IN PARTNERSHIP WITH:
CNRS

Université Pierre et Marie Curie (Paris 6)

Activity Report 2012

Project-Team REGAL

Large-Scale Distributed Systems and Applications

IN COLLABORATION WITH: Laboratoire d'informatique de Paris 6 (LIP6)

RESEARCH CENTER
Paris - Rocquencourt

THEME
Distributed Systems and Services

Table of contents

Project-Team REGAL

Keywords: Data Management, Cloud Computing, Distributed Algorithms, Fault Tolerance, Replication And Consistency, Peer-to-peer, Cloud Computing

Creation of the Project-Team: July 01, 2005 .

1. Members

Research Scientists

Julia Lawall [Senior Researcher]
Gilles Muller [Senior Researcher, HdR]
Marc Shapiro [Senior Researcher, HdR]
Mesaac Makpangou [Junior Researcher, HdR]

Faculty Members

Pierre Sens [Team Leader, Professor Université Paris 6, HdR]
Luciana Arantes [Associate Professor, Université Paris 6]
Bertil Folliot [Professor, Université Paris 6, HdR]
Maria Potop-Butucaru [Associate Professor, Université Paris 6 until August, HdR]
Olivier Marin [Associate Professor, Université Paris 6]
Sébastien Monnet [Associate Professor, Université Paris 6]
Franck Petit [Professor, Université Paris 6, HdR]
Julien Sopena [Associate Professor, Université Paris 6]
Gaël Thomas [Associate Professor, Université Paris 6, HdR]

Engineers

Harris Bakiras
Ikram Chabbouh
Christian Clausen
Véronique Simon [Université Paris 6]
Pierre Sutra [until Sept. 2012]

PhD Students

Koutheir Attouchi [Université Paris 6 - CIFRE Orange Labs]
Lamia Benmouffok [Université Paris 6 until August]
Pierpaolo Cincilla [Université Paris 6]
Florian David [Université Paris 6]
Raluca Diaconu [Université Paris 6 - CIFRE Orange Labs]
Lokesh Gidra [Université Paris 6]
Lisong Guo [Université Paris 6, since September]
Ruijing Hu [Université Paris 6]
Mohsen Koohi-Esfahani [Université Paris 6]
Anissa Lamani [Université Amiens]
Maxime Lorrillere [Université Paris 6, since October]
Mohamed-Hamza Kaaouachi [Université Paris 6, since October]
Sergey Legtchenko [Université Paris 6, PhD October 2012]
Jonathan Lejeune [Université Paris 6]
Jean-Pierre Lozi [Université Paris 6]
Mahsa Najafzadeh [CORDI-S since Nov. 2012; Université Paris 6]
Thomas Preud homme [Université Paris 6]
Karine Pires [Université Paris 6]
Masoud Saeida Ardekani [Université Paris 6]

Suman Saha [Université Paris 6]
Guthemberg Silvestre [Université Paris 6 - CIFRE Orange Labs]
Maxime Véron [Université Paris 6, since October]
Marek Zawirski [Université Paris 6]

Post-Doctoral Fellows

Annette Bieniusa [until Sept. 2012]
Swan Dubois [Université Paris 6 until Aug. 2012]
Arie Middlekoop [Université Paris 6 until September]

Administrative Assistant

Hélène Milome [Secretary]

2. Overall Objectives

2.1. Overall Objectives

The research of the Regal team focuses on large-scale parallel and distributed systems. It addresses the challenges of automated adminstration of highly dynamic networks, of fault tolerance, of consistency in large-scale distributed systems, of information sharing in collaborative groups, and of dynamic content distribution. It aims to design efficient, robust and flexible operating systems for multicore computers.

Regal is a joint research team between LIP6 and Inria Paris-Rocquencourt.

2.2. Highlights of the Year

: Tegawendé F. Bissyandé (LaBRI, Bordeaux), Laurent Réveillère (LaBRI, Bordeaux), Julia Lawall (Regal) and Gilles Muller (Regal) received the best paper award at ASE 2012 for their work on Diagnosys: Automatic Generation of a Debugging Interface to the Linux Kernel.
BEST PAPER AWARD :
[24] **Diagnosys: Automatic Generation of a Debugging Interface to the Linux Kernel in 27th IEEE/ACM International Conference on Automated Software Engineering (ASE 2012)**. T. F. BISSYANDÉ, L. RÉVEILLÈRE, J. LAWALL, G. MULLER.

3. Scientific Foundations

3.1. Research rationale

Peer-to-peer, Cloud computing, distributed system, data consistency, fault tolerance, dynamic adaptation, large-scale environments, replication.

As society relies more and more on computers, responsiveness, correctness and security are increasingly critical. At the same time, systems are growing larger, more parallel, and more unpredictable. Our research agenda is to design Computer Systems that remain correct and efficient despite this increased complexity and in spite of conflicting requirements.[1]

While our work historically focused on distributed systems, we now cover a larger part of the whole Computer Systems spectrum. Our topics now also include Managed Run-time Environments (MREs, a.k.a. language-level virtual machines) and operating system kernels. This holistic approach allows us to address related problems at different levels. It also permits us to efficiently share knowledge and expertise, and is a source of originality.

[1] From the web page of ACM Transactions on Computer Systems: *"The term* 'computer systems' *is interpreted broadly and includes systems architectures, operating systems, distributed systems, and computer networks."* See http://tocs.acm.org/.

Computer Systems is a rapidly evolving domain, with strong interactions with industry. Two main evolutions in the Computer Systems area have strongly influenced our research activities:

3.1.1. Modern computer systems are increasingly distributed.

Ensuring the persistence, availability and consistency of data in a distributed setting is a major requirement: the system must remain correct despite slow networks, disconnection, crashes, failures, churn, and attacks. Ease of use, performance and efficiency are equally important for systems to be accepted. These requirements are somewhat conflicting, and there are many algorithmic and engineering trade-offs, which often depend on specific workloads or usage scenarios.

Years of research in distributed systems are now coming to fruition, and are being used by millions of users of web systems, peer-to-peer systems, gaming and social applications, or cloud computing. These new usages bring new challenges of extreme scalability and adaptation to dynamically-changing conditions, where knowledge of system state can only be partial and incomplete. The challenges of distributed computing listed above are subject to new trade-offs.

Innovative environments that motivate our research include peer-to-peer (P2P) and overlay networks, dynamic wireless networks, cloud computing, and manycore machines. The scientific challenges are scalability, fault tolerance, dynamicity and virtualization of physical infrastructure. Algorithms designed for classical distributed systems, such as resource allocation, data storage and placement, and concurrent access to shared data, need to be revisited to work properly under the constraints of these new environments.

Regal focuses in particular on two key challenges in these areas: the adaptation of algorithms to the new dynamics of distributed systems and data management on large configurations.

3.1.2. Multicore architectures are everywhere.

The fine-grained parallelism offered by multicore architectures has the potential to open highly parallel computing to new application areas. To make this a reality, however, many issues, including issues that have previously arisen in distributed systems, need to be addressed. Challenges include obtaining a consistent view of shared resources, such as memory, and optimally distributing computations among heterogeneous architectures, such as CPUs, GPUs, and other specialized processors. As compared to distributed systems, in the case of multicore architectures, these issues arise at a more fine-grained level, leading to the need for different solutions and different cost-benefit trade-offs.

Recent multicore architectures are highly diverse. Compiling and optimizing programs for such architectures can only be done for a given target. In this setting, MREs are an elegant approach since they permit distributing a unique binary representation of an application, to which architecture-specific optimizations can be applied late on the execution machine. Finally, the concurrency provided by multicore architectures also induces new challenges for software robustness. We consider this problem in the context of systems software, using static analysis of the source code and the technology developed in the Coccinelle tool.

4. Application Domains

4.1. Research domain

To address the evolution of distributed platforms in recent years, we focus on the following areas:

- *Distributed algorithms for dynamic and large networks.* Network topology is no more static; distributed systems are increasingly dynamic, i.e., nodes can join, fail, recover, disconnect and reconnect, and change location. Examples include IaaS cloud computing infrastructures, where virtual machines can be moved according to load peaks, opportunistic networks such as DTNs (Delay-Tolerant Networks), and networks of robots.

- *Management of distributed data.* In emerging architectures such as distributed hash tables (DHTs) and cloud computing, our research topics include replica placement, responsiveness, load balancing, consistency maintenance, consensus algorithms, and synchronisation. This research direction is funded by several new collaborative projects (ConcoRDanT, MyCloud, Nu@age, Odisea, Prose, Shaman, Spades, Streams, R-Discover) and by industrial funding (Google).
- *Performance and robustness of Systems Software in multicore architectures.* Our research focuses on the efficient management of system resources at the user level. Issues considered include efficient synchronization and memory management in large-scale multicore architectures. At the same time, we focus on the robustness of systems software, based on the Coccinelle technology. This work is funded by ANR ABL and InfraJVM.

5. Software

5.1. Coccinelle

Participants: Christian Clausen, Julia Lawall [correspondent], Arie Middlekoop, Gilles Muller [correspondent], Gaël Thomas, Suman Saha.

Coccinelle is a program matching and transformation engine which provides the language SmPL (Semantic Patch Language) for specifying desired matches and transformations in C code. Coccinelle was initially targeted towards performing collateral evolutions in Linux. Such evolutions comprise the changes that are needed in client code in response to evolutions in library APIs, and may include modifications such as renaming a function, adding a function argument whose value is somehow context-dependent, and reorganizing a data structure.

Beyond collateral evolutions, Coccinelle has been successfully used for finding and fixing bugs in systems code. One of the main recent results is an extensive study of bugs in Linux 2.6 that has permitted us to demonstrate that the quality of code has been improving over the last six years, even though the code size has more than doubled.

http://coccinelle.lip6.fr

5.2. SwiftCloud

Participants: Marc Shapiro [correspondent], Marek Zawirski, Annette Bieniusa, Valter Balegas.

SwiftCloud is a platform for deploying large-scale distributed applications on the edge, close to the users. Internet delays are a problem for interactive distributed applications. Truly optimal responsiveness and availability require mutable shared state replicated near the client, at the network edge. This raises serious challenges of consistency, fault tolerance, and programmability. The SwiftCloud system is designed to address these challenges. Our data model is based on client-side caching of shared mutable objects, made practical with a library of synchronisation-free, yet provably correct object types (CRDTs). The consistency model combines eventual consistency, absence of roll-backs, transactional consistency and session guarantees, but stops short of serialisability. This programming model is practically useful, yet does not require synchronisation, thus ensuring scalability. Maintaining the guarantees at a reasonable cost is especially challenging at large scale. Scalability and programmability are helped by several design decisions detailed in the paper. We validated our approach by building the SwiftCloud platform, by deploying three significant applications, and by measuring their performance in different configurations, in order to explore the benefits of replication at different locations.

SwiftCloud was supported by the ConcoRDanT ANR project (Section 7.1.4) and a Google European Doctoral Fellowship (Section 7.2.2.1).

The code is freely available on http://gforge.inria.fr/ under a BSD license.

5.3. Treedoc

Participants: Marc Shapiro [correspondent], Marek Zawirski, Nuno Preguiça.

A Commutative Replicated Data Type (CRDT) is one where all concurrent operations commute. The replicas of a CRDT converge automatically, without complex concurrency control. We designed and developed a novel CRDT design for cooperative text editing, called Treedoc. It is designed over a dense identifier space based on a binary trees. Treedoc also includes an innovative garbage collection algorithm based on tree rebalancing. In the best case, Treedoc incurs no overhead with respect to a linear text buffer. The implementation has been validated with performance measurements, based on real traces of social text editing in Wikipedia and SVN.

Work in 2010 has focused on studying large-scale garbage collection for Treedoc, and design improvements. Future work includes engineering a large-scale collaborative Wiki, and studying CRDTs more generally.

TreeDoc is supported by the Prose, Streams and ConcoRDanT ANR projects (Sections 7.1.7, 7.1.6 and 7.1.4 respectively) and by a Google European Doctoral Fellowship (Section 7.2.2.1).

The code is freely available on http://gforge.inria.fr/ under a BSD license.

5.4. Telex

Participants: Marc Shapiro [correspondent], Lamia Benmouffok, Pierre Sutra, Pierpaolo Cincilla.

Developing write-sharing applications is challenging. Developers must deal with difficult problems such as managing distributed state, disconnection, and conflicts. Telex is an application-independent platform to ease development and to provide guarantees. Telex is guided by application-provided parameters: actions (operations) and constraints (concurrency control statements). Telex takes care of replication and persistence, drives application progress, and ensures that replicas eventually agree on a correct, common state. Telex supports partial replication, i.e., sites only receive operations they are interested in. The main data structure of Telex is a large, replicated, highly dynamic graph; we discuss the engineering trade-offs for such a graph and our solutions. Our novel agreement protocol runs Telex ensures, in the background, that replicas converge to a safe state. We conducted an experimental evaluation of the Telex based on a cooperative calendar application and on benchmarks.

The code is freely available on http://gforge.inria.fr/ under a BSD license.

5.5. Java and .Net runtimes for LLVM

Participants: Harris Bakiras, Bertil Folliot, Julia Lawall, Jean-Pierre Lozi, Gaël Thomas [correspondent], Gilles Muller, Thomas Preud homme, Koutheir Attouchi.

Many systems research projects now target managed runtime environments (MRE) because they provide better productivity and safety compared to native environments. Still, developing and optimizing an MRE is a tedious task that requires many years of development. Although MREs share some common functionalities, such as a Just In Time Compiler or a Garbage Collector, this opportunity for sharing has not been yet exploited in implementing MREs. We are working on VMKit, a first attempt to build a common substrate that eases the development and experimentation of high-level MREs and systems mechanisms. VMKit has been successfully used to build two MREs, a Java Virtual Machine and a Common Language Runtime, as well as a new system mechanism that provides better security in the context of service-oriented architectures.

VMKit is an implementation of a JVM and a CLI Virtual Machines (Microsoft .NET is an implementation of the CLI) using the LLVM compiler framework and the MMTk garbage collectors. The JVM, called J3, executes real-world applications such as Tomcat, Felix or Eclipse and the DaCapo benchmark. It uses the GNU Classpath project for the base classes. The CLI implementation, called N3, is its in early stages but can execute simple applications and the “pnetmark” benchmark. It uses the pnetlib project or Mono as its core library. The VMKit VMs compare in performance with industrial and top open-source VMs on CPU-intensive applications. VMKit is publicly available under the LLVM license.

http://vmkit2.gforge.inria.fr/

6. New Results

6.1. Introduction

In 2012, we focused our research on the following areas:

- *Management of distributed data.*
- *Performance and robustness of Systems Software in multicore architectures.*

6.2. Distributed algorithms for dynamic networks

Participants: Luciana Arantes [correspondent], Olivier Marin, Sébastien Monnet, Franck Petit [correspondent], Maria Potop-Butucaru, Pierre Sens, Julien Sopena, Raluca Diaconu, Ruijing Hu, Anissa Lamani, Sergey Legtchenko, Jonathan Lejeune, Karine Pires, Guthemberg Silvestre, Véronique Simon.

This objective aims to design distributed algorithms adapted to new large scale or dynamic distributed systems, such as mobile networks, sensor networks, P2P systems, Grids, Cloud environments, and robot networks. Efficiency in such demanding environments requires specialised protocols, providing features such as fault or heterogeneity tolerance, scalability, quality of service, and self-stabilization. Our approach covers the whole spectrum from theory to experimentation. We design algorithms, prove them correct, implement them, and evaluate them in simulation, using OMNeT++ or PeerSim, and on large-scale real platforms such as Grid'5000. The theory ensures that our solutions are correct and whenever possible optimal; experimental evidence is necessary to show that they are relevant and practical.

Within this thread, we have considered a number of specific applications, including massively multi-player on-line games (MMOGs) and peer certification.

Since 2008, we have obtained results both on fundamental aspects of distributed algorithms and on specific emerging large-scale applications.

We study various key topics of distributed algorithms: mutual exclusion, failure detection, data dissemination and data finding in large scale systems, self-stabilization and self-* services.

6.2.1. Mutual Exclusion and Failure Detection.

Mutual Exclusion and Fault Tolerance are two major basic building blocks in the design of distributed systems. Most of the current mutual exclusion algorithms are not suitable for modern distributed architectures because they are not scalable, they ignore the network topology, and they do not consider application quality of service constraints. Under the ANR Project *MyCloud* and the FSE *Nu@age*, we study locking algorithms fulfilling some QoS constraints often found in Cloud Computing [38].

A classical way for a distributed system to tolerate failures is to detect them and then recover. It is now well recognized that the dominant factor in system unavailability lies in the failure detection phase. Regal has worked for many years on practical and theoretical aspects of failure detections and pioneered hierarchical scalable failure detectors. [2] Since 2008, we have studied the adaptation of failure detectors to dynamic networks. Following the model introduced in [18], we have proposed new algorithms to detect crashes and Byzantine behaviors [32].

These algorithms were designed as part of the ANR Project SHAMAN.

[2] Recent work by Leners et al published in SOSP 2011 uses our DSN 2003 paper as basis for performance comparison

6.2.2. Self-Stabilization and Self- Services.*

We have also approached fault tolerance through self-stabilization. Self-stabilization is a versatile technique to design distributed algorithms that withstand transient faults. In particular, we have worked on the unison problem, [3] i.e., the design of self-stabilizing algorithms to synchronize a distributed clock. As part of the ANR project *SPADES*, we have proposed several snap-stabilizing algorithms for the message forwarding problem that are optimal in terms of number of required buffers [36]. A snap-stabilizing algorithm is a self-stabilizing algorithm that stabilizes in 0 steps; in other words, such an algorithm always behaves according to its specification.

Finally, we have applied our expertise in distributed algorithms for dynamic and self-* systems in domains that at first glance seem quite far from the core expertise of the team, namely ad-hoc systems and swarms of mobile robots. In the latter, as part of ANR project *R-Discover*, we have studied various problems such as exploration [29], and gathering [15].

6.2.3. Dissemination and Data Finding in Large Scale Systems.

In the area of large-scale P2P networks, we have studied the problems of data dissemination and overlay maintenance, i.e., maintenance of a logical network built over the a P2P network. First, we have proposed efficient distributed algorithms to ensure data dissemination to a large set of nodes. Also, we have introduced a new method to compare dissemination algorithms over various topologies [35].

6.2.4. MMOGs.

Peer-to-peer overlay networks can be used to build scalable infrastructures for MMOGs. Our work on MMOGs has primarily focused on the impact of latency constraints in dynamic distributed systems. In online P2P games, players are connected by a logical graph, implemented as an overlay network. Latency constraints imply that players that interact must remain close in the overlay, even when the mobility of players induces rapid changes in the graph.

We have also addressed problems related to cheating and arbitration. In a distributed system, certification of entities makes it possible to circumscribe malicious behavior, such as cheating in games. Certification requires the use of a trusted third party and is traditionally done centrally. At a large scale, however, centralized certification represents a bottleneck and a single point of attack or failure. We have proposed solutions based on distributed reputations to identify trusted nodes and use them as game referees to detect and prevent cheating [46]. Our method relies on previous work on the subject of trusted node collaboration to ensure reliable distributed certification [4].

6.3. Management of distributed data

Participants: Mesaac Makpangou, Olivier Marin, Sébastien Monnet, Pierre Sens, Marc Shapiro, Julien Sopena, Gaël Thomas, Pierpaolo Cincilla, Raluca Diaconu, Sergey Legtchenko, Jonathan Lejeune, Karine Pires, Thomas Preud homme, Masoud Saeida Ardekani, Guthemberg Silvestre, Pierre Sutra, Marek Zawirski, Annette Bieniusa, Pierpaolo Cincilla, Véronique Simon, Mathieu Valero.

Sharing information is one of the major reasons for the use of large-scale distributed computer systems. Replicating data at multiple locations ensures that the information persists despite the occurrence of faults, and improves application performance by bringing data close to its point of use, enabling parallel reads, and balancing load. This raises numerous issues: where to store or replicate the data, in order to ensure that it is available quickly and remains persistent despite failures and disconnections; how to ensure consistency between replicas; when and how to move data to computation, or computation to data, in order to improve response time while minimizing storage or energy usage; etc. The Regal group works on several key issues related to replication:

[3] C. Boulinier, F. Petit, and V. Villain. Synchronous vs. asynchronous unison. Algorithmica, 51(1):61-80, 2008

[4] Erika Rosas, Olivier Marin and Xavier Bonnaire. CORPS: Building a Community Of Reputable PeerS in Distributed Hash Tables. The Computer Journal, 54(10):1721-1735(2011)

- Replica placement for fault tolerance and latency in the presence of churn,
- scalable strong consistency for replicated databases, and
- theory and practice of eventual consistency.

6.3.1. Distributed hash tables

A DHTs replicates data and spreads the replicas uniformly across a large number of nodes. Being very scalable and fault-tolerant, DHTs are a key component for dependable and secure applications, such as backup systems, distributed file systems, multi-range query systems, and content distribution systems.

Despite the advantages of DHTs, several studies show that they become inefficient in environments subject to churn, i.e., with many node arrivals and departures. We therefore propose a new replication mechanism for DHTs that is churn resilient [20]. RelaxDHT relaxes placement constraints, in order to avoid redundant data transfers and to increase parallelism. RelaxDHT loses up to 50% fewer data blocks that the well-known PAST DHT.

6.3.2. Strong consistency

When data is updated somewhere on the network, it may become inconsistent with data elsewhere, especially in the presence of concurrent updates, network failures, and hardware or software crashes. A primitive such as consensus (or equivalently, total-order broadcast) synchronises all the network nodes, ensuring that they all observe the same updates in the same order, thus ensuring strong consistency. However the latency of consensus is very large in wide-area networks, directly impacting the response time of every update. Our contributions consist mainly of leveraging application-specific knowledge to decrease the amount of synchronisation.

To reduce the latency of consensus, we study *Generalised Consensus* algorithms, i.e., ones that leverage the commutativity of operations or the spontaneous ordering of messages by the network. We propose a novel protocol for generalised consensus that is optimal, both in message complexity and in faults tolerated, and that switches optimally between its fast path (which avoids ordering commuting requests) and its classical path (which generates a total order). Experimental evaluation shows that our algorithm is much more efficient and scales better than competing protocols.

When a database is very large, it pays off to replicate only a subset at any given node; this is known as partial replication. This allows non-overlapping transactions to proceed in parallel at different locations and decreases the overall network traffic. However, this makes it much harder to maintain consistency. We designed and implemented two *genuine* consensus protocols for partial replication, i.e., ones in which only relevant replicas participate in the commit of a transaction.

Another research direction leverages isolation levels, particularly Snapshot Isolation (SI), in order to parallelize non-conflicting transactions on databases. We prove a novel impossibility result, namely that a system cannot have both genuine partial replication and SI. We designed an efficient protocol that maintains the most important features of SI, but side-steps this impossibility. Finally, we study the trade-offs between freshness (and hence low abort rates) and space complexity in computing snapshots, as required by SI and its variants.

Parallel transactions in distributed DBs incur high overhead for concurrency control and aborts. Our Gargamel system proposes an alternative approach by pre-serializing possibly conflicting transactions, and parallelizing non-conflicting update transactions to different replicas. It system provides strong transactional guarantees. In effect, Gargamel partitions the database dynamically according to the update workload. Each database replica runs sequentially, at full bandwidth; mutual synchronisation between replicas remains minimal. Our simulations show that Gargamel improves both response time and load by an order of magnitude when contention is high (highly loaded system with bounded resources), and that otherwise slow-down is negligible. This is published at ICPADS 2012 [27].

Our current experiments aim to compare the practical pros and cons of different approaches to designing large-scale replicated databases, by implementing and benchmarking a number of different protocols.

Our study the trade-offs between freshness and meta-date overhead, is published in HotCDP 2012 [43].

6.3.3. Eventual consistency

Eventual Consistency (EC) aims to minimize synchronisation, by weakening the consistency model. The idea is to allow updates at different nodes to proceed without any synchronisation, and to propagate the updates asynchronously, in the hope that replicas converge once all nodes have received all updates. EC was invented for mobile/disconnected computing, where communication is impossible (or prohibitively costly). EC also appears very appealing in large-scale computing environments such as P2P and cloud computing. However, its apparent simplicity is deceptive; in particular, the general EC model exposes tentative values, conflict resolution, and rollback to applications and users. Our research aims to better understand EC and to make it more accessible to developers.

We propose a new model, called *Strong Eventual Consistency* (SEC), which adds the guarantee that every update is durable and the application never observes a roll-back. SEC is ensured if all concurrent updates have a deterministic outcome. As a realization of SEC, we have also proposed the concept of a Conflict-free Replicated Data Type (CRDT). CRDTs represent a sweet spot in consistency design: they support concurrent updates, they ensure availability and fault tolerance, and they are scalable; yet they provide simple and understandable consistency guarantees.

This new model is suited to large-scale systems, such as P2P or cloud computing. For instance, we propose a "sequence" CRDT type called Treedoc that supports concurrent text editing at a large scale, e.g., for a wikipedia-style concurrent editing application. We designed a number of CRDTs such as counters (supporting concurrent increments and decrements), sets (adding and removing elements), graphs (adding and removing vertices and edges), and maps (adding, removing, and setting key-value pairs). In particular, we publish a study of the concurrency semantics of sets in DISC 2012 [48], [22].

On the theoretical side, we identified sufficient correctness conditions for CRDTs, viz., that concurrent updates commute, or that the state is a monotonic semi-lattice. CRDTs raise challenging research issues: What is the power of CRDTs? Are the sufficient conditions necessary? How to engineer interesting data types to be CRDTs? How to garbage collect obsolete state without synchronisation, and without violating the monotonic semi-lattice requirement?

We are currently developing a very large-scale CRDT platform called SwiftCloud, which aims to scale to millions of clients, deployed inside and outside the cloud.

6.4. Improving the Performance and Robustness of Systems Software in Multicore Architectures

6.4.1. Managed Runtime Environments

Participants: Bertil Folliot, Julia Lawall, Gilles Muller [correspondent], Marc Shapiro, Julien Sopena, Gaël Thomas, Florian David, Lokesh Gidra, Jean-Pierre Lozi, Thomas Preud homme, Suman Saha, Harris Bakiras, Arie Middelkoop, Koutheir Attouchi.

Today, multicore architectures are becoming ubiquitous, found even in embedded systems, and thus it is essential that managed languages can scale on multicore processors. We have found that a major scalability bottleneck is the implementation of high contention locks, which can overload the bus, eliminating all performance benefits from adding more cores. To address this issue, as part of the PhD of Jean-Pierre Lozi, we have developed remote core locking (RCL), in which highly contended locks are implemented on a dedicated server, minimizing bus traffic and improving application scalability (USENIX ATC 2012 [24]). This work initially targeted C code but is now being adapted to the needs of Java applications in the PhD of Florian David. Another bottleneck in the support for managed languages is the garbage collector. As part of the PhD of Lokesh Gidra, we have identified the main sources of overhead.

6.4.2. Systems software robustness

A new area of research for Regal, with the arrival of Gilles Muller in 2009 as Inria Senior Research Scientist and Julia Lawall in 2011 as Inria Senior Research Scientist, is on improving the reliability of operating systems code. Muller and Lawall previously developed Coccinelle, a scriptable program matching and transformation tool for C code that is now commonly used in the open-source development community, including by the developers of Linux, Wine and Dragonfly BSD. Based on Coccinelle, we have developed a new approach to inferring API function usage protocols from software, relying on knowledge of common code structures (Software – Practice and Experience [19]).

We have also proposed a method for automatically identifying bug-fixing patches, with the goal of helping developers maintain stable versions of the software (ICSE 2012 [45]) and have designed an approach to automatically generating a robust interface to the Linux kernel, to provide developers of new kernel-level code more feedback in the case of a misunderstanding of kernel API usage conventions (ASE 2012 [24]).

7. Partnerships and Cooperations

7.1. National initiatives

7.1.1. InfraJVM - (2012–2015)

Members: LIP6 (Regal), Ecole des Mines de Nanes (Constraint), IRISA (Triskell), LaBRI (LSR).

Funding: ANR Infra.

Objectives: The design of the Java Virtual Machine (JVM) was last revised in 1999, at a time when a single program running on a uniprocessor desktop machine was the norm. Today's computing environment, however, is radically different, being characterized by many different kinds of computing devices, which are often mobile and which need to interact within the context of a single application. Supporting such applications, involving multiple mutually untrusted devices, requires resource management and scheduling strategies that were not planned for in the 1999 JVM design. The goal of InfraJVM is to design strategies that can meet the needs of such applications and that provide the good performance that is required in an MRE.

The coordinator of InfraJVM is Gaël Thomas. Infra-JVM brings a grant of 202 000 euros from the ANR to UPMC over three years.

7.1.2. ODISEA2 - (2011–2014)

Members: Orange, LIP6 (Regal), UbiStorage, Technicolor, Institut Telecom

Funding: FUI project, Ile de France Region

Objectives: ODISEA aims at designing new on-line data storage and data sharing solutions. Current solutions rely on big data centers, which induce many drawbacks: (i) a high cost, (ii) proprietary solutions, (iii) inefficiency (one single location, not necessarily close to the user). The goal is to tackle these issues by designing a distributed/decentralized solution that leverage edge resources like set-top boxes.

It involves a grant of 159 000 euros from Region Ile de France over three years.

7.1.3. MyCloud - (2011–2014)

Members: Inria Rhones-Alpes (SARDES), LIP6 (REGAL), EMN, WeAreCloud, Elastic Cloud.

Funding: MyCloud project is funded by ANR Arpège.

Objectives: Cloud Computing is a paradigm for enabling remote, on-demand access to a set of configurable computing resources. The objective of the MyCloud project is to define and implement a novel cloud model: SLAaaS (SLA aware Service). Novel models, control laws, distributed algorithms and languages will be proposed for automated provisioning, configuration and deployment of cloud services to meet SLA requirements, while tackling scalability and dynamics issues. The principal investigators for Regal are Luciana Arantes, Pierre Sens, and Julien Sopena. It involves a grant of 155 000 euros from ANR to LIP6 over three years.

7.1.4. ConcoRDanT - (2010–2013)

Members: Inria Regal, project leader; LORIA, Universdide Nova de Lisboa

Funding: ConcoRDanT is funded by ANR Blanc.

Objectives: CRDTs for consistency without concurrency control in Cloud and Peer-To-Peer systems. Massive computing systems and their applications suffer from a fundamental tension between scalability and data consistency. Avoiding the synchronisation bottleneck requires highly skilled programmers, makes applications complex and brittle, and is error-prone. The ConcoRDanT project investigates a promising new approach that is simple, scales indefinitely, and provably ensures eventual consistency. A Commutative Replicated Data Type (CRDT) is a data type where all concurrent operations commute. If all replicas execute all operations, they converge; no complex concurrency control is required. We have shown in the past that CRDTs can replace existing techniques in a number of tasks where distributed users can update concurrently, such as co-operative editing, wikis, and version control. However CRDTs are not a universal solution and raise their own issues (e.g., growth of meta-data). The ConcoRDanT project engages in a systematic and principled study of CRDTs, to discover their power and limitations, both theoretical and practical. Its outcome will be a body of knowledge about CRDTs and a library of CRDT designs, and applications using them. We are hopeful that significant distributed applications can be designed using CRDTs, a radical simplification of software, elegantly reconciling scalability and consistency. The project leader and principal investigator for Regal is Marc Shapiro. ConcoRDanT involves a grant of 192 637 euros from ANR to Inria over three years.

7.1.5. SPADES - (2009–2012)

Members: LIP, MIS (and LIP6/REGAL), Inria Rennes, Inria Saclay, LIG, LUG, CERFACS, IN2P3

Funding: ANR CONTINT

Objectives: The main goal of SPADES is to propose a non-intrusive but highly dynamic environment, able to take advantages to available resources over very large scale grids. Another challenge of SPADES is to provide a software solution for a service discovery system able to face a highly dynamic platform. This system will be deployed over volatile nodes and thus must tolerate “failures”.

The principal investigator for Regal is Franck Petit. The project was initiated while he was with MIS (UPJV/Amiens) and a non-permanent researcher during 2008-2009 with Inria, within Graal Team (LIP, Lyon). The amount of the grant from ANR to MIS is 125 000 euros.

7.1.6. STREAMS - (2010–2013)

Members: LORIA (Score, Cassis), Inria (Regal, ASAP), Xwiki.

Funding: STREAMS is funded by ANR Arpège.

Objectives: Solutions for a peer-To-peer REAl-tiMe Social web The STREAMS project proposes to design peer-to-peer solutions that offer underlying services required by real-time social web applications and that eliminate the disadvantages of centralised architectures. These solutions are meant to replace a central authority-based collaboration with a distributed collaboration that offers support for decentralisation of services. The project aims to advance the state of the art on peer-to-peer networks for social and real-time applications. Scalability is generally considered as an inherent characteristic of peer-to-peer systems. It is traditionally achieved using replication techniques. Unfortunately, the

current state of the art in peer-to-peer networks does not address replication of continuously updated content due to real-time user changes. Moreover, there exists a tension between sharing data with friends in a social network deployed in an open peer-to-peer network and ensuring privacy. One of the most challenging issues in social applications is how to balance collaboration with access control to shared objects. Interaction is aimed at making shared objects available to all who need them, whereas access control seeks to ensure this availability only to users with proper authorisation. STREAMS project aims at providing theoretical solutions to these challenges as well as practical experimentation. The principal investigators for Regal is Marc Shapiro. It involves a grant of 57 000 euros from ANR to Inria over three years.

7.1.7. PROSE - (2009–2012)

Members: Technicolor, Inria (Regal), EURECOM, PlayAdz, LIAFA.

Funding: PROSE project is funded by ANR VERSO.

Objectives: Content Shared Through Peer-to-Peer Recommendation & Opportunistic Social Environment.

The Prose project is a collective effort to design opportunistic contact sharing schemes, and characterizes the environmental conditions as well as algorithmic and architecture principles that let them operate. The partners of the Prose project will engage in this exploration through various expertise: network measurement, system design, behavioral study, analysis of distributed algorithms, theory of dynamic graph, networking modeling, and performance evaluation.

The principal investigators for Regal are Sébastien Monnet and Marc Shapiro. It involves a grant of 152 000 euros from ANR to Inria over three years.

7.1.8. ABL - (2009–2012)

Members: Gilles Muller, Julia Lawall, Gaël Thomas, Saha Suman.

Funding: ANR Blanc.

Objectives: The goal of the "A Bug's Life" (ABL) project is to develop a comprehensive solution to the problem of finding bugs in API usage in open source infrastructure software. The ABL project has grown out of our experience in using the Coccinelle code matching and transformation tool, which we have developed as part of the former ANR project Blanc Coccinelle, and our interactions with the Linux community. Coccinelle targets the problem of documenting and automating collateral evolutions in C code, specifically Linux code. A collateral evolution is a change that is needed in the clients of an API when the API changes in some way that affects its interface. Coccinelle provides a language for expressing collateral evolutions by means of Semantic Patches, and a transformation tool for performing them automatically.

The main achievements of the ABL project in 2012 include the design of an approach to automatically generating a robust interface to the Linux kernel, which received a best paper award at ASE 2012, and the design of an approach to finding resource-release omission faults in systems software. The latter has led to over 60 patches for various systems software projects, including Linux and Python.

7.2. European Initiatives

7.2.1. FP7 Projects

7.2.2. Collaborations in European Programs, except FP7

7.2.2.1. Google European Doctoral Fellowship "A principled approach to eventual consistency based on CRDTs

Cloud computing systems suffer from a fundamental tension between scalability and data consistency. Avoiding the synchronisation bottleneck requires highly skilled programmers, makes applications complex and brittle, and is error-prone. The Commutative Replicated Data Type (CRDT) approach, based on commutativity,

is a simple and principled solution to this conundrum; however, only a handful of CRDTs are known, and CRDTs are not a universal solution. This PhD research aims to expand our knowledge of CRDTs, to design and implement a re-usable library of composable CRDTs, to maintain study techniques for maintaining strong invariants above CRDTs, and to experiment with CRDTs in applications. We are hopeful that significant distributed applications can be designed using our techniques, which would radically simplify the design of cloud software, reconciling scalability and consistency. This Google European Doctoral Fellowship is awarded to Marek Zawirski, advised by Marc Shapiro. This award includes a grant of 41 000 euros yearly over three years starting September 2010.

7.3. International Initiatives

7.3.1. Participation In International Programs

7.3.1.1. Dependability of dynamic distributed systems for ad-hoc networks and desktop grid (ONDINA) (2011-2013)

Members: Inria Paris Rocquencourt (REGAL), Inria Rhone-Alpes (GRAAL), UFBA (Bahia, Brazil))

Funding: Inria

Objectives: Modern distributed systems deployed over ad-hoc networks, such as MANETs (wireless mobile ad-hoc networks), WSNs (wireless sensor networks) or Desktop Grid are inherently dynamic and the issue of designing reliable services which can cope with the high dynamics of these systems is a challenge. This project studies the necessary conditions, models and algorithms able to implement reliable services in these dynamic environments.

7.3.1.2. Enabling Collaborative Applications For Desktop Grids (ECADeG) (2011–2013)

Members: Inria Paris Rocquencourt (REGAL), USP (Sao Paulo, Brazil))

Funding: Inria

Objectives: The overall objective of the ECADeG research project is the design and implementation of a desktop grid middleware infrastructure for supporting the development of collaborative applications and its evaluation through a case study of a particular application in the health care domain.

7.4. International Research Visitors

7.4.1. Visits of International Scientists

- Kenji Kono, Professor, University Keio, Japan, 1 year, 2012
- Nuno Preguia, Associate Professor, Universidade Nova de Lisboa; 6-month visit
- Valter Balegas, PhD Student, Universidade Nova de Lisboa; 3-month visit

7.4.2. Internships

- David Navalho, PhD Student, Universidade Nova de Lisboa; 3-month visit
- Valter Balegas, PhD Student, Universidade Nova de Lisboa; 3-month visit

8. Dissemination

8.1. Scientific Animation

Luciana Arantes is:

- co-chair of P2PDep: 1st Workshop on dependability on P2P systems in conjunction with EDCC 2012.
- PC member of ICPADS 2012, LADC 2013, P2P-DEP 2012, GPC 2013.
- Reviewer for Journal of Parallel and Distributed Computing (JPDC) and Future Generation Computer System

Bertil Folliot is:

- Member of the “Executive Comittee” of GdR ASR (Hardware, System and Network), CNRS.
- Elected member of the IFIP WG10.3 working group (International Federation for Information Processing - Concurrent systems).
- Member of the “Advisory Board” of EuroPar (International European Conference on Parallel and Distributed Computing), IFIP/ACM.
- Member of the “Steering Committee” of the International Symposium on Parallel and Distributed Computing" (ISPDC).
- PC member of ISPDC 2011, ISPDC 2012, ISPDC 2013.

Maria Potop-Butucaru is:

- Member of PC of ICDCS 2012 (IEEE 32nd International Conference on Distributed Computing Systems)
- Member of PC of SSS 2012(13th International Symposium on Stabilization, Safety, and Security of Distributed Systems), co-chair track Self-Stabilization

Olivier Marin is:

- Reviewer for Distributed Computing, and Techniques et Sciences Informatiques.
- Member of the scientific committee of LIP6.

Sébastien Monnet is:

- Elected member of the administrative committee of LIP6.
- Member of the program commitee of the 1st IEEE International Conference on Cloud Networking, CloudNet, Paris, November 2012.

Gilles Muller is:

- Distributed OS and Middleware track co-chair of ICDCS 2012
- Poster chair of DSN 2012
- PC member of SYSTOR 2012
- PC member of ICDCN 2013
- PC member of EuroSys 2013
- Member of "Comité de selection" of INSA de Lyon (Professeur) and University of Rennes (Professeur)
- Management Committee Substitute Member for the COST action "Transactional Memories: Foundations, Algorithms, Tools, and Applications (Euro-TM)" and leader of the working group on "Hardware’s and Operating System’s Supports"
- Reviewer for the Flanders Research Foundation
- Reviewer for the Swiss National Science Foundation
- Member of IFIP WG 10.4 (Dependability)

Julia Lawall is:

- Chair of the steering committee of Generative Programming and Component Engineering (2011, 2012)
- Secretary of IFIP TC2 (since 2011)
- Member-at-large of the SIGPLAN Executive Committee
- Member of the editorial board of Science of Computer Programming
- Associate editor of Higher Order and Symbolic Computation
- PC member of 21st International Conference on Compiler Construction (CC 2012)
- PC member of Modularity: AOSD 2012
- PC member of OOPSLA 2012
- PC member of OBT: Underrepresented Problems for PL Researchers, with POPL 2012
- PC member of PPDP 2012
- Invited reviewer for PLDI 2012
- Member of IFIP WG 2.11 (Program Generation)

Franck Petit is:

- Vice Chair for the Special Journal Issue (TCS) of Track "Distributed Computing" of ICDCN 2013, IEEE 33nd International Conference on Distributed Computing and Networking, Mumbai, India, janvier 2013.
- PC Member of Renpar'21, 21st Rencontres francophones du Parallélisme, Grenoble, France, janvier 2013.
- PC Member of SSS 2012, 14th International Symposium on Stabilization, Safety, and Security of Distributed Systems, ed. LNCS, Toronto.
- PC Member of ICDCN 2012, 32nd International Conference on Distributed Computing and Networking, Hong Kong, China.
- Invited Editor for TCS, Theoretical Computer Science Special Issue on Stabilization, Safety, and Security of Distributed Systems 2011.
- Invited Editor for TCS, Theoretical Computer Science Special Issue on Stabilization, Safety, and Security of Distributed Systems 2011.
- Co-chair of Department "Networks and Distributed Systems" of LIP6 Laboratory.
- Member of "Vivier d'experts" and member of "Comité de selection" of UPMC Paris 6.

Pierre Sens is:

- co-Chair of P2P-Dep 2012 (1st Workshop on P2P and Dependability in conjonction with EDCC 2012)
- Member of PC of ICDCS 2012 (IEEE 32nd International Conference on Distributed Computing Systems)
- Member of PC of of EDCC'2012 (9th European Dependable Computing Conference)
- Member of PC of of IPDPS'2013 (IEEE International Parallel & Distributed Processing Symposium)
- Member of "Directoire de la recherche" of University Pierre et Marie Curie.
- vice-chair of LIP6 Laboratory.
- Member of the scientific council of AFNIC.
- Member of the scientific committee of LIP6.
- Member of the evaluation committee of the Digiteo DIM LSC program.
- Member of "Comité de selection" of Universities of Grenoble and Paris XI.

Marc Shapiro is:

- Member of Advisory Board for CITI, the Research Center for Informatics and Information Technologies of UNL, the New University of Lisbon (Portugal).
- Member of the steering committee for the LADIS workshop (Large-Scale Distributed Systems and Middleware).
- Promotion reviewer for various European universities (names confidential).
- Reviewer for European Research Council.
- Reviewer for ANR (Agence Nationale de la Recherche), France.
- Reviewer for National Science Foundation, Switzerland.
- Reviewer for USA-Israel Binational Science Foundation.
- Reviewer for Swedish Research Council .
- Reviewer for Springer Distributed Computing.
- Reviewer for IEEE Transactions on Parallel and Distributed Systems (TPDS).
- Member, ACM Distinguished Service Award Committee 2010.
- Member, ACM Europe Council. Co-chair, ACM Europe Council subcommittee on Members and Awards.

Gaël Thomas is :

- DAIS 2012: IFIP 12th conference on Distributed Applications and Interoperable Systems, Stockholm, Sweden.
- Elected member of the administrative committee of LIP6.

8.2. Teaching - Supervision - Juries

8.2.1. Teaching

Licence : Gaël Thomas, Introduction to the C programming language, L1, Université Paris 6

Licence : Gaël Thomas, Introduction to computer architecture, L2, Université Paris 6

Licence : Luciana Arantes, Bertil Folliot, Maria Potop-Butucaru, Julien Sopena, Franck Petit, Principles of operating systems, L3, Université Paris 6

Licence : Mesaac Makpangou, Client/server architecture, L3 professionelle, Université Paris 6

Licence : Sébastien Monnet, System and Internet programmation, L2, Université Paris 6

Licence : Sébastien Monnet, Computer science initiation, L1, Université Paris 6

Master : Luciana Arantes, Sébastien Monnet, Pierre Sens, Julien Sopena, Gaël Thomas, Operating systems kernel, M1, Université Paris 6

Master : Luciana Arantes, O. Marin, Maria Potop-Butucaru, Distributed algorithms, M1, Université Paris 6

Master : Luciana Arantes, Oliver Marin, Pierre Sens, Advanced distributed algorithms, M2, Université Paris 6

Licence : Luciana Arantes, Bertil Folliot, Olivier Marin, POSIX Advanced C system programming, M1 d'Informatique, Université Paris 6

Master : Bertil Folliot, Julien Sopena, Distributed systems and client/serveur, M1 , Université Paris 6

Licence : Bertil Folliot, C programming & systems, L2, Université Paris 6

Licence : Bertil Folliot, Directed projects, L2, Université Paris 6

Licence : Bertil Folliot, Head of the Computer Courses "Applications of Computer Technology and Communication", L2, Université Paris 6

Master : Franck Petit, Maria Potop-Butucaru, Resistance of Distributed Attacks, M2, Université Paris 6,

Master : Luciana Arantes, Sébastien Monnet, Julien Sopena, Gaël Thomas, Middleware for advanced computing systems, M2, Université Paris 6"

Master : Marc Shapiro, Julien Sopena, Gaël Thomas, multicore kernels and virtualisation, M2, Université Paris 6

8.2.2. Supervision

HdR : Gaël Thomas, Amélioration du design et des performances des machines virtuelles langage, UPMC, 09/29/2012

PhD : S. Legtchenko, Exploiting player behavior in distributed architectures for online games, UPMC, 10/25/2012, Sébastien Monnet, Pierre Sens

8.2.3. Juries

Gilles Muller was the reviewer of:

- Maxime Lastera. PhD. INSA Toulouse, (Advisor : J. Arlat)
- Nicolas Berthier. PhD. University of Grenoble, (Advisor : F. Maraninchi)

Pierre Sens was the reviewer of:

- Kiril Georgiev. PhD. LIG, (Advisor : J-F Méhaut)
- Heverson Borba Ribeiro. PhD. IRISA, (Advisor : E. Ancaume)
- Khanh-Toan Tran. PhD. Evry, (Advisor: N. Agoulmine)

Franck Petit was the reviewer of:

- Thomas Morsellino, PhD LaBRI, Bordeaux (Advisor: Y. Métivier)
- Damien Imbs, PhD IRISA, Rennes (Advisor: M. Raynal)

Gaë Thomas was the reviewer of:

- Geoffroy Cogniaux, PhD LIFL, Lilles (Advisor: G. Grimaud)

Julia Lawall was member of the jury of:

- Veronica Uquillas Gomez. PhD. VUB-Univ. Lille, (Advisors : Theo D'Hondt, Stephane Ducasse)

9. Bibliography

Major publications by the team in recent years

[1] E. ANCEAUME, R. FRIEDMAN, M. GRADINARIU POTOP-BUTUCARU. *Managed Agreement: Generalizing two fundamental distributed agreement problems*, in "Inf. Process. Lett.", 2007, vol. 101, n^{o} 5, p. 190-198.

[2] L. ARANTES, D. POITRENAUD, P. SENS, B. FOLLIOT. *The Barrier-Lock Clock: A Scalable Synchronization-Oriented Logical Clock*, in "Parallel Processing Letters", 2001, vol. 11, n^{o} 1, p. 65–76.

[3] J. BEAUQUIER, M. GRADINARIU POTOP-BUTUCARU, C. JOHNEN. *Randomized self-stabilizing and space optimal leader election under arbitrary scheduler on rings*, in "Distributed Computing", 2007, vol. 20, n^{o} 1, p. 75-93.

[4] M. BERTIER, L. ARANTES, P. SENS. *Distributed Mutual Exclusion Algorithms for Grid Applications: A Hierarchical Approach*, in "JPDC: Journal of Parallel and Distributed Computing", 2006, vol. 66, p. 128–144.

[5] M. BERTIER, O. MARIN, P. SENS. *Implementation and performance of an adaptable failure detector*, in "Proceedings of the International Conference on Dependable Systems and Networks (DSN '02)", June 2002.

[6] M. BERTIER, O. MARIN, P. SENS. *Performance Analysis of Hierarchical Failure Detector*, in "Proceedings of the International Conference on Dependable Systems and Networks (DSN '03)", San-Francisco (USA), IEEE Society Press, June 2003.

[7] B. DUCOURTHIAL, S. KHALFALLAH, F. PETIT. *Best-effort group service in dynamic networks*, in "22nd Annual ACM Symposium on Parallel Algorithms and Architectures (SPAA)", 2010, p. 233-242.

[8] N. KRISHNA, M. SHAPIRO, K. BHARGAVAN. *Brief announcement: Exploring the Consistency Problem Space*, in "Symp. on Prin. of Dist. Computing (PODC)", Las Vegas, Nevada, USA, ACM SIGACT-SIGOPS, July 2005.

[9] S. LEGTCHENKO, S. MONNET, G. THOMAS. *Blue banana: resilience to avatar mobility in distributed MMOGs*, in "The 40th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN)", July 2010.

[10] O. MARIN, M. BERTIER, P. SENS. *DARX - A Framework For The Fault-Tolerant Support Of Agent S oftware*, in "Proceedings of the 14th IEEE International Symposium on Sofwat are Reliability Engineering (ISSRE '03)", Denver (USA), IEEE Society Press, November 2003.

[11] N. PALIX, G. THOMAS, S. SAHA, C. CALVÈS, J. LAWALL, G. MULLER. *Faults in Linux: Ten Years Later*, in "Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS 2011)", Newport Beach, CA, USA, March 2011.

[12] N. SCHIPER, P. SUTRA, F. PEDONE. *P-Store: Genuine Partial Replication in Wide Area Networks*, in "Symp. on Reliable Dist. Sys. (SRDS)", New Dehli, India, IEEE Comp. Society, October 2010, p. 214–224.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[13] S. LEGTCHENKO. *Exploiting player behavior in distributed architectures for online games*, Université Pierre et Marie Curie (UPMC), 2012, Type : Thèse de Doctorat – Soutenue le : 2012-10-25 – Dirigée par : Sens, Pierre – Encadrée par : MONNET Sébastien.

[14] G. THOMAS. *Improving the Design and the Performance of Managed Runtime Environments*, Université Pierre et Marie Curie (UPMC), 2012, Type : Habilitation à Diriger des Recherches – Soutenue le : 2012-09-28, Ph. D. Thesis.

Articles in International Peer-Reviewed Journals

[15] Y. DIEUDONNÉ, F. PETIT. *Self-stabilizing gathering with strong multiplicity detection*, in "Theoretical Computer Science", 2012, vol. 428, p. 47-57.

[16] S. DUBOIS, M. GRADINARIU POTOP-BUTUCARU, M. NESTERENKO, S. TIXEUIL. *Self-Stabilizing Byzantine Asynchronous Unison*, in "Journal of Parallel and Distributed Computing (JPDC)", 2012, vol. 72, n^{o} 7, p. 917-923.

[17] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Bounding the Impact of Unbounded Attacks in Stabilization*, in "IEEE Transactions on Parallel and Distributed Systems (TPDS)", March 2012, vol. 23, n^{o} 3, p. 460-466, http://doi.ieeecomputersociety.org/10.1109/TPDS.2011.158.

[18] F. GREVE, P. SENS, L. ARANTES, V. SIMON. *Eventually Strong Failure Detector with Unknown Membership*, in "The Computer Journal", 2012.

[19] J. LAWALL, J. BRUNEL, N. PALIX, R. R. HANSEN, H. STUART, G. MULLER. *WYSIWIB: Exploiting Fine-Grained Program Structure in a Scriptable API-Usage Protocol Finding Process*, in "Software:Practice and Experience", January 2012, Online preprint.

[20] S. LEGTCHENKO, S. MONNET, P. SENS, G. MULLER. *RelaxDHT : A Churn Resilient for Peer-to-Peer Distributed Hash-Tables*, in "TAAS", July 2012, vol. 7, n^{o} 2, http://dx.doi.org/10.1145/2240166.2240178.

Articles in National Peer-Reviewed Journals

[21] L. ARANTES, J. LEJEUNE, M. PIFFARETTI, O. MARIN, P. SENS, J. SOPENA, A. N. BESSANI, V. V. COGO, M. CORREIA, P. COSTA, M. PASIN. *Vers une plate-forme MapReduce tolérant les fautes byzantines*, in "Technique et Science Informatiques (TSI)", 2012, To appear.

International Conferences with Proceedings

[22] A. BIENIUSA, M. ZAWIRSKI, N. PREGUIÇA, M. SHAPIRO, C. BAQUERO, V. BALEGAS, S. DUARTE. *Brief Announcement: Semantics of Eventually Consistent Replicated Sets*, in "Int. Symp. on Dist. Comp. (DISC)", Salvador, Bahia, Brazil, October 2012, p. 449–450, http://lip6.fr/Marc.Shapiro/papers/semantics-sets-BA-DISC-2012.pdf.

[23] T. F. BISSYANDÉ, L. RÉVEILLÈRE, J. LAWALL, G. MULLER. *Diagnosys: Automatic Generation of a Debugging Interface to the Linux Kernel*, in "Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering", Essen, Germany, September 2012, p. 60-69, Best Paper award [*DOI :* 10.1145/2351676.2351686], http://hal.inria.fr/hal-00731064.

[24] *Best Paper*
T. F. BISSYANDÉ, L. RÉVEILLÈRE, J. LAWALL, G. MULLER. *Diagnosys: Automatic Generation of a Debugging Interface to the Linux Kernel*, in "27th IEEE/ACM International Conference on Automated Software Engineering (ASE 2012)", IEEE, September 2012.

[25] F. BONNET, X. DÉFAGO, F. PETIT, M. GRADINARIU POTOP-BUTUCARU, S. TIXEUIL. *Brief Announcement: Discovering and Assessing Fine-grained Metrics in Robot Networks Protocols*, in "Proceedings of the International Conference on Stabilization, Safety, and Security in Distributed Systems (SSS 2012)", Toronto, Canada, Lecture Notes in Computer Science (LNCS), Springer Berlin / Heidelberg, October 2012.

[26] E. CARON, F. CHUFFART, A. LAMANI, F. PETIT. *Optimization in a Self-Stabilizing Service Discovery Framework for Large Scale Systems*, in "Proceedings of the International Conference on Stabilization, Safety, and Security in Distributed Systems (SSS 2012)", Toronto, Canada, Lecture Notes in Computer Science (LNCS), Springer Berlin / Heidelberg, October 2012.

[27] P. CINCILLA, S. MONNET, M. SHAPIRO. *Gargamel: boosting DBMS performance by parallelising write transactions*, in "18th IEEE International Conference on Parallel and Distributed Systems (ICPADS'12)", December 2012.

[28] G. DA SILVA SILVESTRE, P. SENS, S. MONNET, R. KRISHNASWAMY. *Caju: a content distribution system for edge networks*, in "1st Workshop on Big Data Management in Clouds, in Conjunction with Euro-Par 2012", August 2012, http://hal.inria.fr/hal-00727822.

[29] S. DEVISMES, A. LAMANI, F. PETIT, P. RAYMOND, S. TIXEUIL. *Optimal Grid Exploration by Asynchronous Oblivious Robots*, in "Proceedings of the International Conference on Stabilization, Safety, and Security in Distributed Systems (SSS 2012)", Toronto, Canada, Lecture Notes in Computer Science (LNCS), Springer Berlin / Heidelberg, October 2012.

[30] S. DUBOIS, S. TIXEUIL, N. ZHU. *Mariages et Trahisons*, in "AlgoTel'12 - 14èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications", La Grande Motte, France, F. MATHIEU, N. HANUSSE (editors), April 2012, p. 1-4, http://hal.inria.fr/hal-00689348.

[31] S. DUBOIS, S. TIXEUIL, N. ZHU. *The Byzantine Brides Problem*, in "Proceedings of the Sixth International Conference on Fun with Algorithms (FUN 2012)", Venice, Lecture Notes in Computer Science (LNCS), Springer Berlin / Heidelberg, June 2012.

[32] F. GREVE, M. SANTOS DE LIMA, L. ARANTES, P. SENS. *A Time-Free Byzantine Failure Detector for Dynamic Networks*, in "Ninth European Dependable Computing Conference", May 2012.

[33] P. HEIDEGGER, A. BIENIUSA, P. THIEMAN. *Access Permission Contracts for Scripting Languages*, in "39th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages (POPL)", ACM, January 2012.

[34] N. HIDALGO, E. ROSAS, L. ARANTES, O. MARIN, P. SENS, X. BONNAIRE. *Optimized range queries for large scale networks*, in "26th IEEE International Conference on Advanced Information Networking and Applications (AINA-2012)", March 2012.

[35] R. HU, J. SOPENA, L. ARANTES, P. SENS, I. DEMEURE. *Fair Comparison of Gossip Algorithms over Large-Scale Random Topologies*, in "31th IEEE International Symposium on Reliable Distributed Systems (SRDS'12)", IEEE Computer Society Press, October 2012.

[36] A. LAMANI, A. COURNIER, S. DUBOIS, F. PETIT, V. VILLAIN. *Snap-Stabilizing Message Forwarding Algorithm on Tree Topologies*, in "13th International Conference on Distributed Computing and Networking (ICDCN 2012)", Honk-Kong, China, 2012, p. 46-60.

[37] A. LAMANI, S. KAMEI, F. OOSHITA, S. TIXEUIL. *Gathering an even number of robots in a symmetric ring without global multiplicity detection*, in "Proceedings of the International Conference on Mathematical Foundations of Computer Science (MFCS 2012)", Bratislava, Slovakia, Lecture Notes in Computer Science (LNCS), Springer Berlin / Heidelberg, August 2012.

[38] J. LEJEUNE, L. ARANTES, J. SOPENA, P. SENS. *Service Level Agreement for Distributed Mutual Exclusion in Cloud Computing*, in "12th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing (CCGRID'12)", IEEE Computer Society Press, May 2012, p. 180-187.

[39] J.-P. LOZI, F. DAVID, G. THOMAS, J. LAWALL, G. MULLER. *Remote Core Locking: Migrating Critical-Section Execution to Improve the Performance of Multithreaded Applications*, in "USENIX Annual Technical Conference", USENIX, June 2012, p. 65-76.

[40] L. MILLET, M. LORRILLERE, L. ARANTES, S. GANÇARSKI, H. NAACKE, J. SOPENA. *Facing peak loads in a P2P transaction system*, in "Proceedings of the First Workshop on P2P and Dependability (P2PDEP'12)", New York, NY, USA, P2P-Dep '12, ACM, May 2012, p. 1-7, http://dx.doi.org/10.1145/2212346.2212347.

[41] M. NDIAYE NDEYE, P. SENS, O. THIARE. *Performance Comparison of Hierarchical Checkpoint Protocols on Grid Computing*, in "9th International Conference, Distributed Computing and Artificial Intelligence", Springer Verlag, March 2012.

[42] T. PREUD'HOMME, J. SOPENA, G. THOMAS, B. FOLLIOT. *An improvement of OpenMP pipeline parallelism with the BatchQueue algorithm*, in "18th IEEE International Conference on Parallel and Distributed Systems (ICPADS'12)", IEEE Computer Society Press, December 2012.

[43] M. SAEIDA ARDEKANI, M. ZAWIRSKI, P. SUTRA, M. SHAPIRO. *The Space Complexity of Transactional Interactive Reads*, in "Int. W. on Hot Topics in Cloud Data Processing (HotCDP)", Bern, Switzerland, April 2012, http://lip6.fr/Marc.Shapiro/papers/interactive-reads-HotCDP-2012.pdf.

[44] G. SILVESTRE, S. MONNET, R. KRISHNASWAMY, P. SENS. *AREN: a popularity aware replication scheme for cloud storage*, in "18th IEEE International Conference on Parallel and Distributed Systems (ICPADS'12)", December 2012.

[45] Y. TIAN, J. LAWALL, D. LO. *Identifying Linux Bug Fixing Patches*, in "34th International Conference on Software Engineering (ICSE 2012)", Zurich, Switzerland, ACM/IEEE, June 2012, p. 386-396.

[46] M. VÉRON, O. MARIN, S. MONNET, Z. GUESSOUM. *Towards a scalable refereeing system for online gaming*, in "11th International Workshop on Network and Systems Support for Games (NetGames'2012) (Poster)", November 2012.

National Conferences with Proceeding

[47] S. DUBOIS, S. TIXEUIL, N. ZHU. *Mariages et Trahisons*, in "Proceedings of Algotel 2012", La Grande Motte, France, May 2012.

Research Reports

[48] A. BIENIUSA, M. ZAWIRSKI, N. PREGUIÇA, M. SHAPIRO, C. BAQUERO, V. BALEGAS, S. DUARTE. *An optimized conflict-free replicated set*, Inria, October 2012, n^{o} RR-8083, 12, http://hal.inria.fr/hal-00738680.

[49] M. SAEIDA ARDEKANI, P. SUTRA, N. PREGUIÇA, M. SHAPIRO. *Non-Monotonic Snapshot Isolation*, Inria, October 2012, n^{o} RR-7805, 34, http://hal.inria.fr/hal-00643430.

[50] G. SILVESTRE, S. MONNET, R. KRISHNASWAMY, P. SENS. *Caju: a content distribution system for edge networks*, Inria, June 2012, n^{o} RR-8006, http://hal.inria.fr/hal-00712990.