INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Team KerData

Scalable Storage for Clouds and Beyond

Rennes - Bretagne-Atlantique

Theme : Distributed and High Performance Computing

Table of contents

The KerData Team has been officially created on July 1st, 2009. It is a spinoff of the Paris Project-Team. It corresponds to the former "Data management" activity of the Paris Project-Team.

1. Team

Research Scientist

Gabriel Antoniu [Team leader, Junior Researcher (CR1) INRIA, HdR]

Faculty Member

Luc Bougé [Professor, ENS CACHAN Brittany Campus, HdR]

PhD Students

Bogdan Nicolae [MENRT Grant until September 30, 2010. Then, on a temporary ACET research position until December 31, 2010. PhD defended on November 30, 2010.]
Alexandra Carpen-Amarie [INRIA CORDI-S Grant]
Diana Moise [INRIA and Brittany Regional Council Grant]
Viet-Trung Tran [MENRT Grant]
Houssem-Eddine Chihoub [European Marie-Curie Scalus Project. Thesis started in October 2010.]

Visiting Scientists

Cătălin Leordeanu [PhD student, Polytechnic University of Bucharest, 3 months, supported by our bilateral contract]
Eliana Tîrşa [PhD student, Polytechnic University of Bucharest, 3 months, supported by our bilateral contract]
Alexandru Costan [PhD student, Polytechnic University of Bucharest, 3 months, supported by our bilateral contract]
Cristina Băsescu [Master student, Polytechnic University of Bucharest, 3 months, supported by the INRIA Internship program]
Sînziana Mazilu [Master student, Polytechnic University of Bucharest, 4 months, supported by the INRIA Internship program]

Administrative Assistant

Maryse Fouché [Secretary (TR) INRIA.]

Others

Tuan-Viet Dinh [Master Intern, ENS Cachan, supported by a grant of the ENS Cachan International Program]
Thi-Thu-Lan Trieu [Master Intern, ENS Cachan]

2. Overall Objectives

2.1. Introduction

More and more applications today generate and handle 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 of fields where it becomes crucial to efficiently manipulate massive data, which are typically *shared* at a large scale. With the emergence of the recent infrastructures (cloud computing platforms, post-Petascale architectures), achieving highly scalable data management is a critical challenge, as the overall application performance is highly dependent on the properties of the data management service.

2.1.1. Cloud data management

On Infrastructure-as-a-Service (IaaS) cloud infrastructures, computing resources are exploited on a per-need basis: instead of buying and managing hardware, users rent virtual machines and storage space. One important issue is thus the support for storing and processing data on externalized, virtual storage resources. Such needs require simultaneous investigation of important aspects related to performance, scalability, security and quality of service. Moreover, the impact of physical resource sharing also needs careful consideration.

2.1.2. Data management for Post-Petascale systems

In parallel with the emergence of cloud infrastructures, considerable efforts are now under way to build *Petascale computing systems*, such as Blue Waters (http://www.ncsa.illinois.edu/BlueWaters/). Such systems aim to provide sustained Petaflop performance to a much wider spectrum of science and engineering applications. On such infrastructures, data management is again a critical issue with a high impact on the application performance. Such supercomputers exhibit specific architectural features (e.g., a multi-level memory hierarchy scalable to tens to hundreds of thousands of cores) that are specifically designed to support a high degree of parallelism. In order to keep up with such advances, the storage service has to scale accordingly, which is clearly challenging.

Our research activities address the area of distributed data management at challenging scales on various distributed systems, with a particular focus on *clouds*, and *Post-Petascale infrastructures*. We target data-oriented high-performance applications that exhibit the need to handle massive non structured data - BLOBs: binary large objects (in the order of Terabytes) - stored in a large number of nodes (thousands to tens of thousands), accessed under heavy concurrency by a large number of clients (thousands to tens of thousands at a time) with a relatively fine access grain (in the order of Megabytes). Examples of such applications are:

- Cloud data-mining applications (e.g., based on the MapReduce paradigm) handling massive data distributed at a large scale.
- Advanced (e.g., concurrency-optimized, versioning-oriented) cloud services both for user-level data storage and for virtual machine image storage and management at IaaS level.
- Distributed storage for Petaflop computing applications.
- Data storage for desktop grid applications with high write throughput requirements.

2.2. Highlights

Team leadership. The KerData Team is led by G. Antoniu since July 2010. His mission is to submit a proposal to become a fully-fledged Project-Team within the year.

ANR Project with Argonne National Lab, UIUC and IBM. A new project, led by G. Antoniu, has been accepted by the ANR ARPEGE 2010 Program on embedded systems and large infrastructures. This project is devoted to using MapReduce programming paradigm on clouds and hybrid infrastructures.

INRIA-Microsoft Project. A new project, led by G. Antoniu and B. Thirion (Parietal Project-Team, INRIA SACLAY – ÎLE-DE-FRANCE), has started in collaboration with Microsoft Research. This project conducted within the framework the Microsoft Research - INRIA Joint Research Center involves Microsoft's *Azure* cloud computing platform.

New Associate Team created. A new Associate Team led by G. Antoniu (DataCloud@work, http://www.irisa.fr/kerdata/doku.php?id=cloud_at_work:start) was created in 2010, in partnership with the Politehnica University of Bucharest and with the MYRIADS INRIA Team.

Partnership with the INRIA-UIUC Joint Laboratory. We have set up a partnership with the INRIA-UIUC Joint Laboratory for Petascale Computing at Urbana-Champaign. Several mutual visits and internships were organized in this framework and numerous collaborations are on track in the context of the Blue Waters Project, expected to become one of the world's most powerful supercomputers when it comes online in 2011, with sustained Petaflop performance.

TCPP Best PhD Poster Award. It has been awarded to B. Nicolae at the IPDPS 2010 conference in Atlanta, GA, USA.

ENS-INRIA Prize of excellence. G. Antoniu and L. Bougé have initiated the proposal and creation of the ENS-INRIA Prize of excellence. It is targeted to Romanian high-school students who won the National Olympiad of Informatics. A first group of 5 students have been hosted in Rennes and Paris in June 2010. They visited the two sites of ENS Cachan (Cachan and Bruz) and two INRIA centers (INRIA RENNES – BRETAGNE ATLANTIQUE and INRIA PARIS – ROCQUENCOURT).

1 PhD thesis defended. The first PhD thesis of the KerData Team (B. Nicolae) was defended on 30 November 2010.

3. Scientific Foundations

3.1. Introduction

Managing data at large scales is paramount nowadays. Governmental and commercial statistics, climate modeling, cosmology, genetics, bio-informatics, etc. are just a few examples of fields routinely generating huge amounts of data. It becomes crucial to efficiently manipulate these data, which are typically shared at the global scale. In such a context, one important goal is to provide mechanisms allowing to manage massive data blocks (e.g., of several terabytes), while providing efficient fine-grain access to small parts of the data. Several application areas exhibit such a need for efficient scaling to huge data sizes: data mining applications [51], multimedia applications [41], database-oriented applications ([45], [64], [58]), bioinformatic applications, etc.

.

3.2. Transparent, distributed data sharing

The management of massive data blocks naturally requires the use of data fragmentation and of distributed storage. Grid infrastructures, typically built by aggregating distributed resources that may belong to different administration domains, were built during the last years with the goal of providing an appropriate solution. When considering the existing approaches to grid data management, we can notice that most of them heavily rely on *explicit* data localization and on *explicit* transfers of large amounts of data across the distributed architecture: GridFTP [34], LDR [25], Chirp [24], IBP [37], NeST [38], etc. Managing huge amounts of data in such an explicit way at a very large scale makes the design of grid application much more complex. One key issue to be addressed is therefore the *transparency* with respect to data localization and data movements. Such a transparency is highly suitable, as it liberates the user from the need to handle data localization and transfers.

Several approaches to grid data management acknowledge that providing a transparent data access model is important. They integrate this idea at the early stages of their design. *Grid file systems*, for instance, provide a familiar, file-oriented API allowing to transparently access physically distributed data through globally unique, logical file paths. The applications simply open and access such files as if they were stored on a local file system. A very large distributed storage space is thus made available to those existing applications that usually use file storage, with no need for modifications. This approach has been taken by a few projects like GFarm [63], GridNFS [49], LegionFS [67], etc.

On the other hand, the transparent data access model is equally defended by the concept of *grid data-sharing service* [35], illustrated for instance by the JuxMem platform [36]. Such a service provides the grid applications with the abstraction of a globally shared memory, in which data can be easily stored and accessed through global identifiers. To meet this goal, the design of JuxMem leverages the strengths of several building blocks: consistency protocols inspired by Distributed Shared Memory (DSM) systems; algorithms for fault-tolerant distributed systems; protocols for scalability and volatility support from peer-to-peer (P2P) systems.

3.3. Managing massive unstructured data under heavy concurrency on large-scale distributed infrastructures

3.3.1. Massive unstructured data: BLOBs

Studies show more than 80% [48] of data globally in circulation is unstructured. On the other hand, data sizes increase at a dramatic level with more than 1 TB of data gathered per week in common scenarios for

some production applications (e.g., medical experiments [61]). Finally, on Post-Petascale HPC machines, the use of huge storage objects is also currently being considered as a promising alternative to today's dominant approaches to data management. Indeed, these approaches rely on very large numbers of small files, and using huge storage objects reduces the corresponding metadata overhead of the file system. Such huge unstructured data are stored as *binary large objects (BLOBs)* that may continuously be updated by applications. However, traditional databases or file systems can hardly cope in an efficient way with BLOBs which grow to huge sizes.

3.3.2. Scalable processing of massive data: heavy access concurrency

To address the scalability issue, specialized abstractions like MapReduce [43] and Pig-Latin [59] propose high-level data processing frameworks intended to hide the details of parallelization from the user. Such platforms are implemented on top of huge object storage platforms. They target high performance by optimizing the parallel execution of the computation. This leads to *heavy access concurrency* to the BLOBs, thus the need for the storage layer to offer support in this regard. Parallel and distributed file systems also consider using objects for low-level storage (see next subsection [44], [66], [47]). In other application areas, huge BLOBs need to be used concurrently at the highest level layers of applications directly: high-energy physics, multimedia processing [41] or astronomy.

3.3.3. Versioning

When addressing the problem of storing and efficiently accessing very large unstructured data objects [55], [61] in a distributed environment, a challenging case is the one where data is *mutable* and potentially accessed by a very large number of concurrent, distributed processes. In this context, *versioning* is an important feature. Not only it allows to roll back data changes when desired, but it also enables cheap branching (possibly recursively): the same computation may proceed independently on different versions of the BLOB. Versioning should obviously not impact access performance to the object significantly, given that objects are under constant heavy access concurrency. On the other hand, versioning leads to increased storage space usage and becomes a major concern when the data size itself is huge. Versioning efficiency thus refers to both access performance under heavy load and reasonably acceptable overhead of storage space.

3.4. Towards scalable, BLOB-based distributed file systems

Recent research [46] emphasizes a clear move currently in progress from a block-based interface to a object-based interface in storage architectures. The goal is to enable scalable, self-managed storage networks by moving low-level functionalities such as space management to storage devices or to storage server, accessed through a standard object interface. This move has a direct impact on the design of today's distributed file systems: object-based file system would then store data rather as objects than as unstructured data blocks. According to [46], this move may eliminate nearly 90% of management workload which was the major obstacle limiting file systems' scalability and performance.

Two approaches exploit this idea. In the first approach, the data objects are stored and manipulated directly by a new type of storage device called *object-based storage device* (OSD). This approach requires an evolution of the hardware, in order to allow high-level object operations to be delegated to the storage device. The standard OSD interface was defined in the Storage Networking Industry Association (SNIA) OSD working group. The protocol is embodied over SCSI and defines a new set of SCSI commands. Recently, a second generation of the command set, Object-Based Storage Devices - 2 (OSD-2) has been defined. The distributed file systems taking the OSD approach assume the presence of such an OSD in the near future and currently rely on a software module simulating its behavior. Examples of parallel/distributed file systems following this approach are Lustre [62] and Ceph [66]. Recently, research efforts [44] have explored the feasibility and the possible benefits of integrating OSDs into parallel file systems, such as PVFS [40].

The second approach does not rely on the presence of OSDs, but still tries to benefit from an object-based approach to improve performance and scalability: files are structured as a set of objects that are stored on storage servers. Google File System [47], and HDFS (*Hadoop File System*) [28] illustrate this approach.

3.5. Emerging large-scale infrastructures for distributed applications

During the last few years, research and development in the area of large-scale distributed computing led to the clear emergence of several types of physical execution infrastructures for large-scale distributed applications.

3.5.1. Cloud computing infrastructures

The cloud computing model [65], [54], [39] is gaining serious interest from both industry and academia in the area of large-scale distributed computing. It provides a new paradigm for managing computing resources: instead of buying and managing hardware, users rent virtual machines and storage space. Various cloud software stacks have been proposed by leading industry companies, like Google, Amazon or Yahoo!. They aim at providing fully configurable virtual machines or virtual storage (IaaS: *Infrastructure-as-a-Service*), higher-level services including programming environments such as MapReduce [43] (*PaaS: Platform-as-a-Service* [26], [29]) or community-specific applications (*SaaS: Software-as-a-Service* [27], [30]). On the academic side, two of the most visible projects in this area are Nimbus [31], [52] from the Argonne National Lab (USA) and OpenNebula [32], which aim at providing a reference implementation for a IaaS. In parallel to these trends, other research efforts focused on the concept of grid operating system: a distributed operating system for large-scale wide-area dynamic infrastructure spanning multiple administrative domains. XtreemOS [57], [33] is such a grid operating system, which provides native support for virtual organizations. Since both the cloud approach and the grid operating system approach deal with resource management on large-scale distributed infrastructures, the relative positioning of these two approaches with respect to each other are currently subject to on-going investigation within the PARIS/MYRIADS Project-Team (http://www.irisa.fr/myriads/) at INRIA RENNES – BRETAGNE ATLANTIQUE [56].

In the context of the emerging cloud infrastructures, some of the most critical open issues relate to data management. Providing the users with the possibility to store and process data on externalized, virtual resources from the cloud requires simultaneously investigating important aspects related to security, efficiency and quality of service. To this purpose, it clearly becomes necessary to create mechanisms able to provide feedback about the state of the storage system along with the underlying physical infrastructure. The information thus monitored, can further be fed back into the storage system and used by self-managing engines, in order to enable an autonomic behavior [53], [60], [50], possibly with several goals such as self-configuration, self-optimization, or self-healing. Exploring ways to address the main challenges raised by data storage and management on cloud infrastructures is the major factor that motivated the creation of the KerData research team INRIA RENNES – BRETAGNE ATLANTIQUE. These topics are at the heart of our involvement in several projects that we are leading in the area of cloud storage: MapReduce (see Section 8.2), AzureBrain (see Section 7.1), DataCloud@work (see Section 8.3).

3.5.2. Petascale infrastructures

In 2011, a new NSF-funded Petascale computing system, Blue Waters, will go online at the University of Illinois. Blue Waters is expected to be the most powerful supercomputer in the world for open scientific research when it comes online. It will be the first system of its kind to sustain one-Petaflop performance on a range of science and engineering applications. The goal of this facility is to open up new possibilities in science and engineering. It provides unheard computational capability. It makes it possible for investigators to tackle much larger and more complex research challenges across a wide spectrum of domains: predict the behavior of complex biological systems, understand how the cosmos evolved after the Big Bang, design new materials at the atomic level, predict the behavior of hurricanes and tornadoes, and simulate complex engineered systems like the power distribution system and airplanes and automobiles.

To reach sustained-Petascale performance, machines like Blue Waters relies on advanced, dedicated technologies at several levels: processor, memory subsystem, interconnect, operating system, programming environment, system administration tools. In this context, data management is again a critical issue that highly impacts the application behavior and its overall performance. Petascale supercomputers exhibit specific architectural features (e.g., a multi-level memory hierarchy scalable to tens to hundreds of thousands of codes) that needs to be specifically taken into account. Providing scalable data throughput on such unprecedented scales is clearly

an open challenge today. In this context, we are investigating techniques to achieve concurrency-optimized I/O in collaboration with teams from the National Center for Supercomputing Applications (NCSA/UIUC) in the framework of the Joint INRIA-UIUC for Petascale Computing (see Section 8.4).

3.5.3. Desktop grids

During the recent years, Desktop grids have been extensively investigated as an efficient way to build cheap, large-scale virtual supercomputers by gathering idle resources from a very large number of users. A possible approach is to rely on clusters of workstations belonging to institutions and interconnected through dedicated, high-throughput wide-area interconnect, which is the typical physical infrastructure for Grid Computing. In contrast, Desktop grids rely on desktop computers from individual users, interconnected through Internet, provided by *volunteer users*. The initial, widely-spread usage of Desktop grids for parallel applications consisting in non-communicating tasks with small input/output parameters is a direct consequence of the physical infrastructure. Actually, volatile nodes and low bandwidth are not suitable for communication-intensive parallel applications with high input or output requirements. However, the increasing popularity of volunteer computing projects has progressively lead to enlarge the set of application classes that might benefit of Desktop Grid infrastructures. If we consider distributed applications where tasks need very large input data, it is no longer feasible to rely on regular centralized server-based Desktop Grid architectures. Actually, the input data is there typically embedded in the job description and sent to workers. Such a strategy could lead to significant bottlenecks as the central server gets overwhelmed by download requests. To cope with such data-intensive applications, alternative approaches based on P2P techniques and Content Distribution Networks [42] have been proposed, with the goal of offloading the transfer of the input data from the central servers to the other nodes participating to the system, with potentially under-used bandwidth.

In the general case, Desktop Grids rely on resources contributed by volunteers. Enterprise Desktop Grids are a particular case of Desktop Grids leveraging unused processing cycles and storage space available within the enterprise. The emergence of cloud infrastructures has opened new perspectives to the development of Desktop Grids, as new types of usage may benefit from a *hybrid*, simultaneous use of these two types of infrastructures. In a typical scenario of this kind, an enterprise would not use dedicated, on-site hardware resources for a particular need for data-intensive analysis, e.g., to process commercial statistics. It would rather rely on free unused internal resources using the Enterprise Desktop Grid model, and, in extension to them, would rent resources from the cloud. Both architectures are suitable for massively parallel processing and this is why we intend to explore the potential advantages of using such hybrid infrastructures in the framework of the MapReduce project (see Section 8.2).

3.6. Emerging programming models for scalable data-management

MapReduce is a parallel programming paradigm successfully used by large Internet service providers to perform computations on massive amounts of data. A computation takes a set of input key/value pairs, and produces a set of output key/value pairs. The user of a MapReduce library expresses the computation as two functions: *map*, that processes a key/value pair to generate a set of intermediate key/value pairs, and *reduce*, that merges all intermediate values associated with the same intermediate key. The framework takes care of splitting the input data, scheduling the jobs' component tasks, monitoring them and re-executing the failed ones. After being strongly promoted by Google, it has also been implemented by the open source community through the Hadoop project, maintained by the Apache Foundation and supported by Yahoo! and even by Google itself. This model is currently getting more and more popular as a solution for rapid implementation of distributed data-intensive applications. The key strength of the MapReduce model is its inherently high degree of potential parallelism that should enable processing of Petabytes of data in a couple of hours on large clusters consisting of several thousand nodes.

At the core of the MapReduce frameworks stays a key component: the storage layer. To enable massively parallel data processing to a high degree over a large number of nodes, the storage layer must meet a series of specific requirements. Firstly, since data is stored in huge files, the computation will have to efficiently process small parts of these huge files concurrently. Thus, the storage layer is expected to provide efficient *fine-grain*

access to the files. Secondly, the storage layer must be able to sustain a *high throughput* in spite of *heavy access concurrency* to the same file, as thousands of clients simultaneously access data.

These critical needs of data-intensive distributed applications have not been addressed by classical, POSIX-compliant distributed file systems. Therefore, specialized file systems have been designed, such as HDFS, the default storage layer of Hadoop. HDFS has however some difficulties in sustaining a high throughput in the case of concurrent accesses to the same file. Amazon's cloud computing initiative, Elastic MapReduce, employs Hadoop on their Elastic Compute Cloud infrastructure (EC2) and inherits these limitations. The storage back-end used by Hadoop is Amazon's Simple Storage Service (S3), which provides limited support for concurrent accesses to shared data. Moreover, many desirable features are missing altogether, such as the support for versioning and for concurrent updates to the same file. Finally, another important requirement for the storage layer is its ability to expose an interface that enables the application to be *data-location aware*. This is critical in order to allow the scheduler to use this information to place computation tasks close to the data and thus reduce network traffic, contributing to a better global data throughput. These topics are at the core of KerData's contribution to the MapReduce ANR project and to the Hemera large wingspan project (both started in 2010, see Section 8.2).

4. Application Domains

4.1. Introduction

The research carried out within the KerData team targets applications that handle massive data that are fragmented, distributed, shared and accessed under heavy concurrency at a large scale.

- Massively parallel data-mining applications (e.g., MapReduce-based data analysis).
- Advanced PaaS-level cloud data services requiring efficient data sharing under heavy concurrency.
- I/O-intensive scientific simulations for Post-Petascale infrastructures.
- Desktop grid applications with high write throughput requirements.

In the current projects started in 2010 we specifically work on providing concurrency-optimized data storage and management for the following applications.

4.2. Structural protein analysis on clouds based on MapReduce: SuMo and MED-SuMo

In the framework of the MapReduce ANR project lead by KerData (started in October 2010) we will validate our techniques for concurrency-optimized data management with an application study from the bioinformatics field. It will focus on the SuMo application proposed by Institute for Biology and Chemistry of the Proteins from Lyon (a partner of the MapReduce project). This application performs structural protein analysis by comparing a set of protein structures against a very large set of structures stored in a huge database. This is a typical data-intensive application that can leverage the MapReduce model for a scalable execution on large-scale distributed platforms.

If the results are convincing, then they can immediately be applied to the derivative version of this application for drug design in industrial context called MED-SuMo, managed by the MEDIT SME (also a partner of this project). Regarding pharmaceutical and biotech industries, such a scalable implementation run over a cloud computing facility opens new perspectives for drug design. Rather than searching for 3D similarity into biostructural data, it will become possible to classify the entire biostructural space and to periodically update all derivative predictive models with new experimental data. The applications of that complete chemo-proteomic vision address the identification of new druggable protein target, the detection of new allosteric binding site suitable to increase the selectivity of a drug compound, the generation of new drug candidates by a fragment-based approach over protein-ligand biostructural data, and other new protocols under development at MEDIT.

4.3. Joint genetic and neuroimaging data analysis on clouds

The AzureBrain Project started in October 2010 within the Microsoft Research-INRIA Joint Research Center. In this framework, we focus on a data-analysis application whose goal is to find statistically relevant correlations across two huge sets containing genetic data and neuroimaging data respectively, for large cohorts of subjects. In the genome dimension, genotyping DNA chips allow to record several hundreds of thousands of values per subject, whereas in the imaging dimension a fMRI volume may contain hundreds of thousands to millions of voxels. Finding the brain and genome regions that may be involved in this link entails a huge number of hypotheses, hence a drastic correction of the statistical significance of pairwise relationships, which in turn crucially reduces the sensitivity of statistical procedures that aims at detecting the association.

We collaborate with the PARIETAL team from INRIA SACLAY – ÎLE-DE-FRANCE, who works on such optimized techniques for joint genetic and neuroimaging analysis. We plan to redesign the application using a cloud-oriented programming model such as MapReduce, and then to adapt and evaluate the whole software stack (application, programming engine, BlobSeer-based storage components) on Microsoft's Azure platform. The input application data will be taken from the Imagen FP6 project (http://www.imagen-europe.com/) that aims at investigating factors of addiction in a population of adolescents; Imagen's database contains multi-modal neuroimaging as well as genetics and psychological data on more than 1000 (possibly 2000 within a few years) subjects. This database is hosted and processed at Neuropsin and is available to the grant partners for research purpose.

4.4. I/O intensive tornado simulation for the Blue Waters post-Petascale machine

The Blue Waters machine (http://www.ncsa.illinois.edu/BlueWaters/) is expected to be one of the most powerful supercomputers in the world when it comes online in 2011. It will have a peak performance of 10 Petaflops (10 quadrillion calculations every second) and will achieve *sustained* performance of 1 Petaflop running a range of science and engineering codes. Research at the Joint INRIA-UIUC (University of Illinois at Urbana-Champaign) Lab for Petascale computing (JLPC) is currently in progress in several directions, with the global goal of efficiently exploiting this machine that will serve to run heavy, data-intensive or computation-intensive simulations.

Such simulations usually require to be coupled with visualization tools. On supercomputers, previous studies already showed the need of adapting the I/O path from data generation to visualization. In the framework of the JLPC we started to investigate concurrency-optimized I/O techniques to achieve this goal. We focus on a particular tornado simulation called CM1, which is intended to be run on the BlueWaters machine. This simulation currently generates large amount of data in many files, in a way that is not adapted for later visualization. We started to explore the use of BlobSeer, a large-scale data management service designed by the KerData team, as an intermediate layer between the simulation, the filesystem and visualization tools. Concurrency control optimizations enabled by BlobSeer will be tuned to ensure efficient access to the files managed by the underlying file system. A preliminary study done by Matthieu Dorier (Master student at ENS Cachan - Brittany) during a 3-month internship at UIUC, co-advised by Marc Snir, Franck Cappello and G. Antoniu, has demonstrated the benefits of a new approach using dedicated I/O cores (see Section 8.4).

5. Software

5.1. BlobSeer

Participants: Gabriel Antoniu, Luc Bougé, Bogdan Nicolae.

Contacts: `Bogdan.Nicolae@inria.fr, Gabriel.Antoniu@inria.fr`

URL: http://blobseer.gforge.inria.fr/

License: GNU Lesser General Public License (LGPL) version 3.

Status: This software is available on INRIA's forge. Registration of version 1.0 (released late 2010) with APP is in progress.

Presentation: BlobSeer is a data storage service specifically designed to deal with the requirements of large-scale data-intensive distributed applications that abstract data as huge sequences of bytes, called BLOBs (Binary Large OBjects). It exports a simple, yet versatile versioning interface to manipulate BLOBs that enables reading, writing and appending to them. BlobSeer offers both scalability and performance with respect to a series of issues typically associated with the data-intensive context: *scalable aggregation of storage space* from the participating nodes with minimal overhead, ability to store *huge data objects*, *efficient fine-grain access* to data subsets, *high throughput in spite of heavy access concurrency*, as well as *fault-tolerance*.

Development started in January 2008. The implementation is built on top of the Boost collection of C++ libraries, Berkeley DB and libconfig. Additional scripting in Perl/Python handles deployment on GRID'5000, which is done through the *OAR* resource scheduler. Benchmarking so far has proven correctness and scalable performance with up to 400 nodes from 3 different sites.

The latest stable version of BlobSeer, v1.0, brings a large set of new features and improvements whose usefulness was experimentally validated during the course of 2010. Of particular importance to the user are two new features: (1) the support to efficiently clone BLOBs by using a new, dedicated primitive that was added to the access interface; and (2) a POSIX access interface to BLOBs (implemented over FUSE) that enables applications to access BLOBs using standard I/O calls, while retaining the ability to perform BLOB-specific manipulations (such as access to past versions and cloning) through *ioctls*.

6. New Results

6.1. BlobSeer

Participants: Bogdan Nicolae, Gabriel Antoniu, Luc Bougé, Diana Moise, Alexandra Carpen-Amarie.

Several contributions were achieved that relate directly to the core functionality of BlobSeer.

First, we refined the design principles behind BlobSeer and placed them in the context of scalable distributed storage systems: if combined together, these principles can help designers of distributed storage systems to meet the need for highly scalable data management. In particular, we focused on the potentially large benefits of using versioning to improve application data access performance under heavy concurrency. In this context, we extended the versioning-based access interface of BlobSeer with new primitives that further enhance the potential to exploit the inherent parallelism of data workflows efficiently.

Second, we proposed a generalization for a set of versioning algorithms for data management originally implemented in BlobSeer and published in the previous years. We have introduced new data structures and redesigned several aspects to account for better decentralized metadata management, fine-grain access at arbitrary offsets, asynchrony, fault tolerance and last but not least allow the user to explicitly control written data layout such that it is optimally distributed for reading.

Third, we extended the scope of our experimental evaluation and performed synthetic benchmarks that push the system to its limits. It demonstrated a high throughput under heavy access concurrency, even when metadata is replicated in order to provide fault tolerance. Furthermore, we extended the evaluation of BlobSeer as a storage back-end for Hadoop MapReduce and highlighted a series of improvements in the context of MapReduce data-intensive applications.

These contributions materialized in a reference publication [7] about BlobSeer that provides a complete view over its design principles, algorithms, consistency and fault tolerance considerations, as well as experimental evaluations. A more compact overview of BlobSeer [14] was also published in the PhD Forum of IPDPS'10, where the corresponding poster, presented during the conference, won the TCPP Best PhD Student Poster Award.

Complementary to these results, further work was undertaken to improve the usability of BlobSeer in the context of cloud computing. More specifically, we evaluated the trade-off resulting from transparently applying data compression to save storage space and bandwidth at the cost of slight computational overhead. The aim is to reduce the storage space and bandwidth needs with minimal impact on I/O throughput when under heavy access concurrency. To this end, we introduced a generic sampling-based compression technique that dynamically adapts to the heterogeneity of data and applied it to BlobSeer. It led to significant improvement over the original implementation: almost no performance overhead when dealing with incompressible data, as well as significant saving in storage space and bandwidth for compressible data, with the added benefit of improved aggregated read throughput. These results were obtained as a consequence of extensive experiments on the Grid'5000 testbed and were published in [15].

Finally, B. Nicolae successfully defended his PhD thesis on November 30, 2010. The thesis document details the contributions that relate to the core of BlobSeer since the beginning of the project.

6.2. MapReduce

Participants: Diana Moise, Bogdan Nicolae, Gabriel Antoniu, Luc Bougé, Thi-Thu-Lan Trieu.

The features exhibited by BlobSeer meet the storage needs of MapReduce applications. To evaluate the benefits of using BlobSeer as the storage back-end in such a context, we used Hadoop - Yahoo!'s implementation of the MapReduce framework. We substituted the original data storage layer of Hadoop, the *Hadoop Distributed File System* (HDFS) with our BlobSeer-based file system - BSFS. To measure the impact of our approach, we performed experiments both with synthetic microbenchmarks and real MapReduce applications. The results showed that BSFS is capable to deliver a higher throughput than HDFS, and to sustain it when the number of clients significantly increases. This work on integrating BlobSeer with Hadoop [13] brought up various issues that could be improved in the Hadoop framework [11].

One of these aspects concerns the append operation for which HDFS does not offer support. In [10] we show how providing the functionality of concurrently appending data to existing files, can bring substantial benefits to MapReduce applications as well as to other classes of applications. Since BlobSeer efficiently supports concurrent appends, we modified the Hadoop MapReduce framework to use the append operation in the "reduce" phase of the application. Our experiments showed that massively concurrent append and read operations have a low impact on each other; furthermore, measurements with an application available with Hadoop showed that the support for concurrent appends to shared files is introduced with no extra cost, whereas the number of files managed by the MapReduce framework is substantially reduced.

We also addressed the problem of managing intermediate data, which is data generated during MapReduce computations. In the original Hadoop MapReduce framework, intermediate data (data produced as output of the "map" phase and transferred as input to the "reduce" phase) is stored on the local file system of the machines executing the "map" function; in case of failures, the data is lost and the map computation is re-executed on another machine. Our approach was to store the intermediate data in a distributed file system, so that, when a failure occurs, the computation can resume on another machine; moreover, by using BSFS as storage for intermediate data, the execution time is reduced due to the high throughput BSFS delivers. These issues have been developed with the Master thesis of Lan Trieu [23].

6.3. Introspective BlobSeer and security

Participants: Alexandra Carpen-Amarie, Alexandru Costan, Gabriel Antoniu, Luc Bougé, Cătălin Leordeanu, Cristina Băsescu.

This work has been done in collaboration with Jing (Tylor) Cai, Master student at the City University of Hong Kong, and Mihaela-Camelia Vlad, Master student at the Polytechnic University of Bucharest. Both of them visited the KerData Team in 2009–2010 for several months, supported by the INRIA Internship program.

The goal of this research direction is to enable autonomic storage for BlobSeer-based cloud services. This work has been carried out in the framework of the DataCloud@work Associated Team between KerData and the Computer Science Department from Politehnica University of Bucharest - PUB (http://www.irisa.fr/kerdata/doku.php?id=cloud_at_work:start).

The first step towards an autonomic data-sharing system was to equip the BlobSeer platform with introspection capabilities, which can serve as input data for a self-adaptive engine deployed on top of the system, possibly with several goals such as self-configuration, self-optimization, self-healing or self-protection. This work has been published in [9].

Further, we implemented a distributed architecture for storing and processing monitoring data. Our solution was designed as a new BlobSeer component that does not interfere with its efficient data-access primitives. Instead, it builds a distributed user-activity history to obtain real-time information about the users in the system. Then we proposed a preliminary approach for enabling self-protection for the BlobSeer system, through a malicious client detection component, which analyzes protocol breaches specific to BlobSeer. These results have been published as INRIA research reports [21], [16].

We developed the self-protection direction within a generic security management framework allowing providers of Cloud data management systems to define and enforce complex security policies. In addition, we designed an expressive policy description language so as to be able to define a wide range of security attacks and to detect them in a security violation detection engine. We integrated our security framework with BlobSeer and we showed that we can provide a secure environment for data management systems without any significant overhead, while being able to define and detect complex attack scenarios. These results have been published in [8].

Moreover, we developed a specific security mechanism which assigns a trust level to each client by continually monitoring and analyzing the client activity and the state of the system to detect security threats, malicious activity or other kinds of intrusions. Additionally, we addressed the problem of securely running web services on top of BlobSeer. We implemented mechanisms that handle authentication and authorization of the users, as well as secure data transfers for web services that use BlobSeer as a storage back-end.

Another direction was to introduce self-management and self-adaptation facilities in BlobSeer. We enhanced BlobSeer with self-adaptive features by dynamically changing and maintaining the replication factors of the data. When a specific BLOB is under a heavy load (in terms of read operations), the system automatically increases its replication factor and handles all the necessary data transfers. In contrast, when some data is less (or never) used, its replication factor is transparently reduced. Moreover, we developed a component able to dynamically contract and expand the pool of storage providers based on the system's load, so as to adapt the resource usage to the needs of the clients accessing the data. Several Master research internships and Bachelor theses at PUB focused on these tasks.

6.4. Concurrency-optimized I/Os for Petascale computing

Participants: Viet-Trung Tran, Gabriel Antoniu, Bogdan Nicolae, Luc Bougé.

This work has been done in collaboration with Matthieu Dorier, student at ENS Cachan, Brittany Campus, during his summer 2010 internship at the INRIA-UIUC Joint Laboratory for Petascale Computing (JLPC) at Urbana-Champaign.

High-performance concurrent I/O accesses are a major requirement of data-intensive scientific applications, particularly for those applications deployed on Petascale infrastructures. The larger the scale of the execution infrastructure, the higher the potential performance bottlenecks that could be caused by a lack of performance of the data input/output (I/O) layers. We focused on specific scenarios that exhibit the need for efficient access to huge, shared data under heavy concurrency workload. We identified two main issues that require closer consideration.

First, there is still a trade-off between high-performance data communication and atomic I/O capabilities of concurrent overlapped updates in the context of scientific applications. Current lock-based approaches mainly perform locking around the operations, imposing lock overhead and slowing down the overall performance. In this context, we aim to exploit the potential benefits of BlobSeer. By leveraging a versioning-based scheme, an atomic I/O operation is expected to be done in a lock-free manner, even when overlapped accesses occur. Following this direction, we conducted several experimental evaluations on Grid'5000 and obtained very promising results described in [20].

In the second direction, our major research topic comes from the context of HPC, and targets scientific simulations running on Petascale machines. The goal is to explore how to efficiently record and visualize data during the simulation without impacting the performance of the corresponding computation generating that data. Conventional practice of storing data on disk, moving it off-site, reading it into a workflow, and analyzing it to produce scientific data becomes increasingly harder to use, due to large data volumes generated at fast rates compared to limited back-end speeds. Therefore, scalable approaches to deal with these I/O limitations are of utmost importance. We propose to adapt concurrency control techniques introduced in BlobSeer in order to optimize the level of parallelization between visualization and simulation with respect to I/O. It allows periodic data backup and online visualization to proceed without blocking computation, and vice versa.

A first step has been taken in this direction by studying the behavior, with respect to I/O, of a tornado simulation code called CM1, targeting the next IBM supercomputer BlueWaters. This behavior induces large overheads due to the generation of many small files at the same time. We proposed a first solution using dedicated I/O cores as staging areas in order to overlap I/O with computation at the simulation level. Such a solution has demonstrated to be capable of bringing a better balance in throughput and to avoid overheads in I/O phases, as well as an ability to perform efficiently data preprocessing. Coupled with the BlobSeer approach, we intend to provide a full solution for efficiently coupling simulations with visualization tools for very large scales. This work has been initiated during Matthieu Dorier's master intership at JLPC.

6.5. BlobSeer-based management of virtual machines in Nimbus

Participants: Bogdan Nicolae, Alexandra Carpen-Amarie, Tuan-Viet Dinh, Eliana Tîrşa, Gabriel Antoniu.

Providing efficient virtual machine image storage solutions is crucial in the context of Infrastructure-as-a-Service (IaaS) cloud computing, as users rent resources in terms of virtual machines that are instantiated from virtual machine images. One of those challenges in this context is the need to deploy a large number (hundreds or even thousands) of VM instances simultaneously. Once the VM instances are deployed, another challenge is to simultaneously take a snapshot of many images and transfer them to persistent storage to support management tasks, such as suspend-resume and migration.

During a 2-month visit at Argonne National Lab, USA, B. Nicolae adapted BlobSeer to address these needs. More specifically, a series of optimization techniques were proposed that minimize resource consumption (execution time, network traffic and storage space) which translate into lower end-user costs. While conventional approaches transfer the whole VM image contents between the persistent storage service and the computing nodes, we proposed a lazy transfer scheme based on object-versioning that transfers only the needed content on-demand: this greatly reduces total time for execution time, network traffic and storage space. The benefits of this approach were demonstrated through extensive experiments operating on hundreds of nodes, showing improvements in time to boot virtual machines from a shared image by a factor of up to 25, while at the same time reducing storage and bandwidth usage by as much as 90% when compared with conventional approaches. This work is described in [19].

Furthermore, the cloud users need mechanisms to upload Virtual Machine (VM) images into a Cloud storage service, before they are deployed to the physical nodes. We investigated this issue for the Nimbus Cloud environment, by replacing its default repository with Blobseer. This work has been published in [17].

In the context of the Associated Team between KerData and the Computer Science Department from Politehnica University of Bucharest, we made available BlobSeer as a storage service on the Cloud, by integrating it within the Nimbus Cloud. We added mechanisms for bringing BlobSeer to a consistent state before

stopping it and then for starting/stopping/restarting BlobSeer inside the Nimbus Cloud, while preserving the data it stored during previous runs. Additionally, we investigated the advantages of using BlobSeer as a storage system for XtreemOS, by conducting a series of performance evaluations targeted towards MapReduce applications. We experimented with Hadoop applications deployed on top of HDFS, BlobSeer and XtreemFS, the default file system of XtreemOS.

6.6. Using Global Behavior Modeling to Improve QoS in Cloud Data Storage Services

Participants: Bogdan Nicolae, Houssem-Eddine Chihoub, Gabriel Antoniu, Alexandra Carpen-Amarie.

MapReduce is emerging as a highly scalable programming paradigm that enables high-throughput data-intensive processing as a cloud service. However, the associated performance is highly dependent on the underlying storage service, responsible to efficiently support massively parallel data accesses by guaranteeing a high throughput under heavy access concurrency. In this context, quality of service plays a crucial role: the storage service needs to sustain a stable throughput regarding each access individually, in addition to achieving a high aggregated throughput under concurrency.

We propose [12] a technique to address this problem using component monitoring, application-side feedback and behavior pattern analysis. It allows to automatically infer useful knowledge about the causes of a poor quality of service, and to provide an guidelines toward potential improvements. We apply our proposal to BlobSeer, as a representative data storage service specifically designed to achieve high aggregated throughputs. Through an extensive experimentation, we demonstrated substantial improvements in the stability of individual data read accesses under MapReduce workloads. Within the SCALUS Marie-Curie project (see Section 8.3) we plan to refine this work using the OpenNebula as a IaaS cloud environment.

7. Contracts and Grants with Industry

7.1. AzureBrain: INRIA-Microsoft project

Participants: Gabriel Antoniu, Luc Bougé.

Joint genetic and neuroimaging data analysis on large cohorts of subjects is a new approach used to assess and understand the variability that exists between individuals. This approach has remained poorly understood so far and brings forward very significant challenges, as progress in this field can open pioneering directions in biology and medicine. As both neuroimaging- and genetic-domain observations represent a huge amount of variables (of the order of 106), performing statistically rigorous analyses on such amounts of data represents a computational challenge that cannot be addressed with conventional computational techniques. This project started in October 2010 for two years in the framework of the Microsoft Research - INRIA Joint Research Center and aims to explore cloud computing techniques to address the above computational challenge. The project will rely on Microsoft's Azure cloud platform and will leverage the complementary expertise of two INRIA teams: KerData (Rennes) in the area of scalable cloud data management and PARIETAL (Saclay) in the field of neuroimaging. For more details, see the official press release http://www.microsoft.com/france/espace-presse/communiques-de-presse/fiche-communique.aspx?EID=75da32ee-5ed3-42b2-a847-4971f716df31.

8. Other Grants and Activities

8.1. Regional intiatives

8.1.1. PhD grant

Participant: Diana Moise.

The Brittany Regional Council provides half of the financial support for the PhD thesis of D. Moise (GRID5000BD project). This support amounts to a total of around 14,000 Euros/year. This support ends in September 2011.

8.2. National initiatives

8.2.1. MapReduce: an ANR project with international partners

Participants: Gabriel Antoniu, Luc Bougé, Bogdan Nicolae, Alexandra Carpen-Amarie, Diana Moise, Houssem-Eddine Chihoub.

KerData is leading the MapReduce project (October 2010 – March 2014) funded by the ANR ARPEGE 2010 Program on embedded systems and large infrastructures. This project is devoted to using MapReduce programming paradigm on clouds and hybrid infrastructures. It started in October 2010 in partnership with Argonne National Lab (USA), the University of Illinois at Urbana Champaign (USA), the UIUC-INRIA Joint Lab on Petascale Computing, IBM France, IBCP, MEDIT (SME) and the GRAAL INRIA project-team. In this project we explore advanced techniques for scalable, high-throughput, concurrency-optimized data and metadata management. Recent preliminary experiments with the BlobSeer storage platform designed by the KerData have shown substantial potential improvements of the data throughput compared to Hadoop, which acts as today's reference MapReduce platform.

8.2.2. Hemera: an Inria large-wingspan project

Participants: Gabriel Antoniu, Diana Moise.

Hemera (http://www.grid5000.fr/mediawiki/index.php/Hemera) is an INRIA Large Wingspan project, started in 2010. (Hemera is the Greek goddess of the daytime, http://en.wikipedia.org/wiki/Hemera.) It aims to demonstrate ambitious up-scaling techniques for large scale distributed computing by carrying out several dimensioning experiments on the Grid'5000 infrastructure. It also aims to animate the scientific community around Grid'5000 and to enlarge the Grid'5000 community by helping newcomers to make use of Grid'5000. It is not restricted to INRIA teams. Within Hemera, G. Antoniu (KerData INRIA team) and Gilles Fedak (GRAAL INRIA project-team) co-lead the MapReduce scientific challenge, whose goal is to carry out scalable experiments with MapReduce applications on Grid'5000. KerData is also involved in a working group called *Efficient management of very large volumes of information for data-intensive applications*, co-led by G. Antoniu with Jean-Marc Pierson (IRIT, Toulouse).

8.3. European initiatives

8.3.1. SCALUS: Marie Curie Initial Training Network (FP7)

Participants: Gabriel Antoniu, Houssem-Eddine Chihoub, Bogdan Nicolae, Alexandra Carpen-Amarie.

The SCALUS Marie Curie Initial Training Network (http://www.scalus.eu) project aims at elevating education, research, and development inside the area of large-scale, distributed ubiquitous storage with a focus on cluster, grid, and cloud storage. The vision of this MCITN is to deliver the foundation for ubiquitous storage systems, which can be scaled in arbitrary directions (capacity, performance, distance, security, etc.) The consortium's goal is to build the first interdisciplinary teaching and research network on storage issues. It consists of top European institutes and companies in storage and cluster technology, building a demanding but rewarding interdisciplinary environment for young researchers. This interdisciplinary research consortium is the foundation for young researchers to be able to perform the innovative research tasks outlined in this proposal. The academic partners include INRIA RENNES – BRETAGNE ATLANTIQUE, Universidad Politécnica de Madrid, Barcelona Supercomputing Center, University of Paderborn, Ruprecht-Karls-Universität Heidelberg, Durham University, FORTH, École des Mines de Nantes, XLAB, CERN, NEC, Microsoft Research, Fujitsu, Sun Microsystems. The project started on December 1, 2009, and it is lasting for 4 years. It involves the KerData and the MYRIADS Teams. G. Antoniu serves as a coordinator for INRIA RENNES – BRETAGNE ATLANTIQUE.

Two PhD parallel theses funded by the SCALUS Project are co-advised by G. Antoniu (KerData) and María Pérez (Universidad Politécnica de Madrid, UPM). Both started in September 2010: Houssem-Eddine Chihoub, hosted by KerData, and Bunjamin Memishi, hosted at UPM. Both theses will explore ways to continue the preliminary joint work started by our teams involving BlobSeer and GloBeM (see Section 6.6) in the framework of real cloud infrastructures, with real applications. Discussions and preliminary experiments are in progress on how the OpenNebula cloud toolkit developed at Universidad Complutense de Madrid could be used as a global framework for this work.

8.3.2. DataCloud@Work: INRIA's Associate Team Programme

Participants: Gabriel Antoniu, Luc Bougé, Alexandra Carpen-Amarie, Alexandru Costan, Diana Moise, Bogdan Nicolae, Cătălin Leordeanu, Eliana Tîrşa, Cristina Băsescu, Sînziana Mazilu.

DataCloud@work was initiated in 2010 by G. Antoniu (KerData) as an Associate Team in partnership with Politehnica University of Bucharest (PUB) and the MYRIADS Team (INRIA RENNES – BRETAGNE ATLANTIQUE). It aims to investigate ways to provide advanced, autonomic storage mechanisms for cloud services. More specifically, the goal is to explore how to build an efficient, secure and reliable storage service for data-intensive distributed applications running in cloud environments by enabling an autonomic behavior. A secondary goal is to leverage the grid operating system approach as a cloud technology (e.g., by relying on its OS-support for virtual organizations). The project builds on preliminary prototypes: the BlobSeer data-sharing platform (designed by the KerData Team), on the MonALISA monitoring framework (whose main technical contributor is the PUB Team), and on the XtreemOS grid operation system (designed under the leadership of the MYRIADS Team). This work uses as a framework the Nimbus cloud toolkit from Argonne National Lab.

In 2010 we addressed the following topics: 1) Introduce of self-adaptation capabilities in BlobSeer, based on the MonALISA monitoring framework; 2) Design and prototype an implementation of a generic security management framework for BlobSeer-based cloud storage; 3) Design mechanisms facilitating the deployment of BlobSeer on XtreemOS-enabled IaaS clouds based on Nimbus.

The main results achieved this year are described in detail at http://www.irisa.fr/kerdata/doku.php?id=cloud_at_work:work_programme:work_programme_2010:work_programme_2010. We would like to emphasize the following facts:

Collaboration formally extended to Argonne National Lab, USA: B. Nicolae visited ANL (USA) thanks to the INRIA Explorateur Programme for 3 months (April to July 2010). This served as a preliminary step preparing the MapReduce ANR project started in October 2010 in partneship with ANL.

Visiting PhD students: In 2010, 3 PhD students from PUB hosted in Rennes for 3 months each (9 months overall). One PhD student from Rennes hosted in Bucharest twice (two weeks overall).

Publications and workshops: In 2010, 3 joint publications involving at least 2 of the 3 partners of the Associate Team have been made, 2 joint publications with Argonne National Lab and a large number of Master and Bachelor theses. The results were presented at 3 internal workshops organized in Rennes.

PhD defenses: In 2010, 2 PhD theses strongly related to the Associate Team have been defended: B .Nicolae (KerData) in Rennes and Alexandru Costan (PUB) in Bucharest. The French and Romanian leaders of the Associate Team participated to both PhD committees.

Master and Bachelor theses: Overall, 6 Bachelor theses locally carried out in Bucharest and 4 Master theses in Rennes were dedicated to subtasks derived from the scientific schedule of the Data-Cloud@work Associate Team. Out of these, 2 Master students from PUB were hosted by the KerData team through INRIA's Internship Programme (co-funded by KerData on its own resources).

8.4. International initiatives

8.4.1. MapReduce: an ANR project with ANL (USA), UIUC (USA) and JLPC (France-USA)

MapReduce is an ANR project with international partners: Argonne National Lab (USA), the University of Illinois at Urbana-Champaign (UIUC, USA) and the Joint INRIA-UIUC Lab for Petascale Computing (JLPC). See Section 8.2 for details.

8.4.2. INRIA-UIUC Joint Laboratory on Petascale Computing

Participants: Gabriel Antoniu, Luc Bougé, Bogdan Nicolae, Viet-Trung Tran.

This work has been done in collaboration with Matthieu Dorier, student at ENS Cachan, Brittany Campus, during his summer 2010 internship at the INRIA-UIUC Joint Laboratory for Petascale Computing at Urbana-Champaign.

Preliminary discussions have been held at the 2nd workshop of the INRIA-UIUC Joint Laboratory for Petascale Computing (JLPC, http://jointlab.ncsa.illinois.edu/) in December 2009. As a follow-up, a specific topic was identified for the involvement of the KerData team in a collaboration with JLPC in the area of distributed storage for Petascale architectures. It focuses on the Blue Waters machine (http://www.ncsa.illinois.edu/BlueWaters/), expected to become one of the the world's most powerful supercomputer in 2011.

G. Antoniu and B. Nicolae visited the National Center for Supercomputing Applications (NCSA) at UIUC in April 2010 to explore how the BlobSeer BLOB-based approach developed by KerData could be used to optimize the management of concurrent data I/O requests generated by massively parallel simulations that run simultaneously with parallel visualization tools. A preliminary study in this context was performed by Matthieu Dorier, Master student (M1) at ENS Cachan/Brittany, during a 3-month internship at NCSA/UIUC, in collaboration with several researchers at NCSA/UIUC involved in the JLPC (Marc Snir, Franck Cappello, Dave Semeraro). This study showed the benefit of a new approach using dedicated I/O cores.

We intend to extend this approach in two directions: 1) Compare the use of dedicated cores with the use of dedicated nodes, and model the performance of both approaches in order to select the best one according the the applications and execution platforms I/O characteristics; 2) Build a BlobSeer-based metadata software layer enabled to schedule I/O operations coming from the simulation. This work will continue during the master internship (M2) of Matthieu Dorier at KerData in 2011. It is expected to be pursued further during his PhD thesis in the KerData Team. This topic is also part of INRIA's proposed contribution in the framework of an IP European project proposal to be submitted in January 2011. This IP project will involve 2 INRIA teams: KerData in Rennes and GRAND-LARGE in Saclay (through the JLPC at Urbana-Champaign).

8.4.3. FP3C: an ANR-JST Project

Participants: Gabriel Antoniu, Viet-Trung Tran, Bogdan Nicolae.

FP3C (Framework and Programming for Post-Petascale Computing) is a joint project co-funded by the French National Research Agency (ANR) and by the Japan Science and Technology Agency (JST). It started in September 2010 for 3 years. Its main goal is to develop a programming chain and associated runtime systems which will allow scientific end-users to efficiently execute their applications on Post-Petascale, highly hierarchical computing platforms making use of multi-core processors and accelerators. This project gathers majors actors involved in HPC research in France (INRIA, CEA, CNRS) and Japan (University of Tsukuba, University of Tokyo, Tokyo Institute of Technology, University of Kyoto).

Within this framework, we collaborate with Osamu Tatebe from the University of Tsukuba in the area of large-scale data-sharing. The goal of this collaboration is to design, implement and validate an integrated architecture for a Petascale storage system by weaving the best properties of global file systems (transparency, standard access interface) and RAM-based, BLOB storage systems (versioning, access efficiency under heavy concurrency). More specifically, we intend to explore how a hierarchical approach can be used to build a BLOB-based storage system file system.

While such an approach has been used in classical, non-distributed computer architecture to explore the combined usage of file storage and RAM storage, no convincing tentative has been made regarding Post-Petascale distributed storage systems. As a first step, our objective in 2011 is to specify the the joint architecture for a BLOB-based file storage architecture.

8.5. Other contacts

8.5.1. Orange Labs, Issy-les-Moulineaux

Several informal discussions took place with Ruby Krishnaswamy from Orange Labs, Issy-les-Moulineaux on potential collaborations in the area of cloud storage. Orange Labs is interested in BlobSeer-based concurrency-optimized storage support for virtual machine images and cloud application data.

9. Dissemination

9.1. Committees

9.1.1. Leaderships, Steering Committees and community service

Euro-Par Conference Series. L. Bougé serves as a Vice-Chair of the *Steering Committee* of the *Euro-Par* annual conference series on parallel computing. G. Antoniu serves as a Local Chair for the *Parallel and Distributed Data Management* topic of *Euro-Par 2011*, to be held in Bordeaux.

NAS-2010 Conference. G. Antoniu served as a Vice-Chair of the *Program Committee* for the storage track of the *IEEE NAS* international conference on Networking, Architecture, and Storage.

MapReduce ANR Project. G. Antoniu serves as a coordinator for the MapReduce ANR project (ARPEGE 2010 call), started in October 2010 in collaboration with Argonne National Lab, the University of Illinois at Urbana Champaign, the UIUC/INRIA Joint Lab on Petascale Computing, IBM, IBCP, MEDIT and the GRAAL INRIA Project-Team.

AzureBrain Microsoft-INRIA Project. G. Antoniu and B. Thirion (PARIETAL Project-Team, INRIA SACLAY – ÎLE-DE-FRANCE) co-lead the AzureBrain Microsoft-INRIA Project started in October 2010 in the framework of the Microsoft Research - INRIA Joint Center (2010-2012).

DataCloud@work Associate Team. G. Antoniu serves as a coordinator for the DataCloud@work Associate Team, a project involving the KerData and MYRIADS INRIA Teams in Rennes and the Distributed Systems Group from Politehnica University of Bucharest (2010–2012).

SCALUS Marie-Curie Initial Training Networks project. G. Antoniu coordinates the involvement of the INRIA RENNES – BRETAGNE ATLANTIQUE Research Center in the SCALUS Project of the Marie-Curie Initial Training Networks Programme (ITN), call FP7-PEOPLE-ITN-2008 (2009-2013).

CoreGRID ERCIM Working Group. G. Antoniu coordinates the involvement of the INRIA RENNES – BRETAGNE ATLANTIQUE Research Center in the CoreGRID ERCIM Working Group.

Agrégation of Mathematics. L. Bougé serves as a Vice-Chair of the National Selection Committee for High-School Mathematics Teachers, Informatics Track.

9.1.2. Editorial boards, direction of program committees

L. Bougé is a member of the *Editorial Advisory Board* of the *Scientific Programming* Journal.

9.1.3. Program Committees

G. Antoniu served in the Program Committees for the following conferences and workshops: CloudCom 2010, ICPADS 2010, 3PGCIC-2010, MapReduce 2010, MAPRED 2010, ADiS 2010, SRMPDS 2010, AINA 2011, CISIS 2011, PDP 2011, RenPar'19, RenPar'20.

L. Bougé served in the Program Committee for the following conferences: NPC 2010.

9.1.4. Evaluation committees, consulting

L. Bougé served as a member of the Selection Committee for the *Gilles Kahn PhD Thesis Award 2010*.

L. Bougé was the chair of the national evaluation committee for the 2010 Scientific Excellence Award (*Prime d'excellence scientifique*, PES) targeted to the researchers on an academic teaching position in France.

9.2. Invited talks

G. Antoniu gave a keynote talk entitled *Autonomic cloud storage: challenges at stake* at the ADiS workshop held in February 2010 in Krakow, Poland.

G. Antoniu gave an invited talk entitled *BlobSeer: Enabling Efficient Lock-Free, Versioning-Based Storage for Massive Data under Heavy Access Concurrency* at the Parallel@Illinois Special Event Series, University of Illinois at Urbana-Champaign, IL, USA, in April 2010.

G. Antoniu gave a keynote talk entitled *Scalable MapReduce Data Processing on Clouds: the BlobSeer Approach* at the International Conference on High Performance Computing and Simulation (HPCS 2010) conference held in June 2010 in Caen, France.

G. Antoniu gave a a talk entitled *BlobSeer: Efficient, Versioning-Based Storage for Massive Data under Heavy Access Concurrency on Clouds* at Microsoft Research - INRIA Workshop on Extreme Operating Systems held in November 2010 in Paris, France.

G. Antoniu gave an invited talk entitled *Concurrency-optimized I/O for visualizing HPC simulations: An Approach Using Dedicated I/O cores* at the 4nd workshop of the Joint Laboratory for Petascale Computing held in November 2010 at NCSA/UIUC, Urbana-Champaign, IL, USA.

9.3. Doctoral teaching

Only the teaching contributions of project-team members on non-teaching positions are mentioned below.

G. Antoniu gave lectures on peer-to-peer systems within the *Peer-to-Peer Systems* Module of the Master Program (2nd year), UNIVERSITY RENNES 1. He gave lectures on Grid Data Management within the *Distributed Architectures* Module of the ALMA Master Program (2nd year) of the University of Nantes. He also taught a full course on *Grid Computing* for final year engineering students at the ESIEA Engineering School, Paris.

9.4. Administrative responsibilities

G. Antoniu serves as the Scientific Correspondent for the International Relations Office of the INRIA RENNES – BRETAGNE ATLANTIQUE Research Center.

G. Antoniu serves as the Scientific Leader of the KerData research team.

L. Bougé chairs the Computer Science and Telecommunication Department (*Département Informatique et Télécommunications, DIT*) of the Brittany Extension of ENS CACHAN. He leads the Master Program (*Magistère*) in Computer Science at the Brittany Extension of ENS CACHAN.

9.5. Miscellaneous

L. Bougé is a member of Scientific Committee of INRIA RENNES – BRETAGNE ATLANTIQUE (*Comité des projets*), standing for the ENS CACHAN partner.

G. Antoniu is a member of Scientific Committee of INRIA RENNES – BRETAGNE ATLANTIQUE (*Comité des projets*), standing for the KerData research team.

10. Bibliography

Major publications by the team in recent years

[1] G. ANTONIU, L. CUDENNEC, M. JAN, M. DUIGOU. *Performance scalability of the JXTA P2P framework*, in "Proc. IEEE International Parallel and Distributed Processing Symposium (IPDPS 2007)", Long Beach, USA, 2007, 108, http://hal.inria.fr/inria-00178653/en/.

[2] G. ANTONIU, J.-F. DEVERGE, S. MONNET. *How to bring together fault tolerance and data consistency to enable grid data sharing*, in "Concurrency and Computation: Practice and Experience", 2006, n^o 17, p. 1-19, http://hal.inria.fr/inria-00000987/en/.

[3] R. MORALES, S. MONNET, I. GUPTA, G. ANTONIU. *MOve:Design and Evaluation of A Malleable Overlay for Group-Based Applications*, in "IEEE Transactions on Network and Service Management, Special Issue on Self-Management", 2007, vol. 4, p. 107-116 [*DOI :* 10.1109/TNSM.2007.070903], http://hal.inria.fr/inria-00446067/en/.

[4] B. NICOLAE, G. ANTONIU, L. BOUGÉ, D. MOISE, A. CARPEN-AMARIE. *BlobSeer: Next Generation Data Management for Large Scale Infrastructures*, in "Journal of Parallel and Distributed Computing", February 2011, vol. 71, n^o 2, p. 169-184, Special issue on data intensive computing. To appear, http://hal.inria.fr/inria-00511414/en/.

[5] B. NICOLAE, D. MOISE, G. ANTONIU, L. BOUGÉ, M. DORIER. *BlobSeer: Bringing High Throughput under Heavy Concurrency to Hadoop Map-Reduce Applications*, in "24th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2010)", Atlanta, IEEE and ACM, Apr 2010, A preliminary version of this paper has been published as INRIA Research Report RR-7140., http://hal.inria.fr/inria-00456801.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[6] B. NICOLAE. *BlobSeer: Towards efficient data storage management for large-scale, distributed systems*, University Rennes 1, IRISA/INRIA, Rennes, France, November 2010, To appear.

Articles in International Peer-Reviewed Journal

[7] B. NICOLAE, G. ANTONIU, L. BOUGÉ, D. MOISE, A. CARPEN-AMARIE. *BlobSeer: Next Generation Data Management for Large Scale Infrastructures*, in "Journal of Parallel and Distributed Computing", February 2011, vol. 71, n^o 2, p. 169-184, Special issue on data intensive computing. To appear, http://hal.inria.fr/inria-00511414/en/.

International Peer-Reviewed Conference/Proceedings

[8] C. BASESCU, A. CARPEN-AMARIE, C. LEORDEANU, A. COSTAN, G. ANTONIU. *Managing Data Access on Clouds: A Generic Framework for Enforcing Security Policies*, in "The 25th International Conference on Advanced Information Networking and Applications (AINA-2011)", Singapore, Institute for Infocomm Research (I2R), in cooperation with the Singapore Chapter of ACM, 2011, http://hal.inria.fr/inria-00536603/en/.

[9] A. CARPEN-AMARIE, J. CAI, A. COSTAN, G. ANTONIU, L. BOUGÉ. *Bringing Introspection Into the BlobSeer Data-Management System Using the MonALISA Distributed Monitoring Framework*, in "First International Workshop on Autonomic Distributed Systems (ADiS 2010)", Krakow, Poland, 2010, p. 508-513, Held in conjunction with CISIS 2010 Conference, http://hal.inria.fr/inria-00419978/en/.

[10] D. MOISE, G. ANTONIU, L. BOUGÉ. *Improving the Hadoop Map/Reduce Framework to Support Concurrent Appends through the BlobSeer BLOB management system*, in "The First International Workshop on MapReduce and its Applications (MAPREDUCE'10)", Chicago, IL, USA, June 2010, http://hal.inria.fr/inria-00476861/en/.

[11] D. MOISE. *Large-Scale Distributed Storage for Highly Concurrent MapReduce Applications*, in "PhD Forum of IPDPS '10: 24th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2010)", Atlanta, GA, USA, April 2010, http://hal.inria.fr/inria-00458143/en/.

[12] J. MONTES, B. NICOLAE, G. ANTONIU, A. SÁNCHEZ, M. PÉREZ HERNÁNDEZ. *Using Global Behavior Modeling to Improve QoS in Cloud Data Storage Services*, in "CloudCom'10: Proc. 2nd IEEE International Conference on Cloud Computing Technology and Science", Indianapolis, IN, USA, October 2010, http://hal.inria.fr/inria-00527650/en/.

[13] B. NICOLAE, D. MOISE, G. ANTONIU, L. BOUGÉ, M. DORIER. *BlobSeer: Bringing High Throughput under Heavy Concurrency to Hadoop Map-Reduce Applications*, in "24th IEEE International Parallel and Distributed Processing Symposium (IPDPS 2010)", Atlanta, GA, USA, IEEE and ACM, April 2010, A preliminary version of this paper has been published as INRIA Research Report RR-7140, http://hal.inria.fr/inria-00456801/en/.

[14] B. NICOLAE. *BlobSeer: Efficient Data Management for Data-Intensive Applications Distributed at Large-Scale*, in "PhD Forum of IPDPS '10: 24th IEEE International Symposium on Parallel and Distributed Processing", Atlanta, GA, USA, 2010, p. 1-4, Best Poster Award, http://hal.inria.fr/inria-00457809/en/.

[15] B. NICOLAE. *High Throughput Data-Compression for Cloud Storage*, in "3rd International Conference on Data Management in Grid and P2P Systems (Globe 2010)", Espagne Bilbao, June 2010, vol. 6265, p. 1-12, http://hal.inria.fr/inria-00490541/en/.

Research Reports

[16] A. CARPEN-AMARIE, J. CAI, A. COSTAN, G. ANTONIU, L. BOUGÉ. *Bringing Introspection into BlobSeer: Towards a Self-Adaptative Distributed Data Management System*, INRIA, November 2010, RR-7452, http://hal.inria.fr/inria-00536556/en/.

[17] A. CARPEN-AMARIE, TUAN-VIET. DINH, G. ANTONIU. *Efficient VM Storage for Clouds Based on the High-Throughput BlobSeer BLOB Management System*, INRIA, October 2010, RR-7434, http://hal.inria.fr/inria-00528928/en/.

[18] J. MONTES SÁNCHEZ, B. NICOLAE, G. ANTONIU, A. SÁNCHEZ CAMPOS, M. PÉREZ HERNÁNDEZ. *Using Global Behavior Modeling to Improve QoS in Large-scale Distributed Data Storage Services*, INRIA, May 2010, RR-7271, http://hal.inria.fr/inria-00482568/en/.

[19] B. NICOLAE, J. BRESNAHAN, K. KEAHEY, G. ANTONIU. *Going Back and Forth: Efficient Virtual Machine Image Deployment and Snapshotting on IaaS Clouds*, INRIA, 2010, http://hal.archives-ouvertes.fr/inria-00545232/en/.

[20] V.-T. TRAN, B. NICOLAE, G. ANTONIU, L. BOUGÉ. *Efficient support for MPI-IO atomicity based on versioning*, INRIA, 2010, Submitted for publication, http://hal.archives-ouvertes.fr/inria-00546956/en/.

[21] M.-C. VLAD. *Distributed Monitoring for User Accounting in the BlobSeer Distributed Storage System*, INRIA, September 2010, RR-7436, http://hal.inria.fr/inria-00531049/en/.

Other Publications

[22] TUAN-VIET. DINH. *Using BlobSeer Data Sharing Platform for Cloud Virtual Machine Repository*, ENS Cachan/Bretagne, IRISA/INRIA, Rennes, France, June 2010, To appear.

[23] THI-THU-LAN. TRIEU. *Intermediate Data Management for Map/Reduce Applications*, ENS Cachan/Bretagne, IRISA/INRIA, Rennes, France, June 2010, To appear.

References in notes

[24] *Chirp protocol specification*, 2009, http://www.cs.wisc.edu/condor/chirp/.

[25] *Lightweight Data Replicator*, 2009, http://www.lsc-group.phys.uwm.edu/LDR/.

[26] *Google App Engine*, 2009, http://code.google.com/appengine/.

[27] *Google Docs*, 2009, http://www.google.com/google-d-s/tour1.html.

[28] *HadoopFS*, 2009, http://hadoop.apache.org/hdfs/docs/current/.

[29] *Microsoft Azure*, 2009, http://www.microsoft.com/azure/.

[30] *Microsoft Office Live*, 2009, http://www.officelive.com/.

[31] *The Nimbus project*, 2009, http://workspace.globus.org/.

[32] *OpenNebula*, 2010, http://www.opennebula.org/.

[33] *The XtreemOS project*, 2009, http://www.xtreemos.eu/.

[34] B. ALLCOCK, J. BESTER, J. BRESNAHAN, A. L. CHERVENAK, I. FOSTER, C. KESSELMAN, S. MEDER, V. NEFEDOVA, D. QUESNEL, S. TUECKE. *Data management and transfer in high-performance computational grid environments*, in "Parallel Comput.", 2002, vol. 28, n^{o} 5, p. 749–771, http://dx.doi.org/10.1016/S0167-8191(02)00094-7.

[35] G. ANTONIU, M. BERTIER, E. CARON, F. DESPREZ, L. BOUGÉ, M. JAN, S. MONNET, P. SENS. *GDS: An Architecture Proposal for a grid Data-Sharing Service*, in "Future Generation Grids", CoreGRID series, Springer, 2006, p. 133-152.

[36] G. ANTONIU, L. BOUGÉ, M. JAN. *JuxMem: An Adaptive Supportive Platform for Data Sharing on the Grid*, in "Scalable Computing: Practice and Experience", November 2005, vol. 6, n^{o} 3, p. 45–55, http://hal.inria.fr/inria-00000984.

[37] A. BASSI, M. BECK, G. FAGG, T. MOORE, J. S. PLANK, M. SWANY, R. WOLSKI. *The Internet Backplane Protocol: A Study in Resource Sharing*, in "Proc. 2nd IEEE/ACM Intl. Symp. on Cluster Computing and the Grid (CCGRID '02)", Washington, DC, USA, IEEE Computer Society, 2002, 194.

[38] J. BENT, V. VENKATARAMANI, N. LEROY, A. ROY, J. STANLEY, A. ARPACI-DUSSEAU, R. ARPACI-DUSSEAU, M. LIVNY. *Flexibility, Manageability, and Performance in a Grid Storage Appliance*, in "Proc. 11th IEEE Symposium on High Performance Distributed Computing (HPDC 11)", 2002.

[39] R. BUYYA, C. S. YEO, S. VENUGOPAL. *Market-Oriented Cloud Computing: Vision, Hype, and Reality for Delivering IT Services as Computing Utilities*, in "HPCC '08: Proceedings of the 2008 10th IEEE International Conference on High Performance Computing and Communications", Washington, DC, USA, IEEE Computer Society, 2008, p. 5–13, http://dx.doi.org/10.1109/HPCC.2008.172.

[40] P. H. CARNS, W. B. LIGON, R. B. ROSS, R. THAKUR. *PVFS: A Parallel File System for Linux Clusters*, in "ALS '00: Proceedings of the 4th Annual Linux Showcase and Conference", Atlanta, GA, USA, USENIX Association, 2000, p. 317–327.

[41] M. A. CASEY, F. KURTH. *Large data methods for multimedia*, in "Proc. 15th Intl. Conf. on Multimedia (Multimedia '07)", New York, NY, USA, ACM, 2007, p. 6–7, http://doi.acm.org/10.1145/1291233.1291238.

[42] F. COSTA, L. SILVA, G. FEDAK, I. KELLEY. *Optimizing data distribution in desktop grid platforms*, in "Parallel Processing Letters (PPL)", 2008, vol. 18, p. 391 - 410, http://dx.doi.org/10.1142/S0129626408003466.

[43] J. DEAN, S. GHEMAWAT. *MapReduce: simplified data processing on large clusters*, in "Communications of the ACM", 2008, vol. 51, n^{o} 1, p. 107–113.

[44] A. DEVULAPALLI, D. DALESSANDRO, P. WYCKOFF, N. ALI, P. SADAYAPPAN. *Integrating parallel file systems with object-based storage devices*, in "SC '07: Proceedings of the 2007 ACM/IEEE conference on Supercomputing", New York, NY, USA, ACM, 2007, p. 1–10, http://dx.doi.org/10.1145/1362622.1362659.

[45] K. DOUGLAS, S. DOUGLAS. *PostgreSQL*, New Riders Publishing, Thousand Oaks, CA, USA, 2003.

[46] M. FACTOR, K. METH, D. NAOR, O. RODEH, J. SATRAN. *Object storage: the future building block for storage systems*, in "Local to Global Data Interoperability - Challenges and Technologies, 2005", 2005, p. 119–123, http://dx.doi.org/10.1109/LGDI.2005.1612479.

[47] S. GHEMAWAT, H. GOBIOFF, S.-T. LEUNG. *The Google file system*, in "SOSP '03: Proceedings of the nineteenth ACM symposium on Operating systems principles", New York, NY, USA, ACM Press, 2003, p. 29–43, http://dx.doi.org/10.1145/945445.945450.

[48] S. GRIMES. *Unstructured Data and the 80 Percent Rule*, 2008, Carabridge Bridgepoints.

[49] P. HONEYMAN, W. A. ADAMSON, S. MCKEE. *GridNFS: global storage for global collaborations*, in "Proc. IEEE Intl. Symp. Global Data Interoperability - Challenges and Technologies", Sardinia, Italy, IEEE Computer Society, June 2005, p. 111–115.

[50] M. IBRAHIM, R. ANTHONY, T. EYMANN, A. TALEB-BENDIAB, L. GRUENWALD. *Exploring Adaptation & Self-Adaptation in Autonomic Computing Systems*, in "Database and Expert Systems Applications, International Workshop on", 2006, vol. 0, p. 129-138, http://doi.ieeecomputersociety.org/10.1109/DEXA.2006.57.

[51] R. JIN, G. YANG. *Shared Memory Parallelization of Data Mining Algorithms: Techniques, Programming Interface, and Performance*, in "IEEE Trans. on Knowl. and Data Eng.", 2005, vol. 17, n^{o} 1, p. 71–89, http://dx.doi.org/10.1109/TKDE.2005.18.

[52] K. KEAHEY, T. FREEMAN. *Science Clouds: Early Experiences in Cloud Computing for Scientific Applications*, in "Cloud Computing and Its Applications 2008 (CCA-08)", Chicago, IL, 2008.

[53] J. O. KEPHART, D. M. CHESS. *The Vision of Autonomic Computing*, in "Computer", 2003, vol. 36, n^{o} 1, p. 41–50, http://dx.doi.org/10.1109/MC.2003.1160055.

[54] A. LENK, M. KLEMS, J. NIMIS, S. TAI, T. SANDHOLM. *What's inside the Cloud? An architectural map of the Cloud landscape*, in "Software Engineering Challenges of Cloud Computing (CLOUD '09)", 2009, p. 23 - 31, ICSE Workshop.

[55] M. MESNIER, G. R. GANGER, E. RIEDEL. *Object-based storage*, in "Communications Magazine, IEEE", 2003, vol. 41, n^{o} 8, p. 84–90, http://dx.doi.org/10.1109/MCOM.2003.1222722.

[56] C. MORIN, J. GALLARD, Y. JÉGOU, P. RITEAU. *Clouds: a new playground for the XtreemOS Grid operating system*, in "Parallel Processing Letters", 2009, vol. 19, n^{o} 3, p. 435-449, To appear.

[57] C. MORIN. *XtreemOS: a Grid Operating System Making your Computer Ready for Participating in Virtual Organizations*, in "IEEE International Symposium on Object/component/service-oriented Real-time distributed Computing (ISORC)", Santorini Island, Greece, 2007.

[58] M. NICOLA, M. JARKE. *Performance Modeling of Distributed and Replicated Databases*, in "IEEE Trans. on Knowl. and Data Eng.", 2000, vol. 12, n^{o} 4, p. 645–672, http://dx.doi.org/10.1109/69.868912.

[59] C. OLSTON, B. REED, U. SRIVASTAVA, R. KUMAR, A. TOMKINS. *Pig latin: a not-so-foreign language for data processing*, in "SIGMOD '08: Proceedings of the 2008 ACM SIGMOD international conference on Management of data", New York, NY, USA, ACM, 2008, p. 1099–1110, http://doi.acm.org/10.1145/1376616.1376726.

[60] M. PARASHAR, S. HARIRI. *Autonomic computing: An overview*, in "Unconventional Programming Paradigms", Springer Verlag, 2005, p. 247–259.

[61] A. RAGHUVEER, M. JINDAL, M. F. MOKBEL, B. DEBNATH, D. DU. *Towards efficient search on unstructured data: an intelligent-storage approach*, in "CIKM '07: Proceedings of the sixteenth ACM conference on Conference on information and knowledge management", New York, NY, USA, ACM, 2007, p. 951–954, http://doi.acm.org/10.1145/1321440.1321583.

[62] P. SCHWAN. *Lustre: Building a file system for 1000-node clusters*, in "Proceedings of the Linux Symposium", 2003, http://www.kernel.org/doc/ols/2003/ols2003-pages-380-386.pdf.

[63] O. TATEBE, Y. MORITA, S. MATSUOKA, N. SODA, S. SEKIGUCHI. *Grid Datafarm Architecture for Petascale Data Intensive Computing*, in "Proc. 2nd IEEE/ACM Intl. Symp. on Cluster Computing and the Grid (Cluster 2002)", Washington DC, USA, IEEE Computer Society, 2002, 102.

[64] A. THOMASIAN. *Concurrency control: methods, performance, and analysis*, in "ACM Computing Survey", 1998, vol. 30, n° 1, p. 70–119, http://doi.acm.org/10.1145/274440.274443.

[65] L. M. VAQUERO, L. RODERO-MERINO, J. CACERES, M. LINDNER. *A break in the clouds: towards a cloud definition*, in "SIGCOMM Comput. Commun. Rev.", 2009, vol. 39, n° 1, p. 50–55, http://doi.acm.org/10.1145/1496091.1496100.

[66] S. A. WEIL, S. A. BRANDT, E. L. MILLER, D. D. E. LONG, C. MALTZAHN. *Ceph: a scalable, high-performance distributed file system*, in "OSDI '06: Proceedings of the 7th symposium on Operating systems design and implementation", Berkeley, CA, USA, USENIX Association, 2006, p. 307–320, http://portal.acm.org/citation.cfm?id=1298455.1298485.

[67] B. S. WHITE, M. WALKER, M. HUMPHREY, A. S. GRIMSHAW. *LegionFS: a secure and scalable file system supporting cross-domain high-performance applications*, in "Proc. 2001 ACM/IEEE Conf. on Supercomputing (SC '01)", New York, NY, USA, ACM Press, 2001, p. 59–59.