IN PARTNERSHIP WITH:
CNRS

Université Pierre et Marie Curie (Paris 6)

Activity Report 2011

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, Data Storage, Peer-to-Peer

Regal is a joint research group with CNRS and Université Pierre et Marie Curie (Paris 6) through the Laboratoire d'Informatique de Paris 6, LIP6 *(UMR 7606).*

1. Members

Research Scientists

Julia Lawall [DR Inria since October]
Gilles Muller [DR Inria, HdR]
Marc Shapiro [DR Inria, HdR]
Mesaac Makpangou [CR Inria, 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, 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]

Technical Staff

Arie Middlekoop [Université Paris 6 since September]
Ikram Chabbouh [Université Paris 6 since October]
Harris Bakiras [since September]

PhD Students

Pierpaolo Cincilla [Université Paris 6 since September]
Lamia Benmouffok [Université Paris 6 until August]
Mathieu Bouillaguet [Université Paris 6 until August]
Raluca Diaconu [Université Paris 6 - CIFRE Orange Labs since November]
Swan Dubois [Université Paris 6 - PhD December 2011]
Lokesh Gidra [Université Paris 6 since August]
Nicolas Hidalgo [Université Paris 6 - PhD November 2011]
Anissa Lamani [Université Amiens]
Sergey Legtchenko [Université Paris 6]
Jean-Pierre Lozi [Université Paris 6 since September]
Corentin Méhat [Université Paris 6]
Thomas Preud'homme [Université Paris 6]
Erika Rosas [Université Paris 6 - PhD November 2011]
Masoud Saeida Ardekani [Université Paris 6]
Guthemberg Silvestre [Université Paris 6 - CIFRE Orange Labs]
Suman Saha [Université Paris 6 since September]
Mathieu Valero [Université Paris 6 - PhD December 2011]
Marek Zawirski [Université Paris 6 since September]

Post-Doctoral Fellows

Pierre Sutra

Annette Bieniusa [since September]

Administrative Assistant

Hélène Milome [Secretary]

2. Overall Objectives

2.1. Overall Objectives

The main focus of the Regal team is research on large-scale distributed computing systems, and 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, of dynamic content distribution, and of operating system adaptation. Regal is a joint research team between LIP6 and INRIA-Paris-Rocquencourt.

3. Scientific Foundations

3.1. Fondation 1

Scaling to large configurations is one of the major challenges addressed by the distributed system community lately. The basic idea is how to efficiently and transparently use and manage resources of millions of hosts spread over a large network. The problem is complex compared to classical distributed systems where the number of hosts is low (less than a thousand) and the inter-host links are fast and relatively reliable. In such "classical" distributed architectures, it is possible and reasonable to build a single image of the system so as to "easily" control resource allocation.

In large configurations, there is no possibility to establish a global view of the system. The underlying operating system has to make decisions (on resource allocation, scheduling ...) based only on partial and possibly wrongs view of the resources usage.

Scaling introduces the following problems:

- Failure: as the number of hosts increases, the probability of a host failure converges to one. [1] Compared to classical distributed systems, failures are more common and have to be efficiently processed.
- Asynchronous networks: on the Internet, message delays vary considerably and are unbounded.
- Impossibility of consensus:In such an asynchronous network with failures, consensus cannot be solved deterministically (the famous Fischer-Lynch-Patterson impossibility result of 1985).The system can only approximate, suspecting hosts that are not failed, or failing to suspect hosts that have failed. As a result, no host can form a consistent view of system state.
- Failure models: the classical view of distributed systems considers only crash and omission failures. In the context of large-scale, open networks, the failure model must be generalised to include stronger attacks. For instance, a host can be taken over ("zombie") and become malicious. Arbitrary faults, so-called Byzantine behaviours, are to be expected and must be tolerated.
- Managing distributed state: In contrast to a local-area network, establishing a global view of a large distributed system system is unfeasible. The operating system must make its decisions, regarding resource allocation or scheduling, based on partial and incomplete views of system state.

[1] For instance if we consider a classical host MTBF (Mean Time Between Failure) equals to 13 days, in a middle scale system composed of only 10000 hosts, a failure will occur every 4 minutes.

Three architectures in relation with the scaling problem have emerged during the last years:

Grid computing: Grid computing offers a model for solving massive computational problems using large numbers of computers arranged as clusters interconnected by a telecommunications infrastructure as internet or renater.

If the number of involved hosts can be high (several thousands), the global environment is relatively controlled and users of such systems are usually considered safe and only submitted to host crash failures (typically, Byzantine failures are not considered).

Peer-to-peer overlay network: Generally, a peer-to-peer (or P2P) computer network is any network that does not rely on dedicated servers for communication but, instead, mostly uses direct connections between clients (peers). A pure peer-to-peer network does not have the notion of clients or servers, but only equal peer nodes that simultaneously function as both "clients" and "servers" with respect to the other nodes on the network.

This model of network arrangement differs from the client-server model where communication is usually relayed by the server. In a peer-to-peer network, any node is able to initiate or complete any supported transaction with any other node. Peer nodes may differ in local configuration, processing speed, network bandwidth, and storage capacity.

Different peer-to-peer networks have varying P2P overlays. In such systems, no assumption can be made on the behavior of the host and Byzantine behavior has to be considered.

Cloud computing Cloud computing offers a conceptually infinite amount of computing, storage and network resources for rent. From a user's perspective, cloud computing has many advantages, including low upfront investment and outsourcing of system administration. From the provider's perspective, cloud computing shares many characteristics with both grid computing (e.g., very large geographical and numeric scale, or service-oriented interfaces) and with P2P computing (e.g., self-administration). It also has some unique characteristics, such as systematic virtualisation of all resources, highly variable load, fast elastic adaptation, and quality-of-service objectives that are negociated with clients via SLAs.

Regal is interested in how to adapt distributed middleware to these large scale configurations. We target Grid and Peer-to-peer configurations. This objective is ambitious and covers a large spectrum. To reduce its spectrum, Regal focuses on fault tolerance, replication management, and dynamic adaptation.

We concentrate on the following research themes:

Data management: the goal is to be able to deploy and locate effectively data while maintening the required level of consistency between data replicas.

System monitoring and failure detection: we envisage a service providing the follow-up of distributed information. Here, the first difficulty is the management of a potentially enormous flow of information which leads to the design of dynamic filtering techniques. The second difficulty is the asynchronous aspect of the underlying network which introduces a strong uncertainty on the collected information.

Adaptive replication: we design parameterizable techniques of replication aiming to tolerate the faults and to reduce information access times. We focus on the runtime adaptation of the replication scheme by (1) automatically adjusting the internal parameters of the strategies and (2) by choosing the replication protocol more adapted to the current context.

The dynamic adaptation of application execution support: the adaptation is declined here to the level of the execution support (in either of the high level strategies). We thus study the problem of dynamic configuration at runtime of the low support layers.

4. Application Domains

4.1. Application Domains

As we already mentioned, we focus on two kinds of large scale environments: clouds and peer-to-peer (P2P) systems. Although both environments have the same final objective of sharing large sets of resources, they initially emerged from different communities with different context assumptions and hence they have been designed differently. Clouds provide support for a large number of services needed by scientific communities. They usually target thousands of hosts and hundreds of users. Peer-to-peer environments address millions of hosts with hundreds of thousands of simultaneous users but they offer limited and specialized functionalities (file sharing, parallel computation).

In peer-to-peer configurations we focus on the following applications:

- Internet services such as web caches or content distribution network (CDN) which aim at reducing the access time to data shared by many users,
- Data storage of mutable data. Data storage is a classical peer-to-peer application where users can share documents (audio and video) across the Internet. A challenge for the next generation of data sharing systems is to provide update management in order to develop large cooperative applications.
- multi-player games. The recent involvement of REGAL in the PLAY ALL project gives us the opportunity to consider distributed interactive video games. Theses applications are very interesting for us since they bring new constraints, most specifically on latency.

Our second application domain is based on data sharing. Whereas most work on P2P applications focuses on write-once single-writer multiple-reader applications, we consider the (more demanding) applications that share mutable data in large-scale distributed settings. Some examples are co-operative engineering, collaborative authoring, or entreprise information libraries: for instance co-operative code development tools or decentralized wikis. Such applications involve users working from different locations and at different times, and for long durations. In such settings, each user *optimistically* modifies his private copy, called a replica, of a shared datum. As replicas may diverge, this poses the problem of reconciliation. Our research takes into account a number of issues not addressed by previous work, for instance respecting application semantics, high-level operations, dependence, atomicity and conflict, long session times, etc.

5. Software

5.1. Coccinelle

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

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 [51] 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. 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.

We report on application experience, building a collaborative application for model-oriented software engineering above Telex, in SAC 2011 [50]. Future work includes extending Telex to cloud computing, opportunistic mobile networks, and real-time collaboration, within several ANR projects: PROSE (Section 7.1.5), STREAMS (Section 7.1.4) and ConcoRDanT (Section 7.1.3).

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

5.3. Treedoc

Participants: Marc Shapiro [correspondent], Marek Zawirski.

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. This is the subject the PROSE, STREAMS and ConcoRDanT ANR projects (Sections 7.1.5, 7.1.4 and 7.1.3 respectively).

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

5.4. VMKit and .Net runtimes for LLVM

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

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 hash 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 project 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://vmkit.llvm.org/

6. New Results

6.1. Introduction

In 2011, we focused our research on the following areas:

- distributed algorithms for large and dynamic networks,
- Complex queries over peer-to-peer networks
- Trust and reputation management on P2P networks
- dynamic adaptation of virtual machines,
- services management in large scale environments,
- Formal and practical study of optimistic replication, incorporating application semantics.
- Decentralized commitment protocols for semantic optimistic replication.

6.2. Distributed algorithms

Participants: Luciana Arantes [correspondent], Franck Petit, Maria Potop-Butucaru [correspondent], Swan Dubois, Pierre Sens, Julien Sopena.

Our current research in the context of distributed algorithms focuses on two main axes. We are interested in providing fault-tolerant and self*(self-organizing, self-healing and self-stabilizing) solutions for fundamental problems in distributed computing. More precisely, we target the following basic blocks: mutual exclusion, resources allocation, agreement and communication primitives.

In dynamic systems we are interested in designing building blocks for distributed applications such as: failure detectors, adequate communication primitives (publish/subscribe) and overlays. Moreover, we are interested in solving fundamental problems such as synchronization, leader election, membership and naming, and diffusion of information.

6.2.1. Failure Detectors for Dynamic Systems

Since 2009, we explore a distributed computing model of dynamic networks such as (MANET or Wireless sensor networks). The temporal variations in the network topology implies that these networks can not be viewed as a static connected graph over which paths between nodes are established beforehand. Path between two nodes is in fact built over the time. Furthermore, lack of connectivity between nodes (temporal or not) makes of dynamic networks a *partitionable system*, i.e., a system in which nodes that do not crash or leave the system might be not capable to communicate between themselves. In 2011 we propose a new failure detector protocol which implements an eventually strong failure detectors ($\diamond S$) on a dynamic network with an unknown membership. Failure detector is a fundamental service, able to help in the development of fault-tolerant distributed systems. Our failure detector has the interesting feature to be time-free, so that it does not rely on timers to detect failures; moreover, it tolerates mobility of nodes and message losses [41].

6.2.2. Self- Distributed Algorithms*

The main challenges of our research activity over 2011 year were to develop self-* (self-stabilizing, self-organizing and self-healing) distributed algorithms for various type of networks. Self-stabilization is a general technique to design distributed systems that can tolerate arbitrary transient faults. Since topology changes can be considered as a transient failures, self-stabilization turns out to be a good approach to deal with dynamic networks. This is particularly relevant when the distributed (self-stabilizing) protocol does not require any global parameters, like the number of nodes or the diameter of the network. With such a self-stabilizing protocol, it is not required to change global parameters in the program when nodes join or leave the system. Therefore, self-stabilization is very desirable to achieve scalability and dynamicity.

- **Snap-stabilizing Committee Coordination.** The classic *committee coordination problem* characterizes a general type of synchronization called n-ary *rendezvous* as follows [2]:

[2] In *Parallel program design: a foundation*, K. M. Chandy and J. Misra, Addison-Wesley Longman Publishing Co., Inc., 1988.

"*Professors in a certain university have organized themselves into committees. Each committee has an unchanging membership roster of one or more professors. From time to time a professor may decide to attend a committee meeting; it starts waiting and remains waiting until a meeting of a committee of which it is a member is started. All meetings terminate in finite time. The restrictions on convening a meeting are as follows: (1) meeting of a committee may be started only if all members of that committee are waiting, and (2) no two committees may convene simultaneously, if they have a common member. The problem is to ensure that (3) if all members of a committee are waiting, then a meeting involving some member of this committee is convened.*"

In [31], we propose two *snap-stabilizing* distributed algorithms for the committee coordination problem. Snap-stabilization is a versatile technique allowing to design algorithms that efficiently tolerate transient faults. Indeed, after a finite number of such faults (*e.g.* memory corruptions, message losses, etc), a snap-stabilizing algorithm immediately operates correctly, without any external intervention. The first algorithm maximizes the concurrency, whereas the latter maximizes the fairness.

- **Snap-stabilizing Message Forwarding.** We focus on end-to-end request and response delivery of messages that are carried over the network. This problem is also known as the *message forwarding* problem. It consists in the management of network resources in order to forward messages, *i.e.,* protocols allowing messages to move from a sender to receiver over the network. Combined with an self-stabilizing routing protocol, achieving snap-stabilization for the message forwarding problem is a very desirable property because every message sent by the sender is delivered in finite time to the receiver. In other words, no message that was actually sent after the system started is lost. In [46], we present a snap-stabilizing algorithm for the message forwarding that works on a tree topology. It uses a constant number of buffers per link, mainly $2\delta + 1$ buffers by node, where δ is the degree of the node. Therefore, it is particularly well suited for large-scale and dynamic systems, *e.g*, overlays used in peer-to-peer systems.
- **Self-Organizing Swarms of Robots.**

  Consider a distributed system where the computing units are *weak mobile robots*, *i.e.,* devices equipped with sensors and are designed to move. By weak, we mean that the robots are *anonymous*, *autonomous*, *disoriented*, and *oblivious*, *i.e.,* devoid of (1) any local parameter (such that an identity) allowing to differentiate any of them, (2) any central coordination mechanism or scheduler, (3) any common coordinate mechanism or common sense of direction, and (4) any way to remember any previous observation nor computation performed in any previous step. Furthermore, all the robots follow the same program (*uniform* or *homogeneous*), and there is no kind of explicit communication medium. The robots implicitly "communicate" by observing the position of the others robots. Two different environments are considered: (i) the *continuous* two-dimensional Euclidian space wherein robot can observe, compute and move with infinite decimal precision, and (ii) the *discrete* model in which the space is partitioned into a finite number of locations, conveniently represented by a graph where nodes represent locations that can be sensed, and where edges represent the possibility for a robot to move from one location to the other.

  During 2011, we mainly investigated the following problems: the gathering onto the plane, and the exploration of a finite discrete environment.

  1. **Gathering**. This problem can be stated as follows: Robots, initially located at various positions, gather at the same position in finite time and remain at this position thereafter. In [20], we investigate the *self-stabilizing gathering* problem in the plane, that is gathering the robots deterministically with no kind of restriction on the initial configuration. In particular, robots are allowed to share same positions in the initial configuration. *Strong multiplicity detection* is the ability for the robots to count the exact number of robots located at a given position. We show that assuming strong multiplicity detection, it is possible to solve the self-stabilizing gathering problem with n weak robots in the semi-synchronous model if, and only if, n is odd. By contrast, we show that with an odd number of robots, the problem becomes solvable. Our proof is constructive, as we present and prove a deterministic self-stabilizing algorithm for the gathering problem.

In [19], we address the gathering in the discrete environment. We prove some asymptotical time and space complexity lower bounds to solve the problem. We propose an algorithm that is asymptotically optimal in both space and round complexities. Finally, we show that most of the assumptions we made are necessary to deterministically solve the rendezvous considering our initial scenario.

2. **Graph Exploration**.

   The exploration problem is to visit a discrete and finite environment by a swarm of robots. We consider two types of explorations: The *finite exploration* and the *perpetual exploration*. The former requires that k robots, initially placed at different nodes, collectively explore a graph before stopping moving forever. By "collectively" explore we mean that every node is eventually visited by at least one robot. In [73], we propose optimal (*w.r.t*, the number of robots) solutions for the deterministic exploration of a grid shaped network by a team of k asynchronous oblivious robots in the asynchronous and non-atomic asynchronous model. In more details, we first show that no exploration protocol exists with less than three robots for any grid with more than three nodes, less than four robots for the $(2, 2)$-Grid, and less than five robots for the $(3, 3)$-Grid. Next, we show that the problem is solvable using only 3 robots for any (i, j)-Grid, provided that $j > 3$. Our result is constructive as we present a deterministic algorithm that performs in the non-atomic asynchronous model. We also present specific deterministic protocols for the $(3, 3)$-Grid using five robots.

   The second type of exploration is the perpetual exploration. It requires every possible location to be visited by each robot infinitely often. In [32], we investigate the exclusive perpetual exploration of grid shaped networks. We focus on the minimal number of robots that are necessary and sufficient to solve the problem in general grids. In more details, we prove that three deterministic robots are necessary and sufficient, provided that the size of the grid is nxm with $3 \leq n \leq m$ or $n = 2$ and $m \geq 4$. Perhaps surprisingly, and unlike results for the exploration with stop problem (where grids are "easier" to explore and stop than rings with respect to the number of robots), exclusive perpetual exploration requires as many robots in the ring as in the grid. Furthermore, we propose a classification of configurations such that the space of configurations to be checked is drastically reduced. This pre-processing lays the bases for the automated verification of our algorithm for general grids as it permits to avoid combinatorial explosion.

6.2.3. Combining Fault-Tolerance and Self-stabilization in Dynamic Systems

Recently, we started to investigate complex faults scenarios. Distributed fault-tolerance can mask the effect of a limited number of permanent faults, while self-stabilization provides forward recovery after an arbitrary number of transient faults hit the system. FTSS (Fault-Tolerant Self-Stabilizing) protocols combine the best of both worlds since they tolerate simultaneously transient and (permanent) crash faults. To date, deterministic FTSS solutions either consider static (i.e., fixed point) tasks, or assume synchronous scheduling of the system components. We proposed in [30] a fault-tolerant and stabilizing simulation of an atomic register. The simulation works in asynchronous message-passing systems, and allows a minority of processes to crash. The simulation stabilizes in a pragmatic manner, by reaching a long execution in which it runs correctly. A key element in the simulation is a new combinatorial construction of a bounded labeling scheme accommodating arbitrary labels, including those not generated by the scheme itself. Our simulation uses a self-stabilizing implementation of a data-link over non-FIFO channels [61]. In [23] we present the first study of deterministic FTSS solutions for dynamic tasks in asynchronous systems, considering the unison problem as a benchmark. Unison can be seen as a local clock synchronization problem as neighbors must maintain digital clocks at most one time unit away from each other, and increment their own clock value infinitely often. We present several impossibility results for this difficult problem and propose a FTSS solution (when the problem is solvable) for the state model that exhibits optimal fault containment.

6.3. Peer-to-peer systems

Participants: Pierre Sens [correspondent], Nicolas Hidalgo, Sergey Legtchenko, Olivier Marin, Sébastien Monnet, Gilles Muller, Maria Potop-Butucaru, Mathieu Valero.

6.3.1. Peer-to-peer storage

Distributed Hash Table (DHTs) provide a means to build a completely decentralized, large-scale persistent storage service from the individual storage capacities contributed by each node of the peer-to-peer overlay. However, persistence can only be achieved if nodes are highly available, that is, if they stay most of the time connected to the overlay. Churn (i.e., nodes connecting and disconnecting from the overlay) in peer-to-peer networks is mainly due to the fact that users have total control on theirs computers, and thus may not see any benefit in keeping its peer-to-peer client running all the time.

When connection/disconnection frequency is too high in the system, data-blocks may be lost. This is true for most current DHT-based system's implementations. To avoid this problem, it is necessary to build really efficient replication and maintenance mechanisms. Since 2008 we study the effect of churn on an existing DHT-based P2P system namely PAST/Pastry. We have proposed RelaxDHT [25], a churn-resilient peer-to-peer DHT. RelaxDHT proposes an enhanced replication strategy with relaxed placement constraints, avoiding useless data transfers and improving transfer parallelization. This new replication strategy is able to cut down by 2 the number of data-block losses compared to PAST DHT. We are now starting to study the use of erasure coding mechanisms along with replication within DHTs. Our goal is to propose hybrid mechanisms to find a good tradeoff among 1) churn-resilience, 2) maintenance cost, and 3) storage space.

6.3.2. Overlays

Large-scale distributed systems gather thousands of peers spread all over the world. Such systems need to offer good routing performances regardless of their size and despite high churn rates. To achieve that requirement, the system must add appropriate shortcuts to its logical graph (overlay). However, to choose efficient shortcuts, peers need to obtain information about the overlay topology. In case of heterogeneous peer distributions, retrieving such information is not straightforward. Moreover, due to churn, the topology rapidly evolves, making gathered information obsolete. State of- the-art systems either avoid the problem by enforcing peers to adopt a uniform distribution or only partially fulfill these requirements. To cope with this problem, we propose DONUT [47], a mechanism to build a local map that approximates the peer distribution, allowing the peer to accurately estimate graph distance to other peers with a local algorithm. The evaluation performed with real latency and churn traces shows that our map increases the routing process efficiency by at least 20% compared to the state-of-the-art techniques. It points out that each map is lightweight and can be efficiently propagated through the network by consuming less than 10 bps on each peer.

6.3.3. Distributed trees

Publish/Subscribe implemented on top of distributed R-trees (DR-trees) overlays offer efficient DHT-free communication primitives. We have then extend the distributed R-trees (DR-trees) in order to reduce event delivery latency in order to meet the requirements of massively distributed video games such that pertinent information is quickly distributed to all the interested parties without degrading the load of nodes neither increasing the number of noisy events. In 2011, we explore how to improve robustness of distributed trees. Since each single crash can potentially break the tree structure connectivity, DR-trees are crash-sensitivite. We this have proposed a fault tolerant approach which exploits replication of non leaf nodes in order to ensure the tree connectivity in presence of crashes. This work will be published in [56].

In [85], we consider a complete binary tree and construct a random pairing between leaf nodes and internal nodes. We prove that the graph obtained by contracting all pairs (leaf-internal nodes) achieves a constant node expansion with high probability. In the context of P2P overlays our result can be interpreted as follows: if each physical node participating to the tree overlay manages a random pair that couples one virtual internal node and one virtual leaf node then the physical-node layer exhibits a constant expansion with high probability which improves the robustness of the overlay.

6.3.4. Complex query over peer-to-peer networks

A major limitation of DHTs is that they only support exact-match queries. In order to offer range queries over a DHT it is necessary to build additional indexing structures. Prefix-based indexes, such as Prefix Hash Tree (PHT), are interesting approaches for building distributed indexes on top of DHTs. Nevertheless, the lookup operation of these indexes usually generates a high amount of unnecessary traffic overhead which degrades system performance by increasing response time. In [42], we propose a novel distributed cache system called Tabu Prefix Table Cache (TPT-C), aiming at improving the performance of the Prefix-trees. We have implemented our solution over PHT, and the results confirm that our searching approach reduces up to a 70% the search latency and traffic overhead. In [44], we propose DRing an efficient layered solution that directly supports range queries over a ring-like DHT structure. We improve load balancing by using only the nodes that store data, and by updating neighbour information through an optimistic approach.

6.3.5. Trust management in peer-to-peer networks

An ongoing research work focuses on trust assessment in dynamic systems. Even if it is near impossible to fully trust a node in a P2P system, managing a set of the most trusted nodes in the system can help to implement more trusted and reliable services. Using these nodes can reduce the probability of introducing malicious nodes in distributed computations. Our work aims at the following objectives: 1. To design a distributed membership algorithm for structured Peer to Peer networks in order to build a group of trusted nodes. 2. To design a maintenance algorithm to periodically clean the trusted group so as to avoid nodes whose reputation has decreased under the minimum value. 3. To provide a way for a given node X to find at least one trusted node. 4. To design a prototype of an information system, such as a news dissemination system, that relies on the trusted group. In 2011, we propose the CORPS system for building a community of reputable peers in Distributed Hash Tables [26].

6.4. Virtual machine (VM)

Participants: Harris Bakiras, Bertil Folliot, Gaël Thomas [correspondent], Gilles Muller [correspondent], Julia Lawall, Arie Middlekoop, Thomas Preud'homme, Suman Saha.

Our research interests are in improving the way systems software is developed. One theme of our research is the development of virtual machines with a specific focus on resource management, isolation and concurrency management. Another theme of our research is related to bug finding in systems software.

6.4.1. Virtual machines

Isolation in OSGi: The OSGi framework is a Java-based, centralized, component oriented platform. It is being widely adopted as an execution environment for the development of extensible applications. However, current Java Virtual Machines are unable to isolate components from each other's. By modifying shared variables or allocating too much memory, a malicious component can freeze the complete platform. We work on I-JVM, a Java Virtual Machines that provides a lightweight approach to isolation while preserving the compatibility with legacy OSGi applications. Our evaluation of I-JVM shows that it solves the 15 known OSGi vulnerabilities due to the Java Virtual Machine with an overhead below 20%. I-JVM has been presented in DSN 2009.

VMKit: Managed Runtime Environments (MREs), such as the JVM and the CLI, form an attractive environment for program execution, by providing portability and safety, via the use of a bytecode language and automatic memory management, as well as good performance, via just-in-time (JIT) compilation. Nevertheless, developing such a fully featured MRE, including features such as a garbage collector and JIT compiler, is a herculean task. As a result, new languages cannot easily take advantage of the benefits of MREs, and it is difficult to experiment with extensions of existing MRE based languages. VMKit is a first attempt to build a common substrate that eases the development of high-level MREs. We have successfully used VMKit to build two MREs: a Java Virtual Machine (J3) and a Common Language Runtime (N3). VMKit has performance comparable to the well established open source MREs Cacao, Apache Harmony and Mono. VMKit is freely distributed under the LLVM licence with the LLVM framework developed by the University of Illinois at Urbana-Champaign and now maintained by Apple.

A third MRE is being build in cooperation with the "Algorithms, Programmes and Resolution" team in LIP6. This integrates a functional machine (the Zinc Abstract Machine) in VMKit and show that the adataptation at the language level of our virtual machine. This project has been funded by the LIP6 in 2009-10 and 2010-11.

6.4.2. Semantic patches

Open source infrastructure software, such as the Linux operating system, Web browsers and n-tier servers, has become a well-recognized solution for implementing critical functions of modern life. Furthermore, companies and local governments are finding that the use of open source software reduces costs and allows them to pool their resources to build and maintain infrastructure software in critical niche areas. Nevertheless, the increasing reliance on open source infrastructure software introduces new demands in terms of security and safety. In principle, infrastructure software contains security features that protect against data loss, data corruption, and inadvertent transmission of data to third parties. In practice, however, these security features are compromised by a simple fact: software contains bugs.

We are developing a comprehensive solution to the problem of finding bugs in API usage in open source infrastructure software based on our experience in using the Coccinelle code matching and transformation tool, and our interactions with the Linux community.

Coccinelle has been successfully used for finding and fixing bugs in systems code. One of our main recent results is an extensive study of bugs in Linux 2.6 [51] 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.

We have used Coccinelle to generate traditional patches for improving the safety of Linux. Some Linux developers have also begun to use the tool. Over 800 patches developed using Coccinelle have been integrated into the mainline Linux kernel. As part of the ABL ANR project, we are building on the results of Coccinelle by designing semantic patches to identify API protocols and detect violations in their usage [24].

Another work done as part of the ANR ABL project, and as the topic of Suman Saha's PhD thesis, is the improvement of error handling code in Linux. We developed a program analysis for identifying the code structures used to represent error handling code and a transformation to convert existing error handling code to use gotos to shared cleanup code, which is the style preferred by the Linux community [53]. We subsequently worked on finding bugs in error handling code, following an approach that focuses on local patterns, i.e., within the current function, rather than patterns occurring across the entire code base. This approach has a low rate of false positives and can find bugs in the use of rarely called functions [39].

6.5. Hosted database replication service

Participant: Mesaac Makpangou [correspondent].

Today, the vast majority of content distributed on the web are produced by web 2.0 applications. Examples of such applications include social networks, virtual universities, multi-players games, e-commerce web sites, and search engines. These applications rely on databases to serve end-users' requests. Hence, the success of these applications/services depends mainly on the scalability and the performance of the database backend.

The objective of our research is to provide a hosted database replication service. With respect to end-users applications, this service offers an interface to create, to register, and to access databases. Internally, each hosted database is fragmented and its fragments are replicated towards a peer-to-peer network. We anticipate that such a service may improve the performance and the availability of popular web applications, thanks to partial replications of backend databases. Partial database replication on top of a peer-to-peer network raises a number of difficult issues: (i) enforcing replica consistency in presence of update transactions, without jeopardizing the scalability and the performance of the system? (ii) accommodating the dynamic and the heterogeneity of a peer-to-peer network with the database requirements?

We designed a database access protocol, capable to spread out a transaction's accesses over multiple database fragments replicas while guaranteeing that each transaction observes a consistent distributed snapshot of a partially replicated database. We have also proposed a replica control substrate that permits to enforce 1-Copy SI for database fragments replicated over a wide area network. For that, unlike most database replication, we separate the synchronisation from the certification concerns.

A small-scale group of schedulers that do not hold database replicas, cooperate with one another to certify update transactions. Only certified transactions are notified to replicas. Furthermore, each replica will be notified only the transactions that impact the that it stores. Thanks to this separation, we avoid waste of computation resource at replicas that will be used to decide whether to abord or commit an update transaction; Our design choices also permit to reduce bandwidth consumption.

In 2010, we focus on the development of a prototype implementation of the complete system. The current prototype includes: a tool that helps fragment a database into fragments; the support to deploy dynamically, for each fragment, the suitable number of replicas towards the network of hosting peers; the implementation of our proposal (i.e. our database access protocol and our replica control substrate); and the JDBC API extension for accessing replicated databases.

6.6. Commitment protocols for WAN replication

Participants: Marc Shapiro [correspondent], Pierre Sutra, Masoud Saeida Ardekani.

In a large-scale distributed system, replication is an essential technique for improving availability and read performance. However, writes raise the issue of consistency, especially in the presence of concurrent updates, network failures, and hardware or software crashes. So-called *consensus* constitutes a major primitive to solving these issues. The performance of large-scale systems depends crucially on the latency of consensus, especially in wide-area networks; to decrease it, we focus on *generalised consensus* algorithms, i.e., ones that leverage the commutativity of operations and/or the spontaneous ordering of messages by the network. One such algorithms is Generalized Paxos, which does not order concurrent commutative operations. However, when a collision occurs (i.e., two replicas receive non-commuting operations in a different order) Generalized Paxos requires a very high latency to recover, completely negating the gain. We designed FGGC, a new generalised consensus algorithm that minimises the cost of recovering from a collision, without decreasing resilience to faults. FGGC achieves the optimal latency (two communication steps when processes receive non-commutative operations in the same order, and three otherwise) when there are no faults. FGGC remains optimally fault-tolerant, as it tolerates $f < n/2$ crash faults and requires only $f + 1$ processes to make progress. Our experimental evaluation of FGGC shows that it is more efficient than the competing protocols. Another topic of relevance in WANs is partial replication, i.e., where any given server holds only a fraction of all shared objects. This decreases the workload per server and improves access times. However, this makes transactional concurrency control more difficult; indeed most existing algorithms assume full replication. We designed and implemented two *genuine* consensus protocols for partial replication, i.e., ones in which only relevant replicas need participate in the commit of a transaction. They were evaluated experimentally above the BerkeleyDB database engine. This work is the topic of Pierre Sutra's PhD thesis.

6.7. Optimistic approaches in collaborative editing

Participants: Marc Shapiro [correspondent], Marek Zawirski, Pierpaolo Cincilla.

In recent years, the Web has seen an explosive growth of massive collaboration tools, such as wiki and weblog systems. By the billions, users may share knowledge and collectively advance innovation, in various fields of science and art. Existing tools, such as the MediaWiki system for wikis, are popular in part because they do not require any specific skills. However, they are based on a centralised architecture and hence do not scale well. Moreover, they provide limited functionality for collaborative authoring of shared documents.

A natural research direction is to use P2P techniques to distribute collaborative documents. This raises the issue of supporting collaborative edits, and of maintaining consistency, over a massive population of users, shared documents, and sites.

In order to avoid complex and unnatural concurrency control and synchronisation, and to enable different styles of collaboration (from online "what you see is what I see" to fully asynchronous disconnected work) we invented the concept of a Commutative Replicated Data Type (CRDT). A CRDT is one where all concurrent operations commute. The replicas of a CRDT converge automatically, without complex concurrency control.

In the context of collaborative editing, we propose, a novel CRDT design called Treedoc. An essential property is that the identifiers of Treedoc atoms are selected from a dense space. We study practical alternatives for implementing the identifier space based on an extended binary tree. We also focus storage alternatives for data and meta-data, and mechanisms for compacting the tree. In the best case, Treedoc incurs no overhead with respect to a linear text buffer. We validate the results with traces from existing edit histories.

Treedoc will be used in ANR projects PROSE (Section 7.1.5) and STREAMS, and will be further studied and developed in ANR project ConcoRDanT (Section 7.1.3) and under a Google European Doctoral Fellowship.

6.8. CRDTs, a principled approach to eventual consistency

Participants: Marc Shapiro [correspondent], Marek Zawirski.

Most well-studied approaches to replica consistency maintain a global total order of operations. This serialisation constitutes a performance and scalability bottleneck, while the CAP theorem imposes a trade-off between consistency and partition-tolerance. An alternative approach, *eventual consistency* or *optimistic replication*, is attractive. A replica may execute an operation without synchronising *a priori* with other replicas. The operation is sent asynchronously to other replicas; every replica eventually applies all updates, but possibly in different orders. This approach ensures that data remains available despite network partitions, and is perceived to scale well and to provide acceptable quality of service. The consensus bottleneck remains but is off the critical path. However, reconciliation is generally complex. There is little theoretical guidance on how to design a correct optimistic system, and ad-hoc approaches have proven brittle and error-prone. We propose a simple, theoretically sound approach to eventual consistency, the concept of a *convergent* or *commutative replicated data type* (CRDT), for which some simple mathematical properties ensure eventual consistency. Provably, any CRDT converges to a common state that is equivalent to some sequential execution. A CRDT requires no synchronisation, thus every update can execute immediately, unaffected by network latency, faults, or disconnection. It is extremely scalable and is fault-tolerant, and does not require much mechanism. Previously, only a handful of CRDTs were known. Our current research aims to push the CRDT envelope, to study the principles of CRDTs, and to design a library of useful CRDTs. So far we have designed variations on registers, counters, sets, maps (key-value stores), graphs, and sequences. Potential application areas include computation in delay-tolerant networks, latency tolerance in wide-area networks, disconnected operation, churn-tolerant peer-to-peer computing, and partition-tolerant cloud computing. CRDTs are the main topic of ANR project ConcoRDanT (Section 7.1.3). This research is also funded in part by a Google European Doctoral Fellowship.

7. Partnerships and Cooperations

7.1. National initiatives

7.1.1. 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.

7.1.2. 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.

7.1.3. ConcoRDanT - (2010–2013)

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

Funding: PROSE project 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.

7.1.4. 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.

7.1.5. PROSE - (2009–2011)

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.

7.1.6. ABL - (2009–2012)

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

We have used Coccinelle to reproduce over 60 collateral evolutions in recent versions of Linux, affecting almost 6000 files. Recently, we have begun using Coccinelle to generate traditional patches for improving the safety of Linux. Over 800 of these patches developed using Coccinelle have been integrated into the mainline Linux kernel. Julia Lawall was among the top 10 in terms of the number of contributed patches in Linux 2.6.36. Finally, about 20 semantic patches are integrated into the Linux sources so that developpers can improve the quality of their programs by running Coccinelle as part of the development process.

In the ABL project, we are building on the results of Coccinelle by 1) designing libraries of semantic patches to identify API protocols and detect violations in their usage, 2) extending Coccinelle to address the needs of bug finding and reporting, and 3) designing complementary tools to help the programmer to track and fix bugs.

7.1.7. SHAMAN - (2009–2011)

Members: LIP6 (NPA), Inria Saclay (Grand-Large), Inria Bretagne (ASAP), LIP6 (Regal)

Funding: SHAMAN project is funded by ANR TELECOM

Objectives: Large-scale networks (e.g. sensor networks, peer-to-peer networks) typically include several thousands (or even hundred thousand) basic elements (computers, processors) endowed with communication capabilities (low power radio, dedicated fast network, Internet). Because of the large number of involved components, these systems are particularly vulnerable to occurrences of failures or attacks (permanent, transient, intermittent). Our focus in this project is to enable the sustainability of autonomous network functionalities in spite of component failures (lack of power, physical damage, software or environmental interference, etc.) or system evolution (changes in topology, alteration of needs or capacities). We emphasize the self-organization, fault-tolerance, and resource

saving properties of the potential solutions. In this project, we will consider two different kinds of large-scale systems: on one hand sensor networks, and on the other hand peer to peer networks.

7.1.8. R-DISCOVER - (2009–2011)

Members: MIS, LASMEA, GREYC, LIP6 (Regal), Thales

Funding: R-DISCOVER project is funded by ANR CONTINT

Objectives: This project considers a set of sensors and mobile robots arbitrarily deployed in a geographical area. Sensors are static. The robots can move and observe the positions of other robots and sensors in the plane and based on these observations they perform some local computations. This project addresses the problem of topological and cooperative navigation of robots in such complex systems.

7.1.9. PACTOL - (2009–2011)

Members: LIP6 (NPA, Regal), CNAM

Funding: Ile de France Region

Objectives: The scope of PACTOL is to propose verification tools for self-stabilizing distributed algorithms.

7.2. European Initiatives

7.2.1. 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.

7.2.2. FTH-GRID - (2009–2011)

Members: Université de Lisbonne (LASIGE), LIP6 (Regal)

Funding: Egide

Objectives: FTH-Grid, Fault-Tolerant Hierarchical Grid Scheduling, is a cooperation project between the Laboratoire d'Informatique de Paris 6 (LIP6/CNRS, France) and the Large-Scale Informatics Systems Laboratory (LASIGE/FCUL, Portugal).

Its goal is to foster scientific research collaboration between the two research teams. The project aims at rendering Map Reduce on top of Grid tolerant to byzantine failure. Map Reduce is a programming model for large-scale data-parallel applications whose implementation is based on master-slave scheduling of bag-of-tasks. MapReduce breaks a computation into small tasks that run in parallel on different machines, scaling easily to several cluster. The core research activities of the project consist mainly in extending the execution and programming model to make Byzantine fault-tolerant MapReduce applications. The project was extended for another year, after a results assessment by Egide.

7.3. International Initiatives

7.3.1. DEMEDYS - INRIA-CNPq - (2010–2011)

Members: INRIA Bretagne (ASAP), INRIA Paris Rocquencourt (REGAL), UFBA (Bahia, Brazil), IME (Sao Paulo, Brazil)

Funding: INRIA / CNPq

Objectives: DEMEDYS Project (Dependable Mechanisms for Dynamic Systems) will study fundamental aspects of dynamic distributed systems.

7.3.2. 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.

7.3.3. 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.

7.3.4. Bi-lateral collaborations

JAIST (Japon). With the group of Prof. Xavier Defago we investigate various aspects of self-organization and fault tolerance in the context of robots networks.

UNLV (USA) With the group of Prof. Ajoy Datta we collaborate in designing self* solutions for the computations of connected covers of query regions in sensor networks.

Technion (Israel). We collaborate with Prof. Roy Friedman on divers aspects of dynamic systems ranging from the computation of connected covers to the design of agreement problems adequate for P2P networks.

Ben Gurion (Israel). We collaborate recently with prof. Shlomi Dolev on the implementation of self-stabilizing atomic memory.

Kent University (SUA) With prof. Mikhail Nesterenko we started recently a collaboration on FTSS solutions for dynamic tasks.

Nagoya Institute of Technology (Japon). With prof. Taisuke Izumi we started this year a collaboration on the probabilistic aspects of robot networks.

COFECUB (Brazil). With the group of Prof. F. Greve. (Univ. Federal of Bahia), we investigate various aspects of failure detection for dynamic environement such as MANET of P2P systems.

CONYCIT (Chili). Since 2007, we start on new collaboration with the group of X. Bonnaire Fabre (Universidad Técnica Federico Santa María - Valparaiso). The main goal is to implement trusted services in P2P environment. Even if it is near impossible to fully trust a node in a P2P system, managing a set of the most trusted nodes in the system can help to implement more trusted and

reliable services. Using these nodes, can reduce the probability to have some malicious nodes that will not correctly provide the given service. The project will have the following objectives: 1. To design a distributed membership algorithm for structured Peer to Peer networks in order to build a group of trusted nodes. 2. To design a maintenance algorithm to periodically clean the trusted group so as to avoid nodes whose reputation has decreased under the minimum value. 3. To provide a way for a given node X to find at least one trusted node. 4. To design a prototype of an information system, such as a news dissemination system, that relies on the trusted group.

Collaboration with CITI-UNL, Portugal. Our collaboration with CITI, the Research Center for Informatics and Information Technologies of UNL, the New University of Lisbon (Portugal), is materialised by several joint articles. Furthermore, Marc Shapiro is an advisor to the project "RepComp - Replicated Components for Improved Performance or Reliability in Multicore Systems," funded by Fundacio para a Ciancia e a Tecnologia (FCT, Portuguese equivalent of ANR). Finally, Marc Shapiro is a Member of the CITI Advisory Board.

8. Dissemination

8.1. Animation of the scientific community

Luciana Arantes is:

- Member of the program committee of the 6ème Conférence fran caise sur les systèmes d'exploitation, CFSE-6, Friburg, Switzerland, february 2008.
- Member of PC of Workshop de Sistemas Operacionais, SBC, Brésil, 2007-2010.
- Member of PC of WTF - Workshop of Fault Tolerance, Brésil, 2009-201
- Member of PC of International Conference on Grid and Pervasive Computing, 2009-2011.
- Member of PC of LADC - Fifth Latin-American Symposium on Dependable Computing 2011.
- Reviewer for JPDC and TPDS journals.

Bertil Folliot is:

- Member of the scientific committee of LIP6.
- 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".
- Member of the program committee of the 10th International Symposium on Parallel and Distributed Computing (ISPDC 2011), Cluj-Napoca, Roumanie, juillet 2011
- Member of the program committee of the 2011 International Conference on High Performance Computing & Simulation (HPCS 2011), Istanbul, Turquie, juillet 2011.
- Member of the program committee of the 9th International Conference on the Principles and Practice of Programming in Java (PPPJ 2011), Kongens Lyngby, Denmark, aout 2011

Maria Potop-Butucaru is:

- Member of the program commitee of SSS 2011(13th International Symposium on Stabilization, Safety, and Security of Distributed Systems)
- Member of the program commitee of DISC 2011 (25th International Symposium on Distributed Computing)
- External member of the "Commission de specialistes" INSA Toulouse
- Reviewer for the PhD Thesis of Fouzi Mekhaldi, Universite Paris 11, Advisors Veronique Veque and Colette Johnen

Olivier Marin is:

- Reviewer for Distributed Computing, and Techniques et Sciences Informatiques.
- Elected vice-president of the “Conseil d’experts” of the Paris 6 University, section 27.
- Elected member of the “Commission des primes de recherche” of the Paris 6 University.
- Elected member of the “Commission des avancements de carrière - rang B” of the Paris 6 University.
- Member of the scientific committee of LIP6.
- Member of “Comité de selection" of Grenoble INP - ESISAR (Maitre de Conférences)

Sébastien Monnet is:

- Elected member of the administrative committee of LIP6.

Gilles Muller is:

- Member of the PC and co-organizer of the PLOS workshop, October 2011, Portugal, http://plosworkshop.org/2011/
- Member of PC of the SRDS 2011 conference, October 2011 http://lsd.ls.fi.upm.es:11888/srds/
- Member of PC of the SYSTOR 2011 conference, May 2011 https://www.research.ibm.com/haifa/conferences/systor2011/
- Member of the jury of the best European thesis on systems (EuroSys) 2011
- Member of PC of the 8ème Conférence francaise sur les systèmes d’exploitation, CFSE-8, May 2011 http://renpar.irisa.fr/CFP-CFSE.html
- Member of PC of the 5th EuroSys Doctoral workshop, Saltzburg, April 2011
- Member of "Comité de selection" of INSA de Lyon (Professeur) and Univerity of Nice (Maitre de Conférence)
- 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 University of Leuven
- Reviewer for the Swiss National Science Foundation

Julia Lawall is:

- Chair of the steering committee of Generative Programming and Component Engineering (2011, 2012)
- Secretary of IFIP TC2 (from 2011)
- 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 OBT: Underrepresented Problems for PL Researchers, with POPL 2012
- Member of the Needham Award Review Committee (PhD dissertation award of EuroSys).
- Invited reviewer for PLDI 2012
- Member of IFIP WG 2.11 (Program Generation)

Franck Petit is:

- PC Member of SSS 2012, 14^{th} International Symposium on Stabilization, Safety, and Security of Distributed Systems, ed. LNCS, Toronto, 2012.
- Co-program chair and co-organizing chair of SSS 2011, 13^{th} International Symposium on Stabilization, Safety, and Security of Distributed Systems, ed. LNCS, Grenoble, 2011. 14^{th} International Workshop on Logical Aspects of Fault-Tolerance, co-collated with 26^{th} Annual IEEE Symposium on Logic in Computer Science (LICS 2011) Toronto, Canada.
- PC Member of LAFT 2011, 2^{nd} International Workshop on Logical Aspects of Fault-Tolerance, co-collated with 26^{th} Annual IEEE Symposium on Logic in Computer Science (LICS 2011) Toronto, Canada.
- PC Member of Renpar 2011, 20^{th} Rencontres francophones du Parallélisme, France, 2011.
- Invited Editor for ACM TAAS, Transactions on Autonomous and Adaptive Systems, Special Issue on Stabilization, Safety, and Security of Distributed Systems,
- Invited Editor for TCS, Theoretical Computer Science Special Issue on Stabilization, Safety, and Security of Distributed Systems,
- Member of scientific committee for the AREA evaluation of “INRIA Bretagne”.
- Member of “Vivier d’experts” of UPMC Paris 6.

Pierre Sens is:

- Local chair of Topic 8 “Distributed systems and algorithms” of EuroPar 2011
- 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 "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 scientific committee of ANR project Blanc International.
- Member of "Comité de selection" of Universities of Amiens, Grenoble, and Paris X.

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).
- Member of PC of EuroSys 2010.
- PC Chair of CFSE 2011, and member of PC of CFSE 2009 (Conférence francaise sur les systèmes d’exploitation).
- 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.

8.2. PhD reviews

Bertil Folliot was part of the PhD committee of:

- Jérôme Gallard. Services et mécanismes système fondés sur la virtualisation pour une gestion flexible des infrastructures distribuées étendues. Thèse de Doctorat de l'Université de Rennes I (directeur : Christine Morin), Rennes, April 2011. Reviewer.
- Maarten Bynens. A System of Patterns for the Design of Reusable Aspect Libraries. Degree of Doctor in Engineering, Katholieke Universiteit Leuven (directeurs : Wouter Joosen, Eddy Truyen), Leuven, Belgium, July 2011. Reviewer.
- Erika Rosas. Building Trustworthy Services in P2P networks. Thèse de Doctorat de l?Université Paris VI, novembre 2011. President.

Gilles Muller was part of the committee of:

- Juan Navas. PhD. Univ. Brest, May 2011, Reviewer.
- Patrick Marlier. PhD. Univ. Neuchâtel, August 2011, Member.
- Kiev Gama. PhD. Univ. Grenoble, October 2011, Reviewer.
- Laurent Réveillère. HDR. Univ. Bordeaux, November 2011, Member.
- Jean-Marc Menaud. HDR. école des Mines de Nantes, June 2011, Member.

Julia Lawall was part of the committee of:

- Laurent Réveillère. HDR. Univ. Bordeaux, November 2011, Member.

Franck Petit was member of thesis committees of:

- Lélia Blin, HDR LiP6, Commitee Chair
- Nicolas Hidalgo, PhD LiP6 (Advisors: P. Sens, L. Arantes, and X. Bonnaire), Commitee Chair
- Thomas Aynaud, PhD LiP6, (Advisors: M. Latapy, J-L. Guillaume), Commitee Chair

Pierre Sens was the reviewer of:

- A. Guermouche. PhD. LRI, (Advisor : F. Cappello)
- P. Riteau. PhD. IRISA, (Advisor : C. Morin)
- A. Harbaoui. PhD. LIG, (Advisor: B. Plateau, J-M. Vincent)
- W. Malvault. PhD LIG, (Advisors: JB. Stefani, V. Quema)

8.3. Teaching

- Luciana Arantes
  - Principles of operating systems in Licence d'Informatique, Université Paris 6
  - Operating systems kernel in Master Informatique, Université Paris 6
  - Distributed algorithms in Master Informatique, Université Paris 6
  - Responsible of Advanced distributed algorithms in Master Informatique, Université Paris 6
  - Unix system programming, Licence and Master d'Informatique, Université Paris 6
- Bertil Folliot
  - Head of the second year of professional Licence d'Informatique, Université Paris 6
  - Principles of operating systems in Licence d'Informatique, Université Paris 6
  - Responsible Advanced POSIX & C, Université Paris 6
  - Distributed algorithms and systems in Master Informatique, Université Paris 6
  - Distributed systems and client/serveur in Master Informatique, Université Paris 6
  - Projects in distributed programming in Master Informatique, Université Paris 6
- Maria Potop-Butucaru
  - Principles of Operating Systems, Licence d'Informatique, Université Paris 6
  - Distributed algorithms, Master d'Informatique, Université Paris 6
  - Resitance of Distributed Attacks, Master Paris 6, Network track
  - Embedded communicant systems, Master Paris 6, Network track
- Mesaac Makpangou
  - Client/server architecture, Licence professionelle d'Informatique, Université Paris 6: TD, 16 heures
- Oliver Marin
  - Advanced operating systems programming, Master d'Informatique, Université Paris 6
  - Advanced distributed algorithms in Master Informatique, Université Paris 6
  - Responsible of operating systems programming, Licence d'Informatique, Université Paris 6
  - Parallel and distributed systems, Master d'Informatique, Université Paris 6
  - Responsible of the "Parcours professionnalisant L3 - Développeur d'Applications - Nouvelles Technologies (DANT)"
  - Responsible of Deployment of Cooperative Objects, Licence d'Informatique - parcours DANT, Université Paris 6
- Sébastien Monnet
  - Responsible of "Middleware for advanced computing systems in Master d'Informatique (2), Université Paris 6"
  - Responsible of System and Internet programmation in Licence d'Informatique (2), Université Paris 6
  - Operating systems kernel in Master Informatique (1), Université Paris 6
  - Principles of operating systems in Licence d'Informatique (3), Universié Paris 6

  - Computer science initiation in Licence d'Informatique (1), Université Paris 6

- Pierre Sens
  - Responsible of "Principles of operating systems" in Licence d'Informatique, Université Paris 6
  - Responsible of "Operating systems kernel in Master Informatique", Université Paris 6
  - Distributed systems and algorithms in Master Informatique, Université Paris 6

- Marc Shapiro
  - Teaches NMV (*Noyaux Multi-cœurs et Virtualisation*, i.e., multicore kernels and virtualisation) at Université Paris 6, Master 2.

- Gaël Thomas
  - Responsible for the Master 1 module "Systèmes Répartis Clients/Serveurs" in Master Informatique at the Univeristy Université Paris 6
  - Responsible for the Master 2 module "Middleware Orientés Composants" in Master Informatique at the Univeristy Université Paris 6
  - Responsible for Master 2 module NMV (*Noyaux Multi-cœurs et Virtualisation*, i.e., multicore kernels and virtualisation)
  - Responsible for the Master 2 module "Répartition et Client/Serveur" in Master Informatique at the Univeristy Université Paris 6
  - "Noyau des Systèmes d'exploitation" in Master Informatique at the Université Paris 6
  - "Systèmes" at PolyTech' Paris

9. Bibliography

Major publications by the team in recent years

[1] E. ANCEAUME, R. FRIEDMAN, M. GRADINARIU. *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, 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] N. HIDALGO. *Tolérer les fautes transitoires, permanentes et intermittentes*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, decembre 2011.

[14] N. HIDALGO. *Towards an Efficient Support for Complex Queries on Structured Peer-to-Peer Networks*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, november 2011.

[15] E. ROSAS. *Building Trustworthy Services in P2P networks*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, november 2011.

[16] M. VALERO. *Amélioration des performances et de la fiabilité des architectures pair à pair arborescentes*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, decembre 2011.

Articles in International Peer-Reviewed Journal

[17] L. BLIN, M. POTOP-BUTUCARU, S. ROVEDAKIS. *Self-stabilizing minimum degree spanning tree within one from the optimal degree*, in "Journal of Parallel and Distributed Computing (JPDC)", 2011, vol. 3 71, p. 438–449.

[18] L. BURGY, L. RÉVEILLÈRE, J. LAWALL, G. MULLER. *Zebu: A Language-Based Approach for Network Protocol Message Processing*, in "IEEE Transactions on Software Engineering", 2011, vol. 37, n^{o} 4, p. 575–591.

[19] F. CARRIER, S. DEVISMES, F. PETIT, Y. RIVIERRE. *Asymptotically Optimal Deterministic Rendezvous*, in "Int. J. Found. Comput. Sci.", 2011, vol. 22, n^{o} 5, p. 1143-1159.

[20] Y. DIEUDONNÉ, F. PETIT. *Self-stabilizing Gathering with Strong Multiplicity Detection*, in "Theoretical Computer Science", 2012, to appear.

[21] S. DOLEV, S. DUBOIS, M. POTOP-BUTUCARU, S. TIXEUIL. *Stabilizing data-link over non-FIFO channels with optimal fault-resilience*, in "Information Processing Letters", September 2011, http://hal.inria.fr/inria-00627760/en.

[22] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Bounding the Impact of Unbounded Attacks in Stabilization*, in "IEEE Transactions on Parallel and Distributed Systems (TPDS)", 2011.

[23] S. DUBOIS, M. POTOP-BUTUCARU, S. TIXEUIL. *Dynamic FTSS in Asynchronous Systems: the Case of Unison*, in "Theoretical Computer Science (TCS)", 2011, vol. 29 412, p. 3418–3439.

[24] 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", 2012, to appear.

[25] S. LEGTCHENKO, S. MONNET, P. SENS, G. MULLER. *RelaxDHT: a churn-resilient replication strategy for peer-to-peer distributed hash-tables1*, in "ACM Transactions on Autonomous and Adaptive Systems", 2011.

[26] E. ROSAS, O. MARIN, X. BONNAIRE. *CORPS: Building a Community Of Reputable PeerS in Distributed Hash Tables*, in "The Computer Journal", September 2011, http://hal.inria.fr/inria-00627499/en.

[27] M. SHAPIRO, N. PREGUIÇA, C. BAQUERO, M. ZAWIRSKI. *Convergent and Commutative Replicated Data Types*, in "Bulletin of the European Association for Theoretical Computer Science (EATCS)", June 2011, n^{o} 104, p. 67–88, http://lip6.fr/Marc.Shapiro/papers/CRDTs-beatcs-2011-06.pdf.

Articles in National Peer-Reviewed Journal

[28] S. DEVISMES, F. PETIT, V. VILLAIN. *Autour de l'Auto-Stabilisation: Partie I : Techniques généralisant l'approche*, in "Technique et Science Informatiques", 2011, vol. 30, p. 1-22.

[29] S. DEVISMES, F. PETIT, V. VILLAIN. *Autour de l'Auto-Stabilisation: Partie II : Techniques spécialisant l'approche*, in "Technique et Science Informatiques", 2011, vol. 30, p. 23-50.

International Conferences with Proceedings

[30] N. ALON, H. ATTIYA, S. DOLEV, S. DUBOIS, M. POTOP-BUTUCARU, S. TIXEUIL. *Pragmatic Self-Stabilization of Atomic Memory in Message-Passing Systems*, in "Proceedings of SSS 2011", X. DÉFAGO, F. PETIT, V. VILLAIN (editors), Springer Berlin / Heidelberg, 2011.

[31] B. BONAKDARPOUR, S. DEVISMES, F. PETIT. *Snap-Stabilizing Committee Coordination*, in "25th IEEE International Symposium on Parallel and Distributed Processing, IPDPS 2011", 2011, p. 231-242, http://doi.ieeecomputersociety.org/10.1109/IPDPS.2011.31.

[32] F. BONNET, A. MILANI, M. POTOP-BUTUCARU, S. TIXEUIL. *Asynchronous exclusive perpetual grid exploration without sense of direction*, in "Proceedings of OPODIS 2011", A. F. ANTA (editor), Springer Berlin / Heidelberg, 2011.

[33] D. CASSOU, E. BALLAND, C. CONSEL, J. LAWALL. *Leveraging software architectures to guide and verify the development of sense/compute/control applications*, in "Proceedings of the 33rd International Conference on Software Engineering, ICSE 2011", Waikiki, Honolulu, HI, USA, May 2011, p. 431–440.

[34] S. DOLEV, S. DUBOIS, M. POTOP-BUTUCARU, S. TIXEUIL. *Communication Optimalement Stabilisante sur Canaux non Fiables et non FIFO*, in "13es Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel)", Cap Estérel, France, B. DUCOURTHIAL, P. FELBER (editors), 2011, http://hal.inria.fr/inria-00587089/en.

[35] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Auto-Stabilisation et Confinement de Fautes Malicieuses : Optimalité du Protocole min+1*, in "CoRR Proceedings of Algotel 2011", 2011, vol. abs/1104.4022, http://arxiv.org/abs/1104.4022.

[36] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Auto-Stabilisation et Confinement de Fautes Malicieuses : Optimalité du Protocole min+1*, in "13es Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel)", Cap Estérel, France, B. DUCOURTHIAL, P. FELBER (editors), 2011, http://hal.inria.fr/inria-00587517/en.

[37] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Maximum Metric Spanning Tree made Byzantine Tolerant*, in "Proceedings of DISC 2011", D. PELEG (editor), Springer Berlin / Heidelberg, 2011, http://arxiv.org/abs/1104.5368.

[38] P. FLORIANO, A. GOLDMAN, L. ARANTES. *Formalization of the Necessary and Sufficient Connectivity Conditions to the Distributed Mutual Exclusion Problem in Dynamic Networks*, in "Proceedings of The Tenth IEEE International Symposium on Networking Computing and Applications, NCA 2011", 2011, p. 203-210.

[39] L. GIDRA, G. THOMAS, J. SOPENA, M. SHAPIRO. *Assessing the Scalability of Garbage Collectors on Many Cores*, in "6th Workshop on Programming Languages and Operating Systems (PLOS'11).", Cascais, Portugal, October 2011.

[40] F. GREVE, L. ARANTES, P. SENS. *What model and what conditions to implement unreliable failure detectors in dynamic networks?*, in "Proceedings of the 3rd International Workshop on Theoretical Aspects of Dynamic Distributed Systems", TADDS '11, 2011, p. 13–17.

[41] F. GREVE, P. SENS, L. ARANTES, V. SIMON. *A Failure Detector for Wireless Networks with Unknown Membership*, in "Euro-Par", 2011, p. 27–38.

[42] N. HIDALGO, L. ARANTES, P. SENS, X. BONNAIRE. *A Tabu Based Cache to Improve Latency and Load Balancing on Prefix Trees*, in "IEEE International Conference on Parallel and Distributed Systems (ICPADS)", December 2011.

[43] N. HIDALGO, L. ARANTES, P. SENS, X. BONNAIRE. *A Tabu Based Cache to Improve Latency and Load Balancing on Prefix Trees*, in "ICPADS 2011 - IEEE International Conference on Parallel and Distributed Systems", Tainan, Taiwan, Province Of China, December 2011, http://hal.inria.fr/inria-00627479/en.

[44] N. HIDALGO, E. ROSAS, L. ARANTES, O. MARIN, P. SENS, X. BONNAIRE. *DRing: A Layered Scheme for Range Queries over DHTs*, in "IEEE International Conference on Computer and Information Technology (CIT)", Press, IEEE Society, August 2011, p. 29-34, http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=6036587.

[45] N. HIDALGO, E. ROSAS, L. ARANTES, O. MARIN, P. SENS, X. BONNAIRE. *DRing: A Layered Scheme for Range Queries over DHTs*, in "IEEE International Conference on Computer and Information Technology", Paphos, Cyprus, August 2011, http://hal.inria.fr/inria-00627476/en.

[46] 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, to appear.

[47] S. LEGTCHENKO, S. MONNET, P. SENS. *DONUT: Building Shortcuts in Large-Scale Decentralized Systems with Heterogeneous Peer Distributions*, in "30th Symposium on Reliable Distributed Systems (SRDS 2011)", Madrid, Spain, IEEE Computer Society, October 2011.

[48] W. MALDONADO, P. MARLIER, P. FELBER, J. LAWALL, G. MULLER, E. RIVIERE. *Deadline-Aware Scheduling for Software Transactional Memory*, in "Proceedings of the International Conference on Dependable Systems and Networks (DSN 2011)", Hong Kong, June 2011, p. 257–268.

[49] W. MALDONADO, P. MARLIER, P. FELBER, J. LAWALL, G. MULLER, E. RIVIERE. *Kernel-Assisted Scheduling and Deadline Support for Software Transactional Memory*, in "Proceedings of the Conférence Française en Systèmes d'Exploitation (CFSE)", May 2011.

[50] J. MICHAUX, X. BLANC, P. SUTRA, M. SHAPIRO. *A Semantically Rich Approach for Collaborative Model Edition*, in "ACM Symp. on Applied Computing (SAC)", TaiChung, Taiwan, ACM SIGAPP, March 2011, vol. 26, p. 1470–1475, http://lip6.fr/Marc.Shapiro/papers/SAC2011-Cpraxis_wholePaper.pdf.

[51] 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.

[52] M. POTOP-BUTUCARU, M. RAYNAL, S. TIXEUIL. *Distributed Computing with Mobile Robots: an Introductory Survey*, in "Proceedings of the International Conference on Network-Based Information Systems (NBIS)", 2011.

[53] S. SAHA, J. LAWALL, G. MULLER. *An approach to improving the structure of error-handling code in the linux kernel*, in "Proceedings of the 2011 SIGPLAN/SIGBED conference on Languages, compilers and tools for embedded systems", ACM, 2011, p. 41–50.

[54] S. SAHA, J. LAWALL, G. MULLER. *Finding Resource-Release Omission Faults in Linux*, in "6th Workshop on Programming Languages and Operating Systems", Cascais, Portugal, October 2011.

[55] P. SUTRA, M. SHAPIRO. *Fast Genuine Generalized Consensus*, in "30th IEEE Symposium on Reliable Distributed Systems (SRDS 2011)", 2011, p. 255-264.

[56] M. VALERO, L. ARANTES, M. POTOP-BUTUCARU, P. SENS. *Enhancing Fault Tolerance of Distributed R-Tree*, in "5th Latin-American Symposium on Dependable Computing", 2011, p. 25–34.

[57] M. ZAWIRSKI, M. SHAPIRO, N. PREGUIÇA. *Asynchronous rebalancing of a replicated tree*, in "cfse", Saint-Malo, France, May 2011, 12.

[58] M. S. DE LIMA, F. GREVE, L. ARANTES, P. SENS. *The time-free approach to Byzantine failure detection in dynamic networks*, in "Dependable Systems and Networks Workshops", 2011, p. 3–8.

National Conferences with Proceeding

[59] L. ARANTES, J. LEJEUNE, M. PIFFARETTI, O. MARIN, P. SENS, J. SOPENA, A. N. BESSANI, V. V. COGO, M. CORREIA, P. COSTA, M. PASIN, F. A. B. SILVA. *Étude d'une architecture MapReduce tolérant les fautes byzantines.*, in "Actes des 20éme Rencontres francophones du parallélisme (RENPAR'11)", May 2011.

[60] L. BLIN, S. DOLEV, M. POTOP-BUTUCARU, S. ROVEDAKIS. *Construction auto-stabilisante d'un arbre couvrant de poids minimum*, in "13es Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications (AlgoTel)", Cap Estérel, France, B. DUCOURTHIAL, P. FELBER (editors), 2011, http://hal.inria.fr/inria-00587591/en.

[61] S. DOLEV, S. DUBOIS, M. POTOP-BUTUCARU, S. TIXEUIL. *Communication Optimalement Stabilisante sur Canaux non Fiables et non FIFO*, in "CoRR Proceedings of Algotel 2011", 2011, vol. abs/1104.3947, http://arxiv.org/abs/1104.3947.

[62] F. HERMENIER, J. LAWALL, G. MULLER, J.-M. MENAUD. *Consolidation dynamique d'applications Web haute disponibilité*, in "Proceedings of the Conférence Française en Systèmes d'Exploitation (CFSE)", May 2011.

[63] T. PREUD'HOMME, J. SOPENA, G. THOMAS, B. FOLLIOT. *BatchQueue : file producteur / consommateur optimisée pour les multi-cœurs.*, in "8ème Conférence Française sur les Systèmes d'Exploitation (CFSE'11), Chapitre français de l'ACM-SIGOPS, GDR ARP", May 2011.

[64] P. SUTRA, M. SHAPIRO. *Résolution efficace du consensus généralisé dans les systèmes répartis par passage de messages*, in "ALGOTEL 2011 - 13es Rencontres Francophones sur les Aspects Algorithmiques de Télécommunications", Cap Estérel, France, B. DUCOURTHIAL, P. FELBER (editors), April 2011, http://hal.inria.fr/inria-00586591/en.

[65] M. VALERO, L. ARANTES, M. POTOP-BUTUCARU, P. SENS. *Architectures de filtrage reconfigurables dynamiquement.*, in "Conférence Française en Systèmes d'Exploitation", 2011.

Scientific Books (or Scientific Book chapters)

[66] X. BONNAIRE, P. SENS. *Design Principles of Large-Scale Distributed System*, in "Distibuted Systems: Design and Algorithms", Wiley, 2011.

[67] P. DARCHE. *Architecture des ordinateurs-Mémoires à semi-conducteurs : Principe de fonctionnement et organisation interne des mémoires vives*, Editions Vuibert, 2011, vol. 1.

[68] O. MARIN, S. MONNET, G. THOMAS. *Peer-to-Peer Storage*, in "Distibuted Systems: Design and Algorithms", Wiley, 2011, p. 59-80.

[69] S. MONNET, G. THOMAS. *Large-Scale Peer-to-Peer Game Applications*, in "Distibuted Systems: Design and Algorithms", Wiley, 2011, p. 81-103.

Books or Proceedings Editing

[70] A. K. DATTA, F. PETIT, R. GUERRAOUI (editors). *Journal of Theoretical Computer Science, Special Issue on Stabilization, Safety, and Security of Distributed Systems*, Elsevier, 2011, vol. 412, n^{o} 33.

[71] A. K. DATTA, F. PETIT, R. GUERRAOUI (editors). *Transactions on Autonomous and Adaptive Systems, Special Issue on Stabilization, Safety, and Security of Distributed Systems*, ACM, 2011.

[72] X. DÉFAGO, F. PETIT, V. VILLAIN (editors). *Proceedings of the 13th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2011)*, Lecture Notes in Computer Science, Springer, Grenoble, France, 2011, vol. 6976.

Research Reports

[73] S. DEVISMES, A. LAMANI, F. PETIT, P. RAYMOND, S. TIXEUIL. *Optimal grid exploration by asynchronous oblivious robots*, UPMC, 2011, 21 pages, http://hal.inria.fr/hal-00591963/en.

[74] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Maximum Metric Spanning Tree made Byzantine Tolerant*, UPMC, April 2011, http://hal.inria.fr/inria-00589234/en.

[75] S. DUBOIS, T. MASUZAWA, S. TIXEUIL. *Self-Stabilization, Byzantine Containment, and Maximizable Metrics: Necessary Conditions*, UPMC, March 2011, http://hal.inria.fr/inria-00577062/en.

[76] S. DUBOIS, S. TIXEUIL. *A Taxonomy of Daemons in Self-stabilization*, UPMC, October 2011, 26 pages, http://hal.inria.fr/hal-00628390/en.

[77] F. GREVE, P. SENS, L. ARANTES, V. MARTIN. *Asynchronous Implementation of Failure Detectors with partial connectivity and unknown participants*, INRIA, 2011, n^{o} RR-6088, http://hal.inria.fr/inria-00122517/en.

[78] F. HERMENIER, J. LAWALL, J.-M. MENAUD, G. MULLER. *Dynamic Consolidation of Highly Available Web Applications*, INRIA, February 2011, n^{o} RR-7545, http://hal.inria.fr/inria-00567102/en.

[79] N. HIDALGO, L. ARANTES, P. SENS, X. BONNAIRE. *TPT-C: A Heuristic-Based Cache to Improve Range Queries over DHTs*, INRIA, March 2011, n^{o} RR-7576, http://hal.inria.fr/inria-00578189/en.

[80] S. LEGTCHENKO, S. MONNET, P. SENS. *DONUT: Building Shortcuts in Large-Scale Decentralized Systems with Heterogeneous Peer Distributions*, INRIA, May 2011, n^{o} RR-7614, http://hal.inria.fr/inria-00591922/en.

[81] M. SHAPIRO, N. PREGUIÇA, C. BAQUERO, M. ZAWIRSKI. *A comprehensive study of Convergent and Commutative Replicated Data Types*, inria, rocq, January 2011, n^{o} 7506, http://hal.archives-ouvertes.fr/inria-00555588/.

[82] M. SHAPIRO, N. PREGUIÇA, C. BAQUERO, M. ZAWIRSKI. *A comprehensive study of Convergent and Commutative Replicated Data Types*, INRIA, January 2011, n^{o} RR-7506, http://hal.inria.fr/inria-00555588/en.

[83] M. SHAPIRO, N. PREGUIÇA, C. BAQUERO, M. ZAWIRSKI. *Conflict-free Replicated Data Types*, inria, rocq, July 2011, n^{o} RR-7687, http://hal.inria.fr/inria-00609399/en/.

[84] M. SHAPIRO, N. PREGUIÇA, C. BAQUERO, M. ZAWIRSKI. *Conflict-free Replicated Data Types*, INRIA, July 2011, n^{o} RR-7687, http://hal.inria.fr/inria-00609399/en.

Other Publications

[85] T. IZUMI, M. POTOP-BUTUCARU, M. VALERO. *Physical expander in Virtual Tree Overlay*, 2011, http://hal.inria.fr/inria-00569098/en.