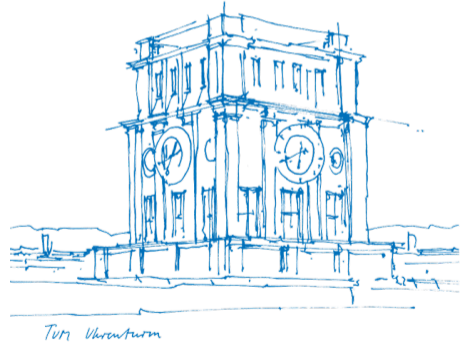


Testing the multi-component preCICE ecosystem

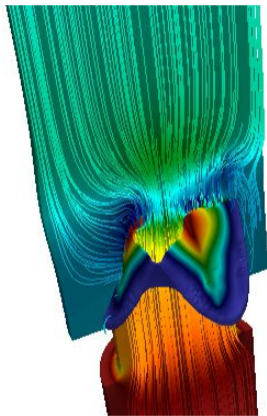
SIGDIUS seminar (online)

Gerasimos Chourdakis
Technical University of Munich

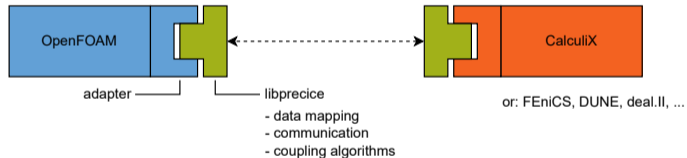
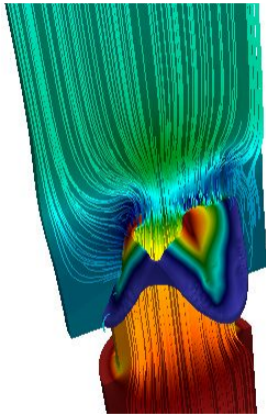
May 4, 2022



Partitioned multi-physics simulations

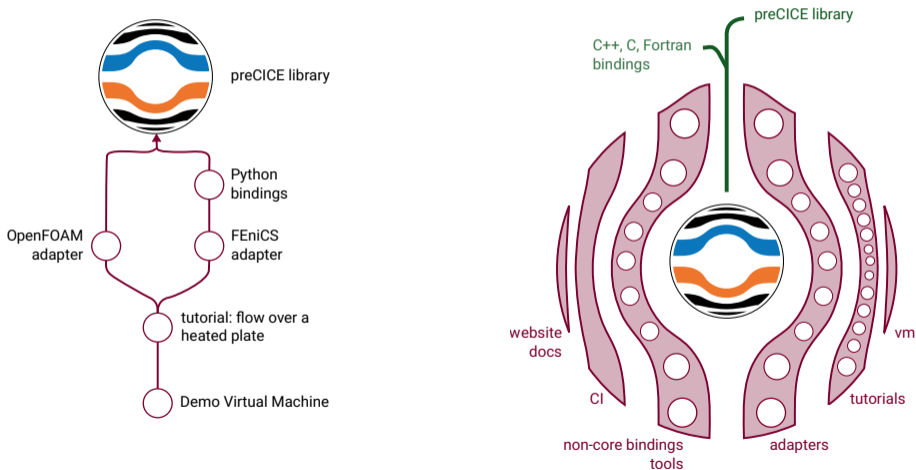


Partitioned multi-physics simulations



Left: "Evaluation of heart valve biomechanics", Kyle Davis (Univ. Free State, South Africa) – preCICE Community Stories on [precice.org](https://www.precice.org)

The multi-component preCICE ecosystem



Different levels of tests

Unit tests

```
enum struct Require { PETSc };

BOOST_TEST(ParallelPETScFeature) {
    // Test using 2 ranks and requiring PETSc to be setup.
    PRECICE_TEST(2_ranks, Require::PETSc);
    // test feature
}
```

Integration tests

```
PRECICE_TEST("Fluid"_on(2_ranks), "Solid"_on(1_rank));
    ^^^^^^^^.....^..... Participant names
      ^^^.....^..... String-literal
        ^.....^..... operator()
          ^^^^^^^^..... Rank information
```

Different levels of tests



better scientific software Resources ▾ Blog Events About ▾ 🔍

[HOME](#) > [BLOG](#) > [Overcoming Complexity in Testing Multiphysics Coupling...](#)

Overcoming Complexity in Testing Multiphysics Coupling Software

SHARE in [f](#) [t](#) [s](#)

Testing complex software can easily get out of hand, especially when your product is a multiphysics coupling library. Fortunately, we've found some strategies that have helped tame the nightmare.

PUBLISHED FEB 07, 2022

AUTHORS [FRÉDÉRIC SIMONIS](#), [GERASIMOS CHOURDAKIS](#), AND [BENJAMIN UEKERMANN](#)

TOPICS [BETTER PLANNING](#) [BETTER RELIABILITY](#) [BETTER DEVELOPMENT](#) [SOFTWARE INTEROPERABILITY](#) [TESTING](#) [DEVELOPMENT TOOLS](#)

Different levels of tests

System tests

Testing the complete system for regressions:

- Using multiple layers together
- Running complete tutorials
- Comparing results

(regression) testing \neq validation!

But preCICE already has system tests! ...right?

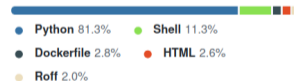
Tests 1 hr 31 min 36 sec

✓	# 2959.18	AMD64	Bionic	[18.04] OpenFOAM <-> OpenFOAM	5 min 16 sec
✓	# 2959.19	AMD64	Bionic	[18.04] Calculix <-> OpenFOAM	6 min 43 sec
✓	# 2959.20	AMD64	Bionic	[18.04] FEniCS <-> FEniCS	4 min 40 sec
✓	# 2959.21	AMD64	Bionic	[18.04] Bindings/Solverdummies	5 min 10 sec
✓	# 2959.22	AMD64	Bionic	[18.04] deal.II <-> OpenFOAM [FSI][3D]	8 min 2 sec
✓	# 2959.23	AMD64	Bionic	[18.04] deal.II <-> OpenFOAM [FSI][2D]	7 min 43 sec
✓	# 2959.24	AMD64	Bionic	[20.04 PETS] deal.II <-> OpenFOAM [FSI][2D]	6 min 59 sec
✓	# 2959.25	AMD64	Bionic	[18.04] OpenFOAM <-> OpenFOAM [nearest projection]	5 min 13 sec
✓	# 2959.26	AMD64	Bionic	[18.04] OpenFOAM <-> Code_Aster	5 min 21 sec
✓	# 2959.27	AMD64	Bionic	[18.04] ElasticaTube1D - Python	6 min 47 sec
✓	# 2959.28	AMD64	Bionic	[18.04] ElasticaTube1D - C++	4 min 9 sec
✗	# 2959.29	AMD64	Bionic	[18.04] SU2 <-> Calculix [unstable][failure allowed]	7 min 40 sec
✗	# 2959.30	AMD64	Bionic	[20.04 PETS] SU2 <-> Calculix [unstable][failure allowed]	6 min 51 sec
✗	# 2959.31	AMD64	Bionic	[18.04] nutils <-> OpenFOAM [failure allowed]	7 min 58 sec
✓	# 2959.32	AMD64	Bionic	[18.04 PETS] OpenFOAM <-> Calculix [FSI][failure allowed]	4 min 58 sec

Contributors 9



Languages



Since June 15th, 2021, the building on travis-ci.org is ceased. Please use travis-ci.com from now on.

But preCICE already has system tests! ...right?

Travis CI [Dashboard](#) [Changelog](#) [Documentation](#) [Help](#)

We are unable to start your build at this time. You exceeded the number of users allowed for your plan. Please review your plan details and follow the steps to resolution.

Search all repositories

My Repositories **Running (0/0)** +

- preCICE/systemtests** # 3411
 - Duration: 1 hr 14 min 1 sec
 - Finished: 8 months ago
- preCICE/python-bindings** # 824
 - Duration: 1 min 37 sec
 - Finished: 10 months ago
- preCICE/mpi-adapter** # 231
 - Duration: 12 min 56 sec
 - Finished: 10 months ago
- preCICE/calculix-adapter** # 105
 - Duration: 4 min 53 sec
 - Finished: 10 months ago
- preCICE/ferics-adapter** # 840
 - Duration: 6 min 1 sec
 - Finished: 10 months ago
- preCICE/openfoam-adapter** # 266
 - Duration: 13 min 51 sec
 - Finished: 10 months ago

preCICE / systemtests **Failed Building**

Current Branches Build History Pull Requests More options

develop **CRON** Generate results with preCICE 2.2.0 and PETSc RBF Mapping. **#3411 failed** Restart build

- Commit 8eabc93
- Branch devvs1ep

BenjaminRodenberg authored and committed

Build jobs View config

Building preCICE 26 min 54 sec

# 3411.1	AMD64	Bionic	Arch Linux	48 sec
# 3411.2	AMD64	Bionic	Ubuntu 18.04 home [PETSc from source]	26 min 53 sec
# 3411.3	AMD64	Bionic	Ubuntu 18.04.package	25 min 21 sec
# 3411.4	AMD64	Bionic	Ubuntu 20.04.package [PETSc from APT]	20 min 59 sec

Building adapters

# 3411.5	AMD64	Bionic	[18.04] SU2 adapter	-
# 3411.6	AMD64	Bionic	[20.04] SU2 adapter [PETSc]	-

Manual testing to the rescue

Merged **Release v2.3.0** #1095
 fsmionis merged 313 commits into `master` from `release-v2.3.0` on Oct 6, 2021

Regression Tests

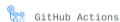
Assign each point below to a responsible person, before you continue. Use `@member`.

Run all these tests manually on your system. If you succeed, please write a comment with the revisions of the components that you used below. Example: `#507 (comment)` and update the table.

State	Success	Failure	Skipped
Write	<code>:o:</code>	<code>:x:</code>	<code>:fast_forward:</code>
Read	○	✘	⏸

State	Tester	Test
○	@IshaanDesai	perpendicular-flap fluid-nutils - solid-calculix
○	@DavidSCN	perpendicular-flap fluid-openfoam - solid-dealii
○	@IshaanDesai	perpendicular-flap fluid-su2 - solid-fenics
○	@DavidSCN	multiple-perpendicular-flaps fluid-openfoam - solid-(left+right)-dealii
○	@MakisH	flow-over-heated-plate fluid-openfoam - solid-openfoam serial + parallel
○	@IshaanDesai	flow-over-heated-plate fluid-openfoam - solid-fenics serial + parallel
○	@IshaanDesai	flow-over-heated-plate fluid-openfoam - solid-nutils
○	@MakisH	flow-over-heated-plate-nearest-projection fluid-openfoam - solid-openfoam
○	@IshaanDesai	flow-over-heated-plate-steady-state fluid-openfoam - solid-codeaster
○	@MakisH	heat-exchanger fluid-(inner+outer)-openfoam - solid-calculix

New tools at hand

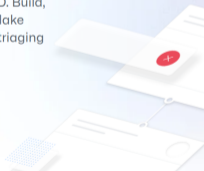


Automate your workflow from idea to production

GitHub Actions makes it easy to automate all your software workflows, now with world-class CI/CD. Build, test, and deploy your code right from GitHub. Make code reviews, branch management, and issue triaging work the way you want.

[Get started with Actions >](#)

Questions? [Contact Sales >](#)



GitLab Continuous Integration (CI)

Outstanding source code exists. For teams and projects big and small, the proof is in the pipeline.

[Get to know CI/CD](#)



GitHub Actions: `precice/precice`

Workflows

[New workflow](#)

All workflows

`.github/workflows/macos-wor...`

Build and Test

Build and Test macOS

Build and store a Docker ima...

Code Style

Microsoft C++ Code Analysis

Release Pipeline

Request Package

clang-tidy

coverity-scan

All workflows

Showing runs from all workflows

2,575 workflow runs

Event ▾ Status ▾ Branch ▾ Actor ▾

<p> coverity-scan coverity-scan #112: Scheduled</p>			<p> yesterday 1h 5m 45s</p>	...
<p> Fix error message. Closes #1188. Build and Test #1374: Commit 42437f3 pushed by BenjaminRodenberg</p>	<code>develo</code>		<p> 2 days ago 20m 44s</p>	...
<p> Waveform interpolation, read direction for parallel im... Build and Test #1373: Pull request #1187 synchronize by BenjaminRodenberg</p>	<code>BenjaminRodenberg:wavefor...</code>		<p> 3 days ago 56m 42s</p>	...
<p> Waveform interpolation, read direction for parallel im... Code Style #1080: Pull request #1187 synchronize by BenjaminRodenberg</p>	<code>BenjaminRodenberg:wavefor...</code>		<p> 3 days ago 33m 6s</p>	...

GitHub Actions: precice/openfoam-adapter

Workflows

[New workflow](#)

All workflows

- Build and store a Docker ima...
- Build with OpenFOAM v2112
- Check code formatting
- Check links (manual)
- Custom build (manual)
- Lint docs
- Lint scripts
- Lint scripts and docs

All workflows

Showing runs from all workflows

✕

742 workflow run results	Event ▾	Status ▾	Branch ▾	Actor ▾
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"></div> <div> <p>Extend Allwmake suggestions for possible problems</p> <p>Lint scripts #374: Pull request #220 opened by MakisH</p> </div> </div>	MakisH:extend-allwmake-su...	🕒 3 days ago	🕒 19s	...
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"></div> <div> <p>Extend Allwmake suggestions for possible problems</p> <p>Lint docs #368: Pull request #220 opened by MakisH</p> </div> </div>	MakisH:extend-allwmake-su...	🕒 3 days ago	🕒 29s	...
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"></div> <div> <p>Extend Allwmake suggestions for possible problems</p> <p>Check code formatting #382: Pull request #220 opened by MakisH</p> </div> </div>	MakisH:extend-allwmake-su...	🕒 3 days ago	🕒 46s	...
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"></div> <div> <p>Extend Allwmake suggestions for possible problems</p> <p>Build with OpenFOAM v2112 #438: Pull request #220 opened by MakisH</p> </div> </div>	MakisH:extend-allwmake-su...	🕒 3 days ago	🕒 14m 6s	...
<div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"></div> <div> <p>Update master (v1.1.0)</p> <p>Lint scripts #354: Pull request #186 synchronize by MakisH</p> </div> </div>	develop	🕒 12 days ago	🕒 21s	...

GitHub Actions: Manual triggering



The image shows a screenshot of the GitHub Actions interface for manually triggering a workflow. At the top right, there is a button labeled "Run workflow" with a dropdown arrow. Below this, a modal dialog box is open, containing the following configuration options:

- Use workflow from**
 - Branch: develop ▾
- Virtual Environment ***
 - ubuntu-18.04 ⌵
- Ref (branch/tag/commit) of the OpenFOAM adapter to build ***
 - develop
- Version of OpenFOAM to build with ***
 - OpenFOAMv2112 ⌵
- Version of preCICE to build with ***
 - 2.3.0
- Run tutorial flow-over-heated-plate
- Run tutorial quickstart
- Run tutorial partitioned-pipe
- Branch of the tutorials to use ***
 - master
-

GitHub Actions: How does the config look like?

```
# File .github/workflows/build-custom.yml
name: Custom build (manual)
on:
  workflow_dispatch:
    inputs:
      # ...
    versionOpenFOAM:
      type: choice
      options:
        - OpenFOAMv2112
        - OpenFOAM8
  jobs:
    build:
      runs-on: ubuntu:latest
      steps:
        - name: Run something
          run: ./run.sh
```

preCICE	Section 5.1	Section 5.4	Section 5.1	Section 5.1	Section 5.1	Section 5.4
	0110010 warning 01error	<ul style="list-style-type: none"> formatting clang-format tests coverage lcov, CodeCov code analysis codacy, codefactor coverity-scan, lgtm 				
	GCC, Clang		Boost.test	Boost.test + MPI	solver dummies	CPack, lintian (+ Spack)
	building	quality assurance	unit tests	integration tests	smoke tests	packaging

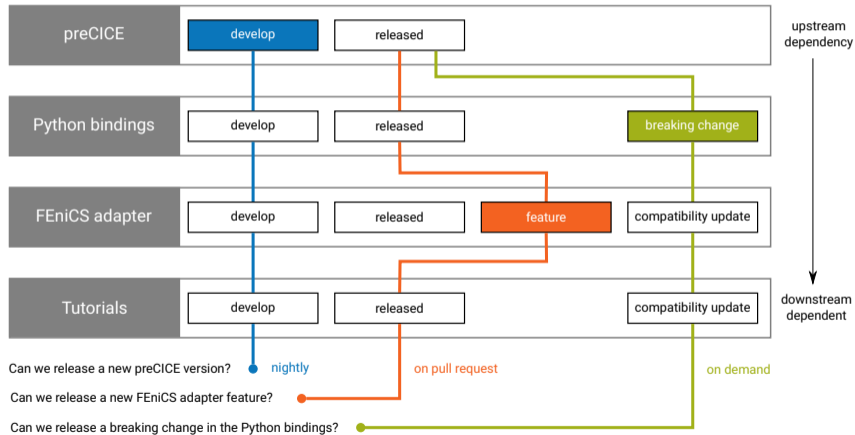
Python bindings	Section 5.2	Section 5.2	Section 5.2	Section 5.2	Section 5.2	Section 5.2
						<ul style="list-style-type: none"> formatting PEP8, markdownlint publishing Docker, twine, PyPI
	unittest	fake preCICE lib	unittest	MagicMock	FEniCS shell	more
	unit tests	integration tests	unit tests	integration tests	system tests	more

MATLAB bindings (under construction)	OpenFOAM adapter build, format, lint scripts	deal.II adapter build	CalculiX adapter (under construction)	SU2 adapter (under construction)	code_aster adapter (under construction)
---	---	--------------------------	--	-------------------------------------	--

Tutorials	Section 5.3					Section 5.4
						<ul style="list-style-type: none"> linting shellcheck formatting PEP8, markdownlint links check markdown-link-check
	flow over heated plate	perpendicular flap	Turek-Hron FSI3	partitioned heat eq.	partitioned pipe	more
	system / regression tests					more

vm	ch-img	doxy	.org
Section 5.4 lint scripts, build, package	Section 5.4 Docker images for ubuntu, centos, arch	Section 5.4 publish Doxygen docs	Section 5.4 format, build, deploy

Different perspectives

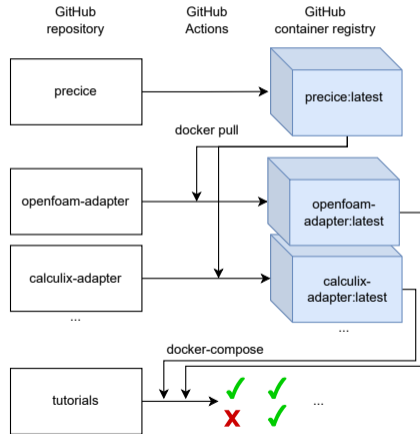


Current state

- ☑ Prototype
 - Running locally and on GitHub Actions
 - Only latest version of specific branches
 - Only preCICE, OpenFOAM adapter, and tutorials
 - Only flow-over-heated-plate OpenFOAM-OpenFOAM
 - No results comparison
- Results comparison
- Manual triggering for any branches (tags)
- Extension to all repositories and tutorials
- Automated triggering

Pull Requests [precice/1090](#), [openfoam-adapter/184](#), [tutorials/220](#)

Architecture



Docker Compose

```
# File: docker-compose.yml
services:
  # ...
  fluid-openfoam:
    image: "ghcr.io/precice/of-adapter:
           ${TAG_OPENFOAM_ADAPTER}"
    user: "${MY_UID}:${MY_GID}"
    volumes:
      - /etc/passwd:/etc/passwd:ro
      - /etc/group:/etc/group:ro
      - ../../:/tests
    command: >
             /bin/bash -c "openfoam2112 ./run.sh"
```

Docker Compose

```
# File: docker-compose.yml
services:
  # ...
  fluid-openfoam:
    image: "ghcr.io/precice/of-adapter:
      ${TAG_OPENFOAM_ADAPTER}"
    user: ${MY_UID}:${MY_GID}
    volumes:
      - /etc/passwd:/etc/passwd:ro
      - /etc/group:/etc/group:ro
      - ../../:/tests
    command: >
      /bin/bash -c "openfoam2112 ./run.sh"

  solid-openfoam:
    image: # same as fluid-openfoam
    user: # same
    volumes:
      - # same
    command: # same, different directory
```

Docker Compose: Running

```
cd tutorials/flow-over-heated-plate/tests
```

```
export TAG_OPENFOAM_ADAPTER=latest
```

```
MY_UID="$(id -u)" MY_GID="$(id -g)" docker-compose up
```

GitHub Container Registry

openfoam-adapter

 Install from the command line:

[Learn more](#)

```
$ docker pull ghcr.io/precice/openfoam-adapter:latest
```

Recent tagged image versions

latest

 8

Published 12 days ago · Digest 

[View and manage all versions](#)

Running the tutorials: logs

The screenshot shows a GitHub repository named 'preceice / tutorials' with a public workflow. The workflow is titled 'Add a cleanup step Run tutorials with docker-compose #19'. The workflow run 'run-flow-over-heated-plate' is shown as successful, having completed 8 steps in 68 seconds. The steps are: Set up job (1s), Checkout repository (2s), Run docker-compose (49s), Archive logs (1s), Archive case files (5s), Post Checkout repository (0s), and Complete job (0s).

preceice / tutorials Public

Unpin Unwatch 8 Fork 55 Starred 51

Code Issues 22 Pull requests 3 Discussions Actions Projects Security Insights Settings

✓ Add a cleanup step Run tutorials with docker-compose #19 Re-run all jobs

Summary

Jobs

✓ run-flow-over-heated-plate

run-flow-over-heated-plate succeeded 12 days ago in 58s

Search logs

- > ✓ Set up job 1s
- > ✓ Checkout repository 2s
- > ✓ Run docker-compose 49s
- > ✓ Archive logs 1s
- > ✓ Archive case files 5s
- > ✓ Post Checkout repository 0s
- > ✓ Complete job 0s

Running the tutorials: artifacts

precice / tutorials Public

Code Issues 22 Pull requests 3 Discussions Actions Projects Security Insights Settings

✓ Add a cleanup step Run tutorials with docker-compose #19
 Re-run all jobs

Summary

Jobs

- run-flow-over-heated-plate

Triggered via push 12 days ago	Status	Total duration	Artifacts
MakisH pushed → 8a77be3 <code>add-nightly-tests</code>	Success	1m 10s	2

```

run-tutorials-compose.yml
on: push

run-flow-over-heated-plate 58s
    
```

Artifacts

Produced during runtime

Name	Size
case-files	6.63 MB
logs	155 KB

What to compare to?

Reference data:

- Solver logs → often not identical
- Solver results → various formats, too much
- preCICE exports → same format, no time-related noise, enough

(demonstrated in TUM FSI Seminar paper 2020 by Mohamad Kanj)

What to compare to?

Reference data:

- Solver logs → often not identical
- Solver results → various formats, too much
- preCICE exports → same format, no time-related noise, enough

(demonstrated in TUM FSI Seminar paper 2020 by Mohamad Kanj)

Tutorials structure extension:

- flow-over-heated-plate/
 - fluid-openfoam/
 - solid-openfoam/
 - precice-config.xml
 - reference-data/
 - fluid-openfoam_solid-openfoam/
 - tests/
 - docker-compose.yml
- tools/
 - run-tests.sh

So, how is the new system better?

	Old	New
Tools	Travis CI + local	GitHub Actions + local
Local runners	–	Possible (proprietary solvers)
Container storage	DockerHub	GHCR
Artifacts storage	Dedicated repo	Next to the job
What to test	All possible combination	Selected by default, any on demand
Goal	Many features, coverage	Minimal code, sustainability

Key reference (fresh!)


 Search

Research and Innovation

Open Research Europe

SUBMIT YOUR RESEARCH

[Browse](#) [Gateways & Collections](#) [How to Publish](#) [About](#) [Blog](#)

[Sign in](#)

8 Views | 8 Downloads | 0 Citations

[Cite](#) [Download](#) [Export](#) [Share](#) [Track](#)

[Home](#) > [Articles](#) > [preCICE v2: A sustainable and user-friendly coupling library](#)

SOFTWARE TOOL ARTICLE

preCICE v2: A sustainable and user-friendly coupling library [version 1; peer review: awaiting peer review]

Gerasimos Chourdakis , Kyle Davis , Benjamin Rodenberg , Miriam Schulte , Frédéric Simonis , Benjamin Uekermann [✉](#) , Georg Abrams, Hans-Joachim Bungartz, Lucia Cheung Yau, Ishaan Desai , Konrad Eder, Richard Hertrich, Florian Lindner , Alexander Rusch , Dmytro Sashko, David Schneider , Amin Totounferoush , Dominik Volland, Peter Vollmer , Oguz Ziya Koseomur

This article is included in Excellent Science gateway



Open Peer Review

Approval Status

AWAITING PEER REVIEW

Comments on this article

All Comments (0)

[Sign in to comment](#)

Summary

- the library is tested widely, the ecosystem follows (complex)
- system tests grew, but hit by bus and vendor lock-in
- new, simpler implementation based on new tools and previous lessons
- work in progress

Read more in preCICE v2 paper (in review) and BSSW blog.

Gerasimos Chourdakis (@MakisH)
gerasimos.chourdakis@tum.de

Give me feedback: go.tum.de/820189