Introduction to version control

David Rey – DREAM



Overview

- Collaborative work and version control
- Main CVS user commands
- Advanced use of CVS



Overview

- Collaborative work and version control
- Main CVS user commands
- Advanced use of CVS



Collaborative work and version control: examples

- Development
 - Source files: C, C++, java, Fortran, Tcl, Python, shell scripts, ...
 - Build/config files: Makefile, ant, ...
- Text documents/articles/bibliographies
 - Plain text
 - Latex/bibtex
- Web pages
 - Html
 - Php, javascripts, ...
- XML documents
-

® ASCII documents



A software development process at INRIA

- INRIA recommandations about software development:
 - <u>http://devel.inria.fr/recom/</u>
- In particuliar, concerning version control:
 - http://devel.inria.fr/recom/processus/main005.html#toc8



Version control: main ideas

- Distributed documents/collaborative work
 - Automatic merging
 - Alarms on conflicts
 - Easy communication between users (log messages, emails, ...)
- Version control: incremental versions
 - All previous versions available
 - Minimal necessary disk space (incremental)
 - History of changes/logs



Version control software

- CVS: <u>http://www.cvshome.org</u>
- Arch: <u>http://www.gnu.org/software/gnu-arch/</u>
- Subversion: <u>http://subversion.tigris.org/</u>
- Forges that use version control software: (<u>http://devel.inria.fr/recom/processus/main005.html#developpement_forges</u>)
 - GForge : <u>www.gforge.org</u>
 - Savannah : <u>savannah.gnu.org</u>
 - LibreSource : <u>libresource.inria.fr</u>
 - Visual Source Safe (pour Windows) : <u>msdn.microsoft.com/vstudio/previous/ssafe</u>



CVS: Concurrent Versions System

- Widely used on a lot of different platforms (Linux, Windows, ...)
- Open source license
- For local and/or distant usage
- Recommended for INRIA software developments
- Several possible clients
 - Command line
 - GUI: tkCVS, jCVS, WebCvs, Jalindi igloo, WinCVS, TortoiseCVS



What CVS is for ?

- Several users work together at the same time on the same files (concurrency)
- Version control
 - Tags
 - Version comparisons
 - Multiple lines of development in the same base
 - Branching
 - Support for binary files
 - Event control (e.g., notification)



What CVS is not for ?

- Backup
- Bug tracking
- Source documentation
- Dependencies
- ...



Overview

- Collaborative work and version control
- Main CVS user commands
- Advanced use of CVS



CVS: client/server architecture

• 2 possibilities:





CVS hierarchy

- Repository root or "CVSROOT"
 - Looks like "cvs-sop.inria.fr:/CVS/SmartTools"
 - set the CVSROOT environment variable on UNIX systems
 - configure an item in the setup menus of a GUI client
 - Use the command line: cvs –d CVSROOT
- Modules
 - For example SmartTools uses 3 modules: SmartTools, SmartToolsAddons, Documents
- Directories
 - Each module contains an arbitrary hierarchy of directories
- Files (incremental diff versions are stored)



CVS hierarchy : an example



CVS help

- cvs --help-commands (list available commands)
- cvs --help-options (list general options)
- cvs H import (specific help for a command and its specific options, e.g. "import")
- \rightarrow CVS general options + specific command options are different
 - cvs [general options] <command> [specific options]



Session Edit View Bookmarks Settings Help

C€

| mythos_dovol | oper \$ ave | help_commands |
|---------------------|---------------|--|
| CVS commands | are | merp-commands |
| cio commanus add | , are. 664 | a new file/directory to the renository |
| auu admi | n Admi | a new file/uffectory to the repository |
| aumi | ni Rumi | w last revision where each line was modified |
| anno | verte snor | w last revision where each line was moullied |
| cnet | Koul Chec | kout sources for ealting |
| C O M M | nt cned | k files into the repository |
| alII | . Snow | v allierences detween revisions |
| ealt | Get Get | ready to ealt a watched file |
| edit | fors See | who is editing a watched file |
| expo | ort Expo | ort sources from CVS, similar to checkout |
| hist | ory Show | v repository access history |
| impo | ort Impo | ort sources into CVS, using vendor branches |
| init | : Crea | ate a CVS repository if it doesn't exist |
| kser | ver Kert | peros server mode |
| log | Prin | nt out history information for files |
| logi | n Prom | <pre>npt for password for authenticating server</pre> |
| logo | out Remo | oves entry in .cvspass for remote repository |
| pser | rver Pass | word server mode |
| rann | iotate Show | v last revision where each line of module was modified |
| rdif | ff Crea | ite 'patch' format diffs between releases |
| rele | ease Indi | cate that a Module is no longer in use |
| remo | ove Remo | ove an entry from the repository |
| rlog | y Prir | it out history information for a module |
| rtag | g Add | a symbolic tag to a module |
| serv | ver Serv | ver mode |
| stat | us Disp | play status information on checked out files |
| tag | Add | a symbolic tag to checked out version of files |
| uned | lit Undo | o an edit command |
| upda | ite Brin | ng work tree in sync with repository |
| vers | sion Show | current CVS version(s) |
| wato | h Set | watches |
| wato | hers See | who is watching a file |
| (Specify the | ehelp optic | m for a list of other help options) |
| | | |

٣

KINRIA

ios-developer \$ []

CVS --help-options

Session Edit View Bookmarks Settings Help

| 💽 🗖 | | |
|-------------------|--|---|
| mythos-developer | \$ cvshelp-options | ٨ |
| CVS global option | ns (specified before the command name) are: | |
| H | Displays usage information for command. | |
| -Q | Cause CVS to be really quiet. | |
| -q | Cause CVS to be somewhat quiet. | |
| -r | Make checked-out files read-only. | |
| -w | Make checked-out files read-write (default). | |
| -n | Do not execute anything that will change the disk. | |
| -t | Show trace of program execution try with -n. | |
| -v | CVS version and copyright. | |
| -T tmpdir | Use 'tmpdir' for temporary files. | |
| -e editor | Use 'editor' for editing log information. | |
| -d CVS_root | Overrides \$CVSROOT as the root of the CVS tree. | |
| -f | Do not use the ~/.cvsrc file. | |
| -z # | Use compression level '#' for net traffic. | 2 |
| -x | Encrypt all net traffic. | |
| -a | Authenticate all net traffic. | |
| -s VAR=VAL | Set CVS user variable. | |
| (Specify theh) | elp option for a list of other help options) | |
| mythos-developer | | |
| | | |



CVS -H import

Session Edit View Bookmarks Settings Help



Import

- Used to synchronize a local tree as a whole with the server
- Most often used to initialize the server tree
- Can also be used after this initial step at any time (to avoid one-byone files addition)
- cd local_directory
- cvs import -m "message" directory_on_base branch_name version_name



Checkout and update

- cvs -d cvsroot checkout module_name
- cvs update -d directories/files
 - U: updated
 - P: the version on the server is newer and the local copy has been updated
 - M: modified (or merged), the local file is different from the server file and merge was done in the local copy
 - ?: unknown, this file is present locally but not in the CVS base
 - A: added, this file is present locally and has been added (before commit)
 - R: removed, this file is not present locally and has been removed (before commit)
 - C: conflict, the local file has been marked with conflict markers which identify the different conflict locations



Session Edit View Bookmarks Settings Help

| mythos-test \$ cvs -d :ext:drey@cvs-sop.inria.fr:/CVS/SmartTools checkout Documents |
|---|
| cvs checkout: Updating Documents |
| U Documents/CSSinSmartTools.html |
| U Documents/Composants.html |
| U Documents/Composants.txt |
| U Documents/DidierMemoST |
| U Documents/Event.txt |
| U Documents/MissionDream2004.txt |
| U Documents/SmartTools.ppt |
| U Documents/compilation.txt |
| U Documents/packages.html |
| U Documents/packages.txt |
| cvs checkout: Updating Documents/PawelKowalski |
| U Documents/PawelKowalski/changements.txt |
| U Documents/PawelKowalski/propositions.txt |
| U Documents/PawelKowalski/rapport.txt |
| U Documents/PawelKowalski/soapmon.txt |
| U Documents/PawelKowalski/tests.txt |
| U Documents/PawelKowalski/todos.txt |
| cvs checkout: Updating Documents/PawelKowalski/TransformTutorial |
| U Documents/PawelKowalski/TransformTutorial/TODOS.txt |
| U Documents/PawelKowalski/TransformTutorial/TransformTutorial.tcp |
| U Documents/PawelKowalski/TransformTutorial/TransformTutorial.tex |
| U Documents/PawelKowalski/TransformTutorial/biblio.tex |
| U Documents/PawelKowalski/TransformTutorial/createCoXSL.tex |
| U Documents/PawelKowalski/TransformTutorial/generate-component.vsd |
| U Documents/PawelKowalski/TransformTutorial/generate-wslayer.vsd |
| U Documents/PawelKowalski/TransformTutorial/generateWsLayer.tex |
| U Documents/PawelKowalski/TransformTutorial/graphDeDepart.dot |
| U Documents/PawelKowalski/TransformTutorial/intro.tex |
| U Documents/PawelKowalski/TransformTutorial/output-dir.dot |
| U Documents/PawelKowalski/TransformTutorial/overview.vsd |
| U Documents/PawelKowalski/TransformTutorial/ressource-dir.dot |
| U Documents/PawelKowalski/TransformTutorial/src-directory.dot |
| U Documents/PawelKowalski/TransformTutorial/subdivision.tex |
| U Documents/PawelKowalski/TransformTutorial/tutorial.tex |
| U Documents/PawelKowalski/TransformTutorial/tutorial.txt |



mythos-developer \$ cvs update -d .
? src/fnf.class
? src/ftpWwwZips.class
? src/stgen.class
? src/zipnf.class
cvs update: Updating .
M build.xml
A newfile
cvs update: warning: smarttools.policy was lost
U smarttools.policy
cvs update: Updating src
mythos-developer \$ []



Commit

- cvs commit -m "my message" directories/files
- Always use explicit messages !!!
- List of forbidden messages:
 - "ok" 🙂
 - "" → **?**
 - "modification of file toto.tex" \rightarrow which modification ?
 - "bug fixed" \rightarrow which bug ?
 - ...



Ideal Development (1/4)



Ideal Development (2/4)

development

Developer A











Ideal Development (3/4)



Ideal Development (4/4)



Real Development (1/5)



Real Development (2/5)



Real Development (3/5)



Real Development (4/5)

Conflict Resolution

Developer A



Developer B



base



Real Development (5/5)



Conflicts

- Do not panic!
 - Many problems are easy to fix
 - Generally, it does not require a lot of interaction
- If necessary, discuss with people to find a compromise
 - Meetings
 - Email
 - Phone
 - ...
- Usually can be avoided with regular updates



Status, log, diff, annotate

- cvs status filename
 - Gives information about local file with comparison to base file
- cvs log filename
 - Show all the previous versions number, comitter, date, and number of lines modified
- cvs diff filename
 - Show lines where there are differences between local and base file
- cvs annotate filename
 - Gives information line by line: version of introduction, who, when



Status examples

| Session Edit View Bookmar | ks Setting | s Help | |
|---|--|--|----------|
| | | | |
| <pre>mythos-developer \$ cvs :</pre> | status b | uild.xml | ^ |
| File: build.xml | Status: | Locally Modified | |
| Working revision: Repository revision: Sticky Tag: Sticky Date: Sticky Options: | 1.71 1.71 (none) (none) (none) | /CVS/SmartTools/SmartTools/ant/developer/build.xml.v | |
| mythos-developer \$ [| | | |
| | | | 4 |
| | | | |



Session Edit View Bookmarks Settings Help

G 🛛

wythos-dream-web-sources \$ cvs log seminaires.html.in | more RCS file: /CVS/dream/dream-web/seminaires.html.in.v Working file: seminaires.html.in head: 1.14 branch: locks: strict access list: symbolic names: version1: 1.1.1.1 dream-web: 1.1.1 keyword substitution: kv total revisions: 15; selected revisions: 15 description: revision 1.14 date: 2004/12/14 17:43:23; author: drey; state: Exp; lines: +1 -1 add a draft version of the cvs presentation for the seminar of the 15 dec 2004 revision 1.13 date: 2004/12/14 15:55:25; author: jmi; state: Exp; lines: +5 -0 Updates to add the refactoring seminaires and fixes to the text on the links page. revision 1.12 date: 2004/11/24 19:25:34; author: drey; state: Exp; lines: +6 -6 maj des evenements et des projets drema et ia-odl revision 1.11 date: 2004/11/16 14:08:44; author: dgeld; state: Exp; lines: +1 -1 --Encore--





÷ vthos-developer \$ cvs annotate build.xml | more Annotations for build.xml **** 1.1(drey 28-Jan-04): <?xml version="1.0" encoding="utf-8"?> (drev 1.1 _____ 1.1 (drev 28-Jan-04): 1.1(drev 28-Jan-04): Build file for SmartTools v4 Api - for use with the J akarta Ant java build tool (drev 28-Jan-04): 1.11.128-Jan-04): Build Instructions: (drev 1.1(drev 28-Jan-04): To build, run build.bat (win32) or build.sh (unix) optionally 1.1(drev 28-Jan-04): with a target arg as indicated below - in the directo ry where this 1.1 (drey file is located. 28-Jan-04): 1.41(drev 15-Mar-04): This build.xml uses ./build-properties.xml file. The batch/shell file sets up your classpath and calls 1.41(drev 15-Mar-04): java org.apache.tools.ant.launch.Launcher 1.128-Jan-04): (drev (dparigot 09-Apr-04): Basic target: 1.491.49(dparigot 09-Apr-04): user.create.cmp 1.128-Jan-04): Decoupage (drev 1.1(drey - Defines Global Variables 28-Jan-04): 1.1 (drev 28-Jan-04): - Configuration 1.1 (drey 28-Jan-04): - Clean, Init, Present - Target dependances 1.1(drev 28-Jan-04): 1.1 Code (drey 28-Jan-04): 1.1 - Web Services (drey 28-Jan-04): (drey 1.128-Jan-04): - Documentation (javadoc) 1.1- Zip et Distribution (demo) (drev 28-Jan-04): - User and Core components 1.1(drey 28-Jan-04):

Tagging

- Tag = string marker which is applied to the status of a part of the CVS hierarchy
- Often used to mark a given version of the cvs repository with a string which refers to a released version of the software
- Easily get back specific versions of the software
 - Use/distribute a given released version
 - Reproduce bugs for a given release version
- Only files are tagged
- cvs tag -r 1.2 tagname files
- cvs rtag tagname files (latest version in the base is tagged)



Add, delete, and move files

- cvs add filenames
- Produces a message that explains that a commit is necessary
- cvs commit m "addition of files bla..." directories/files
- Files have to be deleted locally: rm filenames
- Produces a message that explains that a commit is necessary
- cvs remove filenames
- cvs commit -m "removed files bla..." directories/files
- Trace of deleted files in Attic directories on the CVS base

 Move = add + delete, with an explicit message to keep a trace of the old names



Add, delete, and move folders

- As simple as files for addition; do not need a commit command to take effect
- Not possible to delete and/or move folders!
 - Except with "hard/dirty" intervention, cf. advanced use of CVS section



Summary of the main commands

- cvs [cvs-options] command [cmd-options] [directories/files] •
- cvs import import files from a local directory to the cvs base ۲

remove a file from the cvs base

- cvs checkout copy on a local disk a given version of the cvs base •
 - apply the modifications of the local copy to the cvs base cvs commit upgrade the local copy with a version of the cvs base
- cvs update ۲
- cvs add •

•

۲

- cvs remove ۲
- cvs status ۲
- cvs loa •
- cvs diff ۲
 - show information line by line of a version in the cvs base cvs annotate

show the status of a local file wrt the cvs base

add a file to the cvs base

cvs tag (or rtag) ۲

put a tag to identify a given version of the cvs base

show the different previous commit stages

show lines in local and server files that differ



Overview

- Collaborative work and version control
- Main CVS user commands
- Advanced use of CVS



CVS Administration

- Modules administration
- Usage definition
 - Events control (emails, ...)
 - Watch/edit



CVS base configuration

- CVSROOT Module (exists by default)
 - cvs checkout CVSROOT
- Configuration files
 - modules, modules definitions
 - cvswrapper binary files control
 - cvsignore list which files CVS has to ignore (*.o, ...)
 - commitinfo, editinfo, loginfo, notify, rcsinfo, taginfo make it possible to configure actions with respect to specific CVS operations



Binary files in CVS base

- No real version control (diffs don't work)
- Binaries are corrupted if treated as ASCII files
- Can be done with command line: cvs add -kb filename
- Can be forced for known extensions
 - Configuration file: cvswrapper



```
cvswrappers - Bloc-notes
                                                                                               Fichier Edition Format Affichage ?
₩ This file affects handling of files based on their names.
 The -m option specifies whether CVS attempts to merge files.
#
#
  The -k option specifies keyword expansion (e.g. -kb for binary).
  Format of wrapper file ($CVSROOT/CVSROOT/cvswrappers or .cvswrappers)
#
   wildcard
                 [option value][option value]...
#
   where option is one of
#
                 from cvs filter
                                          value: path to filter
   -f
   -t
                 to cvs filter
                                          value: path to filter
                 update methodology
expansion mode
                                          value: MERGE or COPY
   -m
   -k
                                          value: b. o. kkv. &c
   and value is a single-quote delimited value.
  For example:
  .gif -k
           'b
  jar
          'b
          'b
  DD
  doc
         h
  az
  bS
  100
  gif
 .png
 .tif
      -k
          'h
```



Recursion with CVS

- Check each command to see if it is recursive or not
 - Add is not recursive, update is recursive, ...
- Read the cvs manual (note we are in the « advanced section » of the expose)



Windows users

- Several CVS clients: webCVS, winCVS, ...
- Possible use of CVS with ssh under Windows via putty (Windows ssh client) \rightarrow contact me
- Problems with line ending ... always edit files on the platform where the checkout has been done!



Branching

- Branching is easy... merging is not!
- When ?
 - Before a release ("freeze" the code in a branch)
 - For parallel development that will not be integrated directly
 - For someone developing during a training period
 - For some "research code"
 - For a specific application (e.g., dedicated to a given client)









- To restructure a repository (e.g., move or delete folders)
- To check for possible problems



References and links

- CVS: <u>https://www.cvshome.org/</u>
 - <u>https://www.cvshome.org/docs/manual/</u>
 - <u>https://ccvs.cvshome.org/fom/fom.cgi</u>
- GUI:
 - <u>http://www.twobarleycorns.net/tkcvs.html</u>
 - <u>http://www.wincvs.org</u>
 - <u>http://www.tortoisecvs.org</u>
 - <u>http://www.jalindi.com/igloo</u>
 - http://www.jcvs.org
- CVS Configuration et mise en œuvre. Frédéric Lepied. O'Reilly Eds. (available at the documentation centre of INRIA Sophia: ref. O356)



Closing remarks & Questions

- Use version control systems from the start
 - When working alone or in a group
 - For development projects or scientific papers
 - When tag released versions
- Handy FAQ's on the CVS home site

