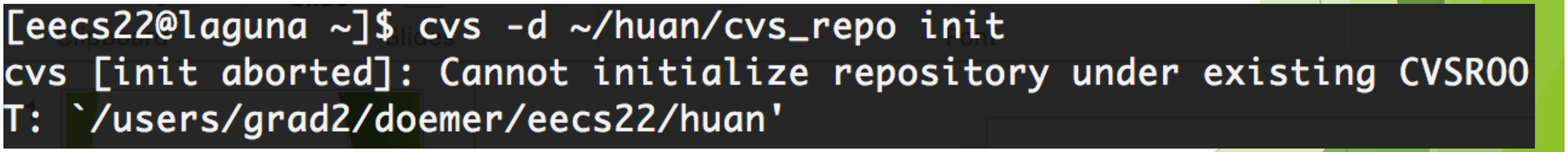
CVS – Create a repository

- Create a directory recursively (to be used as a cvs repo)



```
[eecs22@laguna ~]$ mkdir -p ~/huanc/cvs_repo
[eecs22@laguna ~]$ cvs -d ~/huanc/cvs_repo/ init
[eecs22@laguna ~]$ ls -lsh ~/huanc/cvs_repo/
total 4.0K
4.0K drwxrwxr-x. 3 eecs22 grad 4.0K Jan 25 19:34 CVSROOT/
```

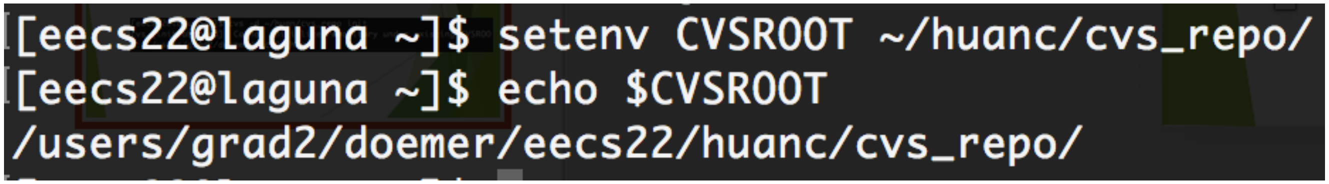
- Cannot create a repo under an existed repo!



```
[eecs22@laguna ~]$ cvs -d ~/huan/cvs_repo init
cvs [init aborted]: Cannot initialize repository under existing CVSROO
T: `/users/grad2/doemer/eecs22/huan'
```

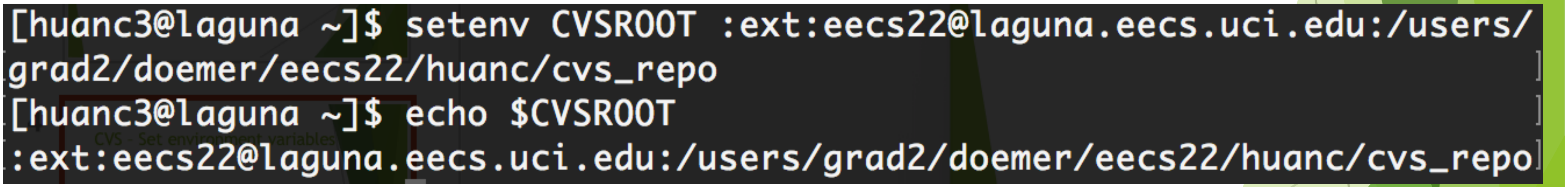
CVS – Set environment variables

- Use your team's account:



```
[eecs22@laguna ~]$ setenv CVSROOT ~/huanc/cvs_repo/
[eecs22@laguna ~]$ echo $CVSROOT
/users/grad2/doemer/eecs22/huanc/cvs_repo/
```

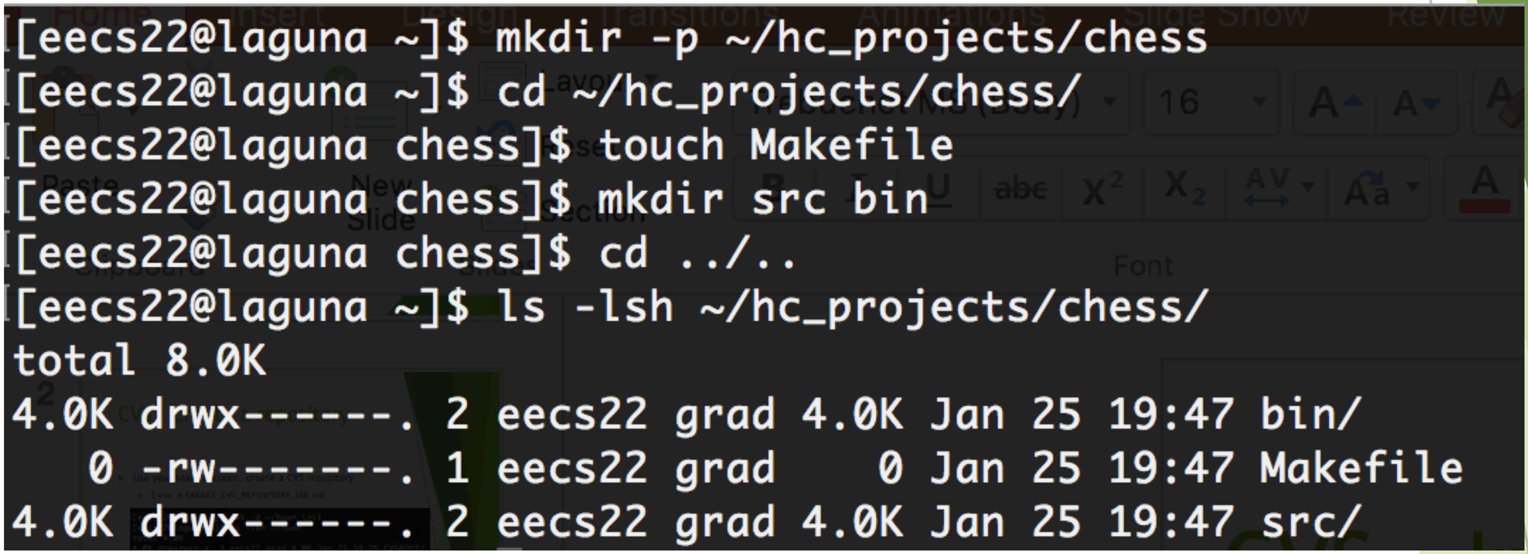
- If you work using your personal account:



```
[huanc3@laguna ~]$ setenv CVSROOT :ext:eecs22@laguna.eecs.uci.edu:/users/
grad2/doemer/eecs22/huanc/cvs_repo
[huanc3@laguna ~]$ echo $CVSROOT
:ext:eecs22@laguna.eecs.uci.edu:/users/grad2/doemer/eecs22/huanc/cvs_repo
```

CVS – Import project to repository

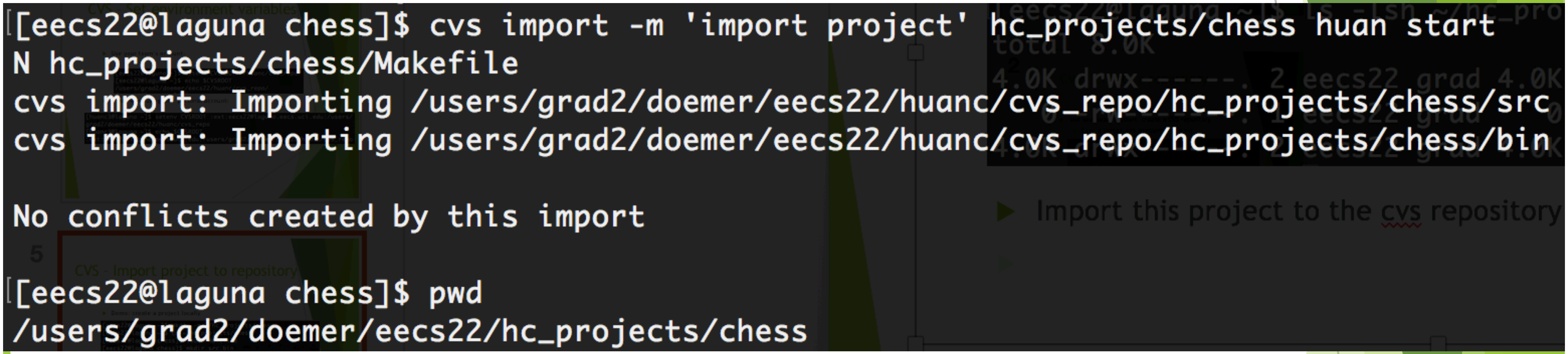
- Demo: create a project locally



```
[eecs22@laguna ~]$ mkdir -p ~/hc_projects/chess
[eecs22@laguna ~]$ cd ~/hc_projects/chess/
[eecs22@laguna chess]$ touch Makefile
[eecs22@laguna chess]$ mkdir src bin
[eecs22@laguna chess]$ cd ../..
[eecs22@laguna ~]$ ls -lsh ~/hc_projects/chess/
total 8.0K
4.0K drwx------. 2 eecs22 grad 4.0K Jan 25 19:47 bin/
   0 -rw-------. 1 eecs22 grad    0 Jan 25 19:47 Makefile
4.0K drwx------. 2 eecs22 grad 4.0K Jan 25 19:47 src/
```

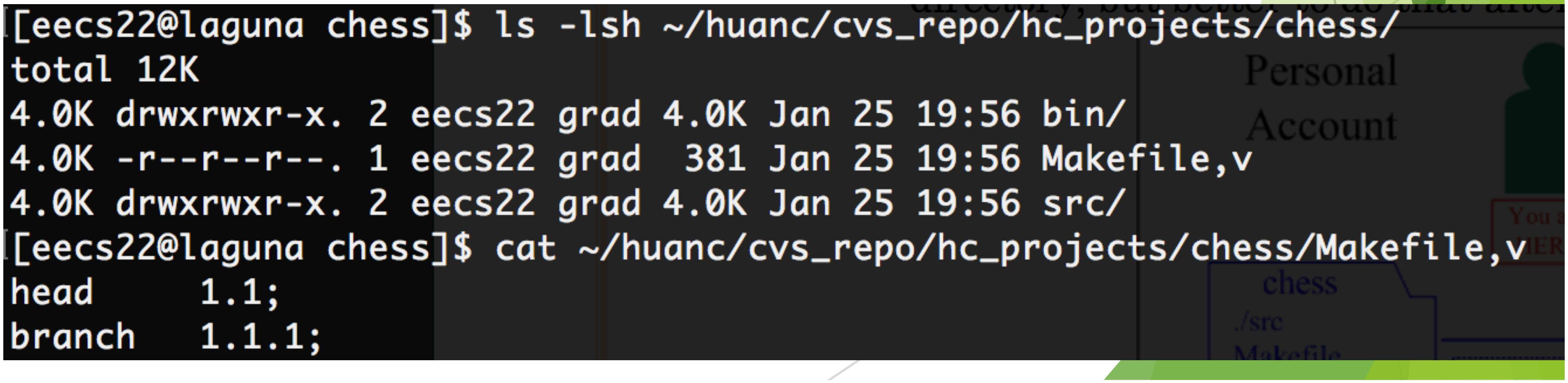
CVS – Import project to repository

- Demo: Import this project to the cvs repository
  - $ cvs import –m ‘YOUR_MSG’ YOUR_PROJECT_NAME VENDOR VERSION



```
[[eecs22@laguna chess]$ cvs import -m 'import project' hc_projects/chess huan start
N hc_projects/chess/Makefile
cvs import: Importing /users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src
cvs import: Importing /users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/bin

No conflicts created by this import

[[eecs22@laguna chess]$ pwd
/users/grad2/doemer/eecs22/hc_projects/chess
```

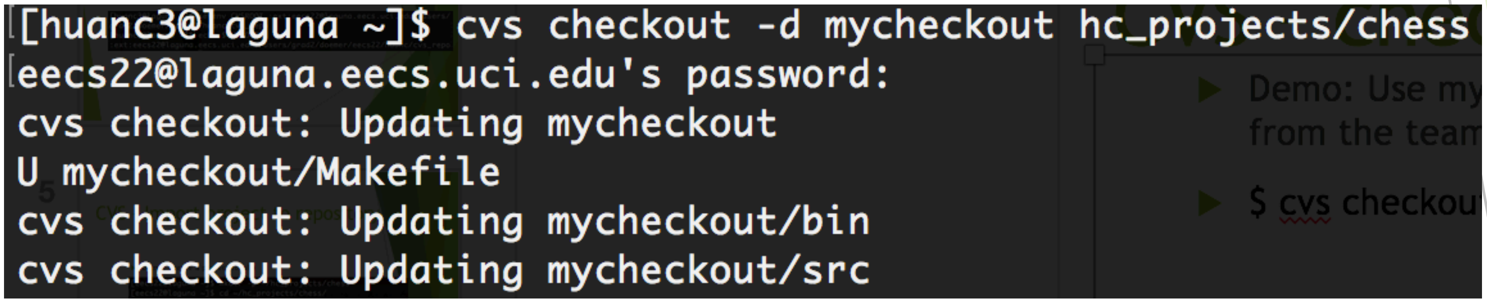
- Verify the project is imported successfully (not real Makefile!)



```
[[eecs22@laguna chess]$ ls -lsh ~/huanc/cvs_repo/hc_projects/chess/
total 12K
4.0K drwxrwxr-x. 2 eecs22 grad 4.0K Jan 25 19:56 bin/
4.0K -r--r--r--. 1 eecs22 grad  381 Jan 25 19:56 Makefile,v
4.0K drwxrwxr-x. 2 eecs22 grad 4.0K Jan 25 19:56 src/
[[eecs22@laguna chess]$ cat ~/huanc/cvs_repo/hc_projects/chess/Makefile,v
head    1.1;
branch  1.1.1;
```

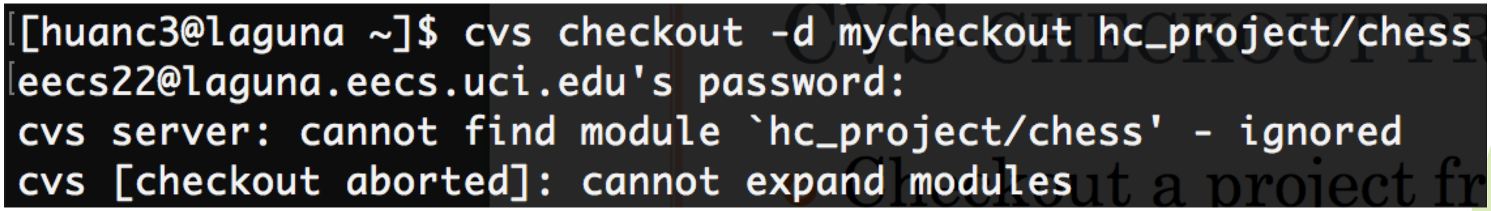
CVS – checkout project from repo

- Demo: Use my personal account huanc3, checkout the chess project (hc_projects/chess) from the team account eecs22
- $ cvs checkout -d YOUR_PATH PROJECT_NAME



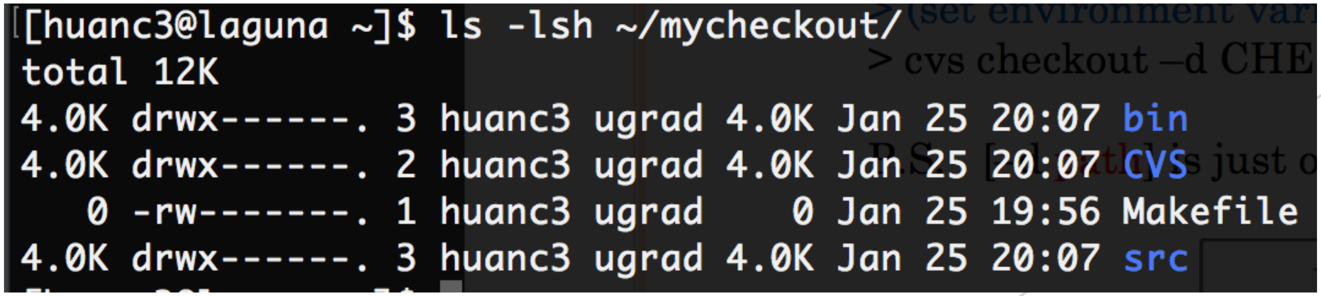
```
[huanc3@laguna ~]$ cvs checkout -d mycheckout hc_projects/chess
eecs22@laguna.eecs.uci.edu's password:
cvs checkout: Updating mycheckout
U mycheckout/Makefile
cvs checkout: Updating mycheckout/bin
cvs checkout: Updating mycheckout/src
```

- Incorrect project name causes failure (hc_project/chess)



```
[huanc3@laguna ~]$ cvs checkout -d mycheckout hc_project/chess
eecs22@laguna.eecs.uci.edu's password:
cvs server: cannot find module `hc_project/chess' - ignored
cvs [checkout aborted]: cannot expand modules
```

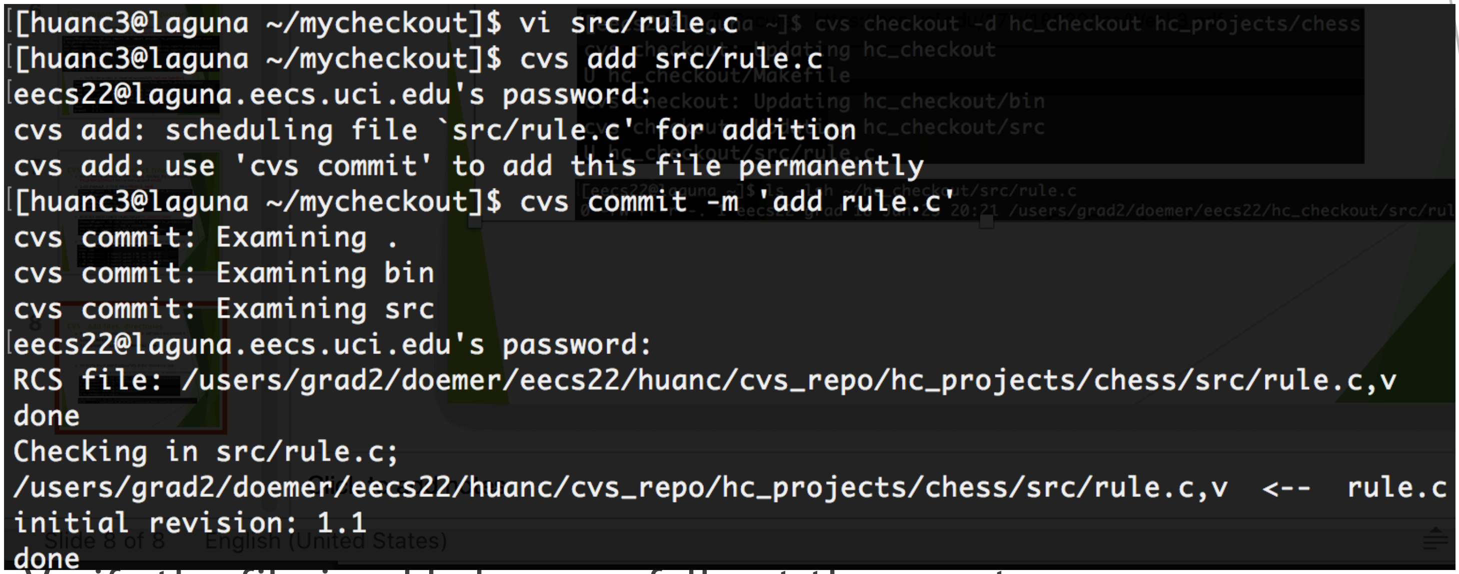
- Verify the project is copied successfully from the remote cvs repo



```
[huanc3@laguna ~]$ ls -lsh ~/mycheckout/
total 12K
4.0K drwx------. 3 huanc3 ugrad 4.0K Jan 25 20:07 bin
4.0K drwx------. 2 huanc3 ugrad 4.0K Jan 25 20:07 CVS
   0 -rw-------. 1 huanc3 ugrad    0 Jan 25 19:56 Makefile
4.0K drwx------. 3 huanc3 ugrad 4.0K Jan 25 20:07 src
```

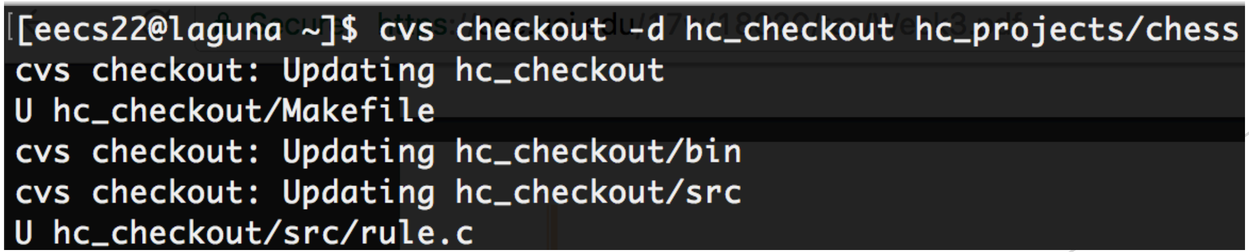
CVS – Add files (commit required)

- Demo: Use my personal account huanc3, add rule.c and commit it
- $ cvs add PATH/FILE_NAME
- $ cvs commit –m “add xx files” PATH/FILE_NAME

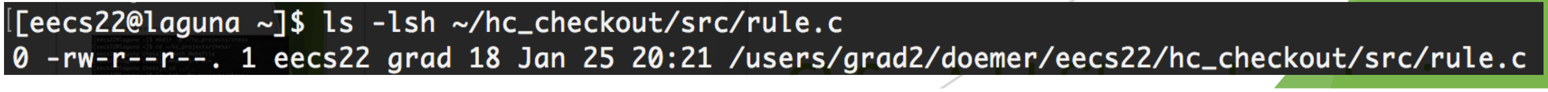


```
[huanc3@laguna ~/mycheckout]$ vi src/rule.c
[huanc3@laguna ~/mycheckout]$ cvs add src/rule.c
eecs22@laguna.eecs.uci.edu's password:
cvs add: scheduling file `src/rule.c' for addition
cvs add: use 'cvs commit' to add this file permanently
[huanc3@laguna ~/mycheckout]$ cvs commit -m 'add rule.c'
cvs commit: Examining .
cvs commit: Examining bin
cvs commit: Examining src
eecs22@laguna.eecs.uci.edu's password:
RCS file: /users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src/rule.c,v
done
Checking in src/rule.c;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src/rule.c,v  <--  rule.c
initial revision: 1.1
done
```

- Verify the file is added successfully at the remote cvs repo
  - by checking out another copy at the team’s account (no password needed)



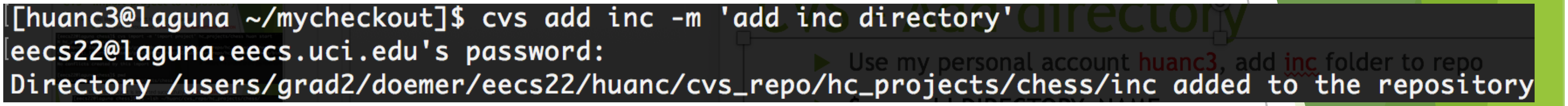
```
[eecs22@laguna ~]$ cvs checkout -d hc_checkout hc_projects/chess
cvs checkout: Updating hc_checkout
U hc_checkout/Makefile
cvs checkout: Updating hc_checkout/bin
cvs checkout: Updating hc_checkout/src
U hc_checkout/src/rule.c
```



```
[eecs22@laguna ~]$ ls -lsh ~/hc_checkout/src/rule.c
0 -rw-r--r--. 1 eecs22 grad 18 Jan 25 20:21 /users/grad2/doemer/eecs22/hc_checkout/src/rule.c
```

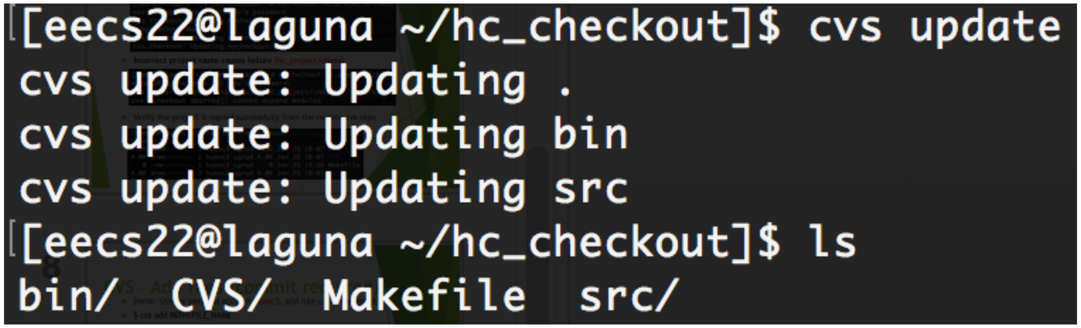
CVS – Add directory

- Use my personal account huanc3, add inc folder to repo
- $ cvs add DIRECTORY_NAME (no need to do cvs commit)



```
[[huanc3@laguna ~/mycheckout]$ cvs add inc -m 'add inc directory'
[eecs22@laguna.eecs.uci.edu's password:
Directory /users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/inc added to the repository
```

- Verify the file is added successfully at the remote cvs repo
- $ cvs update (update project)

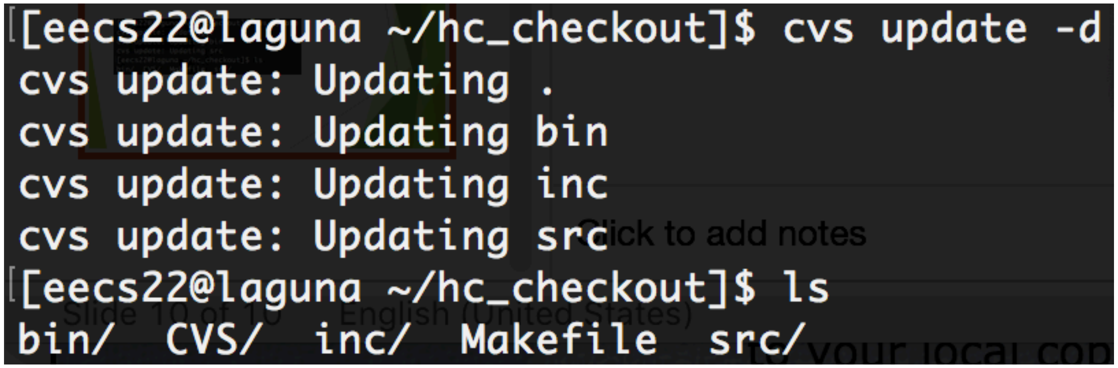


```
[[eecs22@laguna ~/hc_checkout]$ cvs update
cvs update: Updating .
cvs update: Updating bin
cvs update: Updating src
[[eecs22@laguna ~/hc_checkout]$ ls
bin/  CVS/  Makefile  src/
```

- Where is our newly added “inc” folder?

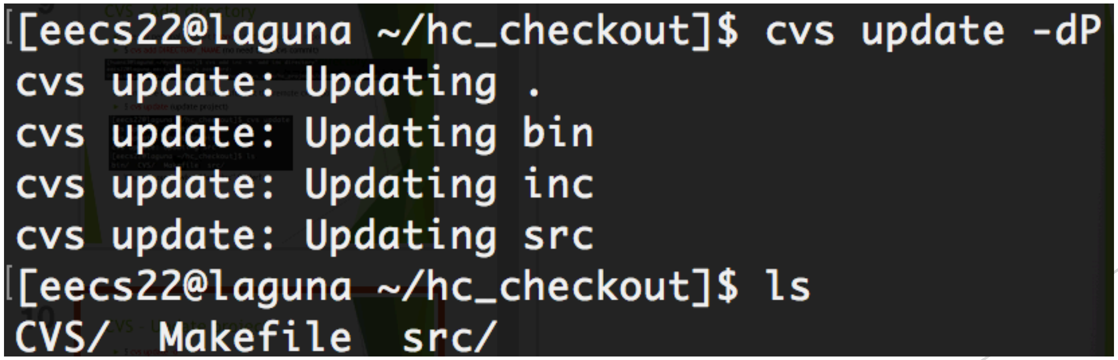
CVS – Update project

- $ cvs update -d
- "-d": create new directories that may have been added to repo



```
[eecs22@laguna ~/hc_checkout]$ cvs update -d
cvs update: Updating .
cvs update: Updating bin
cvs update: Updating inc
cvs update: Updating src
[eecs22@laguna ~/hc_checkout]$ ls
bin/  CVS/  inc/  Makefile  src/
```

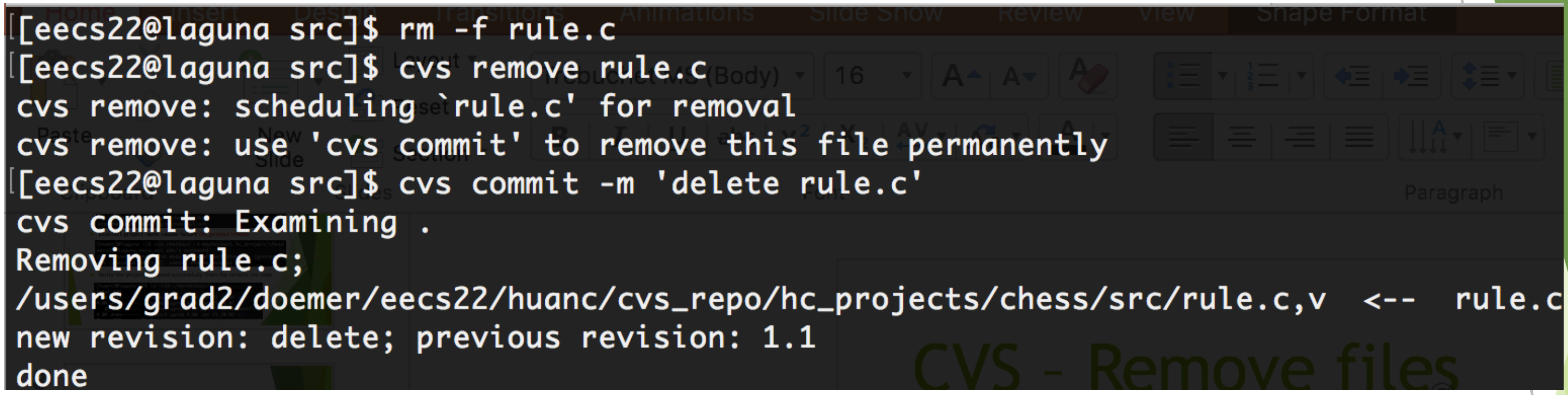
- What if I only want to checkout/update non-empty folders? ("bin", "inc" are empty)
- $ cvs update –dP (cvs checkout –P)
- "-P": prune empty directories



```
[eecs22@laguna ~/hc_checkout]$ cvs update -dP
cvs update: Updating .
cvs update: Updating bin
cvs update: Updating inc
cvs update: Updating src
[eecs22@laguna ~/hc_checkout]$ ls
CVS/  Makefile  src/
```

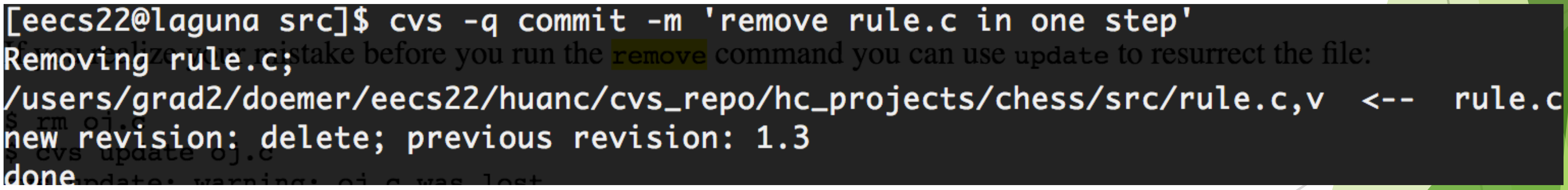
CVS – Remove files

- $ rm –f FILE_NAME
- $ cvs remove FILE_NAME



```
[eecs22@laguna src]$ rm -f rule.c
[eecs22@laguna src]$ cvs remove rule.c
cvs remove: scheduling `rule.c' for removal
cvs remove: use 'cvs commit' to remove this file permanently
[eecs22@laguna src]$ cvs commit -m 'delete rule.c'
cvs commit: Examining .
Removing rule.c;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src/rule.c,v  <--  rule.c
new revision: delete; previous revision: 1.1
done
```

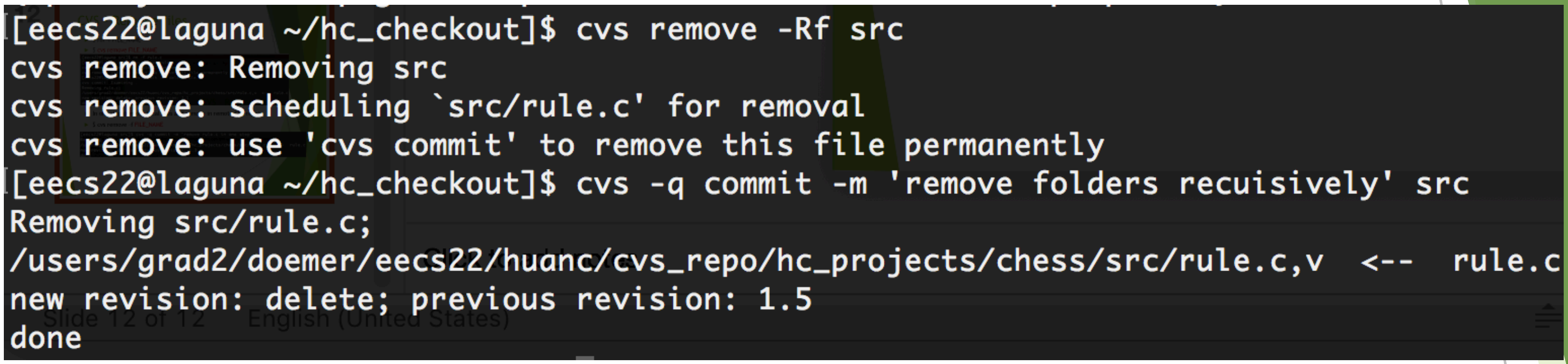
- In one step (remove local files and remove in remote CVS repo)
- $ cvs remove –f FILE_NAME



```
[eecs22@laguna src]$ cvs -q commit -m 'remove rule.c in one step'
Removing rule.c;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src/rule.c,v  <--  rule.c
new revision: delete; previous revision: 1.3
done
```

CVS – Remove folders recursively

- $ cvs remove –Rf FOLDER



```
[eecs22@laguna ~/hc_checkout]$ cvs remove -Rf src
cvs remove: Removing src
cvs remove: scheduling `src/rule.c' for removal
cvs remove: use 'cvs commit' to remove this file permanently
[eecs22@laguna ~/hc_checkout]$ cvs -q commit -m 'remove folders recuisively' src
Removing src/rule.c;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/src/rule.c,v  <--  rule.c
new revision: delete; previous revision: 1.5
done
```

- But folder “src” is not removed in CVS repo…

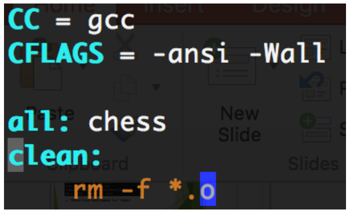
CVS – Configuration file: .csvrc

- What if I am tired of typing the following cvs commands and
  - Tell cvs the default command line options by editing ~/.cvsrc file
  - Recommended default configurations (http://www.linuxhowtos.org/System/cvs_tutorial.htm)

    -
      ```
      cvs -q
      diff -u -b -B
      checkout -P
      update -d -P
      ```
  - $ cvs –q (cause the command to be somewhat quiet, for example, makes the long output of “cvs update” easier to read)

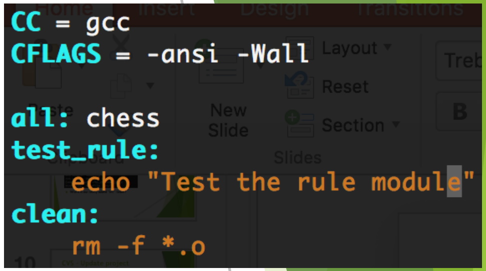
CVS – Generate & resolve conflict

- This morning, two developers checked out the same project from repo (or updated it)
- Original “Makefile”



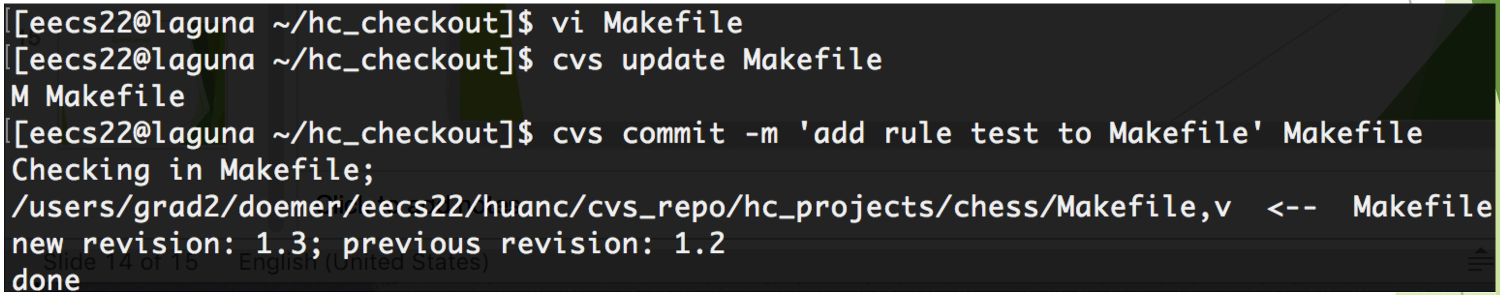
```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
clean:
	rm -f *.o
```

- Developer 1 modifies a file “Makefile”



```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
test_rule:
	echo "Test the rule module"
clean:
	rm -f *.o
```

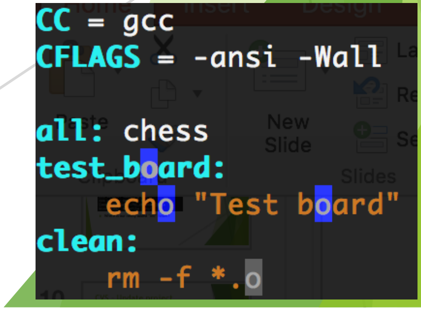
  - $ cvs update Makefile; cvs commit –m ‘add rule test’



```
[eecs22@laguna ~/hc_checkout]$ vi Makefile
[eecs22@laguna ~/hc_checkout]$ cvs update Makefile
M Makefile
[eecs22@laguna ~/hc_checkout]$ cvs commit -m 'add rule test to Makefile' Makefile
Checking in Makefile;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/Makefile,v  <--  Makefile
new revision: 1.3; previous revision: 1.2
done
```

  - Feeling:

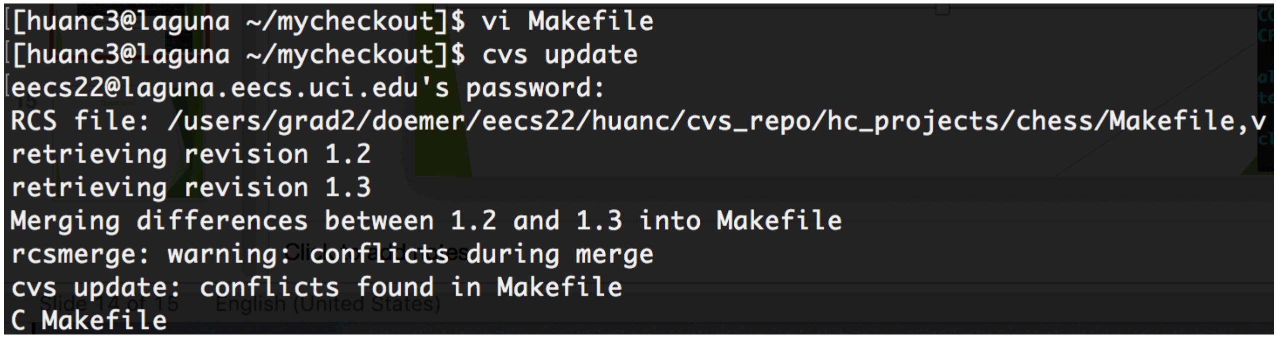
- Developer 2 modifies the same file “Makefile”
  - $ cvs update Makefile; cvs commit –m “add board test”



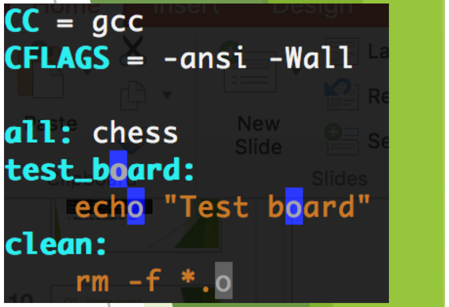
```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
test_board:
	echo "Test board"
clean:
	rm -f *.o
```

CVS – Generate & resolve conflict

- Developer 2 modifies the same file “Makefile”
  - $ cvs update Makefile; cvs commit –m “add board test”



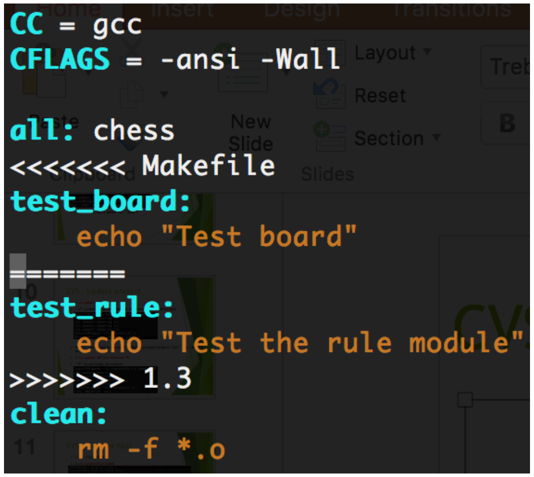
```
[huanc3@laguna ~/mycheckout]$ vi Makefile
[huanc3@laguna ~/mycheckout]$ cvs update
[eecs22@laguna.eecs.uci.edu's password:
RCS file: /users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/Makefile,v
retrieving revision 1.2
retrieving revision 1.3
Merging differences between 1.2 and 1.3 into Makefile
rcsmerge: warning: conflicts during merge
cvs update: conflicts found in Makefile
C Makefile
```



```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
test_board:
        echo "Test board"
clean:
        rm -f *.o
```

  - Feeling:

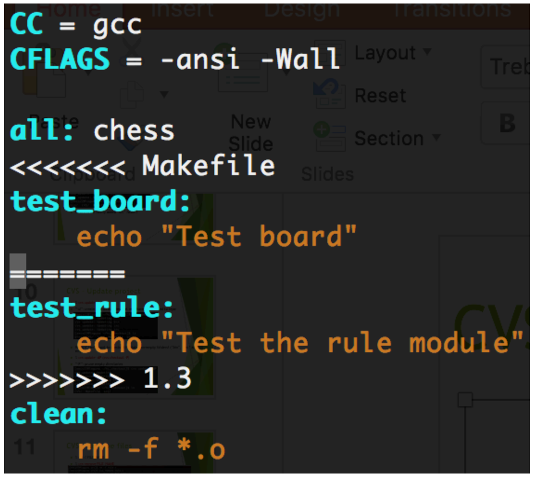
  - $ vi Makefile



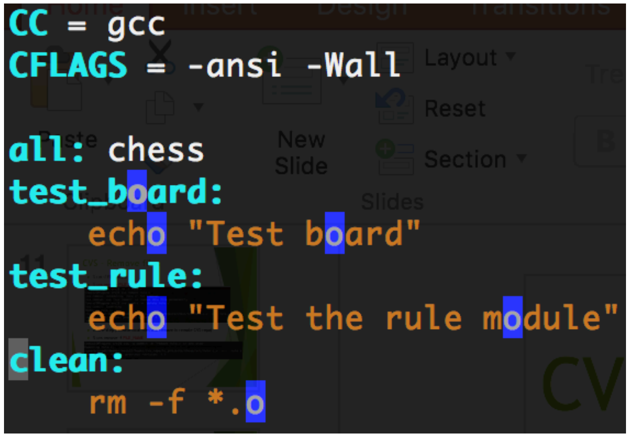
```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
<<<<<<< Makefile
test_board:
        echo "Test board"
=======
test_rule:
        echo "Test the rule module"
>>>>>>> 1.3
clean:
        rm -f *.o
```

CVS – Generate & resolve conflict

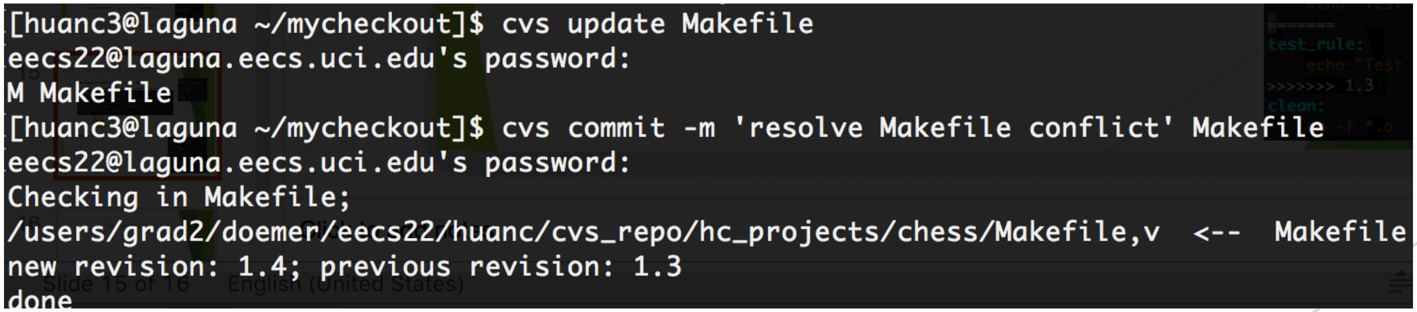
- How to fix it?
- Developer 2 has to fix the conflicting lines listed between <<<< and >>>> markers manually, then update and commit it



```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
<<<<<<< Makefile
test_board:
	echo "Test board"
=======
test_rule:
	echo "Test the rule module"
>>>>>>> 1.3
clean:
	rm -f *.o
```



```
CC = gcc
CFLAGS = -ansi -Wall

all: chess
test_board:
	echo "Test board"
test_rule:
	echo "Test the rule module"
clean:
	rm -f *.o
```



```
[huanc3@laguna ~/mycheckout]$ cvs update Makefile
eecs22@laguna.eecs.uci.edu's password:
M Makefile
[huanc3@laguna ~/mycheckout]$ cvs commit -m 'resolve Makefile conflict' Makefile
eecs22@laguna.eecs.uci.edu's password:
Checking in Makefile;
/users/grad2/doemer/eecs22/huanc/cvs_repo/hc_projects/chess/Makefile,v  <--  Makefile
new revision: 1.4; previous revision: 1.3
done
```

- When developer 1 does "cvs update Makefile", your change will be synchronized too, no need to fix the conflict endlessly

Questions?