



Revision Control Systems

Alan Kyffin
STFC Rutherford Appleton Laboratory
7th Oct 2014

Revision Control Systems

- Repository (a collection of files)
- History of the **changes** to those files

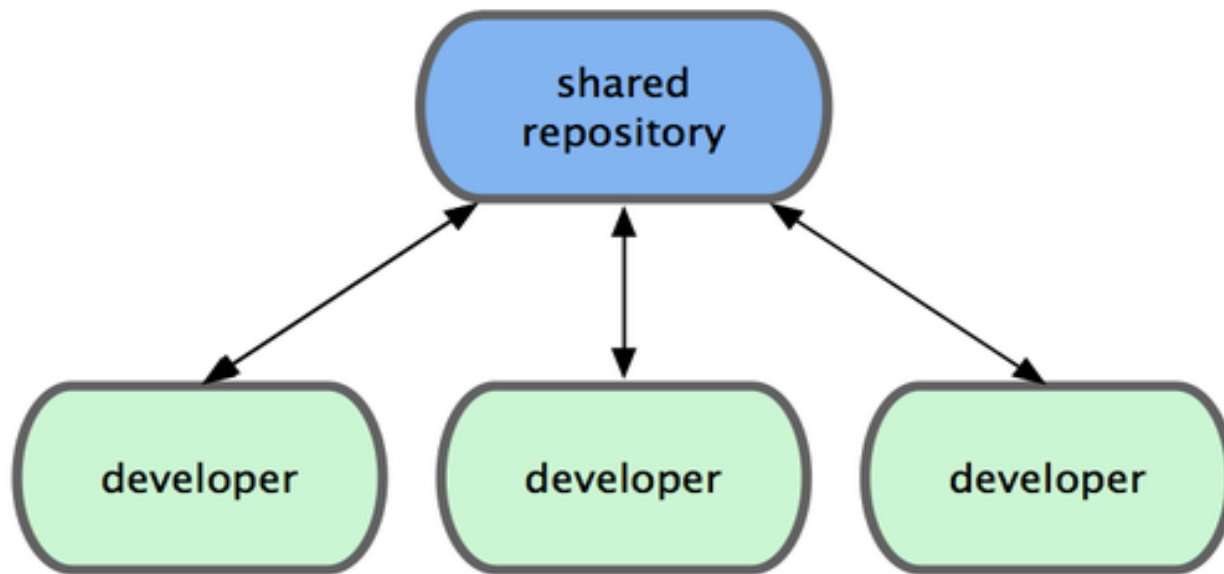


Advantages of RCS

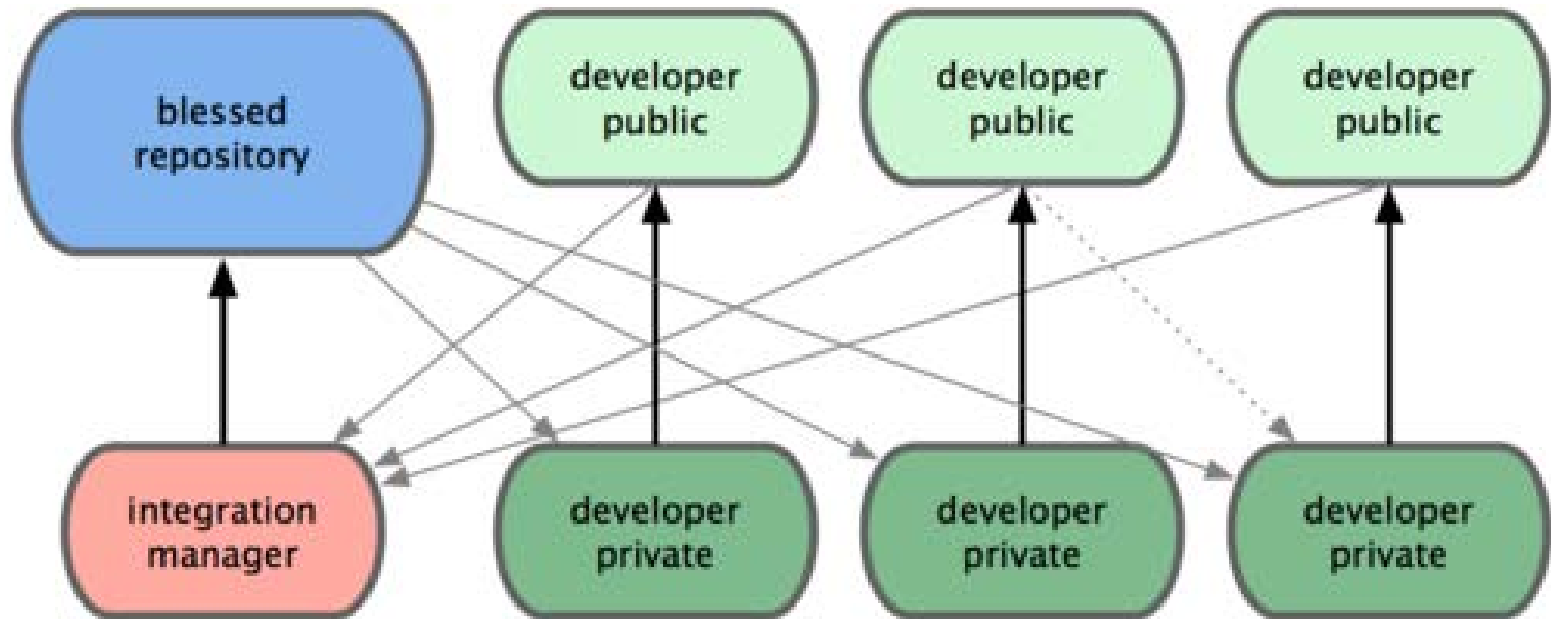
- Collaboration
 - Synchronise changes across a team
 - Multiple users can make changes without fear of overwriting others' work
- Can undo changes
- Can view old versions
- Can see when, by whom, and why a change was made
- Branches



Centralised / Distributed



Centralised / Distributed



(Too) Many RCSs

Centralised:

- CVS
- Subversion
- Perforce
- ...

Distributed:

- BitKeeper
- Git
- Mercurial (hg)
- GNU Bazaar
- ...



Subversion & Git

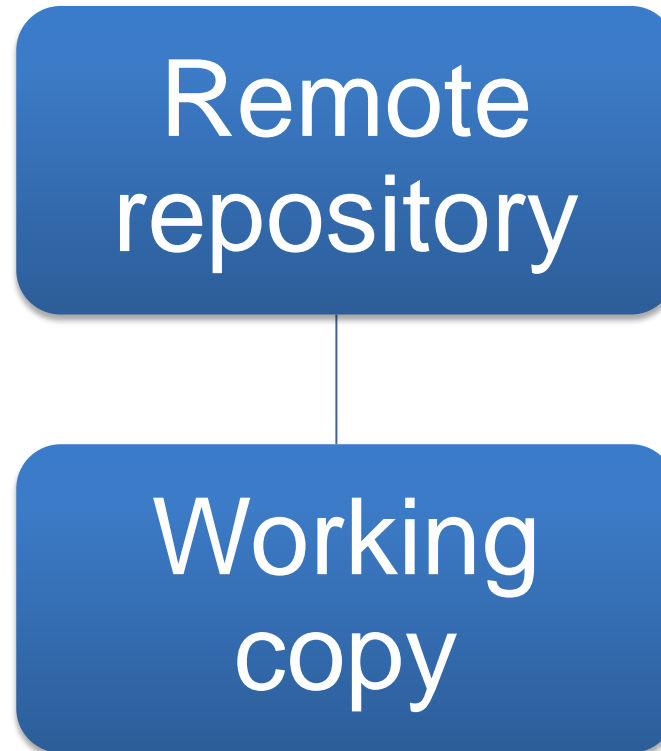


References

- Reference books:
<http://svnbook.red-bean.com>
<http://git-scm.com/book>
- Command help
`svn help <command>`
`git help <command>`



Hierarchy – Subversion



Hierarchy – Git

Remote repository

Local repository

(*index*)

Working copy



Downloading a repository

- Downloading a remote repository:
 `svn checkout <uri>`
 `git clone <uri>`
- Downloading changes from a remote repository:
 `svn update`
 `git pull`



Basic Commands

- Add a file to the repository. For Git, you need to use the 'add' command to tell it about **changes** to files.

```
svn add <file>
```

```
git add <file>
```

- Delete a file:

```
svn delete <file>
```

```
git rm <file>
```



Basic Commands

- These commands are necessary for Subversion to log the history of the file. Git can work out the history.
- To move / rename a file:
`svn move <src> <dest>`
`git mv <src> <dest>`
- To copy a file:
`svn copy <src> <dest>`



Working copy status

- Ask about changes made in your working copy.

svn status

git status



Committing

svn commit

git commit

- You will be asked to write a short ‘commit message’ describing the changes you’ve made.
- For Git, you will probably want to upload changes to the remote repository as well:

git push



Viewing commit history

```
svn log <file>
```

```
git log <file>
```



Undoing

- To undo ***un***committed changes:

```
svn revert <file>
```

```
git reset <file>
```

- To undo committed changes (not the '-' before the revision number):

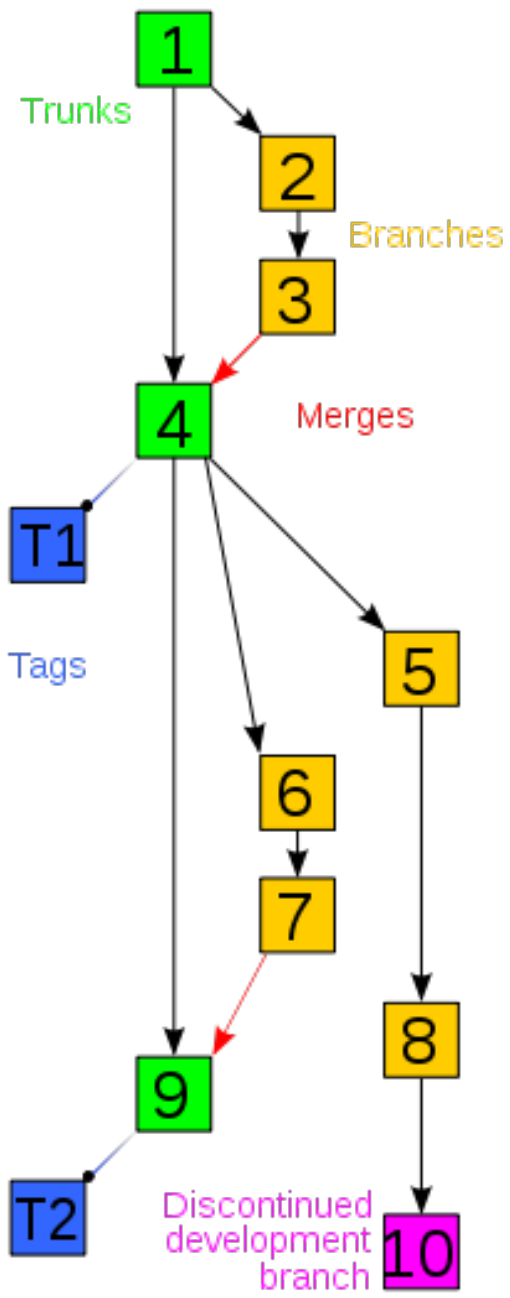
```
svn merge -c -<revision-number> <path>
```

```
git revert <commit-id>
```



Branching & Merging





Branches

- Branching allows the code to diverge.
- Development branches: experimental changes, usually for a specific feature, which will later be merged into the 'trunk'.
- Release branches: allow changes to be made to the 'trunk' without changing the stable release code.



Branching – Subversion

- Layout of a Subversion repository:

```
/
|
|– trunk
|
|– branches
|   |– ...
|
|– tags
|   |– ...
```



Branching – Subversion

- To branch, make a copy of trunk in the branches directory:

```
svn copy trunk branches/<branch>
```



Branching - Git

- To create a new branch:

```
git branch <branch-name>
```

- To view existing branches:

```
git branch          # local branches
```

```
git branch -r      # remote branches
```

- To change to an existing branch:

```
git checkout <branch-name>
```



Merging

- A merge combines the **changes** made on a branch into another branch.

```
svn merge <branch-2> <branch-1>
```

```
git checkout <branch-1>  
git merge <branch-2>
```



Conflicts



Science & Technology
Facilities Council

Conflicts

- Conflicts occur when two changes are merged that alter the same line of a file.
- You have 3 options when a conflict occurs
 - Keep your version
 - Keep the other version
 - Edit the file to create a new version
- Conflicts happen! But they are easily manageable if you keep your commits small and merge frequently.



Top tips

- Work on a specific feature or bug in a development branch – avoid committing directly to trunk.
- Merge trunk into development branches regularly.
- Make many, small commits.
- Make sub-branches to break-up the work of larger features.



Tags

- Tags are used to permanently reference a version of the repository – usually a release.
- To create a tag:
`svn copy trunk tags/<tag-name>`
`git tag <tag-name>`



Contact Details

Alan Kyffin
STFC Rutherford Appleton Laboratory
Chilton
DIDCOT
OX11 0QX

Mail: alan.kyffin@stfc.ac.uk
Tel: 01235 445953



Science & Technology
Facilities Council