Disclaimer

"This offering is not approved or endorsed by OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM® and OpenCFD® trade marks."

Introductory OpenFOAM® Course

From 8th to 12th July, 2013

University of Genoa, DICCA

Dipartimento di **I**ngegneria **C**ivile, **C**himica e **A**mbientale

UNIVERSITÀ DEGLI STUDI
DI GENOVA

Your Lecturer

Joel GUERRERO

joel.guerrero@unige.it

UNIVERSITÀ DEGLI STUDI
DI GENOVA

guerrero@wolfdynamics.com

wolf dynamics

Acknowledgements

These slides and the tutorials presented are based upon personal experience, OpenFOAM® source code, OpenFOAM® user guide, OpenFOAM® programmer's guide, and presentations from previous OpenFOAM® training sessions and OpenFOAM® workshops.

We gratefully acknowledge the following OpenFOAM® users for their consent to use their material:

- Hrvoje Jasak. Wikki Ltd.
- Hakan Nilsson. Department of Applied Mechanics, Chalmers University of Technology.
- Eric Paterson. Applied Research Laboratory Professor of Mechanical Engineering, Pennsylvania State University.

Today’s lecture

1. Debugging OpenFOAM®

Debugging OpenFOAM®

Debugging using Info statements (brute force)

- The simplest way to do debugging is to write out intermediate results to the screen, and check that those results are correct.

- In OpenFOAM® this is done using **Info** statements.

- This kind of debugging does not allow any control of the running of the code while debugging. It will just print out additional information.

- Info debugging requires that new lines are inserted in the source code, and that the source code must be re-compiled whenever a new Info statement has been added.

Debugging OpenFOAM®

Debugging using Info statements (brute force)

- When the code has been debugged, all the Info statements must be deleted, or commented, so that you do not get all that information when you are no longer debugging.

- OpenFOAM® provides an alternative to using user defined Info statements or brute force. This alternative let you choose when hard coded Info statements are switch-on or switch-off.

- This brings us to the next level of debugging in OpenFOAM®.

Debugging OpenFOAM®

Debugging using OpenFOAM® DebugSwitches

- In **$WM_PROJECT_DIR/etc/**controlDict, you will find a sort of super controlDict, that contains a list of DebugSwitches:

```
DebugSwitches
{
        Analytical       0;
        APIdiffCoefFunc     0;
        Ar     0;
        …
        Many more entries
        …
}
```

- Each class thus has its own DebugSwitch.
- DebugSwitches set to zero will produce no debug information.
- Different levels of debugging can be chosen by setting a DebugSwitch to 1, 2, 3 ...
- It is recommended to make a copy of the original file.

"This offering is not approved or endorsed by OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM® and OpenCFD® trade marks."

Debugging OpenFOAM®

Compiling OpenFOAM® in Debug mode

- In **$WM_PROJECT_DIR/etc/**bashrc you will find the environment variable WM_COMPILE_OPTION that can be set to Debug. That is what you need to do if you want to compile OpenFOAM in debug mode.

- After setting the WM_COMPILE_OPTION to Debug in **$WM_PROJECT_DIR/etc/**bashrc, you need to recompile OpenFOAM.

- To be sure that you are using a version compiled in Debug mode, in the terminal type: which icoFoam which should point at a path containing the directory **linux64GccDPDebug**.

Debugging OpenFOAM®

Compiling OpenFOAM® in Debug mode

- To run in debug mode, you should use a debugger (for instance GDB).

- When you run in debug mode the code executes a way much slower than in the optimized version, so never use a debug version for production runs.

- I will not go into details about how to Debug with GDB, but if you are interested you can look for information in internet.

“This offering is not approved or endorsed by OpenCFD Limited, the producer of the OpenFOAM software and owner of the OPENFOAM® and OpenCFD® trade marks.”

Thank you for your attention