

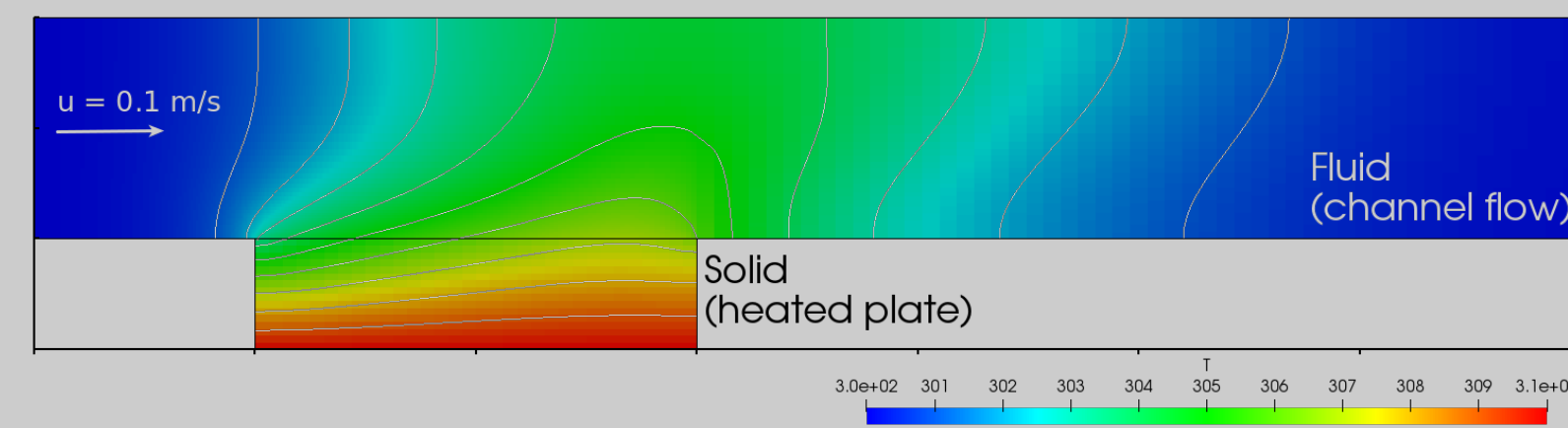
# System regression tests for the preCICE partitioned simulation ecosystem

## Plain language summary

Single computer models are often not enough to describe phenomena and processes across science and engineering. The software library preCICE<sup>[1,2]</sup> allows programmers and end users to combine existing models with minimal effort, producing accurate results fast. Integrations with end-user simulation codes are developed in multiple GitHub repositories, and useful simulations can only be executed by combining multiple codes. To test that development changes do not affect the numerical results, we create reproducible environments from a common description and for the versions of interest, connect them, and check the results at the interface between the domains, solving several challenges on the way.

## A partitioned simulation example

Flow over a heated plate: A preCICE tutorial.<sup>[2]</sup> Choose your "hot solid" code among FEniCS, Nutils, CalculiX, OpenFOAM, and more. Simulate the flow with OpenFOAM or your own code.



## Coupling via preCICE

Add calls to the preCICE library into each code, or use provided adapters. Start each code in a separate process (e.g., two terminals).

OpenFOAM

↔

FEniCS

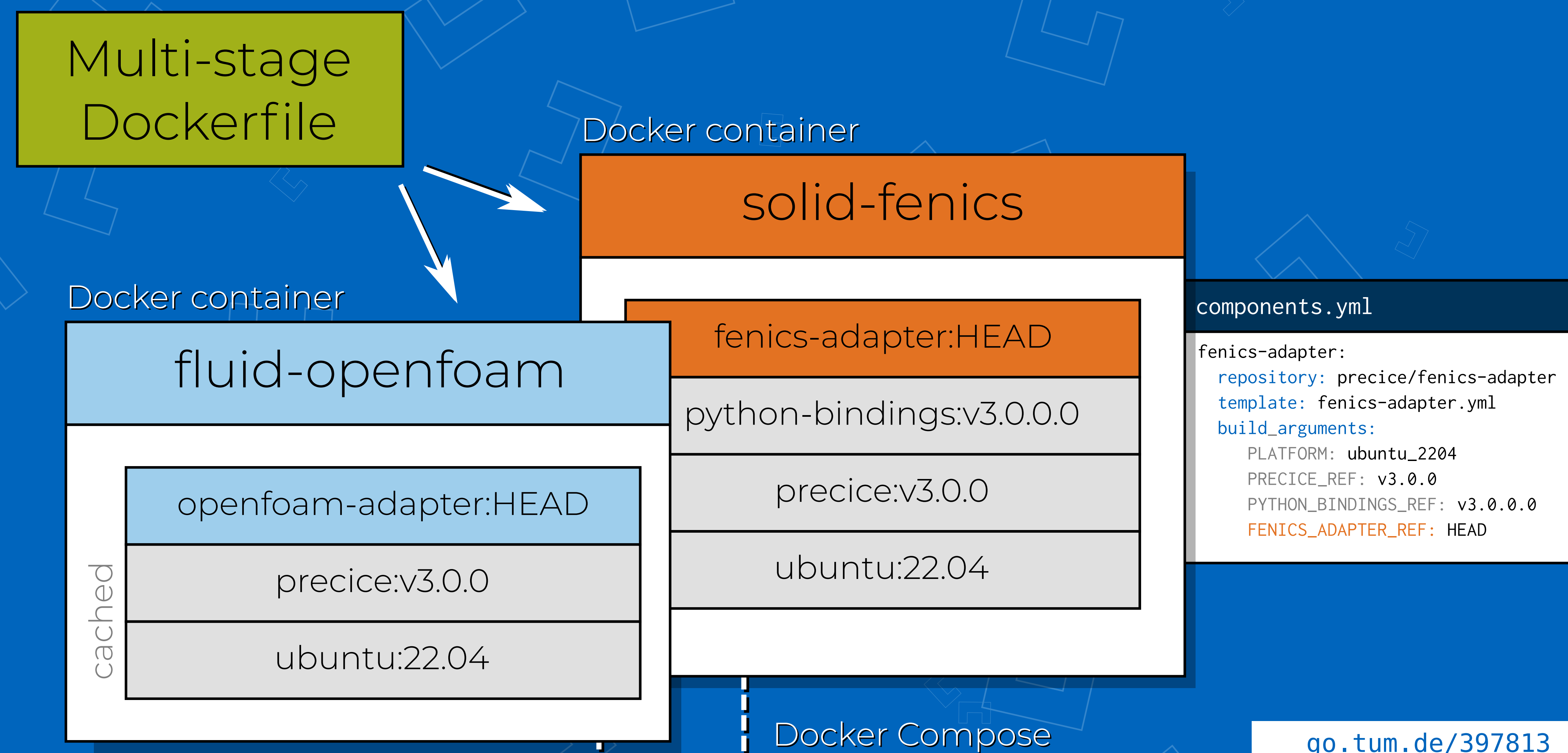
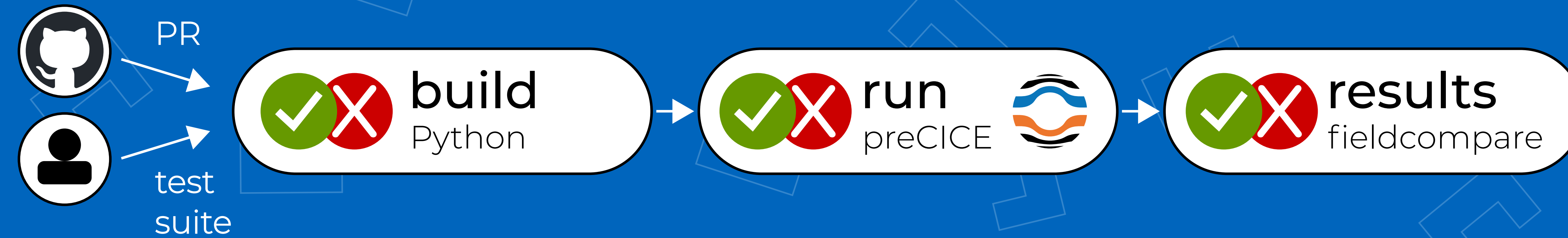
```

import preCICE
v = preCICE.set_mesh_vertices(positions)
preCICE.initialize()
while preCICE.is_coupling_ongoing():
    solve_as_usual()
    preCICE.write_data("temperature", v)
    preCICE.advance()
    preCICE.read_data("heat-flux", v)
    
```

## An ecosystem of components

citable preCICE Distribution (on DaRUS <sup>[3]</sup> )			
Website + Documentation	Nix pkgs	VM	
Tutorials			
OpenFOAM	CalculiX	FEniCS	deal.II
SU2	DUNE	DuMuX	MBDyn
Python	Julia	Matlab	Rust
preCICE (C++ core, C & Fortran bindings)			

# Testing across repositories



Gerasimos Chourdakis, Valentin Seitz, Benjamin Uekermann  
 gerasimos.chourdakis@tum.de  
 SCCS, TUM School of Computation, Information and Technology



- H.-J. Bungartz, F. Lindner, B. Gatzhammer, M. Mehl, K. Scheufele, A. Shukaev, B. Uekermann (2016). preCICE – A fully parallel library for multi-physics surface coupling. In *Comput. & Fluids*, 141, 250–258. <https://doi.org/10.1016/j.compfluid.2016.04.003>
- G. Chourdakis, K. Davis, B. Rodenberg, M. Schulte, F. Simonis, B. Uekermann, et al., preCICE v2: A sustainable and user-friendly coupling library [version 2; peer review: 2 approved] *Open Res Europe* 2022, 2:51, <https://doi.org/10.12688/openreseurope.14445.2>
- G. Chourdakis, K. Davis, I. Desai, B. Rodenberg, D. Schneider, F. Simonis, B. Uekermann, B. Ariguib, P. Cardiff, A. Jaust, P. Kharitenko, R. Klöforn, N. Kotarsky, B. Martin, E. Scheurer, V. Schüller, G. van Zwieten, K. Yurt (2023) preCICE Distribution Version v2211.0 <https://doi.org/10.18419/darus-3576>, DaRUS, V1.



## Reference results and comparisons

Stored next to the cases VTK compared with

git-lfs.com

gitlab.com/dglaeser/fieldcompare

```

flow-over-heated-plate/
fluid-openfoam/
solid-fenics/
solid-nutils/
reference-results/
fluid-openfoam_solid-fenics.tar.gz
fluid-openfoam_solid-nutils.tar.gz
# One directory for each example
# One directory for each code
# Multiple code options
# Reference results versioned
# One file for each combination
    
```

fluid-openfoam	preCICE	solid-fenics
OpenFOAM output .foam, logs	mesh exports (.vtk) iterations.log	FEniCS output .vtk, logs
specific to OpenFOAM, controlled by others	standard formats (VTK, CSV)	specific to FEniCS, controlled by others

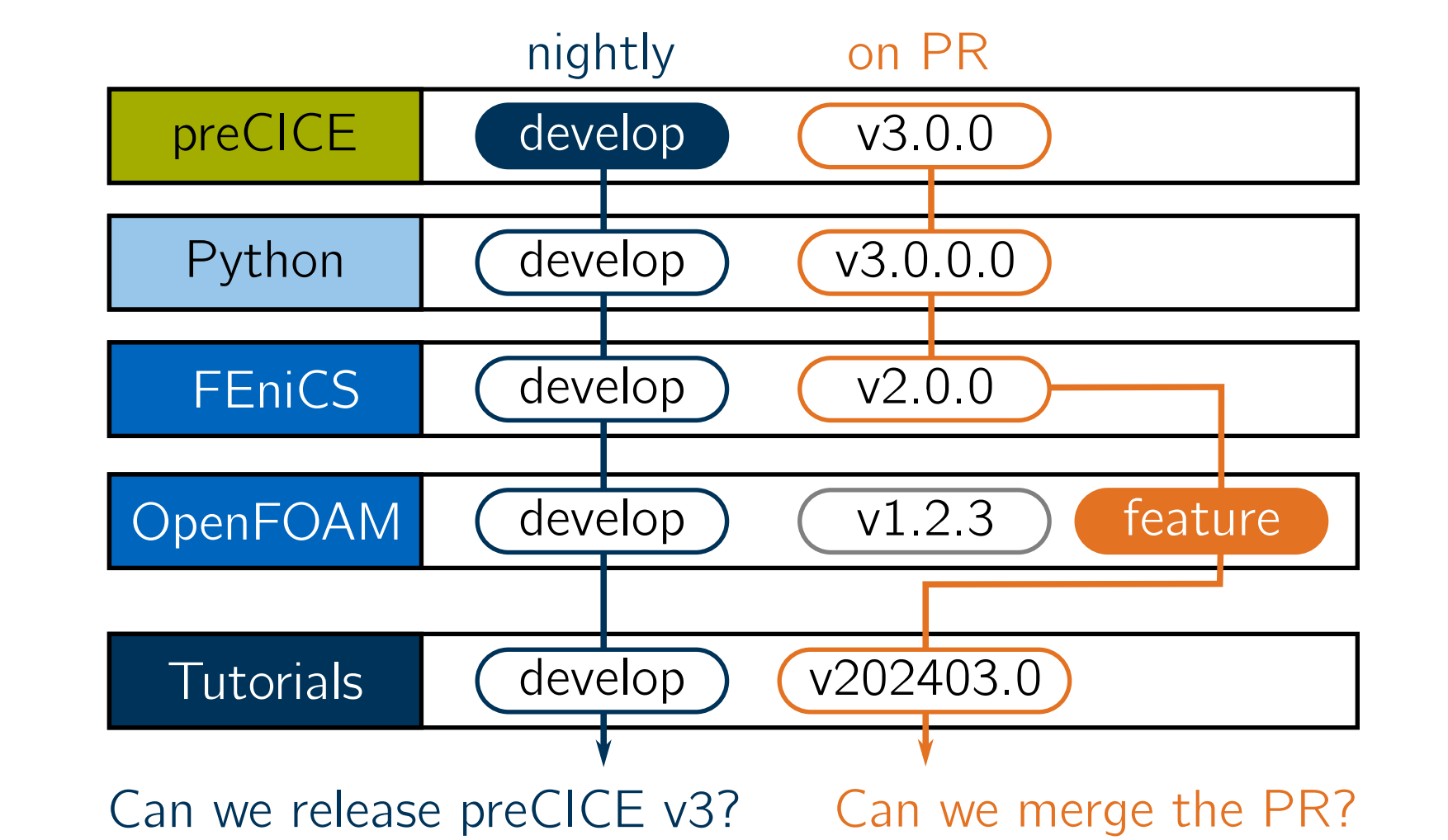
## Metadata, tests, and test suites

```

metadata.yml (preCICE/tutorials/flow-over-heated-plate/)
name: Flow over heated plate
path: flow-over-heated-plate
participants:
  - Fluid
  - Solid
cases:
  fluid-openfoam:
    participant: Fluid
    directory: ./fluid-openfoam
    run: ./run.sh
    component: openfoam-adapter
  solid-fenics:
    participant: Solid
    directory: ./solid-fenics
    run: ./run.sh
    component: fenics-adapter
tests.yml (preCICE/tutorials)
test_suites:
  openfoam_adapter_pr:
    tutorials:
      - path: flow-over-heated-plate
        case_combination:
          - fluid-openfoam
          - solid-fenics
        reference_result:
          fluid-of-solid-fenics.tar.gz
    
```

## Multiple repos, multiple perspectives

Partitioned approach makes CI complex. Tests can run locally or on GitHub Actions, triggered manually or by pull requests.



```

openfoam-adapter-workflow.yml (preCICE/openfoam-adapter)
jobs:
  testing-approved-pr:
    if: github.event.review.state == 'APPROVED'
    uses: preCICE/tutorials/.github/workflows/workflow.yml@develop
    with:
      suites: openfoam_adapter_pr
      build_args: OPENFOAM_ADAPTER_REF:${{github.event.pr.head.sha}}
      systests_branch: develop
    
```

```

workflow.yml (preCICE/tutorials)
jobs:
  run_testsuite:
    runs-on: [self-hosted, linux, x64, preCICE-tests-vm]
    steps:
      - name: Run tests
        run: |
          cd tools/tests
          python systemtests.py
          --build_args=${{inputs.build_args}}
          --suites=${{inputs.suites}}
    
```