

GETTING STARTED

Idefix

PART 1: ON YOUR LAPTOP



this presentation

PART 2: ON BIGFOOT (GPU)



Geoffroy's presentation

<https://idefix.readthedocs.io/en/v2.0/quickstart.html>

- a Wifi connection
- cmake ≥ 3.16 (+ ccmake)
- c++17 compatible compiler
(GCC > 8 or Xcode command line tools for MacOS)
- external MPI library
(OpenMPI, IntelMPI)
- git and git-lfs
- GitHub account

1. Go to <https://github.com/idefix-code/idefix>

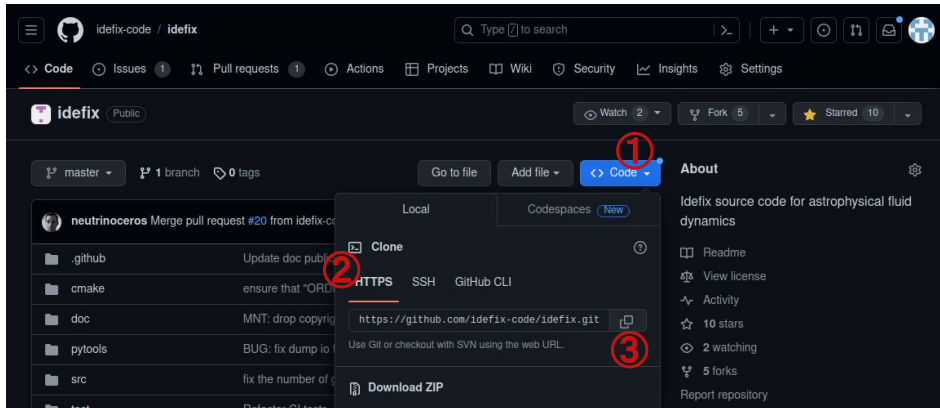
1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



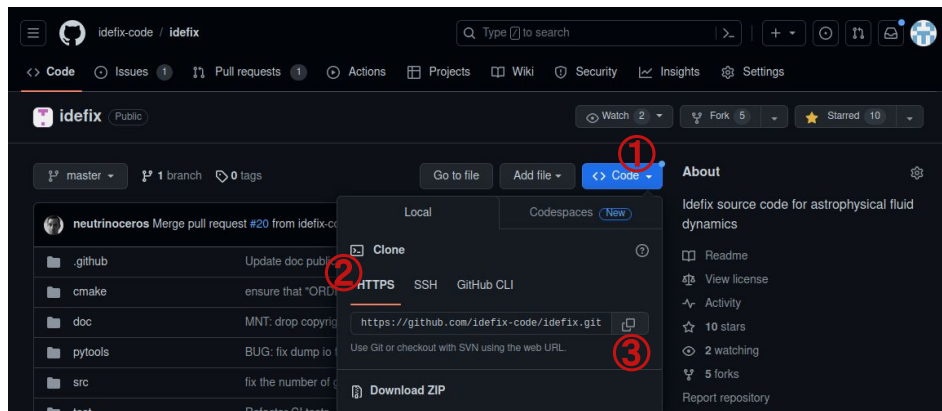
3. Click on <Code>, then <HTTPS> then copy the link (📄)



1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



3. Click on <Code>, then <HTTPS> then copy the link (📄)



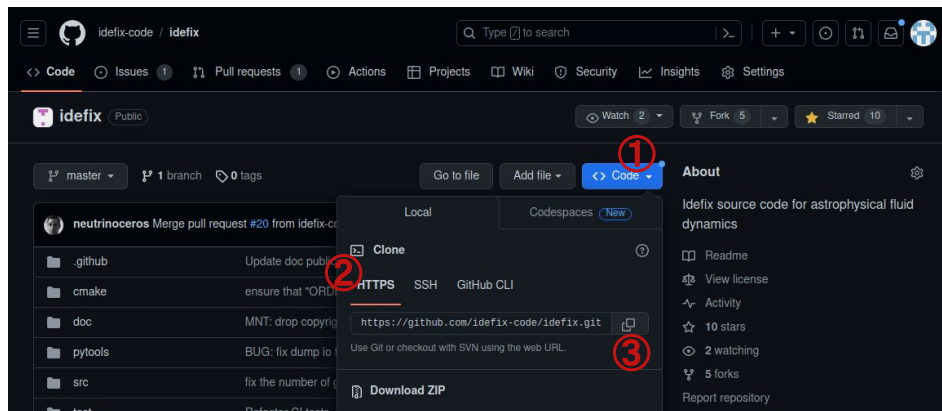
4. Go to your terminal, do e.g. `cd ~/src`, and type `git clone -b v2.0 https://github.com/idefix-code/idefix.git`

see Clément's tutorial (git), thursday p.m

1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



3. Click on <Code>, then <HTTPS> then copy the link (📄)



4. Go to your terminal, do e.g. `cd ~/src`, and type `git clone -b v2.0 https://github.com/idefix-code/idefix.git`
`cd idefix`

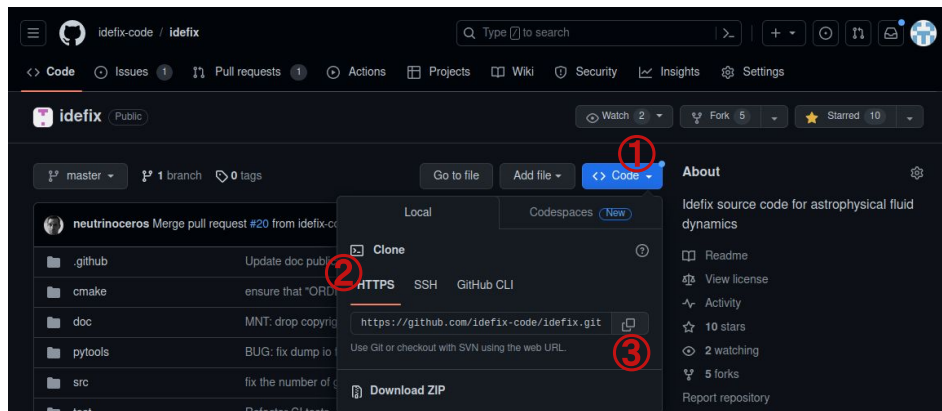
Idefix - GETTING STARTED

Download and install IDEFIX

1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



3. Click on <Code>, then <HTTPS> then copy the link (📄)



4. Go to your terminal, do e.g. `cd ~/src`, and type
`git clone -b v2.0 https://github.com/idefix-code/idefix.git`
`cd idefix`
`git submodule update --init`

Idefix relies on the [Kokkos](#) library, embedded as a submodule.

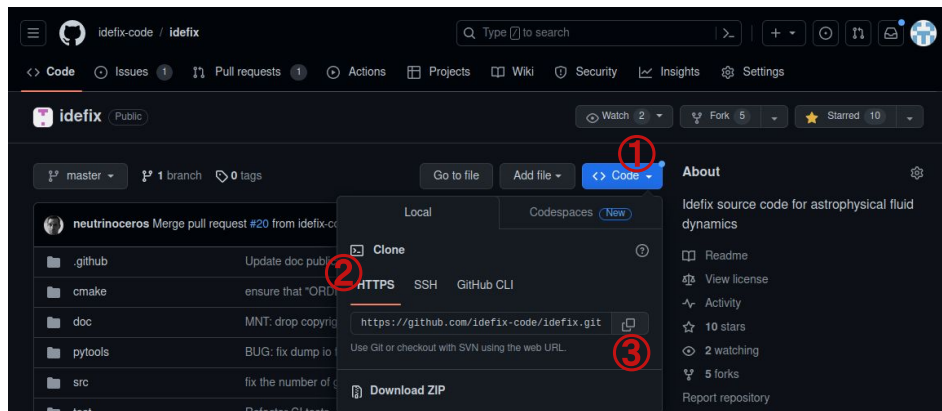
Idefix - GETTING STARTED

Download and install IDEFIX

1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



3. Click on <Code>, then <HTTPS> then copy the link (📄)



4. Go to your terminal, do e.g. `cd ~/src`, and type
`git clone -b v2.0 https://github.com/idefix-code/idefix.git`
`cd idefix`
`git submodule update --init`

Idefix relies on the [Kokkos](#) library, embedded as a submodule.

5. Idefix relies on the environment variable `$IDEFIX_DIR`
`export IDEFIX_DIR=~/.src/idefix`
You can add this line to your .profile or .bashrc

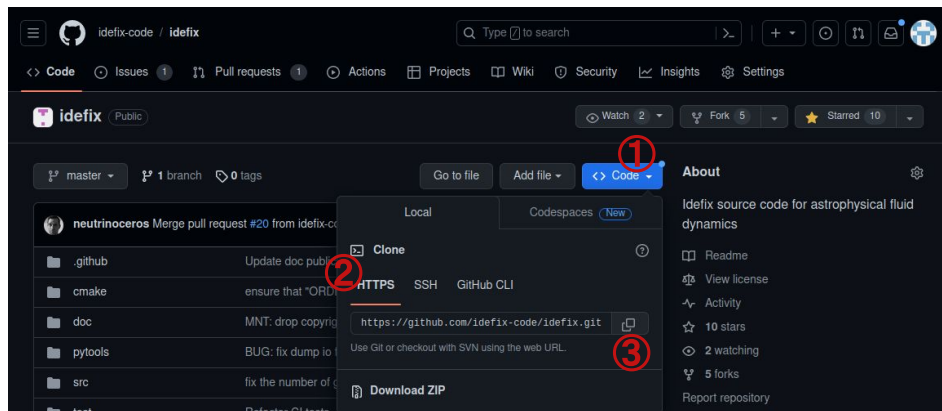
Idefix - GETTING STARTED

Download and install IDEFIX

1. Go to <https://github.com/idefix-code/idefix>
2. Sign in to your GitHub account



3. Click on <Code>, then <HTTPS> then copy the link (📄)



4. Go to your terminal, do e.g. `cd ~/src`, and type
`git clone -b v2.0 https://github.com/idefix-code/idefix.git`
`cd idefix`
`git submodule update --init`

Idefix relies on the [Kokkos](#) library, embedded as a submodule.

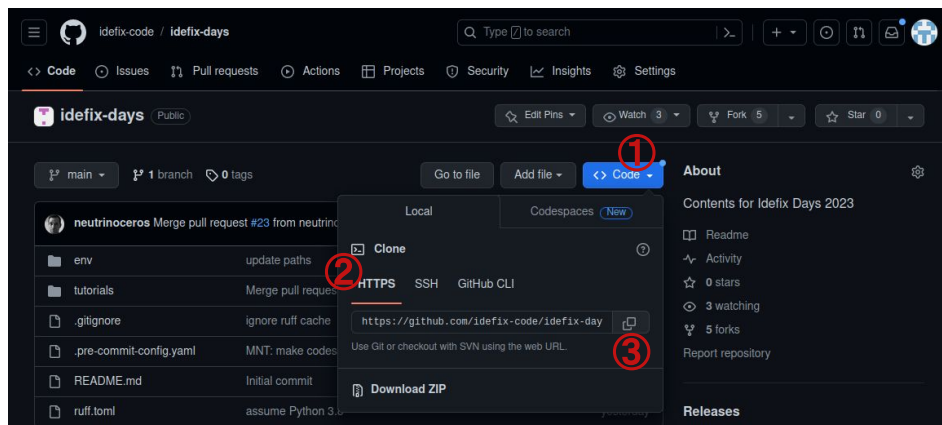
5. Idefix relies on the environment variable `$IDEFIX_DIR`
`export IDEFIX_DIR=~/.src/idefix`
You can add this line to your .profile or .bashrc

6. Validation for testing relies on large reference files
`git lfs install; git lfs pull`

Idefix - GETTING STARTED

Get access to all the tutorials idefix-days

1. Go to <https://github.com/idefix-code/idefix-days>
2. Click on <Code>, then <HTTPS> then copy the link (📄)



3. Go to your terminal, do e.g. `cd ~/src`, and type `git clone https://github.com/idefix-code/idefix-days.git`

```
cd idefix-days/tutorials/getting_started
```

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

Configure, compile and run test problems

Problem 1: OrszagTang

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

```
ccmake $IDEFIX_DIR
```

see Jonah's tutorial (Usage), this afternoon

Configure, compile and run test problems

Problem 1: OrszagTang

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             OFF
Idefix_MPI             OFF

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

Configure, compile and run test problems

Problem 1: OrszagTang

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             OFF
Idefix_MPI             OFF

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

Configure, compile and run test problems

Problem 1: OrszagTang

Idefix - GETTING STARTED

Configure, compile and run test problems

Problem 1: OrszagTang

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS  OFF
BUILD_TESTING      ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG        OFF
Idefix_DEFS          definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN  Default
Idefix_MHD           OFF
Idefix_MPI           OFF

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             OFF
Idefix_MPI             OFF

BUILD_SHARED_LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Configure, compile and run test problems

Problem 1: OrszagTang

Tip

Note that `cmake` keeps a cache of the previous configuration performed in a particular problem directory. To reset the configuration and start from scratch, delete the file `CMakeCache.txt`.

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             OFF
Idefix_MPI             OFF

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Configure, compile and run test problems

Problem 1: OrszagTang

SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS             definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             OFF
Idefix_MPI             OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Configure, compile and run test problems

Problem 1: OrszagTang

SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      *OFF
BUILD_TESTING          *ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   */usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           *OFF
Idefix_DEFS            *definitions.hpp
Idefix_HIGH_ORDER_FARGO *OFF
Idefix_LOOP_PATTERN    *Default
Idefix_MHD             *ON
Idefix_MPI             *OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS             definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD              OFF
Idefix_MPI              OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Configure, compile and run test problems

Problem 1: OrszagTang

SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      *OFF
BUILD_TESTING          *ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   */usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           *OFF
Idefix_DEFS             *definitions.hpp
Idefix_HIGH_ORDER_FARGO *OFF
Idefix_LOOP_PATTERN     *Default
Idefix_MHD              *ON
Idefix_MPI              *OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

ERROR - SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING           ON
CMAKE_BUILD_TYPE        *
CMAKE_INSTALL_PREFIX    /usr/local
Idefix_CXX_FLAGS         *
Idefix_DEBUG            *OFF
Idefix_DEFS             *definitions.hpp
Idefix_HIGH_ORDER_FARGO *OFF
Idefix_LOOP_PATTERN      Default
Idefix_MHD              *OFF
Idefix_MPI              *OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

COMPILE SHOULD FAIL !
→ MHD run

Configure, compile and run test problems

Problem 1: OrszagTang

SERIAL

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      *OFF
BUILD_TESTING           *ON
CMAKE_BUILD_TYPE        *
CMAKE_INSTALL_PREFIX    */usr/local
Idefix_CXX_FLAGS         *
Idefix_DEBUG            *OFF
Idefix_DEFS             *definitions.hpp
Idefix_HIGH_ORDER_FARGO *OFF
Idefix_LOOP_PATTERN      *Default
Idefix_MHD              *ON
Idefix_MPI              *OFF
```

```
BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure      Press [g] to generate and exit
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`./idefix`

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

PARALLEL - MPI library

`rm CMakeCache.txt`

Configure, compile and run test problems

Problem 1: OrszagTang

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG            OFF
Idefix_DEFS             definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD              ON
Idefix_MPI              ON

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

ENTER
ENTER

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Idefix - GETTING STARTED

Well-known test problem for MHD codes

`cd OrszagTang`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             ON
Idefix_MPI             ON

BUILD_SHARED_LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help      Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Configure, compile and run test problems

Problem 1: OrszagTang

PARALLEL - openmp

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 2 of 13
Idefix_MHD             *ON
Idefix_MPI             *OFF
Idefix_PRECISION       *Double
Idefix_RECONSTRUCTION  *Linear
Idefix_WERROR          *OFF
Kokkos_ARCH_A64FX      *OFF
Kokkos_ARCH_AMD64      *OFF
Kokkos_ARCH_ARMV8      *OFF
Kokkos_ENABLE_OPENMP   *ON
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help      Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

`export OMP_NUM_THREADS=2`

3. Run

`./idefix`

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

*will be used in my tutorial (visualization/nonos),
tomorrow afternoon*

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             ON
Idefix_MPI             ON
BUILD_SHARED_LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help      Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

ENTER

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS             definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD              ON
Idefix_MPI              ON

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

“Automatic domain decomposition requires nx1, nx2 and nx3 to be powers of 2. Alternatively, set a manual decomposition with -dec”

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             ON
Idefix_MPI             ON

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

“Automatic domain decomposition requires nx1, nx2 and nx3 to be powers of 2. Alternatively, set a manual decomposition with -dec”

```
[Grid]
X1-grid  1  0.42  192  u  2.14
X2-grid  1  0.0   768  u  6.283185307179586
X3-grid  1 -0.0125  1    u  0.0125
```

idefix.ini

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             ON
Idefix_MPI             ON
```

ENTER

BUILD SHARED LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

“Automatic domain decomposition requires nx1, nx2 and nx3 to be powers of 2. Alternatively, set a manual decomposition with -dec”

```
[Grid]
X1-grid  1  0.42  192  u  2.14
X2-grid  1  0.0   768  u  6.283185307179586
X3-grid  1 -0.0125  1    u  0.0125
```

idefix.ini

-dec n1 n2 n3 : **domain decomposition**

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN     Default
Idefix_MHD             ON
Idefix_MPI             ON
```

ENTER

```
BUILD_SHARED_LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

“Automatic domain decomposition requires nx1, nx2 and nx3 to be powers of 2. Alternatively, set a manual decomposition with -dec”

```
[Grid]
X1-grid  1  0.42  192  u  2.14
X2-grid  1  0.0   768  u  6.283185307179586
X3-grid  1 -0.0125  1    u  0.0125
```

idefix.ini

-dec n1 n2 n3 : **domain decomposition**

→ should be equal to **DIMENSIONS**

```
#define COMPONENTS 3      definitions.hpp
#define DIMENSIONS 2
#define ISOTHERMAL
#define GEOMETRY POLAR
```

see Jonah's tutorial (Usage), this afternoon

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

PARALLEL - MPI library

`rm CMakeCache.txt`

1. Configure :

`ccmake $IDEFIX_DIR`

```
Page 1 of 11
BUILD_SHARED_LIBS      OFF
BUILD_TESTING          ON
CMAKE_BUILD_TYPE
CMAKE_INSTALL_PREFIX   /usr/local
Idefix_CXX_FLAGS
Idefix_DEBUG           OFF
Idefix_DEFS            definitions.hpp
Idefix_HIGH_ORDER_FARGO OFF
Idefix_LOOP_PATTERN    Default
Idefix_MHD             ON
Idefix_MPI             ON
```

ENTER

```
BUILD_SHARED_LIBS: Build shared libraries
Press [enter] to edit option Press [d] to delete an entry
Press [c] to configure
Press [h] for help          Press [q] to quit without generating
Press [t] to toggle advanced mode (Currently Off)
```

Press [c], then [c] to configure and [g] to generate

2. Compile

`make -j 8`

3. Run

`mpirun -np 2 ./idefix -dec 2 1`

Configure, compile and run test problems

Problem 2: PlanetMigration2D

“Automatic domain decomposition requires nx1, nx2 and nx3 to be powers of 2. Alternatively, set a manual decomposition with -dec”

```
[Grid]
X1-grid  1  0.42  192  u  2.14
X2-grid  1  0.0   768  u  6.283185307179586
X3-grid  1 -0.0125  1    u  0.0125
```

idefix.ini

-dec n1 n2 n3 : domain decomposition

→ should be equal to **DIMENSIONS**

```
#define COMPONENTS 3
#define DIMENSIONS 2
#define ISOTHERMAL
#define GEOMETRY POLAR
```

definitions.hpp

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Test and validation

Problem 2: PlanetMigration2D

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Test and validation

Problem 2: PlanetMigration2D

Most of the tests in the test/ directory can be validated against:

- analytical solution (standard test)
- pre-computed solutions (non-regression tests).

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Test and validation

Problem 2: PlanetMigration2D

Most of the tests in the test/ directory can be validated against:

- analytical solution (standard test)
- pre-computed solutions (non-regression tests).

RULE: All the tests **HAVE TO** pass to merge a pull-request.

It is therefore a good practice to run locally the tests before creating a pull-request!

Example: If Clément implements a new module to simulate Lagrangian particles (*see his talk tomorrow morning*), all the pre-existing tests in IDEFIX have to pass in order for his module to be merged!

Idefix - GETTING STARTED

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Test and validation

Problem 2: PlanetMigration2D

Most of the tests in the test/ directory can be validated against:

- analytical solution (standard test)
- pre-computed solutions (non-regression tests).

RULE: All the tests **HAVE TO** pass to merge a pull-request.

It is therefore a good practice to run locally the tests before creating a pull-request!

Example: If Clément implements a new module to simulate Lagrangian particles (**see his talk tomorrow morning**), all the pre-existing tests in IDEFIX have to pass in order for his module to be merged!

```
python -m pip install -r $IDEFIX_DIR/test/python_requirements.txt
```

Problem 2: PlanetMigration2D

Test problem Planet/Disk interaction

`cd PlanetMigration2D`

Most of the tests in the test/ directory can be validated against:

- analytical solution (standard test)
- pre-computed solutions (non-regression tests).

```
Running standard test...
diff_torque=1.726052e-19
Error_dist=0.000000e+00
SUCCESS!
Standard test succeeded
Non-regression test succeeded with error=2.809121e-15
```

RULE: All the tests **HAVE TO** pass to merge a pull-request.

It is therefore a good practice to run locally the tests before creating a pull-request!

Example: If Clément implements a new module to simulate Lagrangian particles (**see his talk tomorrow morning**), all the pre-existing tests in IDEFIX have to pass in order for his module to be merged!

```
python -m pip install -r $IDEFIX_DIR/test/python_requirements.txt
```

1. `./testme.py -all` Full validation of a test (**configure/compile/run/test** various combination of algorithms)
2. `./testme.py` Partial validation of a test (**run/test** various input files)
3. `./testme.py -check` Validation of a test that has already been run

GETTING STARTED

Idefix

PART 1: ON YOUR LAPTOP → this presentation

PART 2: ON BIGFOOT (GPU) → Geoffroy's presentation

<https://idefix.readthedocs.io/en/v2.0/quickstart.html>