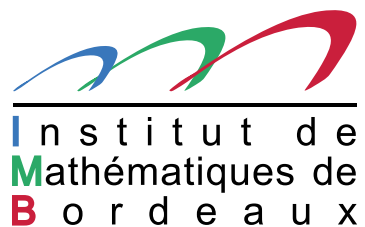

Tutorial: Using GIT with PARI

Karim Belabas

`http://pari.math.u-bordeaux.fr/`



Initial configuration (simplified)

- `git clone http://pari.math.u-bordeaux.fr/git/pari.git`
- `cd pari`
- `./Configure --prefix=GPDIR && make -j4 gp`
- `make doc`
- `make test-all`
- `make install`
- `GPDIR/bin/gp`

Your local `pari/` directory is now in synch with the *master* branch on the pari server.

Simple GIT commands

- `git status` : files that differ from fetched sources (aka local changes).
- `git diff` : show actual differences (in tracked files).
- `git log` : summary of latest commits in current branch.
- `git pull` : update local sources from remote ; problems will arise if there are local changes !

Simple GIT commands

- `git status` : files that differ from fetched sources (aka local changes).
- `git diff` : show actual differences (in tracked files).
- `git log` : summary of latest commits in current branch.
- `git pull` : update local sources from remote ; problems will arise if there are local changes !
- `git rebase origin/master` : update local sources ; more robust.

Simple GIT commands

- `git status` : files that differ from fetched sources (aka local changes).
- `git diff` : show actual differences (in tracked files).
- `git log` : summary of latest commits in current branch.
- `git pull` : update local sources from remote ; problems will arise if there are local changes!
- `git rebase origin/master` : update local sources ; more robust.
- Now tell git who you are (just once !):
`git config --global user.name "YOUR NAME"`
`git config --global user.email "YOUR EMAIL ADDRESS"`

More complex GIT commands (prepare a patch)

- `git checkout -b mybranch` : create a new local branch (forking current branch) and switch to it; always fork from `master` before an edit!
- `git checkout master / git checkout mybranch` : switch between `master` and `mybranch`.
- Edit at will, `Configure` (if you modified the `config/` subsystem or added new files), `make statest-all` (if you modified C source files), `make doc` (if you modified the documentation), `make checkspaces` (always!).
- `git commit -a` : store all current edits in a new commit (-a for *all* edits). Do that from `mybranch`.
- `git format-patch master` : prepare a patch from successive commits that you added to `mybranch` (the argument `master` is the reference point to start from). You can send them to Bill or I by email.
- Once we have applied your patch to `origin/master`, `git branch -D mybranch` : delete your local branch.

GIT branches

- `git branch` : list of local branches
- `git branch -a` : list of local and remote branches
- `git checkout origin/kb-mf -b mykb-mf` : create a local branch (forking the remote branch kb-mf). `Configure`, `make`, etc.
- `git checkout master` : don't forget to come back to the master branch !