UMR IRISA

Activity Report 2018

Team KERDATA

Scalable Storage for Clouds and Beyond

Joint team with Inria Rennes – Bretagne Atlantique

D1 – Large Scale Systems

CentraleSupélec cnrs rennes IMT Atlantique Bretagne-Pays de la Loire École Mines-Télécom Inria INSA RENNES Université Bretagne Sud ubs: UNIVERSITÉ DE RENNES 1

Table of contents

Project-Team KERDATA

Creation of the Team: 2009 July 01, updated into Project-Team: 2012 July 01

Keywords:

Computer Science and Digital Science:

A1.1.4. - High performance computing
A1.1.5. - Exascale
A1.1.9. - Fault tolerant systems
A1.3. - Distributed Systems
A1.3.5. - Cloud
A1.3.6. - Fog, Edge
A1.6. - Green Computing
A3.1.2. - Data management, quering and storage
A3.1.3. - Distributed data
A3.1.8. - Big data (production, storage, transfer)
A6.2.7. - High performance computing
A6.3. - Computation-data interaction
A7.1. - Algorithms
A7.1.1. - Distributed algorithms

Other Research Topics and Application Domains:

B3.2. - Climate and meteorology
B3.3.1. - Earth and subsoil
B8.2. - Connected city
B9.5.6. - Data science

1. Team, Visitors, External Collaborators

Research Scientist

Gabriel Antoniu [Team leader, Inria, Senior Researcher, HDR]

Faculty Members

Luc Bougé [École normale supérieure de Rennes, Professor, HDR]
Alexandru Costan [INSA Rennes, Associate Professor]

Post-Doctoral Fellow

Pedro de Souza Bento Da Silva [INSA Rennes]

PhD Students

Nathanaël Cheriere [École normale supérieure de Rennes]
Paul Le Noac'h [INSA Rennes]
Ovidiu-Cristian Marcu [Inria, until Sep 2018]
Pierre Matri [Universidad Politécnica de Madrid, until Feb 2018]
Mohammed-Yacine Taleb [Inria, until Oct 2018]

Technical staff

Ovidiu-Cristian Marcu [Inria, from Oct 2018]
Hadi Salimi [Inria]

Intern

Laurent Prosperi [Inria, from Apr 2018 until Jul 2018]

Administrative Assistants

Aurélie Patier [Univ de Rennes I, until Feb 2018]
Gaelle Tworkowski [Inria, from Mar 2018]

Visiting Scientist

Pierre Matri [Inria, from Mar 2018 until May 2018]

2. Overall Objectives

2.1. Context: the need for scalable data management

We are witnessing a rapidly increasing number of application areas generating and processing very large volumes of data on a regular basis. Such applications are called *data-intensive*. Governmental and commercial statistics, climate modeling, cosmology, genetics, bio-informatics, high-energy physics are just a few examples in the scientific area. In addition, rapidly growing amounts of data from social networks and commercial applications are now routinely processed.

In all these examples, the overall application performance is highly dependent on the properties of the underlying data management service. It becomes crucial to store and manipulate massive data efficiently. However, these data are typically *shared* at a large scale and *concurrently accessed* at a high degree. With the emergence of recent infrastructures such as cloud computing platforms and post-Petascale high-performance computing (HPC) systems, achieving highly scalable data management under such conditions has become a major challenge.

2.1.1. Our objective

The KerData project-team is namely focusing on designing innovative architectures and systems for *scalable data storage and processing*. We target two types of infrastructures: *clouds* and *post-Petascale high-performance supercomputers*, according to the current needs and requirements of data-intensive applications.

We are especially concerned by the applications of major international and industrial players in cloud computing and extreme-scale high-performance computing (HPC), which shape the long-term agenda of the cloud computing [37], [34] and Exascale HPC [36] research communities. The Big Data area, which has recently captured a lot of attention, emphasized the challenges related to Volume, Velocity and Variety. This is yet another element of context that further highlights the primary importance of designing data management systems that are efficient at a very large scale.

2.1.1.1. Alignment with Inria's scientific strategy

Data-intensive applications exhibit several common requirements with respect to the need for data storage and I/O processing. We focus on some core challenges related to data management, resulted from these requirements. Our choice is perfectly in line with Inria's strategic plan [41], which acknowledges as critical the challenges of *storing, exchanging, organizing, utilizing, handling and analyzing* the huge volumes of data generated by an increasing number of sources. This topic is also stated as a scientific priority of Inria's research centre of Rennes [40]: *Storage and utilization of distributed big data*.

2.1.1.2. Challenges and goals related to cloud data storage and processing

In the area of cloud data processing, a significant milestone is the emergence of the Map-Reduce [47] parallel programming paradigm. It is currently used on most cloud platforms, following the trend set up by Amazon [31]. At the core of Map-Reduce frameworks lies the storage system, a key component which must meet a series of specific requirements that are not fully met yet by existing solutions: the ability to provide efficient *fine-grain access* to the files, while sustaining a *high throughput* in spite of *heavy access concurrency*; the need to provide a high resilience to *failures*; the need to take *energy-efficiency* issues into account.

More recently, it becomes clear that data-intensive processing needs to go beyond the frontiers of single datacenters. In this perspective, extra challenges arise, related to the efficiency of metadata management. This efficiency has a major impact on the access to very large sets of small objects by Big Data processing workflows running on large-scale infrastructures.

2.1.1.3. Challenges and goals related to data-intensive HPC applications

Key research fields such as climate modeling, solid Earth sciences or astrophysics rely on very large-scale simulations running on post-Petascale supercomputers. Such applications exhibit requirements clearly identified by international panels of experts like IESP [39], EESI [35], ETP4HPC [36]. A jump of one order of magnitude in the size of numerical simulations is required to address some of the fundamental questions in several communities in this context. In particular, the lack of data-intensive infrastructures and methodologies to analyze the huge results of such simulations is a major limiting factor.

The challenge we have been addressing is to find new ways to store, visualize and analyze massive outputs of data during and after the simulations. Our main initial goal was to do it without impacting the overall performance, avoiding the *jitter* generated by I/O interference as much as possible. Recently, we started to focus specifically on *in situ processing* approaches and we explored approaches to *model and predict I/O phase occurrences* and to *reduce intra-application and cross-application I/O interference*.

2.1.2. Our approach

KerData's global approach consists in studying, designing, implementing and evaluating distributed algorithms and software architectures for scalable data storage and I/O management for efficient, large-scale data processing. We target two main execution infrastructures: cloud platforms and post-Petascale HPC supercomputers.

2.1.2.1. Platforms and Methodology

The highly experimental nature of our research validation methodology should be emphasized. To validate our proposed algorithms and architectures, we build software prototypes, then validate them at a large scale on real testbeds and experimental platforms.

We strongly rely on the Grid'5000 platform. Moreover, thanks to our projects and partnerships, we have access to reference software and physical infrastructures. In the cloud area, we use the Microsoft Azure and Amazon cloud platforms. In the post-Petascale HPC area, we are running our experiments on systems including some top-ranked supercomputers, such as Titan, Jaguar, Kraken or Blue Waters. This provides us with excellent opportunities to validate our results on advanced realistic platforms.

2.1.2.2. Collaboration strategy

Our collaboration portfolio includes international teams that are active in the areas of data management for clouds and HPC systems, both in Academia and Industry.

Our academic collaborating partners include Argonne National Lab, University of Illinois at Urbana-Champaign, Universidad PolitÃ©cnica de Madrid, Barcelona Supercomputing Center, University Politehnica of Bucharest. In industry, we are currently collaborating with Huawei and Total.

Moreover, the consortiums of our collaborative projects include application partners in the area of climate simulations (e.g., the Department of Earth and Atmospheric Sciences of the University of Michigan, within our collaboration inside JLESC [42]). This is an additional asset, which enables us to take into account application requirements in the early design phase of our solutions, and to validate those solutions with real applications... and real users!

3. Research Program

3.1. Research axis 1: Convergence of Extreme-Scale Computing and Big Data Infrastructures

The tools and cultures of High Performance Computing and Big Data Analytics have evolved in divergent ways. This is to the detriment of both. However, big computations still generate and are needed to analyze Big Data. As scientific research increasingly depends on both high-speed computing and data analytics, the potential interoperability and scaling convergence of these two eco-systems is crucial to the future. Our objective for the next years is premised on the idea that we must begin to systematically map out and account for the ways in which the major issues associated with Big Data intersect with, impinge upon, and potentially change the plans that are now being laid for achieving Exascale computing.

Collaboration. *This axis is addressed in close collaboration with María Pérez (UPM), Rob Ross (ANL), Toni Cortes (BSC), Bogdan Nicolae (formerly at IBM Research, now at Huawei Research).*

Relevant groups with similar interests are the following ones.

- *The group of Jack Dongarra, Innovative Computing Laboratory at University of Tennessee/Oak Ridge National Laboratory, working on joint tools Exascale Computing and Big Data.*
- *The group of Satoshi Matsuoka, Tokyo Institute of Technology, working on system software for Clouds and HPC.*
- *The group of Franck Cappello at Argonne National Laboratory/NCSA working on on-demand data analytics and storage for extreme-scale simulations and experiments.*

3.1.1. High-performance storage for concurrent Big Data applications

We argue that storage is a plausible pathway to convergence. In this context, we plan to focus on the needs of concurrent Big Data applications that require high-performance storage, as well as transaction support. Although blobs (binary large objects) are an increasingly popular storage model for such applications, state-of-the-art blob storage systems offer no transaction semantics. This demands users to coordinate data access carefully in order to avoid race conditions, inconsistent writes, overwrites and other problems that cause erratic behavior.

We argue there is a gap between existing storage solutions and application requirements, which limits the design of transaction-oriented applications. In this context, one idea on which we plan to focus our efforts is exploring how blob storage systems could provide built-in, multi-blob transactions, while retaining sequential consistency and high throughput under heavy access concurrency.

The early principles of this research direction have already raised interest from our partners at ANL (Rob Ross) and UPM (María Pérez) for potential collaborations. In this direction, the acceptance of our paper on the Týr transactional blob storage system as a Best Student Paper Award Finalist at the SC16 conference [10] is a very encouraging step.

3.1.2. Big Data analytics on Exascale HPC machines

Big Data analytics is another interesting direction that we plan to explore, building on top of these converged storage architectures. Specifically, we will examine the ways in which Exascale infrastructures can be leveraged not only by HPC-centric, but also by scientific, cloud-centric applications. Many of the current state-of-the-art Big Data processing approaches, including Hadoop and Spark [43] are optimized to run on commodity machines. This impacts the mechanisms used to deal with failures and the limited network bandwidth.

A blind adoption of these systems on extreme-scale platforms would result in high overheads. It would therefore prevent users from fully benefiting from the high performance infrastructure. The objective that we set here is to explore design and implementation options for new data analytics systems that can exploit the features of extreme-scale HPC machines: multi-core nodes, multiple memory and storage technologies including a large memory, NVRAM, SSDs, etc.

3.2. Research axis 2: Advanced stream data processing

The recent evolutions in the area of Big Data processing have pointed out some limitations of the initial Map-Reduce model. It is well suited for batch data processing, but less suited for real-time processing of dynamic data streams. New types of data-intensive applications emerge, e.g., for enterprises who need to perform analysis on their stream data in ways that can give fast results (i.e., in real time) at scale (e.g., click-stream analysis and network-monitoring log analysis). Similarly, scientists require fast and accurate data processing techniques in order to analyze their experimental data correctly at scale (e.g., collectively analysis of large data sets distributed in multiple geographically distributed locations).

Our plan is to revisit current data management techniques to cope with the volatile requirements of data-intensive applications on large-scale dynamic clouds in a cost-efficient way.

Collaboration. *This axis is addressed in close collaboration with María Pérez (UPM), Kate Keahey (ANL) and Toni Cortes (BSC).*

Relevant groups with similar interests include the following ones.

- *The AMPLab, UC Berkeley, USA, working on scheduling stream data applications in heterogeneous clouds.*
- *The group of Ewa Deelman, USC Information Sciences Institute, working on resource management for workflows in Clouds.*
- *The XTRA group, Nanyang Technological University, Singapore, working on resource provisioning for workflows in the cloud.*

3.2.1. Stream-oriented, Big Data processing on clouds

The state-of-the-art Hadoop Map-Reduce framework cannot deal with stream data applications, as it requires the data to be initially stored in a distributed file system in order to process them. To better cope with the above-mentioned requirements, several systems have been introduced for stream data processing such as Flink [38], Spark [43], Storm [44], and Google MillWheel [46]. These systems keep computation in memory to decrease latency, and preserve scalability by using data-partitioning or dividing the streams into a set of deterministic batch computations.

However, they are designed to work in dedicated environments and they do not consider the performance variability (i.e., network, I/O, etc.) caused by resource contention in the cloud. This variability may in turn cause high and unpredictable latency when output streams are transmitted to further analysis. Moreover, they overlook the dynamic nature of data streams and the volatility in their computation requirements. Finally, they still address failures in a best-effort manner.

Our objective is to investigate new approaches for reliable, stream Big Data processing on clouds. We will explore new mechanisms that expose resource heterogeneity (observed variability in resource utilization at runtime) when scheduling stream data applications. We will also investigate how to adapt to node failures automatically, and to adapt the failure handling techniques to the characteristics of the running application and to the root cause of failures.

3.2.2. Efficient Edge, Cloud and hybrid Edge/Cloud data processing

Today, we are approaching an important technological milestone: applications are generating huge amounts of data and are demanding low-latency responses to their requests. Mobile computing and Internet of Things (IoT) applications are good illustrations of such scenarios. Using only Cloud computing for such scenarios is challenging. Firstly, Cloud resources are most of the time accessed through Internet, hence, data are sent across high-latency wide area networks, which may degrade the performance of applications. Secondly, it may be impossible to send data to the Cloud due to data regulations, national security laws or simply because an Internet connection is not available. Finally, data transmission costs (e.g., Cloud provider fees, carrier costs) could make a business solution impractical.

Edge computing is a new paradigm which aims to address some of these issues. The key idea is to leverage computing and storage resources at the “edge” of the network, i.e., on processing units located close to the data sources. This allows applications to outsource task execution from the main (Cloud) processing data centers to the edge. The development of Edge computing was accelerated by the recent emergence of stream processing, a new model for handling continuous flows of data in real-time, as opposed to batch processing, which typically processes bounded datasets offline.

However, Edge computing is not a silver bullet. Besides being a new concept not fully established in the community, issues like node volatility, limited processing power, high latency between nodes, fault tolerance and data degradation may impact applications depending on the characteristics of the infrastructure. An important question raised in this context is “how close is physically close enough?”. In other words, how much can one improve (or degrade) the performance of an application by performing computation closer to the data sources?

Our objective is to try to answer precisely this question. We are interested in understanding the conditions that enable the usage of Edge or Cloud computing to reduce the time to results and the associated costs. While some state-of- the-art approaches advocate either "100% Cloud” or “100% Edge” solutions, the relative efficiency of a method over the other may vary. Intuitively, it depends on many parameters, including network technology, hardware characteristics, volume of data or computing power, processing framework configuration and application requirements, to cite a few. We plan to study their impact on the overall application performance.

3.3. Research axis 3: I/O management, in situ visualization and analysis on HPC systems at extreme scales

Over the past few years, the increasing amounts of data produced by large-scale simulations have motivated a shift from traditional offline data analysis to in situ analysis and visualization. In situ processing started by coupling a parallel simulation with an analysis or visualization library, to avoid the cost of writing data on storage and reading it back. Going beyond this simple pairwise tight coupling, complex analysis workflows today are graphs with one or more data sources and several interconnected analysis components.

Collaboration. *This axis is worked out in close collaboration with Rob Ross (ANL), Tom Peterka (ANL), Matthieu Dorier (ANL), Toni Cortes (BSC), Bruno Raffin (Inria). Some additional collaborations are in discussion with other members of JLESC, and with CEA and Total.*

Relevant groups with similar interests include the following ones.

- *The group of Manish Parashar at Rutgers University, USA (I/O management for HPC systems, in situ processing).*
- *The group of Scott Klasky at Oak Ridge National Lab, USA (I/O management for HPC systems, in situ processing).*
- *The CNRS IPSL laboratory (Sébastien Denvil, Pôle de modélisation du climat) in Paris, France (in situ data analytics).*

3.3.1. Toward a joint optimized architecture for in situ visualization and advanced processing

From Inria and ANL, four tools at least have emerged to address some challenges of coupling simulations with visualization packages or analysis workflows. Each of them focused on some particular aspect:

Damaris (Inria, [5], [4]) exploits dedicated cores to enable jitter-free I/O and in situ visualization;

Decaf (ANL, [33]) implements a coupling service for workflows;

FlowVR (Inria, [45]) connects workflow components for in situ processing;

Swift (ANL, [48]) focuses on implicitly parallel data flows and was optimized for Big Data processing.

Our plan is to explore how these tools could best leverage their respective strengths in a *joint optimized architecture for in situ visualization and advanced processing* in the HPC area. We published a preliminary study describing the lessons learned from using these tools in production environments with real applications [7]. Such a joint architecture will contribute to address the data volume and velocity challenges raised by data-intensive workflows, including complex data-intensive analytics phases. It may also impact, in a subsequent step, future data analysis pipelines for converged Big Data and HPC architectures.

4. New Software and Platforms

4.1. Damaris

KEYWORDS: Visualization - I/O - HPC - Exascale - High performance computing
SCIENTIFIC DESCRIPTION: Damaris is a middleware for I/O and data management targeting large-scale, MPI-based HPC simulations. It initially proposed to dedicate cores for asynchronous I/O in multicore nodes of recent HPC platforms, with an emphasis on ease of integration in existing simulations, efficient resource usage (with the use of shared memory) and simplicity of extension through plug-ins. Over the years, Damaris has evolved into a more elaborate system, providing the possibility to use dedicated cores or dedicated nodes to in situ data processing and visualization. It proposes a seamless connection to the VisIt visualization framework to enable in situ visualization with minimum impact on run time. Damaris provides an extremely simple API and can be easily integrated into the existing large-scale simulations.

Damaris was at the core of the PhD thesis of Matthieu Dorier, who received an Accessit to the Gilles Kahn Ph.D. Thesis Award of the SIF and the Academy of Science in 2015. Developed in the framework of our collaboration with the JLESC – Joint Laboratory for Extreme-Scale Computing, Damaris was the first software resulted from this joint lab validated in 2011 for integration to the Blue Waters supercomputer project. It scaled up to 16,000 cores on Oak Ridge's leadership supercomputer Titan (first in the Top500 supercomputer list in 2013) before being validated on other top supercomputers. Active development is currently continuing within the KerData team at Inria, where it is at the center of several collaborations with industry as well as with national and international academic partners.
FUNCTIONAL DESCRIPTION: Damaris is a middleware for data management and in-situ visualization targeting large-scale HPC simulations: - In situ data analysis by some dedicated cores/nodes of the simulation platform - Asynchronous and fast data transfer from HPC simulations to Damaris - Semantic-aware dataset processing through Damaris plug-ins - Writing aggregated data (by hdf5 format) or visualizating them either by VisIt or ParaView

- Participants: Gabriel Antoniu, Lokman Rahmani, Luc Bougé, Matthieu Dorier, Orçun Yildiz and Hadi Salimi
- Partner: ENS Rennes
- Contact: Matthieu Dorier
- URL: https://project.inria.fr/damaris/

4.2. OverFlow

FUNCTIONAL DESCRIPTION: OverFlow is a uniform data management system for scientific workflows running across geographically distributed sites, aiming to reap economic benefits from this geo-diversity. The software is environment-aware, as it monitors and models the global cloud infrastructure, offering high and predictable data handling performance for transfer cost and time, within and across sites. OverFlow proposes a set of pluggable services, grouped in a data-scientist cloud kit. They provide the applications with the possibility to monitor the underlying infrastructure, to exploit smart data compression, deduplication and geo-replication, to evaluate data management costs, to set a tradeoff between money and time, and optimize the transfer strategy accordingly.

Currently, OverFlow is used for data transfers by the Microsoft Research ATLE Munich team as well as for synthetic benchmarks at the Politehnica University of Bucharest.

- Participants: Alexandru Costan, Gabriel Antoniu and Radu Marius Tudoran
- Contact: Alexandru Costan

4.3. Pufferbench

KEYWORDS: Distributed Storage Systems - Elasticity - Benchmarking
SCIENTIFIC DESCRIPTION: Pufferbench is a benchmark for evaluating how fast one can scale up and down a distributed storage system on a given infrastructure and, thereby, how viably can one implement storage malleability on it. Besides, it can serve to quickly prototype and evaluate mechanisms for malleability in existing distributed storage systems.
FUNCTIONAL DESCRIPTION: Pufferbench is a benchmark to designed to evaluate whether to use malleable distributed storage systems on a given platform. - It measures the duration of commission and decommission operations. - Its modularity allows to quickly change and adapt each component to the needs of the user. - It can serve as a baseline when implementing commission and decommission mechanisms in a distributed storage system.

RELEASE FUNCTIONAL DESCRIPTION: This is the first release of Pufferbench.

It includes default components for each of the customisable components: - storage: in memory, on drive with file system cache, and on drive without file system cache - network: MPI network - IODispatcher: basic, and with acknowledgements - DataTransferScheduler: basic - DataDistributionGenerator: uniform, and random - MetadataGenerator: Files of same size The diversity of available components enables Pufferbench to fit to multiple use cases.

- Participants: Nathanaël Cheriere, Matthieu Dorier and Gabriel Antoniu
- Partner: ENS Rennes
- Contact: Nathanaël Cheriere
- Publication: Pufferbench: Evaluating and Optimizing Malleability of Distributed Storage
- URL: https://gitlab.inria.fr/Puffertools/Pufferbench/wikis/home

4.4. Tyr

KEYWORDS: Cloud storage - Distributed Storage Systems - Big data
FUNCTIONAL DESCRIPTION: Tyr is the first blob storage system to provide built-in, multiblob transactions, while retaining sequential consistency and high throughput under heavy access concurrency. Tyr offers fine-grained random write access to data and in-place atomic operations.

- Partner: Universidad Politécnica de Madrid
- Contact: Gabriel Antoniu

4.5. Planner

KEYWORDS: Edge elements - Cloud computing - Scheduling
FUNCTIONAL DESCRIPTION: Planner is a middleware for uniform and transparent stream processing across Edge and Cloud. Planner automatically selects which parts of the executiongraph will be executed at the Edge in order to minimize thenetwork cost.

- Partner: ENS Cachan
- Contact: Gabriel Antoniu
- URL: https://team.inria.fr/kerdata/

4.6. KerA

KerAnalytics
KEYWORD: Distributed Storage Systems
FUNCTIONAL DESCRIPTION: A unified architecture for stream ingestion and storage which can lead to the optimization of the processing of Big Data applications. This approach minimizes data movement within the analytics architecture, finally leading to better utilized resources.

- Contact: Gabriel Antoniu

4.7. TailWind

KEYWORDS: Fault-tolerance - Data management. - Distributed Data Management
FUNCTIONAL DESCRIPTION: Replication is essential for fault-tolerance. However, in in-memory systems, it is a source of high overhead. Remote direct memory access (RDMA) is attractive to create redundant copies of data, since it is low-latency and has no CPU overhead at the target. However, existing approaches still result in redundant data copying and active receivers. To ensure atomic data transfers, receivers check and apply only fully received messages. Tailwind is a zero-copy recovery-log replication protocol for scale-out in-memory databases. Tailwind is the first replication protocol that eliminates *all* CPU-driven data copying and fully bypasses target server CPUs, thus leaving backups idle. Tailwind ensures all writes are atomic by leveraging a protocol that detects incomplete RDMA transfers. Tailwind substantially improves replication throughput and response latency compared with conventional RPC-based replication. In symmetric systems where servers both serve requests and act as replicas, Tailwind also improves normal-case throughput by freeing server CPU resources for request processing. We implemented and evaluated Tailwind on RAMCloud, a low-latency in-memory storage system. Experiments show Tailwind improves RAMCloud's normal-case request processing throughput by 1.7×. It also cuts down writes median and 99^{th} percentile latencies by 2x and 3x respectively.

- Contact: Gabriel Antoniu

5. New Results

5.1. Convergence of HPC and Big Data

5.1.1. Large-scale logging for HPC and Big Data convergence

Participants: Pierre Matri, Alexandru Costan, Gabriel Antoniu.

A critical objective set in this convergence context is to foster application portability across platforms. Cloud developers traditionally rely on purpose-specific services to provide the storage model they need for an application. In contrast, HPC developers have a much more limited choice, typically restricted to a centralized parallel file system for persistent storage. Unfortunately, these systems often offer low performance when subject to highly concurrent, conflicting I/O patterns.

This makes difficult the implementation of inherently concurrent data structures such as distributed shared logs. Shared log storage is indeed one of the storage models that are both unavailable and difficult to implement on HPC platforms using the available storage primitives. Yet, this data structure is key to applications such as computational steering, data collection from physical sensor grids, or discrete event generators. A shared log enables multiple processes to append data at the end of a single byte stream. Unfortunately, in such a case, the write contention at the tail of the log is among the worst-case scenarios for parallel file systems, yielding problematically low append performance.

In [25] we introduced SLoG, a shared log middleware providing a shared log abstraction over a parallel file system, designed to circumvent the aforementioned limitations. It features pluggable backends that enable it to leverage other storage models such as object stores or to transparently forward the requests to a shared log storage system when available (e.g., on cloud platforms). SLoG abstracts this complexity away from the developer, fostering application portability between platforms. We evaluated SLoG's performance at scale on a leadership-class supercomputer, using up to 100,000 cores. We measured append velocities peaking at 174 million appends per second, far beyond the capabilities of any shared log storage implementation on HPC platforms. For these reasons, we envision that SLoG could fuel convergence between HPC and big data.

5.1.2. Increasing small files access performance with dynamic metadata replication

Participants: Pierre Matri, Alexandru Costan, Gabriel Antoniu.

Small files are known to pose major performance challenges for file systems. Yet, such workloads are increasingly common in a number of Big Data Analytics workflows or large-scale HPC simulations. These challenges are mainly caused by the common architecture of most state-of-the-art file systems needing one or multiple metadata requests before being able to read from a file. Small input file size causes the overhead of this metadata management to gain relative importance as the size of each file decreases.

In our experiments, with small enough files, opening a file may take up to an order of magnitude more time than reading the data it contains. One key cause of this behavior is the separation of data and metadata inherent to the architecture of current file systems. Indeed, to read a file, a client must first retrieve the metadata for all folders in its access path, that may be located on one or more metadata servers, to check that the user has the correct access rights or to pinpoint the location of the data in the system. The high cost of network communication significantly exceeds the cost of reading the data itself.

In [22] we design a file system from the bottom up for small files without sacrificing performance for other workloads. This enables us to leverage some design principles that address the metadata distribution issues: consistent hashing and dynamic data replication. Consistent hashing enables a client to locate the data it seeks without requiring access to a metadata server, while dynamic replication adapts to the workload and replicates the metadata on the nodes from which the associated data is accessed. The former is often found in key-value stores, while the latter is mostly used in geo-distributed systems. These approaches allow to increase small file access performance up to one order of magnitude compared to other state-of-the-art file systems, while only causing a minimal impact on file write throughput.

5.1.3. Modeling elastic storage

Participants: Nathanaël Cheriere, Gabriel Antoniu.

For efficient Big Data processing, efficient resource utilization becomes a major concern as large-scale computing infrastructures such as supercomputers or clouds keep growing in size. Naturally, energy and cost savings can be obtained by reducing idle resources. Malleability, which is the possibility for resource managers to *dynamically* increase or reduce the resources of jobs, appears as a promising means to progress towards this goal.

However, state-of-the-art parallel and distributed file systems have not been designed with malleability in mind. This is mainly due to the supposedly high cost of storage decommission, which is considered to involve expensive data transfers. Nevertheless, as network and storage technologies evolve, old assumptions on potential bottlenecks can be revisited.

In [28], we establish a lower bound for the duration of the commission operation. We then consider HDFS as a use case, and we show that our lower bound can be used to evaluate the performance of the commission algorithms. We show that the commission in HDFS can be greatly accelerated. With the highlights provided by our lower bound, we suggest improvements to speed the commission in HDFS.

In [29], we explore the possibility of relaxing the level of fault tolerance during the decommission in order to reduce the amount of data transfers needed before nodes are released, and thus return nodes to the resource manager faster. We quantify theoretically how much time and resources are saved by such a fast decommission strategy compared with a standard decommission. We establish lower bounds for the duration of the different phases of a fast decommission. We show that the method not only does not improve performance, but is also unsafe by nature.

In [24], we introduce Pufferbench, a benchmark for evaluating how fast one can scale up and down a distributed storage system on a given infrastructure and, thereby, how viably can one implement storage malleability on it. Besides, it can serve to quickly prototype and evaluate mechanisms for malleability in existing distributed storage systems. We validate Pufferbench against theoretical lower bounds for commission and decommission: it can achieve performance within 16% of them. We use Pufferbench to evaluate in practice these operations in HDFS: commission in HDFS could be accelerated by as much as 14 times! Our results show that: (1) the lower bounds for commission and decommission times we previously established are sound and can be approached in practice; (2) HDFS could handle these operations much more efficiently; most importantly, (3) malleability in distributed storage systems is viable and should be further leveraged for Big Data applications.

During a 3 months visit at Argonne National Lab, the design of an efficient rebalancing algorithm for rescaling operations have been started with Robert Ross. We use the rescaling operation to rebalance the load across the cluster. Performances cannot be sustained without minimizing the amount of data transferred per node, but also the amount of data stored per node. We evaluate a heuristic and show that good approximations of the optimal solutions can be achieved in reasonable time.

5.2. Scalable stream storage

5.2.1. KerA ingestion and storage

Participants: Ovidiu-Cristian Marcu, Alexandru Costan, Gabriel Antoniu.

Big Data is now the new natural resource. Current state-of-the-art Big Data analytics architectures are built on top of a three layer stack: data streams are first acquired by the ingestion layer (e.g., Kafka) and then they flow through the processing layer (e.g., Flink) which relies on the storage layer (e.g., HDFS) for storing aggregated data or for archiving streams for later processing. Unfortunately, in spite of potential benefits brought by specialized layers (e.g., simplified implementation), moving large quantities of data through specialized layers is not efficient: instead, data should be acquired, processed and stored while minimizing the number of copies.

We argue that a plausible path to follow to alleviate from previous limitations is the careful design and implementation of the KerA unified architecture for stream ingestion and storage which can lead to the optimization of the processing of Big Data applications. This approach minimizes data movement within the analytics architecture, finally leading to better utilized resources. We identify a set of requirements for a unified stream ingestion/storage engine. We explain the impact of the different Big Data architectural choices on end-to-end performance. We propose a set of design principles for a scalable, unified architecture for data ingestion and storage: (1) dynamic partitioning based on semantic grouping and sub- partitioning, which enables more flexible and elastic management of stream partitions; (2) lightweight offset indexing (i.e., reduced stream offset management overhead) optimized for sequential record access; (3) adaptive and fine-grained replication to trade-off in-memory storage with performance (low-latency and high throughput with durability). We implement and evaluate the KerA prototype with the goal of efficiently handling diverse access patterns: low-latency access to streams and/or high throughput access to unbounded streams and/or objects [21].

5.2.2. Tailwind: fast and atomic RDMA-based replication

Participants: Yacine Taleb, Gabriel Antoniu.

Replication is essential for fault-tolerance. However, in in-memory systems, it is a source of high overhead. Remote direct memory access (RDMA) is attractive to create redundant copies of data, since it is low-latency and has no CPU overhead at the target. However, existing approaches still result in redundant data copying and active receivers. To ensure atomic data transfers, receivers check and apply only fully received messages.

Tailwind is a zero-copy recovery-log replication protocol for scale-out in-memory databases. Tailwind is the first replication protocol that eliminates *all* CPU-driven data copying and fully bypasses target server CPUs, thus leaving backups idle. Tailwind ensures all writes are atomic by leveraging a protocol that detects incomplete RDMA transfers. Tailwind substantially improves replication throughput and response latency compared with conventional RPC-based replication. In symmetric systems where servers both serve requests and act as replicas, Tailwind also improves normal-case throughput by freeing server CPU resources for request processing. We implemented and evaluated Tailwind on RAMCloud, a low-latency in-memory storage system. Experiments show Tailwind improves RAMCloud's normal-case request processing throughput by 1.7×. It also cuts down writes median and 99^{th} percentile latencies by 2x and 3x respectively [23].

5.3. Hybrid edge/cloud processing

5.3.1. Edge benchmarking

Participants: Pedro Silva, Alexandru Costan, Gabriel Antoniu.

The recent spectacular rise of the Internet of Things and the associated augmentation of the data deluge motivated the emergence of Edge computing as a means to distribute processing from centralized Clouds towards decentralized processing units close to the data sources. This led to new challenges regarding the ways to distribute processing across Cloud-based, Edge-based or hybrid Cloud/Edge-based infrastructures. In particular, a major question is: how much can one improve (or degrade) the performance of an application by performing computation closer to the data sources rather than keeping it in the Cloud?

In the paper "Investigating Edge vs. Cloud Computing Trade-offs for Stream Processing" submitted to CCGrid 2019, it is proposed a methodology to understand such performance trade-offs. Using two representative real-life stream processing applications and state-of-the-art processing engines, we perform an experimental evaluation based on the analysis of the execution of those applications in fully-Cloud computing and hybrid Cloud-Edge computing infrastructures. We derive a set of take-aways for the community, highlighting the limitations of each environment, the scenarios that could benefit from hybrid Edge-Cloud deployments, what relevant parameters impact performance and how.

5.3.2. Planner: cost-efficient execution plans for the uniform placement of stream analytics on Edge and Cloud

Participants: Laurent Prosperi, Alexandru Costan, Pedro Silva, Gabriel Antoniu.

Stream processing applications handle unbounded and continuous flows of data items which are generated from multiple geographically distributed sources. Two approaches are commonly used for processing: Cloud-based analytics and Edge analytics. The first one routes the whole data set to the Cloud, incurring significant costs and late results from the high latency networks that are traversed. The latter can give timely results but forces users to manually define which part of the computation should be executed on Edge and to interconnect it with the remaining part executed in the Cloud, leading to sub-optimal placements.

More recently, a new hybrid approach tries to combine both Cloud and Edge analytics in order to offer better performance, flexibility and monetary costs for stream processing. However, leveraging this dual approach in practice raises some significant challenges mainly due to the way in which stream processing engines organize the analytics workflow. Both Edge and Cloud engines create a dataflow graph of operators that are deployed on the distributed resources; they devise an execution plan by traversing this graph. In order to execute a request over such hybrid deployment, one needs a specific plan for the Edge engines, another one for the cloud engines and to ensure the right interconnection between them thanks to an ingestion system. Manually and empirically deploying this analytics pipeline (Edge-Ingestion-Cloud) can lead to sub-optimal computation placement with respect to the network cost (i.e., high latency, low throughput) between the Edge and the Cloud.

In this [26], we argue that a uniform approach is needed to bridge the gap between Cloud SPEs and Edge analytics frameworks in order to leverage a single, transparent execution plan for stream processing in both environments. We introduce Planner, a streaming middleware capable of finding cost-efficient cuts of execution plans between Edge and Cloud. Our goal is to find a distributed placement of operators on Edge and Cloud nodes to minimize the stream processing makespan. Real-world micro-benchmarks show that Planner reduces the network usage by 40 % and the makespan (end-to-end processing time) by 15 % compared to state-of-the-art.

5.3.3. Integrating KerA and Flink

Participants: Ovidiu-Cristian Marcu, Alexandru Costan, Gabriel Antoniu.

Big Data real-time stream processing typically relies on message broker solutions that decouple data sources from applications. This translates into a three-stage pipeline: (1) event sources (e.g., smart devices, sensors, etc.) continuously generate streams of records; (2) in the *ingestion* phase, these records are acquired, partitioned and pre-processed to facilitate consumption; (3) in the *processing* phase, Big Data engines consume the stream records using a *pull-based* model. Since users are interested in obtaining results as soon as possible, there is a need to minimize the end-to-end latency of the three stage pipeline. This is a non-trivial challenge when records arrive at a fast rate (from producers and to consumers) and create the need to support a high throughput at the same time.

The weak link of the three-stage pipeline is the ingestion phase: it needs to acquire records with a high throughput from the producers, serve the consumers with a high throughput, scale to a large number of producers and consumers, and minimize the write latency of the producers and, respectively, the read latency of the consumers to facilitate low end-to-end latency. Since producers and consumers communicate with message brokers through RPCs, there is inevitably *interference* between these operations which can lead to increased processing times. Moreover, since consumers (i.e., source operators) depend on the networking infrastructure, its characteristics can limit the read throughput and/or increase the end-to-end read latency. One simple idea is to co-locate processing workers (source and other operators) with brokers managing stream partitions. We implement this approach by integrating KerA with Flink through a shared-memory approach. Experiments results demonstrate the effectiveness of our approach.

5.4. Scalable I/O, storage and in-situ visualization

5.4.1. HDF-based storage

Participants: Hadi Salimi, Gabriel Antoniu.

Extreme-scale scientific simulations that are deployed on thousands of cores usually store the resulted datasets in standard formats such as HDF5 or NetCDF. In the data storage process, two different approaches are traditionally employed: 1) file-per-process and 2) collective I/O. In the former approach, each computing core creates its own file at the end of each simulation iteration. However, this approach cannot scale up to thousands of cores because creating and updating thousands of files at the end of each iteration, leads to a poor performance. On the other hand, the latter is based on the coordination of processes to write on a single file that is also expensive in terms of performance.

The proposed approach in this research is to use Damaris for data aggregation and data storage. In the case, the computing resources are partitioned such that a subset of cores in each node or a subset of nodes of the underlying platform is dedicated to data management. The data generated by the simulation processes are transferred to these dedicated cores/nodes either through shared memory (in the case of dedicated cores) or through the MPI calls (in the case of dedicated nodes) and can be processed asynchronously. Afterwards, the aggregated data can be stored in HDF5 format using out-of-the-box Damaris plug-in.

The benefits of using Damaris for storing simulation results into HDF5 is threefold: firstly, Damaris aggregates data from different processes in one process, as a result, the number of I/O writers is decreased; secondly, the write phase becomes entirely asynchronous, so the simulation processes do not have to wait for the write phase to be completed; and finally, the Damaris API is much more straightforward for simulation developers. Hence it can be easily integrated in simulation codes and easily maintained as well. The performance evaluation of the implemented prototype shows that using Damaris for storing simulation data can lead up to 297 % improvement compared to the standard file-per-process approach [32].

5.4.2. Leveraging Damaris for in-situ visualization in support of GeoScience and CFD simulations

Participants: Hadi Salimi, Gabriel Antoniu.

In the context of an industrial collaboration, KerData managed to sign a contract with Total around Damaris. Total is one of the industrial pioneers of HPC in France and owns the fastest supercomputer in France, named Pangea. On this machine, lots of geoscience simulations (oil exploration, oil extraction, seismic, etc.) are executed everyday and the results of these simulations are used by company's geoscientists.

This feasibility study on using Damaris on Total's geoscience simulations has been subject to an expertise contract between Total and KerData. The main goal of the contract is to show that Damaris is capable of supporting Total simulations to provide asynchronous I/O and in situ visualization. To this aim, by instrumenting two wave propagation simulation codes (prepared by Total), it was shown that Damaris can be applied to Total's wave propagation simulations in support of in situ visualization and asynchronous I/O.

The amount of changes made into the target simulations to support Damaris shows that for simple and complex simulations, the amount of changes in the simulation source code remain nearly the same. In addition, those part of the simulation code that are dedicated to dumping of the results can be totally removed, because Damaris supports this feature in a simpler and even more efficient way.

6. Bilateral Contracts and Grants with Industry

6.1. Bilateral Contracts with Industry

6.1.1. Total: In situ Visualization with Damaris (2017-2018)

Participants: Hadi Salimi, Matthieu Dorier, Gabriel Antoniu, Luc Bougé.

The goal of this expertise contract is to 1) disseminate the usage of Damaris for engineers at Total; 2) to realize a feasibility study for the usage of Damaris for in situ analysis of data for Total's HPC reservoir simulations.

6.1.2. Huawei: HIRP Low-Latency Storage for Stream Data (2017–2018)

Participants: Alexandru Costan, Ovidiu-Cristian Marcu, Gabriel Antoniu.

The goal of this project is to explore the plausible paths towards a dedicated storage solution for low-latency stream storage. Such a solution should provide on the one hand traditional storage functionality and on the other hand stream-like performance (i.e., low-latency I/O access to items and ranges of items).

We have investigated the main requirements and challenges, evaluated the different design choices (e.g., a standalone component vs. an extension of an existing Big Data solution like HDFS) and proposed a new converged architecture for smart storage.

7. Partnerships and Cooperations

7.1. National Initiatives

7.1.1. ANR

7.1.1.1. OverFlow (2015–2019)

Participants: Alexandru Costan, Pedro Silva, Paul Le Noac'h.

Project Acronym: OverFlow.

Project Title: Workflow Data Management as a Service for Multisite Applications.

Coordinator: Alexandru Costan.

Duration: Octobre 2015–October 2019.

Other Partners: None (Young Researcher Project).

External collaborators: Kate Keahey (University of Chicago and Argonne National Laboratory), Bogdan Nicolae (Argonne National Lab).

Abstract: This JCJC project led by Alexandru Costan investigates approaches to data management enabling an efficient execution of geographically distributed workflows running on multi-site clouds.

In 2018, we focused on the challenges of executing workflows on hybrid environments combining the Cloud and the Edge. First, processing live data sources at the Edge can offer a potential solution that deals with the explosion of data sizes, as the data is filtered and aggregated locally, before it gets a chance to accumulate. Then, partial results instead of full data are sent to the Cloud for stream processing. In this context, we designed Planner, a middleware for uniform and transparent stream processing across Edge and Cloud. Planner automatically selects which parts of the execution graph will be executed at the Edge in order to minimize the network cost. We also focused on understanding the conditions that enable the usage of Edge or Cloud to improve the performance or reduce costs of an application.

7.1.2. Other National Projects

7.1.2.1. HPC-Big Data Inria Project Lab (IPL)

Participants: Gabriel Antoniu, Alexandru Costan, Pedro Silva.

Project Acronym: HPC-BigData

Project Title: The HPC-BigData Inria Project Lab

Coordinator: Bruno Raffin.

Duration: 2018–2022.

Abstract: The goal of this HPC-BigData IPL is to gather teams from the HPC, Big Data and Machine Learning (ML) areas to work at the intersection between these domains. Research is organized along three main axes: high performance analytics for scientific computing applications, high performance analytics for big data applications, infrastructure and resource management. Gabriel Antoniu is a member of the Advisory Board and leader of the Frameworks work package.

7.1.2.2. ADT Damaris

Participants: Hadi Salimi, Gabriel Antoniu, Luc Bougé.

Project Acronym: ADT Damaris

Project Title: Technology development action for te Damaris environment.

Coordinator: Alexandru Costan.

Duration: 2016–2018.

Abstract: This action aims to support the development of the Damaris software. Inria's *Technological Development Office* (D2T, *Direction du Développement Technologique*) provided 2 years of funding support for a senior engineer.

Hadi Salimi has been funded through this project to document, test and extend the Damaris software and make it a safely distributable product.

In 2018, the main goal was to enforce the support Damaris provides for HDF5 storage.

7.1.2.3. Grid'5000

We are members of Grid'5000 community and run experiments on the Grid'5000 platform on a daily basis.

7.2. European Initiatives

7.2.1. FP7 and H2020 Projects

7.2.1.1. BigStorage

Title: BigStorage: Storage-based Convergence between HPC and Cloud to handle Big Data.

Programme: H2020 ETN.

Duration: January 2015–December 2018.

Coordinator: Universidad Politecnica de Madrid (UPM).

Partners:

- Barcelona Supercomputing Center — Centro Nacional de Supercomputacion (Spain)
- CA Technologies Development Spain (Spain)
- CEA — Commissariat à l'énergie atomique et aux énergies alternatives (France)
- Deutsches Klimarechenzentrum (Germany)
- Foundation for Research and Technology Hellas (Greece)
- Fujitsu Technology Solutions (Germany)
- Johannes Gutenberg Universitaet Mainz (Germany)
- Universidad Politecnica de Madrid (Spain)
- Seagate Systems UK (United Kingdom)

Inria contact: Gabriel Antoniu and Adrien Lèbre.

URL: http://www.bigstorage-project.eu/.

Description: BigStorage is a European Training Network (ETN) whose main goal is to train future *data scientists*. It aims at enabling them and us to apply holistic and interdisciplinary approaches to take advantage of a data-overwhelmed world. This world requires *HPC* and *Cloud* infrastructures with a redefinition of *storage* architectures underpinning them — focusing on meeting highly ambitious performance and *energy* usage objectives. The KerData team has hosted 2 *Early Stage Researchers* in this framework and has co-advised an extra PhD student.

7.2.2. Collaborations with Major European Organizations

7.2.2.1. BDVA and ETP4HPC

Gabriel Antoniu and Alexandru Costan are serving as Inria representatives in the working groups dedicated to *HPC-Big Data* convergence within the Big Data Value Association (BDVA) and the European Technology Platform in the area of High-Performance Computing (ETP4HPC). They are contributing to the definition of the respective Strategic Research Agendas of BDVA and ETP4HPC. A special focus this year of their contributions was the Joint BDVA-ETP4HPC report on technology convergence.

7.3. International Initiatives

7.3.1. Inria International Labs

7.3.1.1. JLESC: Joint Laboratory for Extreme Scale Computing

The Joint Laboratory on Extreme-Scale Computing is jointly run by Inria, UIUC, ANL, BSC, JSC and RIKEN/AICS. It has been created in 2014 as a follow-up of the Inria-UIUC JLPC, the *Joint Laboratory for Petascale Computing*.

The KerData team is collaborating with teams from ANL and UIUC within this lab since 2009 on several topics in the areas of I/O, storage and in situ processing and cloud computing. This collaboration has been initially formalized as the *Data@Exascale* Associate Team with ANL and UIUC (2013–2015) followed by *Data@Exascale 2* Associate Team with ANL (2016–2018). Our activities in this framework are described here: http://www.irisa.fr/kerdata/data-at-exascale/

Since 2015, Gabriel Antoniu serves as a topic leader for Inria for the *I/O, Storage and In Situ Processing* topic. Ongoing lab research directions and projects he is co-supervising in this area are described here: https://jlesc.github.io/projects/ in the *I/O, Storage and In-Situ Processing* section.

Since 2017, Gabriel Antoniu is serving as *Vice-Executive Director* of JLESC for Inria.

7.3.1.2. Associate Team involved in the JLESC International Lab: Data@Exascale 2

Title: Convergent Data Storage and Processing Approaches for Exascale Computing and Big Data Analytics

Partner: Argonne National Laboratory (United States), Department of Mathematics, Symbolic Computation Group, Robert Ross

Web site: http://www.irisa.fr/kerdata/data-at-exascale/

Start year: 2016

In the past few years, countries including United States, the European Union, Japan and China have set up aggressive plans to get closer to what appears to be the next goal in terms of high-performance computing (HPC): Exascale computing, a target which is now considered reachable by the next-generation supercomputers in 2020–2023. While these government-led initiatives have naturally focused on the big challenges of exascale for the development of new hardware and software architectures, the quite recent emergence of the Big Data phenomenon introduces what could be called a tectonic shift that is impacting the entire research landscape for exascale computing. As data generation capabilities in most science domains are now growing substantially faster than computational capabilities, causing these domains to become data-intensive, new challenges appeared in terms of volumes and velocity for data to be stored, processed and analyzed on the future exascale machines.

To face the challenges generated by the exponential data growth (a general phenomenon in many fields), a certain progress has already been made in the recent years in the rapidly-developing, industry-led field of cloud-based Big Data analytics, where advanced tools emerged, relying on machine-learning techniques and predictive analytics. Unfortunately, these advances cannot be immediately applied to exascale computing: the tools and cultures of the two worlds, HPC (High-Performance Computing) and BDA (Big Data Analytics) have developed in a divergent fashion (in terms of major focus and technical approaches), to the detriment of both. The two worlds share however multiple similar challenges and unification now appears as essential in order to address the future challenges of major application domains that can benefit from both.

The scientific program of the *Data@Exascale 2* Associate Team is defined from this new, highly-strategic perspective and builds on the idea that the design of innovative approaches to data I/O, storage and processing allowing Big Data analytics techniques and the newest HPC architectures to leverage each other clearly appears as a key catalyst factor for the convergence process.

Activities in 2018 are described on the web site of the Associate Team.

7.3.2. Inria International Partners

7.3.2.1. DataCloud@Work

Title: DataCloud@Work.

International Partner:

- Polytechnic University of Bucharest (Romania), Computer Science Department, Nicolae Tapus and Valentin Cristea.

Duration: 5 years.

Start year: 2013. The status of IIP was established right after the end of our former *DataCloud@work* Associate Team (2010–2012).

URL: https://www.irisa.fr/kerdata/doku.php?id=cloud_at_work:start.

Description: Our research topics address the area of distributed data management for cloud services, focusing on autonomic storage. The goal is explore how to build an efficient, secure and reliable storage IaaS for data-intensive distributed applications running in cloud environments by enabling an autonomic behavior.

7.3.2.2. Informal International Partners

Instituto Politécnico Nacional, IPN, Ciudad de México: We continued our informal collaboration in the area of stream processing.

7.3.3. Participation in Other International Programs

7.3.3.1. International Initiatives

7.3.3.1.1. BDEC: Big Data and Extreme Computing

Since 2015, Gabriel Antoniu has been invited to participate to the yearly workshops of the international Big Data and Extreme-scale Computing (BDEC) working group focused on the convergence of Extreme Computing (the latest incarnation of High-Performance Computing - HPC) and Big Data. BDEC is organized as an yearly series of invitation-based international workshops.

In 2018 Gabriel Antoniu was invited again to contribute to the first workshop of the BDEC2 series, where he presented a white paper on HPC-Big Data convergence at the level of data processing.

7.4. International Research Visitors

7.4.1. Visits of International Scientists

Rob Ross: Argonne National Laboratory, USA

Ryan Stutsman: University of Utah, USA

Tilmann Rabl: TU Berlin/DFKI/Berlinn Big Data Center, Germany

Nicolae Tapus: Politehnica University of Bucharest

7.4.2. Internships

Laurent Prosperi (M1, ENS Cachan) has done a 4-month internship within the team, working with Alexandru Costan and Pedro Silva on hybrid Edge/Cloud stream processing. This work lead to the Planner middleware [26], presented at the WORKS workshop at the IEEE/ACM SC18 conference.

7.4.3. Visits to International Teams

7.4.3.1. Research Stays Abroad

Nathanaël Cheriere has done a 3-month internship at Argonne National Lab, to work on optimizing data migration for efficient distributed storage system rescaling under the supervision of Robert Ross. See Section 5.1 for details.

Yacine Taleb has done a 1-month internship at the University of Utah to work on RDMA replication for in-memory storage systems under the supervision of Ryan Stutsman. See Section 5.1 for details.

Yacine Taleb has done a 3-month internship at Barcelona Supercomputing Center, to work on in-memory storage for Big Data analytics under the supervision of Toni Cortés. See Section 5.1 for details.

8. Dissemination

8.1. Promoting Scientific Activities

8.1.1. Scientific Events Organisation

8.1.1.1. General Chair, Scientific Chair

Luc Bougé: Chair of the Steering Committee of the Euro-Par Series of conferences since August 2017.

8.1.2. Scientific Events Selection

8.1.2.1. Chair of Conference Program Committees

Alexandru Costan:

- Program Co-Chair of the ScienceCloud 2018 international workshop held in conjunction with ACM HPDC 2018, Tucson, AZ, USA.

8.1.2.2. Member of the Conference Program Committees

Gabriel Antoniu: ACM HPDC 2018, IEEE IPDPS 2018, ACM/IEEE SC'18: Papers Committee, Posters Committee, Best Poster Award and Student Research Competition Committee, IEEE CSE-2018, BDA 2018.

Alexandru Costan: IEEE/ACM SC'18 (Posters and ACM Student Research Competition), ACM/IEEE CCGrid 2018, IEEE Cluster 2018, IEEE/ACM UCC 2018, ARMS-CC 2018 workshop (held in conjunction with PODC 2018), IEEE Big Data 2018, IEEE CSE 2018, MLDS 2018, EBDMA 2018.

Pedro Silva: ScienceCloud 2018 Workshop – Co-located with ACM HPDC 2018.

8.1.2.3. Reviewer

Alexandru Costan: ACM HPDC 2018, IEEE/ACM SC 2018, IEEE IPDPS 2018

Pedro Silva: IEEE Big Data, IEEE Cloud 2018, IEEE Cluster, IEEE CCGrid 2018, ACM HPDC 2018, IEEE/ACM SC 2018.

8.1.3. Journals

8.1.3.1. Member of the Editorial Boards

Gabriel Antoniu: Future Generation Computer Systems: Special Issue on Mobile, hybrid, and heterogeneous clouds for cyberinfrastructures (Guest Editor); Special Issue on Resource Management for Big Data Platforms (Guest Editor).

8.1.3.2. Reviewer - Reviewing Activities

Alexandru Costan: IEEE Transactions on Parallel and Distributed Systems, Future Generation Computer Systems, Concurrency and Computation Practice and Experience, IEEE Transactions on Cloud Computing, IEEE Transactions on Big Data.

8.1.4. Invited Talks

Gabriel Antoniu and Alexandru Costan: Huawei European Research Center, Munich, January 2018. *Low-latency Storage for Stream Data*.

Gabriel Antoniu:

- First workshop of the BDEC2 workshop series, Bloomington, November 2018. *The Sigma Data Processing Architecture: Leveraging Future Data for Extreme-Scale Data Analytics to Enable High-Precision Decisions*.
- 8th workshop of the Joint Laboratory for Extreme-Scale Computing (JLESC), Barcelona, April 2018. *HPC-Big Data convergence at the processing level: a vision*.
- Sintef, Oslo, October 2018: *HPC-Big Data convergence at the processing level: a vision*.

8.1.5. Leadership within the Scientific Community

Gabriel Antoniu:

International lab management *Vice Executive Director of JLESC* for Inria. JLESC is the Joint Inria-Illinois-ANL-BSC-JSC-RIKEN/AICS Laboratory for Extreme-Scale Computing. Within JLESC, he also serves as a *Topic Leader* for Data storage, I/O and in situ processing for Inria.

Team management *Head of the KerData Project-Team* (Inria-ENS Rennes-INSA Rennes).

International Associate Team management Leader of the Data@Exascale Associate Team with Argonne National Lab (2013–2018).

PI of a collaborative research project with industry : Huawei Technologies, HIRP program(Huawei Innovative Research Project Program, 2016-2018).

Technology development project management Coordinator of the Damaris ADT project (2016–2018), to be continued with the Damaris 2 ADT project (2019–2021).

Luc Bougé: Vice-President of the *French Society for Informatics* (*Société informatique de France*, SIF), in charge of Teaching.

Alexandru Costan: Leader of the *Smart Cities* Working Group within the BigStorage H2020 ETN project.

8.1.6. Scientific Expertise

Luc Bougé: Member of the jury for the *Agrégation de mathématiques* and the *CAPES of mathématiques*. These national committees select permanent mathematics teachers for secondary schools and high-schools, respectively.

8.2. Teaching, Supervision, Juries

8.2.1. Teaching

Gabriel Antoniu

- Master (Engineering Degree, 5th year): Big Data, 24 hours (lectures), M2 level, ENSAI (*École nationale supérieure de la statistique et de l'analyse de l'information*), Bruz, France.
- Master: Scalable Distributed Systems, 12 hours (lectures), M1 level, SDS Module, EIT ICT Labs Master School, France.
- Master: Infrastructures for Big Data, 12 hours (lectures), M2 level, IBD Module, SIF Master Program, University of Rennes, France.
- Master: Cloud Computing and Big Data, 10 hours (lectures), M2 level, Cloud Module, MIAGE Master Program, University of Rennes, France.

Luc Bougé

- Bachelor: Introduction to programming concepts, 36 hours (lectures), L3 level, Informatics program, ENS Rennes, France.
- Bachelor: Introduction to scientific research, 24 hours. Research center visits, individual research project supervised by local researchers, student seminars, summer internships, etc.
- Master Program, Rennes: Invited presentation to the M2 students about *Preparing your applications after your PhD* (November 2018); *Informatics as a scientific activity: Toward a responsible research* (November 2018).

Alexandru Costan

- Bachelor: Software Engineering and Java Programming, 28 hours (lab sessions), L3, INSA Rennes.
- Bachelor: Databases, 68 hours (lectures and lab sessions), L2, INSA Rennes, France.
- Bachelor: Practical case studies, 24 hours (project), L3, INSA Rennes.
- Master: Big Data Storage and Processing, 28h hours (lectures, lab sessions), M1, INSA Rennes.
- Master: Algorithms for Big Data, 28 hours (lectures, lab sessions), M2, INSA Rennes.
- Master: Big Data Project, 28 hours (project), M2, INSA Rennes.

Pedro Silva

- Master: Algorithms for Big Data, 4 hours (lectures), M2, INSA Rennes.
- Master: Algorithms for Big Data, 6 hours (lab sessions), M2, INSA Rennes.

8.2.2. Supervision

8.2.2.1. PhD completed this year

Pierre Matri: *Tyr: Storage-Based HPC and Big Data Convergence Using Transactional Blobs*, Universidad Politécnica de Madrid, defended June 10, 2018, co-advised by María Pérez (Universidad Politécnica de Madrid) and Gabriel Antoniu.

Mohammed-Yacine Taleb: *Energy-impact of data consistency management in Clouds and Beyond* [15], ENS Rennes, defended on October 2, 2018, co-advised by Gabriel Antoniu and Toni Cortés (Barcelona Supercomputing Center).

Ovidiu-Cristian Marcu: *Efficient data transfer and streaming strategies for workflow-based Big Data processing*, INSA Rennes, defended on December 18, 2018, co-advised by Alexandru Costan and Gabriel Antoniu.

8.2.2.2. PhD in progress

Nathanaël Cheriere: *Resource Management and Scheduling for Big Data Applications in Large-scale Systems*, thesis started in September 2016, co-advised by Gabriel Antoniu and Matthieu Dorier.

Paul Le Noac'h: *Workflow Data Management as a Service for Multi-Site Applications*, thesis started in November 2016, co-advised by Alexandru Costan and Luc Bougé.

8.2.3. Juries

Gabriel Antoniu:

Barcelona Supercomputing Center STAR Postdoctoral Programme: Member of the evaluation panel in 2018 (30 applications reviewed).

PhD juries: Referee for 3 PhD juries for PhD defenses at Barcelona Supercomputing Center, Université de Grenoble and UPMC, Paris.

8.3. Popularization

8.3.1. Internal or external Inria responsibilities

Gabriel Antoniu:

Big Data Value Association: Inria representative in the working group on HPC-Big Data convergence since 2018.

ETP4HPC: Inria representative in the working group on HPC-Big Data convergence since 2018.

Luc Bougé: Co-ordinator between ENS Rennes and the Inria Research Center and the IRISA laboratory.

Alexandru Costan:

- In charge of internships at the Computer Science Department of INSA Rennes.
- In charge of the organization of the IRISA D1 Department Seminars.

8.3.2. Articles and contents

HDF5 Blog: Invited blog article explaining how Damaris can support HDF5-based storage.

9. Bibliography

Major publications by the team in recent years

[1] N. CHERIERE, M. DORIER. *Design and Evaluation of Topology-aware Scatter and AllGather Algorithms for Dragonfly Networks*, November 2016, Supercomputing 2016, Poster, https://hal.inria.fr/hal-01400271

[2] A. COSTAN, R. TUDORAN, G. ANTONIU, G. BRASCHE. *TomusBlobs: Scalable Data-intensive Processing on Azure Clouds*, in "CCPE - Concurrency and Computation: Practice and Experience", May 2013, https://hal.inria.fr/hal-00767034

[3] B. DA MOTA, R. TUDORAN, A. COSTAN, G. VAROQUAUX, G. BRASCHE, P. J. CONROD, H. LEMAITRE, T. PAUS, M. RIETSCHEL, V. FROUIN, J.-B. POLINE, G. ANTONIU, B. THIRION. *Machine Learning Patterns for Neuroimaging-Genetic Studies in the Cloud*, in "Frontiers in Neuroinformatics", April 2014, vol. 8, https://hal.inria.fr/hal-01057325

[4] M. DORIER, G. ANTONIU, F. CAPPELLO, M. SNIR, L. ORF. *Damaris: How to Efficiently Leverage Multicore Parallelism to Achieve Scalable, Jitter-free I/O*, in "CLUSTER - IEEE International Conference on Cluster Computing", Beijing, China, IEEE, September 2012, https://hal.inria.fr/hal-00715252

[5] M. DORIER, G. ANTONIU, F. CAPPELLO, M. SNIR, R. SISNEROS, O. YILDIZ, S. IBRAHIM, T. PETERKA, L. ORF. *Damaris: Addressing Performance Variability in Data Management for Post-Petascale Simulations*, in "ACM Transactions on Parallel Computing", 2016, https://hal.inria.fr/hal-01353890

[6] M. DORIER, G. ANTONIU, R. ROSS, D. KIMPE, S. IBRAHIM. *CALCioM: Mitigating I/O Interference in HPC Systems through Cross-Application Coordination*, in "IPDPS - International Parallel and Distributed Processing Symposium", Phoenix, United States, May 2014, https://hal.inria.fr/hal-00916091

[7] M. DORIER, M. DREHER, T. PETERKA, G. ANTONIU, B. RAFFIN, J. M. WOZNIAK. *Lessons Learned from Building In Situ Coupling Frameworks*, in "ISAV 2015 - First Workshop on In Situ Infrastructures for Enabling Extreme-Scale Analysis and Visualization (held in conjunction with SC15)", Austin, United States, November 2015 [*DOI :* 10.1145/2828612.2828622], https://hal.inria.fr/hal-01224846

[8] M. DORIER, S. IBRAHIM, G. ANTONIU, R. ROSS. *Omnisc'IO: A Grammar-Based Approach to Spatial and Temporal I/O Patterns Prediction*, in "SC14 - International Conference for High Performance Computing, Networking, Storage and Analysis", New Orleans, United States, IEEE, ACM, November 2014, https://hal.inria.fr/hal-01025670

[9] M. DORIER, S. IBRAHIM, G. ANTONIU, R. ROSS. *Using Formal Grammars to Predict I/O Behaviors in HPC: the Omnisc'IO Approach*, in "TPDS - IEEE Transactions on Parallel and Distributed Systems", October 2015 [*DOI :* 10.1109/TPDS.2015.2485980], https://hal.inria.fr/hal-01238103

[10] P. MATRI, A. COSTAN, G. ANTONIU, J. MONTES, M. S. PÉREZ-HERNÁNDEZ. *Týr: Blob Storage Meets Built-In Transactions*, in "IEEE ACM SC16 - The International Conference for High Performance Computing, Networking, Storage and Analysis 2016", Salt Lake City, United States, November 2016, https://hal.inria.fr/hal-01347652

[11] B. NICOLAE, G. ANTONIU, L. BOUGÉ, D. MOISE, A. CARPEN-AMARIE. *BlobSeer: Next-Generation Data Management for Large-Scale Infrastructures*, in "JPDC - Journal of Parallel and Distributed Computing", February 2011, vol. 71, n^{o} 2, pp. 169–184, http://hal.inria.fr/inria-00511414/en/

[12] B. NICOLAE, J. BRESNAHAN, K. KEAHEY, G. ANTONIU. *Going Back and Forth: Efficient Multi-Deployment and Multi-Snapshotting on Clouds*, in "HPDC 2011 - The 20th International ACM Symposium on High-Performance Parallel and Distributed Computing", San José, CA, United States, June 2011, http://hal.inria.fr/inria-00570682/en

[13] R. TUDORAN, A. COSTAN, G. ANTONIU. *OverFlow: Multi-Site Aware Big Data Management for Scientific Workflows on Clouds*, in "IEEE Transactions on Cloud Computing", June 2015 [*DOI :* 10.1109/TCC.2015.2440254], https://hal.inria.fr/hal-01239128

Publications of the year

Doctoral Dissertations and Habilitation Theses

[14] O.-C. MARCU. *KerA: A Unified Ingestion and Storage System for Scalable Big Data Processing*, INSA Rennes, December 2018, https://tel.archives-ouvertes.fr/tel-01972280

[15] M. Y. TALEB. *Optimizing Distributed In-memory Storage Systems Fault-tolerance, Performance, Energy Efficiency*, École normale supérieure de Rennes, October 2018, https://tel.archives-ouvertes.fr/tel-01891897

Articles in International Peer-Reviewed Journals

[16] J. CARRETERO, J. GARCIA-BLAS, G. ANTONIU, D. PETCU. *New directions in mobile, hybrid, and heterogeneous clouds for cyberinfrastructures*, in "Future Generation Computer Systems", October 2018, vol. 87, pp. 615 - 617 [*DOI :* 10.1016/J.FUTURE.2018.05.073], https://hal.archives-ouvertes.fr/hal-01892931

[17] J. LIU, L. PINEDA, E. PACITTI, A. COSTAN, P. VALDURIEZ, G. ANTONIU, M. MATTOSO. *Efficient Scheduling of Scientific Workflows using Hot Metadata in a Multisite Cloud*, in "IEEE Transactions on Knowledge and Data Engineering", 2018, pp. 1-20 [*DOI :* 10.1109/TKDE.2018.2867857], https://hal-lirmm.ccsd.cnrs.fr/lirmm-01867717

[18] P. MATRI, Y. ALFOROV, A. BRANDON, M. S. PÉREZ-HERNÁNDEZ, A. COSTAN, G. ANTONIU, M. KUHN, P. CARNS, T. LUDWIG. *Mission Possible: Unify HPC and Big Data Stacks Towards Application-Defined Blobs at the Storage Layer*, in "Future Generation Computer Systems", July 2018, pp. 1-10 [*DOI :* 10.1016/J.FUTURE.2018.07.035], https://hal.archives-ouvertes.fr/hal-01892682

[19] P. MATRI, M. S. PÉREZ-HERNÁNDEZ, A. COSTAN, L. BOUGÉ, G. ANTONIU. *Keeping up with storage: Decentralized, write-enabled dynamic geo-replication*, in "Future Generation Computer Systems", September 2018, vol. 86, pp. 1093-1105 [*DOI :* 10.1016/J.FUTURE.2017.06.009], https://hal.inria.fr/hal-01617658

[20] F. POP, R. PRODAN, G. ANTONIU. *RM-BDP: Resource management for Big Data platforms*, in "Future Generation Computer Systems", September 2018, vol. 86, pp. 961 - 963 [*DOI :* 10.1016/J.FUTURE.2018.05.018], https://hal.archives-ouvertes.fr/hal-01892942

International Conferences with Proceedings

[21] O.-C. MARCU, A. COSTAN, G. ANTONIU, M. S. PÉREZ-HERNÁNDEZ, B. NICOLAE, R. TUDORAN, S. BORTOLI. *KerA: Scalable Data Ingestion for Stream Processing*, in "ICDCS 2018 - 38th IEEE International Conference on Distributed Computing Systems", Vienna, Austria, IEEE, July 2018, pp. 1480-1485 [*DOI :* 10.1109/ICDCS.2018.00152], https://hal.inria.fr/hal-01773799

[22] P. MATRI, M. S. PÉREZ-HERNÁNDEZ, A. COSTAN, G. ANTONIU. *TýrFS: Increasing Small Files Access Performance with Dynamic Metadata Replication*, in "CCGRID 2018 - 18th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing", Washington, United States, IEEE, May 2018, pp. 452-461 [*DOI :* 10.1109/CCGRID.2018.00072], https://hal.archives-ouvertes.fr/hal-01892691

[23] M. Y. TALEB, R. STUTSMAN, G. ANTONIU, T. CORTES. *Tailwind: Fast and Atomic RDMA-based Replication*, in "ATC '18 - USENIX Annual Technical Conference", Boston, United States, July 2018, pp. 850-863, https://hal.inria.fr/hal-01676502

Conferences without Proceedings

[24] N. CHERIERE, M. DORIER, G. ANTONIU. *Pufferbench: Evaluating and Optimizing Malleability of Distributed Storage*, in "PDSW-DISCS 2018: 3rd Joint International workshop on Parallel Data Storage & Data Intensive Scalable computing Systems", Dallas, United States, November 2018, pp. 1-10, https://hal.archives-ouvertes.fr/hal-01892713

[25] P. MATRI, P. CARNS, R. ROSS, A. COSTAN, M. S. PÉREZ-HERNÁNDEZ, G. ANTONIU. *SLoG: Large-Scale Logging Middleware for HPC and Big Data Convergence*, in "ICDCS 2018 - IEEE 38th International Conference on Distributed Computing Systems", Vienna, Austria, IEEE, July 2018, pp. 1-6 [*DOI :* 10.1109/ICDCS.2018.00156], https://hal.archives-ouvertes.fr/hal-01892685

[26] L. PROSPERI, A. COSTAN, P. SILVA, G. ANTONIU. *Planner: Cost-efficient Execution Plans Placement for Uniform Stream Analytics on Edge and Cloud*, in "WORKS 2018: 13th Workflows in Support of Large-Scale Science Workshop, held in conjunction with the IEEE/ACM SC18 conference", Dallas, United States, November 2018, pp. 1-10, https://hal.archives-ouvertes.fr/hal-01892718

Books or Proceedings Editing

[27] D. BLANCO HERAS, L. BOUGÉ, E. JEANNOT, R. SAKELLARIOU, R. M. BADIA, J. G. BARBOSA, L. RICCI, S. L. SCOTT, S. LANKES, J. WEIDENDORFER (editors). *Euro-Par 2017 International Workshops, Santiago de Compostela, Spain, August 28-29, 2017, Revised Selected Papers*, Lecture Notes in Computer Science, Springer, Santiago de Compostella, Spain, 2018, vol. 10659 [*DOI :* 10.1007/978-3-319-75178-8], https://hal.inria.fr/hal-01962797

Research Reports

[28] N. CHERIERE, M. DORIER, G. ANTONIU. *A Lower Bound for the Commission Times in Replication-Based Distributed Storage Systems*, Inria Rennes - Bretagne Atlantique, June 2018, n° RR-9186, pp. 1-26, https://hal.archives-ouvertes.fr/hal-01817638

[29] N. CHERIERE, M. DORIER, G. ANTONIU. *Lower Bounds for the Duration of Decommission Operations with Relaxed Fault Tolerance in Replication-based Distributed Storage Systems*, Inria Rennes - Bretagne Atlantique, December 2018, n° RR-9229, pp. 1-28, https://hal.archives-ouvertes.fr/hal-01943964

[30] O.-C. MARCU, A. COSTAN, G. ANTONIU, M. S. PÉREZ-HERNÁNDEZ, R. TUDORAN, S. BORTOLI, B. NICOLAE. *Storage and Ingestion Systems in Support of Stream Processing: A Survey*, Inria Rennes - Bretagne Atlantique and University of Rennes 1, France, November 2018, n° RT-0501, pp. 1-33, https://hal.inria.fr/hal-01939280

References in notes

[31] *Amazon Elastic Map-Reduce (EMR)*, 2017, https://aws.amazon.com/emr/

[32] *Damaris: An Asynchronous Data Aggregator Middleware in Support of HDF5 Library*, 2018, https://www.hdfgroup.org/2018/06/damaris-an-asynchronous-data-aggregator-middleware-in-support-of-hdf5-library/

[33] *The Decaf Project*, 2017, https://bitbucket.org/tpeterka1/decaf

[34] *Digital Single Market*, 2015, https://ec.europa.eu/digital-single-market/en/digital-single-market

[35] *European Exascale Software Initiative*, 2013, http://www.eesi-project.eu

[36] *The European Technology Platform for High-Performance Computing*, 2012, http://www.etp4hpc.eu

[37] *European Cloud Strategy*, 2012, https://ec.europa.eu/digital-single-market/en/european-cloud-computing-strategy

[38] *Apache Flink*, 2016, http://flink.apache.org

[39] *International Exascale Software Program*, 2011, http://www.exascale.org/iesp/Main_Page

[40] *Scientific challenges of the Inria Rennes-Bretagne Atlantique research centre*, 2016, https://www.inria.fr/en/centre/rennes/research

[41] *Inria's strategic plan "Towards Inria 2020"*, 2016, https://www.inria.fr/en/institute/strategy/strategic-plan

[42] *Joint Laboratory for Extreme Scale Computing (JLESC)*, 2017, https://jlesc.github.io

[43] *Apache Spark*, 2017, http://spark.apache.org

[44] *Storm*, 2014, http://storm.apache.org

[45] *The FlowVR Project*, 2014, http://flowvr.sourceforge.net/

[46] T. AKIDAU, A. BALIKOV, K. BEKIROĞLU, S. CHERNYAK, J. HABERMAN, R. LAX, S. MCVEETY, D. MILLS, P. NORDSTROM, S. WHITTLE. *MillWheel: fault-tolerant stream processing at internet scale*, in "Proceedings of the VLDB Endowment", 2013, vol. 6, n° 11, pp. 1033–1044

[47] J. DEAN, S. GHEMAWAT. *MapReduce: simplified data processing on large clusters*, in "Communications of the ACM", 2008, vol. 51, n° 1, pp. 107–113

[48] S. WILDE, M. HATEGAN, J. M. WOZNIAK, B. CLIFFORD, D. KATZ, I. T. FOSTER. *Swift: A language for distributed parallel scripting*, in "Parallel Computing", 2011, vol. 37, n° 9, pp. 633–652, http://dx.doi.org/10.1016/j.parco.2011.05.005