

Feat_FloWer Manual

Raphael Münster

April 17, 2018

Table Of Contents

1	Installation	2
1.1	Prerequisites	2
1.2	Checking out the Code from the Repository	2
1.3	Building the Code from Source	3
1.3.1	Linux Systems	3
2	Test	5
2.1	TestSec	5
2.1.1	SubTest	5
2.2	Appendix	5

Chapter 1

Installation

1.1 Prerequisites

The Feat_FloWer software package is accessible through the version control system *Git*. The main repository of the software is hosted on the servers of the TU Dortmund. In order to checkout a copy of the code an account that provides access to the LS3 servers is required, so it is recommended to get such an account. Furthermore, the code repository does not include the meshes used in the example applications. The meshes are stored in a different repository. So, at first we will get this repository and set up an environment variable so that the configuration mechanism of Feat_FloWer can find the mesh repository. The mesh repo can be checked out by:

```
> git clone ssh://username@lannister/home/user/git/mesh_repo.git
```

After you have checked out the mesh repository, it is time to set up the environment variable for Feat_FloWer. For the `BASH` or `ZSH` shells this can be done by adding the following line to your `.bashrc` or respectively `.zsh`:

```
> export Q2P1_MESH_DIR=/path/to/meshrepo/mesh_repo
```

1.2 Checking out the Code from the Repository

When you have acquired an appropriate account you can clone the Feat_FloWer repository by opening a terminal and entering the following sequence of commands:

```
> mkdir FeatFlow && cd FeatFlow
```

```
> git clone --recursive ssh://username@lannister/home/user/git/Feat_FloWer.git
```

This will create a parent folder called FeatFlow and within this folder the source

code will be contained. This is a preparation for building the code from source. The Feat_Flower code supports only out-of-source builds, meaning the binary files are not built in the same directory as the source code. This is done in order to prevent that binary files or object files litter the source directory or that these files show up in the version control system.

1.3 Building the Code from Source

The Feat_FloWer code can be build on Linux and Windows operating systems (and probably on MacOS, but this is not tested yet). On Linux systems an installation of the GNU compiler is needed, including the gfortran compiler and the matching OpenMPI libraries. The minimum requirement for the GCC is version 4.9.2. On Linux systems the Intel compiler can be used as an alternative compiler. On Windows systems the Intel compiler is a necessary requirement.

1.3.1 Linux Systems

After you have cloned the repository, navigate to the *FeatFlower* folder that you have created in the previous step and create another folder that will contain the binaries:

```
> mkdir bin
```

Verify with the command **pwd** that your FeatFlower folder now contains two subfolders *bin* and *Feat_FloWer*. At the core of the Feat_FloWer build system is the multi-platform build files generator CMake. On the servers of the TU Dortmund CMake is available as a loadable module. In order to successfully compile the code with the GNU compiler the following components are needed:

- GCC the GNU Compiler Collection including the gfortran compiler
- A matching OpenMPI installation
- CMake with a minimum version of 2.8

On the servers of the TU Dortmund load for example the following modules or newer versions of them:

- ```
> module load gcc/6.1.0
```
- ```
> module load openmpi/gcc6.1.x/1.10.2/non-threaded/no-cuda/ethernet
```
- ```
> module load cmake/3.5.2-ssl
```

After you have typed these commands verify that these module are loaded by checking the output of the command **module list**. The build of the basic software package is initiated by navigating into the *bin* folder that you have created before and evoking CMake from there:

```
> cd bin
```

```
> cmake ../Feat_FloWer
```

The build process is then started by the command:

```
> make -j 5
```

The option `<-j 5>` starts a parallel build using 5 processes.

# Chapter 2

## Test

### 2.1 TestSec

#### 2.1.1 SubTest

His rebus adducti et auctoritate Orgetorigis permoti constituerunt ea quae ad proficiscendum pertinerent comparare, iumentorum et carrorum quam maximum numerum coemere, sementes quam maximas facere, ut in itinere copia frumenti suppeteret, cum proximis civitatibus pacem et amicitiam confirmare. Ad eas res conficiendas biennium sibi satis esse duxerunt; in tertium annum profectionem lege confirmant. Ad eas res conficiendas Orgetorix deligitur. Is sibi legationem ad civitates suscipit. In eo itinere persuadet Castico, Catamantaloedis filio, Sequano, cuius pater regnum in Sequanis multos annos obtinuerat et a senatu populi Romani amicus appellatus erat, ut regnum in civitate sua occuparet, quod pater ante habuerit; itemque Dumnorigi Haeduo, fratri Diviciaci, qui eo tempore principatum in civitate obtinebat ac maxime plebi acceptus erat, ut idem conaretur persuadet eique filiam suam in matrimonium dat. Perfacile factu esse illis probat conata perficere, propterea quod ipse suae civitatis imperium obtenturus esset: non esse dubium quin totius Galliae plurimum Helvetii possent; se suis copiis suoque exercitu illis regna conciliaturum confirmat. Hac oratione adducti inter se fidem et ius iurandum dant et regno occupato per tres potentissimos ac firmissimos populos totius Galliae sese potiri posse sperant.

### 2.2 Appendix

TODO