RESEARCH CENTRE

Inria Centre at the University of Bordeaux

IN PARTNERSHIP WITH:

Institut Polytechnique de Bordeaux, Université de Bordeaux

2024

ACTIVITY REPORT

Project-Team

TADAAM

Topology-aware system-scale data management for high-performance computing

IN COLLABORATION WITH: Laboratoire Bordelais de Recherche en Informatique (LaBRI)

DOMAIN

Networks, Systems and Services, Distributed Computing

THEME

Distributed and High Performance Computing

Inria

Contents

Project-Team TADAAM

Creation of the Project-Team: 2017 December 01

Keywords

Computer sciences and digital sciences

A1.1.1. – Multicore, Manycore

A1.1.2. – Hardware accelerators (GPGPU, FPGA, etc.)

A1.1.3. – Memory models

A1.1.4. – High performance computing

A1.1.5. – Exascale

A1.1.9. – Fault tolerant systems

A1.2.4. – QoS, performance evaluation

A2.1.7. – Distributed programming

A2.2.2. – Memory models

A2.2.3. – Memory management

A2.2.4. – Parallel architectures

A2.2.5. – Run-time systems

A2.6.1. – Operating systems

A2.6.2. – Middleware

A2.6.4. – Ressource management

A3.1.2. – Data management, quering and storage

A3.1.3. – Distributed data

A3.1.8. – Big data (production, storage, transfer)

A6.1.2. – Stochastic Modeling

A6.2.3. – Probabilistic methods

A6.2.6. – Optimization

A6.2.7. – High performance computing

A6.3.3. – Data processing

A7.1.1. – Distributed algorithms

A7.1.2. – Parallel algorithms

A7.1.3. – Graph algorithms

A8.1. – Discrete mathematics, combinatorics

A8.2. – Optimization

A8.7. – Graph theory

A8.9. – Performance evaluation

Other research topics and application domains

B6.3.2. – Network protocols

B6.3.3. – Network Management

B9.5.1. – Computer science

B9.8. – Reproducibility

1 Team members, visitors, external collaborators

Research Scientists

- Emmanuel Jeannot [Team leader, INRIA, Senior Researcher, until Jun 2024]
- Brice Goglin [Team leader, INRIA, Senior Researcher, from Jul 2024]
- Alexandre Denis [INRIA, Researcher]
- Brice Goglin [INRIA, Senior Researcher, until Jun 2024]
- Luan Teylo Gouveia Lima [INRIA, ISFP]
- Emmanuel Jeannot [INRIA, Senior Researcher, from Jul 2024 until Aug 2024]
- Mihail Popov [INRIA, ISFP, from Oct 2024]

Faculty Members

- Guillaume Mercier [BORDEAUX INP, Associate Professor Delegation]
- François Pellegrini [UNIV BORDEAUX, Professor]
- Francieli Zanon-Boito [UNIV BORDEAUX, Associate Professor Delegation, from Sep 2024]
- Francieli Zanon-Boito [UNIV BORDEAUX, Associate Professor, until Aug 2024, in maternity leave from December 2023 to April 2024]

PhD Students

- Alexis Bandet [INRIA]
- Clément Gavoille [INRIA, until Mar 2024]
- Charles Goedefroit [BULL, CIFRE, from Mar 2024]
- Thibaut Pepin [CEA]
- Richard Sartori [INRIA, from Apr 2024 until Nov 2024]
- Richard Sartori [BULL, until Apr 2024]
- Meline Trochon [INRIA, from Nov 2024]

Technical Staff

- Mahamat Younous Abdraman [INRIA, Engineer, from Oct 2024]
- Clément Barthelemy [INRIA, Engineer, until Aug 2024]
- Quentin Buot [INRIA, Engineer, until Sep 2024]
- Pierre Clouzet [INRIA, Engineer]
- Axel Malmgren [INRIA, from Nov 2024]

Interns and Apprentices

- Mahamat Younous Abdraman [INRIA, Intern, from Apr 2024 until Aug 2024]
- Aymen Ali Yahia [INRIA, Intern, from Jun 2024 until Aug 2024]
- Jean-Alexandre Collin [INRIA, Intern, from Feb 2024 until Jul 2024]
- Alexandre Laffargue [INRIA, Intern, from Jul 2024 until Aug 2024]
- Mathis Lamiroy [ENS DE LYON, from Mar 2024 until Jul 2024]
- Axel Malmgren [INRIA, Intern, from Jul 2024 until Aug 2024]

Administrative Assistant

- Fabienne Cuyollaa [INRIA]

External Collaborators

- Douglas Brum [Federal Fluminense University, Brazil, co-advised by Luan Teylo]
- Charles Goedefroit [ATOS, until Feb 2024]
- Julien Rodriguez [DGA, from Sep 2024]
- Julien Rodriguez [UNIV MONTPELLIER, until Aug 2024]

2 Overall objectives

In TADAAM, we propose a new approach where we allow the application to explicitly express its resource needs about its execution. The application needs to express its behavior, but in a different way from the compute-centric approach, as the additional information is not necessarily focused on computation and on instructions execution, but follows a high-level semantics (needs of large memory for some processes, start of a communication phase, need to refine the granularity, beginning of a storage access phase, description of data affinity, etc.). These needs will be expressed to a service layer though an API. The service layer will be system-wide (able to gather a global knowledge) and stateful (able to take decision based on the current request but also on previous ones). The API shall enable the application to access this service layer through a well-defined set of functions, based on carefully designed abstractions.

Hence, **the goal of TADAAM is to design a stateful system-wide service layer for HPC systems, in order to optimize applications execution according to their needs**.

This layer will abstract low-level details of the architecture and the software stack, and will allow applications to register their needs. Then, according to these requests and to the environment characteristics, this layer will feature an engine to optimize the execution of the applications at system-scale, taking into account the gathered global knowledge and previous requests.

This approach exhibits several key characteristics:

- It is independent from the application parallelization, the programming model, the numerical scheme and, largely, from the data layout. Indeed, high-level semantic requests can easily be added to the application code after the problem has been modeled, parallelized, and most of the time after the data layout has been designed and optimized. Therefore, this approach is – to a large extent – orthogonal to other optimization mechanisms and does not require application developers to rewrite their code.
- Application developers are the persons who know best their code and therefore the needs of their application. They can easily (if the interface is well designed and the abstractions are correctly exposed), express the application needs in terms of resource usage and interaction with the whole environment.

- Being stateful and shared by all the applications in the parallel environment, the proposed layer will therefore enable optimizations that:
  - cannot be performed statically but require information only known at launch- or run-time,
  - are incremental and require minimal changes to the application execution scheme,
  - deal with several parts of the environment at the same time (e.g., batch scheduler, I/O, process manager and storage),
  - take into account the needs of several applications at the same time and deal with their interaction. This will be useful, for instance, to handle network contention, storage access or any other shared resources.

3 Research program

3.1 Need for System-Scale Optimization

Firstly, in order for applications to make the best possible use of the available resources, it is impossible to expose all the low-level details of the hardware to the program, as it would make impossible to achieve portability. Hence, the standard approach is to add intermediate layers (programming models, libraries, compilers, runtime systems, etc.) to the software stack so as to bridge the gap between the application and the hardware. With this approach, optimizing the application requires to express its parallelism (within the imposed programming model), organize the code, schedule and load-balance the computations, etc. In other words, in this approach, the way the code is written and the way it is executed and interpreted by the lower layers drives the optimization. In any case, this approach is centered on how computations are performed. Such an approach is therefore no longer sufficient, as the way an application is executing does depend less and less on the organization of computation and more and more on the way its data is managed.

Secondly, modern large-scale parallel platforms comprise tens to hundreds of thousand nodes [1]. However, very few applications use the whole machine. In general, an application runs only on a subset of the nodes [2]. Therefore, most of the time, an application shares the network, the storage and other resources with other applications running concurrently during its execution. Depending on the allocated resources, it is not uncommon that the execution of one application interferes with the execution of a neighboring one.

Lastly, even if an application is running alone, each element of the software stack often performs its own optimization independently. For instance, when considering an hybrid MPI/OpenMP application, one may realize that threads are concurrently used within the OpenMP runtime system, within the MPI library for communication progression, and possibly within the computation library (BLAS) and even within the application itself (pthreads). However, none of these different classes of threads are aware of the existence of the others. Consequently, the way they are executed, scheduled, prioritized does not depend on their relative roles, their locations in the software stack nor on the state of the application.

The above remarks show that in order to go beyond the state-of-the-art, it is necessary to design a new set of mechanisms allowing cross-layer and system-wide optimizations so as to optimize the way data is allocated, accessed and transferred by the application.

3.2 Scientific Challenges and Research Issues

In TADAAM, we will tackle the problem of efficiently executing an application, at system-scale, on an HPC machine. We assume that the application is already optimized (efficient data layout, use of effective libraries, usage of state-of-the-art compilation techniques, etc.). Nevertheless, even a statically optimized application will not be able to be executed at scale without considering the following dynamic constraints: machine topology, allocated resources, data movement and contention, other running applications,

[1] More than 22,500 XE6 compute node for the BlueWaters system; 5040 B510 Bullx Nodes for the Curie machine; more than 49,000 BGQ nodes for the MIRA machine.

[2] In 2014, the median case was 2048 nodes for the BlueWaters system and, for the first year of the Curie machine, the median case was 256 nodes

access to storage, etc. Thanks to the proposed layer, we will provide a simple and efficient way for already existing applications, as well as new ones, to express their needs in terms of resource usage, locality and topology, using a high-level semantic.

It is important to note that we target the optimization of each application independently but also several applications at the same time and at system-scale, taking into account their resource requirement, their network usage or their storage access. Furthermore, dealing with code-coupling application is an intermediate use-case that will also be considered.

Several issues have to be considered. The first one consists in providing relevant **abstractions and models to describe the topology** of the available resources **and the application behavior**.

Therefore, the first question we want to answer is: **"How to build scalable models and efficient abstractions enabling to understand the impact of data movement, topology and locality on performance?"** These models must be sufficiently precise to grasp the reality, tractable enough to enable efficient solutions and algorithms, and simple enough to remain usable by non-hardware experts. We will work on (1) better describing the memory hierarchy, considering new memory technologies; (2) providing an integrated view of the nodes, the network and the storage; (3) exhibiting qualitative knowledge; (4) providing ways to express the multi-scale properties of the machine. Concerning abstractions, we will work on providing general concepts to be integrated at the application or programming model layers. The goal is to offer means, for the application, to express its high-level requirements in terms of data access, locality and communication, by providing abstractions on the notion of hierarchy, mesh, affinity, traffic metrics, etc.

In addition to the abstractions and the aforementioned models we need to **define a clean and expressive API in a scalable way**, in order for applications to express their needs (memory usage, affinity, network, storage access, model refinement, etc.).

Therefore, the second question we need to answer is: "**how to build a system-scale, stateful, shared layer that can gather applications needs expressed with a high-level semantic?**". This work will require not only to define a clean API where applications will express their needs, but also to define how such a layer will be shared across applications and will scale on future systems. The API will provide a simple yet effective way to express different needs such as: memory usage of a given portion of the code; start of a compute intensive part; phase where the network is accessed intensively; topology-aware affinity management; usage of storage (in read and/or write mode); change of the data layout after mesh refinement, etc. From an engineering point of view, the layer will have a hierarchical design matching the hardware hierarchy, so as to achieve scalability.

Once this has been done, the service layer, will have all the information about the environment characteristics and application requirements. We therefore need to design a set of **mechanisms to optimize applications execution**: communication, mapping, thread scheduling, data partitioning / mapping / movement, etc.

Hence, the last scientific question we will address is: "**How to design fast and efficient algorithms, mechanisms and tools to enable execution of applications at system-scale, in full a HPC ecosystem, taking into account topology and locality?**" A first set of research is related to thread and process placement according to the topology and the affinity. Another large field of study is related to data placement, allocation and partitioning: optimizing the way data is accessed and processed especially for mesh-based applications. The issues of transferring data across the network will also be tackled, thanks to the global knowledge we have on the application behavior and the data layout. Concerning the interaction with other applications, several directions will be tackled. Among these directions we will deal with matching process placement with resource allocation given by the batch scheduler or with the storage management: switching from a best-effort application centric strategy to global optimization scheme.

4 Application domains

4.1 Mesh-based applications

TADAAM targets scientific simulation applications on large-scale systems, as these applications present huge challenges in terms of performance, locality, scalability, parallelism and data management. Many of

these HPC applications use meshes as the basic model for their computation. For instance, PDE-based simulations using finite differences, finite volumes, or finite elements methods operate on meshes that describe the geometry and the physical properties of the simulated objects.

Mesh-based applications not only represent the majority of HPC applications running on existing supercomputing systems, yet also feature properties that should be taken into account to achieve scalability and performance on future large-scale systems. These properties are the following:

Size Datasets are large: some meshes comprise hundreds of millions of elements, or even billions.

Dynamicity In many simulations, meshes are refined or coarsened at each time step, so as to account for the evolution of the physical simulation (moving parts, shockwaves, structural changes in the model resulting from collisions between mesh parts, etc.).

Structure Many meshes are unstructured, and require advanced data structures so as to manage irregularity in data storage.

Topology Due to their rooting in the physical world, meshes exhibit interesting topological properties (low dimensionality embedding, small maximum degree, large diameter, etc.). It is very important to take advantage of these properties when laying out mesh data on systems where communication locality matters.

All these features make mesh-based applications a very interesting and challenging use-case for the research we want to carry out in this project. Moreover, we believe that our proposed approach and solutions will contribute to enhance these applications and allow them to achieve the best possible usage of the available resources of future high-end systems.

5 Social and environmental responsibility

5.1 Footprint of research activities

Team members make common use of small to large-scale high performance computing platforms, which are energy consuming.

For this reason, previous research in the team [28] leveraged an existing consolidated simulation tool — SimGrid — for the bulk of experiments, using an experimental platform for validation only. For comparison, the validation experiments required ≈ 88 hours on nine nodes, while the simulation results that made into the paper would take at least 569 days to run. Although using and adapting the simulation tool took a certain effort, it allowed for more extensive evaluation, in addition to decreasing the footprint of this research. A similar strategy is being used in other projects since then.

5.2 Impact of research results

The digital sector is an ever-growing consumer of energy. Hence, it is of the utmost importance to increase the efficiency of use of digital tools. Our work on performance optimization, whether for high-end, energy consuming supercomputers, or more modest systems, aims at reducing the footprint of computations.

Because the aim of these machines is to be used at their maximum capacity, given their high production cost to amortize, we consider that our research results will not lead to a decrease in the overall use of computer systems; however, we expect them to lead to better modeling the energy consumption of application and hence a usage of their energy, hence resulting in “more science per watt”. Of course it is always hard to evaluate the real impact as a possible rebound effect is for more users to run on these machines, or users deciding to run extra experiments “because it is possible”.

5.3 Influence of team members

Members of the team participated to the writing of the *Inria global Action plan on F/M professional equality for 2021-2024*.

Moreover, Méline TROCHON, Ph.D. student in the team, is a member of the *Groupe de Travail Parité-Egalité* from the Inria Center at the University of Bordeaux (project.inria.fr/pariteegalitebordeaux/).

6 Highlights of the year

- Mihail Popov, Inria researcher (Inria Starting Faculty Position) joined the team in October 2024.
- The Associate Team DECoHPC, a collaboration of the team with Brazilian institutions including the National Laboratory for Scientific Computing (LNCC) and the Federal Fluminense University (UFF), was approved in 2024 for three years.
- Mahamat Younous Abdraman was hired as an engineer in 2024 in the context of the Numpex Project. Mahamat also completed an internship with the team in the same year.

6.1 Awards

Brice Goglin received the Académie des Sciences - Inria - Dassault Systèmes innovation award with Samuel Thibault (STORM) for their work on the HWLOC software.

7 New software, platforms, open data

7.1 New software

7.1.1 hwloc

Name: Hardware Locality

Keywords: NUMA, Multicore, GPU, Affinities, Open MPI, Topology, HPC, Locality

Functional Description: Hardware Locality (hwloc) is a library and set of tools aiming at discovering and exposing the topology of machines, including processors, cores, threads, shared caches, NUMA memory nodes and I/O devices. It builds a widely-portable abstraction of these resources and exposes it to applications so as to help them adapt their behavior to the hardware characteristics. They may consult the hierarchy of resources, their attributes, and bind task or memory on them.

hwloc targets many types of high-performance computing applications, from thread scheduling to placement of MPI processes. Most existing MPI implementations, several resource managers and task schedulers, and multiple other parallel libraries already use hwloc.

News of the Year: In 2024, the support for heterogeneous memory was further improved to ease the selection of best memory targets in a more portable way. Support for GPUs from several vendors was also enhanced. Newest discovery and binding features in different operating systems were also leveraged in hwloc. Many internal changes were implemented to prepare the 3.0 major release in 2025.

URL: http://www.open-mpi.org/projects/hwloc/

Publications: inria-00429889, hal-00985096, hal-01183083, hal-01330194, hal-01400264, hal-01402755, hal-01644087, hal-02266285

Contact: Brice Goglin

Participants: Brice Goglin, Valentin Hoyet

Partners: Open MPI consortium, Intel, AMD, IBM

7.1.2 Hsplit

Name: Hardware communicators split

Keyword: Topology

Scientific Description: Hsplit is a library that implements an abstraction allowing the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.

Functional Description: Hsplit implements an abstraction that allows the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.

URL: https://gitlab.inria.fr/hsplit/hsplit

Publications: hal-01937123v2, hal-01621941, hal-01538002

Contact: Guillaume Mercier

Participants: Guillaume Mercier, Brice Goglin, Emmanuel Jeannot, Thibaut Pepin

7.1.3 TopoMatch

Scientific Description: TopoMatch embeds a set of algorithms to map processors/cores in order to minimize the communication cost of the application.

Important features are : the number of processors can be greater than the number of applications processes , it assumes that the topology is a tree and does not require valuation of the topology (e.g. communication speeds) , it implements different placement algorithms that are switched according to the input size.

Some core algorithms are parallel to speed-up the execution. Optionally embeds scotch for fix-vertex mapping. enable exhaustive search if required. Several metric mapping are computed. Allow for oversubscribing of ressources. multithreaded.

TopoMatch is integrated into various software such as the Charm++ programming environment as well as in both major open-source MPI implementations: Open MPI and MPICH2.

Functional Description: TreeMatch is a library for performing process placement based on the topology of the machine and the communication pattern of the application.

URL: https://gitlab.inria.fr/ejeannot/topomatch

Publication: hal-03780662

Contact: Emmanuel Jeannot

Participants: Adele Villiermet, Emmanuel Jeannot, François Tessier, Guillaume Mercier, Pierre Celor

7.1.4 NewMadeleine

Name: NewMadeleine: An Optimizing Communication Library for High-Performance Networks

Keywords: High-performance calculation, MPI communication

Functional Description: NewMadeleine is the fourth incarnation of the Madeleine communication library. The new architecture aims at enabling the use of a much wider range of communication flow optimization techniques. Its design is entirely modular: drivers and optimization strategies are dynamically loadable software components, allowing experimentations with multiple approaches or on multiple issues with regard to processing communication flows.

The optimizing scheduler SchedOpt targets applications with irregular, multi-flow communication schemes such as found in the increasingly common application conglomerates made of multiple programming environments and coupled pieces of code, for instance. SchedOpt itself is easily extensible through the concepts of optimization strategies (what to optimize for, what the optimization goal is) expressed in terms of tactics (how to optimize to reach the optimization goal). Tactics themselves are made of basic communication flows operations such as packet merging or reordering.

The communication library is fully multi-threaded through its close integration with PIOMan. It manages concurrent communication operations from multiple libraries and from multiple threads. Its MPI implementation MadMPI fully supports the MPI_THREAD_MULTIPLE multi-threading level.

URL: https://pm2.gitlabpages.inria.fr/newmadeleine/

Publications: inria-00127356, inria-00177230, inria-00177167, inria-00327177, inria-00224999, inria-00327158, tel-00469488, hal-02103700, inria-00381670, inria-00408521, hal-00793176, inria-00586015, inria-00605735, hal-00716478, hal-01064652, hal-01087775, hal-01395299, hal-01587584, hal-02103700, hal-02407276, hal-03012097, hal-03118807

Contact: Alexandre Denis

Participants: Alexandre Denis, Clément Foyer, Nathalie Furmento, Raymond Namyst, Adrien Guilbaud, Florian Reynier, Philippe Swartvagher

7.1.5 IOPS

Name: I/O Performance Evaluation Suite

Keywords: I/O, HPC, Benchmarking

Functional Description: The tool generates a full characterization of the storage system in a high-performance computing environment. It utilizes well-known benchmarks (such as IOR) and manages the creation and execution of the tests, varying different parameters in order to create a comprehensive picture of how the I/O system behaves under different I/O patterns and parameters. In the first version, it includes the following functionalities:

1 Support for the IOR benchmark 2. Generation of an HTML report with graphs and the parameters that include the peak performance 3. Support for peak search considering the number of compute nodes, file size, number of object storage targets, and two distinct I/O patterns (one shared file and one file per process)

Release Contributions: Launch Version 1.0 of IOPS The first version of IOPS is static but functional. At this stage, we aim to have a functional version that implements the basic concepts of our codebase, specifically the rounds and the reports. Therefore, this version will be simple but very important for testing and validating our code architecture. For now, our focus will be on delivering the following features:

1. Support sequential test execution by varying the parameters in a predefined sequence, such as incremental variation. 2. Use IOR, considering only write operations. 3. Generate a simple report that will display the graphs and the parameters that achieved peak performance.

Contact: Luan Teylo Gouveia Lima

Participants: Luan Teylo Gouveia Lima, Francieli Zanon-Boito, Mahamat Younous Abdraman

7.1.6 AGIOS

Name: Application-guided I/O Scheduler

Keywords: High-Performance Computing, Scheduling

Functional Description: A user-level I/O request scheduling library that works at file level. Any service that handles requests to files (parallel file system clients and/or data servers, I/O forwarding frameworks, etc) may use the library to schedule these requests. AGIOS provides multiple scheduling algorithms, including dynamic options that change algorithms during the execution. It is also capable of providing many statistics in general and per file, such as average offset distance and time between requests. Finally, it may be used to create text-format traces.

URL: https://github.com/francielizanon/agios

Publications: hal-03758890, hal-01994677, hal-02079899, hal-01247942

Contact: Francieli Zanon-Boito

Participants: Luan Teylo Gouveia Lima, Alessa Mayer, Louis Peyrondet

7.1.7 SCOTCH

Name: Scotch / PT-Scotch

Keywords: Mesh partitioning, Domain decomposition, Graph algorithmics, High-performance calculation, Sparse matrix ordering, Static mapping

Scientific Description: The aim of the Scotch project is to tackle the problems of partitioning and mapping very large graphs, by way of algorithms that rely only on graph topology, and to devise efficient shared-memory, distributed-memory, and hybrid parallel algorithms for this purpose.

Functional Description: Scotch is a graph partitioner. It helps optimise the division of a problem, by means of a graph, into a set of independent sub-problems of equivalent sizes. These sub-problems can also be solved in parallel.

Release Contributions: SCOTCH has many interesting features:

- Its capabilities can be used through a set of stand-alone programs as well as through the libSCOTCH library, which offers both C and Fortran interfaces.

- It provides algorithms to partition graph structures, as well as mesh structures defined as node-element bipartite graphs and which can also represent hypergraphs.

- The SCOTCH library dynamically takes advantage of POSIX threads to speed-up its computations. The PT-SCOTCH library, used to manage very large graphs distributed across the nodes of a parallel computer, uses the MPI interface as well as POSIX threads.

- It can map any weighted source graph onto any weighted target graph. The source and target graphs may have any topology, and their vertices and edges may be weighted. Moreover, both source and target graphs may be disconnected. This feature allows for the mapping of programs onto disconnected subparts of a parallel architecture made up of heterogeneous processors and communication links.

- It computes amalgamated block orderings of sparse matrices, for efficient solving using BLAS routines.

- Its running time is linear in the number of edges of the source graph, and logarithmic in the number of vertices of the target graph for mapping computations.

- It can handle indifferently graph and mesh data structures created within C or Fortran programs, with array indices starting from 0 or 1.

- It offers extended support for adaptive graphs and meshes through the handling of disjoint edge arrays.

- It is dynamically parametrizable thanks to strategy strings that are interpreted at run-time.

- It uses system memory efficiently, to process large graphs and meshes without incurring out-of-memory faults,

- It is highly modular and documented. Since it has been released under the CeCILL-C free/libre software license, it can be used as a testbed for the easy and quick development and testing of new partitioning and ordering methods.

- It can be easily interfaced to other programs..

- It provides many tools to build, check, and display graphs, meshes and matrix patterns.

- It is written in C and uses the POSIX interface, which makes it highly portable.

News of the Year: The Member's contract for the Scotch Consortium has been finalized. A full-time core software engineer has been hired.

URL: http://www.labri.fr/~pelegrin/scotch/

Publications: hal-04404141, hal-01671156, hal-01968358, hal-00648735, tel-00540581, hal-00301427, hal-00402893, tel-00410402, hal-00402946, hal-00410408, hal-00410427

Contact: François Pellegrini

Participants: François Pellegrini, Sebastien Fourestier, Marc Fuentes, Clément Barthelemy, Jun Ho Her, Cedric Chevalier, Xavier Muller, Amaury Jacques, Selmane Lebdaoui, Tetsuya Mishima

Partners: Université de Bordeaux, IPB, CNRS, Region Aquitaine

7.1.8 Raisin

Keywords: Hypergraph, Partitioning, Graph algorithmics, Static mapping, FPGA

Functional Description: Raisin is a multi-valued oriented hypergraph partitioning software whose objective function is to minimize the length of the longest path between some types of vertices while limiting the number of cut hyper-arcs.

Release Contributions: Raisin has been designed to solve the problem of circuit placement onto multi-FPGA architectures. It models the circuit to map as a set of red-black, directed, acyclic hypergraphs (DAHs). Hypergraph vertices can be either red vertices (which represent registers and external I/O ports) or black vertices (which represent internal combinatorial circuits). Vertices bear multiple weights, which define the types of resources needed to map the circuit (e.g., registers, ALUs, etc.). Every hyper-arc comprises a unique source vertex, all other ends of the hyper-arcs being sinks (which models the transmission of signals through circuit wiring). A circuit is consequently represented as set of DAHs that share some of their red vertices.

Target architectures are described by their number of target parts, the maximum resource capacities within each target part, and the connectivity between target parts.

The main metric to minimize is the length of the longest path between two red vertices, that is, the critical path that signals have to traverse during a circuit compute cycle, which correlates to the maximum frequency at which the circuit can operate on the given target architecture.

Raisin computes a partition in which resource capacity constraints are respected and the critical path length is kept as small as possible, while reducing the number of cut hyper-arcs. It produces an assignment list, which describes, for each vertex of the hypergraphs, the part to which the vertex is assigned.

Raisin has many interesting features:

- It can map any weighted source circuit (represented as a set of red-black DAHs) onto any weighted target graph.

- It is based on a set of graph algorithms, including a multi-level scheme and local optimization methods of the “Fiduccia-Mattheyses” kind.

- It contains two greedy initial partitioning algorithms that have a computation time that is linear in the number of vertices. Each algorithm can be used for a particular type of topology, which can make them both complementary and efficient, depending on the problem instances.

- It takes advantage of the properties of DAHs to model path lengths with a weighting scheme based on the computation of local critical paths. This weighting scheme allows to constrain the clustering algorithms to achieve better results in smaller time.

- It can combine several of its algorithms to create dedicated mapping strategies, suited to specific types of circuits.

- It provides many tools to build, check and convert red-black DAHs to other hypergraph and graph formats.

- It is written in C.

Publications: hal-03596218, hal-04008677, hal-04379716, hal-03604540v1

Contact: Julien Rodriguez

Participants: François Galea, François Pellegrini, Lilia Zaourar, Julien Rodriguez

7.2 New platforms

7.2.1 PlaFRIM

Participants: Brice Goglin.

Name: Plateforme Fédérative pour la Recherche en Informatique et Mathématiques

Website: plafrim.fr

Description: PlaFRIM is an experimental platform for research in modeling, simulations and high performance computing. This platform has been set up from 2009 under the leadership of Inria Bordeaux Sud-Ouest in collaboration with computer science and mathematics laboratories, respectively LaBRI and IMB with a strong support in the region Aquitaine.

It aggregates different kinds of computational resources for research and development purposes. The latest technologies in terms of processors, memories and architecture are added when they are available on the market. As of 2023, it contains more than 6,000 cores, 50 GPUs and several large memory nodes that are available for all research teams of Inria Bordeaux, Labri and IMB.

Brice Goglin is in charge of PlaFRIM since June 2021.

7.3 Open data

Write performance with different numbers of OSTs for BeeGFS in PlaFRIM

Contributors: Francieli Zanon-Boito and Luan Teylo Gouveia Lima

Description: This data set contains performance measured with the IOR benchmarking tool when writing to the BeeGFS parallel file system following different strategies and using different numbers of OSTs.

Dataset PID (DOI,...): doi.org/10.5281/zenodo.10518126

Project link: zenodo.org/records/10518127

Publications: [4]

8 New results

8.1 Phase-based Data Placement Optimization in Heterogeneous Memory

Participants: Pierre Clouzet, Brice Goglin, Emmanuel Jeannot.

While scientific applications show increasing de- mand for memory speed and capacity, the performance gap between compute cores and the memory subsystem continues to spread. In response, heterogeneous memory systems integrat- ing high-bandwidth memory (HBM) and non-volatile memory (NVM) alongside traditional DRAM on the CPU side are gaining traction. Despite the potential benefits of optimized memory selection for improved performance and efficiency, adapting applications to leverage diverse memory types often requires extensive modifications. Moreover, applications often comprise multiple execution phases with varying data access patterns. Since the capacity of the "fastest" memory is limited, relying solely on fixed data placement decisions may not yield optimal performance. Thus, considering allocation lifetimes and dynamically migrating data between memory types becomes imperative to ensure that performance-critical data for each phase resides in fast memory. To address these challenges, we developed a workflow in- corporating memory access profiling, optimization techniques and a runtime system, which selects initial data placement for allocations and performs data migration during execution, considering the platform's memory subsystem characteristics and capacities. We formalize the optimization problems for initial and phase-based data placement and propose heuristics derived from memory profiling metrics to solve it. Additionally, we outline the implementation of these approaches, including allocation inter- ception to enforce placement decisions. Experiments conducted with several applications on an Intel Ice Lake (DRAM+NVM) and Sapphire Rapids (HBM+DRAM) system demonstrate that our methodology can effectively bridge the performance gap between slow and fast memory in heterogeneous memory environments. This work [11] is performed in collaboration with RWTH Aachen and Université of Reims Champagne Ardenne in the context of the H2M ANR-DFG project.

8.2 Performance Projection for Design-Space Exploration on future HPC Architectures

Participants: Clément Gavoille, Brice Goglin, Emmanuel Jeannot.

To address the growing need for performance from future HPC machines, their processor designs are constantly evolving. Assessing the impact of changes in hardware, software stack, and applications on performance is crucial in a codesign process. Here, we propose a performance projection workflow to facilitate the initial exploration of design space for multicore nodes and multi-threaded applications. For this purpose, we analyze the architectural efficiency of an accessible source machine and determine the maximum sustainable flop/s performance of a hypothetical target machine based on its software stack on a per-thread basis. Finally, we use these characterizations to project the performance evolution from the source machine to the target machine.

In this work, we assess the strengths and weaknesses of our approach by integrating it into the Fugaku-Next Feasibility Study. We compare the accuracy and overhead of our approach with the gem5 cycle-level simulations and a fast exploration methodology based on Machine Code Analyzer (MCA), using NAS Parallel benchmarks and CCS-QCD, a quantum chromodynamics miniapp. The study demonstrates that, compared to gem5, our approach has a prediction deviation of 5% for most cases and up to 30% for extreme cases. Additionally, it exhibits an execution overhead an order of magnitude bigger than MCA but orders of magnitude smaller than gem5.

Finally, we demonstrate our approach's capability to study larger scale and more representative applications than gem5, such as QWS and Genesis, two applications of RIKEN optimized for Fugaku.

This work [6] was performed in collaboration with CEA/DAM and RIKEN.

8.3 User-space interrupts for HPC communications

Participants: Alexandre Denis, Brice Goglin, Charles Goedefroit.

In HPC, network are programmed directly from user space, since system call have a significant cost with low latency networks. Usually, the user performs polling: the network is polled at regular intervall to check whether a new message has arrived. However, it wastes some resources. Another solution is to rely on interrupts instead of polling, but since interrupts are managed by the kernel, they involve system calls we are precisely willing to avoid.

Intel introduced user-level interrupts on its lates Sapphire Rapids CPUs, allowing to use interrupts from user space. These user space interrupts may be a viable alternative to polling, by using interrupts without the cost of systems calls.

We have extended [8] to Atos BXI network to make it trigger user-space interrupts so as to benefit from uintr in inter-node communications.

8.4 Interrupt-safe data structures

Participants: Alexandre Denis, Charles Goedefroit.

With the addition of interrupt-based communication in NewMadeleine, synchronization issues have emerged in some data structures. NewMadeleine relies on lock-free queues for a lot of its activities: progression through Pioman, submission queue, completion queue, deferred tasks. However, our implementation of lock-free queues was not non-blocking and was not suitable for use in an interrupt handler.

Other implementations found in the litterature target scalability but exhibit high latency in the uncontended case. We have shown that, since latency of network and queues are different by several orders of magnitude, even highly contented network operation do not impose a high pressure on queues.

We have proposed [5] a new non-blocking queue algorithm that is optimized for low contention, while degrading nicely in case of higher contention. We have shown that it exhibits the best performance in NewMadeleine when compared to 15 other queue designs on four different architectures.

8.5 Communication priorities with StarPU/NewMadeleine

Participants: Alexandre Denis, Jean-Alexandre Collin.

We worked on the integration between StarPU and NewMadeleine. NewMadeleine supports priority-based packet scheduling, used by StarPU. However, the behavior of this strategy is not well understood, in particular in regard with interaction of priorities and rendez-vous protocol, and between multiple flows with different destination.

We have run some applications and traced their behavior. We have developped a tool to merge StarPU traces and NewMadeleine traces to have a good overview of events.

8.6 MPI Application Skeletonization

Participants: Quentin Buot, Emmanuel Jeannot, Guillaume Mercier, Richard Sartori.

Fine tuning MPI meta parameters is a critical task for HPC systems, but measuring the impact of each parameters takes a lot of time. In a previous work, we used the Bayesian Optimization technique [9] in order to speed up considerably the time taken to explore the solution space and to determine a relevant parametrization.

Another solution, yet complementary to the first one, is to work on a modified version of the target application to retain only the parts dealing with its communication pattern. This modified version, called the skeleton, preserves the communication pattern while removing other compute instructions. It faithfully represents the original program's communication behavior while running significantly faster.

However, this process of communication pattern extraction is tedious and error-prone when performed manually. Leveraging the LLVM infrastructure, this tool adresses the issue by automatically extracting this standalone mini-app (skeleton) from any MPI application.

It can then be used as a proxy during the optimization phase, reducing its duration by 95%. When paired with a generic optimization tool called ShaMAN [30] it allows to generate a MPI tuning configuration that exhibit the same performances of the configuration obtained through exhaustive exploration and benchmarking.

8.7 Network Topology Reconstruction

Participants: Brice Goglin, Guillaume Mercier, Thibaut Pépin.

With the increase in size and complexity of supercomputers as well as the development of specialized modules, it has become important to match applications and communication libraries to the underlying network topology. This matching allows to minimize the time spent in high latency communications and limits potential contention on the network.

To this end, we developed a tool that reconstructs the network topology from a series of latency measurements on a cluster. We enhanced this tool by handling more edge cases to be as generic as possible and integrated it into other libraries such as MPC (an MPI implementation developed at CEA) and TADaaM's Hsplit (7.1.2).

Those integrations allowed us to experimentally expose the usefulness of the information about the network topology. As such, we noticed a non-negligible reduction in the execution time for the broadcast and gather MPI collective communication operations as well as an important reduction in the amount of data transiting on the network links of our test cluster. To help application developers, we developed a visualization tool to better show those phenomenons.

8.8 Adding topology and memory awareness in data aggregation algorithms

Participants: Emmanuel Jeannot.

With the growing gap between computing power and the ability of large-scale systems to ingest data, I/O is becoming the bottleneck for many scientific applications. Improving read and write performance thus becomes decisive, and requires consideration of the complexity of architectures. In this paper, we introduce TAPIOCA, an architecture-aware data aggregation library. TAPIOCA offers an optimized implementation of the two-phase I/O scheme for collective I/O operations, taking advantage of the many levels of memory and storage that populate modern HPC systems, and leveraging network topology. We show that TAPIOCA can significantly improve the I/O bandwidth of synthetic benchmarks and I/O kernels of scientific applications running on leading supercomputers. For example, on HACC-IO, a cosmology code, TAPIOCA improves data writing by a factor of 13 on nearly a third of the target supercomputer. [3]

This publication is the result of a former collaboration with Argonne National Lab and Inria Rennes.

8.9 Scheduling distributed I/O resources in HPC systems

Participants: Alexis Bandet, Guillaume Pallez, Francieli Zanon-Boito.

Parallel file systems cut files into fixed-size stripes and distribute them across a number of storage targets (OSTs) for parallel access. Moreover, a layer of I/O nodes is often placed between compute nodes and the PFS. In this context, it is important to notice both OST and I/O nodes are potentially shared by running applications, which may lead to contention and low I/O performance.

Contention-mitigation approaches usually see the shared I/O infrastructure as a single resource capable of a certain bandwidth, whereas in practice it is a distributed set of resources from which each application can use a subset. In addition, using X% of the OSTs, for example, does not grant a job X% of the PFS' peak performance. Indeed, depending on their characteristics, each application will be impacted differently by the number of used I/O resources.

We conducted a comprehensive study of the problem of scheduling shared I/O resources — I/O nodes, OSTs, etc — to HPC applications. We tackled this problem by proposing heuristics to answer two questions: 1) how many resources should we give each application (allocation heuristics), and 2) which resources should be given to each application (placement heuristics). These questions are not independent, as using more resources often means sharing them. Nonetheless, our two-step approach allows for simpler heuristics that would be usable in practice.

In addition to overhead, an important aspect that impacts how "implementable" algorithms are is their input regarding applications' characteristics, since this information is often not available or at least imprecise. Therefore, we proposed heuristics that use different input and studied their robustness to inaccurate information.

This work was published in Euro-Par 2024 [4] (with an extended version published as a technical report [21]) and included the publication of a large data set of I/O performance results.

8.10 Prediction and Interpretability of HPC I/O Resources Usage with Machine Learning

Participants: Alexis Bandet, Guillaume Pallez, Francieli Zanon-Boito.

During the work on heuristics for allocation of I/O resources to HPC applications, we observed that the best algorithm requires to know the number of resources that maximize application I/O performance. Nonetheless, this information is not typically available, and obtaining it would involve running the application multiple times with multiple configurations. Instead, in this work [20], we focus on finding a good estimate of the number of I/O resources (e.g., OSTs and I/O nodes) that provides the maximal bandwidth while minimizing the system occupation and taking into account the natural I/O variability. We use machine learning techniques to do so, focusing on intrinsic application features and system configurations. We show I/O resource usage is predictable and further study the impact of different features. We also validate our models with four I/O kernels from real applications. Finally, we show that our model, when used for resource allocation, can improve application performance.

8.11 Implementation of an unbalanced I/O Bandwidth Management system in a Parallel File System

Participants: Clément Barthelemy, Francieli Zanon-Boito, Emmanuel Jeannot, Axel Malmgren, Guillaume Pallez, Luan Teylo.

I/O scheduling is a technique to mitigate contention in the access to the shared parallel file system. Despite its popularity in the literature, most proposed I/O scheduling techniques are *not* used in practice

due to the difficulty of obtaining the needed information and due to overhead concerns. Previously, members of the team proposed IO-Sets [28], which compared to other techniques uses little information on the applications, making it attractive to be used in practice.

Since then, we have worked to put the technique in practice. For that, we chose to implement it inside the parallel file system to make it transparent to applications, and without a centralized control. That means relaxing the method, but mitigates its limitations regarding overhead and fault tolerance.

In a first report[22], we discuss a BeeGFS-based implementation in details and present preliminary results. Afterwards, during the internship of Axel Malmgren, this implementation was improved and debugged. Since then, we have been working in evaluating it under different situations to fully understand the impact of the adaptations made to the IO-Sets method. In other words, we want to know to what extent it remains a good approach in a production environment.

8.12 FTIO: Detecting I/O Periodicity Using Frequency Techniques

Participants: Alexis Bandet, Guillaume Pallez, Francieli Zanon-Boito.

The work on IO-Sets evidenced that knowing the periodicity of applications' I/O phases is useful to improve I/O performance and mitigate contention. However, describing the temporal I/O behavior in terms of I/O phases is a challenging task. Indeed, the HPC I/O stack only sees a stream of issued requests and does not provide I/O behavior characterization. Contrary, the notion of an I/O phase is often purely logical, as it may consist of a set of independent I/O requests, issued by one or more processes and threads during a particular time window, and popular APIs do not require that applications explicitly group them.

Thus, a major challenge is to draw the borders of an I/O phase. Consider, for example, an application with 10 processes that writes 10 GB by generating a sequence of two 512 MB write requests per process, then performs computation and communication for a certain amount of time, after which it writes again 10 GB. How do we assert that the first 20 requests correspond to the first I/O phase and the last 20 to a second one? An intuitive approach is to compare the time between consecutive requests with a given threshold to determine whether they belong to the same phase. Naturally, the suitable threshold should depend on the system. The reading or writing method can make this an even more complex challenge, as accesses can occur, e.g., during computational phases in the absence of barriers. Hence, the threshold would not only be system dependent but also application dependent, making this intuitive approach more complicated than initially expected.

Even assuming that one is able to find the boundaries of various I/O phases, this might still not be enough. Consider for example an application that periodically writes large check- points with all processes. In addition, a single process writes, at a different frequency, only a few bytes to a small log file. Although both activities clearly constitute I/O, only the period of the checkpoints is relevant to contention-avoidance techniques. If we simply see I/O activity as belonging to I/O phases, we may observe a profile that does not reflect the behavior of interest very well.

In this research [15], published at IPDPS 2024, we proposed FTIO, a tool for characterizing the temporal I/O behavior of an application using frequency techniques such as DFT and autocorrelation. FTIO imposes generate only a modest amount of information and hence imposes minimal overhead. We also proposed metrics that quantify the confidence in the obtained results and further characterize the I/O behavior based on the identified period.

This work was a collaboration with Ahmad TARRAF and Felix WOLF from the Technical University of Darmstadt, Germany, in the context of the ADMIRE project.

8.13 Temporal I/O Behavior of HPC Applications analysis

Participants: Luan Teylo, Mihail Popov, Francieli Zanon-Boito.

We studied the temporal I/O behavior of over 440,000 jobs running on four HPC systems, all different in terms of infrastructure, scale, and users, covering several time periods over the last 11 years. The data we analyzed came either from parallel file systems (system-side traces) or from I/O monitoring tools (application-side traces).

The aim of analyzing these traces is to provide an in-depth study of data accesses by HPC applications in the wild. We have thus identified and addressed a number of questions dealing with the temporality of I/Os, their periodicity, the existence and prevalence of certain patterns, I/O concurrency between applications or user practices. We also proposed a classification of temporal I/O behaviors, which shows a few patterns are able to represent a vast majority of jobs. Overall, the results of this study provide relevant information for anyone working to improve high-performance I/O. They also serve as a basis for future research into both behavior detection tools and the use of trace analysis, particularly for scheduling and application optimization.

This work was accepted for publication at IPDPS'2025 [27] and is the result of a collaboration between Inria Bordeaux, Inria Rennes, the Technical University of Darmstadt, and the National Laboratory for Scientific Computing (LNCC).

8.14 Improvement of the usability of SCOTCH and PT-SCOTCH

Participants: Aymen Ali Yahia, Clément Barthélemy, Mark Fuentes, Xavier Muller, François Pellegrini.

The SCOTCH software has undergone continuous development. The first axis of work, in line with the work of the previous year regarding the shared-memory parallelization of the graph partitioning algorithms, has allowed SCOTCH to use native Windows threads. Fixes have been provided to improve the use of SCOTCH on this platform, as well as on ARM platforms. These improvements have been made available to the community by way of two releases, v7.0.5 and v7.0.6.

Also, the SCOTCHPY module, which aims at providing a Pyton interface for SCOTCH, has been finalized, thanks to the internship of Aymen Ali Yahia. The Python interface, which is built on top of NUMPY and MPI4PY, provides an interface to SCOTCH as well as PT-SCOTCH.

8.15 Mapping circuits onto multi-FPGA platforms

Participants: François Pellegrini, Julien Rodriguez.

The work of Julien Rodriguez concerns the placement of digital circuits onto a multi-FPGA platform, in the context of a PhD directed by François Pellegrini, in collaboration with François Galea and Lilia Zaourar at CEA Saclay, and which has been defended this year [18]. Its aim was to design and implement mapping algorithms that do not minimize the cut, as it is the case in most par/titioning toolboxes, but the length of the longest path between sets of vertices. This metric strongly correlates to the critical path that signals have to traverse during a circuit compute cycle, hence to the maximum frequency at which a circuit can operate.

A common procedure for partitioning very large circuits is to apply the most expensive algorithms to smaller instances that are assumed to be representative of the larger initial problem. One of the most widely used methods for reducing the size of the problem is to use a circuit clustering algorithm in which bigger clusters (merging more than two vertices) can be created by a single round of the algorithm. We have studied clustering algorithms such as heavy edge matching, for which we have developed a new weighting function that favors the grouping of vertices along the critical path, *i.e.*, the longest path in the red-black hypergraph. We also developed our own clustering algorithm [14], which yields better results than heavy edge matching.

Furthermore, in the same work, we demonstrated that if the delay cost between two clusters is greater than the critical path, and if the number of clusters is sufficiently large (larger than the ratio between the

delay cost and the critical path), the known approximation ratio of $M^2 + M$ (M being the size of cluster) is reduced to M.

All the aforementioned algorithms have been integrated into the RAISIN software 7.1.8.

8.16 Quantum algorithms for graph partitioning

Participants: Julien Rodriguez.

With the recent availability of Noisy Intermediate-Scale Quantum (NISQ) devices, quantum variational and annealing-based methods have received increased attention. In particular, these methods are presented as promising for solving optimization problems. In this context, the community proposed different works to evaluate the potential of these new methods in comparison to classical approaches. However, application-level benchmarking of quantum process units (QPUs) usually requires consideration of the entire quantum computer programming stack. Hence, we established in [7] a new protocol to generate graph instances with their associated near-optimal minor-embedding mappings to D-Wave Quantum Annealers (QA). This set of favorable mappings is used to generate a wide variety of optimization problem instances. These large instances of constrained and unconstrained optimization problems are used to compare the performance of the QPU against efficient exact classical solvers. The benchmark aims to evaluate and quantify the key characteristics of instances that could benefit from the use of a quantum computer. In this context, existing QPUs appear to be best suited for unconstrained problems on instances with densities less than 10%.

8.17 Predicting and Fixing Errors in Parallel Applications with AI

Participants: Asia Auville, Jad El-Karchi, Emmanuelle Saillard, Mihail Popov.

Investigating if parallel applications are correct is a very challenging task. Yet, recent progress in ML and text embedding show promising results in characterizing source code or the compiler intermediate representation to identify optimizations. We propose to transpose such characterization methods to the context of verification. In particular, we train ML models that take as labels the code correctness along with intermediate representations embeddings as features. Results over small MPI verification benchmarks including MBI and DataRaceBench demonstrate that we can train models that detect if a code is correct with 90% accuracy and up to 75% over new unseen errors. This work [10] is a collaboration with the Iowa State University.

In the context of Asia Auville Ph.D. thesis, we are currently investigating the prediction capabilities of ML models to detected errors beyond simple errors, by considering more complicated errors through github repositories crawling. We are also planning to use LLMs models to not only detect errors, but also to propose fixes. This work is done in collaboration with the University of Versailles and Intel.

8.18 Optimizing Performance and Energy with AI Guided Exploration

Participants: Lana Scravaglieri, Olivier Aumage, Mihail Popov.

HPC systems expose configuration options that help users optimize their applications'execution. Questions related to the best thread and data mapping, number of threads, or cache prefetching have been posed for different applications, yet they have been mostly limited to a single optimization objective (e.g., performance) and a fixed application problem size. Unfortunately, optimization strategies that work well in one scenario may generalize poorly when applied in new contexts.

In previous work [1], we investigated the impact of configuration options and different problem sizes over both performance and energy: NUMA-related options and cache prefetchers provide significantly more gains for energy (5.9x) than performance (1.85x) over a standard baseline configuration.

In the context of Lana Scravaglieri's Ph.D. thesis and in collaboration with IFP Energies nouvelles (IFPEN), we carry this research further, by focusing on the exploration of SIMD transformations over carbon storage applications. To do so, we are designing a more general exploration infrastructure, CORHPEX, that can easly incorporate more diverse optimization knobs and applications. This work has been accepted at IPDPS'2025.

In collaboration with the University of Uppsala, we are also investigating the hardware prefetching intercation with the new hybrid architectures (Intel's Efficiency- and Performance-cores or ARM big.LITTLE). Preliminary results [29] showcase how enery gains can be achieved by tuning the system to the applications.

8.19 An Analysis of Performance Variability in AWS Virtual Machines

Participants: Miguel de Lima, Luan Teylo, Lucia Drummond.

Cloud computing platforms are essential for a wide range of applications, including High-Performance Computing (HPC) and artificial intelligence. However, the performance variability of virtual machines (VMs) in these shared environments presents significant challenges.

This work provides an extensive month-long analysis of the performance variability of C family VMs on Amazon Web Services (AWS) across two regions (us-east-1 and sa-east-1), various instance generations, and market types. Our findings indicate that Graviton processors (c6g.12xlarge and c7g.12xlarge) exhibit minimal performance variability and high cost-effectiveness, with the c7g.12xlarge instance, in particular, offering significantly reduced execution times and lower costs. Intel and AMD instances, while showing performance improvements from generation c6 to c7, exhibited up to 20% variability.

This study [12] was done in collaboration with the Federal Fluminense University in Brazil, through the Equipe Associée DecoHPC, and was published in SSCAD 2024.

8.20 A Framework for Executing Long Simulation Jobs Cheaply in the Cloud

Participants: Alan Nunes, Daniel Sodré, Cristina Boeres, José Viterbo, Lúcia Drummond, Vinod Rebello, Luan Teylo, Felipe Portella, Paulo Estrela, Renzo Malini.

This work presents the framework SIM@ClOUD that optimizes cost-related resource allocation decisions for simulation jobs in cloud environments. SIM@CLOUD offers comprehensive management of simulations throughout their execution life-cycle in the cloud, including the selection of Virtual Machine (VM) types across different regions and markets.

By leveraging Spt VMs and application checkpointing, the framework transparently reduces the monetary costs associated with the execution without client intervention. Historical data analysis enables the prediction of simulation execution times, which is refined further by a dynamic predictor for adaptive VM selection.

SIM@CLOUD is being deployed in an industrial setting and employs a cachebased storage solution to improve access latency to in-house data by VMs located in geographically distinct regions. An evaluation carried out on AWS EC2, using real oil reservoir simulations, demonstrates the effectiveness of the framework.

This study [13] was done in collaboration with the Federal Fluminense University in Brazil, through the Equipe Associée DecoHPC, and was published in the IEEE International Conference on Cloud Engineering (IC2E), 2024.

9 Bilateral contracts and grants with industry

9.1 Bilateral contracts with industry

CEA

Participants: Clément Gavoille, Brice Goglin, Guillaume Mercier, Thibaut Pépin.

- CEA/DAM granted the funding of the PhD thesis of Thibaut Pépin on communication on modular supercomputer architectures.
- CEA/DAM granted the fundind of the PhD thesis of Clément Gavoille, defended in January, which recently led to publication with RIKEN [6].

ATOS

Participants: Quentin Buot, Alexandre Denis, Brice Goglin, Emmanuel Jeannot, Guillaume Mercier, Richard Sartori.

- ATOS/Bull/eviden is funding the CIFRE PhD Thesis of Richard Sartori on the determination of optimal parameters for MPI applications deployment on parallel architectures
- ATOS/Bull/Eviden is funding the CIFRE PhD Thesis of Charles Goedefroit on Delivering Userspace Interrupts from the BXI network interface
- Quentin Buot is payed by Inria under a *plan de relance* contract with ATOS/Bull to work at Eviden Facilities at Grenoble (80% of the time)

IFPEN

Participants: Mihail Popov, Lana Scravaglieri.

- IFPEN is funding the PhD Thesis of Lana Scravaglieri on the designs of models to optimize numerical simulations by adjusting the programs to the underline HPC systems.

10 Partnerships and cooperations

10.1 International initiatives

10.1.1 Associate Teams in the framework of an Inria International Lab or in the framework of an Inria International Program

DECoHPC

Participants: Francieli Boito, Laércio Lima, Luan Teylo, Diego Carvalho, Lucia Drummond, Mariza Ferro, Philippe Navaux, Kary Ocaña, Carla Osthoff.

Title: Data movement, Energy COnsumption and performance in High-Performance Computing

Partner Institution(s):

- National Laboratory for Scientific Computing (LNCC), Brazil
- Federal Fluminense University (UFF) Brazil

- Federal University of Rio Grande do Sul (UFRGS), Brazil
- Federal Center for Technological Education of Rio de Janeiro (CEFET-RJ), Brazil

Date/Duration: from 2024 to 2026

Website: team.inria.fr/decohpc/

Additionnal info/keywords: Supercomputers were conceived to efficiently run traditional HPC applications, namely numerical simulations. However, in the context of the convergence between HPC and big data, their workload is becoming more heterogeneous. In this new scenario, efficient application execution becomes more challenging. Moreover, energy consumption has emerged as an important concern for HPC and computer science in general. First, with the effects of climate change, environmental concerns have become a major focus across various scientific fields. Second, as more and more exascale machines emerge, the energy budget has become one of the main concerns for these machines, driven not only by environmental considerations but also by economic ones.

The previous HPCProSol associate team (2021–2023) provided us with performance insights about two kinds of representative applications from the Santos Dumont system from the LNCC (the largerst supercomputer in Latin America): finite element methods (HPC) and bioinformatics workflows (HPDA). Moreover, we collaborated on advancing the system's monitoring infrastructure by developing software to efficiently process it. Now, in the DECoHPC associate team, we aim to take these insights and tools and extend them towards our three main goals:

- (WP1) Based on the Santos Dumont's traces (recently made available), to obtain a holistic view of the I/O behavior of HPC applications. We want to classify applications according to their behaviors — and on their different needs from the system.
- (WP2) To study and characterize the energy consumption of moving applications' data through the network and I/O infrastructure.
- (WP3) To characterize the I/O performance and energy consumption of AI applications, which have not been explored in HPCProSol, but are now among one of the most important users of HPC facilities.

10.2 International research visitors

10.2.1 Visits of international scientists

Other international visits to the team

Scott Klasky and Ana Gainaru

Status researchers

Institution of origin: Oak-Ridge National Labs

Country: USA

Dates: June 10 to June 13.

Context of the visit: they gave a tutorial on the ADIOS2 I/O library and worked with the team on joint research on the allocation of OSTs in HPC.

Mobility program/type of mobility: research stay and lecture.

Jay Lofstead

Status researcher

Institution of origin: Sandia National Laboratories.

Country: USA

Dates: June 24 to June 26.

Context of the visit: he presented his recent work on using relational databases for HPC and discussed with the team about the IOPS tool and its possible integration with IO500.

Mobility program/type of mobility: research stay and lecture.

Douglas Brum

Status Master's student

Institution of origin: Federal Fluminense University (UFF)

Country: Brazil

Dates: October 3 to 21.

Context of the visit: collaboration in the DECoHPC Associate Team.

Mobility program/type of mobility: research stay and lecture.

André Ramos Carneiro

Status technical staff.

Institution of origin: National Laboratory for Scientific Computing (LNCC).

Country: Brazil.

Dates: October 15 to 18.

Context of the visit: collaboration in the DECoHPC Associate Team.

Mobility program/type of mobility: research stay and lecture.

10.2.2 Visits to international teams

Research stays abroad

Luan Teylo Gouveia Lima

Visited institution: Federal Fluminense University (UFF)

Country: Brazil

Dates: September 15 to 30.

Context of the visit: collaboration in the DECoHPC Associate Team.

Mobility program/type of mobility: research stay and lecture.

10.3 European initiatives

10.3.1 H2020 projects

ADMIRE ADMIRE project on cordis.europa.eu

Participants: Alexis Bandet, Clément Barthelemy, Emmanuel Jeannot, Luan Teylo, Francieli Zanon-Boito.

Title: Adaptive multi-tier intelligent data manager for Exascale

Duration: From April 1, 2021 to June 30, 2024

Partners:

- DATADIRECT NETWORKS FRANCE, France
- INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET AUTOMATIQUE (INRIA), France
- JOHANNES GUTENBERG-UNIVERSITAT MAINZ, Germany
- KUNGLIGA TEKNISKA HOEGSKOLAN (KTH), Sweden
- FORSCHUNGSZENTRUM JULICH GMBH (FZJ), Germany
- INSTYTUT CHEMII BIOORGANICZNEJ POLSKIEJ AKADEMII NAUK, Poland
- UNIVERSITA DI PISA (UNIPI), Italy
- E 4 COMPUTER ENGINEERING SPA (E4), Italy
- UNIVERSITE DE BORDEAUX (UBx), France
- UNIVERSITA DEGLI STUDI DI MILANO (UMIL), Italy
- PARATOOLS SAS (PARATOOLS SAS), France
- TECHNISCHE UNIVERSITAT DARMSTADT, Germany
- MAX-PLANCK-GESELLSCHAFT ZUR FORDERUNG DER WISSENSCHAFTEN EV (MPG), Germany
- CINECA CONSORZIO INTERUNIVERSITARIO (CINECA), Italy
- UNIVERSIDAD CARLOS III DE MADRID (UC3M), Spain
- BARCELONA SUPERCOMPUTING CENTER CENTRO NACIONAL DE SUPERCOMPUTACION (BSC CNS), Spain
- CONSORZIO INTERUNIVERSITARIO NAZIONALE PER L'INFORMATICA (CINI), Italy

Inria contact: Emmanuel JEANNOT

Coordinator: Jesus Carretero (Universidad Carlos 3 de Madrid)

Summary: The growing need to process extremely large data sets is one of the main drivers for building exascale HPC systems today. However, the flat storage hierarchies found in classic HPC architectures no longer satisfy the performance requirements of data-processing applications. Uncoordinated file access in combination with limited bandwidth make the centralised back-end parallel file system a serious bottleneck. At the same time, emerging multi-tier storage hierarchies come with the potential to remove this barrier. But maximising performance still requires careful control to avoid congestion and balance computational with storage performance. Unfortunately, appropriate interfaces and policies for managing such an enhanced I/O stack are still lacking.

The main objective of the ADMIRE project is to establish this control by creating an active I/O stack that dynamically adjusts computation and storage requirements through intelligent global coordination, malleability of computation and I/O, and the scheduling of storage resources along

all levels of the storage hierarchy. To achieve this, we will develop a software-defined framework based on the principles of scalable monitoring and control, separated control and data paths, and the orchestration of key system components and applications through embedded control points.

Our software-only solution will allow the throughput of HPC systems and the performance of individual applications to be substantially increased – and consequently energy consumption to be decreased – by taking advantage of fast and power-efficient node-local storage tiers using novel, European ad-hoc storage systems and in-transit/in-situ processing facilities. Furthermore, our enhanced I/O stack will offer quality-of-service (QoS) and resilience. An integrated and operational prototype will be validated with several use cases from various domains, including climate/weather, life sciences, physics, remote sensing, and deep learning.

EUPEX

Participants: Brice Goglin.

- EUPEX: European Pilot for Exascale
- Program: H2020 EuroHPC
- Grant Agreement number: 101033975 – H2020-JTI-EuroHPC-2020-01
- 2022-2026
- Partners: Atos, FZJ, CEA, GENCI, CINECA, E4, ICS-FORTH, Cini National Lab, ECMWF, IT4I, FER, ParTec, EXAPSYS, INGV, Goethe University, SECO, CybeleTech
- The EUPEX pilot brings together academic and commercial stakeholders to co-design a European modular Exascale-ready pilot system. Together, they will deploy a pilot hardware and software platform integrating the full spectrum of European technologies, and will demonstrate the readiness and scalability of these technologies, and particularly of the Modular Supercomputing Architecture (MSA), towards Exascale.

  EUPEX's ambition is to support actively the European industrial ecosystem around HPC, as well as to prepare applications and users to efficiently exploit future European exascale supercomputers.
- Website: eupex.eu
- TADaaM funding: 150k€

Textarossa

Participants: Brice Goglin.

- Textarossa: Towards EXtreme scale Technologies and Accelerators for euROhpc hw/Sw Supercomputing Applications for exascale
- Program: H2020 EuroHPC
- Grant Agreement number: 956831 — TEXTAROSSA — H2020-JTI-EuroHPC-2019-1
- 2021-2024
- Partners: Fraunhofer Gesellschaft zur Foerderung der Angewandten Forshung E.V.; Consorzio Interuniversitario Nazionale per l'Informatica; Institut National de Recherche en Informatique et Automatique; Bull SAS; E4 Computer Engineering SPA; Barcelona Supercomputing Center; Instytut Chemii Bioorganicznej Polskiej; Istituto Nazionale di Fisica Nucleare; Consiglio Nazionale delle Ricerche; In Quattro SRL.

- To achieve high performance and high energy efficiency on near-future exascale computing systems, a technology gap needs to be bridged: increase efficiency of computation with extreme efficiency in HW and new arithmetics, as well as providing methods and tools for seamless integration of reconfigurable accelerators in heterogeneous HPC multi-node platforms. TEXTAROSSA aims at tackling this gap through applying a co-design approach to heterogeneous HPC solutions, supported by the integration and extension of IPs, programming models and tools derived from European research projects, led by TEXTAROSSA partners.
- Website: textarossa.eu
- TADaaM funding: 200k€

10.3.2 EuroHPC

MICROCARD-2

Participants: François Pellegrini.

MICROCARD-2 on EuroHPC-Ju

Title: MICROCARD-2: numerical modeling of cardiac electrophysiology at the cellular scale

Duration: from November 1, 2024 to April 30, 2027

Partners:

- Inria, France
- Karlsruher Institut Für Technologie, Germany
- Megware, Germany
- Simula Research Laboratory (Simula), Norway
- Technical University München (TUM), Germany
- Università degli Studi di Pavia, Italy
- Università di Trento (UTrento), Italy
- Université de Bordeaux, France
- Université de Strasbourg, France

Inria contact: Olivier AUMAGE (Storm)

Coordinator: Mark POTSE, Université de Bordeaux

Summary: The MICROCARD-2 project is coordinated by Université de Bordeaux and involves the Inria teams CARMEN, STORM, and TADAAM in Bordeaux and CAMUS in Strasbourg, among a total of ten partner institutions in France, Germany, Italy, and Norway. This Centre of Excellence for numerical modeling of cardiac electrophysiology at the cellular scale builds on the MICROCARD project (2021–2024) and has the same website.

The modelling of cardiac electrophysiology at the cellular scale requires thousands of model elements per cell, of which there are billions in a human heart. Even for small tissue samples such models require at least exascale supercomputers. In addition the production of meshes of the complex tissue structure is extremely challenging, even more so at this scale. MICROCARD-2 works, in concert, on every aspect of this problem: tailored numerical schemes, linear-system solvers, and preconditioners; dedicated compilers to produce efficient system code for different CPU and GPU architectures (including the EPI and other ARM architectures); mitigation of energy usage; mesh production and partitioning; simulation workflows; and benchmarking.

The contribution of TADAAM concerns parallel mesh partitioning. Through the funding of a two-year engineer position, the scientific aim is to allow PT-SCOTCH to partition, on very large HPC systems comprising up to 10,000+ nodes, the huge meshes produced within the MICROCARD-2 project. The engineer will be hired from January 1st, 2025.

10.3.3 Other european programs/initiatives

ANR-DFG H2M

Participants: Pierre Clouzet, Brice Goglin, Emmanuel Jeannot.

- Title: Heuristics for Heterogeneous Memory
- Website: h2m.gitlabpages.inria.fr
- AAPG ANR 2020, 2021 - 2024 (48 months)
- Coordinator: Christian Terboven (German coordinator) and Brice Goglin (French coordinator).
- Abstract: H2M is a ANR-DFG project between the TADaaM team and the HPC Group at RWTH Aachen University (Germany) and Université of Reims Chanmpagne Ardenne, from 2021 to 2024. The overall goal is to leverage HWLOC's knowledge of heterogeneous memory up to programming languages such as OpenMP to ease the allocations of data sets in the appropriate target memories.

10.4 National initiatives

InriaSoft: Scotch Consortium

Participants: François Pellegrini, Clément Barthélemy.

- Scotch Consortium
- Program: InriaSoft
- 2024–
- Website: gitlab.inria.fr/scotch/scotch
- Coordinator: François Pellegrini
- Abstract:

  The Scotch Consortium, supported by InriaSoft[3], has been created to bring together organizations interested in furthering the SCOTCH software currently developed within the TADAAM project. It will take care of the sustainability and development of the Scotch software environment, sharing the governance between its members. It will also allow every member to participate in the software roadmap, and to get adequate support. It will ensure SCOTCH stays permanently maintained, and available to the worldwide community under a free/libre software license.

  While the consortium has not officially been launched, Inria has started populating the Scotch consortium engineering team by agreeing to hire a full-time core software engineer. Clément Barthélemy was recruited and started working on September 1st, joining Marc Fuentes, the part-time environment software engineer.

[3] Inria has launched the InriaSoft program to help support open source software products authored by Inria and its partners when their usage has gone beyond the academic circles of their initial research context and when some key users are willing to become involved to support future developments.

Numpex PC2: Exa-Soft

Participants: Alexandre Denis.

- Exa-SofT: HPC softwares and tools
- Program: project PC2 in PEPR Numpex
- 2023-2029
- Partners: Université Paris-Saclay, Telecom SudParis, Bordeaux INP, ENSIIE, Université de Bordeaux, Université de Grenoble-Alpes, Université de Rennes 1, Université de Strabourg, Université de Toulouse, CEA, CNRS, Inria.
- Website: numpex.org/exasoft-hpc-software-and-tools
- Coordinator: Raymond NAMYST (Storm)
- Abstract:

  Though significant efforts have been devoted to the implementation and optimization of several crucial parts of a typical HPC software stack, most HPC experts agree that exascale supercomputers will raise new challenges, mostly because the trend in exascale compute-node hardware is toward heterogeneity and scalability: Compute nodes of future systems will have a combination of regular CPUs and accelerators (typically GPUs), along with a diversity of GPU architectures. Meeting the needs of complex parallel applications and the requirements of exascale architectures raises numerous challenges which are still left unaddressed. As a result, several parts of the software stack must evolve to better support these architectures. More importantly, the links between these parts must be strengthened to form a coherent, tightly integrated software suite. Our project aims at consolidating the exascale software ecosystem by providing a coherent, exascale- ready software stack featuring breakthrough research advances enabled by multidisciplinary collaborations between researchers. The main scientific challenges we intend to address are: productivity, performance portability, heterogeneity, scalability and resilience, performance and energy efficiency.

Numpex PC3: Exa-DoST

Participants: Emmanuel Jeannot, Luan Teylo, Francieli Zanon-Boito.

- Exa-DoST: Data-oriented Software and Tools for the Exascale
- Program: project PC3 in PEPR Numpex
- 2023-2029
- Partners: Université Paris-Saclay, Telecom SudParis, Bordeaux INP, ENSIIE, Université de Bordeaux, Université de Grenoble-Alpes, Université de Rennes 1, Université de Strabourg, Université de Toulouse, CEA, CNRS, Inria.
- Website: numpex.org/exadost-data-oriented-software-and-tools-for-the-exascale/
- Coordinator: Gabriel ANTONIU (KerData)
- Abstract:

  The advent of future Exascale supercomputers raises multiple data-related challenges. To enable applications to fully leverage the upcoming infrastructures, a major challenge concerns the scalability of techniques used for data storage, transfer, processing and analytics. Additional key challenges

emerge from the need to adequately exploit emerging technologies for storage and processing, leading to new, more complex storage hierarchies. Finally, it now becomes necessary to support more and more complex hybrid workflows involving at the same time simulation, analytics and learning, running at extreme scales across supercomputers interconnected to clouds and edge-based systems. The Exa-DoST project will address most of these challenges, organized in 3 areas: 1. Scalable storage and I/O; 2. Scalable in situ processing; 3. Scalable smart analytics. As part of the NumPEx program, Exa-DoST will address the major data challenges by proposing operational solutions co-designed and validated in French and European applications. This will allow filling the gap left by previous international projects to ensure that French and European needs are taken into account in the roadmaps for building the data-oriented Exascale software stack.

Inria Exploratory Action

Participants: Asia Auville, Emmanuelle Saillard, Mihail Popov.

- Title: Large Language Models for Detection and Correction of Errors
- Website: LLM4DiCE
- 2024 - 2027 (36 months)
- Coordinator: Emmanuelle Saillard and Mihail Popov
- Abstract: Large Language Models (LLMs) are a hot and rapidly evolving research topic. In particular, their recent successes in summarization, question-answering, and code generation with AI pair programming make them attractive candidates in the field of error verification. We propose to harness these LLMs capabilities with fine-tuning on carefully generated datasets through a novel clustering strategy based on Natural Language Processing (NLP) techniques and code embedding to assist bug detection and correction, targeting hard domains such as parallel program verification.

11 Dissemination

11.1 Promoting scientific activities

11.1.1 Scientific events: organisation

Member of the organizing committees

- Emmanuel Jeannot is member the ICPP steering committee.
- Mihail Popov is co-organizer of the HPC Bugs Fest during Supercomputing.

11.1.2 Scientific events: selection

Member of the conference program committees

- Brice Goglin was a member of the following program committees: Supercomputing, Cluster and HiPC 2024.
- Emmanuel Jeannot was a member of the following program committees: PPAM 2024, HiPC 2024, CCGRID 2024 and DynResHPC workshop (in conjunction with EuroPar).
- Mihail Popov was a member of the following program committees: ICPP, IPDPSW GrAPL.
- Alexandre Denis was a member of the APDCM program committee.

- Francieli Zanon-Boito was a member of the following program committees: test of time award for Supercomputing 2024, Birds-of-a-Feather for Supercomputing 2024, PMBS workshop (in conjunction with Supercomputing) 2024, Bench 2024, HPCAsia 2024 and 2025, ComPas 2024.
- Luan Teylo Gouveia Lima was a member of the following program committees: PMBS workshop (in conjunction with Supercomputing) 2024.

Reviewer

- Emmanuel Jeannot was a reviewer for Cluster 2024.
- Mihail Popov was a reviewer for Supercomputing.
- Alexandre Denis was a reviewer for EuroMPI 2024, Euro-Par 2024, and SC 2024.
- Francieli Zanon-Boito was a reviewer for IPDPS 2025.

11.1.3 Journal

Member of the editorial boards

- Emmanuel Jeannot is associate editor of IEEE Transaction on Parallel and Distributed Systems (TPDS)
- Emmanuel Jeannot is member of the editorial board of the Journal of Parallel Emergent & Distributed Systems.

Reviewer - reviewing activities

- Emmanuel Jeannot was a reviewer for FGCS, Parallel Computing, Computers and Electrical Engineering.
- Luan Teylo was a reviewer for FGCS and Parallel Computing.

11.1.4 Invited talks

- Luan Teylo presented the IOPS tool in a keynote at the CHPH National Conference in South Africa.
- Francieli Zanon-Boito gave an invited talk during the Per3S workshop on storage - “Research on HPC I/O in the context of the PEPR NumPEx project”, Paris, France, May 2024.

11.1.5 Leadership within the scientific community

- Emmanuel Jeannot is responsible of the international cooperation within the NumPex project.
- Emmanuel Jeannot was responsible for the Bordeaux site of Slices-FR.
- Emmanuel Jeannot is vice-head, for Inria, of the scientific board of the Joint Lab of Exascale Computing (JLESC).
- François Pellegrini is a co-pilot, with Roberto Di Cosmo, of the College on Source Codes and Software of the Committee for Open Science of the French Ministry of Higher Education and Research. In this context, he co-authored several publications related to open science in informatics [25, 2, 24, 26, 23].
- Francieli Zanon-Boito is co-responsible for the WP on Parallel I/O (WP1) from PC3 (Exa-DoST) within the NumPEx project.
- Francieli Zanon-Boito is the French PI of the DECoHPC Inria Associate Team.

11.1.6 Scientific expertise

- Brice Goglin was a member of the Khronos OpenCL Advisory Panel as well as the Unified Acceleration Foundation (former oneAPI) Hardware Abstraction SIG.
- Brice Goglin is involved in the expertise of HPC projects in Africa with IRD and AFD.
- François Pellegrini was a member of the ERC ethics screening or assessment panels for ERC calls SyG-2023, CoG-2023, AdG-2023, StG-2024, SyG-2024 and CoG-2024.

11.1.7 Research administration

- Brice Goglin is a member of the executive committee of the Abaca project for Inria's nation-wide computing infrastructure.
- Brice Goglin is in charge of the computing infrastructures of the Inria Bordeaux research center.
- Emmanuel Jeannot was head of science of the Inria Bordeaux research center until June 2024.
- Emmanuel Jeannot was a member of the Inria evaluation committee until June 2024.
- Francieli Zanon-Boito is a member of the council of the SIN department of the University of Bordeaux since 2022.

11.1.8 Standardization Activities

Participation in the MPI Forum

- TADAAM attended the MPI Forum meetings on behalf of Inria (where the MPI standard for communication in parallel applications is developed and maintained). Guillaume Mercier leads the *Topologies* working group. He participates in serveral other Working Groups (Hybrid WG, ABI WG) and is also an editor of the MPI Standard, as a member of several chapter committees (Contexts, Topologies and Info object). He also serves as the Context chapter committee chair. This year, some minor updates (mostly corrections and errata) have been added to the mechanisms we successfully introduced in prior years. The major addition to the MPI standard is an Abstract Binary Interface (ABI) that should improve applications portability since several implementations of MPI do coexist. This ABI support will enable applications to switch (more or less) effortlessly from one implementation to another (or even from one implementation *version* to another). This feature was deemed important enough to promote the next MPI version number from 4.2 (i.e a revision) to 5.0 (a major revision). The ratification and publication process has been engaged for a while and was pursued in 2024. Final ratification is expected in 2025.

Participation in the PMIx ASC

- TADAAM is a member of the Administrative Steering Committee of PMIx standard focused on orchestration of application launch and execution.

11.2 Teaching - Supervision - Juries

11.2.1 Teaching

Members of the TADAAM project gave hundreds of hours of teaching at Université de Bordeaux and the Bordeaux INP engineering school, covering a wide range of topics from basic use of computers, introduction to algorithmic and C programming to advanced topics such as probabilities and statistics, scheduling, computer networks, computer architecture, operating systems, big data, cryptography, parallel programming and high-performance runtime systems, as well as software law and personal data law.

- François Pellegrini did the final conference of the *Numerics* graduate program at Université de Bordeaux, on the ethical issues of automated data processing.

- François Pellegrini did a course in English on "*Software Law*" and "*Personal data law*" to 12 PhD students of Université de Bordeaux.
- François Pellegrini did a presentation on "personal data protection and data security" at the annual regional seminar on cyber-security for high school and technical college informatics teachers, organized by the Rectorat de Bordeaux.
- Luan Teylo taught a course on data visualization with Python to undergraduate students from various fields of study at the Université de Bordeaux.

11.2.2 Supervision

- PhD finished: Clément Gavoille, A performance projection approach for design-space exploration on Arm HPC environment. [17] Started in Started in January 2021 and defended in January 2024, co-advised with CEA and ARM. Inria Advisors: Brice Goglin and Emmanuel Jeannot.
- PhD finished: Julien Rodriguez, Circuit mapping onto multi-FPGA platforms. [18] Started in October 2020 and defended in September 2024. Advisors: François Pellegrini, François Galea and Lilia Zaourar.
- PhD finished: Alexis Bandet, I/O characterization and monitoring of the new generation of HPC applications. [16] Started in October 2021 and defended in December 2024. Advisors: Francieli Zanon-Boito and Guillaume Pallez.
- PhD finished: Richard Sartori, Determination of optimal parameters for MPI applications deployment on parallel architectures. [19] Started in April 2021 and defended in December 2024, co-advised with ATOS/Bull/Eviden in Grenoble. Inria Advisors: Guillaume Mercier and Emmanuel Jeannot.
- PhD in progress: Thibaut Pepin, MPI communication on modular supercomputing architectures, started in May 2023. Advisors: Guillaume Mercier.
- PhD in progress: Charles Goedefroit, Delivering userspace interrupts from the BXI network interface. co-advised with ATOS/Bull/Eviden. Started in March 2024. Advisors: Alexandre Denis and Brice Goglin.
- PhD in progress: Méline Trochon, Adaptive checkpointing strategies depending on the network load. Started in November 2024. Advisors: Francieli Zanon-Boito, François Tessier, Brice Goglin and Jean-Thomas Acquaviva (DDN).
- PhD in progress: Lana Scravaglieri, Portable vectorization with numerical accuracy control for multi-precision simulation codes. Advisors: Olivier Aumage, Mihail Popov, Thomas Guignon (IFPEN) and Ani Anciaux-Sedrakian (IFPEN).
- PhD in progress: Asia Auville, Large Language Models for Detection and Correction of Errors in HPC Applications. Advisors: Emmanuelle Saillard, Mihail Popov, Pablo Oliveira (UVSQ) and Eric Petit (Intel).

11.2.3 Juries

- Emmanuel Jeannot was the president of the thesis committee of Maël MADON, from Université de Toulouse (Paul Sabatier).
- François Pellegrini was a member of the habilitation committee of Mark POTSE, from Université de Bordeaux.
- François Pellegrini was the president of the thesis committee of Nouha LAAMECH, from Université de Pau et des Pays de l'Adour.
- Francieli Zanon-Boito was a member of the Ph.D. jury of Julien MONNIOT, from Université de Rennes.

11.3 Popularization

11.3.1 Participation in Live events

- Brice Goglin gave talks about research in computer science and high-performance computing to high-school students as part of the *Chiche* programme and to ENS Lyon students.
- Emmanuel Jeannot participated to the panel: *Reflections on the Impact of Generative AI on Our Society and Professions* at Inria.

12 Scientific production

12.1 Major publications

[1] L. Scravaglieri, M. Popov, L. Lima Pilla, A. Guermouche, O. Aumage and E. Saillard. 'Optimizing Performance and Energy Across Problem Sizes Through a Search Space Exploration and Machine Learning'. In: *Journal of Parallel and Distributed Computing* 180 (28th June 2023), p. 104720. DOI: 10.1016/j.jpdc.2023.104720. URL: https://hal.science/hal-03810305 (cit. on p. 21).

12.2 Publications of the year

International journals

[2] F. Pellegrini. 'Qu'est-ce qu'un logiciel de recherche ?' In: *Revue Lamy Droit de l'immatériel* 210 (Jan. 2024), pp. 39–40. URL: https://inria.hal.science/hal-04476872 (cit. on p. 31).

[3] F. Tessier, V. Vishwanath and E. Jeannot. 'Adding topology and memory awareness in data aggregation algorithms'. In: *Future Generation Computer Systems* 159 (Oct. 2024), pp. 188–203. DOI: 10.1016/j.future.2024.05.016. URL: https://hal.science/hal-04783379 (cit. on p. 16).

International peer-reviewed conferences

[4] A. Bandet, F. Boito and G. Pallez. 'Scheduling distributed I/O resources in HPC systems'. In: 30th International European Conference on Parallel and Distributed Computing 26 - 30 August 2024 Madrid, Spain 30th International European Conference on Parallel and Distributed Computing. Madrid, Spain, 26th Aug. 2024. URL: https://inria.hal.science/hal-04394004 (cit. on pp. 14, 17).

[5] A. Denis and C. Goedefroit. 'NBLFQ: a lock-free MPMC queue optimized for low contention'. In: IPDPS 2025 - 39th International Parallel & Distributed Processing Symposium. International Parallel & Distributed Processing Symposium. Milan, Italy: IEEE, June 2025. URL: https://inria.hal.science/hal-04851700 (cit. on p. 15).

[6] C. Gavoille, H. Taboada, J. Domke, B. Goglin and E. Jeannot. 'Performance Projection for Design-Space Exploration on future HPC Architectures'. In: *Proceedings of the 39th IEEE International Parallel and Distributed Processing Symposium (IPDPS)*. 39th IEEE International Parallel & Distributed Processing Symposium (IPDPS). Milano, Italy: IEEE, June 2025. URL: https://inria.hal.science/hal-04856139 (cit. on pp. 15, 22).

[7] V. Gilbert, J. Rodriguez and S. Louise. 'Benchmarking Quantum Annealers with Near-Optimal Minor-Embedded Instances'. In: 2024 International Conference on Quantum Computing and Engineering. Vol. 1. Montréal, Canada, 2024, pp. 531–537. DOI: 10.1109/QCE60285.2024.00068. URL: https://hal.science/hal-04570920 (cit. on p. 20).

[8] C. Goedefroit. 'Interruptions en espace utilisateur délivrées par la carte réseau BXI'. In: Compas 2024 - Conférence francophone d'informatique en Parallélisme, Architecture et Système. Nantes, France, 2nd July 2024. URL: https://inria.hal.science/hal-04693786 (cit. on p. 15).

[9] E. Jeannot, P. Lemarinier, G. Mercier, S. Robert-Hayek and R. Sartori. 'Application-Agnostic Auto-Tuning of Open MPI Collectives Using Bayesian Optimization'. In: IPDPSW 2024 - 38th IEEE International Parallel and Distributed Processing Symposium Workshops. San Francisco, United States: IEEE, 27th May 2024, pp. 771–781. DOI: 10.1109/IPDPSW63119.2024.00141. URL: https://inria.hal.science/hal-04873139 (cit. on p. 16).

[10] J. E. Karchi, H. Chen, A. Tehranijamsaz, A. Jannesari, M. Popov and E. Saillard. 'MPI Errors Detection using GNN Embedding and Vector Embedding over LLVM IR'. In: IPDPS 2024 - 38th International Symposium on Parallel and Distributed Processing. San francisco, United States, 27th May 2024. URL: https://inria.hal.science/hal-04724011 (cit. on p. 20).

[11] J. Klinkenberg, C. Foyer, P. Clouzet, B. Goglin, E. Jeannot, C. Terboven and A. Kozhokanova. 'Phase-based Data Placement Optimization in Heterogeneous Memory'. In: CLUSTER 2024 - International Conference on Cluster Computing. Kobe, Japan, Japan: IEEE, 24th Sept. 2024. URL: https://inria.hal.science/hal-04711658 (cit. on p. 14).

[12] M. D. Lima, L. Teylo and L. Drummond. 'An Analysis of Performance Variability in AWS Virtual Machines'. In: SSCAD 2024 - XXV Simpósio em Sistemas Computacionais de Alto Desempenho. São Carlos, Brazil, 23rd Oct. 2024, pp. 312–323. DOI: 10.5753/sscad.2024.244526. URL: https://hal.science/hal-04839930 (cit. on p. 21).

[13] A. Nunes, D. Sodré, C. Boeres, J. Viterbo, L. Drummond, V. Rebello, L. Teylo, F. Portella, P. Estrela and R. Malini. 'A Framework for Executing Long Simulation Jobs Cheaply in the Cloud'. In: IC2E 2024 - IEEE International Conference on Cloud Engineering. Paphos, Cyprus: IEEE, 12th Sept. 2024, pp. 233–244. DOI: 10.1109/IC2E61754.2024.00033. URL: https://hal.science/hal-04839966 (cit. on p. 21).

[14] J. Rodriguez, F. Galea, F. Pellegrini and L. Zaourar. 'Hypergraph Clustering with Path-Length Awareness'. In: *Computational Science – ICCS 202424th International Conference, Malaga, Spain, July 2–4, 2024, Proceedings, Part V*. ICCS 2024 - 24th International Conference on Computational Science. Vol. 14836. Lecture Notes in Computer Science. Malaga, Spain, 28th June 2024, pp. 90–104. DOI: 10.1007/978-3-031-63775-9_7. URL: https://hal.science/hal-04706759 (cit. on p. 19).

[15] A. Tarraf, A. Bandet, F. Zanon Boito, G. Pallez and F. Wolf. 'Capturing Periodic I/O Using Frequency Techniques'. In: IPDPS 2024 - 38th IEEE International Parallel & Distributed Processing Symposium. San Francisco, United States, 2024, pp. 1–13. URL: https://inria.hal.science/hal-04382142 (cit. on p. 18).

Doctoral dissertations and habilitation theses

[16] A. Bandet. 'Characterization and monitoring of I/O for HPC workloads'. Université de Bordeaux, 11th Dec. 2024. URL: https://theses.hal.science/tel-04933980 (cit. on p. 33).

[17] C. Gavoille. 'A performance projection approach for design-space exploration on Arm HPC environment'. Université de Bordeaux, 15th Jan. 2024. URL: https://theses.hal.science/tel-04468260 (cit. on p. 33).

[18] J. Rodriguez. 'Circuit partitioning for multi-FPGA platforms'. Université de Bordeaux, 6th Sept. 2024. URL: https://theses.hal.science/tel-04731886 (cit. on pp. 19, 33).

[19] R. Sartori. 'Optimal parameters determination for the execution of MPI applications on parallel architectures'. Université de Bordeaux, 19th Dec. 2024. URL: https://theses.hal.science/tel-04907812 (cit. on p. 33).

Reports & preprints

[20] A. Bandet, F. Boito and G. Pallez. *Prediction and Interpretability of HPC I/O Resources Usage with Machine Learning*. 2024. URL: https://inria.hal.science/hal-04698511 (cit. on p. 17).

[21] A. Bandet, F. Zanon Boito and G. Pallez. *Allocation and Placement Algorithms for Scheduling Distributed I/O Resources in HPC Systems*. RR-9549. Inria Bordeaux; Inria Rennes, 28th May 2024, pp. 1–27. URL: https://hal.science/hal-04593977 (cit. on p. 17).

[22] C. Barthélemy, F. Zanon Boito, E. Jeannot, G. Pallez and L. Teylo. *Implementation of an unbalanced I/O Bandwidth Management system in a Parallel File System*. RR-9537. Inria, Jan. 2024. URL: https://inria.hal.science/hal-04417412 (cit. on p. 18).

[23] I. Blanc, R. Di Cosmo, M. Giraud, D. Le Berre, V. Louvet, N. P. Rougier, S. Granger and F. Pellegrini. *Highlights of the "Software Pillar of Open Science" workshop*. Comité pour la Science Ouverte, June 2024. DOI: 10.52949/53. URL: https://hal-lara.archives-ouvertes.fr/hal-04600258 (cit. on p. 31).

[24] R. Di Cosmo, F. Pellegrini, M. Giraud, M. Gruenpeter, D. Le Berre, V. Louvet and S. Granger. *"Software and source codes" College: Minutes from the work session, November 30th 2023*. Comité pour la Science Ouverte, 2024. DOI: 10.52949/58. URL: https://hal-lara.archives-ouvertes.fr/hal-04683957 (cit. on p. 31).

[25] D. Le Berre, J.-Y. Jeannas, R. Di Cosmo and F. Pellegrini. *Higher Education and Research Forges in France - Definition, uses, limitations encountered and needs analysis*. Comité pour la science ouverte, 29th Sept. 2024. DOI: 10.52949/37. URL: https://hal-lara.archives-ouvertes.fr/hal-04208924 (cit. on p. 31).

[26] V. Louvet, S. Granger, R. Di Cosmo and F. Pellegrini. *Vers un catalogue des logiciels issus de la recherche: Etat des lieux et analyse des besoins*. Comité pour la science ouverte, July 2024. DOI: 10.52949/79. URL: https://hal.science/hal-04779846 (cit. on p. 31).

[27] F. Zanon Boito, L. Teylo, M. Popov, T. Jolivel, F. Tessier, J. Luettgau, J. Monniot, A. Tarraf, A. Carneiro and C. Osthoff. *A Deep Look Into the Temporal I/O Behavior of HPC Applications*. 2025. URL: https://inria.hal.science/hal-04887809 (cit. on p. 19).

12.3 Cited publications

[28] F. Boito, G. Pallez, L. Teylo and N. Vidal. 'IO-SETS: Simple and efficient approaches for I/O bandwidth management'. In: *IEEE Transactions on Parallel and Distributed Systems* 34.10 (Aug. 2023), pp. 2783–2796. DOI: 10.1109/TPDS.2023.3305028. URL: https://inria.hal.science/hal-03648225 (cit. on pp. 7, 18).

[29] A. Lorén. *Hybrid E/P Cores Prefetch Optimization*. 2024. URL: https://www.diva-portal.org/smash/get/diva2:1888603/FULLTEXT01.pdf (visited on 30/12/2024) (cit. on p. 21).

[30] S. Robert, S. Zertal and G. Goret. 'SHAMan: an intelligent framework for HPC auto-tuning of I/O accelerators'. In: *Proceedings of the 13th International Conference on Intelligent Systems: Theories and Applications*. SITA'20. Rabat, Morocco: Association for Computing Machinery, 2020. DOI: 10.1145/3419604.3419775. URL: https://doi.org/10.1145/3419604.3419775 (cit. on p. 16).