RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

cea cnrs Inria

FRANCE 2030
PROGRAMME DE RECHERCHE
NUMÉRIQUE POUR L'EXASCALE

Efficient and composable programming models

Work Package 1

M. Perache (CEA), C. Perez (Inria)

WP1 Goals

- Abstract the application (code and data) from the machines
  - Ease the porting to new machines
  - Ease the exploitation of machine heterogeneity
  - Ease debugging of complex software stack
- Improve code composability
  - Simplify code reuse to reduce the development
  - Improve the separation of concerns between domain experts and HPC experts

WP1 Members

- WP1 leaders
  - Marc Perache (CEA) & Christian Perez (Inria)
- Other permanent researchers:
  - Benoit Combebale (U. RennesI)
  - Dorian Leroy (CEA)
  - Jerry Lacmou Zeutouo (UPJV) – external collaborator
- PhD Students:
  - Ugo Battiston (WP1+WP2, since October 2023)
  - Ewen Brune (since October 2023, CEA funding)
  - *Ke-Wei Cheng (WP1, December 20250 – hiring in progress*
  - *Julien Gaupp (WP1+WP4, December 2025) – hiring in progress*
- Internship:
  - Alix Peigue (WP!, September 2025 – March 2026)

PhDs: Current and Future

- Ugo Battiston (WP1+WP2, since October 2023)
  - *Connecting Kokkos with the polyhedral model*
  - Advisors: Philippe Clauss (Inria, WP2) & Marc Pérache
- Ewen Brune (since October 2023, CEA funding)
  - *Testing Framework For Scientific Computing*
  - Advisors: Benoit Combemale (Univ Rennes), Arnaud Blouin (Univ Rennes - IRISA – INRIA) & Dorian Leroy (CEA)
- *Ke-Wei Cheng (WP1, December 2025, Inria)*
  - *Foundation of an HPC Composition Model*
  - *Advisors: Benoit Combemale (Univ Rennes – IRISA – INRIA), Christian Perez (Inria)*
- *Julien Gaupp (WP1+WP4 December 2025, Inria)*
  - *Composability from numerical algorithms to programming model*
  - *Adviors: Emmanuel Agullor (Inria, WP4), Christian Perez (Inria, WP1)*

C++ complexity disambiguation for advanced optimizing and parallelizing code transformations

C++ complexity disambiguation for advanced optimizing and parallelizing code transformations

Connecting Kokkos with the polyhedral model

- See WP2 presentation

Limitation

- Focus on Kokkos only
- No real applications

➔ Next step: CEA/IFPEN Post-Doc on Arcane framework, Python, Kokkos and real application

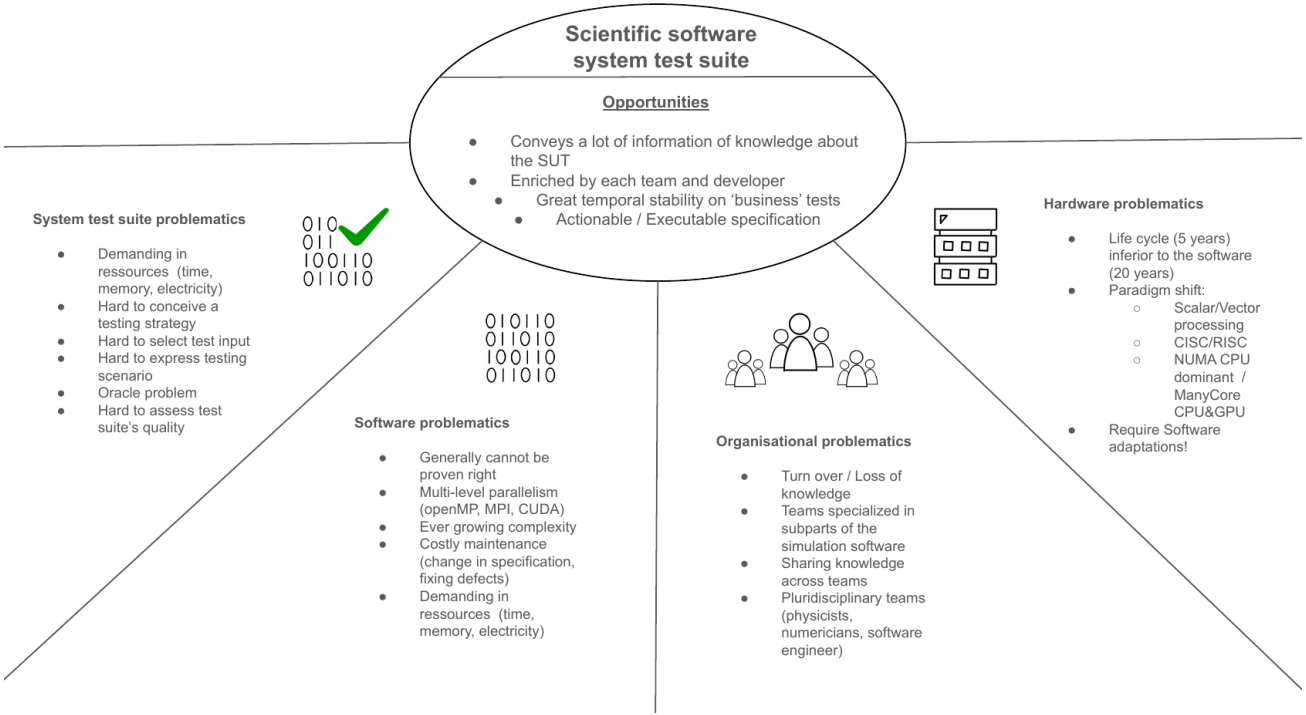
Tools for parallel heterogeneous scientific application at scale

Testing Framework for scientific computing

Challenges

- Maintainability of large scientific software systems
- Trustworthiness of large scientific software systems
- Capitalization of effort and knowledge transferred in large scientific software
- systems

Testing Framework for scientific computing



Testing Framework for scientific computing

Results

- One paper published
- An empirical Study on test in scientific computing codes
- Prototype of a testing framework

Next steps

- Applying the tool to open source Arcane based simulation system : MaHyCo, MicroHydro, QuickSilver.

Foundation of an HPC Composition Model

Improving parallel code composability

Towards a general parallel code composition

Some models exists for some specific composition patterns

- Parallel RMI between SPMD codes (GridCCM, some CCA variants), data sharing, algorithmic skeletons, etc.
- Dataflow/Workflow/stream models / Partitioned dataflow model (composition of task graphs: COMET), etc.

HLCM: a first generic parallel composition model

- Solve some features (hierarchy and efficient communication, genericity for component and connectors)
- However the transformation into a resource aware assembly was intractable in practice

Goal of this task: propose a more tractable model, with an efficient support of current machines

Relationships with

- WP4 (composable numerical algorithms)
- HE ODISSEE (composability studies for SKA and LHC use cases)

High level data description and partitioning for reusable parallel building blocks model

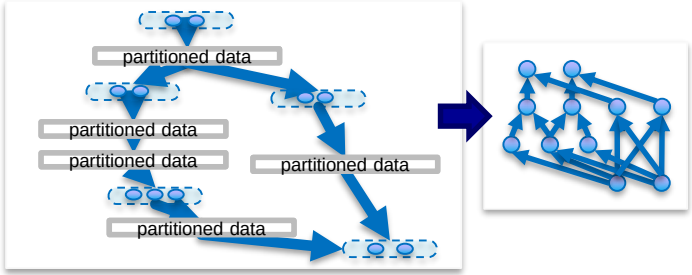
Improving partitioned dataflow model

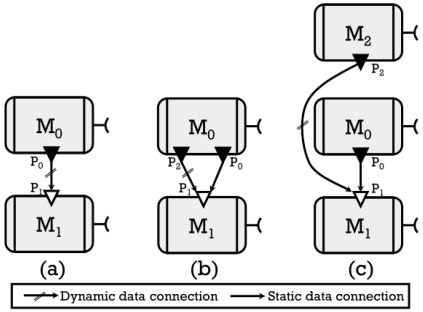
Work with Jerry Lacmou Zeutouo (was post-doc ExaSoft)

Motivation: the AEVOL application

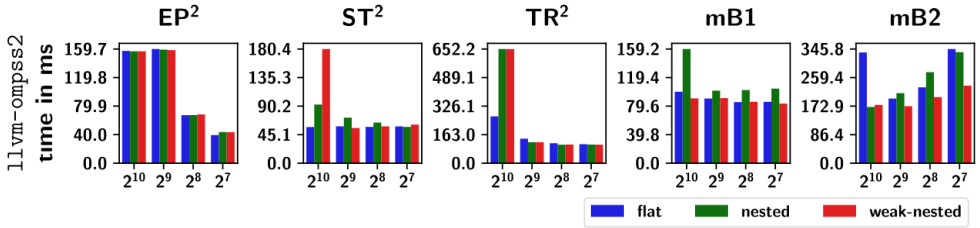
Result: Improved COMET model

- Dynamic workflows into COMET
  - 3 types of task dependencies: flat, nested, weak-nested
- Adding StarPU support to COMET (with WP3)
  - CPU support done, GPU support in progress
- Journal article almost completed





EP: Embarrassingly parallel **ST**: Stencil **TR**: Transpose
X²: dataflow of X followed by **X**
mb1: dataflow of **EP** and **ST** **mb2**: dataflow of **EP**, **ST**, and **TR**

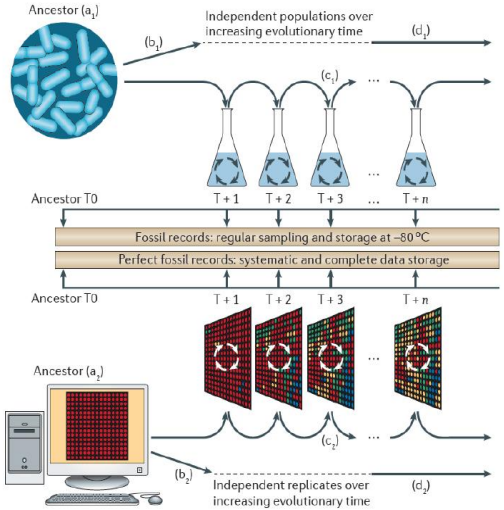


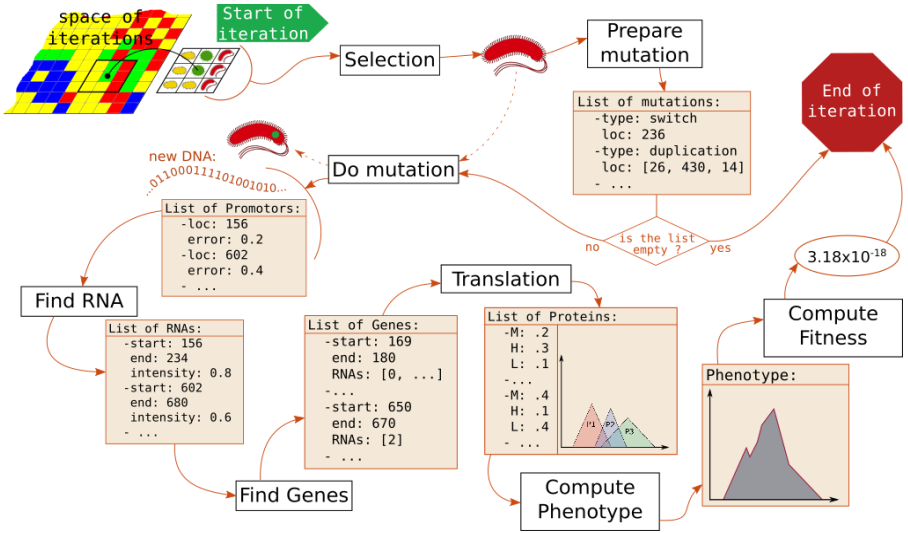
Software
- COMET updated

Next step
- Generalize to hierarchical data and integrate that into the model of the previous task
- PhD to be hired next year in collaboration with WP3 (runtimes)

Aevol: A simulator of Darwinian evolution

Developed by the BioTic team @ Lyon
URL: https://www.aevol.fr/





Conclusion

Conclusion

Results

- First Kokkos optimization at compiler level
- Test framework prototype
- Improvement of COMET for Aevol applications

Next steps

- Finalize the hirings
- Work on composition models and partitionned hierarchical data
- Target real applications

RÉPUBLIQUE FRANÇAISE
Liberté
Égalité
Fraternité

cea cnrs Inria

FRANCE 2030

PROGRAMME DE RECHERCHE

NUMÉRIQUE POUR L'EXASCALE

Retrouvez toutes nos actualités

NumPEx

Generating parallel variants

Code generation in COMET

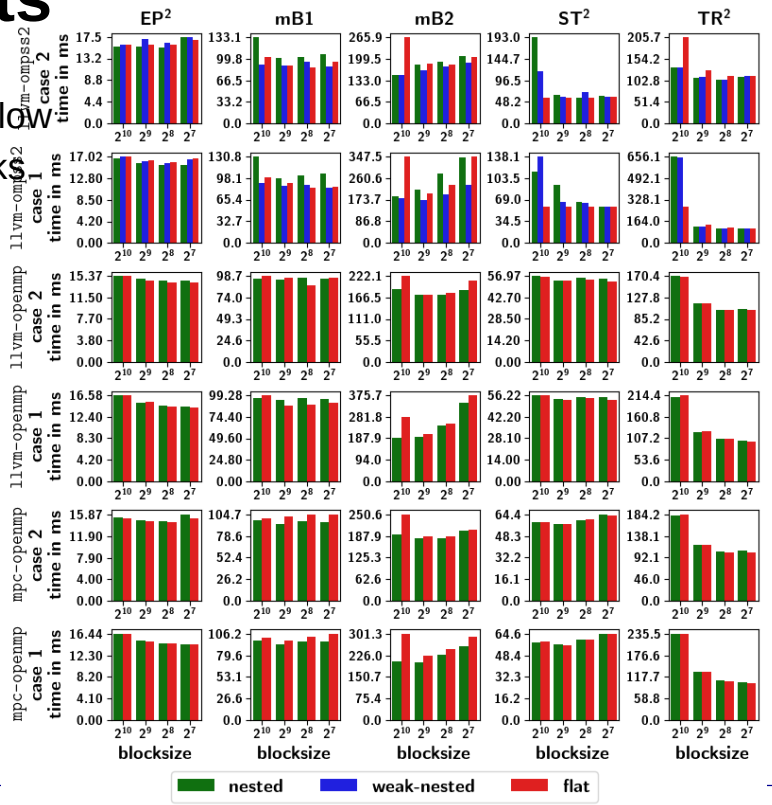
- From a data-parallel dataflow
- To a code that submit tasks

Task dependencies variants

- OpenMP “Flat” tasks
- OpenMP “Nested” tasks
- “Weak-nested” tasks
  - Not in OpenMP standard
- (in progress) StarPU

Experiments

- 3 strategies
- 3 OpenMP runtime
  - llvm
  - mpc (CEA)
  - llvm-opms2 (BSC)



High Level Component Model

Hierarchical model

Generic model

- Support meta-programming (template *à la* C++)

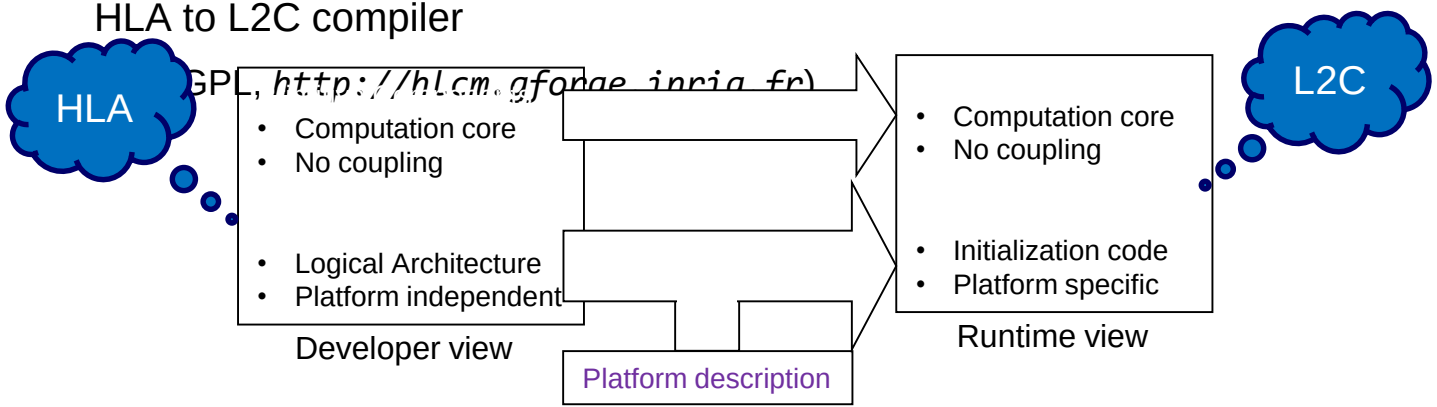
Connector based

- Primitive and composite

Static

HLA to L2C compiler

(GPL, *http://hlcm.gforge.inria.fr*)



Julien Bigot, and Christian Pérez.
High Performance Computing: From Grids and Clouds to Exascale, chap. On High Performance Composition Operators in Component Models.
Advances in Parallel Computing, vol 20, pp 182-201, IOS Press, 2011.