SUBVERSION, NESTED LOOPS, BOOLEAN VALUES

CSSE 120—Rose Hulman Institute of Technology

Software Engineering Tools

- The computer is a powerful tool
- So use it to make software development easier and less error prone!
- Some software engineering tools:
  - IDEs, like Eclipse
  - Version Control Systems—like Subversion
  - Diagramming applications—like Violet or Visio
  - Modeling languages—like Alloy, Z, or JML

Version Control Systems

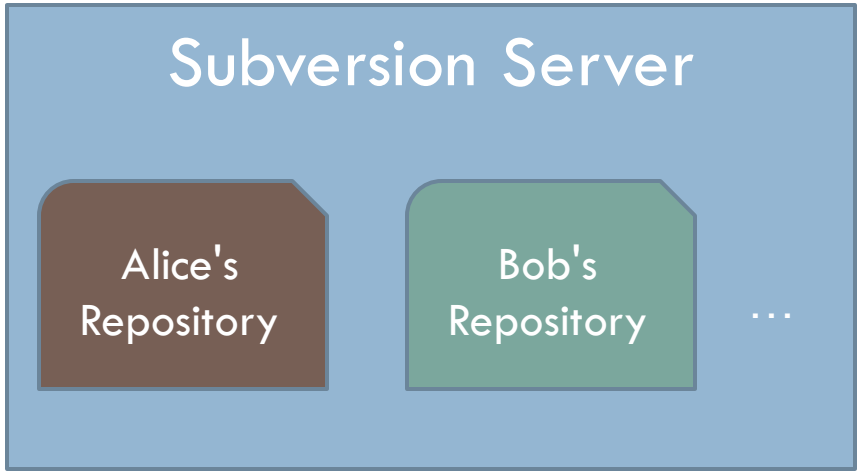
- Store "snapshots" of all the changes to a project over time
- Benefits:
  - Allow multiple users to share work on a project
  - Act as a "global undo"
  - Record who made what changes to a project
  - Maintain a log of the changes made
  - Can simplify debugging
  - Allow engineers to maintain multiple different versions of a project simultaneously

Our Version Control System

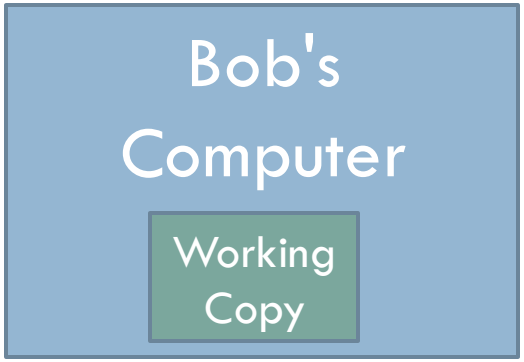
- Subversion, sometimes called SVN
- A free, open-source application
- Lots of tool support available
  - Works on all major computing platforms
  - TortoiseSVN for version control in Windows Explorer
  - Subclipse for version control inside Eclipse

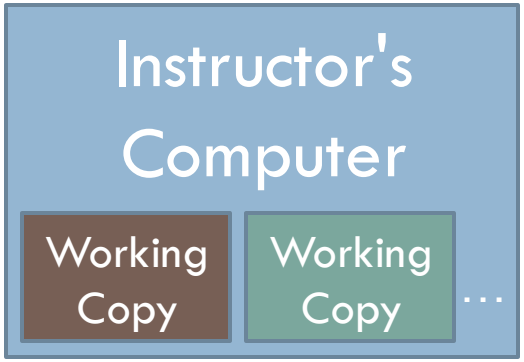
Version Control Terms

Repository: the copy of your data on the server, includes ***all*** past versions

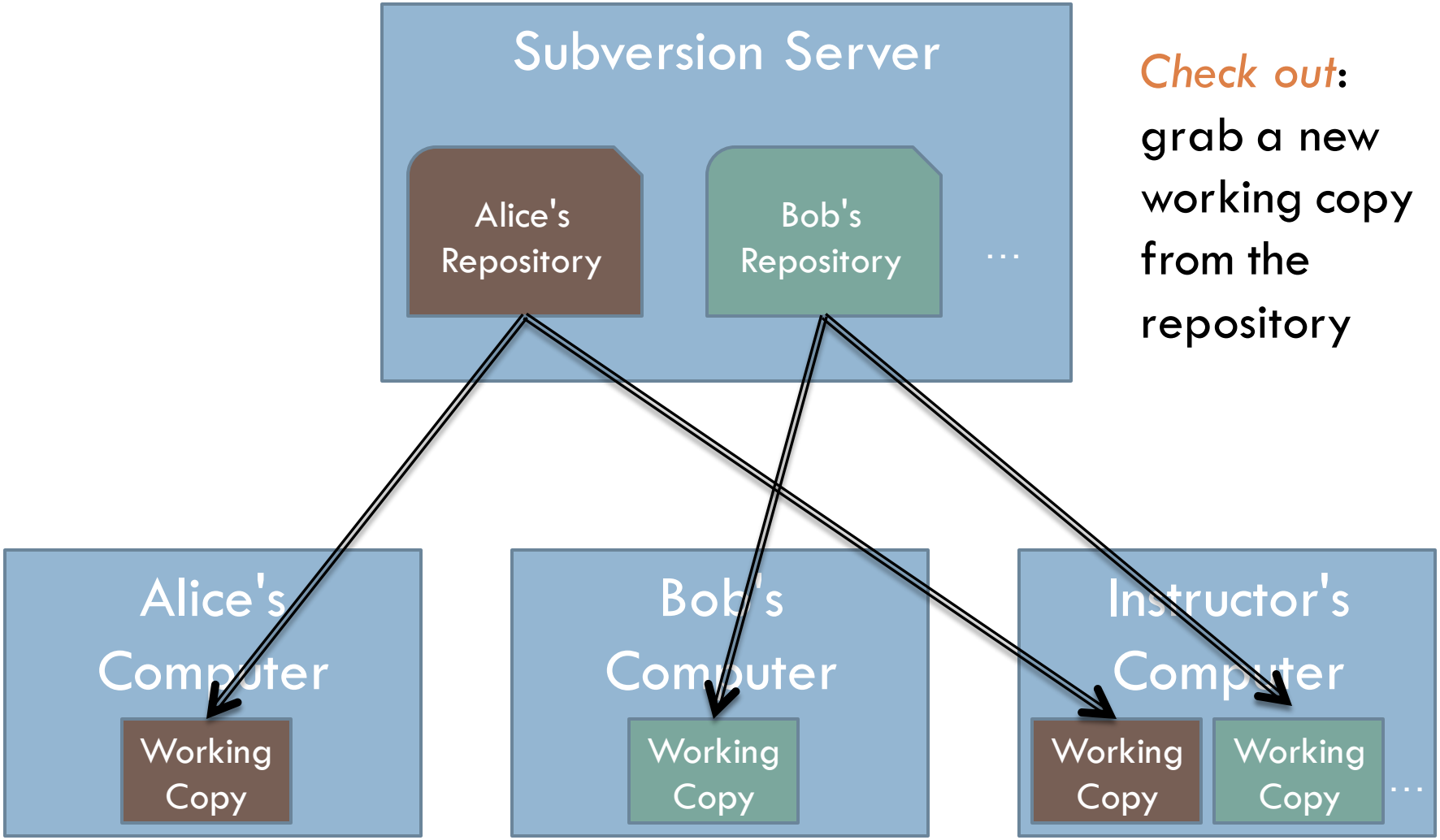


Working copy: the ***current*** version of your data on your computer



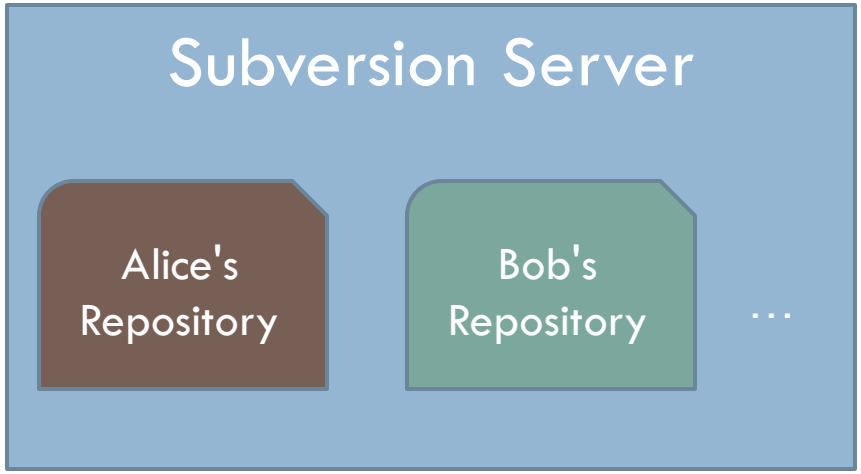


Version Control Steps—Check Out

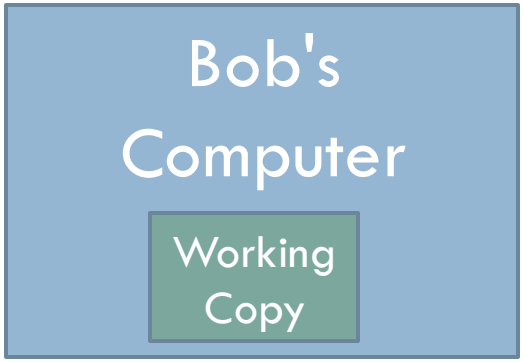


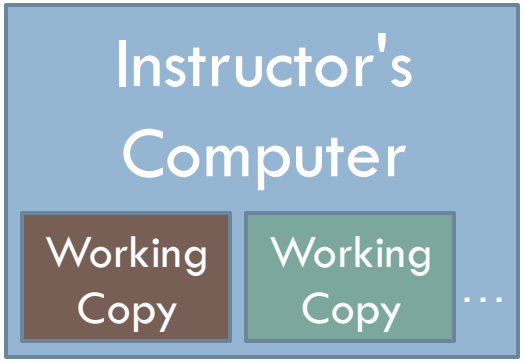
Check out: grab a new working copy from the repository

Version Control Steps—Edit

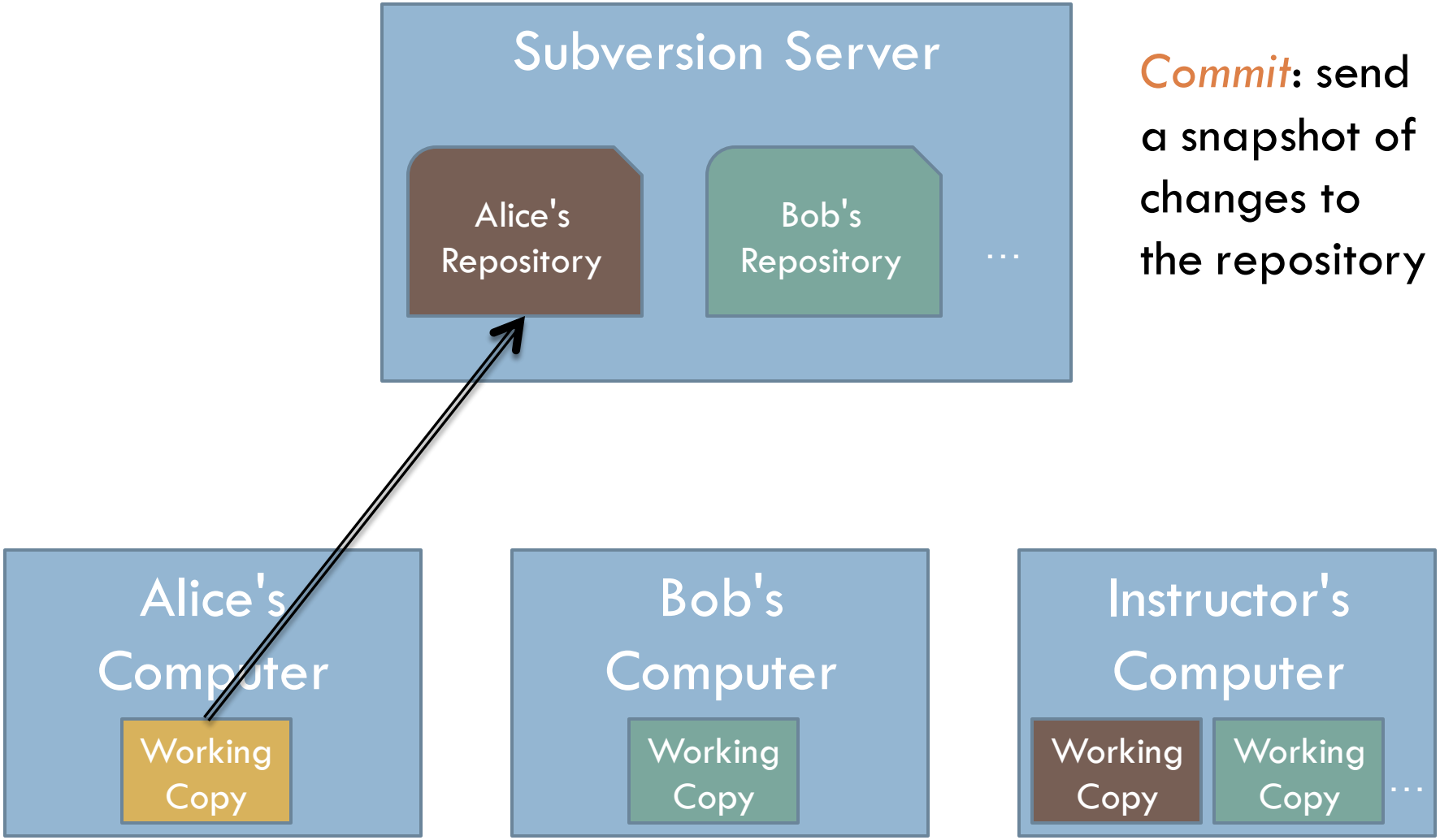


Edit: make ***independent*** changes to a working copy



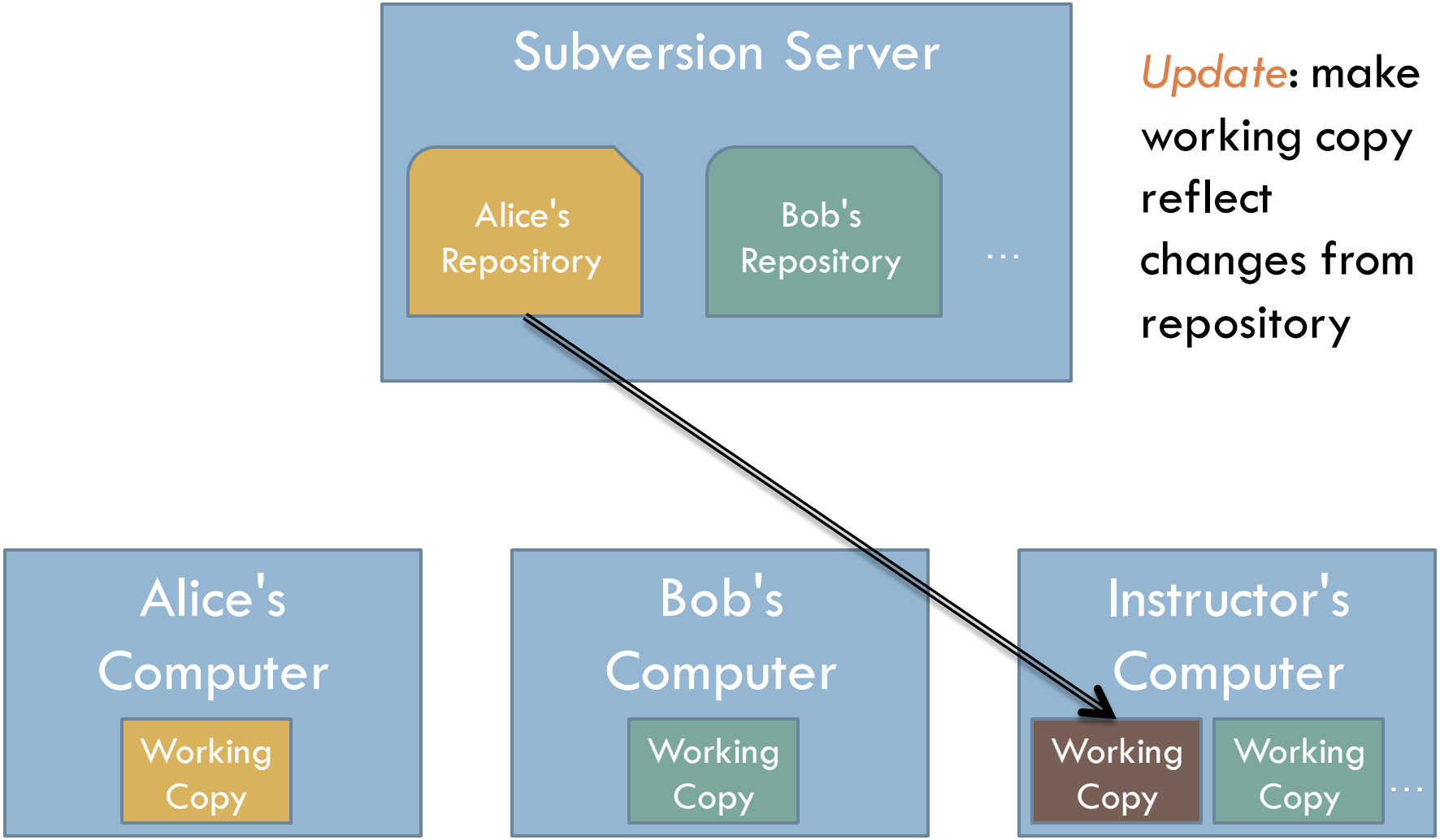


Version Control Steps—Commit



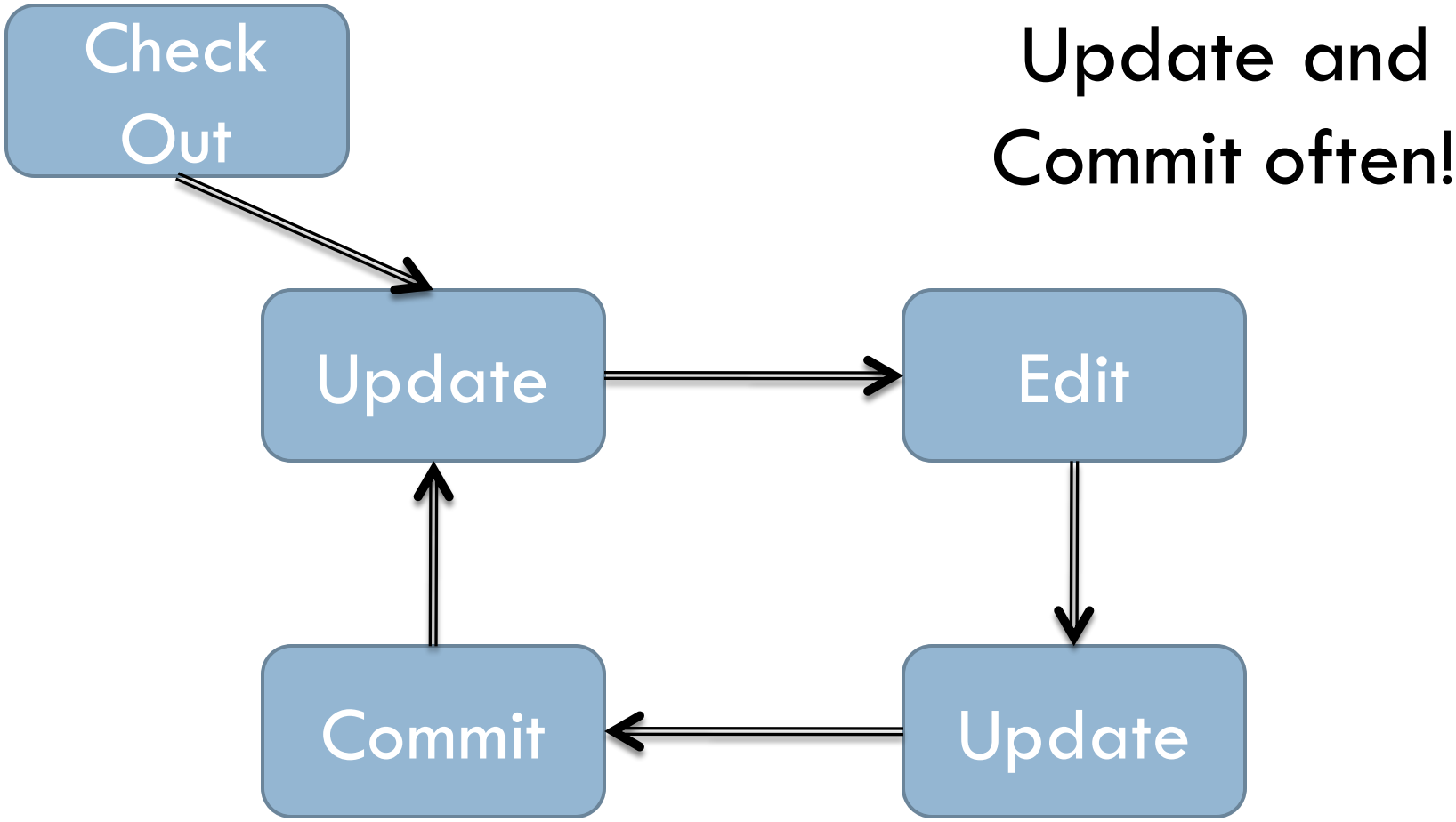
Commit: send a snapshot of changes to the repository

Version Control Steps—Update

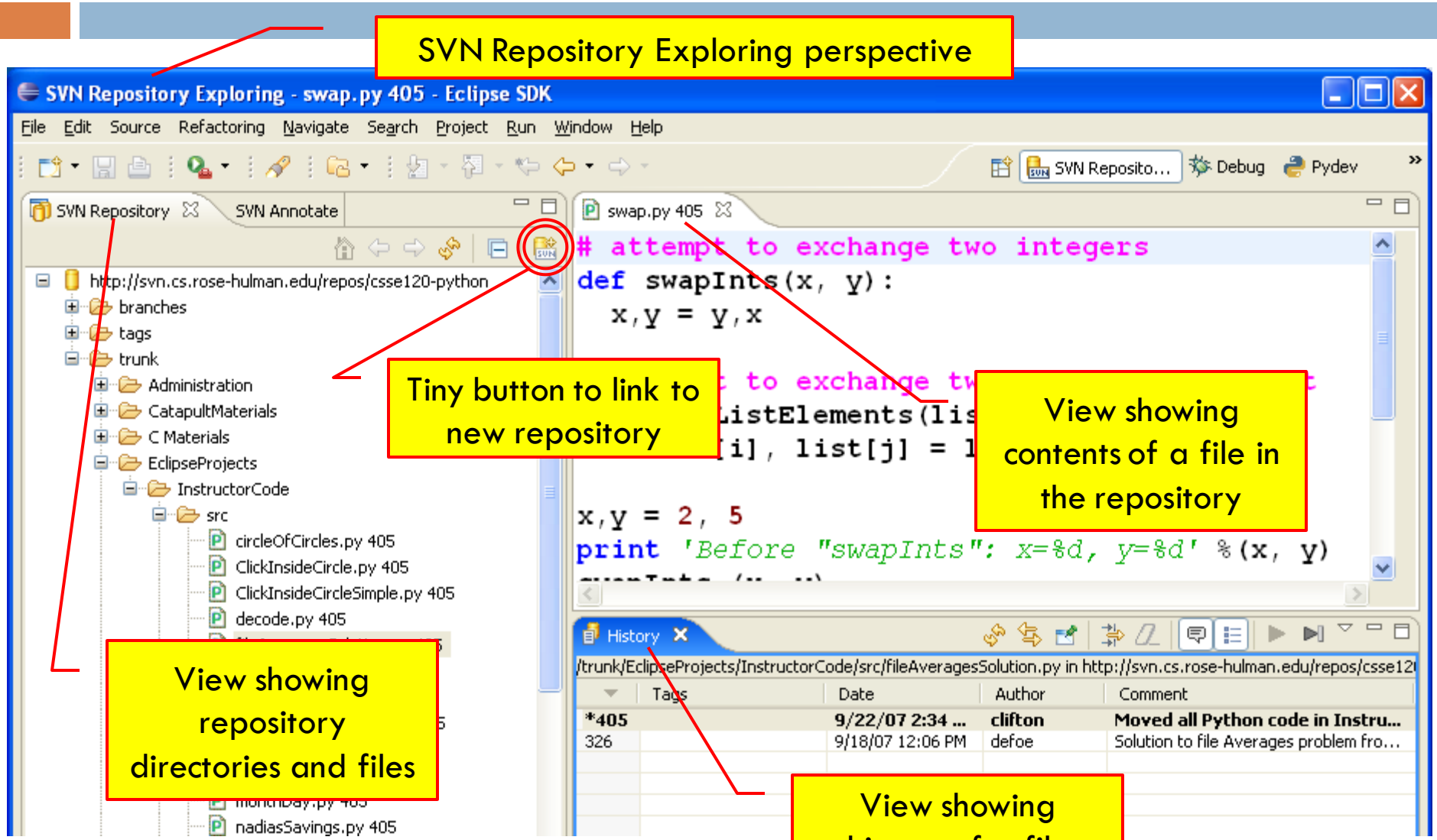


Update: make working copy reflect changes from repository

The Version Control Cycle



Subversion in Eclipse—Subclipse



Individual Exercise, Part 1

- Install (or verify) Subclipse on your laptop
  - ANGEL Course → Resources → Course Resources → Software Installation → Installing Subclipse
- Open the SVN Repository Exploring perspective
  - Window → Open Perspective → Other…
- Add a new repository:
  - Click the tiny icon: 
  - URL: **http://svn.cs.rose-hulman.edu/repos/csse120-fall07-*username*** replacing *username* with your actual username
  - **When prompted use the password that we emailed!**

Get help if you're stuck!
Go on to Part 2 if you're done.

Individual Exercise, Part 2

- Browse the SVN Repository view for the FirstSVNProject project
- Right-click it, and choose Check Out.
- Confirm all of the options presented
- Switch to PyDev perspective
- In Package Explorer, find spam.py inside your FirstSVNProject project
- Follow the instructions in the comments at the start of that file

Boolean Variables and Operations

- Boolean constants: **True** **False**
- Relational operators (<, etc.) produce Boolean values.

```
>>> 4 < 5
True
>>> 6 != 6
False
```

- Other Boolean operators: **and** **or** **not**

P	Q	P and Q
T	T	T
T	F	F
F	T	F
F	F	F

P	Q	P or Q
T	T	T
T	F	T
F	T	T
F	F	F

P	not P
T	F
F	T

Nested Loops

- A *nested if* is an **if** inside an **if.**
- A *nested loop* is a loop inside a loop.
- Example:

```
for i in range(4):
    for j in range(3):
        print i, j, i*j
```

- What does it print?
- What if we change the second range expression to `range(i+1)`?

Nested Loop Practice—Example

- Put this code inside NestedLoopPatterns.py in FirstSVNProject
- You will do several exercises that involve writing functions to generate patterned output.
- In each, you will accumulate each line's output in a string, then print it.
- First, a function to generate a pattern of asterisks like

```
**********
**********
**********
```

- To produce the above pattern, call `rectangleOfStars(3, 11)`

Nested Loop Practice – Your Turn

- Complete these definitions and test your functions
  - triangleOfStars(n) produces a triangular pattern of asterisks. For example, triangleOfStars(6) produces

```
*
**
***
****
*****
******
```

Hint: Use the same idea as the previous example. Start each line with an empty string. As you go through your inner loop, accumulate the line's characters. Print the line before the next iteration of the outer loop.

  - triangleOfSameNum(n) produces a triangular pattern of numbers. For example, triangleOfSameNum(5) produces

```
1
22
333
4444
55555
```

If you finish with these in class, continue with the remaining homework problems.