
Intro to SCM with **CVS** & **SVN**

A Presentation for **CINLUG** by **David Brown**

David has 15 years of systems development experience with EDS, and has been writing Linux-based real-time data acquisition systems for Rolls-Royce in Indianapolis since 1999. He has been a member of the CINLUG Board of Directors for the past 2 years.

The format of this presentation is more of a demo of CVS and Subversion than a discussion of the philosophy and history of open source revision control systems.

Convention used: **CVS** commands are in **BLUE** and **SVN** commands are in **RED**

What is a revision control system?

- When developing software, programmers often use a revision control system to assist them with source code management.
- These systems are tools for recording, tracking and manipulating changes made to source code.
- Originally there was RCS (The GNU Revision Control System) for free software and open source development.
- CVS (The Concurrent Versions System) was later created to address some of the major drawbacks of RCS.
- SVN (Subversion) was recently created to address some of the major drawbacks of CVS.

Differences of RCS, CVS, & SVN

- RCS is file-centric, used the **lock-modify-unlock** development style, and was not network-aware.
- CVS initially depended on RCS and used the same format as RCS for storing history data, but CVS was directory-aware*, introduced the **copy-modify-merge** development style, and evolved to become network-aware**.
- SVN was designed as a replacement for CVS which kept the same basic development model and features, but fixed the obvious flaws: real directory tree versions*, true version history (move/rename), choice of network layers**, consistent data handling, efficient branching, all or none (atomic) commits.

Install/Initialize **CVS** & **SVN**

- apt-get install gcc make cvs
(or yum install)
- groupadd cvs
- mkdir /usr/local/cvs
- chgrp cvs /usr/local/cvs
- chmod 2775 /usr/local/cvs
- usermod -G cvs <user>
- cvs -d /usr/local/cvs init
- ls -al /usr/local/cvs/CVSROOT
- apt-get install gcc make svn
(or yum install)
- groupadd svn
- mkdir /usr/local/svn
- chgrp svn /usr/local/svn
- chmod 2775 /usr/local/svn
- usermod -G svn <user>
- svnadmin create /usr/local/svn
- ls -al /usr/local/svn

Importing Source Into **CVS** & **SVN**

- `cd /home/<user>/cvs/xmelt`
- `export CVSROOT=/usr/local/cvs`
- `cvs import -m "init import" xmelt xmelt start`
`<proj> <vendortag> <releasetag>`
- `ls -al /usr/local/cvs/xmelt`
- `cd /home/<user>/cvs`
- `rm -rf xmelt`
- `cvs rls`
- `cvs rls xmelt`
- `cd /home/<user>/svn`
- `svn import xmelt -m "init import" file:///usr/local/svn/xmelt`
- `cd /home/<user>/svn`
- `rm -rf xmelt`
- `svn list file:///usr/local/svn`
- `svn list file:///usr/local/svn/xmelt/trunk`

Check Out Code With **CVS** & **SVN**

- `cd /home/<user>/cvs`
- `cvs co xmelt`
- `rm -rf xmelt` (then through ssh tunnel)
- `export CVSROOT=/usr/local/cvs`
- `cvs -d :ext:<user>@localhost:/usr/local/cvs rls`
- `cvs -d :ext:<user>@localhost:/usr/local/cvs co xmelt`
- `cd /home/<user>/cvs/xmelt`
- `make` (notice warnings)
- `vi *.c` and `ADD <stdlib.h>`
- `make` (notice NO warnings)
- `cd /home/<user>/svn`
- `svn co file:///usr/local/svn/xmelt`
- `rm -rf xmelt` (then through ssh tunnel)
- `svn list svn+ssh://localhost/usr/local/svn`
- `svn co svn+ssh://localhost/usr/local/svn/xmelt`
- `cd /home/<user>/svn/xmelt/trunk`
- `make` (notice warnings)
- `vi *.c` and `ADD <stdlib.h>`
- `make` (notice NO warnings)

Make/Review Changes **CVS** & **SVN**

- `cv`s add README.txt JUNK.txt
- `cv`s remove JUNK.txt
- `rm` JUNK.txt; `cv`s remove JUNK.txt
- `cv`s -n update
- `cv`s diff
- `cv`s diff xmelt.c
- `rm` Makefile
- `cv`s update
- `cv`s status; `cv`s status xmelt.c
- `cv`s log; `cv`s log xmelt.c
- `sv`n add README.txt JUNK.txt
- `sv`n delete JUNK.txt
- `sv`n delete --force JUNK.txt
- `sv`n status
- `sv`n diff
- `sv`n diff xmelt.c
- `rm` Makefile
- `sv`n update
- `sv`n info *; `sv`n info xmelt.c
- `sv`n log; `sv`n log xmelt.c

Check In Code with **CVS** & **SVN**

- `cv`s co xmelt (somewhere else)
- Add the following lines to *.c
 - `#ifndef lint`
 - `static char *rcsid = "$Id:$";`
 - `#endif`
- `cv`s -n update
- `cv`s ci -m "add ident"
- `cv`s -n update (original tree notice conflicts)
- `cv`s update; `cv`s ci -m "fix bugs" (merge)
- `make`; `ident` xmelt
- `sv`n co file:///usr/local/svn/xmelt
- Add the following lines to *.c
 - `#ifndef lint`
 - `static char *rcsid = "$Id:$";`
 - `#endif`
- `sv`n propset `sv`n:keywords "Id" *.c
- `sv`n status
- `sv`n ci -m "add ident"
- `sv`n status (original tree notice conflicts)
- `sv`n update; `sv`n ci -m "fix bugs" (merge)
- `make`; `ident` xmelt

Change/Revert/Move **CVS** & **SVN**

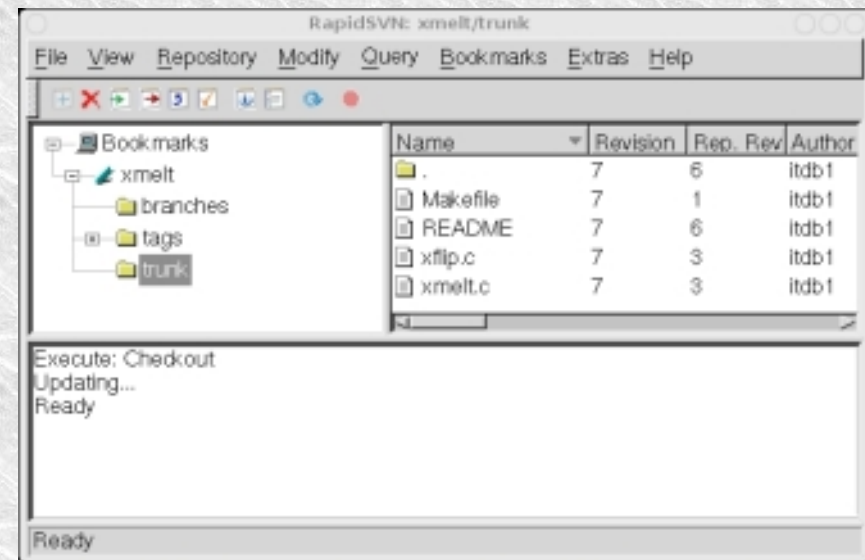
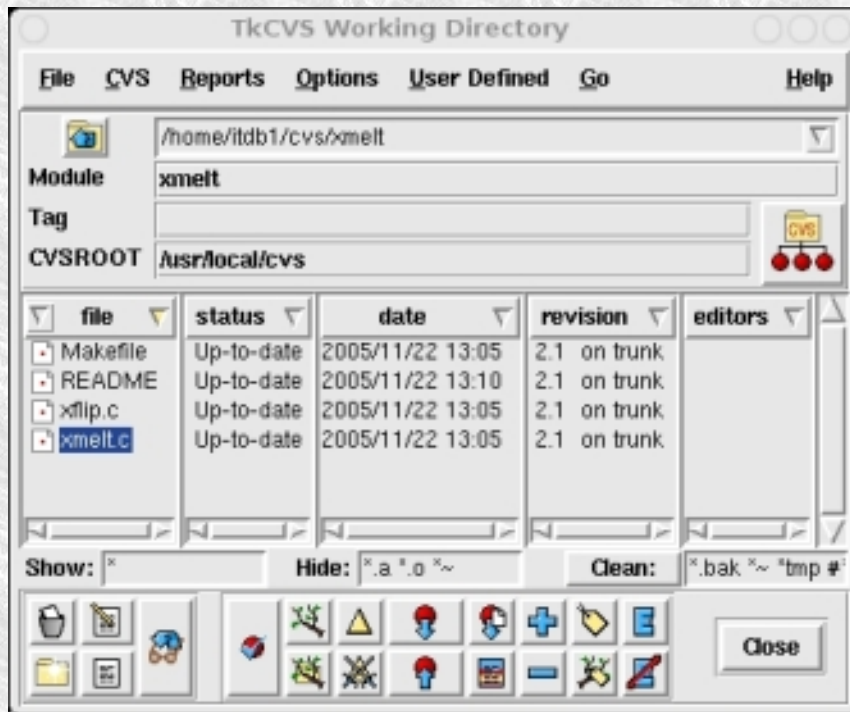
- edit README.txt
- rm README.txt; cvs update
- edit README.txt again
- cvs ci -r 2 -m "new" (force revision 2)
- cvs log README.txt
- cvs diff -r 2.1 -r 1.1 README.txt
- cvs update -j 2.1 -j 1.1 README.txt
- cvs ci -m "back to old"
- mv README.txt README
- cvs remove README.txt; cvs add README; cvs ci; cvs log README
- edit README.txt
- svn revert README.txt
- edit README.txt again
- svn ci -m "new" (can't force revision)
- svn log README.txt
- svn diff -r 4:3 README.txt
- svn merge -r 4:3 README.txt
- svn ci -m "back to old"
- svn move README.txt README
- svn ci; svn log README

Tagging With **CVS** & **SVN**

- `cvstag -c revision-1`
- `vi README` again and commit
- `cvscob -r revision-1 xmelt`
(notice this version doesn't have change)
- For more detail on tagging, branching, and merging see:
<http://cvsbook.red-bean.com/>
- `svn copy file:///usr/local/svn/xmelt/trunk \`
`file:///usr/local/svn/xmelt/tags/release-1.0 \`
`-m "tagging the 1.0 release"`
- `svn update`
- `ls -al /home/<user>/svn/xmelt/tags`
(notice that tagging is just copying)
- For more detail on tagging, branching, and merging (as well as locking/unlocking) see:
<http://svnbook.red-bean.com/>

GUI Applications for CVS & SVN

- apt-get install tkcvs (or yum install)
- cd /home/<user>/cvs/xmelt; tkcvs
- apt-get install rapidsvn (or yum install)
- cd /home/<user>/svn; rapidsvn;
- Repository->Checkout



Other Tools for **CVS** & **SVN**

- **CVS** <http://www.nongnu.org/cvs>
- *cvs2svn* (conversion tool to convert existing CVS repositories to Subversion)
<http://cvs2svn.tigris.org>
- *viewcvs* (software to serve CVS source code anonymously via HTTP)
<http://viewcvs.sourceforge.net>
- *cvsweb* (another CGI-based HTTP server)
<http://cvsweb.freebsd.org>
- *gcvs* (another GUI tool like tkcvs)
- Subversion <http://subversion.tigris.org>
- *websvn* (software to server SVN source code anonymously via HTTP)
<http://websvn.tigris.org>
- *esvn* (another GUI tool like rapidsvn)