

15th OpenFOAM Workshop - June 22, 2020 - Technical Session I-D

# A flexible and preCICE solver coupling ecosystem

Gerasimos Chourdakis, Technical University of Munich  
[chourdak@in.tum.de](mailto:chourdak@in.tum.de)



Benjamin Uekermann, Eindhoven University of Technology  
[b.w.uekermann@tue.nl](mailto:b.w.uekermann@tue.nl)



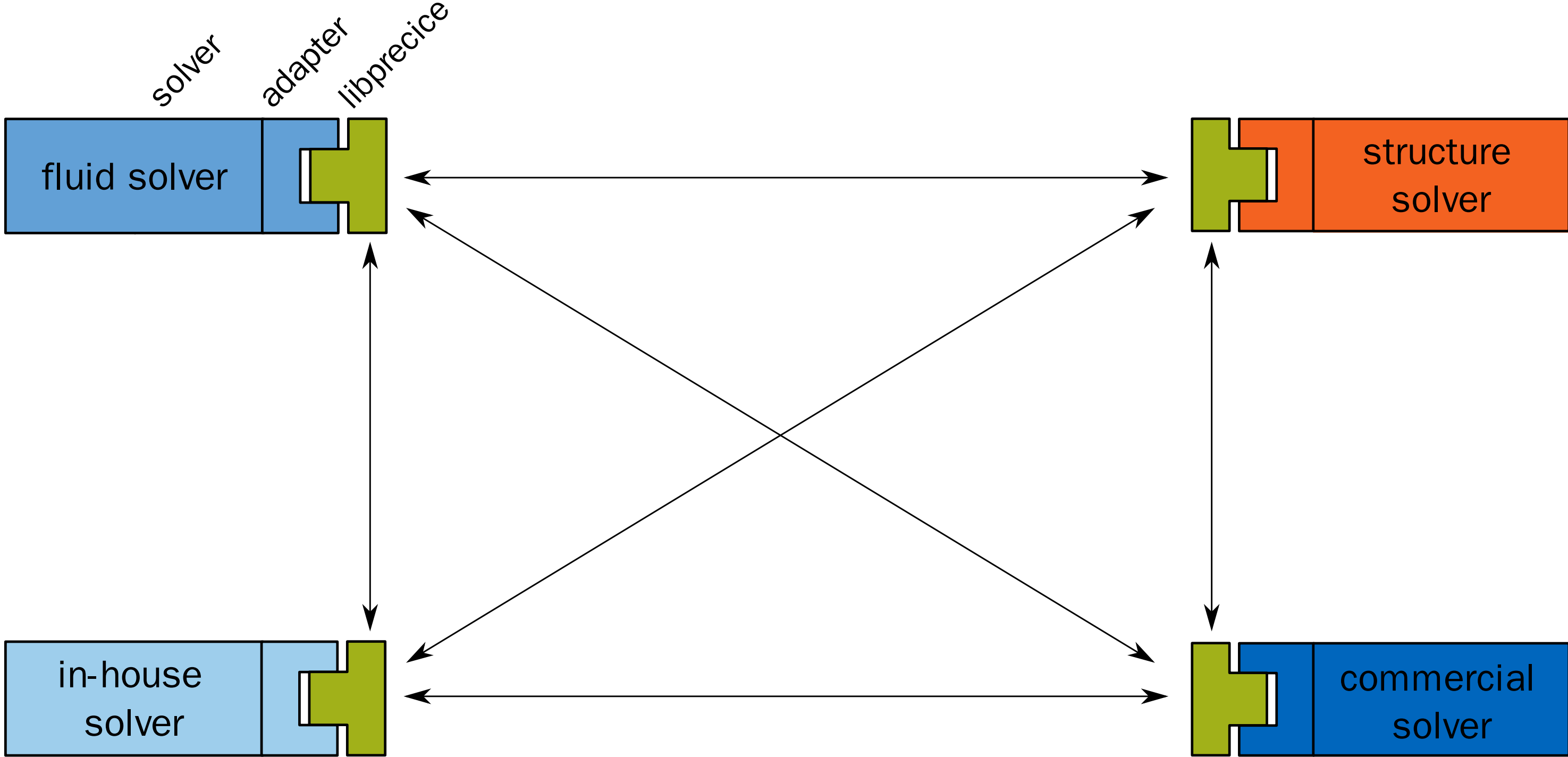
Find these slides on GitHub

<https://github.com/MakisH/ofw15-slides>

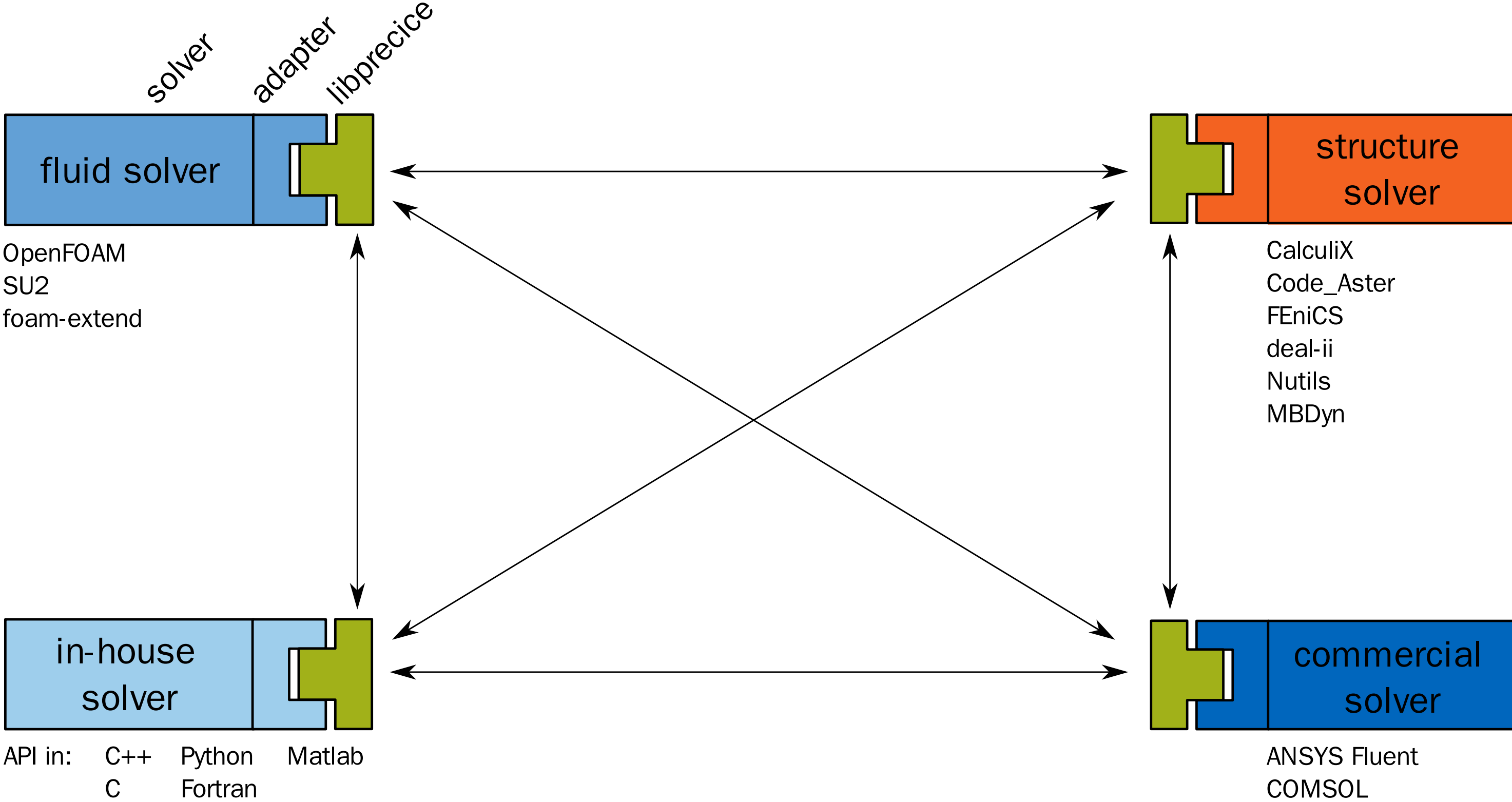
# The Big Picture



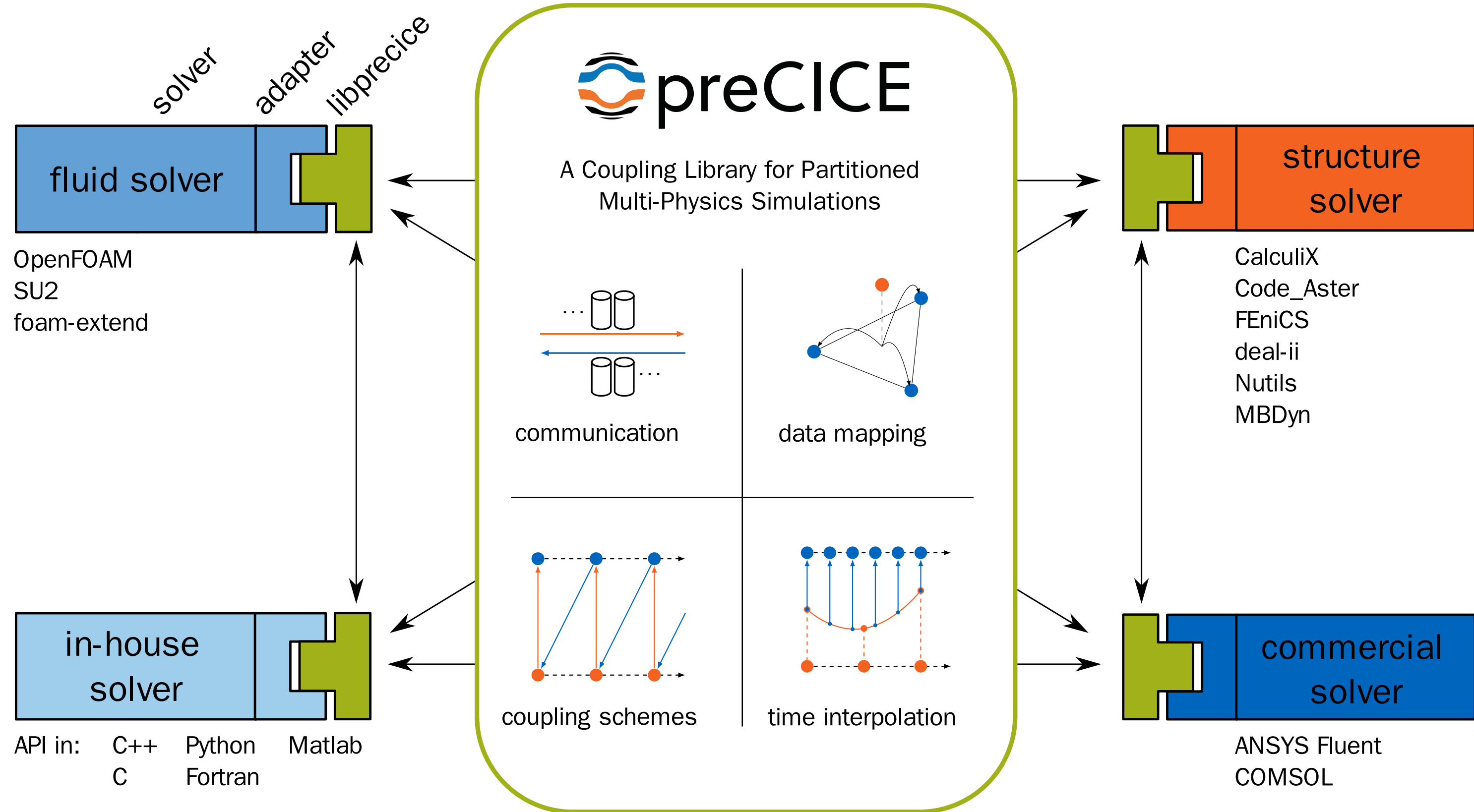
# The Big Picture



# The Big Picture



# The Big Picture



# News

# preCICE v2.0

- Simplified config & API
- XML reference & visualizer
- Faster initialization
- Spack / Debian / AUR packages
- Better building & testing
- [xSDK](#) member
- Better Python bindings
- New Matlab bindings

Upgrade guide [in the wiki](#)

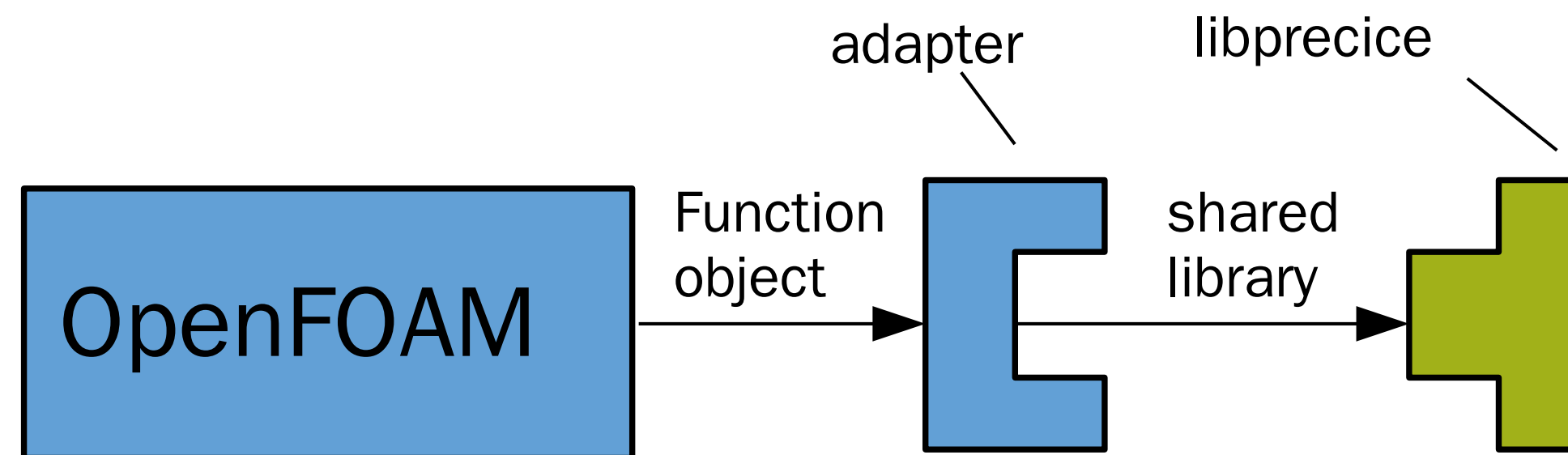


# Other news

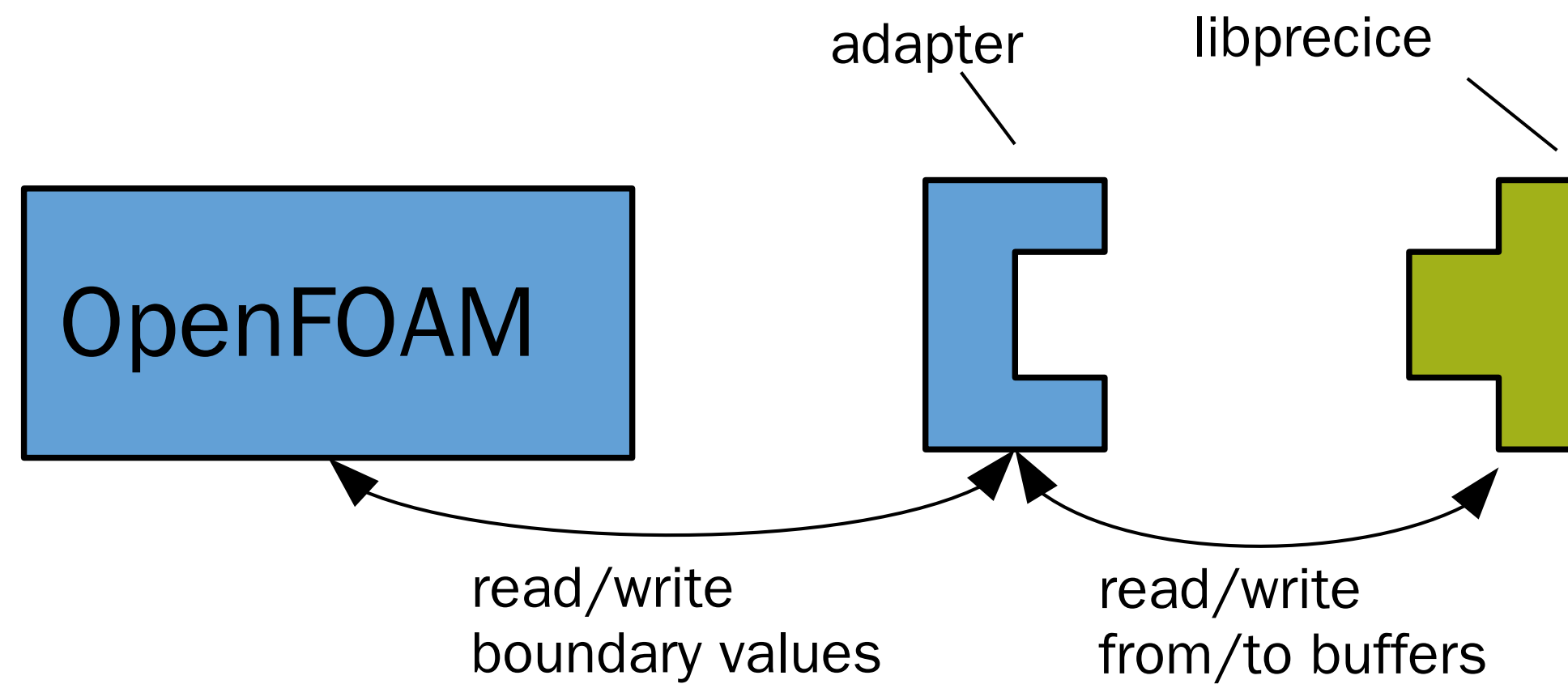
- [deal.ii adapter](#)
  - new non-linear example for FSI
- [FEniCS adapter](#)
  - new example for FSI
- [code\\_aster adapter](#)
  - revived for code\_aster 14 and preCICE v2

# The OpenFOAM adapter

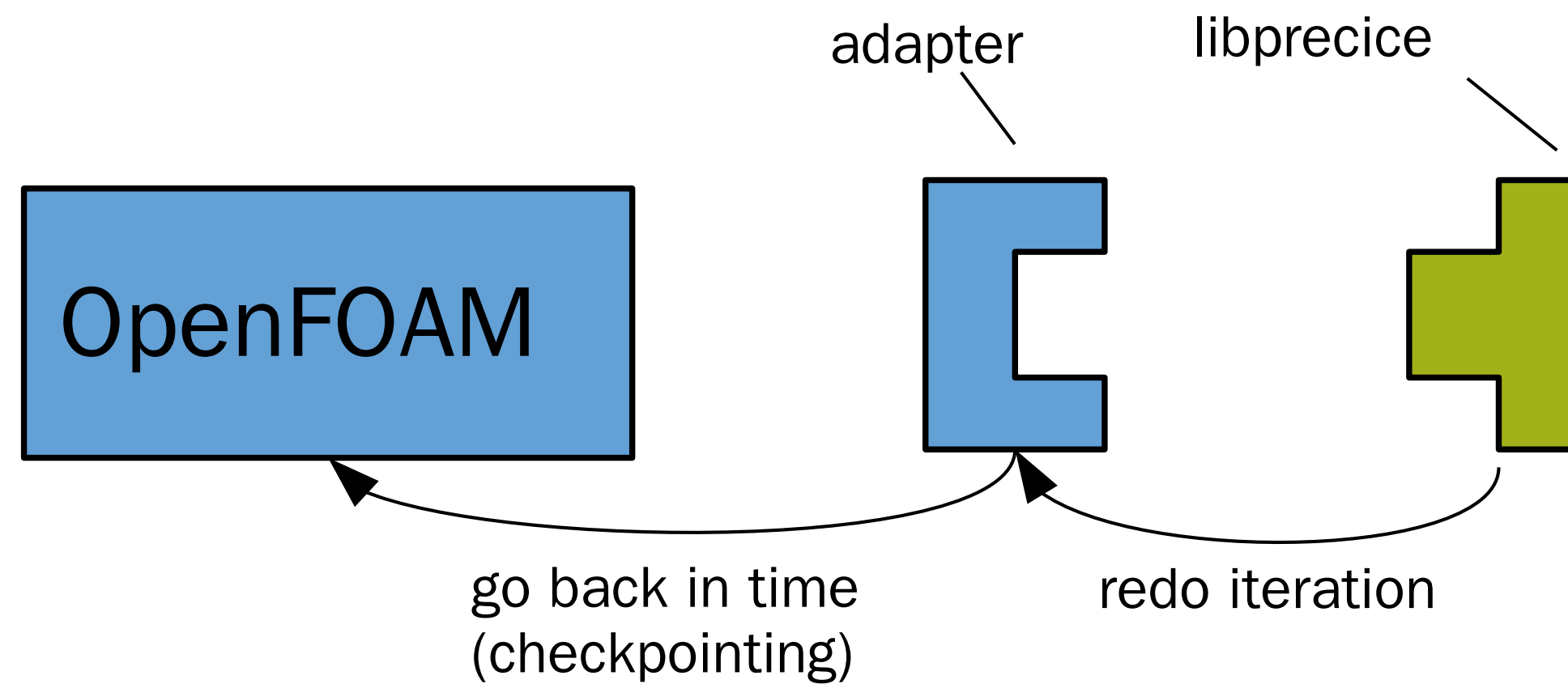
# What does the adapter do?



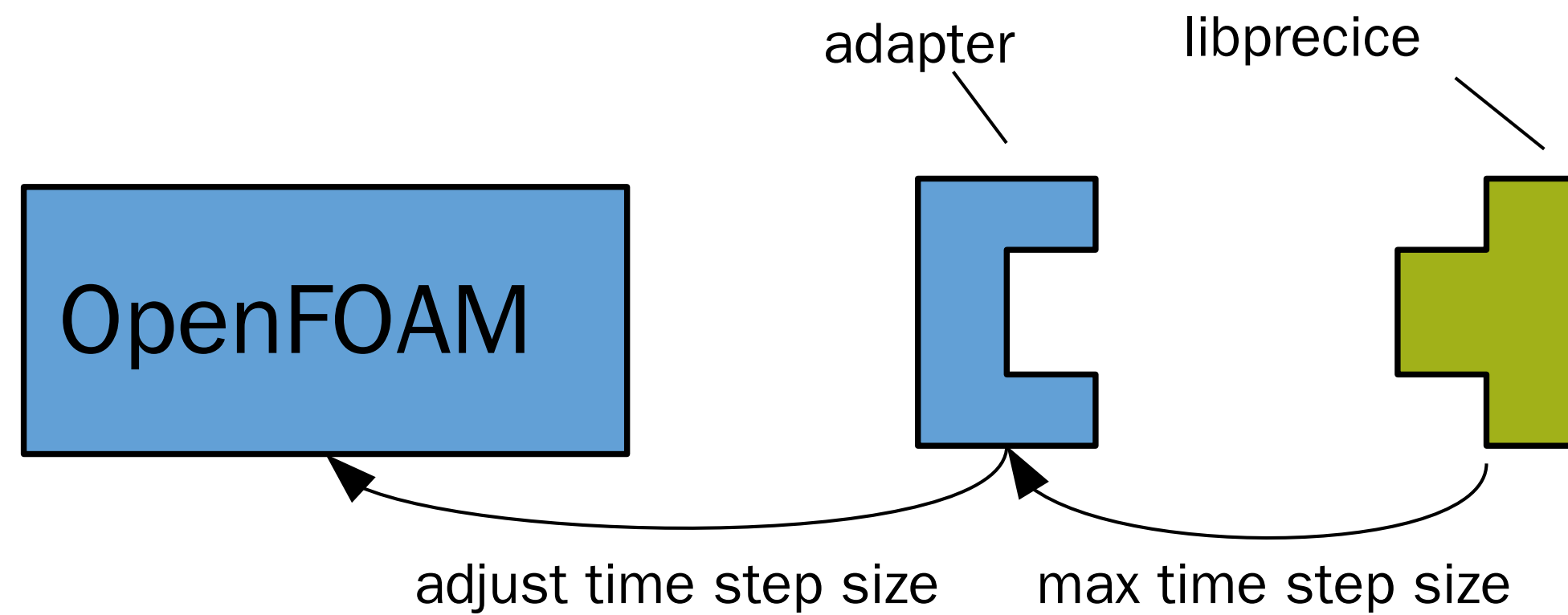
# What does the adapter do?



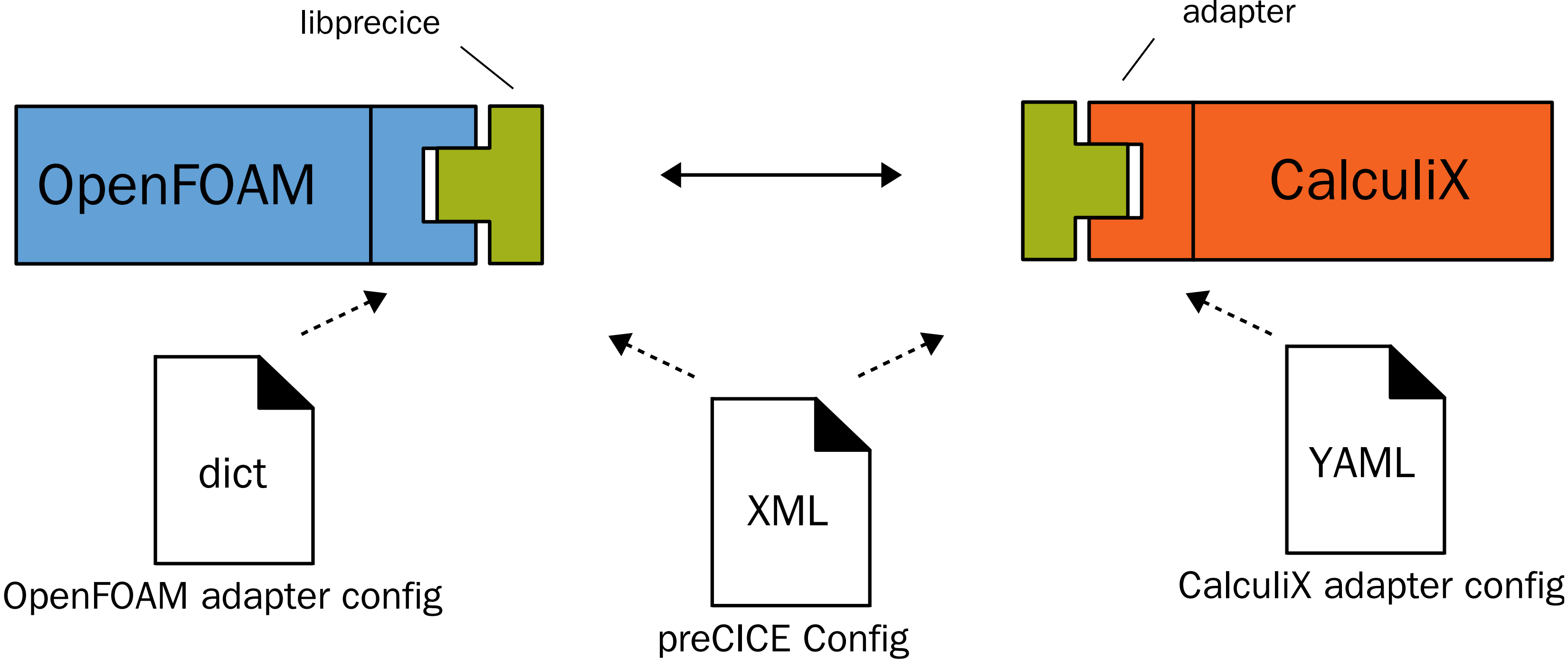
# What does the adapter do?



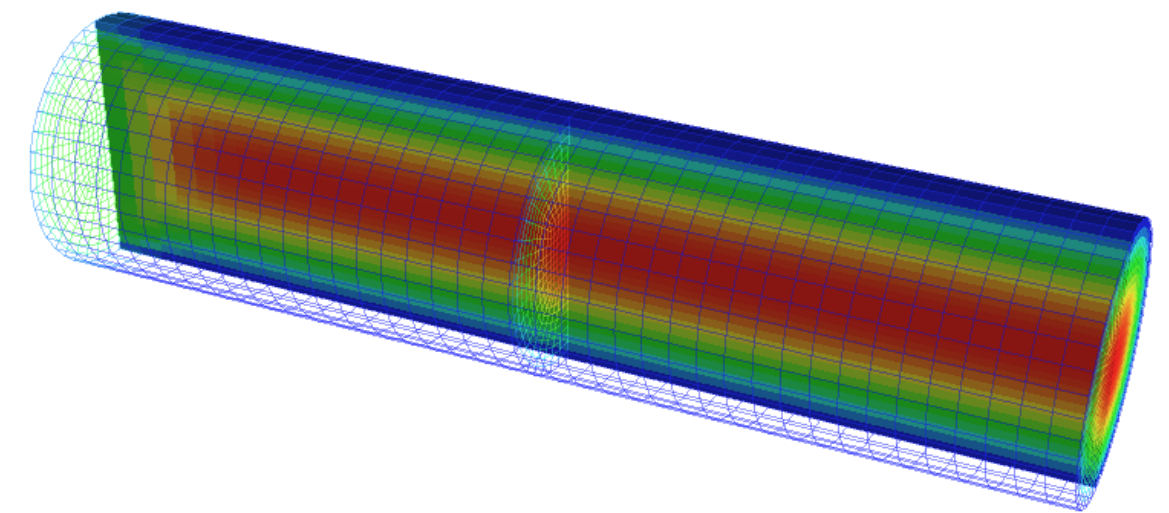
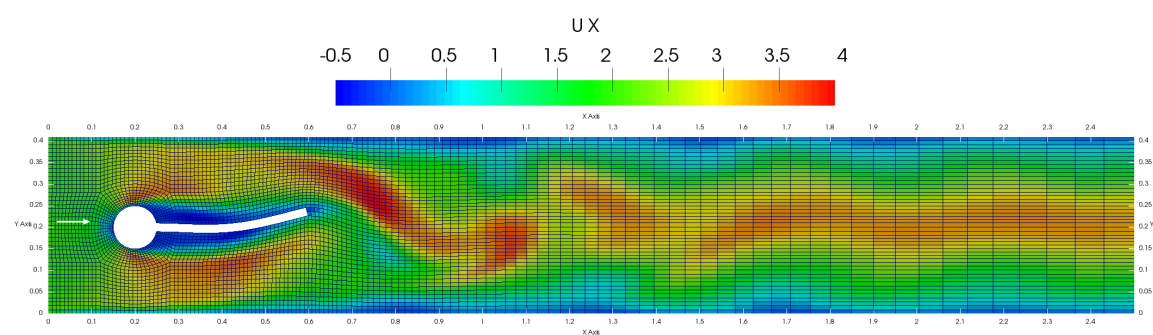
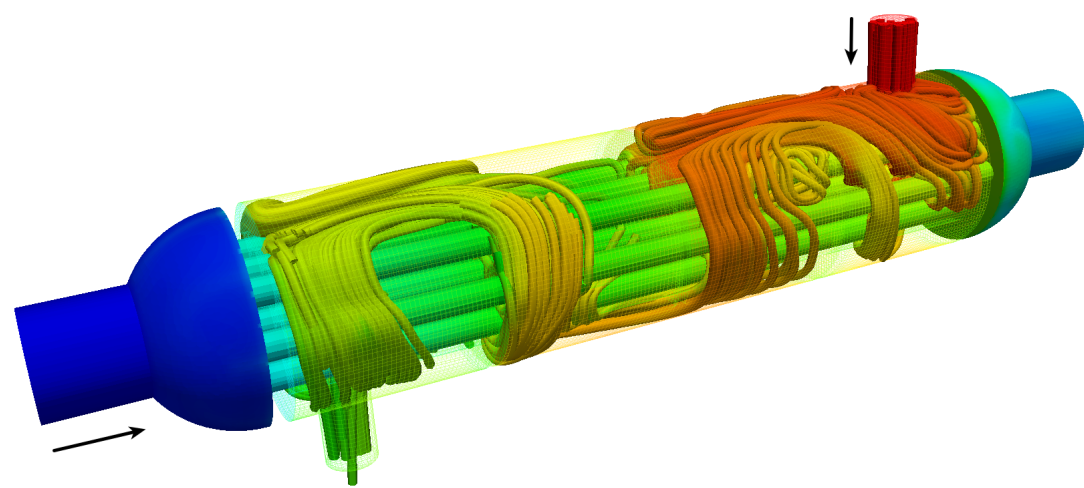
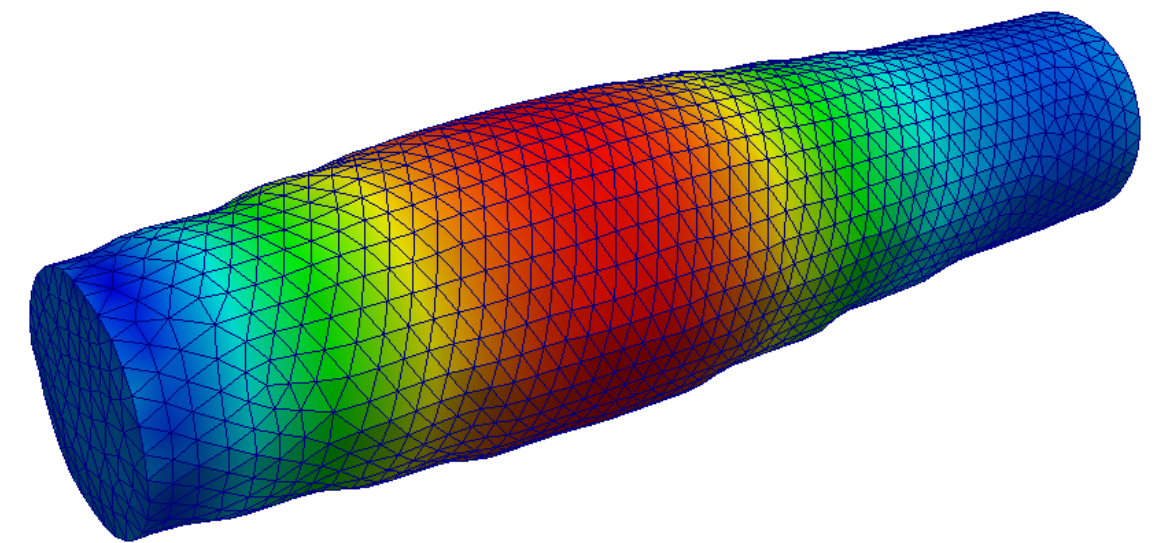
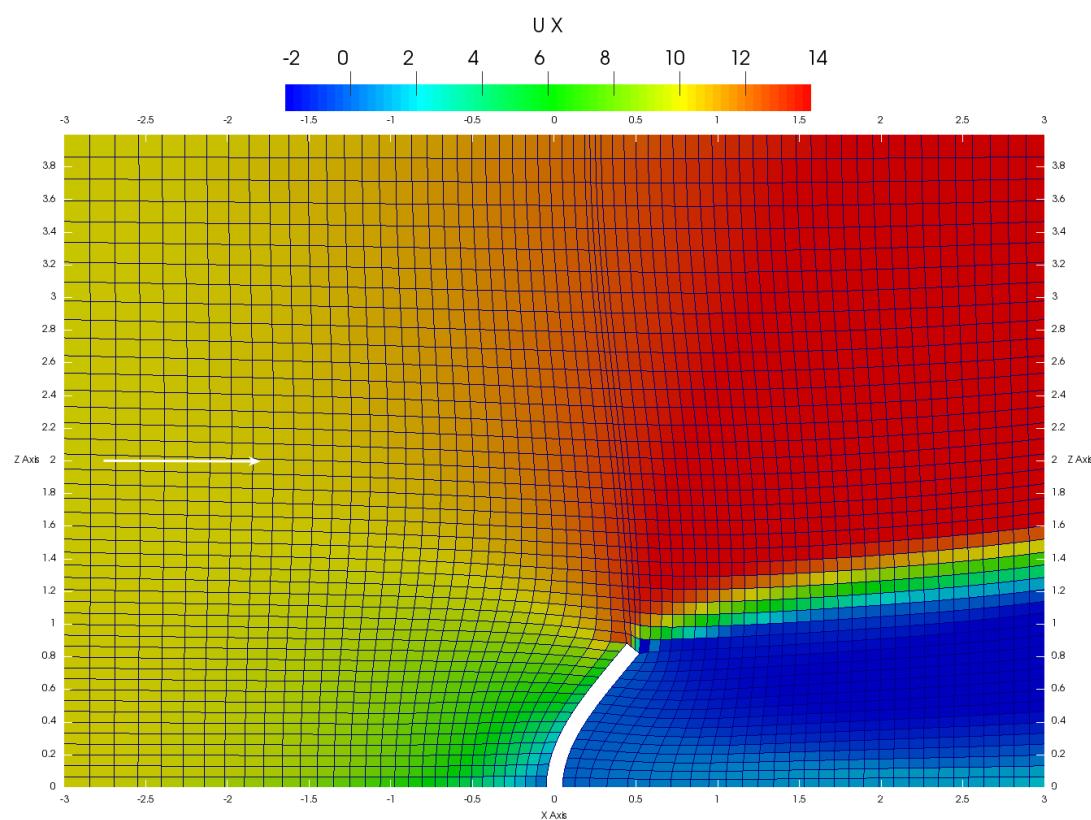
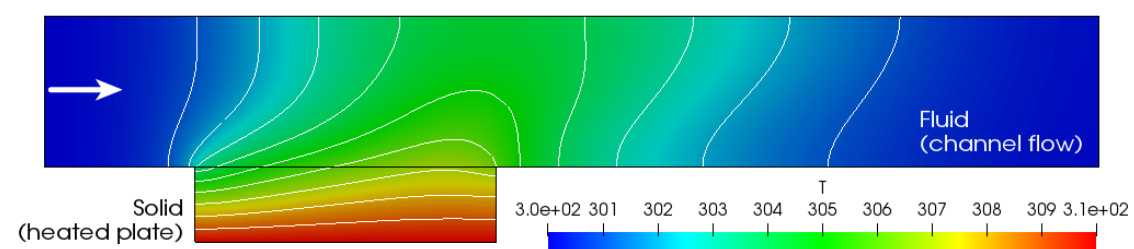
# What does the adapter do?



# Configuration



# Tutorials



Tutorials using OpenFOAM in our [wiki](#).



```

---[preciceAdapter] [DEBUG] connectivity : 0 | (base) ~/git/hub/tutorials/FSI/flap_perp/OpenFOAM-CalculiX/Solid [master]$ ls
---[preciceAdapter] [DEBUG] patches : | flap.12d flap.inp mesh.sh post_flap.fbd pre_flap.fbd
---[preciceAdapter] [DEBUG] - flap | (base) ~/git/hub/tutorials/FSI/flap_perp/OpenFOAM-CalculiX/Solid [master]$ cg
---[preciceAdapter] [DEBUG] writeData : | x -bg pre_flap.fbd
---[preciceAdapter] [DEBUG] - Forces0 | on a Linux machine, nodename Magdalena, release 4.15.0-106-generic, version #
---[preciceAdapter] [DEBUG] readData : | 107~16.04.1-Ubuntu SMP Thu Jun 4 15:40:05 UTC 2020, machine x86_64
---[preciceAdapter] [DEBUG] - mesh : Fluid-Mesh-Nodes | parameters:3 arguments:2
---[preciceAdapter] [DEBUG] locations : faceNodes |
---[preciceAdapter] [DEBUG] connectivity : 0 | pre_flap.fbd opened
---[preciceAdapter] [DEBUG] patches : | reading file
---[preciceAdapter] [DEBUG] - flap | #!/bin/bash
---[preciceAdapter] [DEBUG] writeData : | #For the beam elements
---[preciceAdapter] [DEBUG] readData : | please wait for 'ready'
---[preciceAdapter] [DEBUG] - Displacements0 | makeSurfaces
---[preciceAdapter] [DEBUG] Configuring the FSI module... | getElemNormalen
---[preciceAdapter] [DEBUG] user-defined solver type : | realloc_colNr
---[preciceAdapter] [DEBUG] pointDisplacement field name : pointDisplacem | add the faces
ent | updateDispLists
---[preciceAdapter] [DEBUG] Unknown solver type. Determining the solver type. | delSet
.. | ready
---[preciceAdapter] [DEBUG] Automatically determined solver type : incompressible | please wait for 'ready'
---[preciceAdapter] [DEBUG] Checking the timestep type (fixed vs adjustable) | write abaqus data
.. | file all.msh opened
---[preciceAdapter] [DEBUG] Timestep type: fixed. | ready
---[preciceAdapter] [DEBUG] Creating the preCICE solver interface... | please wait for 'ready'
---[preciceAdapter] [DEBUG] Number of processes: 1 | write file: fix1.nam
---[preciceAdapter] [DEBUG] MPI rank: 0 | ready
---[precice] This is preCICE version 2.0.2 | ready
---[precice] Revision info: v2.0.2 | please wait for 'ready'
---[precice] Configuring preCICE with configuration: "precice-config.xml" | write file: surface.nam
---[preciceAdapter] [DEBUG] preCICE solver interface was created. | ready
---[preciceAdapter] [DEBUG] Creating interfaces... | # #rot z
---[preciceAdapter] [DEBUG] Number of face centres: 33 | #rot r 50
---[preciceAdapter] [DEBUG] Interface created on mesh Fluid-Mesh-Faces | #rot u 50
---[preciceAdapter] [DEBUG] Adding coupling data writers... | # Add thing here for the boundary condition
---[preciceAdapter] [DEBUG] Added writer: Force. | # sys echo
---[preciceAdapter] [DEBUG] Adding coupling data readers... | # quit
---[preciceAdapter] [DEBUG] Number of face nodes: 68 | # plus la all
---[preciceAdapter] [DEBUG] Interface created on mesh Fluid-Mesh-Nodes | # plus pa all
---[preciceAdapter] [DEBUG] Adding coupling data writers... | done
---[preciceAdapter] [DEBUG] Adding coupling data readers... |
---[preciceAdapter] [DEBUG] Added reader: Displacement. | (base) ~/git/hub/tutorials/FSI/flap_perp/OpenFOAM-CalculiX/Solid [master]$ cd
---[preciceAdapter] [DEBUG] Initalizing the preCICE solver interface... | ..
---[precice] Setting up master communication to coupling partner/s | (base) ~/git/hub/tutorials/FSI/flap_perp/OpenFOAM-CalculiX [master]$ c
cx_preC
ICE -i Solid/flap -precice-participant Calculix

```



# New: `system/preciceDict`

```
preciceConfig "precice-config.xml";  
  
participant Fluid;  
  
modules (FSI);  
  
interfaces  
{  
  Interface1  
  {  
    mesh          Fluid-Mesh-Faces;  
    patches       (flap);  
    locations     faceCenters;  
  
    readData  
    (  
    );  
  }  
}
```

No need for `yaml-cpp` anymore!



# preciceDict - implementation

```
IOdictionary preciceDict
(
    IOobject
    (
        "preciceDict",
        runtime_.system(),
        mesh_,
        IOobject::MUST_READ_IF_MODIFIED,
        IOobject::NO_WRITE
    )
);

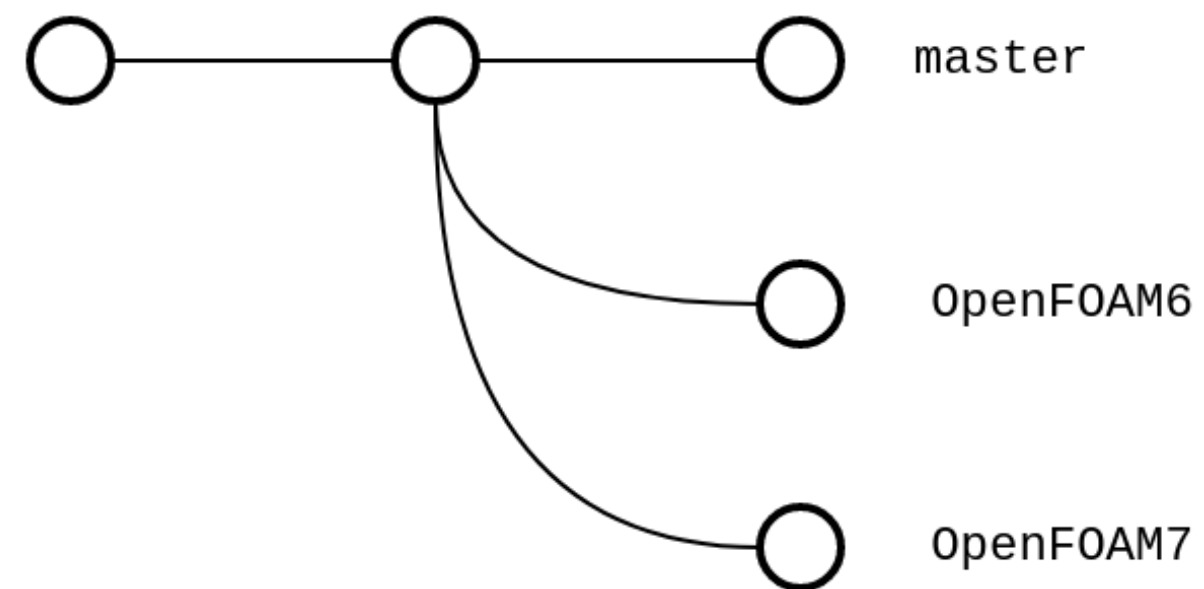
// lookupType<T>("name") is deprecated in openfoam.com since v1812,
// which recommends get<T>("name") instead.
preciceConfigFilename_ = preciceDict.lookupType<fileName>("preciceConfig")

// ...
```

Many thanks to Mark Olesen (ESI) for the hints



# Before: Special cases in branches



Multiple Git branches, differing in small details

# New: One source to rule them all

```
// Adapter.C  
  
// Version-specific code with possible variants:  
// - const_cast<Time&>(runTime_).setDeltaT(timestepSolver_, false);  
// - const_cast<Time&>(runTime_).setDeltaTNoAdjust(timestepSolver_);  
#include "version-specific/setDeltaT.H"
```

```
# Make/options  
  
EXE_INC = \  
    # ...  
    -Ivariants/$(WM_PROJECT_VERSION) \  
    # e.g. -Ivariants/v1912
```



# New: One source to rule them all

```
variants/  
├── 4.0  
│   └── version-specific  
│       └── setDeltaT.H  
├── 4.1 -> 4.0  
├── 5.0  
│   └── version-specific  
│       └── setDeltaT.H -> ../../4.0/version-specific/setDeltaT.H  
├── 6  
│   └── version-specific  
│       └── setDeltaT.H  
├── 7  
│   └── version-specific  
│       └── setDeltaT.H -> ../../6/version-specific/setDeltaT.H  
├── v1712  
│   └── version-specific  
│       └── setDeltaT.H -> ../../4.0/version-specific/setDeltaT.H  
└── 1.0.0 -> 1.7.12/
```



# New: One source to rule them all

```
variants/  
├── 4.0  
│   └── version-specific  
│       ├── directory_type.H  
│       ├── init.H  
│       └── setDeltaT.H  
├── 4.1 -> 4.0  
├── 5.0  
│   └── version-specific  
│       ├── directory_type.H -> ../../4.0/version-specific/directory_type  
│       ├── init.H  
│       └── setDeltaT.H -> ../../4.0/version-specific/setDeltaT.H  
├── 6  
│   └── version-specific  
│       ├── directory_type.H -> ../../5.0/version-specific/directory_type  
│       ├── init.H  
│       └── setDeltaT.H
```



# New: Pressure-based solver type selection

```
dimensionSet pressureDimensionsCompressible(1, -1, -2, 0, 0, 0, 0);
dimensionSet pressureDimensionsIncompressible(0, 2, -2, 0, 0, 0, 0);

if (mesh_.foundObject<volScalarField>("p"))
{
    volScalarField p_ = mesh_.lookupObject<volScalarField>("p");

    if (p_.dimensions() == pressureDimensionsCompressible)
        solverType = "compressible";
    else if (p_.dimensions() == pressureDimensionsIncompressible)
        solverType = "incompressible";
}
```

Thanks to David Schneider (TUM) for [adding this](#).



## New: Write stresses (FSI)

- Before: write forces, read displacements
- Now: write forces or stresses, read displacements
  - No need for conservative mapping!

Thanks to David Schneider (TUM) for [adding this](#).

# Upcoming: Unit & integration tests

- Already CI with system tests
- Wish: Test specific parts of the adapter
  - Unit tests with [Catch2](#)
  - Integration tests with [Google Test](#)
  - Other ideas?

Prototype [contributed](#) by Qunsheng Huang (TUM).

A community

# New: Forces for all

## Forces calculation for compressible, multiphase and turbulence #64

Merged **MakisH** merged 14 commits into `precice:develop` from `JSeuffert:FSI` on Sep 18, 2019

Conversation 5 Commits 14 Checks 1 Files changed 5



**JSeuffert** commented on Jan 22, 2019

Contributor

Concerning issues [#57](#) [#42](#)

Maybe in parts the same as pull request [#51](#)

Forces calculation is implemented as in the forces functionObject.

I tested the correct implementation with: `pimpleDyMFoam`, `rhoPimpleDyMFoam`, `interDyMFoam`, `compressibleInterDyMFoam`; laminar flow and RAS turbulence model

Tested with Openfoam 4 (<https://github.com/OpenFOAM/OpenFOAM-4.x>), `precice` 1.3.0, `Calculix` 2.13

Review



Assign



Label



None





Contributed by Julian Seuffert (KIT).






# WIP: Volume coupling


## WIP: Add volume coupling of temperature #97

 Open JaegerSt wants to merge 1 commit into `precice:master` from `JaegerSt:dev_volume_coupling_AIT` 

 Conversation 15  Commits 1  Checks 1  Files changed 11

 **JaegerSt** commented on Aug 20, 2019 First-time contributor  

openFOAM volume coupling is implemented for the temperature field located in the AITS directory

 1

Contributed by @JaegerSt and @StefanScG (AIT).

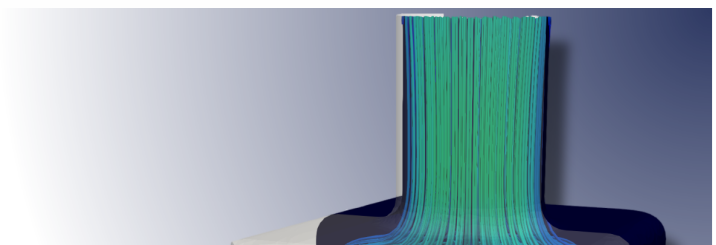
# Testimonials

## Do you use preCICE?

We are always happy to hear about interesting projects that use preCICE.

Tell us your story!

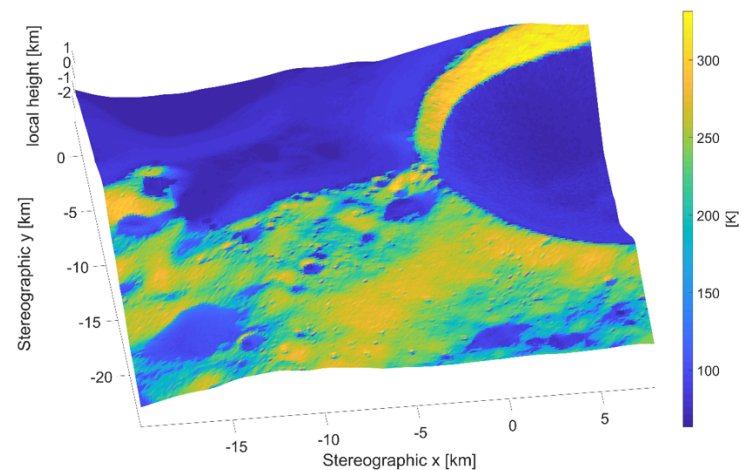
### A GUI for OpenFOAM/CalculiX FSI Coupling with preCICE



DHCAE Tools provides services to support companies and universities for the implementation of open-source solver technology. Beside this, DHCAE Tools offers calculation services for flow applications, whereby fluid-structure applications are a long-standing field of activity with commercial solvers as well as with self-developed couplings. Due to preCICE's outstanding capabilities, the setup of a coupling between OpenFOAM and CalculiX is now supported by DHCAE Tools in the graphical interface CastNet. This simplifies the mesh generation, the case setup for both OpenFOAM and CalculiX, the definition of the coupling parameters and finally the monitoring of the simulation. [Learn more](#)

*Ulrich Heck, DHCAE Tools GmbH, Krefeld, Germany*

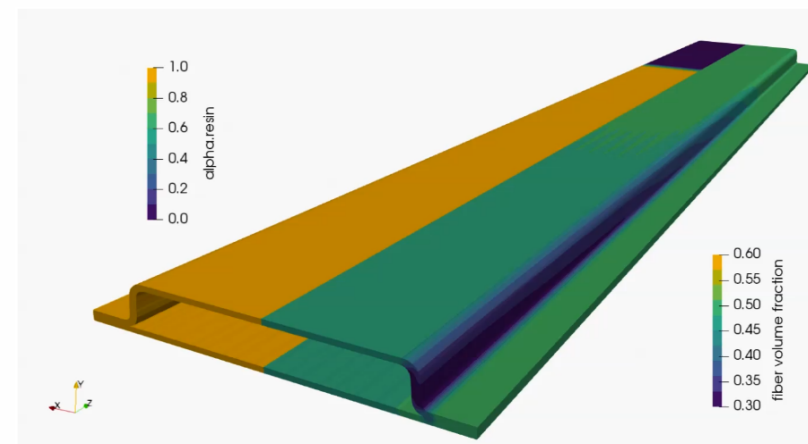
### Simulation of Temperatures on the Moon with THERMoS



We are developing **THERMoS**, a tool for simulation of temperatures on the surface of the Moon including rovers and astronauts operating on the lunar surface. Ray tracing on a single or multiple GPUs with NVIDIA Optix calculates the heat transfer by radiation between surface elements, while a MATLAB routine solves the equation of heat diffusion. preCICE couples the two domains and handles the communication between MATLAB and NVIDIA Optix. Other solvers and simulation approaches (instead of MATLAB and NVIDIA Optix) are going to be tested in the future with the aid of the flexibility that preCICE offers. [Learn more](#)

*Matthias Killian, Chair of Astronautics, Technical University of Munich (TUM), Germany*

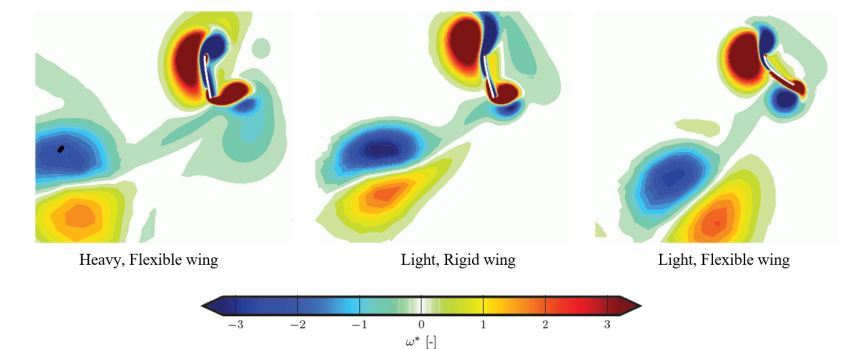
### Fluid-Structure Interaction during Sandwich Manufacturing



Structural sandwich components made of continuous fiber reinforced plastics (CFRP) are increasingly demanded by the automotive industry. In the Resin Transfer Molding (RTM) manufacturing process, a polymer foam core is embedded between dry fibers. During manufacturing, a liquid polymer resin infiltrates the fibers with high injection pressure, which leads to a deformation of the foam core. We simulate the mold filling with OpenFOAM and the foam core deformation with CalculiX. By coupling OpenFOAM and CalculiX via preCICE, we can now predict foam core deformation and mold filling behavior correctly. This allows us to optimize the manufacturing of high performance lightweight CFRP sandwich components. [Learn more](#)

*Julian Seuffert, Lightweight Technology, Institute of Vehicle System Technology (FAST), Karlsruhe Institute of Technology (KIT), Germany*

### Fluid-Structure Interaction on Flapping Wings



The flow around flapping wings allows them to create high lift using various of unsteady flow phenomena. Adding flexibility to the wing can help to reduce drag and increase performance. At TU Delft, we are investigating flapping wings and the implication of flexibility in these wings. The high, nonlinear deformations of these wings require a strongly coupled simulation to find a solution. For this means a framework is set up using CalculiX and OpenFOAM, coupled with preCICE. For this work, the OpenFOAM adapter was extended to support force and displacement coupling in FSI simulations. The large number of coupling functionalities in preCICE gives the user the opportunity to build advanced and scalable simulations with ease. [Learn more](#)

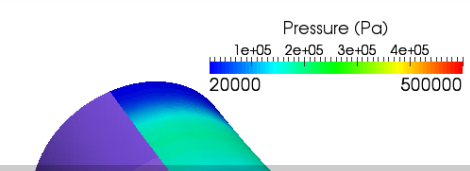
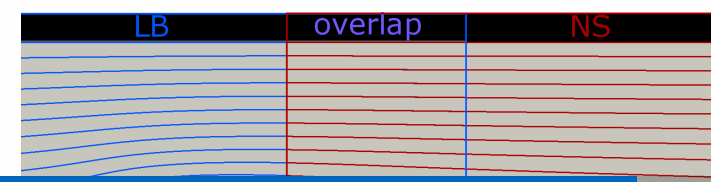
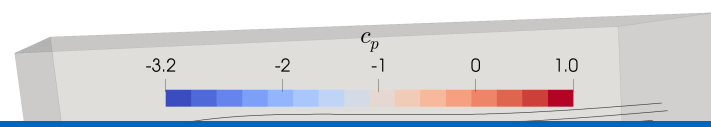
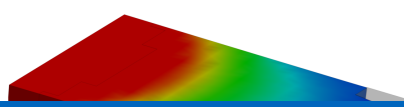
*Derek Risseuw, Aerodynamics, Faculty of Aerospace Engineering, TU Delft, The Netherlands*

### Coupled Simulation of the Continuous Casting Process

### Fluid-Structure Interaction of Inflatable Wing Sections

### Hybrid simulation methods for wind modelling in urban areas

### FSI Simulations of High Impact Loads on Structures



# Resources



# Start here: precice.org



## Welcome to precICE

**News:** precICE release [v2.0.2](#) available since Apr 3, 2020

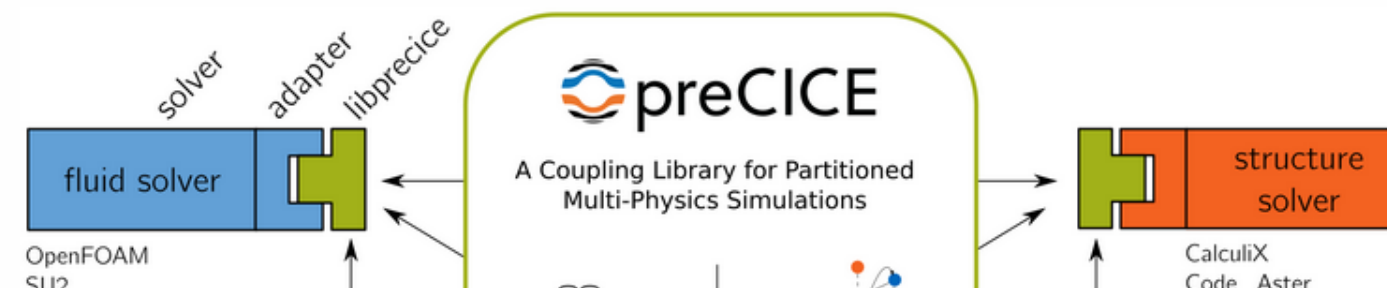
**Now also on YouTube:** Subscribe to our new [YouTube channel](#), where you can also find the talks from the precICE Workshop 2020.

**precICE in xSDK:** precICE is now part of the [xSDK](#) framework. [Read the compatibility policies...](#)

precICE (Precise Code Interaction Coupling Environment) is a coupling library for partitioned multi-physics simulations, including, but not restricted to fluid-structure interaction and conjugate heat transfer simulations. Partitioned means that precICE couples existing programs (solvers) capable of simulating a subpart of the complete physics involved in a simulation. This allows for the high flexibility that is needed to keep a decent time-to-solution for complex multi-physics scenarios. precICE runs efficiently on a wide spectrum of systems, from low-end workstations up to complete compute clusters and has proven scalability on 10000s of MPI Ranks.

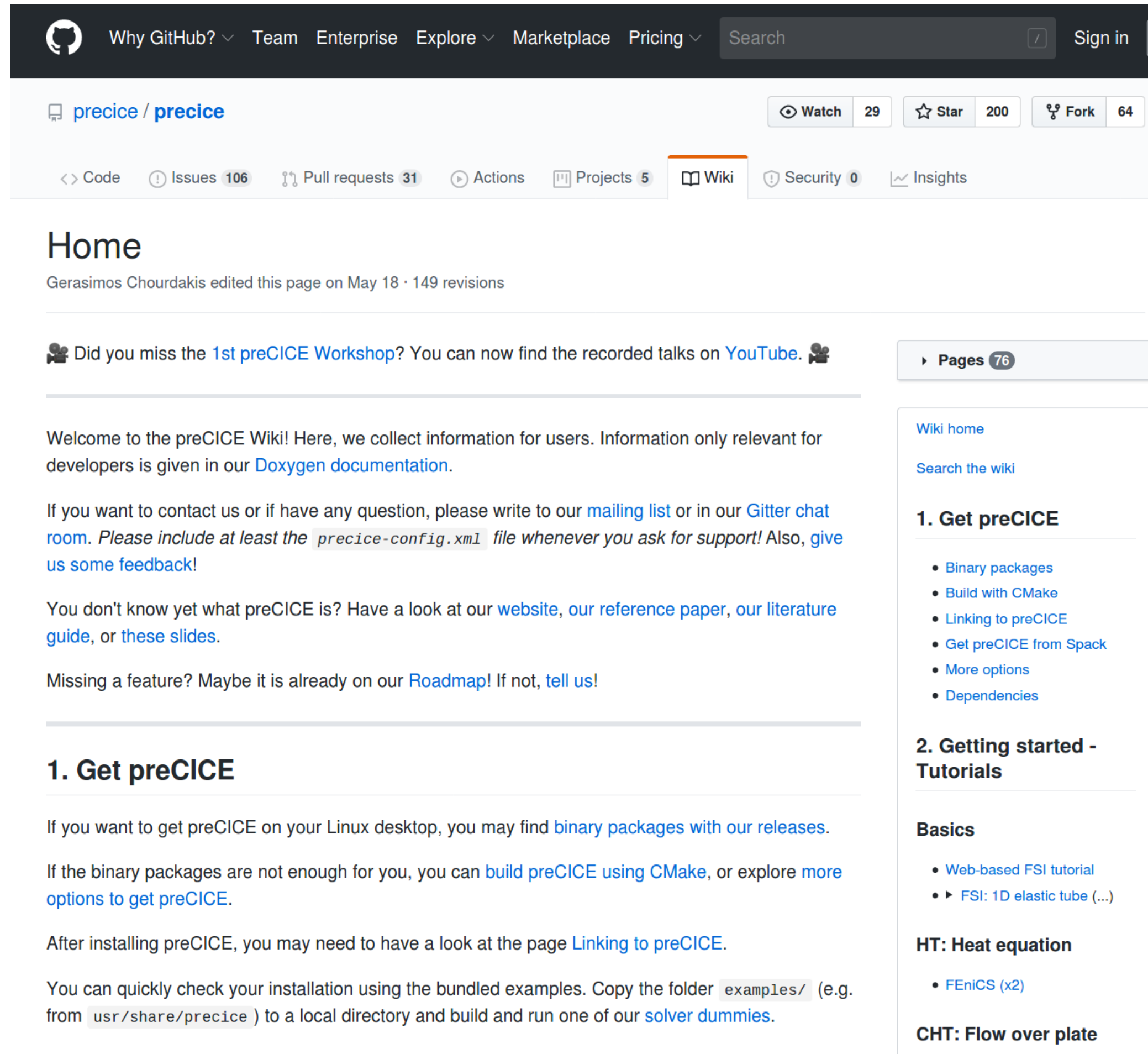
The software offers methods for transient equation coupling, communication means, and data mapping schemes. precICE is written in C++ and offers additional bindings for C, Fortran, Matlab, and Python. [Ready-to-use adapters](#) for well-known commercial and open-source solvers, such as OpenFOAM, deal.II, FEniCS, SU2, or CalculiX, are available. Due to the minimally-invasive approach of precICE, adapters for in-house codes can be implemented and validated in only a few weeks.

precICE is an open-source software under the LGPL3 license and available on [GitHub](#).





# Documentation



The screenshot shows the GitHub repository page for 'preCICE'. At the top, there is a navigation bar with links for 'Why GitHub?', 'Team', 'Enterprise', 'Explore', 'Marketplace', and 'Pricing'. A search bar and a 'Sign in' button are also present. Below the navigation bar, the repository name 'preCICE / preCICE' is displayed, along with statistics for 'Watch' (29), 'Star' (200), and 'Fork' (64). A secondary navigation bar includes links for 'Code', 'Issues' (106), 'Pull requests' (31), 'Actions', 'Projects' (5), 'Wiki' (highlighted), 'Security' (0), and 'Insights'.

## Home

Gerasimos Chourdakis edited this page on May 18 · 149 revisions

Did you miss the [1st preCICE Workshop](#)? You can now find the recorded talks on [YouTube](#).

Welcome to the preCICE Wiki! Here, we collect information for users. Information only relevant for developers is given in our [Doxygen documentation](#).

If you want to contact us or if have any question, please write to our [mailing list](#) or in our [Gitter chat room](#). Please include at least the `preCICE-config.xml` file whenever you ask for support! Also, give us some feedback!

You don't know yet what preCICE is? Have a look at our [website](#), [our reference paper](#), [our literature guide](#), or [these slides](#).

Missing a feature? Maybe it is already on our [Roadmap](#)! If not, [tell us](#)!

### 1. Get preCICE

If you want to get preCICE on your Linux desktop, you may find [binary packages with our releases](#).

If the binary packages are not enough for you, you can [build preCICE using CMake](#), or explore [more options to get preCICE](#).

After installing preCICE, you may need to have a look at the page [Linking to preCICE](#).

You can quickly check your installation using the bundled examples. Copy the folder `examples/` (e.g. from `usr/share/preCICE`) to a local directory and build and run one of our [solver dummies](#).

**Pages** 76







- Wiki home
- Search the wiki
- 1. Get preCICE**
  - Binary packages
  - Build with CMake
  - Linking to preCICE
  - Get preCICE from Spack
  - More options
  - Dependencies
- 2. Getting started - Tutorials**
- Basics**
  - Web-based FSI tutorial
  - FSI: 1D elastic tube (...)
- HT: Heat equation**
  - FEniCS (x2)
- CHT: Flow over plate**



# Discuss & get help (threaded)

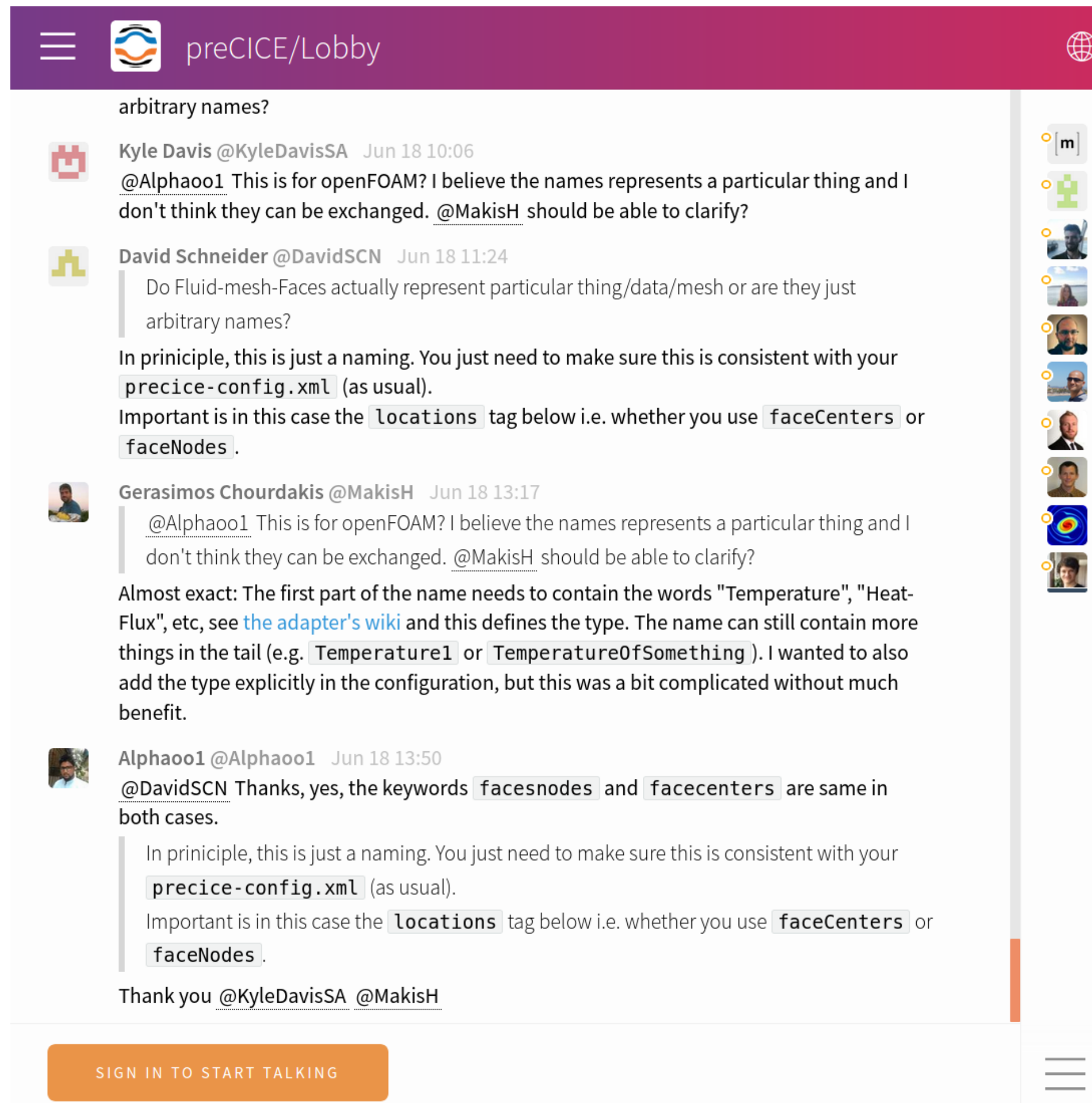
all categories all tags all Categories Latest Top

Category	Topics
<b>News</b> News, announcements, "blog"-like posts	4
<b>Is preCICE for me?</b> General questions regarding preCICE as a coupling solution.	11
<b>Installing preCICE</b> Any issues with getting the preCICE library installed	15
<b>Using preCICE</b> Using the preCICE API, configuring a new simulation	44
<b>Official adapters and tutorials</b> Installing, configuring, extending adapters, running or modifying the tutorials	33
<b>Site Feedback</b> Discussion about this site, its organization, how it works, and how we can improve it.	2
<b>Uncategorized</b> Topics that don't need a category, or don't fit into any other existing category.	1

Latest	
 <input checked="" type="checkbox"/> Welcome to the preCICE Forum on Discourse ■ Site Feedback	2 Nov '19
 <input type="checkbox"/> Status of Quads on the Interface ■ Is preCICE for me?	2 3h
 <input checked="" type="checkbox"/> Energy conservation through two independent consistent mappings ■ Using preCICE ■ configuration	4 9d
 <input type="checkbox"/> IQN fails for quickly changing forces during wetting in a free-surface FSI problem ■ Using preCICE	3 9d
 <input checked="" type="checkbox"/> Assertion failed on ResidualSumPreconditioner ■ Using preCICE	3 9d
 <input checked="" type="checkbox"/> RBF mapping vs. nearest projection mapping ■ Using preCICE ■ openfoam ■ data-mapping ■ calculix ■ fsi	3 14d



# Discuss & get help (quick)



The screenshot shows a GitHub discussion thread in the preCICE/Lobby channel. The thread is titled "arbitrary names?". It features four messages from Kyle Davis, David Schneider, Gerasimos Chourdakis, and Alphaoo1, discussing naming conventions for configuration files and the importance of consistency in terms like "faceCenters" and "faceNodes".

**preCICE/Lobby**

arbitrary names?

**Kyle Davis @KyleDavisSA** Jun 18 10:06  
@Alphaoo1 This is for openFOAM? I believe the names represents a particular thing and I don't think they can be exchanged. @MakisH should be able to clarify?

**David Schneider @DavidSCN** Jun 18 11:24  
Do Fluid-mesh-Faces actually represent particular thing/data/mesh or are they just arbitrary names?  
In principle, this is just a naming. You just need to make sure this is consistent with your `precice-config.xml` (as usual).  
Important is in this case the `locations` tag below i.e. whether you use `faceCenters` or `faceNodes`.

**Gerasimos Chourdakis @MakisH** Jun 18 13:17  
@Alphaoo1 This is for openFOAM? I believe the names represents a particular thing and I don't think they can be exchanged. @MakisH should be able to clarify?  
Almost exact: The first part of the name needs to contain the words "Temperature", "Heat-Flux", etc, see [the adapter's wiki](#) and this defines the type. The name can still contain more things in the tail (e.g. `Temperature1` or `TemperatureOfSomething`). I wanted to also add the type explicitly in the configuration, but this was a bit complicated without much benefit.

**Alphaoo1 @Alphaoo1** Jun 18 13:50  
@DavidSCN Thanks, yes, the keywords `facesnodes` and `facecenters` are same in both cases.  
In principle, this is just a naming. You just need to make sure this is consistent with your `precice-config.xml` (as usual).  
Important is in this case the `locations` tag below i.e. whether you use `faceCenters` or `faceNodes`.  
Thank you @KyleDavisSA @MakisH

**SIGN IN TO START TALKING**





# Get news: Twitter



**preCICE**  
@preCICE\_org

A free/open-source coupling library for partitioned multi-physics simulations, including fluid-structure interaction, conjugate heat transfer, and more.

📍 Germany 🔗 [precice.org](https://precice.org) 📅 Joined April 2018

**54** Following   **174** Followers

[Follow](#)



# Get news: Mailing list

## preCICE -- preCICE Parallel Coupling Environment

### About preCICE

English (USA)

To see the collection of prior postings to the list, visit the [preCICE Archives](#).

### Using preCICE

To post a message to all the list members, send email to [precice@mailman.informatik.uni-stuttgart.de](mailto:precice@mailman.informatik.uni-stuttgart.de).

You can subscribe to the list, or change your existing subscription, in the sections below.

### Subscribing to preCICE

Subscribe to preCICE by filling out the following form. You will be sent email requesting confirmation, to prevent others from gratuitously subscribing you. This is a private list, which means that the list of members is not available to non-members.

Your email address:

Your name (optional):

You may enter a privacy password below. This provides only mild security, but should prevent others from messing with your subscription. **Do not use a valuable password** as it will occasionally be emailed back to you in cleartext.



# Advertisements



# preCICE Workshop 2021



preCICE Workshop 2020 @ TUM





# See also

## Training III-E: Multiphysics Modeling with the preCICE Coupling Library: Coupling OpenFOAM with CalculiX/deal.II

📅 Wed. Jun 24, 2020 ⌚ 8:30 AM - 10:00 AM



**Gerasimos Chourdakis**

Technical University of Munich, Germ...

Send Message

## Training II-E: OpenFOAM for Multiphysics Applications

📅 Tue. Jun 23, 2020 ⌚ 8:30 AM - 10:00 AM



**Holger Marschall**

Technical University Darmstadt, Germ...

Send Message



# preCICE is free because of



H2020 grant 754462



and the code/issues/testing/documentation contributions of people like you (thank you!).

# Summary

Easily exchange your coupled solvers and explore advanced  
& performant numerical algorithms

Gerasimos Chourdakis (TUM)  
Benjamin Uekermann (TU/e)  
+ many more (see [precice.org/about](https://precice.org/about))

🗨 Write me at [chourdak@in.tum.de](mailto:chourdak@in.tum.de)  
or start a thread on [our forum](#)



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).  
Get this presentation from [GitHub](#).