

Laboratory 4: Getting Started with CVS

Name : _____

Student Number : _____

Laboratory Time : _____

Objectives

- Set up a CVS repository on UNIX
- Use Eclipse as a CVS client

Preamble

CVS is an acronym for Concurrent Versions System. It is a version control system and is used to record the history of your source files. Bugs can creep in when software is modified, and may not be detected until a long time after the modification is made. With CVS, you can retrieve old versions to find which change caused the bug.

CVS can also help when a project is being worked on by multiple people, where overwriting each others' changes is easy to do. CVS solves this problem by having each developer work in his/her own directory and then instructing CVS to merge the work when each developer is done.

In this laboratory, you will create a CVS repository on UNIX. You will then access this repository from the Eclipse workbench. You will then have a "sandbox" or working copy of the files.

Grading Checklist

- Configure UNIX Shell
- Connect to CVS Repository from Eclipse
- Check in a Project
- Check out a Project

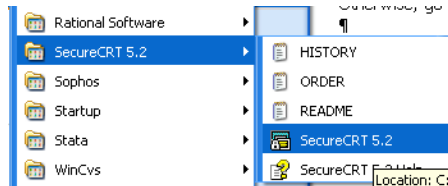
TA Initials: _____

Instructions for the Laboratory

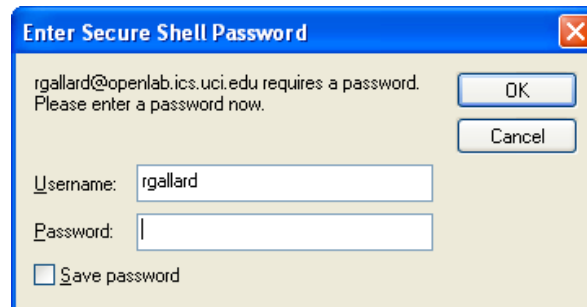
Task 1: Configure Your UNIX Shell

For this task, you will verify that CVS is available in your UNIX environment. If CVS is not available, you will add it. CVS needs to be available so that you can create a CVS repository in your home directory and connect to it using Eclipse in later tasks.

- a) Log in to your UNIX account
 1. Open SecureCRT 5.2

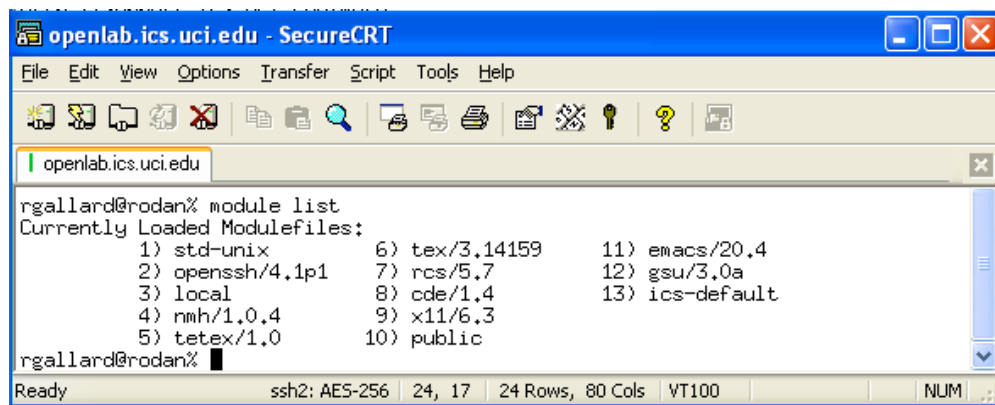


2. Select openlab.ics.uci.edu and click Connect. If a "New Host Key" window appears, click the "Accept and Save" button. Then, it will show a "SecureCRT Window" indicating that the "Access is denied". Click "OK".
3. Enter your username and click "OK". Then, enter your password and click "OK". You will see the Unix prompt.



- b) Type in the command "module list"

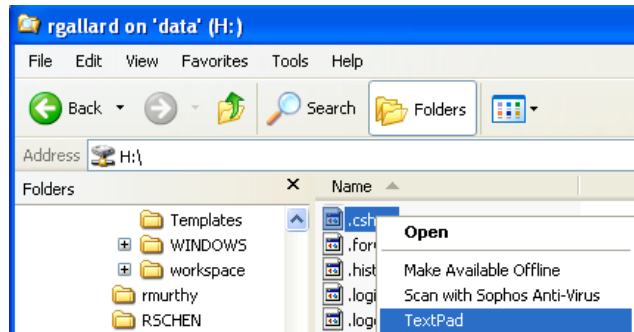
This command will produce a list of modules, or packages of programs, that are available to you in your UNIX environment. The output should look something like the following.



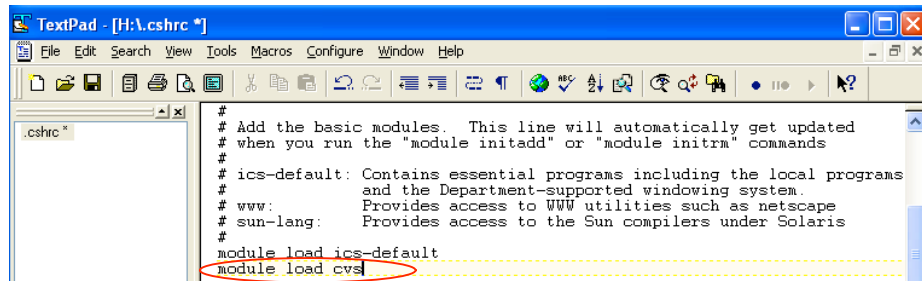
If the output includes CVS, as in item similar to "cvs/1.11.6", then you are done with this task. Otherwise, go to step c).

- c) Add the line "module load cvs" to your .cshrc file (notice that there is a period at the beginning of that file name). The following steps will show you how to do this change using Text Pad. If you want to use a UNIX editor such as vi, you can use it.

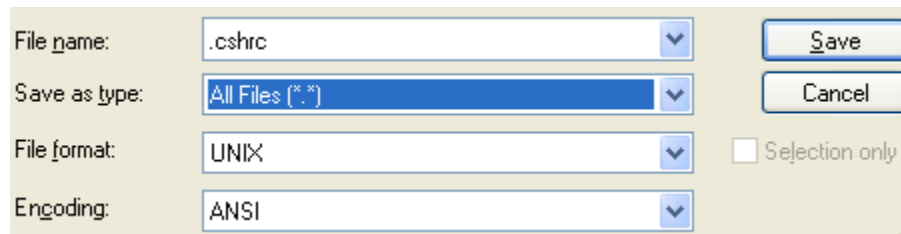
1. Go to your H: drive using Windows Explorer and find the file .cshrc. Right click on it and open it using Text Pad.



2. You should add your line "module load cvs" close to the other module load commands in the .cshrc file.



3. Make sure you save the file as a UNIX FILE FORMAT. Go to the Save As.. option and indicate: Save as Type: All Files (*.*) and File format: UNIX. If you are asked if you want to replace the file, click "Yes."



- d) Log out from SecureCRT using the "exit" command and repeat steps (a) and (b) to verify that the edit was completed correctly. You should see the cvs/1.11.6 module listed.

```

openlab.ics.uci.edu
rgallard@mothra% module list
Currently Loaded Modulefiles:
  1) std-unix          6) tex/3.14159      11) emacs/20.4
  2) openssh/4.1p1   7) rcs/5.7         12) gsu/3.0a
  3) local            8) cde/1.4         13) ics-default
  4) nmh/1.0.4       9) x11/6.3        14) cvs/1.11.6
  5) tetex/1.0      10) public
rgallard@mothra%
Ready      ssh2: AES-256  24, 18  24 Rows, 80 Cols  VT100  NUM

```

If you want to know more about how modules work, visit <http://www.ics.uci.edu/computing/unix/modules/>

Task 2: Add a Directory for Your CVS Repository

This task involves creating a directory for your CVS repository in your UNIX Account.

- Log in to your UNIX account.
- Create a directory called "111-repository" (or 121-repository, if you prefer). The command to do this is "mkdir 111-repository". **Do not place the directory in a subdirectory within your account.**
- Change to this directory. To do this, use the command "cd 111-repository"
- Take note of the full path to this directory. Type the command "pwd" and write down the result.
- Initialize the CVS repository: Type in the command "cvs -d <path> init". Be sure to substitute the string that you wrote down in step (d) instead of <path>.

```

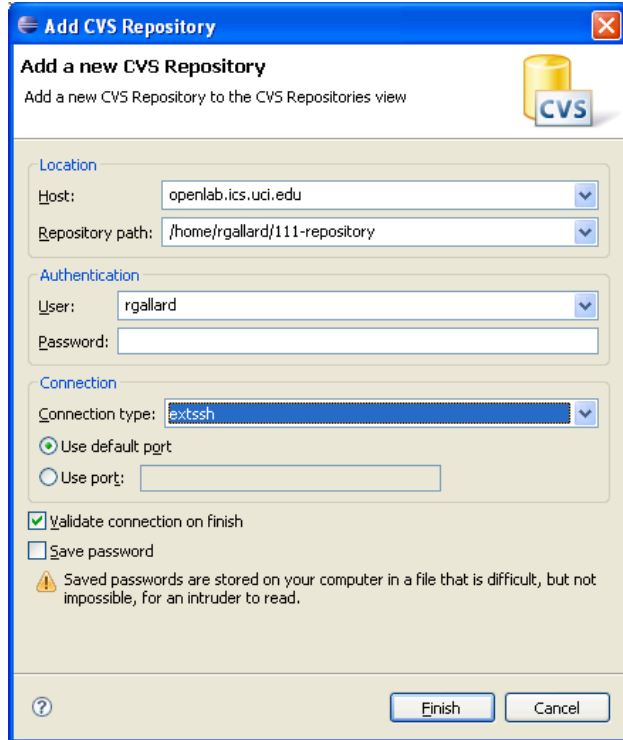
openlab.ics.uci.edu
rgallard@mothra% mkdir 111-repository
rgallard@mothra% cd 111-repository
rgallard@mothra% pwd
/home/rgallard/111-repository
rgallard@mothra% cvs -d /home/rgallard/111-repository init
rgallard@mothra%
Ready      ssh2: AES-256  24, 18  24 Rows, 80 Cols  VT100  NUM

```

Task 3: Connect to Your CVS Repository from Eclipse

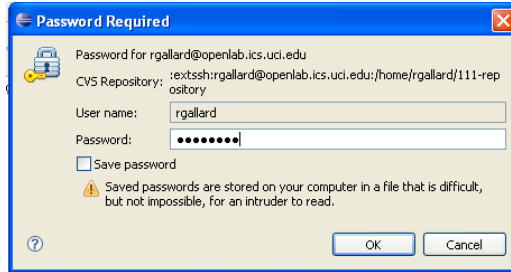
- In Eclipse, go to the CVS Perspective: Select Window -> Open Perspective -> Other -> CVS Repository Exploring.

- b) Create a new Repository Location: In the CVS Repositories tab, right click and select New -> Repository Location.
- c) Fill in the dialog box as indicated in the screen shot below. Then, click Finish.

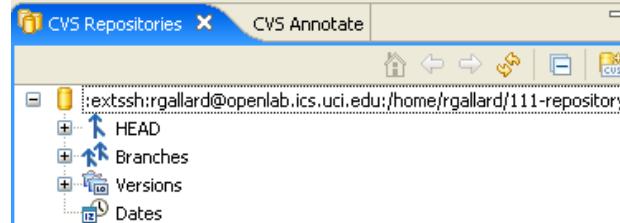


Host: openlab.ics.uci.edu
Repository path: The result from Task 2. Step d)
User: Your UNIX id
Password: **Leave this blank**
Connection type: extssh

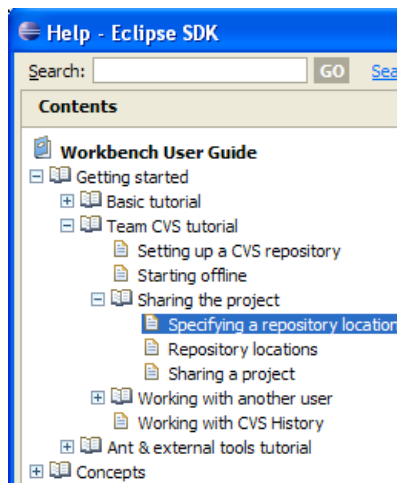
- d) Connect using this new Repository Location
 1. If you have already connected securely to this host before, click on "Finish" and go directly to step (4). Otherwise, just follow the steps (if you do not remember, just check the steps and skip accordingly).
 2. If this is your first time connecting securely to this host, once you click on "Finish" a question box will appear asking if you accept the RSA fingerprint. Click "Yes".
 3. Next you may be asked if you wish to create the known_hosts file to save the RSA fingerprint into. Click "Yes".
 4. Next you may be asked if you want to create the parent directory for ssh. Click "Yes".
 5. You will see a message indicating that the ssh directory has been successfully created. Click "OK".
 6. When using Eclipse it will try to Authenticate using publicKey, password, and keyboard-interactive, in that order. PublicKey Authentication fails silently in the background, and then we are presented with the Password Authentication dialogue box. Type your Unix account password here and click "Ok".



7. You will see the repository you just created:

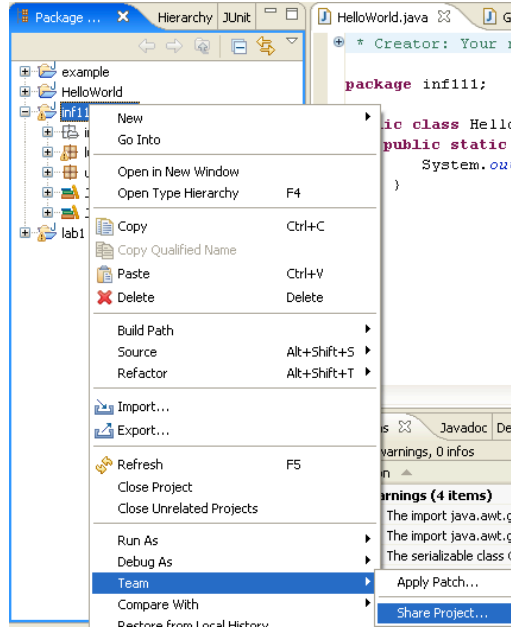


For additional help on setting up a CVS repository, see the online help in Eclipse.

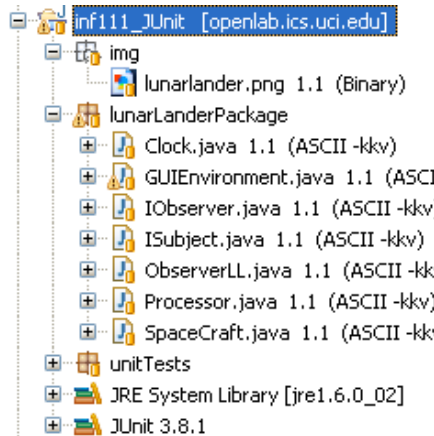


Task 4: Check In a Project

- a) Go to the Java Perspective: select Window -> Open Perspective -> Java. Go to the Package Explorer.
- b) Look for the project inf111_JUnit that we created in Laboratory 2. If the project from Laboratory 2 is not listed in the Package Explorer, open the project from Laboratory 2. If you do not have it, you can create a new project and a new class in this project, but please let the TA know about this.
- c) Check in your project:
 1. Right-click on the project (inf111_JUnit) and select the Team -> Share Project... menu item.



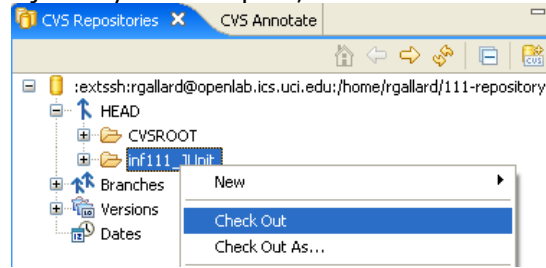
2. Select "use existing repository location" and click "Next".
3. Select "use project name as module name" and click "Next".
4. Click "Finish" to commit the files to the repository.
5. You will see a "Share Project Resources" window. Click "Finish". If you are asked to enter a comment for the commit, enter "First Commit" and click "Finish".
6. You will see your project in the Package Explorer similar to the following screenshot:



Task 5: Check Out a Project

- a) Go to the Java Perspective: Select Window -> Open Perspective -> Package Explorer.
- b) Close the project you just committed (inf111_JUnit): Click on the project name and then select Project -> Close Project.
- c) Check out the project you just committed (inf111_JUnit) from your CVS repository
 1. Go to the CVS Repositories perspective
 2. Select the "Head" and expand it

3. Select the project name (inf111_JUnit)
4. Right-click the project and select the "Check out" menu item (note: If you do not want to overwrite your project in your workspace, select "Check out As.." menu item instead)



5. A "Confirm Overwrite" dialog box will appear. Click "OK"
6. Go to the Java Perspective (Window -> Open Perspective -> Java). Make a minor change to the GUIEnvironment.java file: Change line 23 and 24: (If you are not using inf111_JUnit you can make a minor change in the class you just created, such as adding a comment).

Before changing:

```
int wRec=70;
```

```
int hRec=70;
```

After changing:

```
int wRec=80;
```

```
int hRec=80;
```

7. Save the file you changed (GUIEnvironment.java).

- d) Check In the modified project and verify your changes
 1. Select the project you are working on (inf111_JUnit).
 2. Right-click on the project and select the Team ->Commit... menu item
 3. A "Commit Files" dialog box will appear. Write a comment "I changed the value of wRec and hRec properties" and click "Finish" (note that only the modified file is going to be uploaded to the repository).
 4. Go to the CVS Repositories perspective and select the project you are working on (inf111_JUnit).
 5. Right-click on the project and select "Refresh View". Find the file you modified (GUIEnvironment.java) and you will see the changes you made in this file. You will also see that this file has the version 1.2 now.

