

# Class 11: Tools: Version Management with Subversion

**Held:** Tuesday, March 2, 2010

**Summary:** We consider the use and abuse of source code managers.

## Related Pages:

- EBoard.

## Notes:

- EC for Thursday's Thursday Extra and Friday's CS Table.
- Reading for Thursday: *Beautiful Code* Chapter 3
- Skimming for Thursday: The GNU Make Manual, chapters 1-5, 10.
- I will reserve time at the start of class to address the recent racial bias incident.

## Overview:

- An introduction to Subversion and source-code management.
- Subversion exercises.
- Beautiful Code: the delta editor.

## Source Code Management

- What it is: Software for managing collections of source code
- What do we need to manage?
  - Multiple people editing the same code
  - Multiple versions of the same code
  - Change logs (and documentation of those changes)
  - ...
- Most SCMs are also tied to development methodologies
  - A good SCM supports the methodology your development team has chosen.
  - Some SCM's effectively enforce a particular methodology.
- Typical SCM commands (what do you expect to be able to do)

## Subversion

- Subversion (which I tend to refer to as "svn") is a popular open-source source-code management system often used for open-source projects.
- Designed for distributed code management.
- Primarily used for agile practices.
  - Lots of small updates

- Good subversion practices
  - Always update before committing
  - Only commit working code
  - Try to commit one change at a time

## Hands-on Subversion

- Make a group of three students and two separate logins (different accounts).
  - You can use multiple computers.
  - You can use multiple terminal windows.
- Create a local repository available to both logins.
- Add a file to that repository with one account and obtain it with the other account.
- Write a small Python class that has two methods, add it to the repository, and make sure that both accounts have copies.
- Determine what happens if each account modifies a different part of the code (e.g., the two methods) and tries to commit the revision.
- Determine what happens if each account modifies the same part of the the code (e.g., the first line of the first method)
- Have one account modify the code and commit the code and ask the other to compare its current version to that in the repository.
- ...

## Subversion's Delta Trees

- What is confusing?
- What is beautiful?
- What is ugly?

## Project Discussion

- Tell us your stories.

---

Copyright © 2010 Samuel A. Rebelsky. This work is licensed under a Creative Commons Attribution-NonCommercial 2.5 License. To view a copy of this license, visit <http://creativecommons.org/licenses/by-nc/2.5/> or send a letter to Creative Commons, 543 Howard Street, 5th Floor, San Francisco, California, 94105, USA.