Version Control Systems
Example with svn

Sharing files can get messy ….

A team is working in the same project, makes developpements and they are sharing the same files … how can they work together without going mad ?

Idea 1 : they start to use different names, write names of the files with *date*, *names, ….*

Idea 2 : they share tar files after each modification

→ no one knows which is the correct version, or which one was the last working one, who make each modification and why

Sharing files can get messy ….

A team is working in the same project, makes developpements and they are sharing the same files … how can they work together without going mad ?

Idea 1 : they start to use different names, write names of the files with *date*, *names, ….*

Idea 2 : they share tar files after each modification

→ no one knows which is the correct version, or which one was the last working one, who make each modification and why

What is a Version Control Systems ?

Keep track of changes done over the time

→ Backup and store all previous versions

→ Centralize all existing developments done in INCA

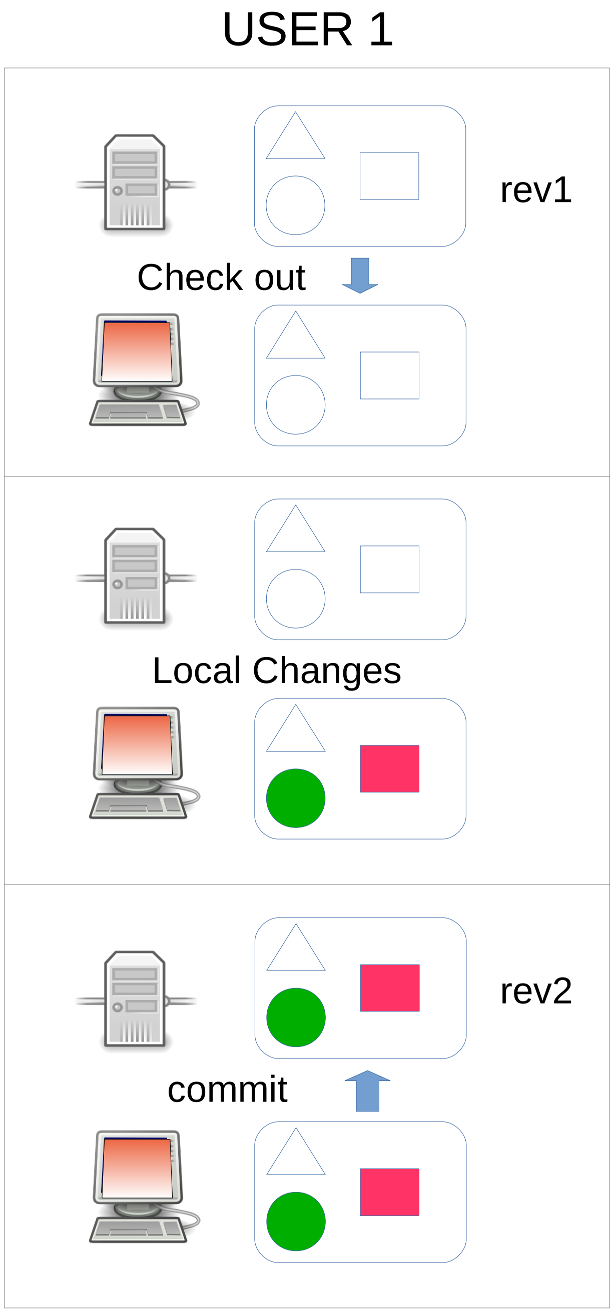
→ Makes it easier to work in a group on the same version and exchange developments before inclusion in the main version

Many differents options : CVS – SVN – Git – Mercurial

How do Version Control Systems work ?

- The VCS allows the user to :
- Review the history of changes from all collaborators
- Revert to a previous version of the work
- Branch from any point from the history for parallel development
- Merge changes from different branches back together
- Easily, quickly, and reliably update copies on several machines.

How do Version Control Systems work ?



USER 1

rev1

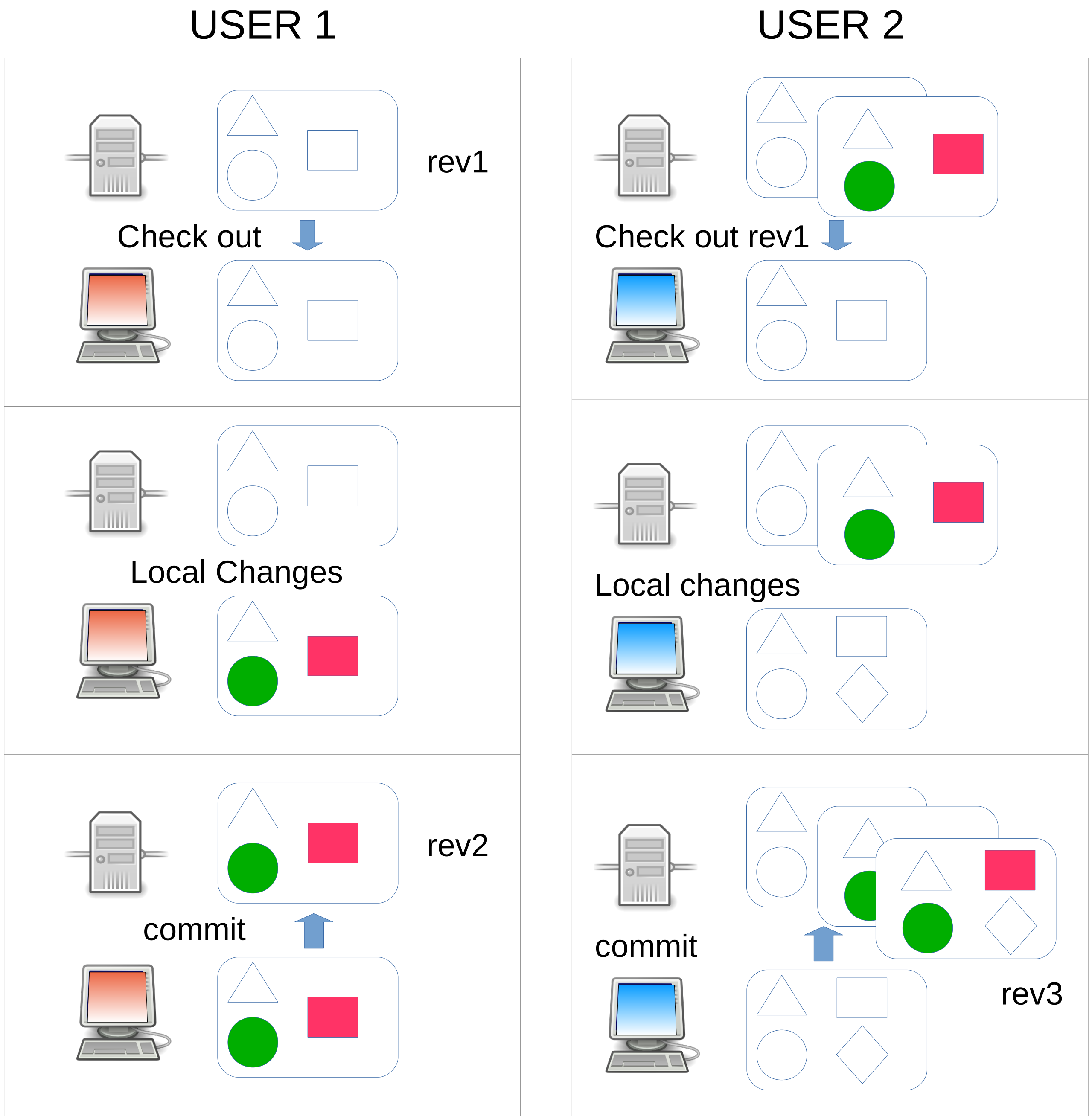
Check out

Local Changes

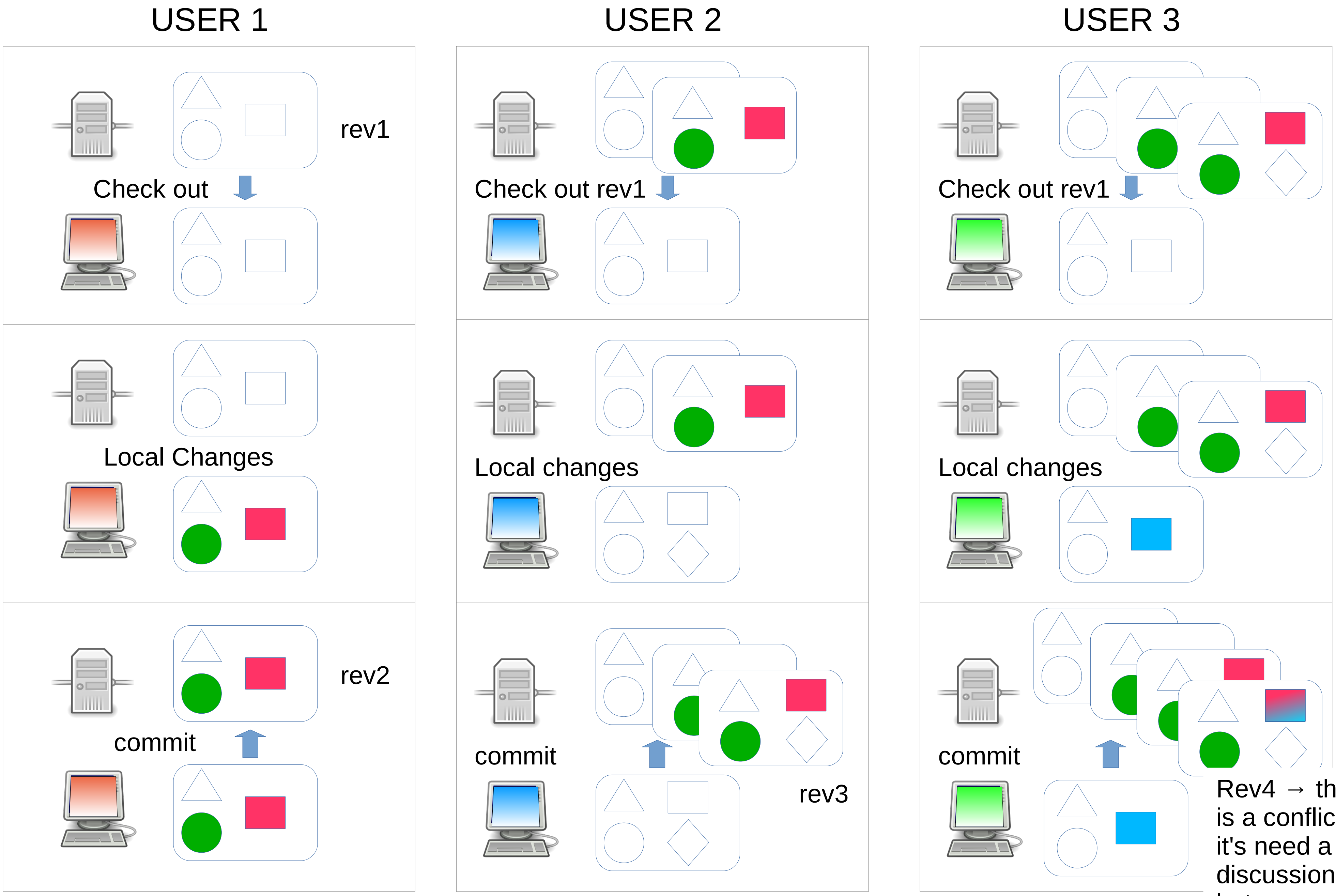
rev2

commit

How do Version Control Systems work ?



How do Version Control Systems work ?



Rev4 → there is a conflict – it's need a discussion between users

Basic use of SVN

svn co repository : Extract a directory and it's sub-directories.
Add « -r rev » for a specific revision.

Example : Extract latest revision of tag INCA5.1.2 of model INCA :

```
>  svn co svn://forge.ipsl.jussieu.fr/inca/svn/tags/INCA5.1.2
MyINCA
```

Example : Extract revision 485 of trunk :

```
>  svn co -r 485
svn://forge.ipsl.jussieu.fr/inca/svn/trunk/INCA5    MyINCA
```

Basic use of SVN

svn info : Information will be printed on the screen about extracted version

Example :

```
> cd modipsl/modeles/INCA
> svn info
Path: .
URL: http://forge.ipsl.jussieu.fr/inca/svn/tags/INCA5.1.2
Repository Root: http://forge.ipsl.jussieu.fr/inca/svn
Repository UUID: dc8988e9-b232-0410-ba9d-85c4b96cce30
Revision: 487
Node Kind: directory
Schedule: normal
Last Changed Author: acosce
Last Changed Rev: 480
Last Changed Date: 2016-01-07 11:03:01 +0100 (Thu, 07 Jan 2016)
```

In this example the version of INCA is tags/INCA5.1.2 and the Version Control revision is 487 for all repositories. You can also see that no changes for this subdirectory are done since revision 480 which is the latest modified revision.

Basic use of SVN

svn stat / svn status : To know which files have been modified compared to extracted version

```
> svn stat
?       arch.fcm
?       config
?       arch.path
M       src/INCA_SRC/mkdvel.F90
M       src/INCA_SRC/ub_inti.F90
M       src/INCA_SRC/mksflx.F90
M       src/INCA_SRC/adjh2o.F90
```

```
> svn help stat
'A' Added
'C' Conflicted
'D' Deleted
'M' Modified
'?' item is not under version control
'!' item is missing
* a newer revision exists on the server
```

Basic use of SVN

svn diff : Show difference compare to extracted version

Example :

```
> svn diff src/INCA_MOD/chem_mod.F90
Index: src/INCA_MOD/chem_mod.F90
==================================================================
=
--- src/INCA_MOD/chem_mod.F90 (revision 486)
+++ src/INCA_MOD/chem_mod.F90 (working copy)
@@ -98,7 +98,7 @@
    REAL, SAVE, ALLOCATABLE        :: nas(:,:,:)   ! non-advected
species( mmr )
 # else
   REAL, SAVE        :: nadv_mass(no_size)
-  REAL, SAVE        :: nas(1)                    ! place holder
+  REAL, SAVE        :: nas(no_size)              ! place holder
 # endif
 !$OMP THREADPRIVATE(nadv_mass)
 !$OMP THREADPRIVATE(nas)
```

Lines starting with “+” are added in the local version (also called working copy).

Lines starting with “-” are removed.

Basic use of SVN

svn update : Update working copy with the latest revision on the server

Updates only with changes on the same branch (the directory and it's sub-directories)
Local changes will be kept. Conflicts can occur if the same file is modified locally and on the server
Changes are done only in the local working directory

Example :
If you extracted a tag or a branch, changes done on the trunk will not be added in your directory.
If there is a conflict on a file, type p for postpone. svn will then save your modifications in a separate file. The file without modifications is also saved in your directory.

Commit with SVN

svn add newfile.f90 Put files and/or directories under version control.
They will be added in next commit

svn rm file.f90 The file will be removed locally and the file is scheduled to be removed from the svn repository in next commit.

svn commit / svn ci Commit all changes in the directory to the server
The revision number is increased.

Best practice for commit
- Add a log message to each commit svn commit -m
- Do whole source directory in each commit, avoid to commit file by file
- If you are several to work on a branch, discuss with the other people before commit
- make all test before to commit, and work with the last revision version.