

# CMSC131

Eclipse, CVS, Submit Server

# Some Logistics

- Projects will be done on an individual basis on “your own time” on your own machine or in a campus open lab.
- Some lab exercises may be “paired” (though you won’t have to work in a pair if you don’t want to).
- My office (3239) and the TA office (1112) are both in the A.V.Williams building.

# Computer Science -vs- Computer Programming

What do you think “computer science” contains that “computer programming” does not?

I expect to earn the following grade:

1. A
2. B
3. C
4. D
5. F
6. W

# Getting Eclipse

- Information about the version of Eclipse we use along with links to download it and the course management plug-ins can be found at <http://www.cs.umd.edu/eclipse/>
- You can even “install” and run it right off a USB keychain!

# Starting Eclipse

- Eclipse is an example of what is known as an Integrated Development Environment (IDE).
- When you run Eclipse the first time and select the directory for your workspace, write this down somewhere so that you know where it is on your hard drive (for cases like if you ever want to make a personal backup copy).
- From the “Welcome to Eclipse” page you’ll probably just want to go the “Workbench” by clicking on the curved arrow.

# Some Eclipse Terminology

- Project
  - Essentially a collection of all of the source and data files associated with what we would call a project.
- Perspectives
  - Different “views” into aspects of Eclipse.
  - Examples we use are: Java, CVS, Debug
- Workspace
  - Essentially this is your local storage space.

# Perspectives

- We will make use of the **Java**, **CVS**, and **Debug** perspectives.
- You can start to add a perspective by going to **Window** on the menu bar, going under **Open Perspective**, and selecting **Other...**
- To add the CVS perspective we will use, select **CVS Repository Exploring** from the list. It will then appear as one of the options on the perspectives tab list.

# What is the CVS repository?

First, CVS stands for

**C**oncurrent

**V**ersioning

**S**ystem

The short answer is that for this class it will allow you to keep a copy of your project on a remote server and work on it from different locations with more ease.

I will post directions over the weekend telling you how to get your CVS account information.

# Connecting to your CVS Repository

- Select the **CVS Repository Exploring** option on the perspectives tab. There should now be a large pane on the left side of the screen with the tab title “CVS Repositories”.
- Right-click in the large white area of this to bring up the context menu. Select **New** and then **Repository Location...** to be able to set up your connection.

Host: **linuxlab.csic.umd.edu**

Path: (you will replace **cs131XXX** with your class account)  
/afs/csic.umd.edu/users/tomr55/cvs131Spring13/cs131XXX

Connection type: **extssh**

# Checking out a project

- Under the class repository entry in the tree (very long name like `extssh:cs131400@linuxlab.csic.umd.edu:/afs/csic.umd.edu/users/tomr55/cvs131Spring13/cs131400`) there will be an entry **HEAD** under which there will be a list of items that can be checked out.
- One of these (next Wednesday) will be called **Spring2013-Lab01** – to check this out, you'll just right-click on that item and select **Check Out**.
- If you now go to the Java perspective, you will see it appear as an item there. This contains your local copy of the files.

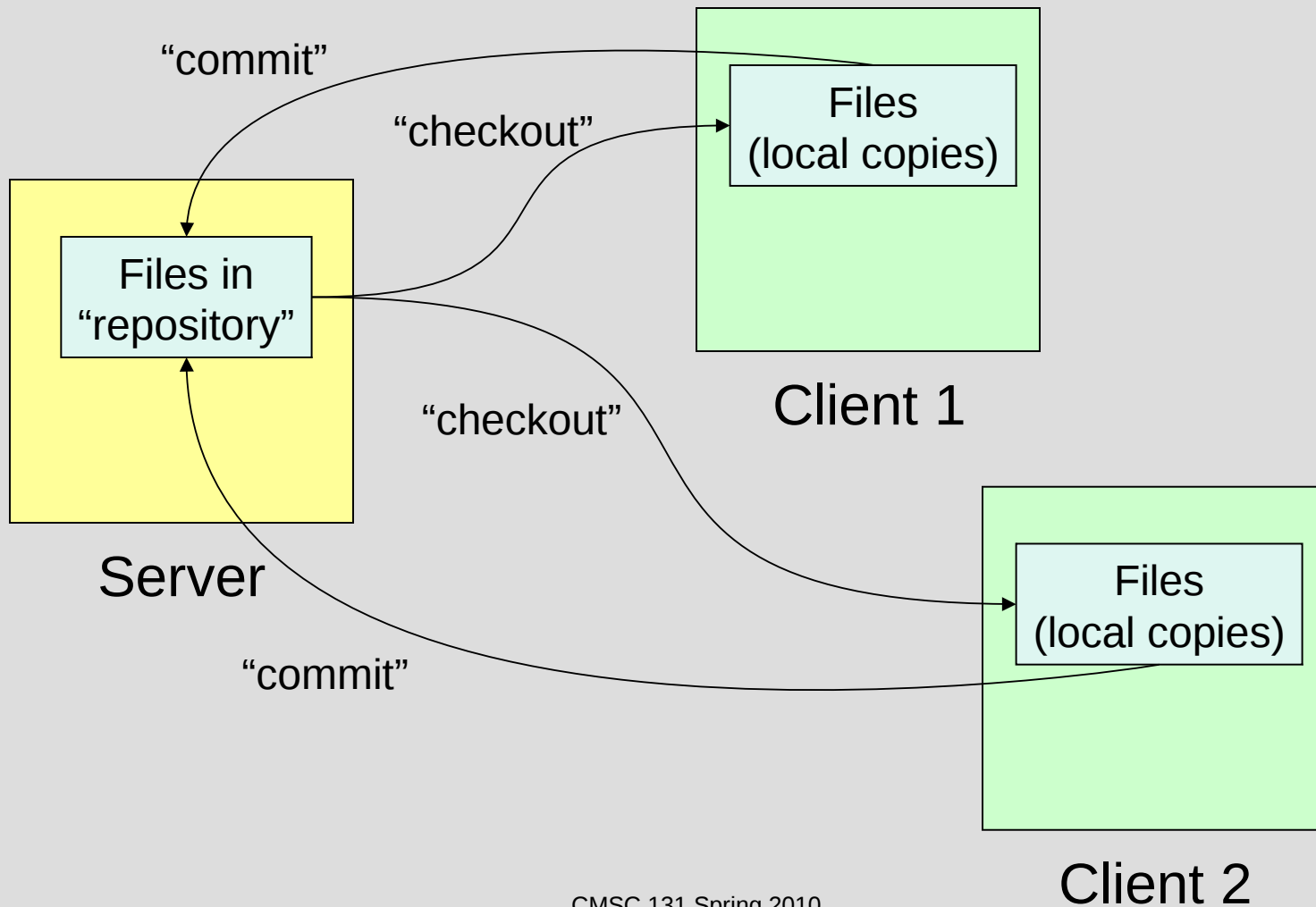
# Working on a project

- Under the **Java** perspective you will essentially see a tree view of your local projects.
- If you expand the **Spring2013-Lab01** node in this tree view, one of the sub-nodes will be src. Expand that and you will see (default package).
- If you expand that sub-node, you will see an entry for **History.java** – double click on that item to load that java source code file into your editor.
- You can now modify the code, compile it, and run the resulting program.

# Checking a project back in

- Each time you compile your project, any modified source code files will be checked back into the CVS repository.
- You can actually view a history of your changes and even compare an older version of a file to your current version.
- If you are working on your class projects from more than one machine, a useful feature is that if you right-click on a file in the project tree you can select **Replace With** and then **Latest from HEAD** to pull down the most recent version from your class CVS repository.

# CVS Worldview



# Submitting a project

- When you are ready to submit a project, right-click on the project item in the tree and select **Submit Project**.
- This will upload your project file(s) to the **submit.cs.umd.edu** server.
- If you want to perform a test of your submission, you can then log into the server and use one of your testing tokens and receive some feedback on your program's correctness.

# Project Testing

- First, test it locally yourself!
- Using the submit server, you can obtain results on:
  - Public Tests
  - Release Tests
- When we grade the projects we will use the public and release tests but might also use a set of **Private Tests**.

Copyright © 2012 : Evan Golub