

Short overview of the multiple usages of SVN, aka Subversion. The goal is to have a depository on the web, which maintains the scripts that are part of ComboCode (excluding any new versions of *MCM* or *GASTRoNoM*).

1 The ComboCode home folder

There is a home folder for ComboCode, and you define yourself what it is. This home folder will have to hold your ComboCode inputfiles. Make the folder, and then run the first check out of SVN (next section) in it. When you do this, a new folder called CC is made automatically in that home folder.

The default is `/home/YOUR_HOME/ComboCode/`. In that case you don't have to specify the folder when running ComboCode with python/ipython; just the filename will do. If you do choose a different path than the default, you will always have to specify the full path when running ComboCode with python/ipython.

2 First check-out

In the example here, I'm using the home folder `/home/robinl/ComboCode/` in which I run the first check out command. When downloading the scripts for the first time, the command you should use is:

```
svn co http://subversion.ster.kuleuven.be/repos/astrophysicalcodes/CC/
```

This will create a new folder called CC in the folder `/home/robinl/ComboCode/`. From then on, that folder is recognized as a sort of working folder for SVN, from which you can add, update, check the status and so on for every file in the folder, i.e. including all the ComboCode scripts that you need. This is obviously also the folder that you want to include in your PYTHONPATH in your `.bash_profile`. The command in `.bash_profile` should be something like this for the example given here:

```
PYTHONPATH=/home/robinl/ComboCode/CC:$PYTHONPATH
export PYTHONPATH
```

When you are using ipython, you also want to run it from this particular folder. If something is missing, give me a shout: I probably just forgot one of the python scripts. Though, make sure you remembered to source the new `.bash_profile` before running the main script, so Python knows where to look for the scripts when you start it up.

3 Updating your scripts

If at some time you want to update the scripts in the folder to the most recent version (it's probably safest to update the scripts regularly, even if nothing changed, then you're sure you're always using the version you should be using), you can use

```
svn up
```

There is no need to use the `co` command again. Only use that command when you make an entirely new CC folder that isn't known yet by SVN. If you want information on what's been updated as compared to a last version:

```
svn log
```

or

```
svn log script.py
```

for a log on a specific file. I will try to keep the log as updated as possible, so you can always check what's been updated or what hasn't, if at any time that would be important. You can ask for the current status of the files in the CC folder as well:

```
svn status
```

inside the `/home/robinl/ComboCode/CC/` folder. Every file will then be listed. If a file is not known by the SVN server, it will be listed as `'?'`. If it's just been updated from an older version to the most recent one, it will be listed as `'U'`. If you added a file to the SVN server or if you made a change to a file before doing a commit (only for those who have writing access), it will be listed as `'A'` and `'M'` respectively. If the file has not been updated and is the same as the version on the SVN server, it is not listed at all.

4 Managing files in the depository

All of you currently have read-only access to the scripts. If that changes at any point, or if you want to set up your own SVN account, you may want to use following commands as well (to be honest, though, I'm including those here so I can remember them myself... :-)):

```
svn add script.py
```

to add a script for the first time to the SVN server.

```
svn commit
```

to update all scripts known by the SVN server with the versions currently in your SVN folder. An Emacs window will pop up, allowing you to give a description of the changes you've made in the updated scripts, since the last version. Save, and you're done!

5 Some final remarks concerning ComboCode

Whenever you do any of the above, you may be prompted for your login and your password. Your login is always your full capitalized name without spaces: RobinLombaert, for instance. I will e-mail you your password in a separate mail. I think it's possible to have it changed, but then you will have to talk to Wim De Meester, of the system group at the IvS.

Please, remember to update your CC scripts regularly! If major changes have been made, I will let you know. Other files on SVN (or not on SVN):

- I'll make sure I keep an updated input file for ComboCode (named `inputComboCode.dat` in CC/) with all possible keywords in the SVN folder. Copy that file to `/home/robinl/ComboCode/` (or whatever your ComboCode home folder is) and name it whatever you want, but don't use this file itself. Any input file is supposed to be in the ComboCode home folder, not in CC/, though you can define the ComboCode home folder as whatever way you want.
- There is also a plot input file in the CC folder, which is used to turn on and off several plotting options. Update it, as you go with what you need, but keep in mind that when you update your scripts, that this file will also be overwritten with whatever is on SVN. You'll still want the most recent version of this though, because it is updated as I add new plotting modules. (`plotInputComboCode.dat`)
- The `CC_Structure` file is also in here. Check that file to see what folders you have to copy over from my home at `/home/robinl/`.
- The inputfiles that manage the keywords vs databases (`Input_Keywords_*.dat`), and the interaction between GASTRoNoM and MCMMax (`Mutable_Parameters_*.dat`) are also here, so they'll remain up-to-date.

ComboCode on Subversion

- The other inputfiles that you need for ComboCode (Dust.dat, Molecule.dat, Star.dat and Pacs_Resolution.dat in /ComboCode/Data) are not on SVN: They should remain the same as before, and you can make your own changes to some of them if you want. If anything changes in them as far as syntax is concerned, I will let you know and send you a new example file manually.
- Any special inputfiles for *MCM* and *GASTRoNOoM* (usually templates for running the codes) are not on SVN either. They should not be changed ever, so it shouldn't be a problem anyhow. These are:
 - inputMCM.dat, Spec.out and visibilities_baseline.out in ~/MCM/src/;
 - vic_job_example.sh, vic_run_jobs_example.sh, inputGASTRoNOoM.dat and execGASTRoNOoM_example in ~/GASTRoNOoM/;
 - any other code-specific stuff you need regardless of using ComboCode (executables, collision rates, opacity files, ... you should get them when you copy the MCM and GASTRoNOoM folders over from me anyway).

Happy modelling!