# Making/Compiling a large project

Programming Concepts in Scientific Programming

EPFL, Master class

September 28, 2017

# Problem

- Using *gcc* manually is cumbersome

```
g++   -Dakantu_EXPORTS -std=c++0x -DBOOST_RESULT_OF_USE_TR1 -Wall -O3
      -DNDEBUG -DAKANTU_NDEBUG -fPIC -Iakantu -I/usr/include/scotch -
     o aka_array.cc.o -c aka_array.cc
```

- No one actually wants to type these things (well I do not)
- When the project is huge, we have long compilation times (Firefox contains currently 7412 files for the browser part)
- Modification of a single file should lead to a single compilation and a single linking step: dependencies ?

# Solutions

## GNU Make and Makefiles

- Create a file *Makefile* with the *rules* to apply in order to create objects
- Syntax

```
rule: dependency1 dependency2
  command1
  command2
```

- Example to build a 2 file project

```
all: file1.o file2.o
  g++ file1.o file2.o -o exec

file1.o: file1.cc
  g++ -c file1.cc -o file1.o

file2.o: file2.cc
  g++ -c file2.cc -o file2.o
```

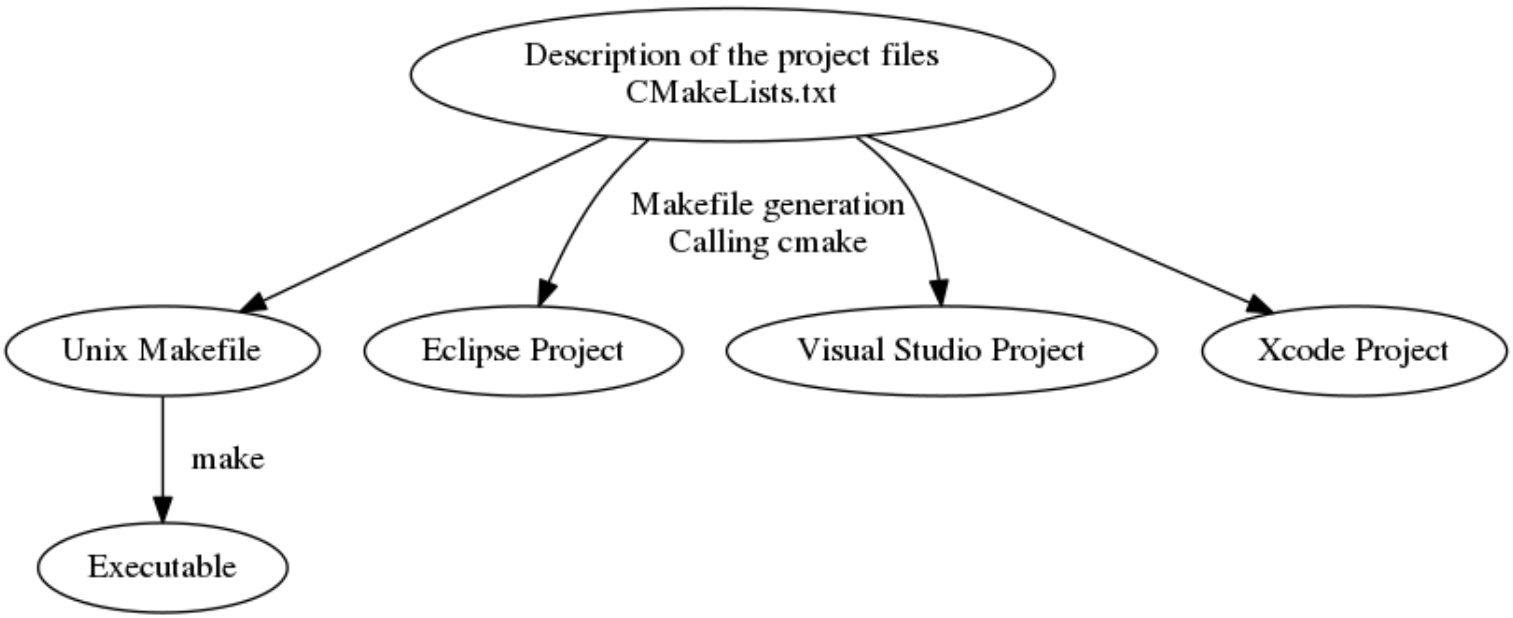
# Makefile generators

- The GNU build system (Autotools) is a suite of programming tools designed to assist in making source code packages (Unix-like systems)
  - The project started in 1994 (first commit)
- CMake is cross-platform free and open-source software for managing the build process of software using a compiler-independent method
  - The project started in 1999
  - Maintained by the Kitware company (which manages Paraview and VTK)
  - Multi-plateform
  - Simple

# CMake flow

## cmake and ccmake program

How to produce the makefiles ?



# Project description

## CMakeLists.txt files

- General information

```
cmake_minimum_required (VERSION 2.6)
project (My super project)
```

- What object to construct

```
add_executable(hello
  main.cc
  hello.cc
)
```

- Adding external libraries dependencies

```
target_link_libraries(hello m eigen zlib)
```

- Managing sub-directories

```
add_subdirectory(sub_directory_name)
```

Let's look at it within CLion

On the repository of the class...