

# CVS - Tutorial

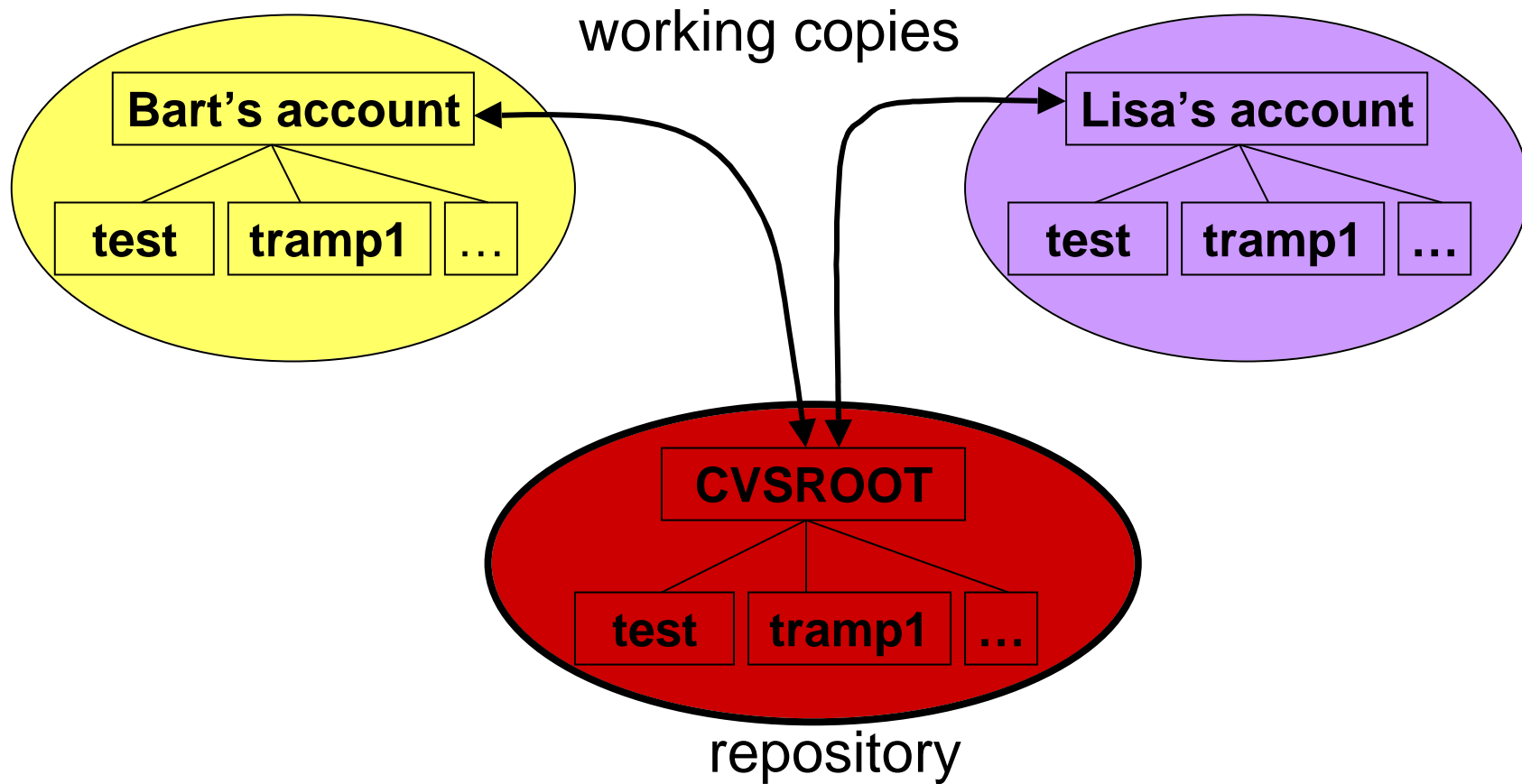
TRAMP – Praktikum

29.10.01

# why CVS

- concurrent version system
- work simultaneously on the same project, file, ...
- don't do the work twice
- be up to date
- share files and have your own working directory

# directory structures



# initial setup

- ***setenv CVS\_RSH ssh***  
***setenv CVSROOT :ext:\$USER@cvsbruegge:/cvs/tramp***

(to always initialize for CVS, place this line in the .tcshrc file in your home directory)

- ***mkdir <xxx>*** (e.g. ***~/CVStramp/***)
- ***cd <xxx>***

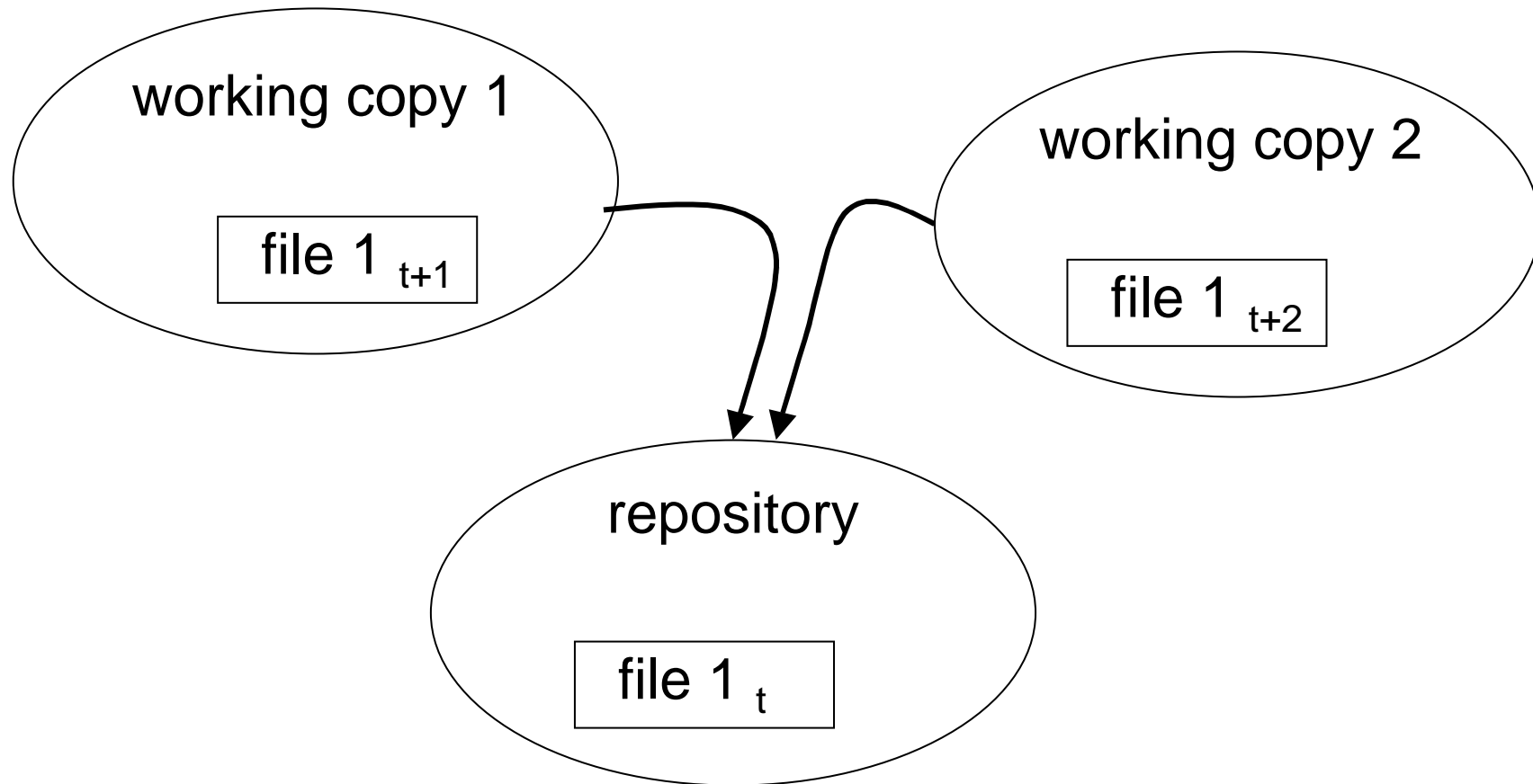
# CVS – ssh-keygen

- with the ssh-keygen there is no need to use the password for every CVS command
- generate pair of keys  
*ssh-keygen -t dsa*
- add your **public** key to the authorized key list  
*cat id\_dsa.pub >> ~/.ssh/authorized\_keys2*

# CVS basics

- get files (repository -> working copy)
  - *cv*s checkout <module> (“test” / “tramp1”)
  - *cv*s update -d (directories)
- submit files (working copy -> repository)
  - *cv*s commit (always add a comment)
  - *cv*s add <filename or directory>

# CVS – file merging

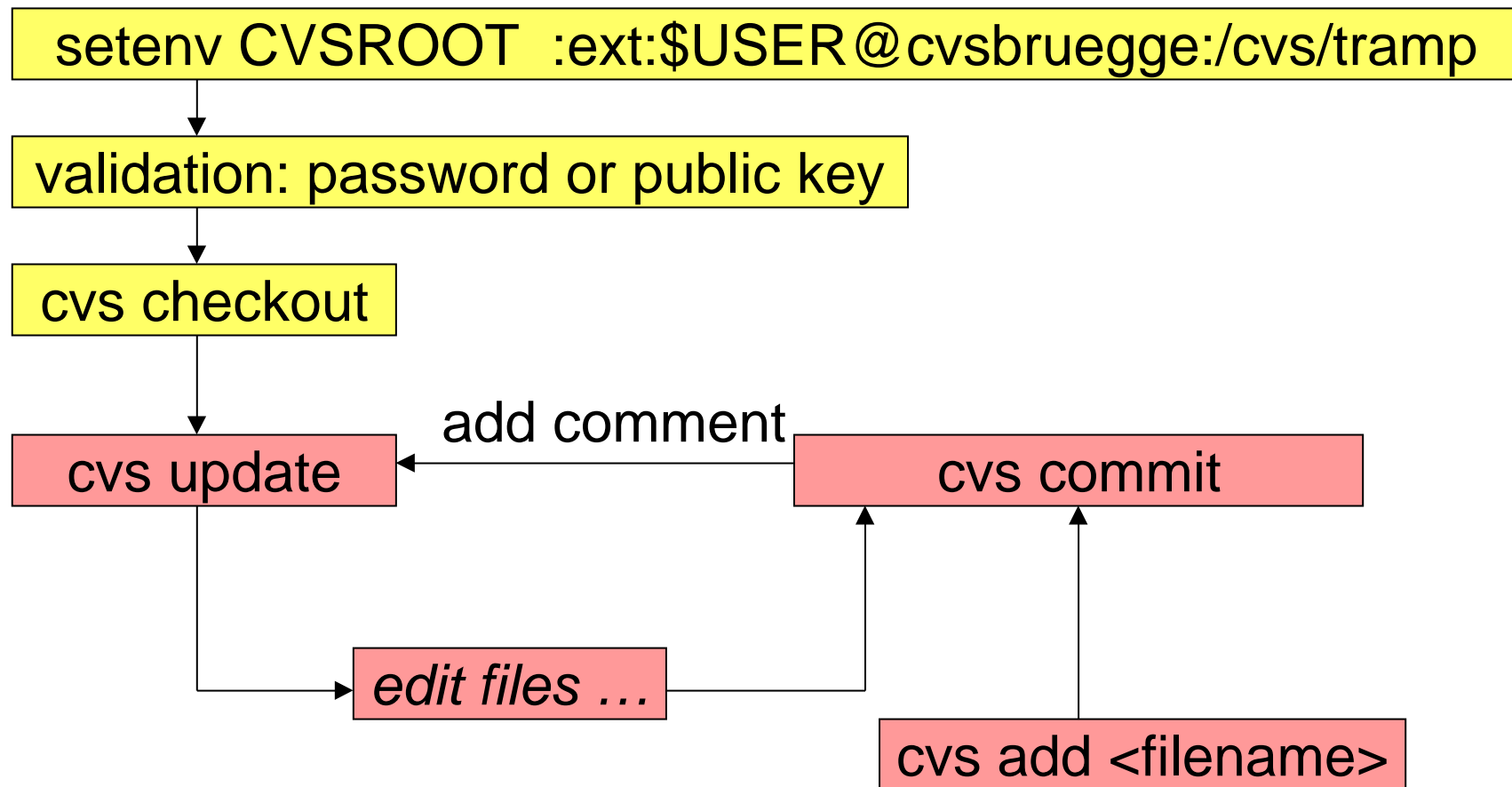


# CVS - log

- log message – *cv*s log <filename>
- CVS editor
  - on default vi
  - change editor (*setenv CVSEEDITOR emacs*)
- special strings in files before check in
  - *\$Id\$ / \$Revision\$ / \$Header\$*



# Development Flow



# CVS - Limitations

- depends on Line Separated Text Files
- files can not be moved
- directories can not be deleted

# CVS - Policies

- usually no binary
- ...think before check in
- ...think before check in
- only one version of each file
  - no check in of files that can be generated
- always add useful comments
  - (e.g. “added method getString ...”)