

Michael Plagge

How to compile and install OpenFOAM on Mac OS X

Last changed: January 22nd, 2012

Revision: 0.1

Contents

1 Basic requirements	4
2 Installing and compiling OpenFOAM 2.1 with GCC 4.2 and OpenMPI 1.5.4	7

1 Basic requirements

This manual assumes that the reader is familiar with Mac OS X, the installation of applications as well as with basic commands in the Terminal.app. This manual covers the installation of OpenFOAM on Mac OS X 10.6.8, also called Snow Leopard.

What do you need?

- Apple Mac OS X 10.6.8
- Apple Developer Tools 3.2.6 (delivered on the DVDs you've got with your mac, or you can download this here: <http://developer.apple.com/>)
- MacPorts software, you can download this here: <http://www.macports.org>

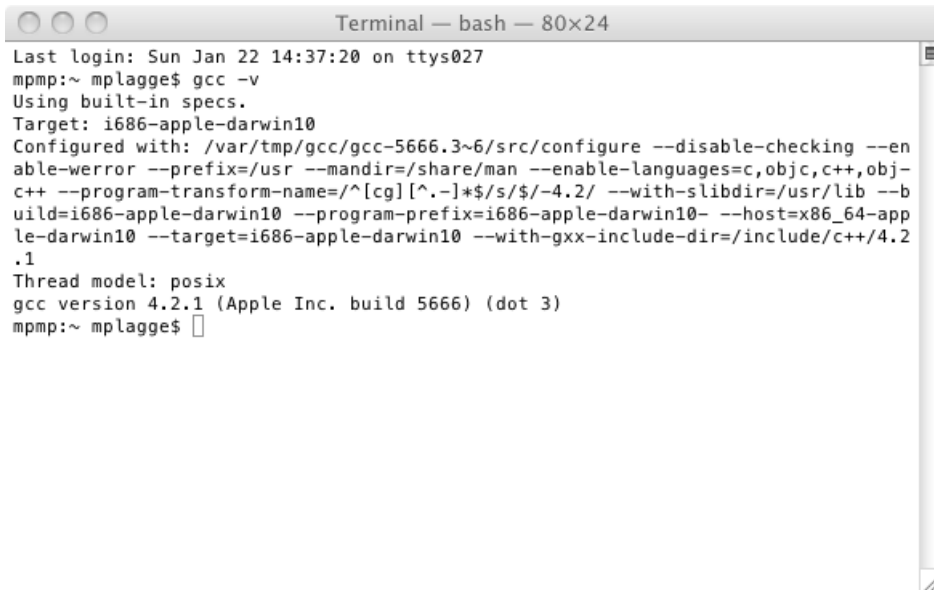
For downloading the Apple Developer Tools you might have to become a member of the Apple developer site, it is for free yet.

How do I know if the above mentioned software is already installed?

Open the Terminal.app you find in your */Applications/Utilities* folder:



Type `gcc -v`, the output should look like this:



```
Terminal — bash — 80x24
Last login: Sun Jan 22 14:37:20 on ttys027
mpmp:~ mplagge$ gcc -v
Using built-in specs.
Target: i686-apple-darwin10
Configured with: /var/tmp/gcc/gcc-5666.3~6/src/configure --disable-checking --enable-werror --prefix=/usr --mandir=/share/man --enable-languages=c,objc,c++,obj-c++ --program-transform-name=/^[cg][^.-]*$/s/$/-4.2/ --with-slibdir=/usr/lib --build=i686-apple-darwin10 --program-prefix=i686-apple-darwin10- --host=x86_64-apple-darwin10 --target=i686-apple-darwin10 --with-gxx-include-dir=/include/c++/4.2.1
Thread model: posix
gcc version 4.2.1 (Apple Inc. build 5666) (dot 3)
mpmp:~ mplagge$
```

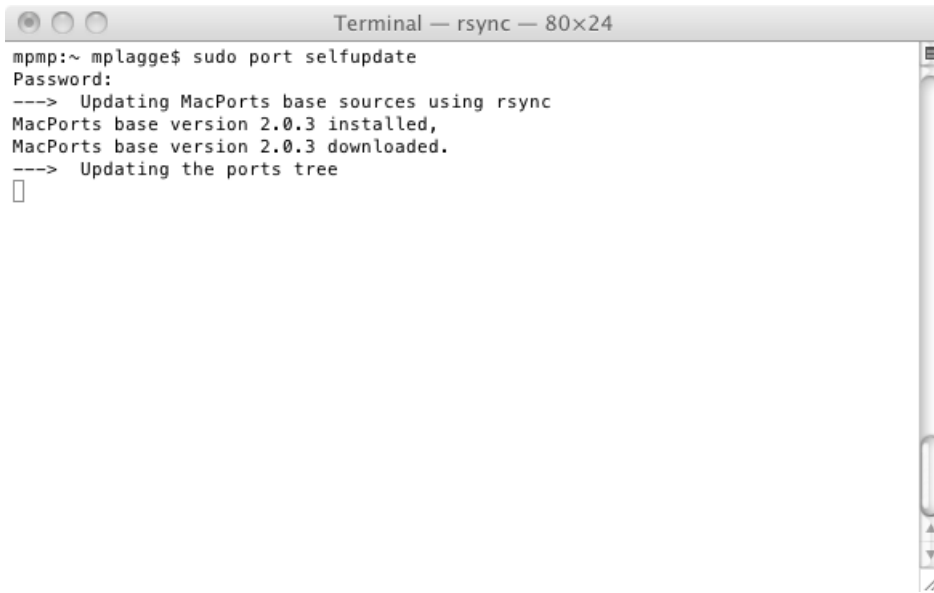
Check for the MPI installation as well by typing `mpirun -V`, please note that the option is different (`-V` instead of `-v`):



```
Terminal — bash — 80x24
mpmp:~ mplagge$ mpirun -V
mpirun (Open MPI) 1.2.8

Report bugs to http://www.open-mpi.org/community/help/
mpmp:~ mplagge$
```

After installing the MacPort software, see <http://www.macports.org/install.php> for installation guide, we check the installation with `sudo port selfupdate`, you have to confirm the sudo with your password:

A terminal window titled "Terminal — rsync — 80x24" showing the execution of the command `sudo port selfupdate`. The output indicates that the MacPorts base sources are updated using rsync, the base version 2.0.3 is installed and downloaded, and the ports tree is updated. The prompt returns to `mpmp:~ mplagge$`.

```
mpmp:~ mplagge$ sudo port selfupdate
Password:
----> Updating MacPorts base sources using rsync
MacPorts base version 2.0.3 installed,
MacPorts base version 2.0.3 downloaded.
----> Updating the ports tree
[]
```

Now we are installing the GCC 4.6 by entering

`sudo port install gcc46`

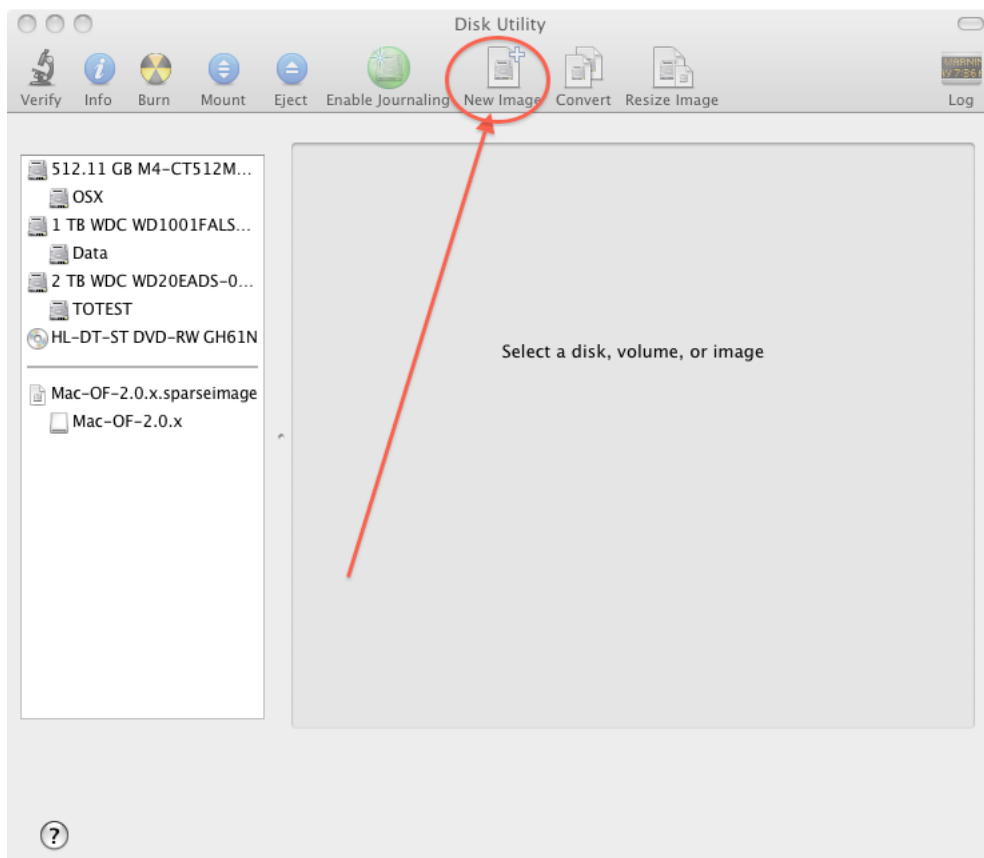
on the command line. After the operation is done you could check the version and it should look like this:

A terminal window titled "Terminal — bash — 80x24" showing the execution of the command `gcc-mp-4.6 -v`. The output displays the configuration details for GCC 4.6.2, including the target architecture (x86_64-apple-darwin10), the prefix path (/opt/local), and various compiler options. The final output is `gcc version 4.6.2 (GCC)`. The prompt returns to `mpmp:~ mplagge$`.

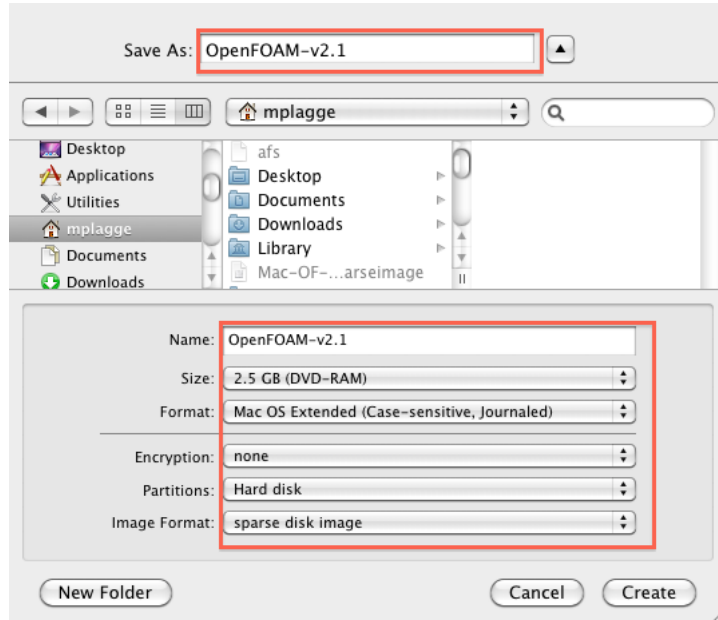
```
mpmp:~ mplagge$ gcc-mp-4.6 -v
Using built-in specs.
COLLECT_GCC=gcc-mp-4.6
COLLECT_LTO_WRAPPER=/opt/local/libexec/gcc/x86_64-apple-darwin10/4.6.2/lto-wrapper
Target: x86_64-apple-darwin10
Configured with: ../gcc-4.6.2/configure --prefix=/opt/local --build=x86_64-apple-darwin10 --enable-languages=c,c++,objc,obj-c++,lto,fortran --libdir=/opt/local/lib/gcc46 --includedir=/opt/local/include/gcc46 --infodir=/opt/local/share/info --mandir=/opt/local/share/man --datarootdir=/opt/local/share/gcc-4.6 --with-local-prefix=/opt/local --with-libiconv-prefix=/opt/local --with-system-zlib --disable-nls --program-suffix=-mp-4.6 --with-gxx-include-dir=/opt/local/include/gcc46/c++/ --with-gmp=/opt/local --with-mpfr=/opt/local --with-mpc=/opt/local --with-mp=/opt/local --enable-stage1-checking --disable-multilib --enable-lto
Thread model: posix
gcc version 4.6.2 (GCC)
mpmp:~ mplagge$ []
```

2 Installing and compiling OpenFOAM 2.1 with GCC 4.2 and OpenMPI 1.5.4

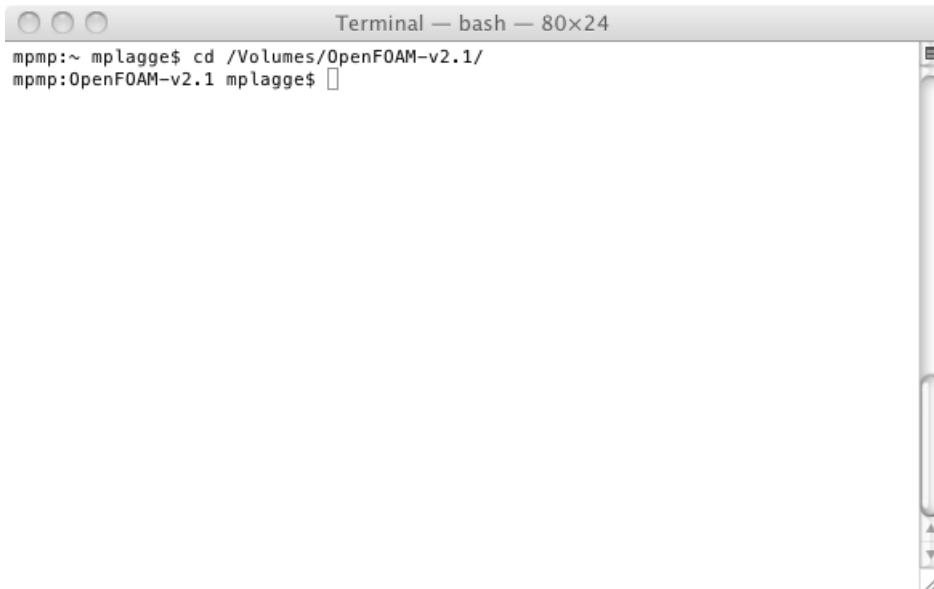
First, we create a disk image with the Apple Disk Utility. Open */Applications/Utilities/Disk Utility*. Click New Image:



It is important to set as Format: Mac OS Extended (Case-sensitive,Journalled). The sparseimage will grow with the place used inside the image. Create a sparseimage with the following settings (examples, you could change the values):



Go to the directory of the image by typing `cd /Volumes/OpenFOAM-v2.1`:

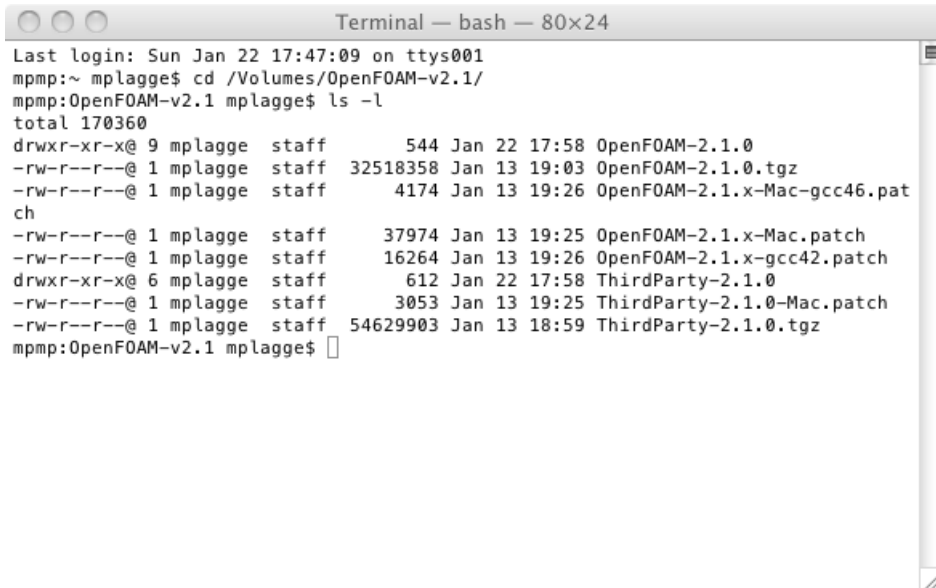


Use a browser to download the archives of OpenFoam 2.1 and ThirdParty 2.1 from

<http://openfoam.org/download/source.php> and copy the archives into the image. Extract the archives by double clicking on each one.

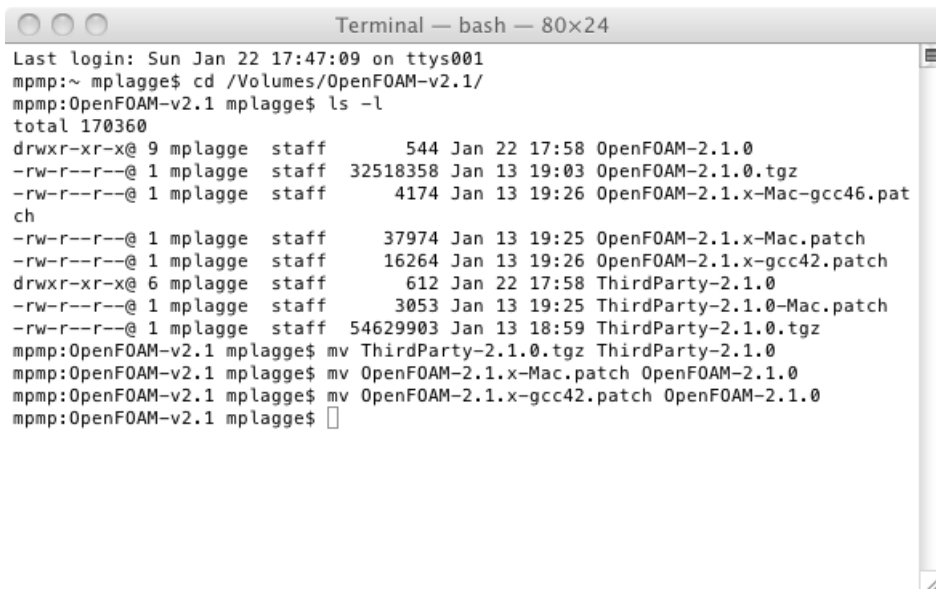
Then download the patches created by Bernhard Gschaider, see <http://www.cfd-online.com/Forums/openfoam-installation-windows-mac/95763-1-9-aka-2-1-mac-os-x.html#post340297>

Copy or move them into the image. Your image content should look like this:



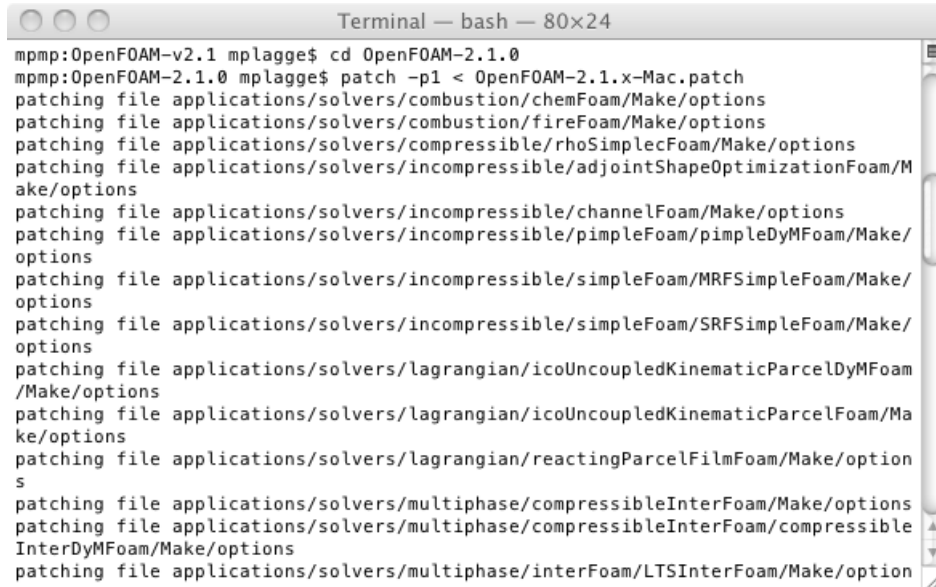
```
Terminal — bash — 80x24
Last login: Sun Jan 22 17:47:09 on ttys001
mpmp:~ mplagge$ cd /Volumes/OpenFOAM-v2.1/
mpmp:OpenFOAM-v2.1 mplagge$ ls -l
total 170360
drwxr-xr-x@ 9 mplagge  staff      544 Jan 22 17:58 OpenFOAM-2.1.0
-rw-r--r--@ 1 mplagge  staff  32518358 Jan 13 19:03 OpenFOAM-2.1.0.tgz
-rw-r--r--@ 1 mplagge  staff   4174 Jan 13 19:26 OpenFOAM-2.1.x-Mac-gcc46.pat
ch
-rw-r--r--@ 1 mplagge  staff   37974 Jan 13 19:25 OpenFOAM-2.1.x-Mac.patch
-rw-r--r--@ 1 mplagge  staff   16264 Jan 13 19:26 OpenFOAM-2.1.x-gcc42.patch
drwxr-xr-x@ 6 mplagge  staff    612 Jan 22 17:58 ThirdParty-2.1.0
-rw-r--r--@ 1 mplagge  staff   3053 Jan 13 19:25 ThirdParty-2.1.0-Mac.patch
-rw-r--r--@ 1 mplagge  staff  54629903 Jan 13 18:59 ThirdParty-2.1.0.tgz
mpmp:OpenFOAM-v2.1 mplagge$
```

Move the appropriate patches into the directories, see below:



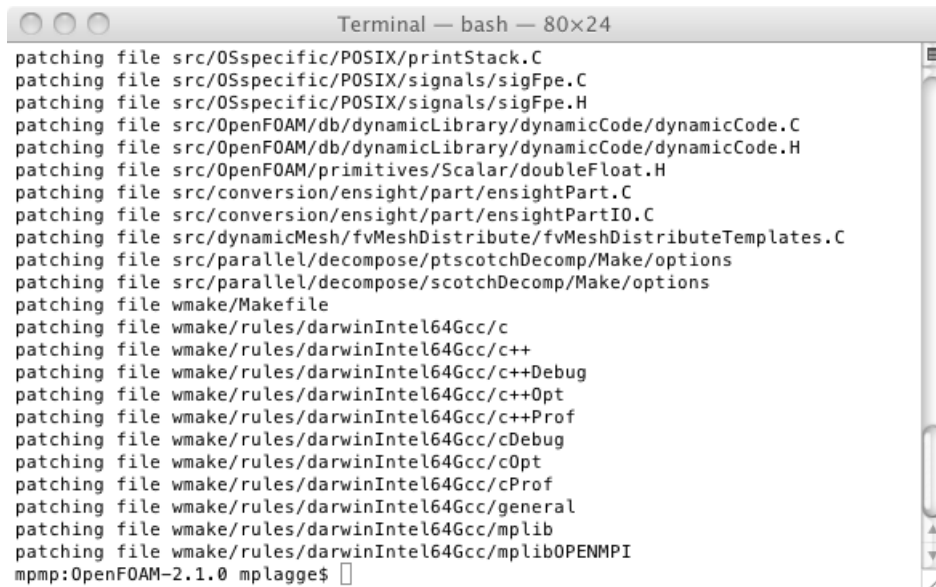
```
Terminal — bash — 80x24
Last login: Sun Jan 22 17:47:09 on ttys001
mpmp:~ mplagge$ cd /Volumes/OpenFOAM-v2.1/
mpmp:OpenFOAM-v2.1 mplagge$ ls -l
total 170360
drwxr-xr-x@ 9 mplagge  staff      544 Jan 22 17:58 OpenFOAM-2.1.0
-rw-r--r--@ 1 mplagge  staff  32518358 Jan 13 19:03 OpenFOAM-2.1.0.tgz
-rw-r--r--@ 1 mplagge  staff   4174 Jan 13 19:26 OpenFOAM-2.1.x-Mac-gcc46.pat
ch
-rw-r--r--@ 1 mplagge  staff   37974 Jan 13 19:25 OpenFOAM-2.1.x-Mac.patch
-rw-r--r--@ 1 mplagge  staff   16264 Jan 13 19:26 OpenFOAM-2.1.x-gcc42.patch
drwxr-xr-x@ 6 mplagge  staff    612 Jan 22 17:58 ThirdParty-2.1.0
-rw-r--r--@ 1 mplagge  staff   3053 Jan 13 19:25 ThirdParty-2.1.0-Mac.patch
-rw-r--r--@ 1 mplagge  staff  54629903 Jan 13 18:59 ThirdParty-2.1.0.tgz
mpmp:OpenFOAM-v2.1 mplagge$ mv ThirdParty-2.1.0.tgz ThirdParty-2.1.0
mpmp:OpenFOAM-v2.1 mplagge$ mv OpenFOAM-2.1.x-Mac.patch OpenFOAM-2.1.0
mpmp:OpenFOAM-v2.1 mplagge$ mv OpenFOAM-2.1.x-gcc42.patch OpenFOAM-2.1.0
mpmp:OpenFOAM-v2.1 mplagge$
```

Change into the OpenFOAM directory and apply the first patch via `patch -p1 < *.patch`:



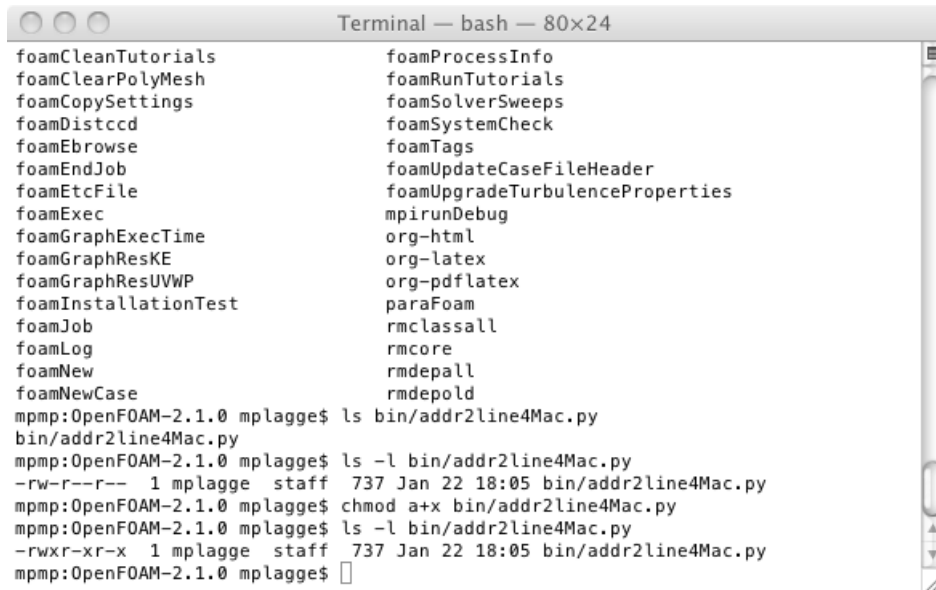
```
Terminal — bash — 80x24
mpmp:OpenFOAM-v2.1 mplagge$ cd OpenFOAM-2.1.0
mpmp:OpenFOAM-2.1.0 mplagge$ patch -p1 < OpenFOAM-2.1.x-Mac.patch
patching file applications/solvers/combustion/chemFoam/Make/options
patching file applications/solvers/combustion/fireFoam/Make/options
patching file applications/solvers/compressible/rhoSimplecFoam/Make/options
patching file applications/solvers/incompressible/adjointShapeOptimizationFoam/Make/options
patching file applications/solvers/incompressible/channelFoam/Make/options
patching file applications/solvers/incompressible/pimpleFoam/pimpleDyMFoam/Make/options
patching file applications/solvers/incompressible/simpleFoam/MRFSimpleFoam/Make/options
patching file applications/solvers/incompressible/simpleFoam/SRFSimpleFoam/Make/options
patching file applications/solvers/lagrangian/icoUncoupledKinematicParcelDyMFoam/Make/options
patching file applications/solvers/lagrangian/icoUncoupledKinematicParcelFoam/Make/options
patching file applications/solvers/lagrangian/reactingParcelFilmFoam/Make/options
patching file applications/solvers/multiphase/compressibleInterFoam/Make/options
patching file applications/solvers/multiphase/compressibleInterFoam/compressibleInterDyMFoam/Make/options
patching file applications/solvers/multiphase/interFoam/LTSInterFoam/Make/options
```

The end of the output should look like this:



```
Terminal — bash — 80x24
patching file src/OSspecific/POSIX/printStack.C
patching file src/OSspecific/POSIX/signals/sigFpe.C
patching file src/OSspecific/POSIX/signals/sigFpe.H
patching file src/OpenFOAM/db/dynamicLibrary/dynamicCode/dynamicCode.C
patching file src/OpenFOAM/db/dynamicLibrary/dynamicCode/dynamicCode.H
patching file src/OpenFOAM/primitives/Scalar/doubleFloat.H
patching file src/conversion/ensight/part/ensightPart.C
patching file src/conversion/ensight/part/ensightPartIO.C
patching file src/dynamicMesh/fvMeshDistribute/fvMeshDistributeTemplates.C
patching file src/parallel/decompose/ptscotchDecomp/Make/options
patching file src/parallel/decompose/scotchDecomp/Make/options
patching file wmake/Makefile
patching file wmake/rules/darwinIntel64Gcc/c
patching file wmake/rules/darwinIntel64Gcc/c++
patching file wmake/rules/darwinIntel64Gcc/c++Debug
patching file wmake/rules/darwinIntel64Gcc/c++Opt
patching file wmake/rules/darwinIntel64Gcc/c++Prof
patching file wmake/rules/darwinIntel64Gcc/cDebug
patching file wmake/rules/darwinIntel64Gcc/cOpt
patching file wmake/rules/darwinIntel64Gcc/cProf
patching file wmake/rules/darwinIntel64Gcc/general
patching file wmake/rules/darwinIntel64Gcc/mplib
patching file wmake/rules/darwinIntel64Gcc/mplibOPENMPI
mpmp:OpenFOAM-2.1.0 mplagge$
```

Add executive rights to a python script like this:



```
Terminal — bash — 80x24
foamCleanTutorials      foamProcessInfo
foamClearPolyMesh      foamRunTutorials
foamCopySettings       foamSolverSweeps
foamDistccd            foamSystemCheck
foamEbrowse            foamTags
foamEndJob             foamUpdateCaseFileHeader
foamEtcFile            foamUpgradeTurbulenceProperties
foamExec               mpirunDebug
foamGraphExecTime      org-html
foamGraphResKE         org-latex
foamGraphResUVWP       org-pdflatex
foamInstallationTest   paraFoam
foamJob                rmclassall
foamLog               rmcore
foamNew                rmdepall
foamNewCase            rmdepold
mpmp:OpenFOAM-2.1.0 mplagge$ ls bin/addr2line4Mac.py
bin/addr2line4Mac.py
mpmp:OpenFOAM-2.1.0 mplagge$ ls -l bin/addr2line4Mac.py
-rw-r--r--  1 mplagge  staff  737 Jan 22 18:05 bin/addr2line4Mac.py
mpmp:OpenFOAM-2.1.0 mplagge$ chmod a+x bin/addr2line4Mac.py
mpmp:OpenFOAM-2.1.0 mplagge$ ls -l bin/addr2line4Mac.py
-rwxr-xr-x  1 mplagge  staff  737 Jan 22 18:05 bin/addr2line4Mac.py
mpmp:OpenFOAM-2.1.0 mplagge$
```

Apply the second patch for the compiler, in this case we use the GCC 4.6 so choose the right patch by typing

```
patch -p1 < OpenFOAM-2.1.x-Mac-gcc46.patch
```

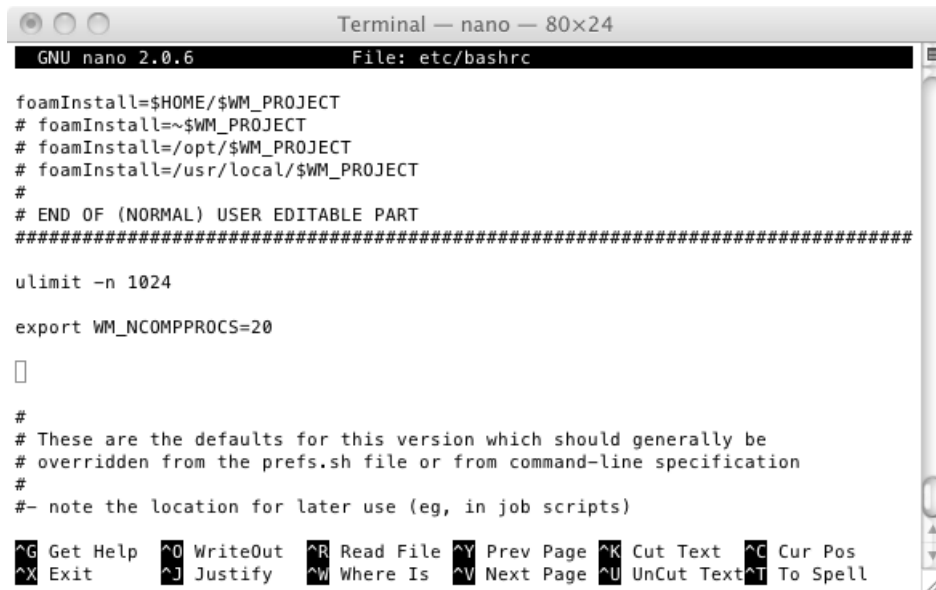
Open the file `/etc/bashrc` with a text editor, e. g. Text Wrangler or `pico` and add these lines:

```
ulimit -n 1024
```

```
export WM_NCOMPPROCS=20
```

The first entry is to increase the number of files which one process may open at a time, the second you should set to the number of processes your CPU could manage in parallel, e. g. 4, 8, 16, ...

It should look like this:



```
Terminal — nano — 80x24
GNU nano 2.0.6 File: etc/bashrc

foamInstall=$HOME/$WM_PROJECT
# foamInstall=~$WM_PROJECT
# foamInstall=/opt/$WM_PROJECT
# foamInstall=/usr/local/$WM_PROJECT
#
# END OF (NORMAL) USER EDITABLE PART
#####

ulimit -n 1024

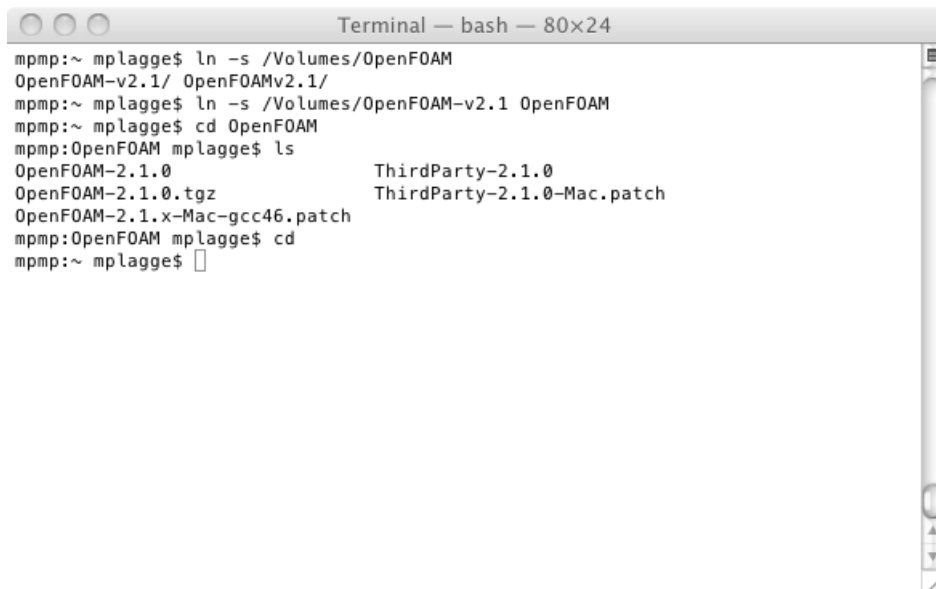
export WM_NCOMPPROCS=20

[]

#
# These are the defaults for this version which should generally be
# overridden from the prefs.sh file or from command-line specification
#
#- note the location for later use (eg, in job scripts)

^G Get Help ^O WriteOut ^R Read File ^Y Prev Page ^K Cut Text ^C Cur Pos
^X Exit ^J Justify ^W Where Is ^V Next Page ^U UnCut Text ^T To Spell
```

Go back to your home directory by simply typing `cd`. Then create a symbolic link with `ln -s /Volumes/OpenFOAM-v2.1 OpenFOAM`.



```
Terminal — bash — 80x24
mpmp:~ mplagge$ ln -s /Volumes/OpenFOAM
OpenFOAM-v2.1/ OpenFOAMv2.1/
mpmp:~ mplagge$ ln -s /Volumes/OpenFOAM-v2.1 OpenFOAM
mpmp:~ mplagge$ cd OpenFOAM
mpmp:OpenFOAM mplagge$ ls
OpenFOAM-2.1.0 ThirdParty-2.1.0
OpenFOAM-2.1.0.tgz ThirdParty-2.1.0-Mac.patch
OpenFOAM-2.1.x-Mac-gcc46.patch
mpmp:OpenFOAM mplagge$ cd
mpmp:~ mplagge$ []
```

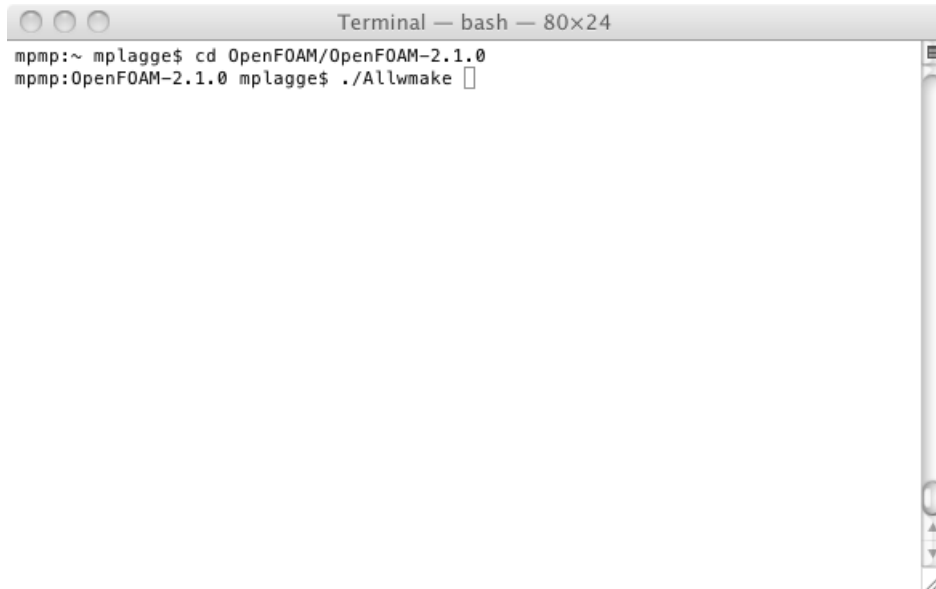
Go back to your home directory again. Now you have to source the settings by `. OpenFOAM/OpenFOAM-2.1.0/etc/bashrc`:

A terminal window titled "Terminal — bash — 80x24" with three window control buttons (red, yellow, green) on the left. The terminal content shows a user named "mpmp" in a directory "~ mplugge" executing the command ". OpenFOAM/OpenFOAM-2.1.0/etc/bashrc". The prompt changes to "mpmp:~ mplugge\$" and a cursor is visible on the second line.

```
mpmp:~ mplugge$ . OpenFOAM/OpenFOAM-2.1.0/etc/bashrc
mpmp:~ mplugge$
```

No error or other output should occur.

Now we go to the OpenFOAM directory and start compilation. Take a coffee. And a cookie. Or two:

A screenshot of a terminal window titled "Terminal — bash — 80x24". The terminal shows the user navigating to the OpenFOAM directory and running the compilation command. The prompt changes from the user's home directory to the OpenFOAM directory.

```
mpmp:~ mplagge$ cd OpenFOAM/OpenFOAM-2.1.0
mpmp:OpenFOAM-2.1.0 mplagge$ ./Allwmake
```

If you want to keep a logfile in case of errors, type `./Allwmake | tee Allwmake.log`. You're done.