Very short Guide
to install *Code_Saturne* and
to run a simulation

1 – Installation

A Version of *Code_Saturne* supporting offload to GPU is available from CS_5.3_PRACE_UEABS.tar.gz.

The archive contains a foldfer CS_5.3_PRACE_UEABS, with the installer InstallHPC.sh (executable) and the folder code_saturne-5.3_GPU_VERSION_UEABS which has the source code.

An example of the last lines of the installer (meant for INTEL compiler & MPI in this example) reads:

```
$KERSRC/configure \
--host=x86_64 \
--disable-shared \
--enable-openmp \
--disable-cuda-offload \
--without-modules \
--disable-rpath \
--disable-dlloader \
--disable-gui \
--disable-sockets \
--with-libxml2=no \
--enable-long-gnum \
--disable-mei \
--prefix=$KEROPT \
CC=mpicc FC=mpif90 CXX=mpicxx CFLAGS="-O3 -xCORE-AVX512 -Ofast" FCFLAGS="-O3 -xCORE-AVX512
-Ofast" CXXFLAGS="-std=gnu++0x"
#
make -j 8
make install
```

You might have to adapt CC, FC, CFLAGS, FCFLAGS, LDFLAGS and LIBS for your machine, compilers, MPI installation and also for GPU utilisation, either with the option --disable-cuda-offload (CPU only) or --enable-cuda-offload (CPU and GPU).

To install the code, type, assuming that Y_P is the path where CS_5.3_PRACE_UEABS was copied:

```
cd Y_P/ CS_5.3_PRACE_UEABS
./InstallHPC.sh
```

When the code is installed the command **code_saturne** should return, when typing:

```
Y_P/CS_5.3_PRACE_UEABS/code_saturne-5.3_GPU_VERSION_UEABS/arch/Linux/bin/code_saturne
```

```
Usage:      Y_P/CS_5.3_PRACE_UEABS/code_saturne-5.3_GPU_VERSION_UEABS/arch/Linux/bin/code_saturne
<topic>

Topics:
  help
  studymanager
  smgr
  bdiff
  bdump
  compile
  config
  create
  gui
  studymanagergui
  smgrgui
  trackcvg
  info
  run
  salome
  submit

Options:
  -h, --help  show this help message and exit
```

2 – Preparing the simulation

Two archives are used, namely CS_5.3_PRACE_UEABS_CAVITY_13M.tar.gz and CS_5.3_PRACE_UEABS_CAVITY_111M.tar.gz that contain the information required to run both test cases, with the ***mesh_input*** file (for the mesh) and the usersubroutines in src_saturne.

From the working directory WORKDIR (pick another one than CS_5.3_PRACE_UEABS), you need to create a 'study' (Cavity13M, for instance) and a 'case' (MACHINE, for instance) as:

```
Y_P/CS_5.3_PRACE_UEABS/code_saturne-5.3_GPU_VERSION_UEABS/arch/Linux/bin/code_saturne create
--study Cavity13M MACHINE
```

The Cavity13M directory contains 3 directories, MACHINE, MESH and POST.

The directory MACHINE contains 4 directories, DATA, RESU, SCRIPTS and SRC.

The file ***mesh_input*** should be copied into the MESH directory.

The user subroutines (cs_user* files) contained in src_saturne should be copied into SRC.

The cs_user_scripts.py file is used to manage the simulation. It has to be copied to DATA from DATA/REFERENCE as:

```
cd DATA
cp REFERENCE/cs_user_scripts.py .
```

At Line 138 of this file, you need to change from None to the local path of the mesh, i.e. "../MESH/mesh_input"

To finalise the preparation go to the folder MACHINE and type:

Y_P/CS_5.3_PRACE_UEABS/code_saturne-5.3_GPU_VERSION_UEABS/arch/Linux/bin/code_saturne run --initialize

This should create a folder RESU/YYYYMMDD-HHMM, which should contain the following flles:

```
cs_user_scripts.py
src_saturne
cs_solver
compile.log
mesh_input
summary
run_solver
```

3 – Running the simulation

The name of the executable is ./cs_solver, and you should run it with the option –mpi as

mpirun/mpiexec/poe/aprun/… ……. ./cs_solver --mpi

References:

http://www.code-saturne.org
http://www.fz-juelich.de/ias/jsc/EN/Expertise/High-Q-Club/_node.html
https://github.com/code-saturne/code_saturne/pull/22