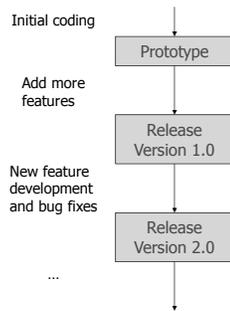# CS520 Web Programming
Version Control with Subversion

Chengyu Sun
California State University, Los Angeles
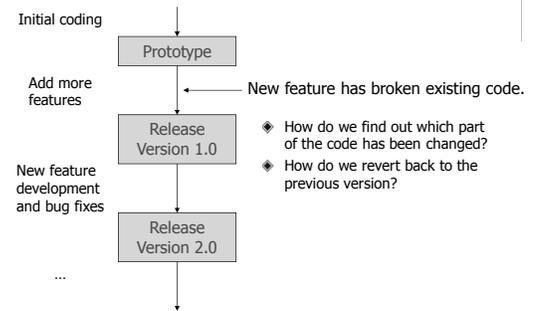

Overview

- ◆ Version control systems
- ◆ Basic concepts
  - Repository and working copies
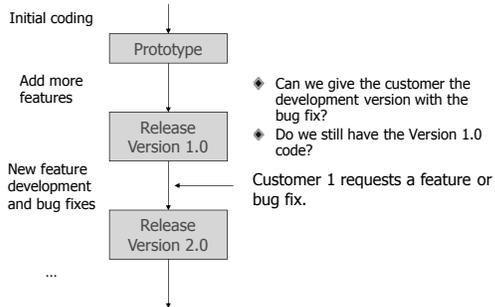 - Tag, branch, and merge
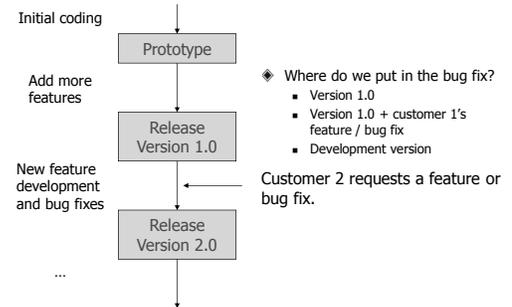- ◆ Using Subversion
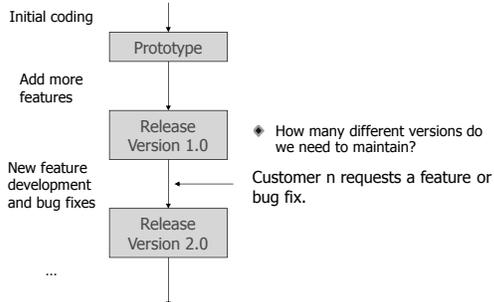

The Process of Application Development

Initial coding

Prototype

Add more features

Release Version 1.0

New feature development and bug fixes

Release Version 2.0

...

Problems During Development

Initial coding

Prototype

Add more features

New feature has broken existing code.

Release Version 1.0

- ◆ How do we find out which part of the code has been changed?
- ◆ How do we revert back to the previous version?

New feature development and bug fixes

Release Version 2.0

...

Problems During Development

Initial coding

Prototype

Add more features

Release Version 1.0

- ◆ Can we give the customer the development version with the bug fix?
- ◆ Do we still have the Version 1.0 code?

Customer 1 requests a feature or bug fix.

New feature development and bug fixes

Release Version 2.0

...

Problems During Development

Initial coding

Prototype

Add more features

Release Version 1.0

- ◆ Where do we put in the bug fix?
  - Version 1.0
  - Version 1.0 + customer 1's feature / bug fix
  - Development version

Customer 2 requests a feature or bug fix.

New feature development and bug fixes

Release Version 2.0

...

Problems During Development

Initial coding

Add more features

New feature development and bug fixes

...

Prototype

Release Version 1.0

Release Version 2.0

◈ How many different versions do we need to maintain?

Customer n requests a feature or bug fix.

Problems During Development

Initial coding

Add more features

New feature development and bug fixes

...

Prototype

Release Version 1.0

Release Version 2.0

Release Version 1.1

◈ How do we collect all the bug fixes into one release?

Put all changes into an intermediate release.

Version Control Systems

- ◈ **CVS**
  - Most popular / well known / widely used open source version control system
  - Somewhat obsolete due to some inherent system limitations
- ◈ **Subversion**, Arch, Monotone, git
- ◈ Commercial
  - Visual SourceSafe
  - ClearCase
  - BitKeeper

Subversion

- ◈ Pros: A better CVS
  - Fixed many annoying aspects of CVS
    - ◆ Recursive add, binary file handling, keyword substitution, local diff, status output etc.
  - Significant improvements
    - ◆ Atomic commit, constant time branching and tagging, better structure design etc.
  - *Feels like CVS*
- ◈ Cons: Just a better CVS
  - Does not scale to large, distributed development environments
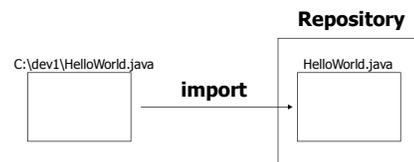

Common Command Syntax

```
svn <command> [src_dir] [dest_dir]
```
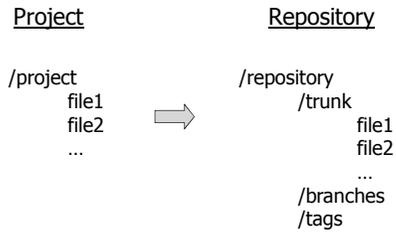
Could be local directories or URLs.

Examples:

svn ls file:///home/cysun/subversion/cs520

svn log http://cs3.calstatela.edu/cs520

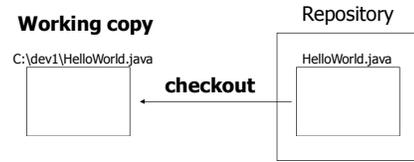svn checkout svn://cs3.calstatela.edu/cs520/csns/trunk csns

---

Import

Repository

C:\dev1\HelloWorld.java

import

HelloWorld.java

◈ Put a project into a repository

Directory Structure

Project   Repository

```
/project              /repository
      file1                 /trunk
      file2                        file1
      ...                          file2
                                   ...
                          /branches
                          /tags
```

Checkout

Working copy   Repository
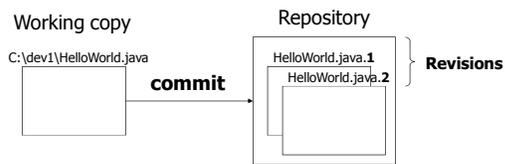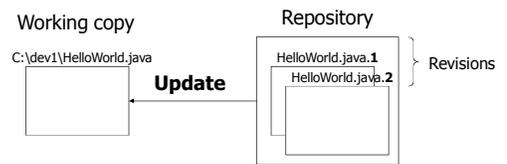
C:\dev1\HelloWorld.java   HelloWorld.java

checkout

◆ Get a copy of the project from the repository
- Working copy is *version controlled*
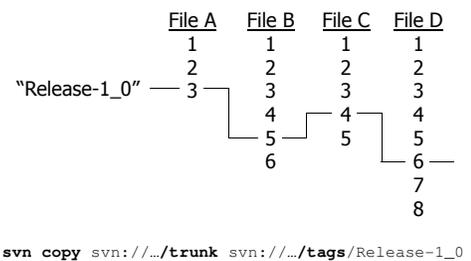
Commit (Checkin)

Working copy   Repository

C:\dev1\HelloWorld.java   HelloWorld.java**1**
HelloWorld.java**2**   **Revisions**

commit

◆ Send changes back to the repository

Update

Working copy   Repository

C:\dev1\HelloWorld.java   HelloWorld.java**1**
HelloWorld.java**2**   Revisions

Update

◆ Pull changes out of the repository and apply them to the working copy

Basic Version Control Operations

Developer   Repository

import   Fresh copy →

checkout   Version controlled copy

Commit (checkin)   changes →

Update   changes ←

Tag – Mark A Moment in Time

File A	File B	File C	File D
1	1	1	1
2	2	2	2
3	3	3	3
	4	4	4
	5	5	5
	6		6
			7
			8

"Release-1_0" — 3

svn copy svn://…/**trunk** svn://…/**tags**/Release-1_0

Branch – Work in Parallel

HelloWorld.java
```
1
2
3 (version 1.0)  ────→  3.1
4                        3.2
5                        ...
...
       │                  │
       ↓                  ↓
   Toward              Toward
  2.0 release         1.1 release
```
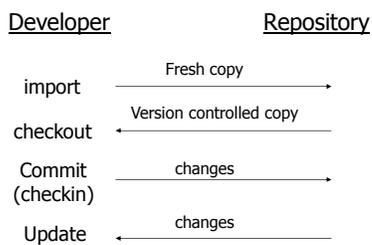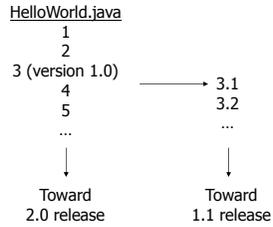
`svn copy` svn://…/`trunk` svn://…/`branch`/R1_1-branch

Merge

◆ Copy changes between different branches

HelloWorld.java
```
1
2
3 (version 1.0) ────→ 3.1
4                      3.2
5                      3.3
6                      3.4
...                    ...
 │  ←──── 3.10 (version 1.1)
 ↓
```

`svn merge` src_url dest_url

Toward
2.0 release
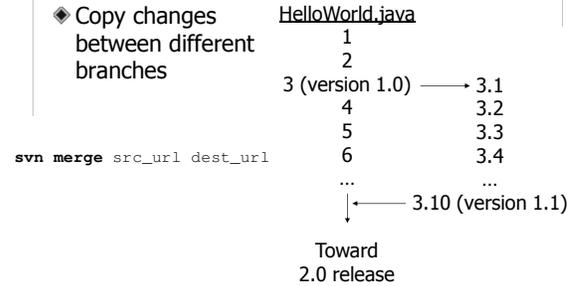
Other Useful Commands

◆ Status   ◆ Ls
◆ Revert   ◆ Log
◆ Add
◆ Remove

Online Resources

◆ The Subversion book - http://svnbook.red-bean.com/
◆ Import and merge with Subclipse - http://csns.calstatela.edu/wiki/content/cysun/course_materials/subversion/