Version Control with Subversion
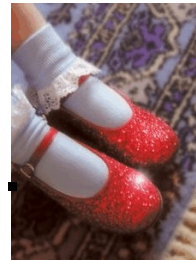
Peter J. Farrell
Maestro Publishing

Our Goals

Some questions I want you to be able to answer in the end:

- What is version control?
- Why you should use version control?
- How does the version control work?
- How does Subversion work?

- It's ok to ask questions…

So What is Version Control?

Your lifesaver in a sea of changes!

- Manage the evolution of digital documents

- Tracks changes to data over time and who makes these changes

Why Do You Need Version Control?

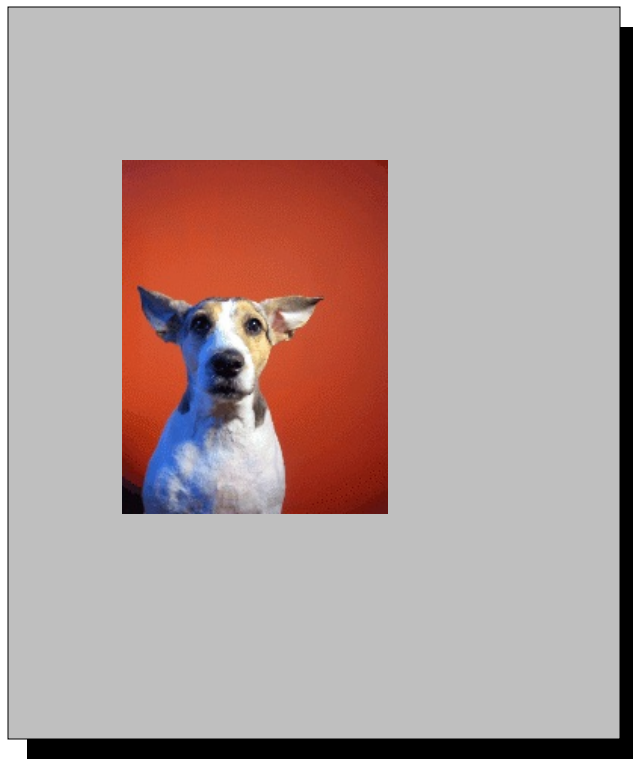Because you're working in the stone age if you don't use it.

- Centralized system for sharing information
- Manage all changes to data over time
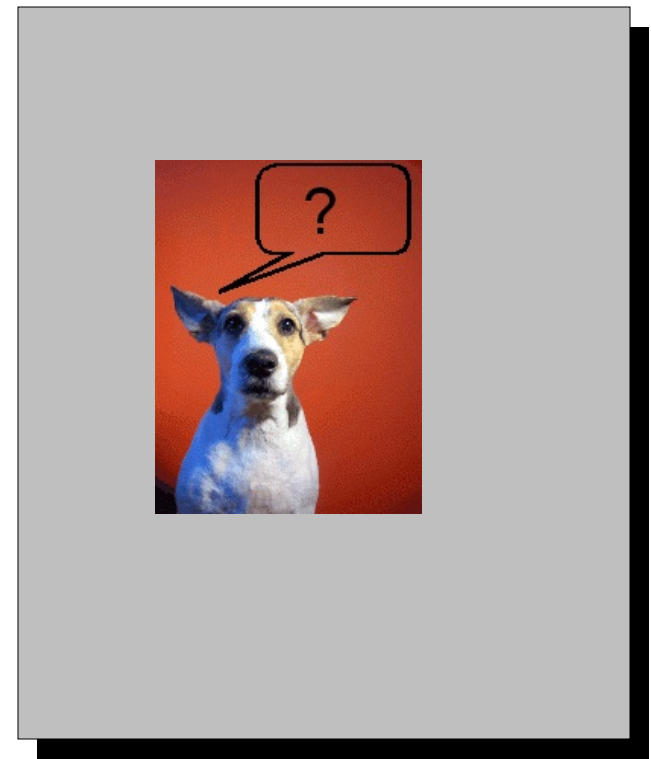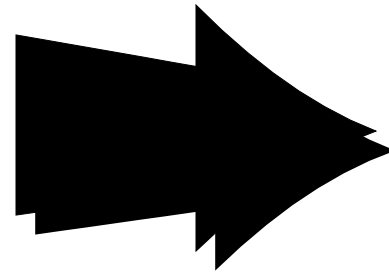- Allows teams to work on the same source code at the same time

Simple Version Control Example

A cop of your important document

Version 1

Version 2

In Short...

A digital librarian and source code traffic cop!

Basically, version control systems keep track of all changes to the central repository.

In turn, this makes solo/team work faster, easier and cleaner.

With version control, you have evey version ever commited to the repository forever including a bunch of nifty features.
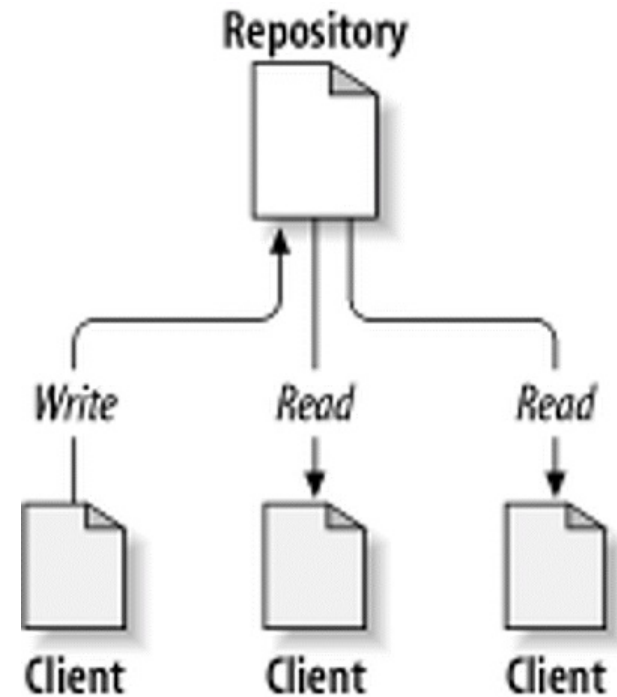
Basic Concepts

A centralized system for sharing information/data.

Most version control systems use the client/server model.

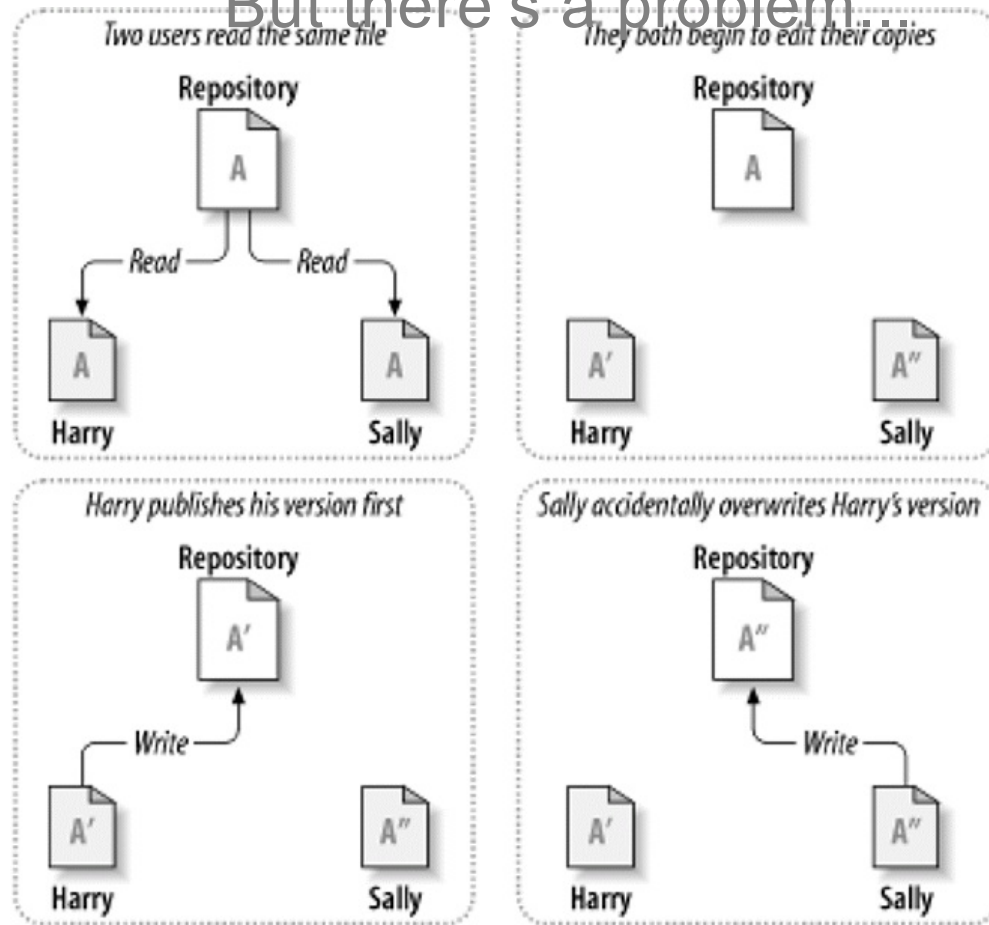Please see author attribution in Appendix

So It's Just A Big File Server?

But there's a problem...

Lock-Modify-Unlock Solution

Sort of like Dreamweaver's check-in/check-out
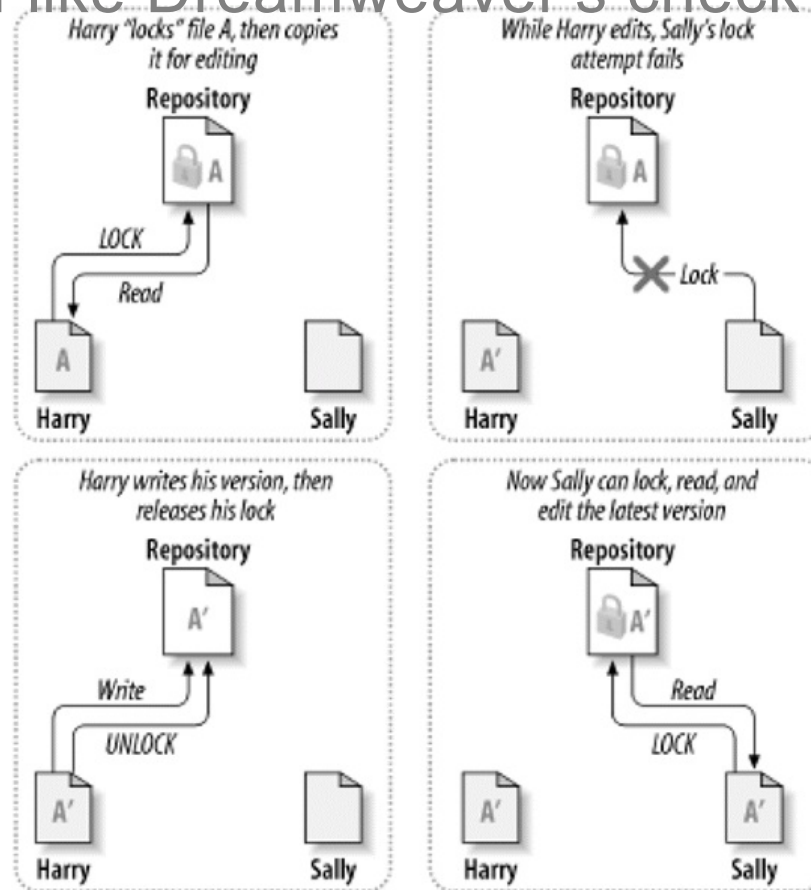
Copy-Modify-Merge Solution

The solution used by Subversion and CVS.
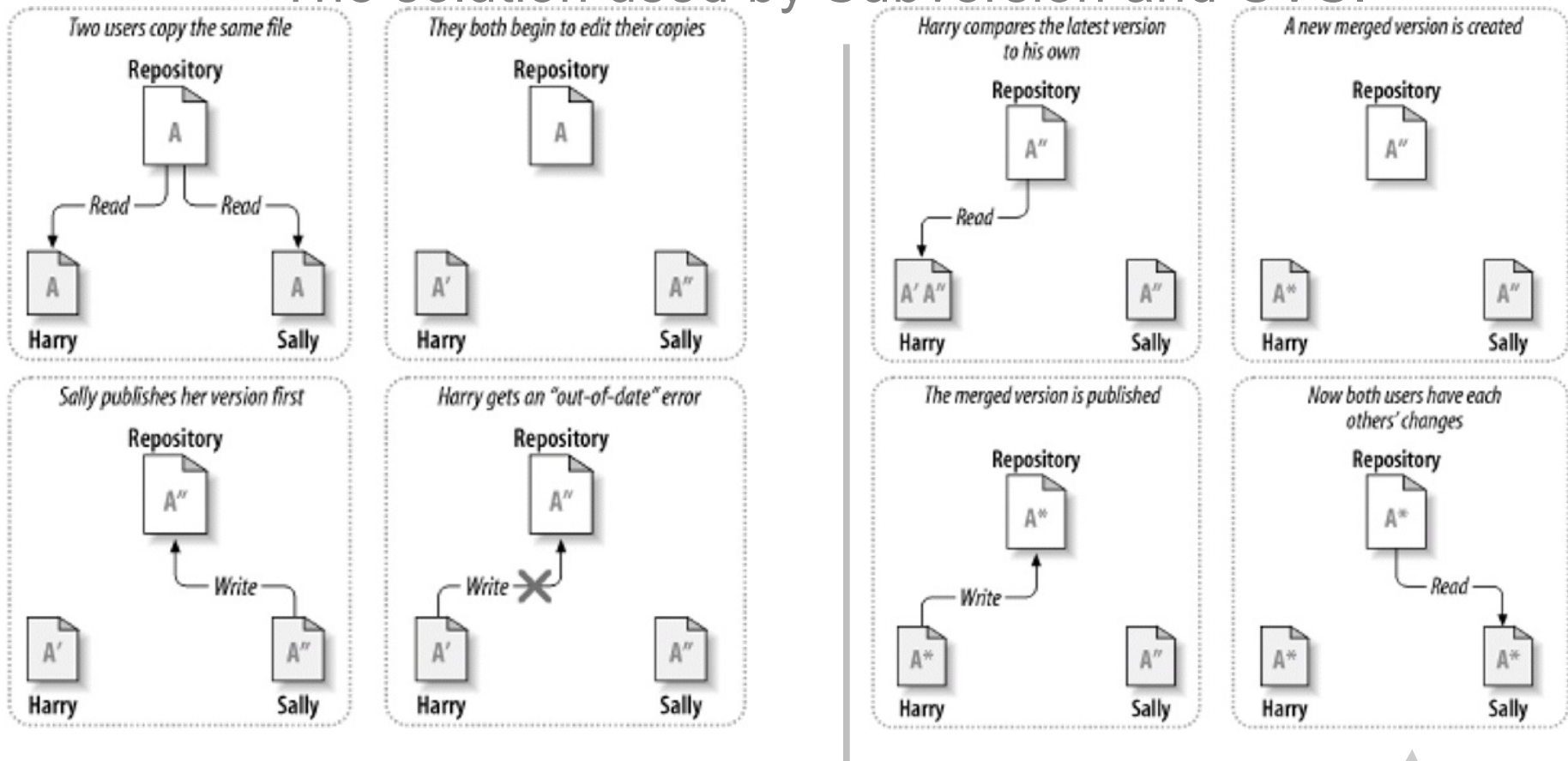
Version Control Choices

- Subversion - SVN
  - ▸ Free/open source
- CVS
  - ▸ Free/open source
- Microsoft Visual Source Safe -VSS
  - ▸ Not free
- And many more...
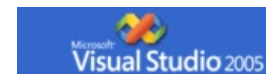
Why Not VSS?

- Not free
  - ‣ Unless you have a MSDN subscription
- Microsoft does not even use it internally
  - ‣ With the exception of two small projects
- Not geared to "team" development
- Sometimes has problems with binary files
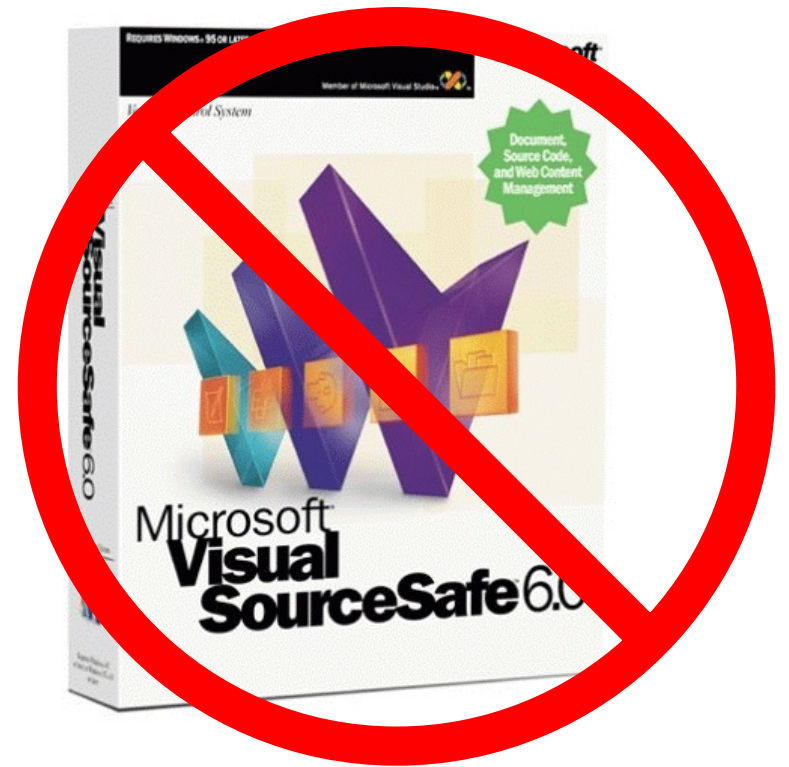
Why I Chose SVN Over the Others

- Total cost per developer

- Features
  - ‣ Or limitations associated with a specific solution

- Integration with development tools
  - ‣ Clients like Subclipse or TortoiseSVN
  - ‣ Intergrates easily with Apache Web Server and WebDAV

- Platforms

Common Gripes About the Limitations of CVS

SVN was designed as modern replacement for CVS and to over come some of the limitations that CVS possesses.

- Version controlled files cannot be renamed after the initial commit.
  - ▸ To rename a file, it must be removed from version control and re-added.

- Directories cannot be moved or renamed.
  - ▸ If you want to rename a directory, you must explicitly deleted and re-add all the files in the directory.

- Limited support for Unicode files
  - ▸ Including filenames with non-ASCII characters

Quick Facts About Subversion

- **How much does SVN cost?**
  - ▸ Free and open source

- **How is SVN licensed?**
  - ▸ It appears that SVN's license is similar to the Apache 1.0 license.

- **What platforms does SVN run on?**
  - ▸ A most Windows platforms to different favors of *nix including Mac OS X.

- **What about documention?**
  - ▸ SVN has great resources including a free online book published by O'Reilly

Key Terms - Part One

- Repository
  - ▸ The repository is where the files are stored -often on a server.

- Commit
  - ▸ A commit occurs when you copy the changes you made your **working copy** to the repository.

- Change
  - ▸ A change represents a specific modification to a file under version control.

- Check-Out
  - ▸ A check-out copies a **working copy** from the repository.

- Update/Sync
  - ▸ An update copies the changes that were made to the repository into your **working copy** (opposite of a commit).

Key Terms - Part Two

- Merge
  - ▸ A merge brings together (merges) concurrent changes into a unified revision.

- Revision/Version
  - ▸ A revision or version is one version in a chain of changes.

- Conflict
  - ▸ A conflict occurs when two changes are made by different parties to the same file or place within a file. Since the software is not intelligent enough to decide which changes are 'correct' – a user is required to resolve the conflict.

- Resolve
  - ▸ The act of user intervention to address a conflict between different changes to the same file.
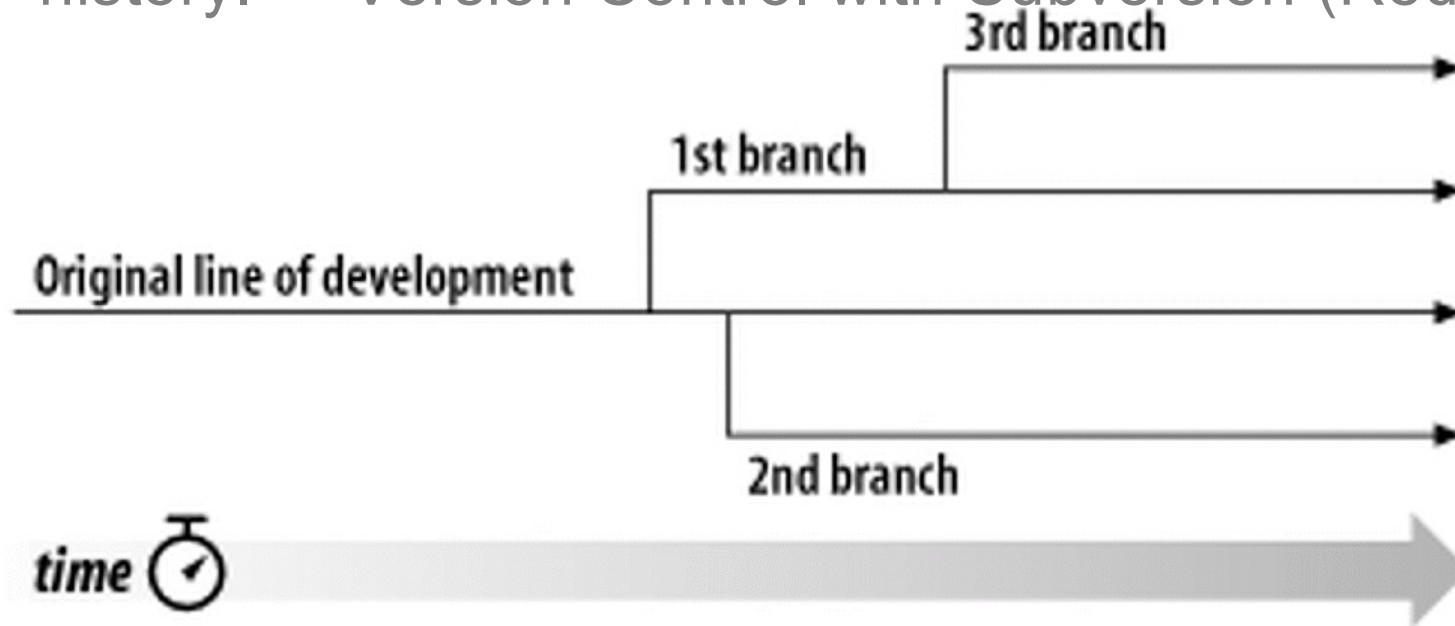
Typical Repository Setup
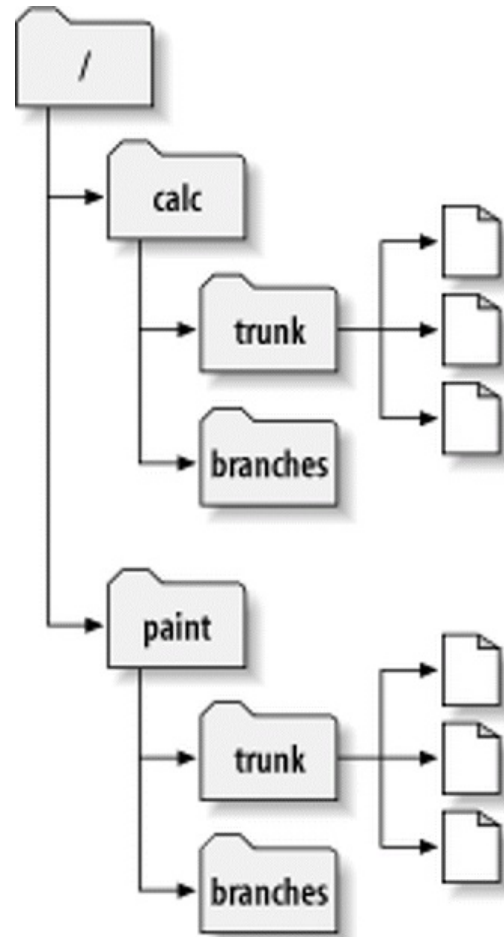
What is a Branch?

"A line of development that exists independently of another line, yet still shares a common history if you look far enough back in time. A branch always begins life as a copy of something, and moves on from there, generating its own history." – Version Control with Subversion (Red-Bean)
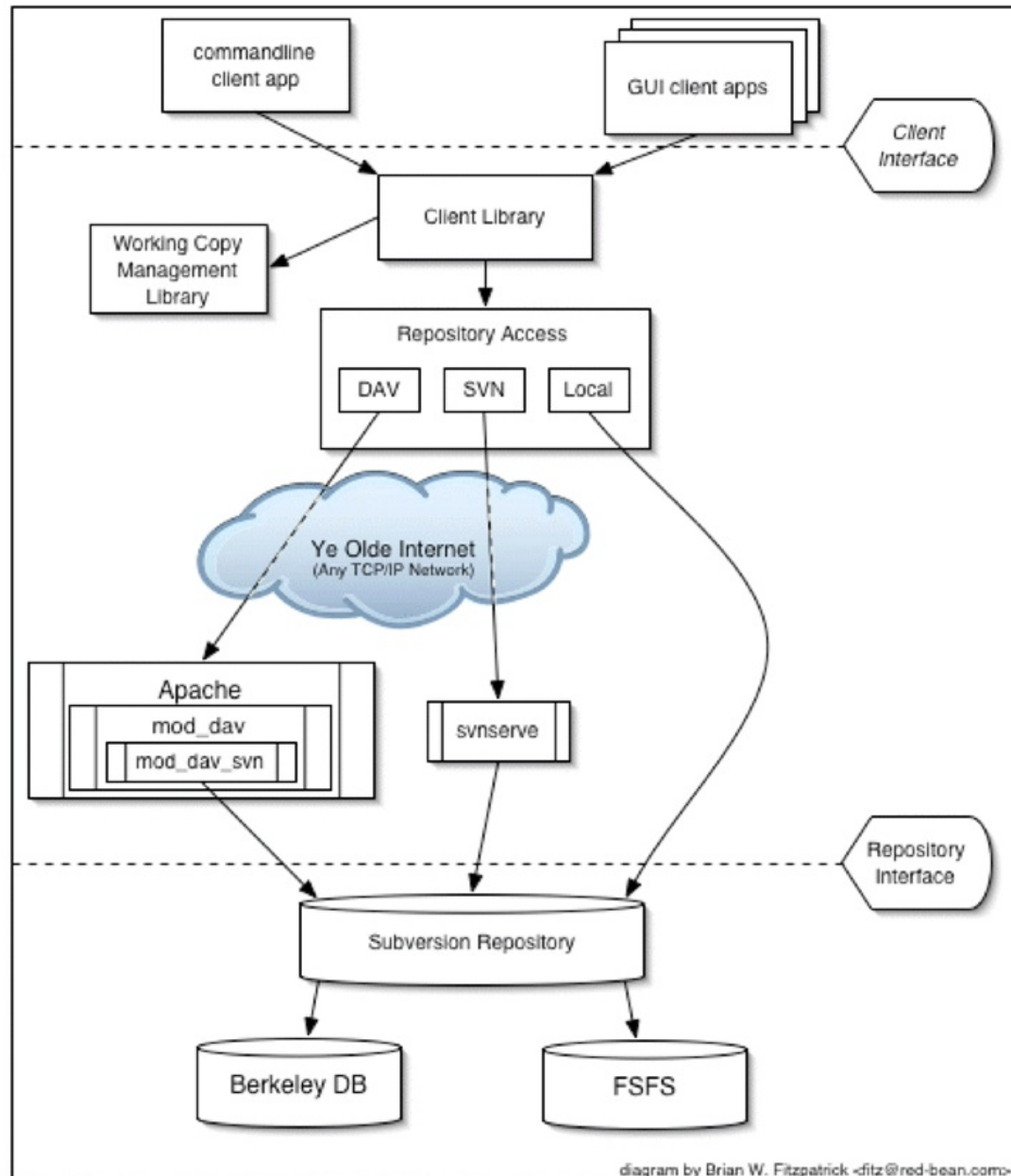
Subversion Demo

Things I'll show...

- Check-out a working copy

- Making a commit

- Updating (syncing) your working copy

- Resolving a conflict with a merge

Subversion Architecture

commandline
client app

GUI client apps

Client
Interface

Client Library

Working Copy
Management
Library

Repository Access

DAV | SVN | Local

Ye Olde Internet
(Any TCP/IP Network)

Apache
mod_dav
mod_dav_svn

svnserve

Repository
Interface

Subversion Repository

Berkeley DB

FSFS

diagram by Brian W. Fitzpatrick <fitz@red-bean.com>

Please see author attribution in Appendix

Installing Subversion

The Red-Bean book tells you how to install and configure it, so I'll let you figure that all out.

Resources

- Subversion – Installers, FAQs, Listserve, etc.
  - http://subversion.tigris.org

- Version Control with Subversion – Book
  - http://svnbook.red-bean.com

- Me – Peter J. Farrell
  - pjf@maestropublishing.com
  - http://blog.maestropublishing.com

Appendix - Copyright