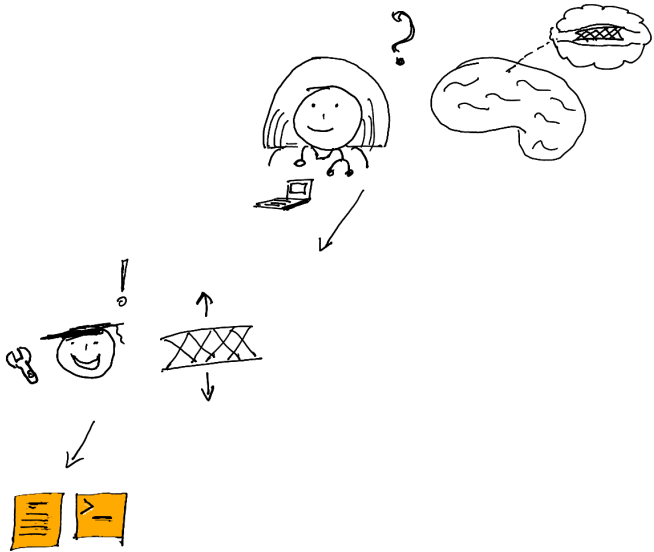
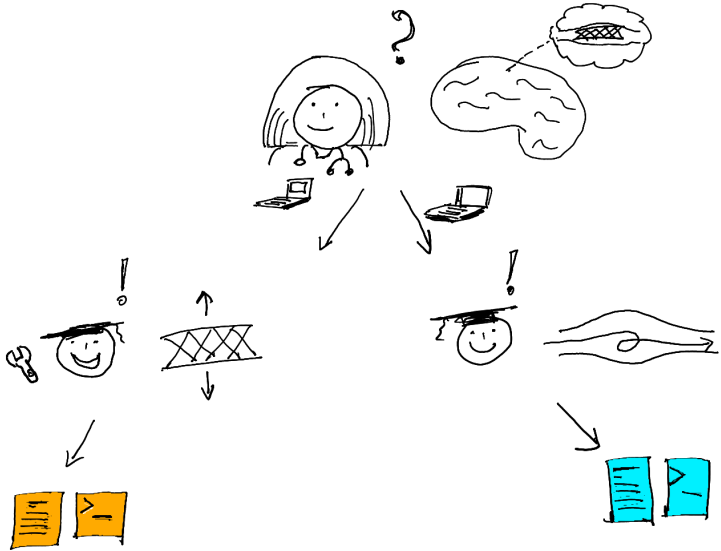
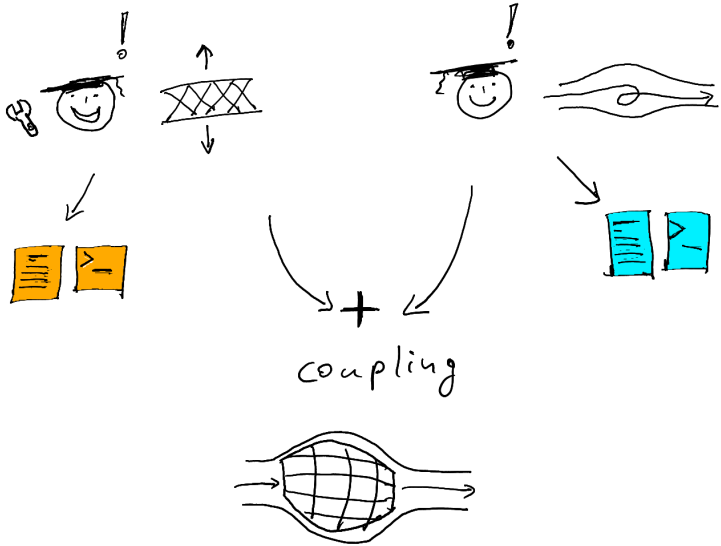
University of Stuttgart
Institute for Parallel and Distributed Systems

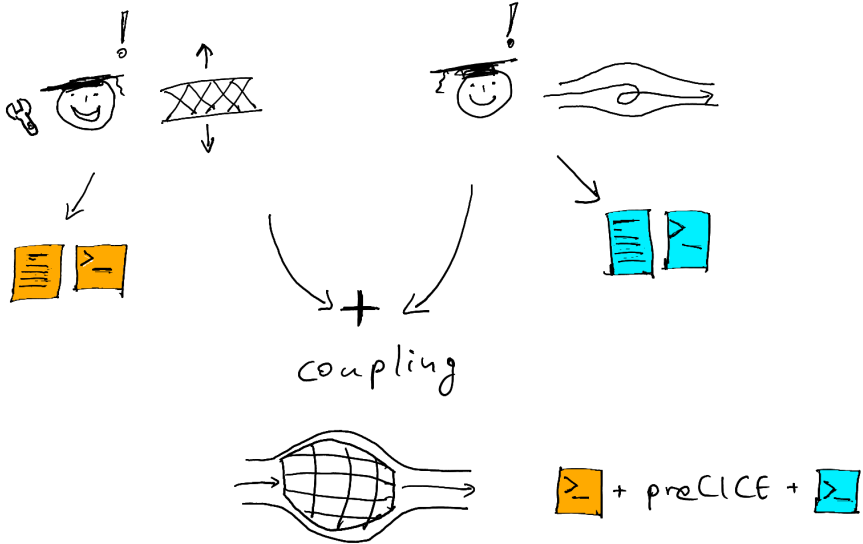
Gerasimos Chourdakis

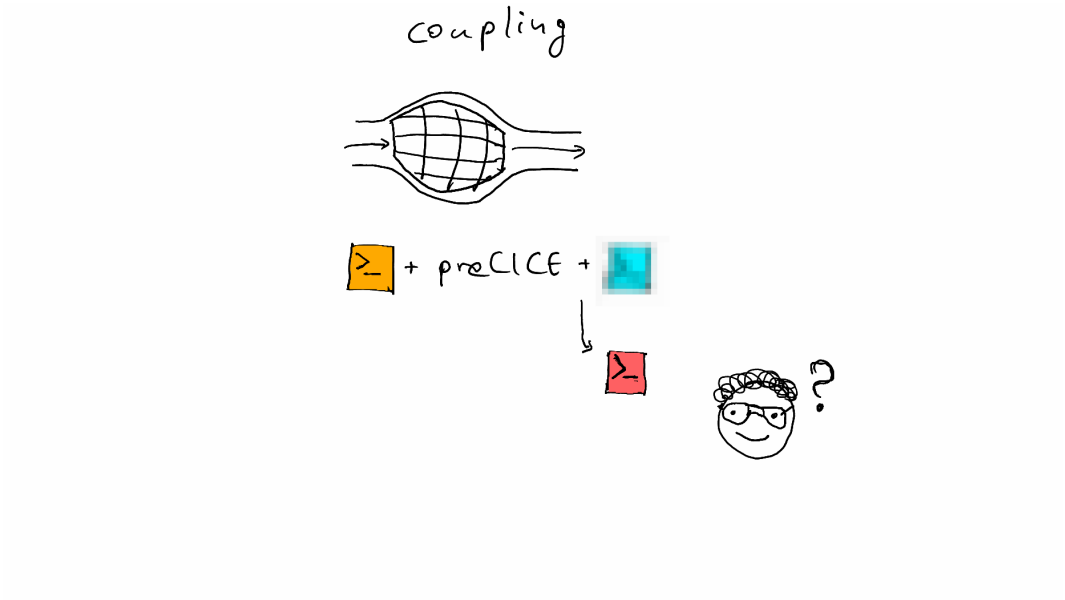
Standardizing the preCICE ecosystem






coupling


coupling
preCICE


coupling
+ preCICE +
?

The preCICE ecosystem

Community numbers:

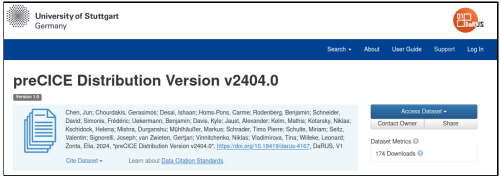
- 10 team members
- 50+ official repos (`github.com/precice`)
- 20+ third-party adapters
- 550+ forum members
- ~50 workshop participants p.a.

The preCICE ecosystem

Community numbers:

- 10 team members
- 50+ official repos (`github.com/precice`)
- 20+ third-party adapters
- 550+ forum members
- ~50 workshop participants p.a.

Bundle of stable official components: preCICE Distribution



v2404.0

doi 10.18419/darus-4167

This is the first release that includes preCICE v3.x.

It comprises the following components:

- preCICE: v3.1.1
- Tools:
  - ASTE: v3.1.0
  - ci-images: commit b421a49
  - config-visualizer: v1.1.3
  - config-visualizer-gui: v0.1.0 (new)
  - FMI runner: v0.2.1
  - Micro manager: v0.4.0
- Bindings:
  - Fortran module: commit dc88c3b
  - Julia bindings: v3.1.0
  - Matlab bindings: v3.1.0 (now three-digit)
  - Python bindings: v3.1.0 (now three-digit)
  - Rust bindings: v3.1.0 (new)
- Adapters:
  - CalculiX adapter: v2.20.1
  - code_aster adapter: commit b797fcc
  - deal.II adapter: 4c6d092
  - DUNE adapter: commit 75edcc3
  - DuMuX adapter: v2.0.0 (new)
  - FEniCS adapter: v2.1.0
  - OpenFOAM adapter: v1.3.0
  - SU2 adapter: 64d4aff (largely re-written and updated)
- Tutorials: v202404.0
- vm: v202404.0.0
- Website and documentation: v202404.0.0, libprecice3_3.3.1_docs_v202404.0.0.pdf

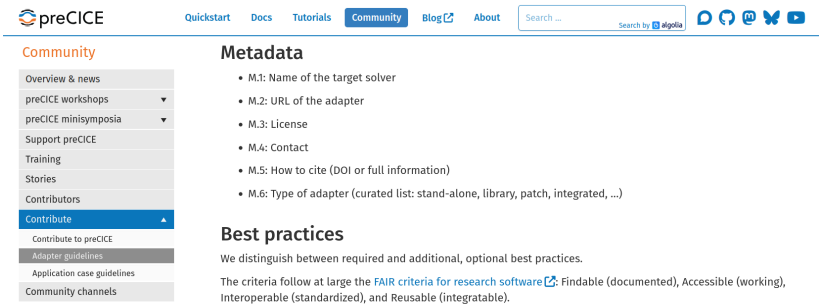
`https://doi.org/10.18419/darus-4167`


clicks
Abcdefg
°F lb ft
run.sh
°C kg m

Defining standards with the community

https://precice.org/community-guidelines-adapters.html

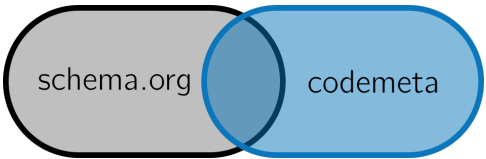
Defining standards with the community



https://precice.org/community-guidelines-adapters.html

preCICE metadata

Idea: Extend `codemeta.json` with preCICE-specific terms



preCICE metadata

Idea: Extend `codemeta.json` with preCICE-specific terms

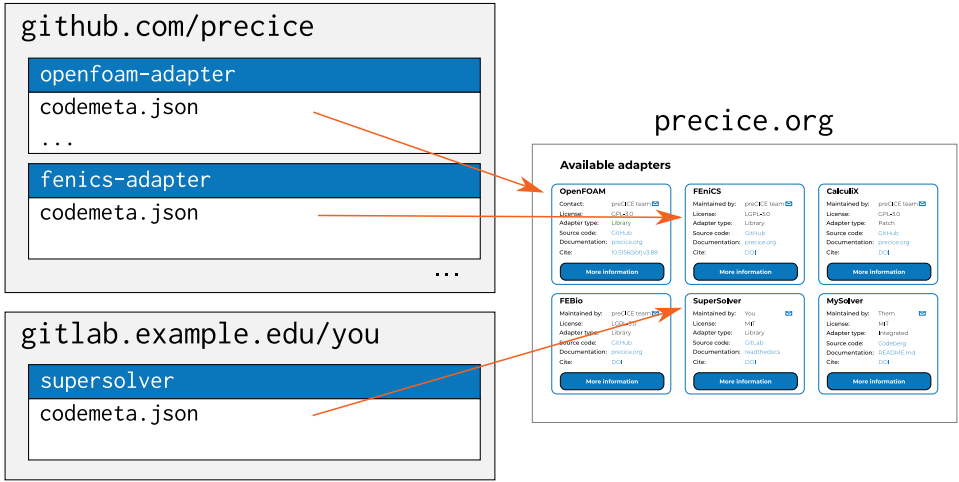
A preCICE metadata file in each repository (concept)

In a `codemeta.json` file:

```
{
"@context": {
  "codemeta": "https://doi.org/10.5063/schema/codemeta-2.0",
  "precice": "http://precice.org/schema/precicemeta-1.0"
  },
  "@type": "SoftwareSourceCode",
  "codemeta:name": "My preCICE adapter for some solver",
  "precice:apiVersion": "3"
}
```

Q: What if an adapter is directly implemented in a larger software package?

From metadata to a rendered list



Also: easier software publication, SBOM generation, ...

Goal: A “preCICE store” of plug-and-play components

Available adapters

OpenFOAM

Contact:	preCICE team
License:	GPL-3.0
Adapter type:	Library
Source code:	GitHub
Documentation:	precice.org
Cite:	10.51560/ofj.v3.88

More information

FEniCS

Maintained by:	preCICE team
License:	LGPL-3.0
Adapter type:	Library
Source code:	GitHub
Documentation:	precice.org
Cite:	DOI

More information

CalculiX

Maintained by:	preCICE team
License:	GPL-3.0
Adapter type:	Patch
Source code:	GitHub
Documentation:	precice.org
Cite:	DOI

More information

FEBio

Maintained by:	preCICE team
License:	LGPL-3.0
Adapter type:	Library
Source code:	GitHub
Documentation:	precice.org
Cite:	DOI

More information

SuperSolver

Maintained by:	You
License:	MIT
Adapter type:	Library
Source code:	GitLab
Documentation:	readthedocs
Cite:	DOI

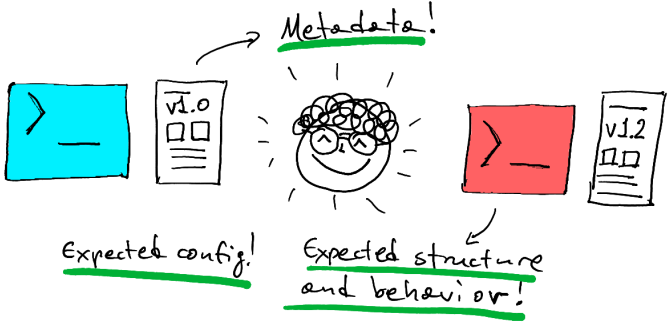
More information

MySolver

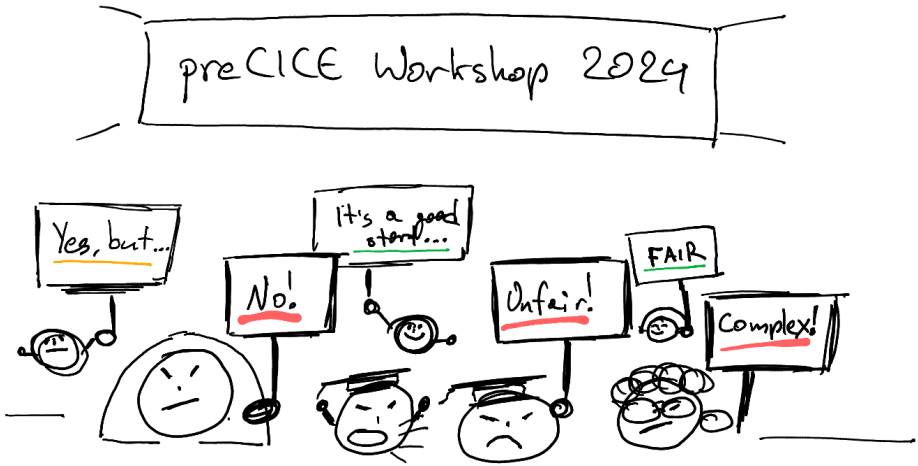
Maintained by:	Them
License:	MIT
Adapter type:	Integrated
Source code:	Codeberg
Documentation:	README.md
Cite:	DOI

More information

Review process: Expectation



Review process: User reaction

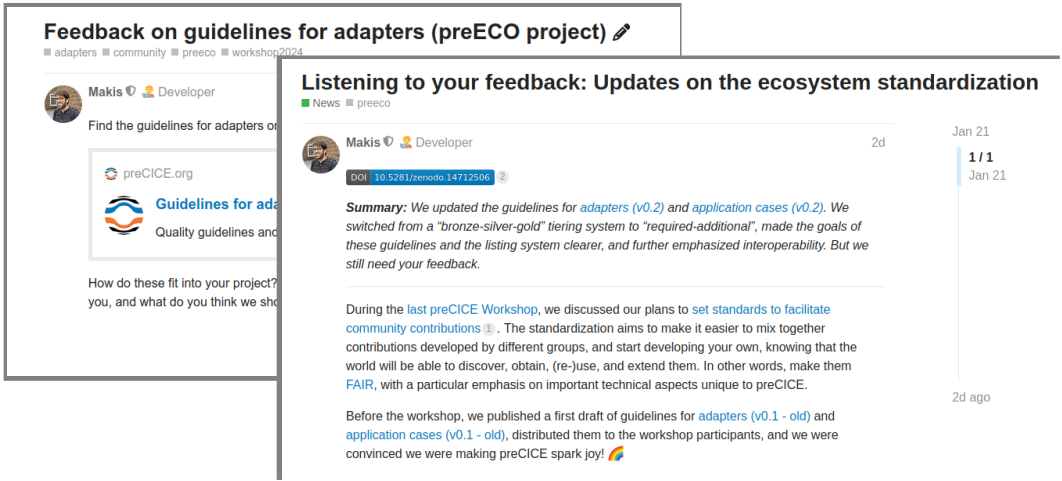


Review process: Collecting feedback

Annual preCICE Workshop: `https://precice.org/precice-workshop.html`

Reviewing pilot project examples on GitHub – Everything open

Review process: Collecting feedback



https://precice.discourse.group/

Outlook

Next steps:

- Implement a first metadata schema
- Implement the rendering on the website
- Discuss guidelines with the community
- Develop training for best practices

Outlook

Next steps:

- Implement a first metadata schema
- Implement the rendering on the website
- Discuss guidelines with the community
- Develop training for best practices

Also happening right now:

- Defining standard configuration schema
- Defining review process
- Developing auxiliary tooling
- Incorporating system tests

Outlook

Next steps:

- Implement a first metadata schema
- Implement the rendering on the website
- Discuss guidelines with the community
- Develop training for best practices

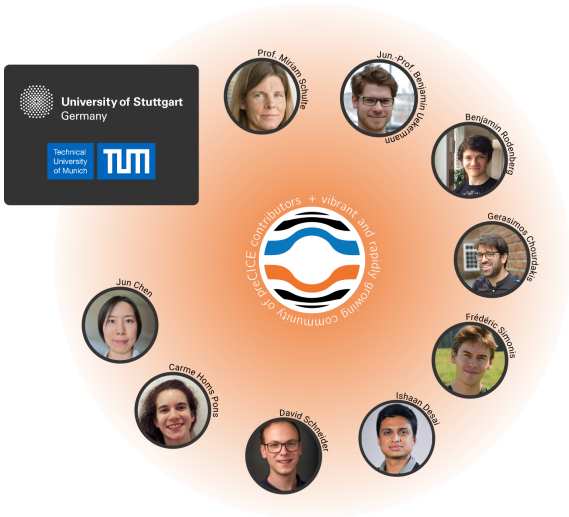
Also happening right now:

- Defining standard configuration schema
- Defining review process
- Developing auxiliary tooling
- Incorporating system tests

See also in other sessions:

- *"Data Model Creation with MetaConfigurator"* **Felix Neubauer**, yesterday at 14:50 (metadata)
- *"Organizing successful software community workshops"* **Jean-Noël Grad** and me, today at 16:40 (community)

The team and the funding



preECO – A community-driven ecosystem of adapters and application cases for the coupling library preCICE

DFG

DFG call on research software usability

Funding number: 528693298

Years running: 2023–2026

The preCICE team has received funding from DFG, SimTech, the BMUV, KONWIHR, SPPEXA, and the EU: precice.org/about.html.

Missing from the picture (new): Felix Neubauer

University of Stuttgart
Institute for Parallel and Distributed Systems

Gerasimos Chourdakis, M.Sc.

e-mail gerasimos.chourdakis@ipvs.uni-stuttgart.de
GitHub MakisH
Find me in teachingRSE working group
de-RSE Munich chapter
OpenFOAM RSE SIG
preCICE community

Current questions

- Should the combined file be called `codemeta.json`? – yes
- Can we have multiple metadata files per directory?
- Codemeta: what is the intention of `hasSourceCode` and `isSourceCodeOf`?
- How do we deal with stand-alone projects that work with preCICE?
- How do we deal with private repositories?