Chair of Scientific Computing
Department of Informatics
Technical University of Munich

preCICE: A dependable open-source coupling library for partitioned multi-physics simulations

Hans-Joachim Bungartz[1], **Gerasimos Chourdakis**[1], **Florian Lindner**[2], **Miriam Mehl**[2], **Benjamin Rüth**[1], **Dmytro Sashko**[1], **Frédéric Simonis**[1], **Benjamin Uekermann**[3]
[1]{bungartz, simonis, chourdak, sashko}@in.tum.de, Chair of Scientific Computing, Technical University of Munich
[2]{Miriam.Mehl, Florian.Lindner}@ipvs.uni-stuttgart.de, Institute for Parallel and Distributed Systemes, University of Stuttgart
[3]b.w.uekermann@tue.nl, Department of Mechanical Engineering, Eindhoven University of Technology

preCICE Coupling Library

- Coupling Schemes: parallel/serial, explicit/implicit
- Fixed-Point Acceleration: IQN-ILS, IQN-IMVJ
- Parallel P2P Communication: MPI, TCP
- Ready-To-Use Adapters: SU2, OpenFOAM, FEniCS
- Data Mapping Schemes: Nearest Projection, RBF
- Configuration-Driven
- Minimally Invasive Adaptation
- API: C++, C, Python, Fortran
- Black-Box Coupling
- Detailed Logging and Profiling
- Cross-Platform: Linux, macOS, Windows
- Parallel Initialization
- Adaptive Dynamic Meshes
- Multirate Time Integration
- Geometric Multiscale
- Mesh Subdivision in Patches

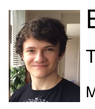
preCICE (Precise Code Interaction Coupling Environment) is a coupling library for partitioned multi-physics simulations used by over 30 research groups in academia, non-university research facilities and in the industry. Its minimally invasive API and scalability on massively parallel systems allows for rapid adaptation, and thus offers the flexibility needed to keep a decent time-to-solution for complex multi-physics scenarios. As a common interface, it encourages collaboration between researchers and ensures compatibility and thus the sustainability of both modern and legacy code.

Core Developers

Gerasimos Chourdakis
Technical University of Munich
Geometric Multiscale, OpenFOAM, DevOps

Kyle Davis
University of Stuttgart
Subdivided Meshes, CalculiX

Florian Lindner
University of Stuttgart
RBF, MPI, Code Modernization, Profiling

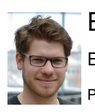


Benjamin Rueth
Technical University of Munich
Multirate Time Integration, Python, FEniCS

Dmytro Sashko
Technical University of Munich
CI, System Tests, run.precice.org

Frédéric Simonis
Technical University of Munich
Dynamic Adaptive Meshes, Build System, DevOps

Amin Totounferoush
University of Stuttgart
Parallel Initialization

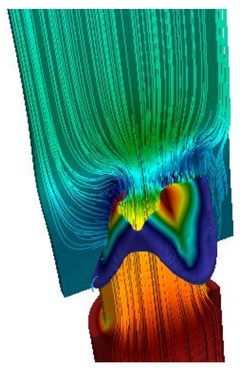


Benjamin Uekermann
Eindhoven University of Technology
Parallelization, Scientific Lead

Previous main contributors:
Alexander Rusch (ETH Zürich), Dr. Bernhard Gatzhammer (Alumni of TUM),
Klaudius Scheufele (University of Stuttgart), and many more researchers and students!

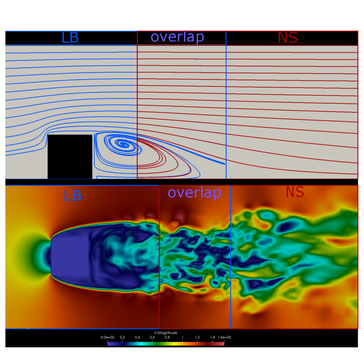
A distributed, multi-cultural, and interdisciplinary team.

Evaluation of Heart Valve Biomechanics



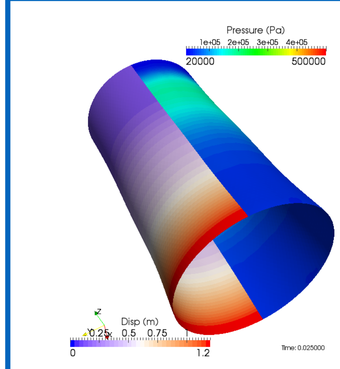
[2] University of the Free State, ZA

Hybrid Methods for Wind modelling in urban areas



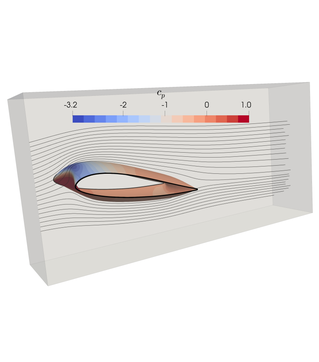
[3] The University of Manchester, GB

Simulation of High Impact Loads on Structures



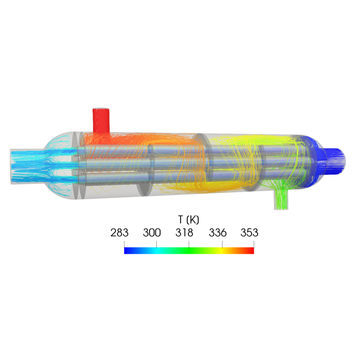
[4] A*STAR, SG

Fluid-Structure Interaction of Inflatable Wing Sections



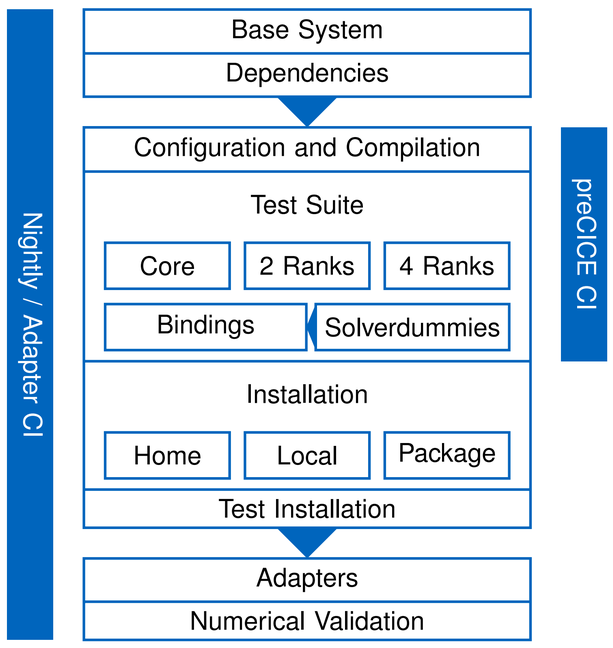
[5] TU Delft, NL

Shell and Tube Heat Exchanger



[6,7,8] TU Munich/SimScale, DE

Testing, Validation, and CI



20+ In-House Codes **30+ Contributors** **45k LOC**
Developed Since 2008 **30+ User Groups** **4.7k Commits**
Adapters for OpenFOAM, CalculiX, SU2, FEniCS, deal.II, ...

Reliabilty and Usability

Stable Foundation
Boost, nlohmann/json, libxml2, MPI, PETSc

Integration and Installation
CMake, pkg-config, pip, Debian, Spack

Code Quality
- Code coverage: lcov, codecov.io
- Static analysis and modernisation: clang-tidy
- External tools: lgtm, codefactor, codacy

Documentation
Extensive GitHub Wiki, Doxygen

Adaptation

```
1   precice::SolverInterface precice("FluidSolver",rank,size);
2   precice.configure("precice-config.xml");
3   precice.setMeshVertices();
4   precice.initialize();
5
6   while (precice.isCouplingOngoing()) { // main time loop
7       solve();
8
9       precice.writeBlockVectorData();
10      precice.advance();
11      precice.readBlockVectorData();
12
13      endTimeStep(); // e.g. write results, increase time
14  }
15
16  precice.finalize();
```

Timesteps, most arguments and less important methods omitted. Full example in the wiki.

Community and Outreach

Communication
- Face-to-face
- Chat on Gitter
- In-depth discussions on mailing lists
- GitHub Issues/Pull Requests

Culture
- Regular short telcos
- Regular structured meetings
- Local casual coding evenings
- Fixed release schedule
- Peer-reviews
- Open discussions

Community
- ECCOMAS mini symposia 2018, 2019
- preCICE Workshop 17-18 Feb 2020, TU Munich
- Cross-advertising via testimonials
- Stand-alone profiling library *precice/EventTimings*
- Valuable asset to teaching
- Co-supervision of student projects
- Contributions to adapters
- Common interface encourages collaboration
- Thriving for xSDK compatibility

References:
[1] preCICE – A Fully Parallel Library for Multi-Physics Surface Coupling, Hans-Joachim Bungartz, Bernhard Gatzhammer, Florian Lindner, Miriam Mehl, Klaudius Scheufele, Alexander Shukaev, Benjamin Uekermann, In Computers and Fluids, Volume 141, p. 250—258. Elsevier, 2016.
[2] Numerical and experimental investigation of the hemodynamics of an artificial heart valve, Kyle Davis, Thesis (MEng)–Stellenbosch University, 2018.
[3] Dual Navier-Stokes / Lattice-Boltzmann method for urban wind flow, Marta Camps Santasmasas, Alistair Revell, Ben Parslew, Adrian Harwood, William Crowther, 2018.
[4] A fluid structure interactions partitioned approach for simulations of explosive impacts on deformable structures, Vinh-Tan Nguyen, Bernhard Gatzhammer, International Journal of Impact Engineering, Volume 80, 2015, Pages 65-75.
[5] Fluid-Structure Interaction Simulations on Kites, Mikko Folkersma, Roland Schmehl, Axello Viré, In Book of Abstracts of the International Airborne Wind Energy Conference 2017 (p. 144).
[6] Conjugate Heat Transfer with the Multiphysics Coupling Library preCICE, Lucia Cheung Yau, Master's thesis, Institut für Informatik, Technische Universität München, 2016.
[7] A general OpenFOAM adapter for the coupling library preCICE, Gerasimos Chourdakis, Master's thesis, Institut für Informatik, Technische Universität München, 2017.
[8] Comparing OpenFOAM's Intrinsic Conjugate Heat Transfer Solver with preCICE-Coupled Simulations, Alexander Rusch, Benjamin Uekermann, Unpublished white paper, 2018.

This project has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 754462

Supported by: Federal Ministry for Economic Affairs and Energy, on the basis of a decision by the German Bundestag

www.precice.org github.com/precice