

# Compiling PARI from the GIT repository

B. Allombert and K. Belabas

IMB  
CNRS/Université de Bordeaux

9/01/2017



This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N° 676541

## Introduction

This talk focuses on the current development version of the PARI library (2.10.\*), available from our GIT repository, see

<http://pari.math.u-bordeaux.fr/anongit.html>

The text of this talk is available in the files `sources.*` in

[http://pari.math.u-bordeaux.fr/Events/  
PARI2017/talks/](http://pari.math.u-bordeaux.fr/Events/PARI2017/talks/)

## Windows users

Download a precompiled 64bit installer `Pari64*.exe` or 32bit binary `Pari32*.exe` from

`http://pari.math.u-bordeaux.fr/pub/pari/windows/snapshots/`

Also available are precompiled 64bit binary `gp64-gmp-git*.exe` or 32bit binary `gp32-gmp-git*.exe`.

## Mac OS users

Download a precompiled DMG `PariGP-*.dmg` from

`http://pari.math.u-bordeaux.fr/pub/pari/mac/snapshots/`

Also are precompiled binary `gp-git*-osx`.

## Debian/Ubuntu

On Debian/Ubuntu, to install all the packages required to build pari from source:

```
sudo apt-get build-dep pari
```

## From source with GIT

Clone the PARI repository with GIT (~ 150MB).

```
git clone http://pari.math.u-bordeaux.fr/git/pari.git
cd pari
```

## From source without GIT

Download `pari-2.10-*.tar.gz` from

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

and unpack it

```
tar xf pari-2.10-*.tar.gz
cd pari-2.10-*
```

## PARI compilation

```
./Configure --prefix=GPDIR --mt=pthread
make -j4 gp
make doc
make statest-all
make install
make install-bin-sta
./Configure --prefix=GPDIR.dbg -g
make -j4 gp
make install -C Olinux-x86_64.dbg
GPDIR/bin/gp
```



## GP configuration

Create and customize `~/.gprc`. Add

```
histfile = "~/.gp\_history"  
colors = "lightbg" \\ or "darkbg"  
lines = 40  
parisizemax = 4G \\ or the maximum amount of memory  
                \\ GP can use (important)  
read "~/.gprc.gp"
```

Create an empty file `~/.gprc.gp`

## GP2C compilation

With GIT (and automake, autoconf):

```
git clone http://pari.math.u-bordeaux.fr/git/gp2c.git
cd gp2c
./autogen.sh
```

Without GIT: download GP2C from <http://pari.math.u-bordeaux.fr/download.html#gp2c>

```
tar xf gp2c-0.0.10.tar.gz
cd gp2c-0.0.10
```

## GP2C compilation

```
./configure --prefix=$PWD/../GPDIR \  
    --with-paricfg=../GPDIR/lib/pari/pari.cfg \  
    --with-paricfg.dbg=../GPDIR.dbg/lib/pari/pari.cfg  
make check  
make install  
cd ..  
GPDIR/bin/gp2c -v
```

## Updating GIT

To update GIT to the most recent revision:

```
git fetch
git rebase origin/master
```

## Accessing remotes branches

**Remote branches** : by default, we only know the official repository (the one we cloned from, `origin`), but you can add other ones.

```
> git branch -a      # -a = add remote branches
...    many branches omitted !
remotes/origin/bill-ellisomatnf
...
```

```
> git checkout origin/bill-ellisomatnf -b
sea-parallel
```

Here we created a local branch `sea-parallel` out of the `origin/bill-ellisomatnf`, and switched to it. (Configure, make, test things...)

`git checkout master` to get back to main development branch.

## Git branches

You can go back to the master branch by doing

```
git checkout master
```

You can see the latest commits in the current branch with

```
git log
```