Menu

Top

Richardson Cluster Materials

Forum

Resources

Manuals/Instructions

In-house codes

Immersed Boundary Code

Multicomponent Flow Code (MFC)

Public softwares/libraries

Miscellaneous Document

Other useful information

Multicomponent Flow Code (MFC)

Overview

Multicomponent flow code (MFC) is an in-house code to simulate compressible multi-component flows.

The code was initially developed by Dr. Vedran Coralic and succeeded by Dr. Jomela Meng, both of whom are former students of the group. The code has been updated by succeeding members.

The latest version of the code is ver_3.0.

The source code is provided on your request, those who work on relevant projects.

Current developers: Kazuki Maeda, Kevin Schmidmayer

Installation

The code is composed of folders associated with master-scripts, pre-process, simulation and post-process.

The source codes are written in fortran90. Scripting is done by python.

For post-processing of simulation data, the code employs silo-hdf5 format, which is aimed for parallel visualization using VisIt.

To use silo-HDF5 format, a silo library linked with HDF5 is required.

The procedure for compiling/installation of HDF5, silo and VisIt is not non-trivial, but highly depends on your environment.

Here we list instructions for the installation of some of key components in workstation/cluster available for the group.

Before going to the following steps, one advice is to add the following to ~/.bash_profile: module load intel and module load mpich. Then source .bash_profile using source ~/.bash_profile.

HDF5 on Richardson

Richardson works on RHEL7, thus pre-built distribution of HDF5 for CentOS7 can be directly used without modification.

1. Download pre-built distribution of HDF5 from http://www.hdfgroup.org/ftp/HDF5/current/bin/linux-centos7-x86_64-gcc485/hdf5-1.8.17-linux-centos7-x86_64-gcc485-shared.tar.gz
2. Untar the archive using tar -zxf [your HDF5 archive]

Silo on Richardson

1. Download Silo 4.10.2 from https://wci.llnl.gov/simulation/computer-codes/silo/downloads and move it to your home directory on Richardson
2. Untar the archive using tar -zxf silo-4.10.2.tar.gz
3. Change directory to the silo directory
4. Run the configure script using the following command:./configure --prefix=[target installation directory] --enable-pythonmodule --enable-optimization --disable-hzip --disable-fpzip --enable-portable-binary FC=mpif90 F77=mpif77 -with-hdf5=[your hdf5 directory]/include,/[your hdf5 directory]/lib --disable-silex
5. Finish the installation by running make install
6. Add the following to ~/.bash_profile: export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$HOME/[your silo directory]/lib:/[your hdf5 directory]/lib
7. Source .bash_profile using source ~/.bash_profile

VisIt on Richardson (version 2.9.0 is confirmed to work)

1. Download makedepend-1.0.5 from http://www.linuxfromscratch.org/blfs/view/7.4/x/makedepend.html and move it to your home directory on Richardson
2. Untar the archive using tar xvfj makedepend-1.0.5.tar.bz2
3. Change directory to the makedepend-1.0.5 directory and configure the installation with the following: ./configure --prefix=[your makedepend directory] $XORG_CONFIG &&
4. make
5. make install
6. Add makedepend-1.0.5 PATH to ~/.bash_profile
7. Add the following line to ~/.bash_profile: export PATH=$PATH:$HOME/.local/bin:$HOME/bin:/opt/intel/impi/5.1.3.181/intel64/bin:/opt/intel/bin
8. Add the following 2 lines to ~/.bashrc:
9. PATH=$PATH:/opt/intel/impi/5.1.3.181/intel64/bin
10. PATH=$PATH:/opt/intel/bin
11. Source .bash_profile and .bashrc using source ~/.bash_profile ~/.bashrc
12. In your home directory on Richardson mkdir visit/
13. Change directory to the visit directory you just created
14. wget http://portal.nersc.gov/project/visit/releases/2.9.0/build_visit2_9_0
15. svn co http://visit.ilight.com/svn/visit/trunk/src/svn_bin/bv_support/
16. chmod 755 build_visit2_9_0
17. set the following variable in build_visit2_9_0 SVN_ANON_ROOT_PATH="http://visit.ilight.com/svn/visit"
18. Run the build_visit script using the following: ./build_visit2_9_0 --console --silo --parallel --hdf5 --szip --mesa
19. Modify ~/.bash_profile as the following: export PATH=$PATH:$HOME/.local/bin:$HOME/bin:/opt/intel/impi/5.1.3.181/intel64/bin:/opt/intel/bin:$HOME/[your visit directory]/visit2.9.0/src/bin
20. source ~/.bash_profile
21. Add the following line to ~/.bashrc: export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$HOME/[your visit directory]/visit/vtk/6.1.0/linux-x86_64_gcc-4.8/lib
22. source ~/.bashrc

Paraview on Richardson

1. The last versions of Paraview have to be installed by an administrator of the cluster. If the version of Paraview that you want is not already installed, ask one of the admins to follow these steps. If it is already installed, go directly to point **12**. Note that the last versions of Paraview do need to install last versions of Qt (check the Paraview web site to be aware of the current needed version).
2. Download the Source version of Paraview that you want to work with ([Nom de votre Paraview].tar.gz) and move it to the /opt/src/ directory on Richardson.
3. Untar the archive using tar xvzf [Nom de votre Paraview].tar.gz
4. cd [Nom de votre Paraview]
5. mkdir build
6. cd build
7. ccmake ..
8. Once in the ccmake page, do c to configure. If the configuration ask you to use an updated version of ccmake, you have to install a new one and then use this new one instead.
9. When the first configure has succeed, activate the MPI option (find it in the ccmake page and turn it on) and change the install prefix to the /opt/ directory.
10. Then type c again to configure another time. If a warning shows up, you would have to configure a last time and finally type g to generate the install when it will be available.
11. make install
12. Once install, use a sbatch script to run Paraview in parallel. The command to write in a pvserver.q is thus:
13. mpirun /opt/Paraview/bin/pvserver --use-offscreen-rendering with the updated path to Paraview.
14. Now you just have to run Paraview using the script on Richardson: sbatch pvserver.q
15. Check the output to see on which node is running your Paraview and then on your local computer use the command:
16. ssh -X -N -L localhost:11111:node[Node number].cluster:11111 [Your profil]@richardson.caltech.edu
17. Run the same version of Paraview on your computer and connect you on your localhost server (default server when you add one in Paraview).

Silo on Comet

1. Download silo-4.7.2.tar.gz from website
2. Untar the zip file
3. Make sure to load the hdf5 module by typing "module load hdf5"
4. In the silo folder, use the following configure command (changing your prefix path): ./configure --prefix=/home/jcmeng/silo-4.7.2/ --enable-pythonmodule --enable-optimization --disable-hzip --disable-fpzip --enable-portable-binary --with-hdf5=/opt/hdf5/intel/mvapich2_ib/include,/opt/hdf5/intel/mvapich2_ib/lib CC=mpicc CXX=mpicxx FC=mpif90 F77=mpif77 --disable-silex
5. Finish by typing "make install"
6. Modify your post_process makefile to point to the silo directory in your home folder

Silo on Hooke (HDF5 is built in; no need for custom installation)

1. Download Silo 4.7.2 from https://wci.llnl.gov/simulation/computer-codes/silo/downloads and move it to your home directory on Hooke
2. Untar the archive using tar -zxf silo-4.7.2.tar.gz
3. Change directory to the silo-4.7.2 directory
4. Run the configure script using the following command:./configure --enable-pythonmodule --enable-optimization --disable-hzip --disable-fpzip --enable-portable-binary FC=/opt/mvapich2/ch3_mrail_gen2-intel12/bin/mpif90 F77=/opt/mvapich2/ch3_mrail_gen2-intel12/bin/mpif77 –with-hdf5=/opt/visit/hdf5/1.8.7/linux-x86_64_gcc-4.1/include,/opt/visit/hdf5/1.8.7/linux-x86_64_gcc-4.1/lib
5. Finish the installation by running make install
6. Add the following to ~/.bash_profile: export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/opt/torque/lib:$HOME/silo-4.7.2/lib:/opt/visit/hdf5/1.8.7/linux-x86_64_gcc-4.1/lib

Menu

Top

Richardson Cluster Materials

Forum

Resources

Manuals/Instructions

In-house codes

Immersed Boundary Code

Multicomponent Flow Code (MFC)

Public softwares/libraries

Miscellaneous Document

Other useful information

7. Source .bash_profile using source ~/.bash_profile

VisIt on Hooke (version 2.6.1 and 2.9.0 are confirmed to work on Hooke)

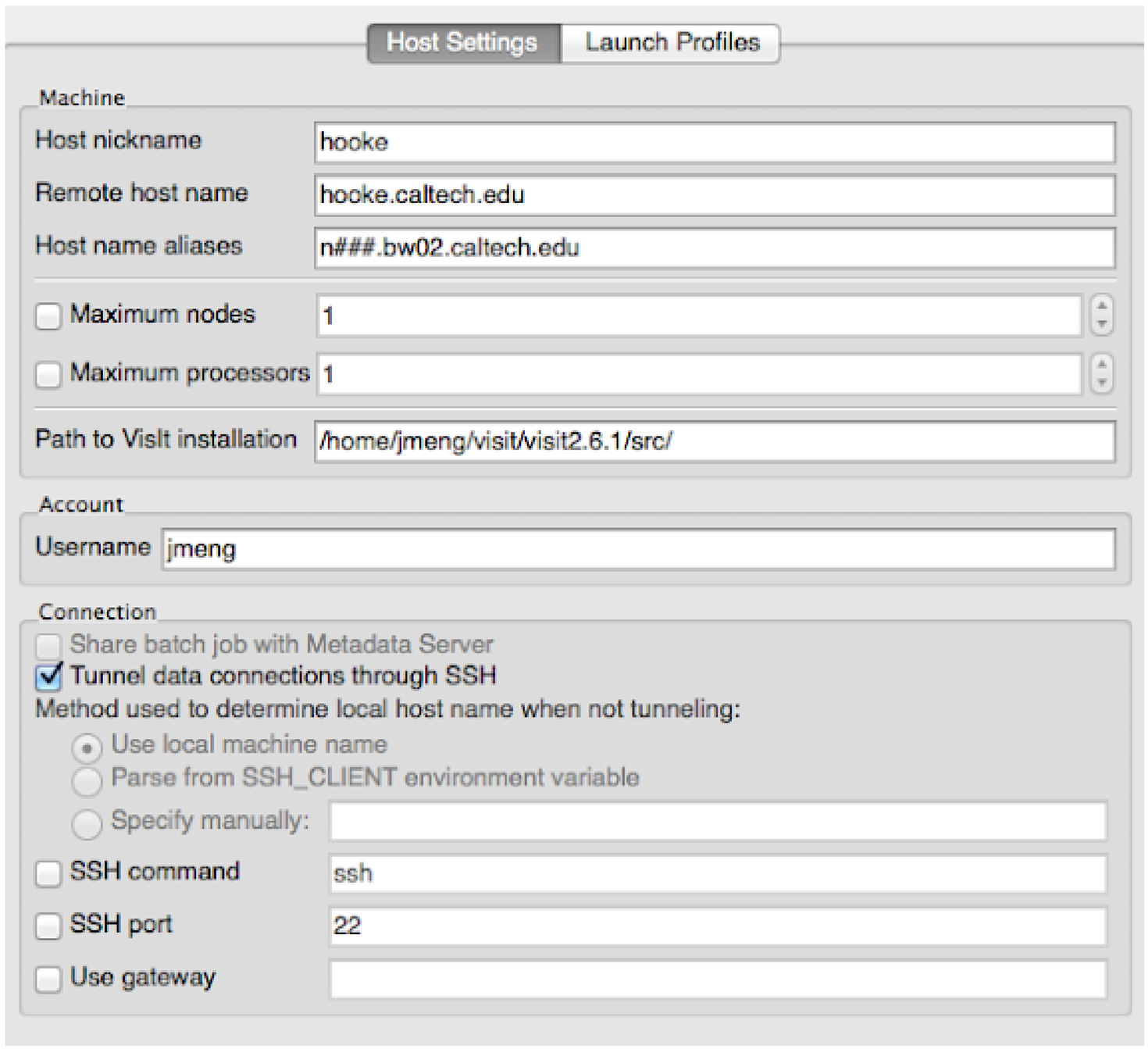
1. In your home directory on Hooke mkdir visit/
2. Change directory to the visit directory you just created
3. svn cat http://portal.nersc.gov/svn/visit/trunk/releases/2.6.1/build_visit2_6_1 > build_visit
4. svn co http://portal.nersc.gov/svn/visit/trunk/src/svn_bin/bv_support (Important (2016/03/25): the address was moved to http://visit.ilight.com/svn/visit/trunk/src/svn_bin/bv_support/)
5. chmod 755 build_visit
6. Run the build_visit script using the following: ./build_visit --console --silo --parallel --hdf5 --szip (Important (2016/03/25): due to the change in bv_support address, you need to set SVN_ANON_ROOT_PATH="http://visit.ilight.com/svn/visit" in the script before running.)
7. In the current directory, open the file **hooke.bw02.caltech.edu.cmake** and make the following changes:
    1. Under line 12 which is SET(VISITARCH linux-x86_64_gcc-4.1)add the following 3 lines:
        1. SET(VISIT_MPI_CXX_FLAGS -I/opt/open-mpi/ib-gnu41/include)
        2. SET(VISIT_MPI_LD_FLAGS -L/opt/open-mpi/ib-gnu41/lib)
        3. SET(VISIT_PARALLEL_CXXFLAGS -I/opt/open-mpi/ib-gnu41/include -lmpi_cxx)
    2. Change the new line 29 (after the addition of 3 lines above) to: VISIT_OPTION_DEFAULT(VISIT_MPI_COMPILER /opt/open-mpi/ib-gnu41/bin/mpic++ TYPE FILEPATH)
    3. Change the new line 74 to: VISIT_OPTION_DEFAULT(VISIT_HDF5_LIBDEP ${VISITHOME}/szip/2.1/${VISITARCH}/lib sz /home/jmeng/visit/visit/hdf5/1.8.7/linux-x86_64_gcc-4.1/lib TYPE STRING)
    4. Save and close the file
8. Change directory to ./visit2.6.1/src/
9. /home/jmeng/visit/cmake-2.8.8/bin/cmake -DVISIT_PARALLEL:BOOL=ON ./
10. sh recmake_visit.sh
11. make
12. Add the following to ~/.bash_profile: export PATH=$PATH:$HOME/bin:$HOME/visit/visit2.6.1/src/bin:/opt/open-mpi/ib-gnu41/bin
13. Source .bash_profile using source ~/.bash_profile

VisIt on Heaviside (version 2.9.0 is confirmed to work on Heaviside, HDF5 and Silo are built-in, so no need for custom installation)

1. Add the following line to ~/.bash_profile: export PATH=$PATH:$HOME/.local/bin:$HOME/bin:/opt/intel/impi/5.0.3.048/intel64/bin:/opt/intel/bin:$HOME/visit/visit2.9.0/src/bin
2. Add the following 2 lines to ~/.bashrc:
3. PATH=$PATH:/opt/intel/impi/5.0.3.048/intel64/bin
4. PATH=$PATH:/opt/intel/bin
5. Source .bash_profile and .bashrc using source ~/.bash_profile ~/.bashrc
6. In your home directory on Heaviside mkdir visit/
7. Change directory to the visit directory you just created
8. wget http://portal.nersc.gov/project/visit/releases/2.9.0/build_visit2_9_0
9. svn co http://portal.nersc.gov/svn/visit/trunk/src/svn_bin/bv_support (Important (2016/03/25): the address was moved to http://visit.ilight.com/svn/visit/trunk/src/svn_bin/bv_support/)
10. chmod 755 build_visit
11. Important (2016/03/25): due to the change in bv_support address, you need to set SVN_ANON_ROOT_PATH="http://visit.ilight.com/svn/visit" in the script before running.
12. Run the build_visit script using the following: ./build_visit2_9_0 --console --silo --parallel --hdf5 --szip --mesa

Configuring Host Setting/Launch Profiles of VisIt on personal machine

1. Download the applicable executable from https://wci.llnl.gov/simulation/computer-codes/visit/executables
    1. For Hooke, you'll want version 2.6.1
    2. For Heaviside, you'll want version 2.9.0
    3. For Richardson, you'll want version 2.9.0
2. Install VisIt as you would any other software program
3. Launch VisIt after installation and set up the following host profiles (**Options --> Host Profiles...**):
    1. Hooke:



Menu

Top

Richardson Cluster Materials

Forum

Resources

Manuals/Instructions

In-house codes

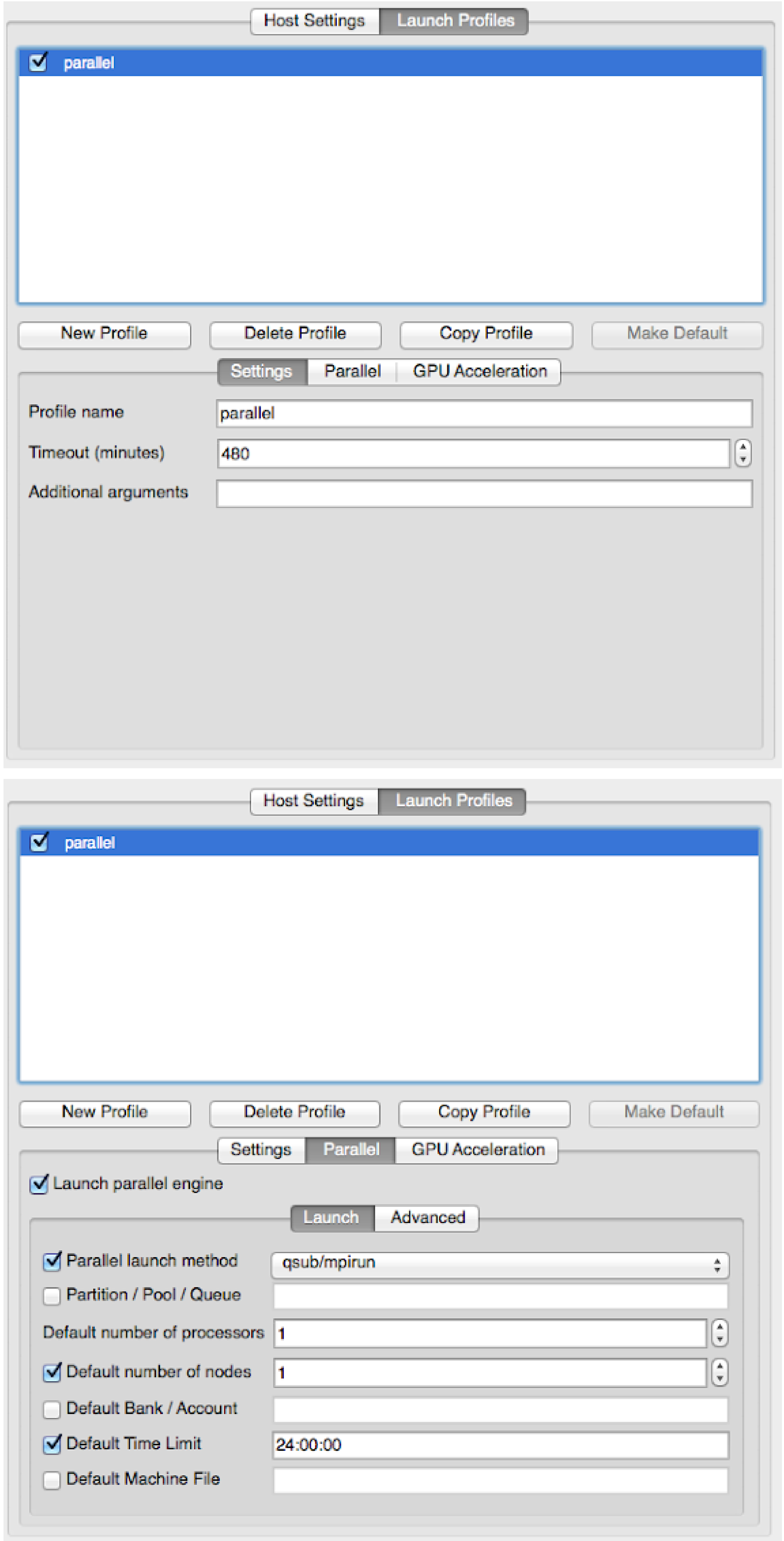
Immersed Boundary Code

Multicomponent Flow Code (MFC)

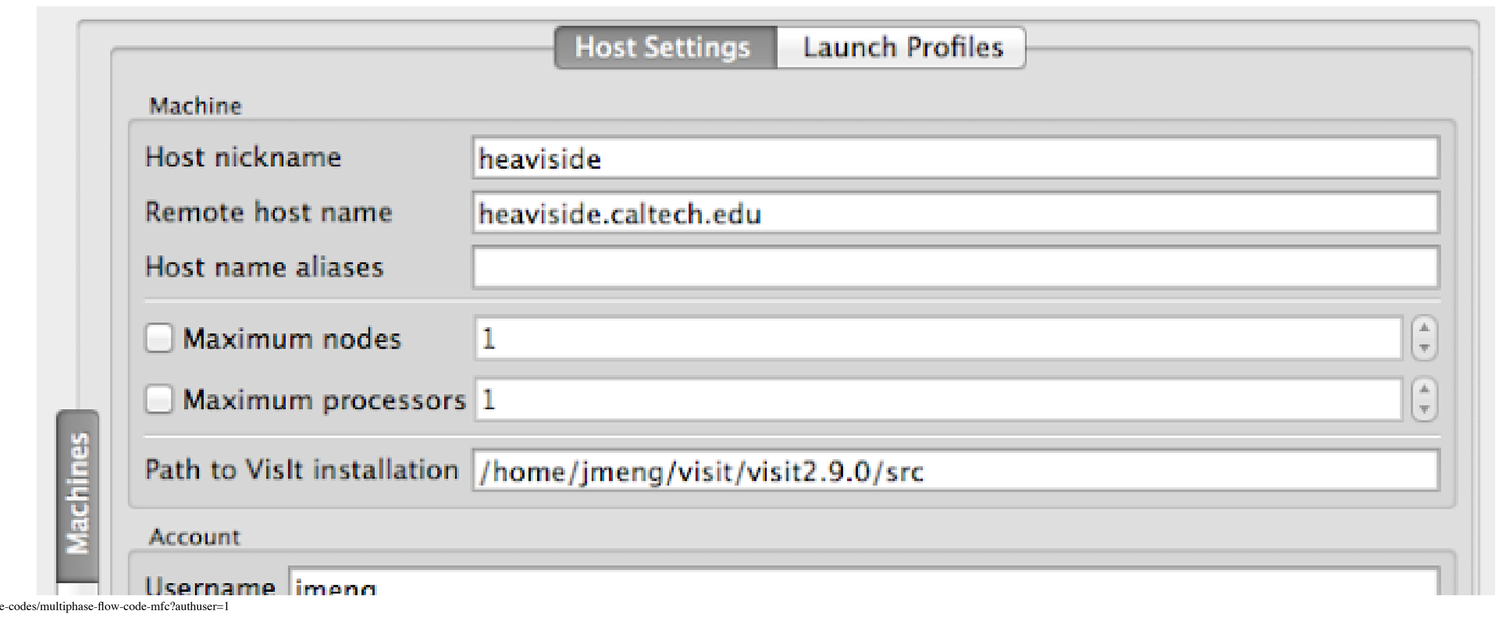
Public softwares/libraries

Miscellaneous Document

Other useful information



1. Heaviside:



Menu

Top

Richardson Cluster Materials

Forum

Resources

Manuals/Instructions

In-house codes

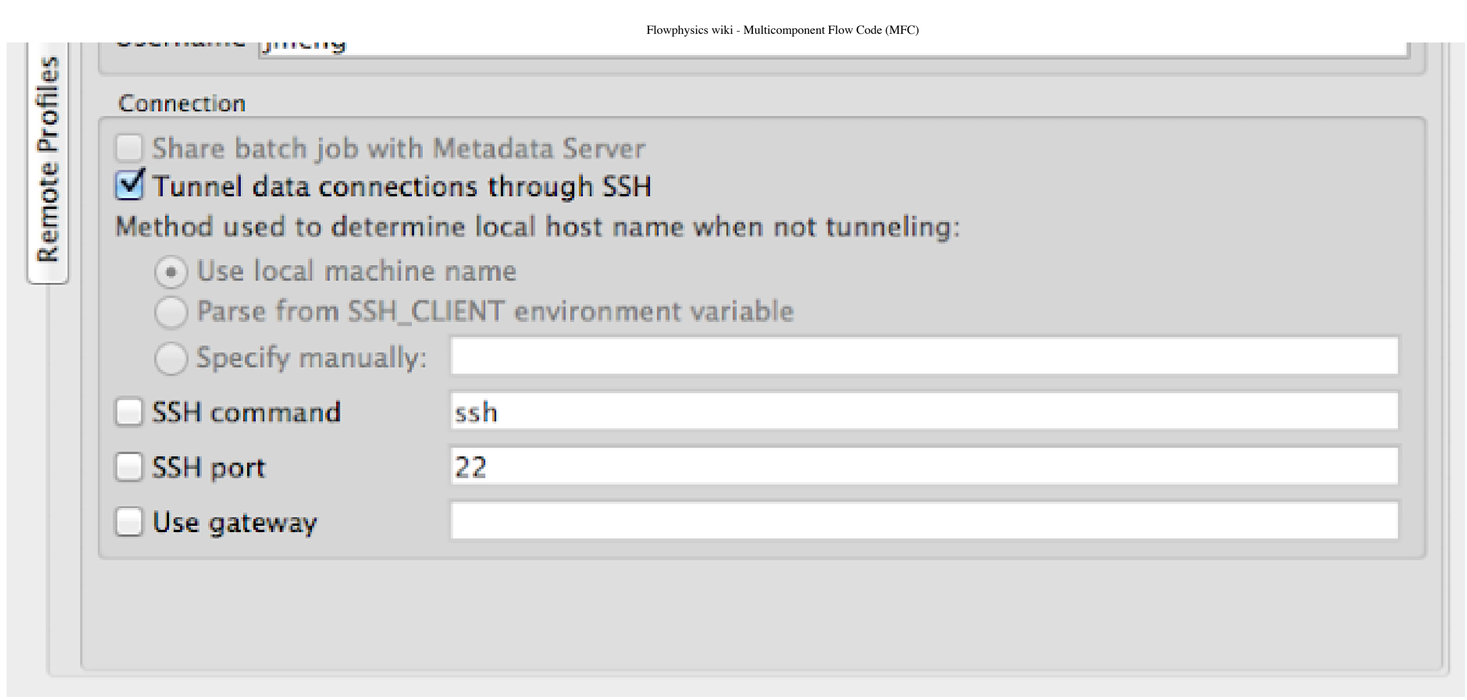
Immersed Boundary Code

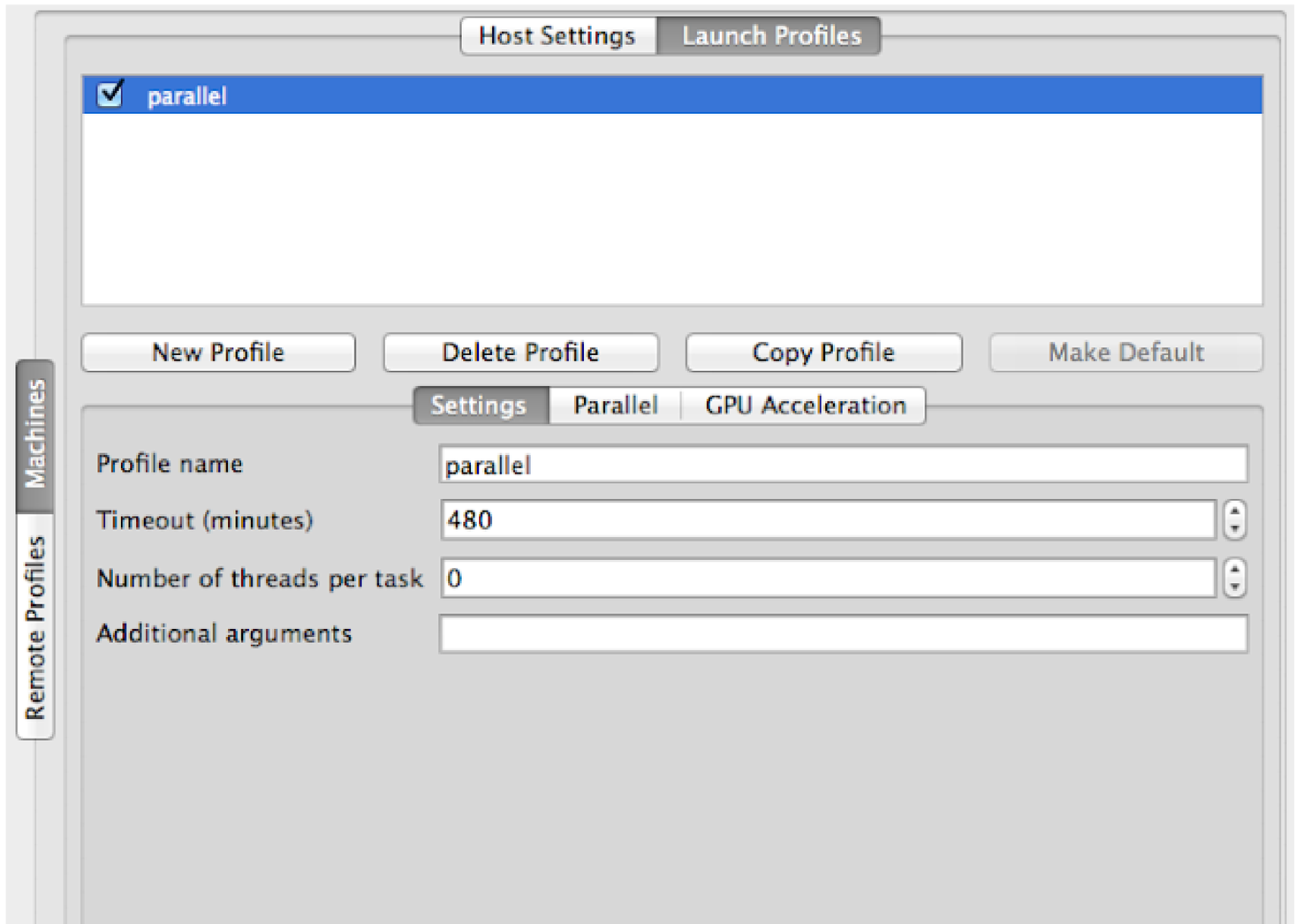
Multicomponent Flow Code (MFC)

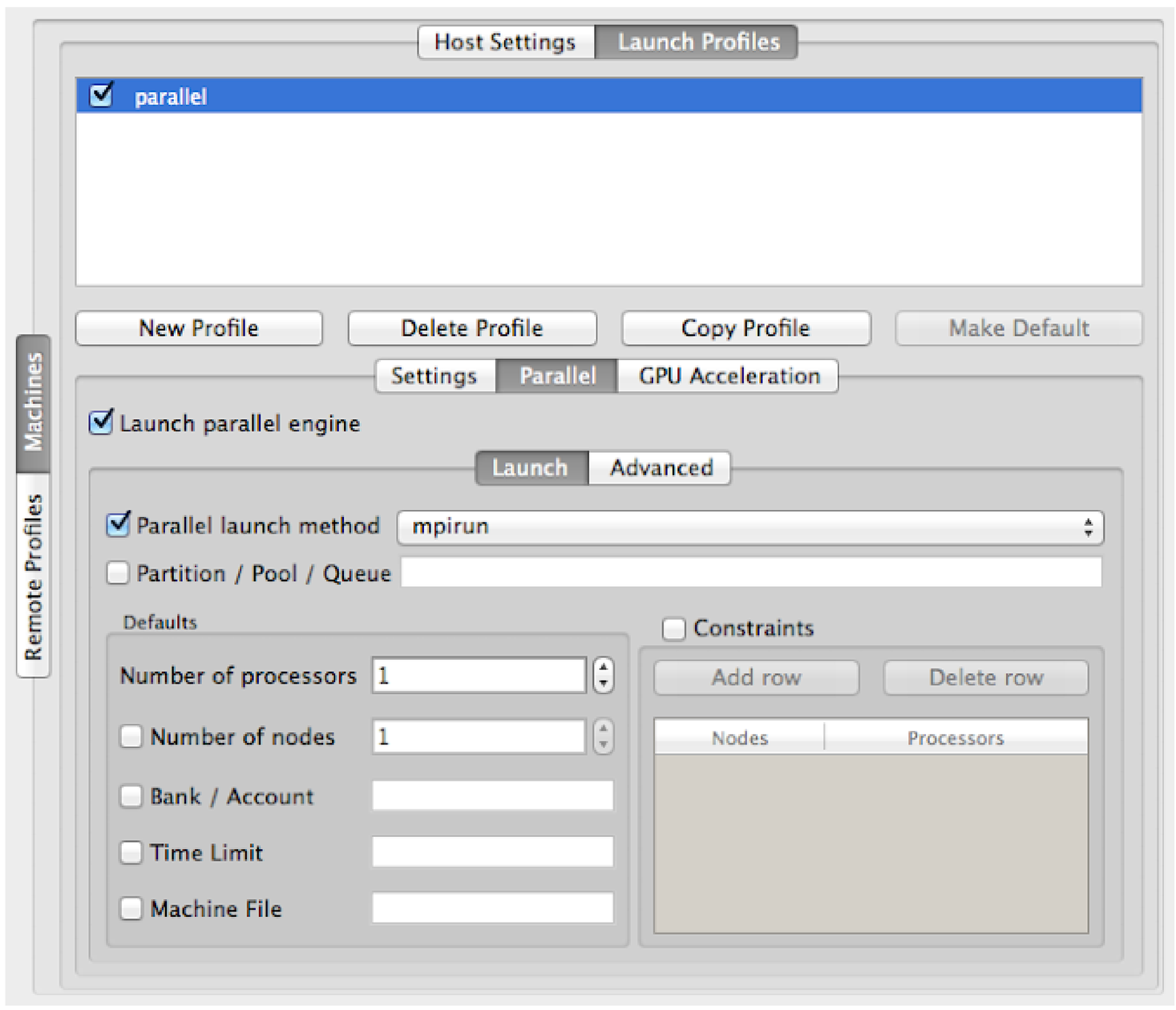
Public softwares/libraries

Miscellaneous Document

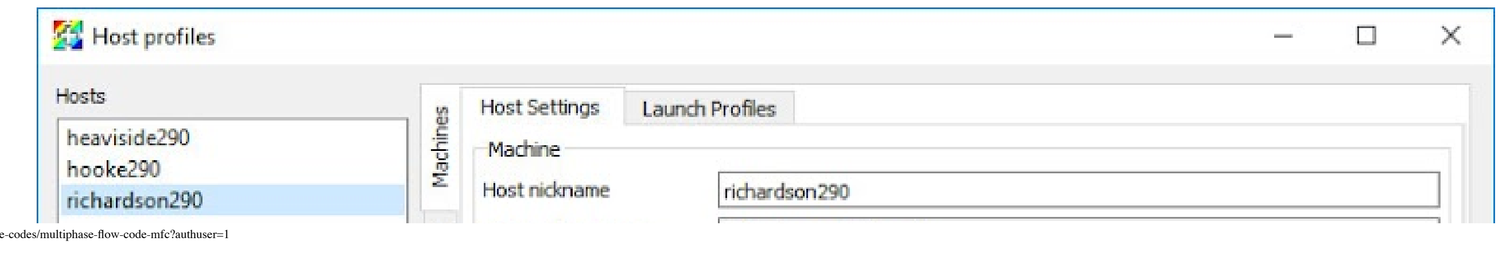
Other useful information

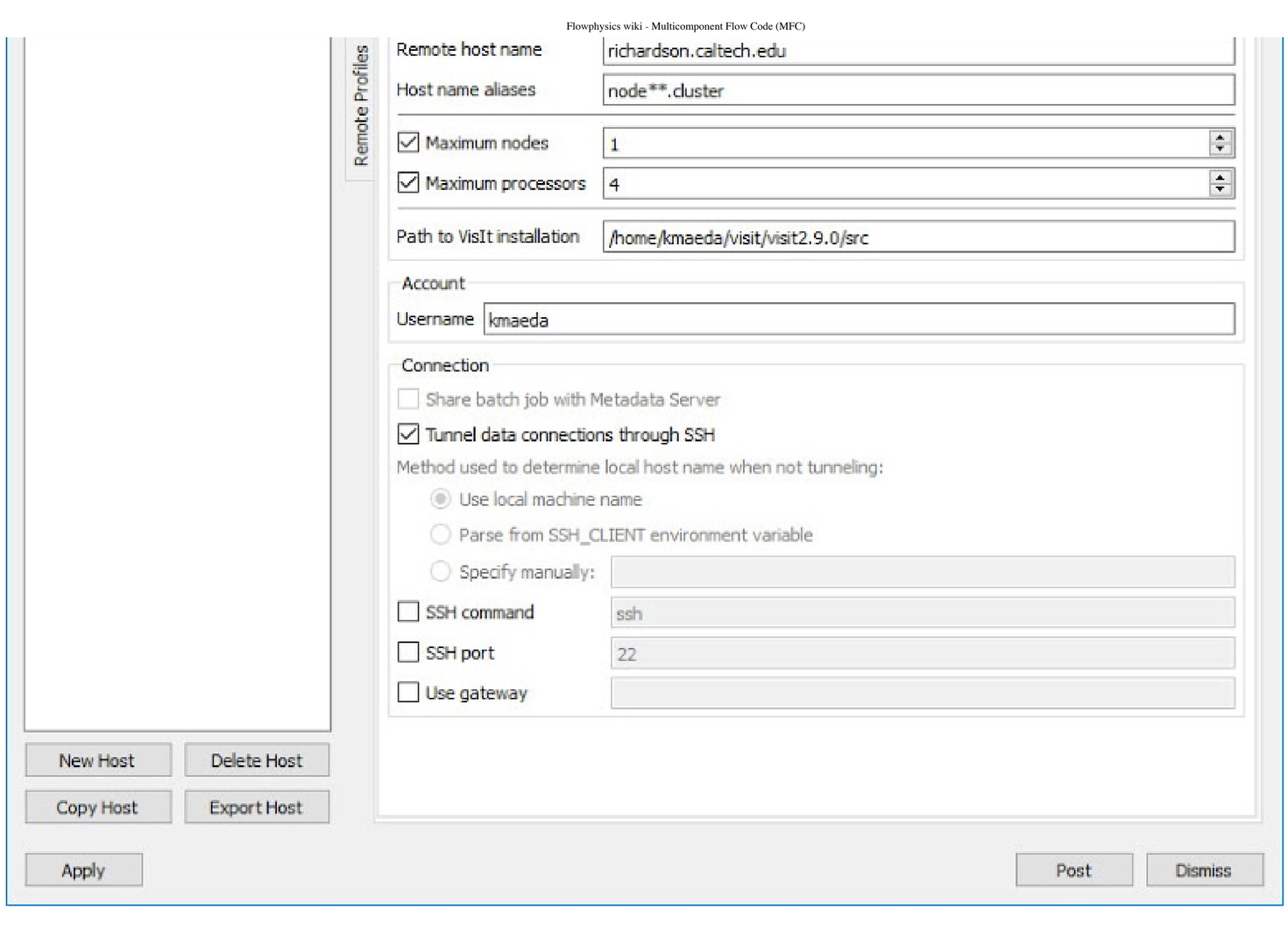


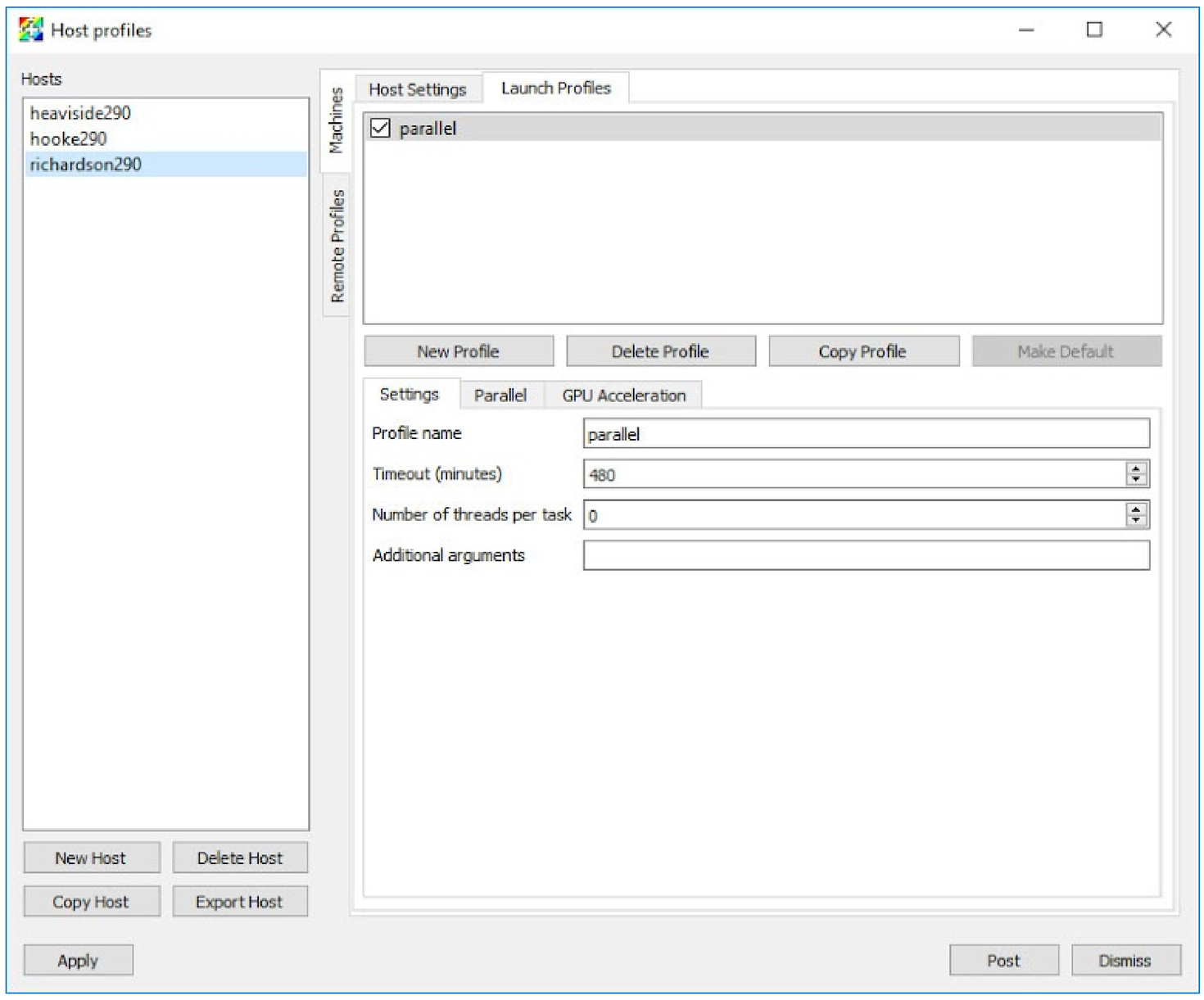


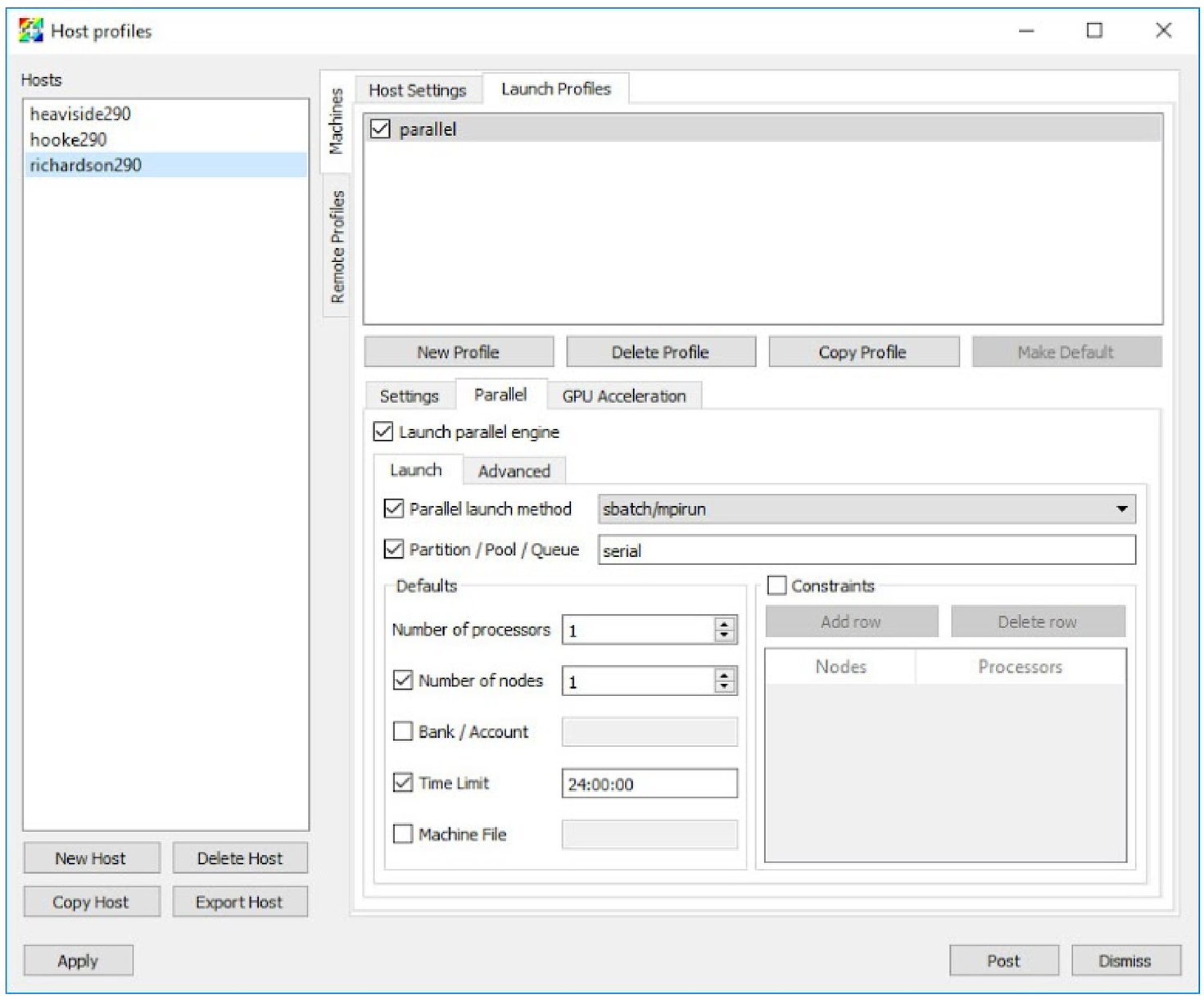


1. Richardson:




Remote Profiles
Remote host name
richardson.caltech.edu
Host name aliases
node**.cluster
Maximum nodes
1
Maximum processors
4
Path to VisIt installation
/home/kmaeda/visit/visit2.9.0/src
Account
Username
kmaeda
Connection
Share batch job with Metadata Server
Tunnel data connections through SSH
Method used to determine local host name when not tunneling:
Use local machine name
Parse from SSH_CLIENT environment variable
Specify manually:
SSH command
ssh
SSH port
22
Use gateway
New Host
Delete Host
Copy Host
Export Host
Apply
Post
Dismiss


Host profiles
Hosts
heaviside290
hooke290
richardson290
Machines
Remote Profiles
Host Settings
Launch Profiles
parallel
New Profile
Delete Profile
Copy Profile
Make Default
Settings
Parallel
GPU Acceleration
Profile name
parallel
Timeout (minutes)
480
Number of threads per task
0
Additional arguments
New Host
Delete Host
Copy Host
Export Host
Apply
Post
Dismiss


Host profiles
Hosts
heaviside290
hooke290
richardson290
Machines
Remote Profiles
Host Settings
Launch Profiles
parallel
New Profile
Delete Profile
Copy Profile
Make Default
Settings
Parallel
GPU Acceleration
Launch parallel engine
Launch
Advanced
Parallel launch method
sbatch/mpirun
Partition / Pool / Queue
serial
Defaults
Number of processors
1
Number of nodes
1
Bank / Account
Time Limit
24:00:00
Machine File
Constraints
Add row
Delete row
Nodes
Processors
New Host
Delete Host
Copy Host
Export Host
Apply
Post
Dismiss

Menu

Top

Richardson Cluster Materials

Forum

Resources

Manuals/Instructions

In-house codes

Immersed Boundary Code

Multicomponent Flow Code (MFC)

Public softwares/libraries

Miscellaneous Document

Other useful information

1. Save these host profiles for future VisIt sessions using **Options --> Save Settings**

Checking out copy of mfc_v3.0 from CVS repository

1. Add the following lines to ~/.bash_profile:
    1. export CVS_RSH=ssh
    2. export CVSROOT=:ext:jmeng@hooke.caltech.edu:/home/jmeng/cvsroot/
2. Source .bash_profile using source ~/.bash_profile
3. Create a directory to hold your local copy of MFC and change to that directory
4. cvs co mfc_v3.0

IMPORTANT:

This website has only 100 MB of storage.

Whenever you would like to share a file on Multicomponent Flow Code (MFC), please upload the file on Caltech Box using your Caltech account

and add the Link of the file ("Add link" tub below).

Access Caltech Box from the following link.

https://imss.caltech.edu/box