



SOFTWARE ENGINEERING: CONFIGURATION MANAGEMENT

Professors M. Brian Blake and Iman Saleh

Objectives

- Understand the motivation for Configuration Management (CM)
- Understand CM terminology
- Understand the tasks and roles associated with CM
- Be able to setup/use SVN for CM in your project

The First Law of System Engineering

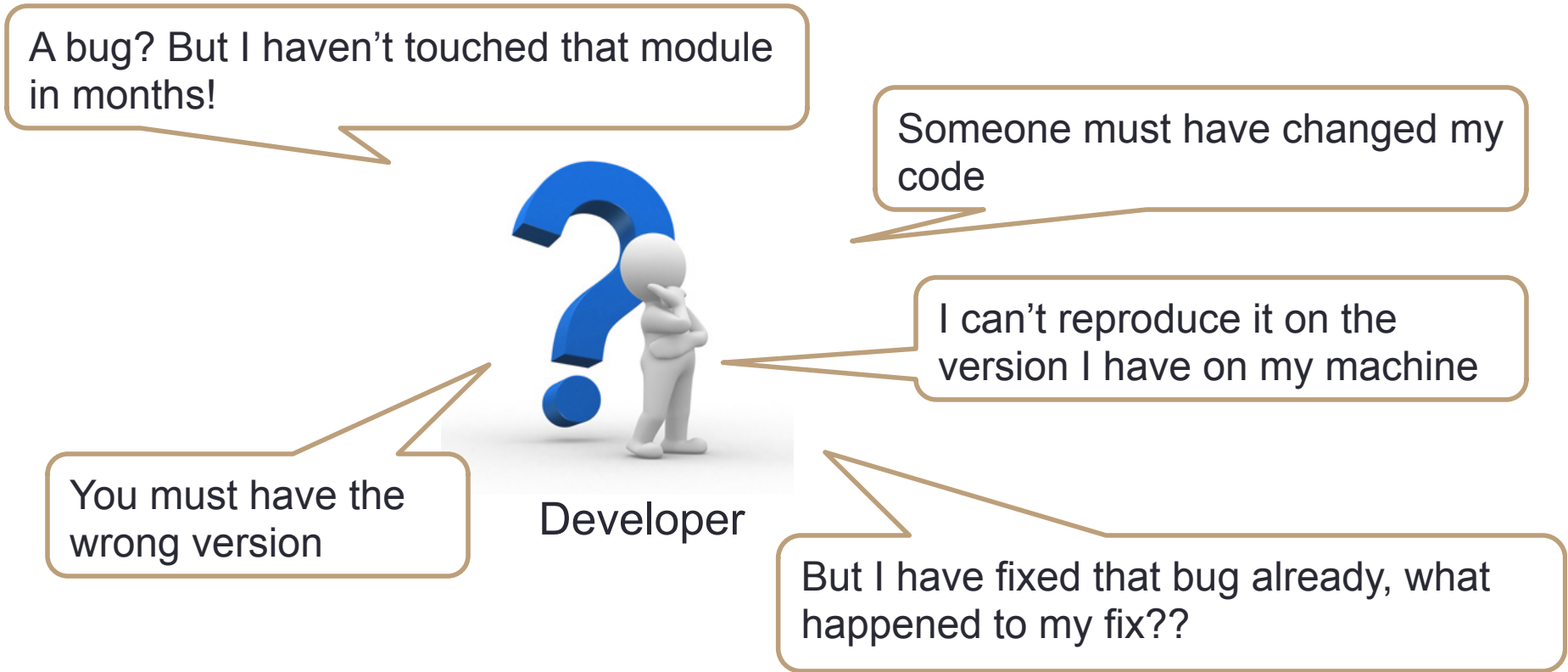
"No matter where you are in the system life cycle, the system will change, and the desire to change it will persist throughout the life cycle"

Bersoff, et al, 1980

What Changes?

- Requirements
- Code
- Technology
- Data
- Hardware
- Project Plan
- Users
- Developers

Signs of a CM problem...



The Problems

- Double Maintenance
 - Change must be applied to multiple copies of the same file.
- Shared Data
 - If a single copy of a file exists then only one person can work on it at a time.
- Simultaneous Update
 - If multiple developers can work on the same file simultaneously then changes can be overwritten.

What is Software Configuration Management?

Software Configuration Management encompasses the disciplines and techniques of initiating, evaluating and controlling change to software products during and after the software engineering process.

- Standards
 - IEEE 828: Software Configuration Management Plans
 - IEEE 1042: Guide to Software Configuration Management

Configuration Item (CI)

- *“An aggregation of hardware, software, or both, that is designated for configuration management and treated as a single entity in the configuration management process.”*

- **Examples:**

- A code file
- Drivers for tests
- Design document
- A manual



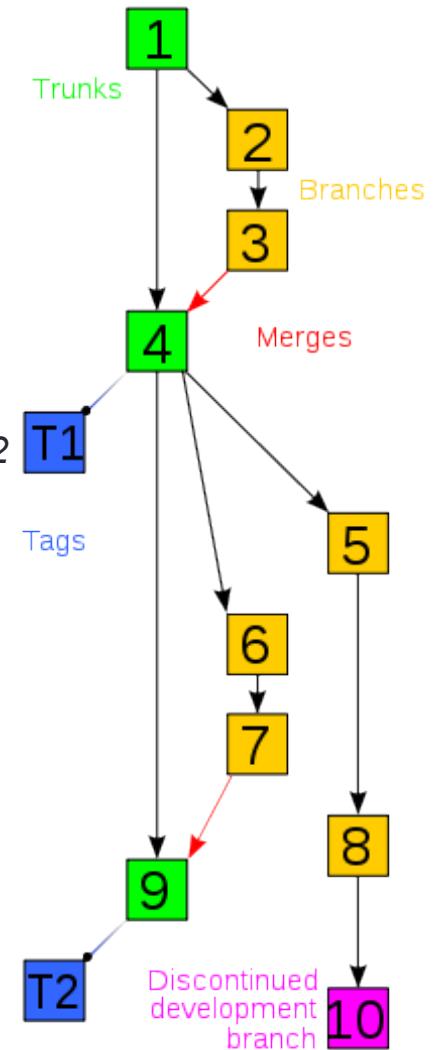
Baseline/Trunk

- “A specification or product that has been formally reviewed and agreed to by responsible management, that thereafter serves as the basis for further development, and can be changed only through formal change control procedures.”
IEEE Std 610.12

- Examples:

Baseline A: All the APIs have completely been defined; the bodies of the methods are empty.

Baseline B: All data access methods are implemented and tested.



Picture source: Wikipedia

Version/Revision

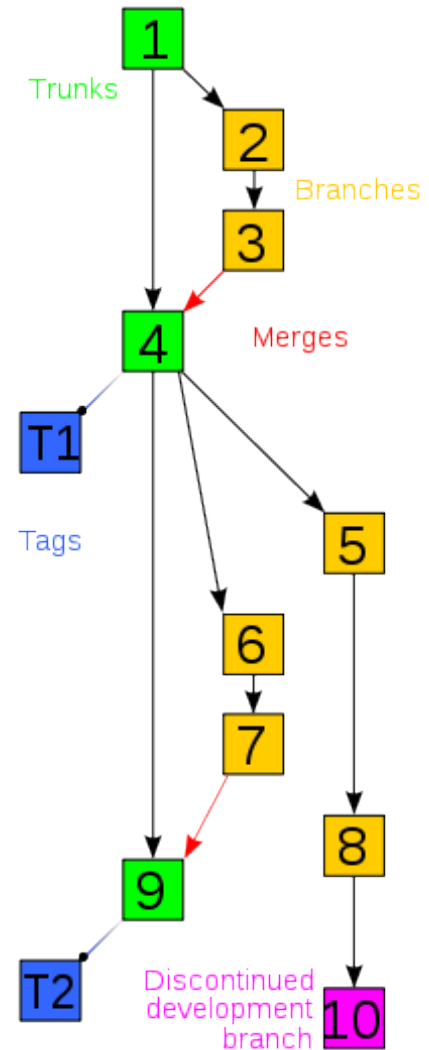
The state of a Configuration Item at a point in time.

Branch

A set of changes that are isolated in a separate development line.

Tag

A user-friendly label given to a collection of revisions at a certain point of time.



Picture source: Wikipedia

Configuration Management Activities

- **Configuration item identification**
 - modeling of the system as a set of evolving components
- **Promotion management**
 - is the creation of versions for other developers
- **Release management**
 - is the creation of versions for the clients and users
- **Change management**
 - is the handling, approval and tracking of change requests
- **Branch management**
 - is the management of concurrent development
- **Variant management**
 - is the management of versions intended to coexist

Configuration Management Roles

- **Configuration Manager**
 - Responsible for identifying configuration items. The configuration manager can also be responsible for defining the procedures for creating promotions and releases
- **Change control board member**
 - Responsible for approving or rejecting change requests
- **Developer**
 - Creates promotions triggered by change requests or the normal activities of development. The developer checks in changes and resolves conflicts
- **Auditor**
 - Responsible for the selection and evaluation of promotions for release and for ensuring the consistency and completeness of this release

Configuration Item Identification

- What should be under configuration control?
- When do you start to place entities under configuration control?

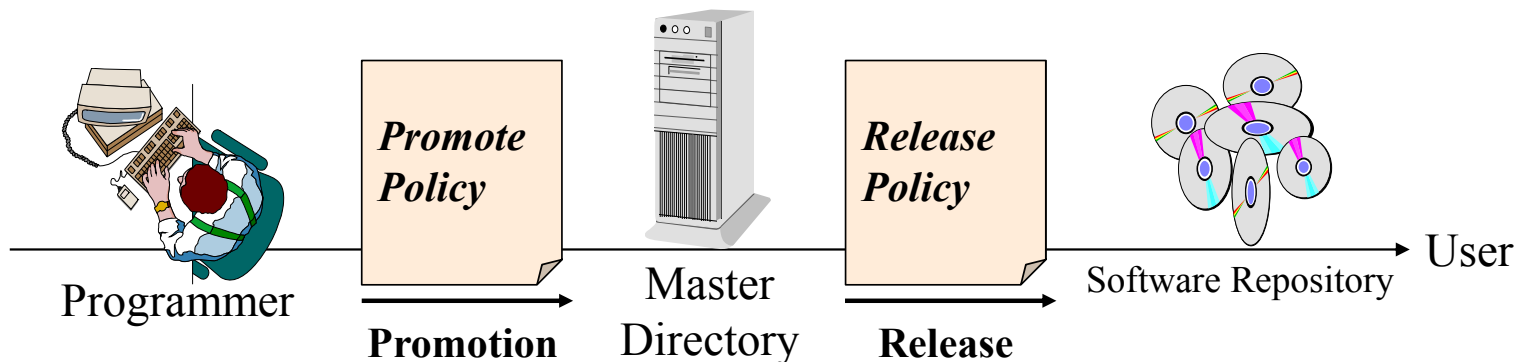
- Starting with CIs too early introduces too much bureaucracy
- Starting with CIs too late introduces chaos



Project Manager

Promotion and Release Management

- *Promotion*: The internal development state of a software is changed.
- *Release*: A changed software system is made visible outside the development organization.



Change Management

- General change process
 1. The change is requested (this can be done by users and developers)
 2. The change request is assessed against project goals
 3. The change is accepted or rejected
 4. If it is accepted, the change is assigned to a developer and implemented
 5. The implemented change is audited.
- Small projects can perform change requests informally and fast while complex projects require detailed change request forms and the official approval by one more managers.

Software Configuration Management Plan

- Defines the *types of documents* to be managed and a document naming scheme.
- Defines *who* takes responsibility for the CM procedures and creation of baselines.
- Defines *policies* for change control and version management.
- Describes the *tools* which should be used to assist the CM process and any limitations on their use.
- Defines the *configuration management database* used to record configuration information.

Software Configuration Management Plan

- Software configuration management planning starts during the early phases of a project.
- The outcome of the SCM planning phase is the *Software Configuration Management Plan* (SCMP) which might be extended or revised during the rest of the project.
- The SCMP can either follow a public standard like the IEEE 828, or an internal (e.g. company specific) standard.

Outline of a Software Configuration Management Plan (SCMP, IEEE 828-1990)

1. Introduction

- Describes purpose, scope of application, key terms and references

2. Management (WHO?)

- Identifies the responsibilities and authorities for accomplishing the planned configuration management activities

3. Activities (WHAT?)

- Identifies the activities to be performed in applying to the project.

4. Schedule (WHEN?)

- Establishes the sequence and coordination of the SCM activities with project mile stones.

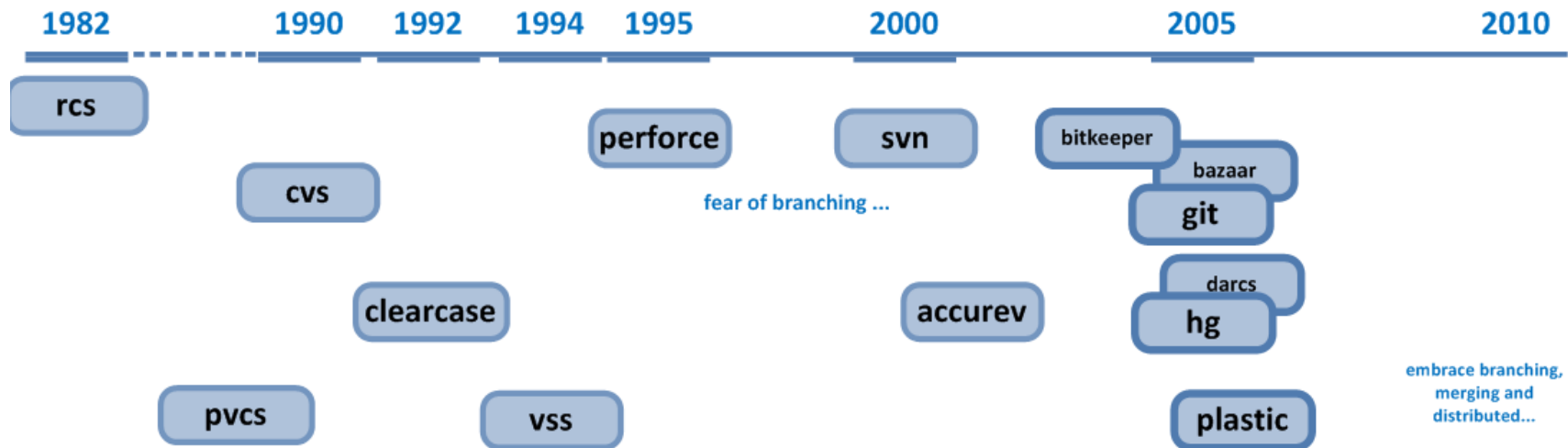
5. Resources (HOW?)

- Identifies tools and techniques required for the implementation of the SCMP

6. Maintenance

- Identifies activities and responsibilities on how the SCMP will be kept current during the life-cycle of the project.

Tools for Software Configuration Management



Source: <http://codicesoftware.blogspot.com/>



IN-CLASS EXERCISE

Define the Configuration Items and Directory Structure
for your Project

Example: Ruby On Rails

```
demo/  
.../app  
...../controller  
...../helpers  
...../models  
...../views  
...../layouts  
.../components  
.../config  
.../db  
.../doc  
.../lib  
.../log  
.../public  
.../script  
.../test  
.../tmp  
.../vendor  
README  
Rakefile
```



HOSTING PROJECTS USING GOOGLE CODE

Brief Introduction to GoogleCode and SVN

Slides are prepared by Yi Wei


Create a project on Google Code

- Go to <http://code.google.com/hosting/>
- Click the “Create a new project” link
- Fill out a simple form to describe the project and click the “Create project” button.
- Note: Please choose *Subversion* as the version control system.

Project Hosting on Google code

Explore project labels:

Python	JavaScript	Django	Web	CMS	Google
Java	AJAX	Rails	Plugin	Student	Android
CPlusPlus	MySQL	DotNet	Game	Windows	AppEngine
PHP	Flash	jQuery	Database	Linux	GWT

 [Create a new project](#)

 [Learn more about Project Hosting](#)

©2010 Google - [Terms](#) - [Privacy](#) - [Project Hosting Help](#)

Powered by [Google Project Hosting](#)

Project name

Example: my-project-name

Project summary

Description

Version control system

Source code license

Project label(s)

[add another row](#)



notredame-se-sample-project

Sample project for Software Engineering Course

Project Home

[Downloads](#)

[Wiki](#)

[Issues](#)

[Source](#)

[Administer](#)

Summary


[Updates](#)

[People](#)

Tip: Project owners, see our [Getting Started](#) guide for steps to configure your project. ✕

Project Information

Sample project

 [Star project](#)

[Activity](#)  None

Code license

[GNU General Public License v3](#)

Feeds

[Project feeds](#)

Owners

[ywei1@nd.edu](#)

Committers

[0 committers](#)

Contributors

[0 contributors](#)

[People details »](#)

[Project Home](#) [Downloads](#) [Wiki](#) [Issues](#) [Source](#) **Administer**

[Project Summary](#) **Project Members** [Downloads](#) [Wiki](#) [Issue Tracking](#) [Source](#) [Tabs](#) [Advanced](#)

Project members

Project owners:

ywei1@nd.edu

Project committers:

Project contributors:

Instructions:

Specify each project participant by his or her Google Account email address. Each person must have already created a Google Account with that email address.

Separate addresses with commas and/or newlines.

Project owners may make any change to this project.

Project committers may work in the project, but may not reconfigure it.

Project contributors start with the same permissions as non-members, but their role in the project is visible.

Additional permissions can be granted to committers and contributors on the project's [people](#) sub-tab.

Save changes

Introduction to SVN

- Version control of source/document files used to be a pain in many projects.
- Different versions of files created/modified by different group member often cause confusion and confliction.
- The Concurrent Versioning System (CVS) is the first solution to this problem.
- SVN aims at becoming the successor of CVS.

Features of SVN

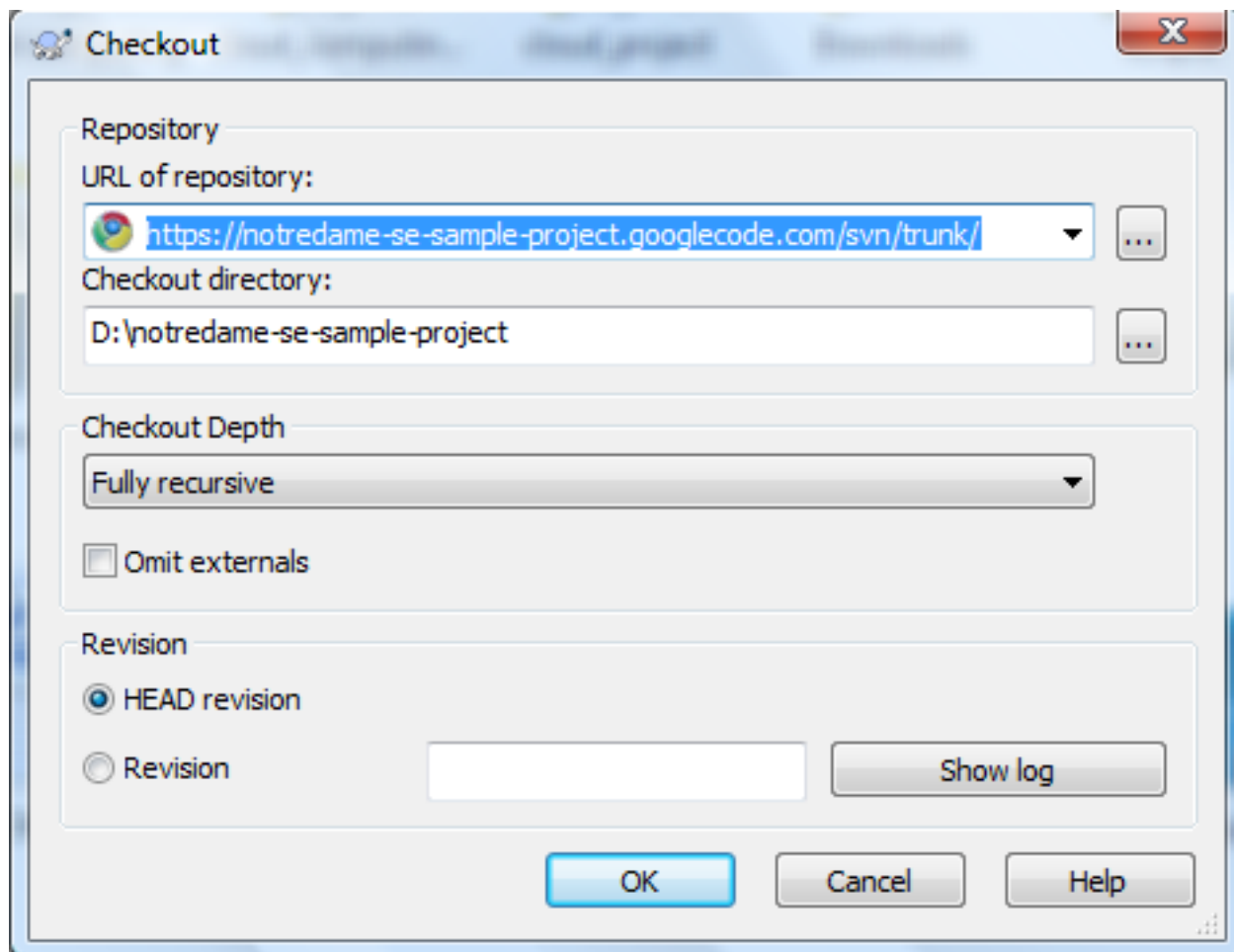
- System maintains versioning for directories, renames and metadata.
- Every change is recorded as a revision. Users can revert to an earlier revision whenever they want.
- Support branching to facilitate parallel development.
- Support file properties to enable fine granularity control.
- Conflict resolve and merge operation for different versions of the same file.

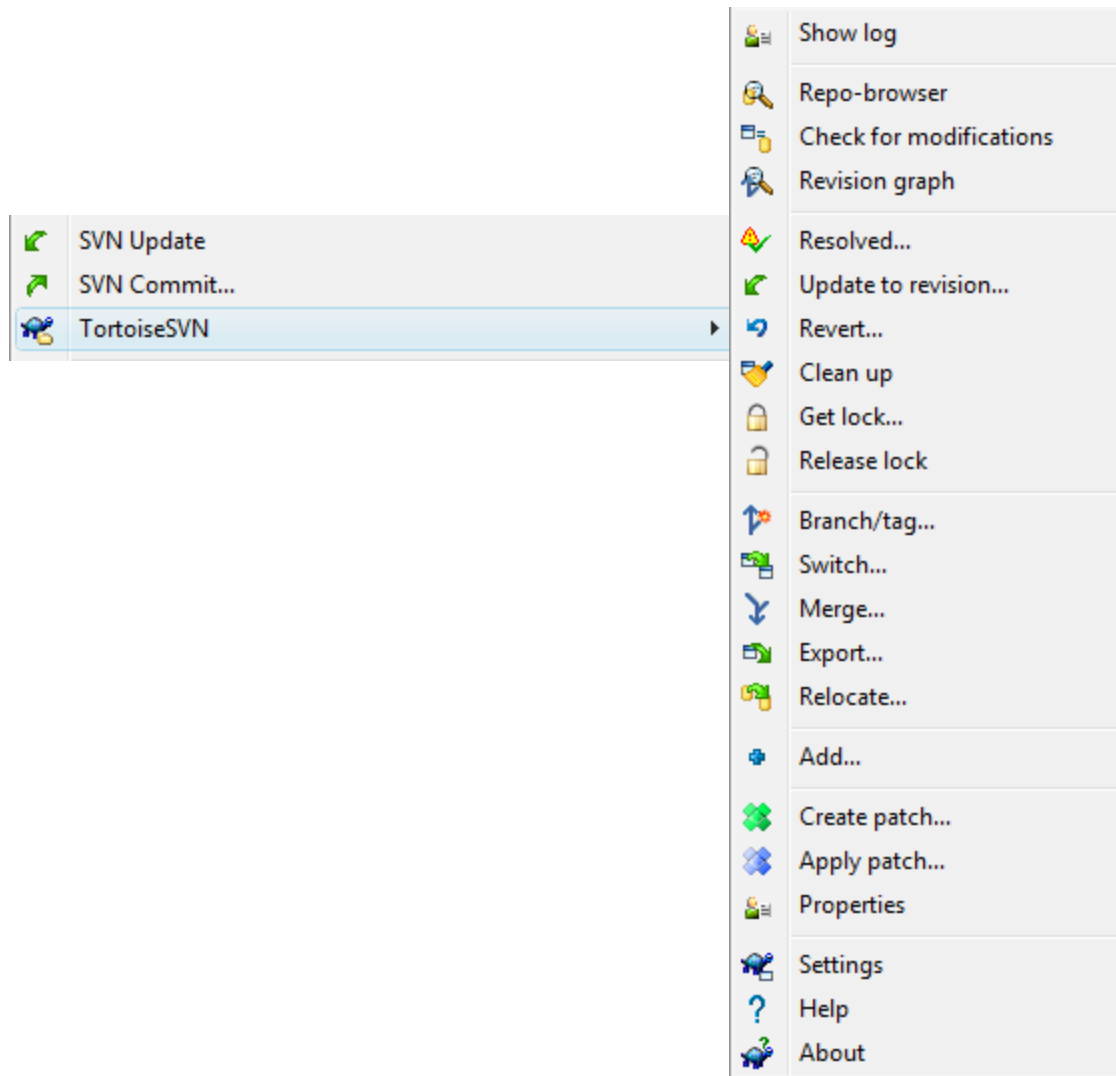
Use SVN on your computer

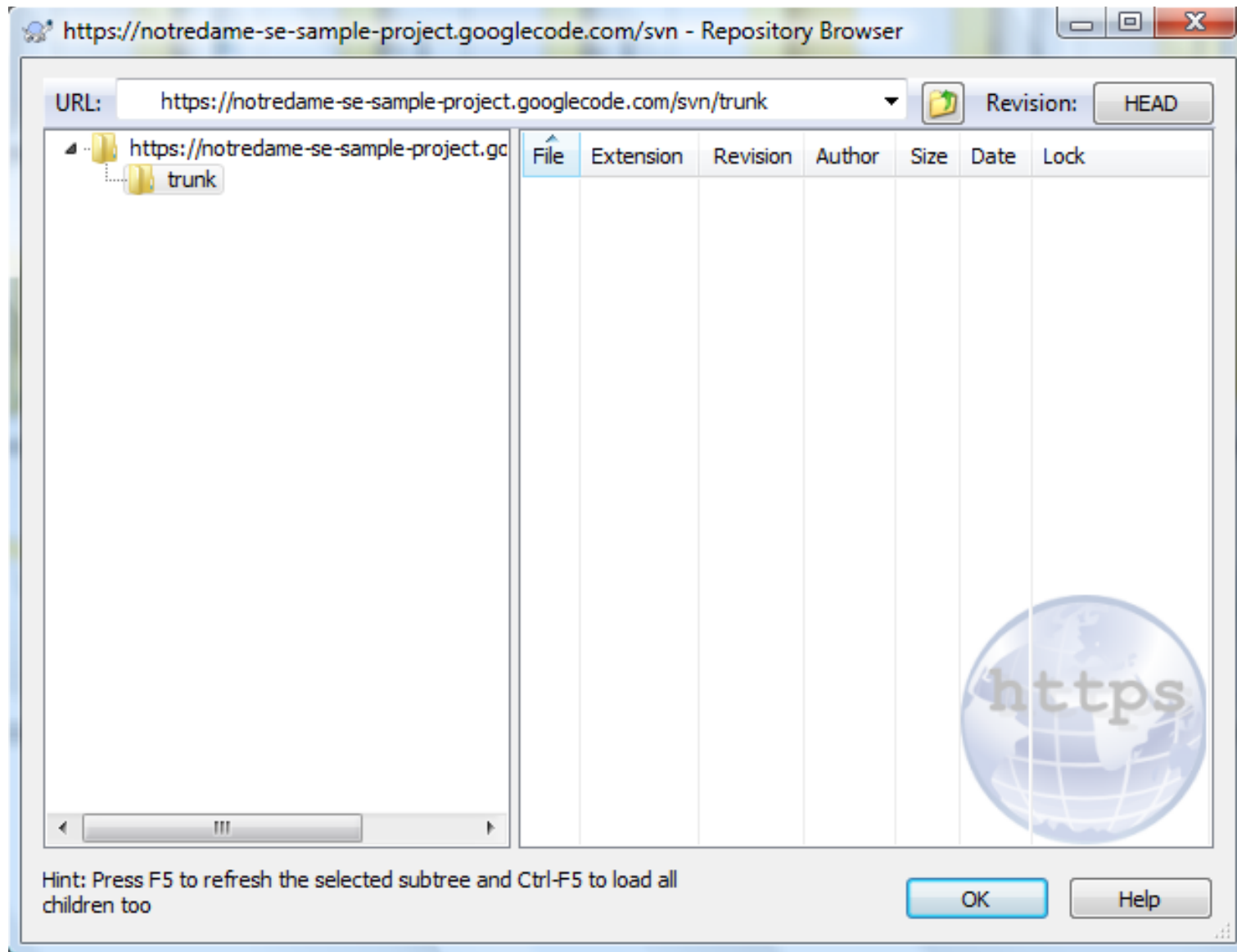
- Open source Tortoise SVN client for Windows users.
- Should be available on most Linux or Mac OS systems.
- If not, go to <http://subversion.apache.org/packages.html> for OS specific instructions.
- Eclipse users can download a plug-in called Subversive.
- Type “svn help” in a Linux shell for a list of svn commands.

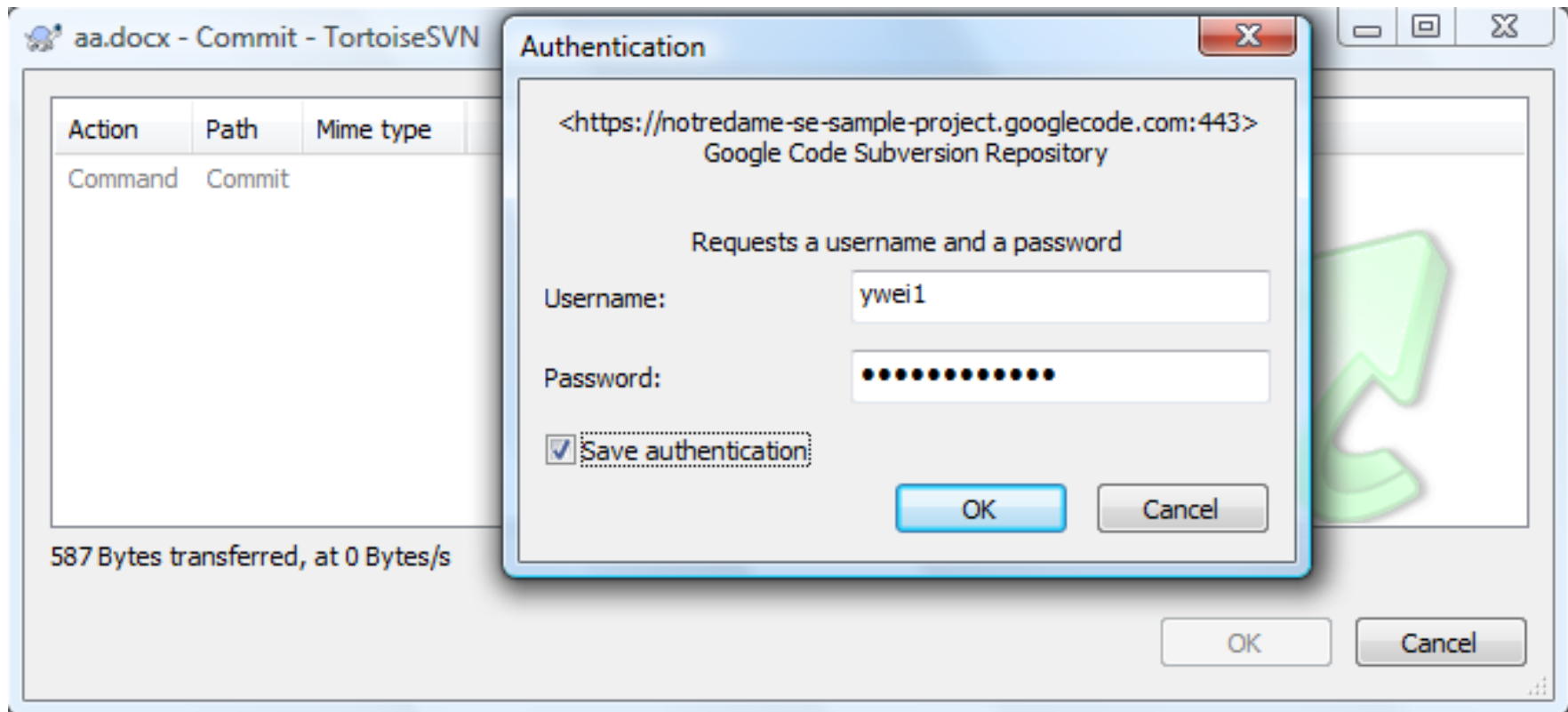
Proposed Mac Installation

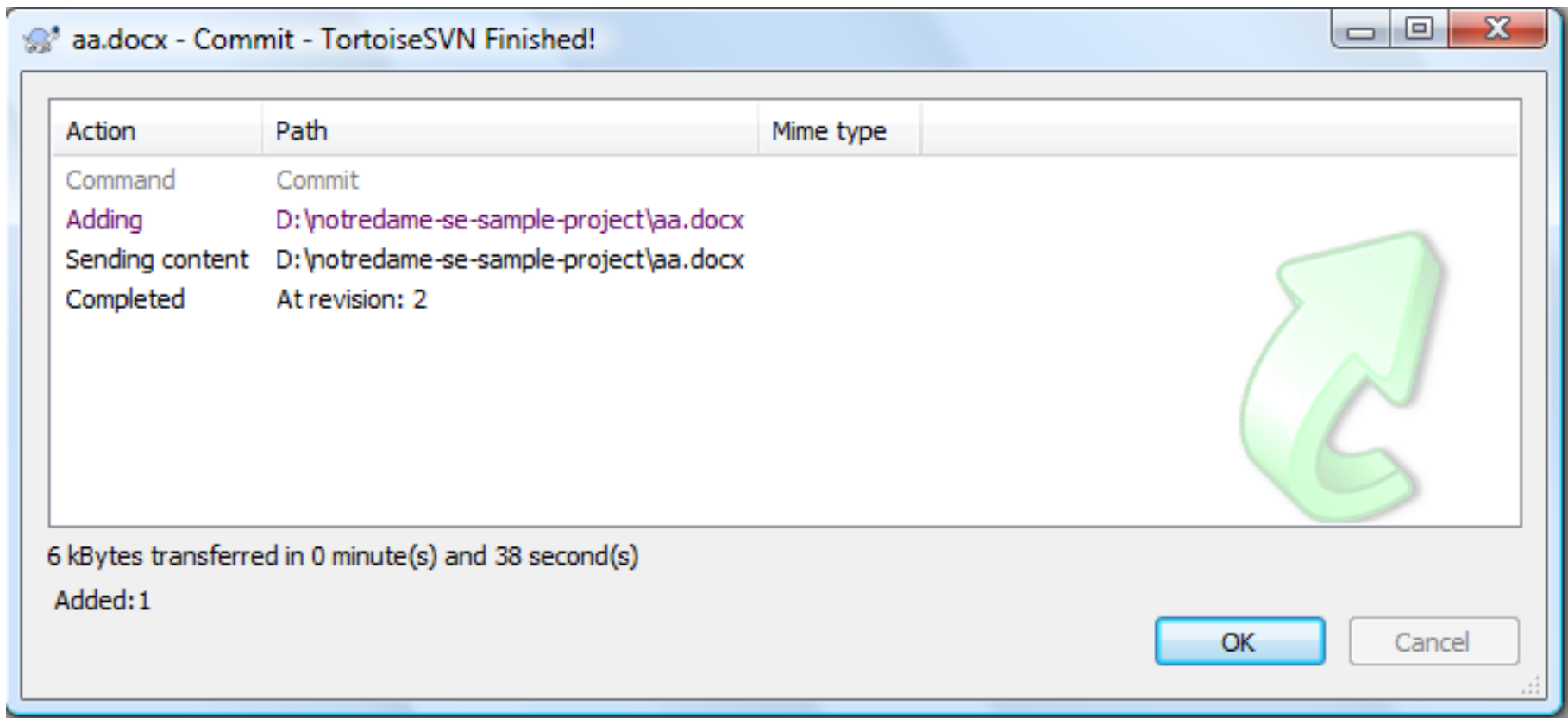
- Install Xcode
 - Make sure to install the command-line tools.
 - Go to Preferences -> Downloads to double check
- Install MacPorts
- Install svn using: `sudo port install subversion`











Common Workflow

- Checkout a new repository or update an existing one.
- Develop, debug on local machine.
- When you are confident that the code is correct, commit the change to the repository.
- Sometimes there may be conflicts due to multiple members revising the same file.

Most often used SVN commands

- **checkout**: Checkout the repository to a local directory.
- **update**: Sync the local directory with the repository.
- **commit**: Commit the changes to the repository
- **add**: Add a local file under version control.
- **delete**: Remove a file from version control.
- **propset**: Set SVN properties for a file.
- **revert**: Discard local changes.
- **help**: Display help messages.

Other features of Google Code



★ Home

One-sentence summary of this page.

Introduction

This is the wiki of a sample project.

Details

Add your content here. Format your content with:

- Text in **bold** or *italic*
- Headings, paragraphs, and lists
- Automatic links to other wiki pages

[Project Home](#) [Downloads](#) [Wiki](#) **Issues** [Source](#) [Administer](#)

[New issue](#) [Search](#) Open issues for [Advanced search](#) [Search tips](#)

Tip: Type ? for issue tracker keyboard shortcut help.

Select: [All](#) [None](#)

	ID ▼	Type ▼	Status ▼	Priority ▼	Milestone ▼	Owner ▼	Summary + Labels ▼
<input type="checkbox"/> ★	1	Defect	New	Medium	---	ywei1@nd.edu	It doesn't work!

[Project Home](#) [Downloads](#) [Wiki](#) **Issues** [Source](#) [Administer](#)

[New issue](#) [Search](#) All issues for [Advanced search](#) [Search tips](#)

Tip: Type ? for issue tracker keyboard shortcut help.

Select: [All](#) [None](#)

	ID ▼	Type ▼	Status ▼	Priority ▼	Milestone ▼	Owner ▼	Summary + Labels ▼
<input type="checkbox"/> ★	1	Defect	Fixed	Medium	---	ywei1@nd.edu	It doesn't work!



Any Questions?

iman@miami.edu