INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Team Regal

Resource management in large scale distributed systems

Paris - Rocquencourt

Theme : Distributed Systems and Services

Activity Report

2009

Table of contents

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

1. Team

Research Scientist

Gilles Muller [DR since September, HdR]
Marc Shapiro [DR, HdR]
Mesaac Makpangou [CR, HdR]

Faculty Member

Pierre Sens [Team Leader, Professor Université Paris 6, HdR]
Luciana Arantes [Associate Professor Université Paris 6]
Bertil Folliot [Professor Université Paris 6, HdR]
Maria Gradinariu Potop-Butucaru [Associate Professor Université Paris 6]
Olivier Marin [Associate Professor Université Paris 6]
Sébastien Monnet [Associate Professor Université Paris 6]
Franck Petit [Professor Université Paris 6, HdR]
Gaël Thomas [Associate Professor Université Paris 6]
Ikram Chabbouh [Assistant Professor Université (ATER) Paris 6 until June]

Technical Staff

Jean-Michel Busca [Research Engineer until June]
Pierpaolo Cincilla [Junior Engineer since September]
Véronique Simon [Associate Engineer until October]

PhD Student

Lamia Benmouffok [Microsoft research grant - Université Paris 6]
Mathieu Bouillaguet [Université Paris 6]
Charles Clement [Université Paris 6 until September]
Swan Dubois [Université Paris 6 until September]
Nicolas Geoffray [Université Paris 6 until September]
Nicolas Hidalgo [Université Paris 6]
Anthony Hocquet [Université Paris 6 since September]
Anissa Lamani [Université Amiens since September]
Sergey Legtchenko [Université Paris 6 since September]
Corentin Méhat [Université Paris 6]
Thomas Preud'homme [Université Paris 6]
Erika Rosas [Université Paris 6]
Julien Sopena [Université Paris 6]
Pierre Sutra [Université Paris 6]
Mathieu Valéro [Université Paris 6]

Post-Doctoral Fellow

Olivier Pérès [Postdoc since September]
Corentin Travers [Postdoc since September]

Visiting Scientist

Julia Lawall [Lecturer University of Copenhagen]

Administrative Assistant

Nadia Mesrar [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.

2.2. Highlights of the year

- Google Research Award “CRDTs: Consistency without concurrency control” awarded to Marc Shapiro (cf. Section 7.2.1).

3. Scientific Foundations

3.1. Scientific Foundations

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.

Two 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, renater or VTHD.

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.

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: computational grids 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. Grids 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.

In Grid configurations we address resource management for two kinds of applications:

- Multi-agent applications which model complex cooperative behaviors.
- Application Service Provider (ASP) environments in cooperation with the DIET project of the GRAAL team.

Our third 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. Pastis: A peer-to-peer file system

Participants: Pierre Sens [correspondent], Jean-Michel Busca.

Pastis is a distributed multi-writer file system. It aims at making use of the aggregate storage capacity of hundreds of thousands of PCs connected to the Internet by means of a completely decentralized peer-to-peer (P2P) network. Replication allows persistent storage in spite of a highly transient node population, while cryptographic techniques ensure the authenticity and integrity of file system data.

Routing and data storage in Pastis are handled by the Pastry routing protocol and the PAST distributed hash table (DHT). The good locality properties of Pastry/PAST allow Pastis to minimize network access latencies, thus achieving a good level of performance when using a relaxed consistency model. Moreover, Pastis does not employ heavy protocols such as BFT (Byzantine Fault Tolerance), like other P2P multi-writer file systems do. In Pastis, for a file system update to be valid, the user must provide a certificate signed by the file owner which proves that he has write access to that file.

5.2. LS3: Large Scale Simulator

Participants: Pierre Sens [correspondent], Jean-Michel Busca.

LS3 is a discrete event simulator originally developed for Pastis, a peer-to-peer file system based on Pastry. LS3 allows to build a network of tenths of thousands nodes on a single computer, and simulate its execution by taking into account message transmission delays. LS3 transparently simulates communication layers between nodes, and executes the same application code (including Pastry, Past and higher layers) as in a real execution of the system.

LS3's modular design consists of three independent layers, allowing the simulator to be reused in areas other than Pastis and Pastry:

- At the kernel level, the system being simulated is described in a generic way in terms of entities triggered by events. Each entity has a current state and a current virtual time, and can be programmed either in synchronous mode (blocking wait of the next event) or in asynchronous mode (activation of an event handler). A multi-threaded event engine delivers events in chronological order to each entity by applying a conservative scheduling policy, based on the analysis of event dependencies.
- At the network level, the system being simulated is modeled in terms of nodes sending and receiving messages, and connected through a network. The transmission delay of a message is derived from the distance between the sending and the receiving nodes in the network, according to the selected topology. Three topologies can be used: local network (all nodes belong to the same local network), two-level hierarchy (nodes are grouped into LANs connected through WANs) and sphere (nodes are located on a sphere). It is possible to set the jitter rate of transmission delays, as well as the rate of message loss in the network.
- The stubs Pastry level interfaces Pastry with LS3: it defines a specialization of Pastry nodes that allows them to interface with standard LS3 nodes. Several parameters and policies that drive the behaviour and the structure of a Pastry network can be set at this level, including: the distribution of node ids, the selection of boostrap nodes, the periodicity of routing tables checks and the rate of node churn. It is also possible to simulate the ping messages that nodes send to supervise each other, and set failure detection thresholds.

Some figures: LS3 can simulate a network of 20 000 Pastry nodes with no application within 512 Mb of RAM, and it takes approximately 12 minutes on a single processor Pentium M 1,7 GHz to build such network. When simulating the Pastis application, event processing speed is about 500 evt/s. The speedup factor depends on the simulated load: as an example, speedup ranges from 20 for a single user to 0,05 for 400 simultaneous users.

5.3. Telex

Participants: Marc Shapiro [correspondent], Lamia Benmouffok, Jean-Michel Busca, Pierre Sutra, Georgios Tsoukalas, 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.

This work is published at CFSE 2009 [21]. The code is freely available on http://gforge.inria.fr under a BSD license.

5.4. Treedoc

Participants: Marc Shapiro [correspondent], Olivier Pérès, Mihai Letia.

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.

This work is published at ICDCS 2009 [46] and LADIS 2009 [44]. The code is freely available on http://gforge.inria.fr under a BSD license.

5.5. VMKit and .Net runtimes for LLVM

Participants: Bertil Folliot [correspondent], Nicolas Geoffray, Gaël Thomas, Charles Clément, 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 2009, we focused our research on the following areas:

- distributed algorithms for large and dynamic networks,
- Peer-to-peer storage
- 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.
- dynamic replication in distributed multi-agent systems.

6.2. Distributed algorithms

Participants: Luciana Arantes [correspondent], Maria Gradinariu [correspondent], Mathieu Bouillaguet, 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. We propose solutions for both static (eg. grid) and dynamic networks (P2P and mobile networks).

6.2.1. Static systems

In 2009, we have proposed a fault tolerant permission-based k-mutual exclusion which does not rely on timers, nor does on failure detectors, neither needs extra messages for detecting node failures. Fault tolerance is integrated in the algorithm itself and it is provided if the underlying system guarantees a Responsiveness Property. Based on Raymond's algorithm, our algorithm exploits the REQUEST-REPLY messages exchanged by processes to get access to one of the k units of the shared resource in order to dynamically detect failures and adapt the algorithm to tolerate them. This work was published in [27].

Recently we started to investigate two communication abstractions in asynchronous systems under various class of faults. The first abstraction deals with synchronizing logical clocks of neighboring nodes also known as unisson. We study the FTSS (fault tolerant and self-stabilizing) version of the problem in asynchronous settings. We addressed both the crash and Byzantine faults in [39]. The major contribution of our work steams in exploring for the first time the limits of FTSS unisson in asynchronous setting exploring both the impossibility and possibility results.

The second abstraction addresses the FTSS coloring of undirected networks. Coloring has a direct application in the implementation of TDMA communication which is one of the most efficient collision free communication primitives for adhoc networks. In [23] we propose some impossibility results and a deterministic solution that work under restricted schedulers. We extend the study in [22] by proposing probabilistic solutions for asynchronous networks.

6.2.2. Dynamic systems

In this context 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 leader election, membership and naming.

- In 2009, we start exploiting the dynamics of MANETs in order to propose a distributed computing model that characterize as much as possible the dynamic and self-organizing behavior of MANETs'. The temporal variations in the network topology implies that MANET 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 MANET 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. To this end, a first work is published in [20] and a second work has been submitted to publication [58].
- One of the main challenges of Delay-Tolerant Networks (DTNs) is on how to define effective routing protocols. Both the dynamics of DTNs and network disruptions make the choice of a routing protocol and its performance evaluation non trivial tasks. Hence, our proposal was to use a graph theoretic model, in particular the Evolving Graph (EG) theory, in order to provide a framework for evaluating least cost routing algorithms which exploit different metrics. Concisely, an EG is a time-step indexed sequence of subgraphs, where the subgraph at a given time-step corresponds to the network connectivity at the time interval indicated by the time-step value. The results of the above mentioned evaluation has been published in [19].
- The main challenges of our research activity over 2009 year were to develop self* (self-stabilizing, self-organizing and self-healing) local algorithms for dynamic networks (P2P, sensor and robot networks). We addressed fundamental problems such as constructions of fault tolerant and reliable infrastructures for networks hit by topological dynamicity. In [24] we study the construction of self-stabilizing Steiner trees in dynamic netwprks. In [25] we address the one to optimal self-stabilizing construction of minimum spanning trees while in [26] we extend the study to the loop-free solutions. That is, solutions where during the dynamicity periods the existing tree is always maintained. Furthermore we investigated fault-tolerant agreement in robot networks under various forms of constraints([29], [28], [30]).
- Another 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.

6.3. Peer-to-peer systems

Participants: Pierre Sens [correspondent], Jean-Michel Busca, Nicolas Hidalgo, Sergey Legtchenko, Sébastien Monnet, Gilles Muller, Corentin Travers, Véronique Simon, Mathieu Valéro.

6.3.1. Peer-to-peer storage

Since 2003, we develop Pastis [7] is a new completely decentralized multi-user read-write peer-to-peer file system. Pastis is based on the FreePastry Distributed Hash Table (DHT) of the Rice University. 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. Since 2007, we study the effects of churn on Pastis, a DHT-based peer-to-peer file system. We evaluate the behavior of Pastis under churn, and investigate whether it can keep up with changes in the peer-to-peer overlay. We used a modified version of the PAST DHT to provide better support for mutable data and to improve tolerance to churn. Our replica regeneration protocol distinguishes between mutable blocks and immutable blocks to minimize the probability of data loss. Read-write quorums provide a good compromise to ensure replica consistency under the presence of node failures. Our experiments use Modelnet to emulate wide-area latencies and the asymmetric band- width of ADSL client links. The results show that Pastis preserves data consistency even at relatively high levels of churn.

However, 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 [43], 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. Peer-to-peer overlay

Peer-to-peer overlays allow distributed applications to work in a wide-area, scalable, and fault-tolerant manner. However, most structured and unstructured overlays present in literature today are inflexible from the application viewpoint. In other words, the application has no control over the structure of the overlay itself. We proposed the concept of an application-malleable overlay, and the design of the first malleable overlay: MOve. In MOve, the communication characteristics of the distributed application using the overlay can influence the overlay's structure itself, with the twin goals of (1) optimizing the application performance by adapting the overlay, while also (2) retaining the scale and fault-tolerance of the overlay approach. The influence could either be explicitly specified by the application or implicitly gleaned by our algorithms. Besides neighbor list membership management, MOve also contains algorithms for resource discovery, update propagation, and churn-resistance. The emergent behavior of the implicit mechanisms used in MOve manifest in the following way: when application communication is low, most overlay links keep their default configuration; however, as application communication characteristics become more evident, the overlay gracefully adapts itself to the application.

We are considering a new class of target applications: massively multi-player online games (MMOG) such as virtual worlds. Within the context of a project funded by the LIP6, we have modeled a P2P distribution of such applications. Following this model, groups of object replicas are moving among peers while players evolve in the virtual world. For this kind of applications, it is important that the underlying overlay is flexible in order to remain adapted to the changing application structure. Since 2009, we are investigating an even more dynamic kind of overlay: a malleable overlay that anticipate application needs in order to adapt itself in advance. Thanks to this anticipation, the overlay is already operational and efficient when the application uses it.

6.3.3. Peer-to-peer publish-subscribe

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. The enhanced structure performs better than the traditional distributed R-tree in terms of delivery latency. Additionally, it does not alter the performances related to the scalability, nor the load balancing of the tree, and neither the rate of false positives and negatives filtered by a node. The results of this work can be found in [57] which was also submitted to publication.

6.4. Virtual machine (VM)

Participant: Bertil Folliot.

Our research interest are in computer systems, particularly operating systems and virtual machines. We focus on resource management, isolation and concurrency management in virtual machines. Since September 2008, we started with Gilles Muller a new complementary research theme on dynamic patches of operating systems.

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. I 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. Nicolas Geoffray has defended his PhD thesis in September on the subject.

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 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. Recently, we have begun using Coccinelle to generate traditional patches for improving the safety of Linux. Some Linux developers have also begun to use the tool. Over 170 of these patches developed using Coccinelle have been integrated into the mainline Linux kernel, and more have been accepted by Linux maintainers and are pending integration. Our current work is to build on the results of Coccinelle by designing libraries of semantic patches to identify API protocols and detect violations in their usage. One of the novelty of this work is to explore how to develop these semantic patches in collaborative manner with the community of Linux open-source developers as a target. In this context, we will investigate the usage of the Telex framework for supporting collaborative developments.

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 [45]. 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 jeapordizing the scalability and the performance of the system? (ii) accommodating the dynamic and the heterogenity of a peer-to-peer network with the database requirements?

In 2009, we focus on the partial database replication protocol. We proposed a database access protocol, capable to spread out a transaction's accesses over multiple database fragments replicas while guarenteeing 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 synchronistation from the certification concerns.

A small-scale group of schedulers that do not hold database replicas, cooperate with one antoher to certify update transactions. Only certified transactions are notified to replicas. Futhermore, each replica will be notified only the transactions that impact the that it stores. Thanks to this separation, we avoid waste of computaion 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.

6.6. Fault-Tolerant Partial Replication in Large-Scale Database Systems

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

In distributed systems, information is replicated. Data replication enables cooperative work, improves access latency to data shared through the network, and improves availability in the presence of failures. When the information is updated, maintaining consistency between replicas is a major challenge.

Previous studies of data replication considered different areas separately, often ignoring the requirements of other areas. For instance, OS researchers often assume updates are independent; CSCW researchers ignore conflicts; algorithms research mostly ignores semantics; peer-to-peer systems often ignore mutable data and hence consistency; none of the above have addressed partial replication.

We study optimistic replication for multi-user collaborative applications such as co-operative engineering (e.g., co-operative code development), collaborative authoring (e.g., a decentralized wikipedia), or entreprise information libraries. We propose a general-purpose approach, subsuming the previous work in different areas. It takes addresses respecting application semantics, high-level operations, dependence, atomicity and conflict, long session times, etc.

We investigated a decentralized approach to committing transactions in a replicated database, under partial replication. Previous protocols either reexecute transactions entirely and/or compute a total order of transactions. In contrast, ours applies update values, and generate a partial order between mutually conflicting transactions only. Transactions execute faster, and distributed databases commit in small committees. Both effects contribute to preserve scalability as the number of databases and transactions increase. Our algorithm ensures serializability, and is live and safe in spite of faults.

The work described above takes place in the context of several joint projects: Grid4All, Respire and Prose. It is published at CFSE 2009 [21].

6.7. Optimistic approaches in collaborative editing

Participant: Marc Shapiro [correspondent].

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. This work is published at ICDCS 2009 [46] and LADIS 2009 [44].

6.8. Fault tolerance in multi-agent systems

Participants: Olivier Marin [correspondent], Corentin Méhat.

Distributed agent systems stand out as a powerful tool for designing scalable software. The general outline of distributed agent software consists of computational entities which interact with one another towards a common goal that is beyond their individual capabilities

Our research focuses on middleware to deploy agent on large-scale environments and mobile networks. Our main topics of interest comprise: fault tolerance, process replication, and dynamicity with respect to both environments and applications. The ongoing research projects we are working on are all related to these topics.

The FRAME (Failure Resilient Agents in Mobile Environments) project – funded by LIP6 in 2006 and 2007 – aims at designing a middleware for the deployment of distributed algorithms among mobile devices. The originality of our approach is double: (i) we view partial and total disconnection as types of failures and aim to integrate fault tolerance solutions in order to guarantee the continuity of the computation in such a context, and (ii) we provide a modeling language which is close to Pi-calculus and yet focuses on communication channels in order to represent replicated applications and introduce failures. The ongoing PhD effort of Corentin Méhat is at the core of this project.

Our current work addresses the resiliency of group communications among mobile devices [50]: the exchanged messages are transparently rerouted inside a structured P2P overlay – in our case Pastry – and can thus be accessed asynchronously. This has lead to a new platform design: we are presently implementing the resulting design over FreePastry in order to evaluate its performances.

The DARX (Dynamic Agent Replication eXtension) project aims at building an architecture for fault-tolerant agent computing in multi-cluster networks. The originality of our approach lies in two features: (i) an automated replication service which chooses for the application which of its computational components are to be made dependable, to which degree, and at what point of the execution, and (ii) the hierarchic architecture of the middleware which ought to provide suitable support for large-scale applications. DARX is now a component of the FACOMA project, which is supported in the context of the ANR-SETIN frame. FACOMA was originally supposed to end in 2009, but its extension has been approved by the ANR until 2010.

The latest advances include building a distributed exception-handling system which can be shared by the agent application and the dynamic replication service, and integrating heuristics on the system-level servers in order to drive the load-balancing decisions related to replicating agents.

The DDEFCON (Dependable DEployment oF Code in Open eNvironments) project addresses the safe and secure deployment of collaborative software components over large-scale networks. We seek to achieve a deployment platform that can be implemented on top of a structured peer to peer overlay. DDEFCON is funded as part of the LIP6 young projects initiative; it started in 2008, and has been extended for a second year. It serves as a basis for the PhD thesis of Nicolas Gibelin.

We are currently working on a DHT based service for registering resources and allowing multi-criteria searches of these resources. We have already implemented and tested two different implementations on a local cluster. The results are promising, and we are now deploying our implementations on Grid5000 for further performance evaluations.

7. Other Grants and Activities

7.1. National initiatives

7.1.1. PROSE - (2009–2011)

Members: THOMSON, 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.

7.1.2. ABL - (2009–2011)

Members: Gilles Muller, Gaël Thomas

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. Some Linux developers have also begun to use the tool. Over 400 of these patches developed using Coccinelle have been integrated into the mainline Linux kernel, and more have been accepted by Linux maintainers and are pending integration. In the ABL project, we will build 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.

7.1.3. 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.

7.1.4. 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.

7.1.5. SPREADS - (2008–2010)

Members: UbiStorage, LACL, Inria Sophia, Inria (Regal)

Funding: SPREADS project is funded by ANR TELECOM

Objectives: This project proposes a collaborative research effort to study and design highly dynamic secure P2P storage systems on large scale networks like the Internet. The scientific program covered by this proposal is mainly the design of new mathematical safety, security and performance models, secure patterns, simulation to evaluate the quality of service of a peer-to-peer storage system in the context of a dynamic large scale network. These models and simulations will eventually be corroborated by experimentation on the Grid 5000 and Grid eXplorer Platforms.

7.1.6. Facoma - (2007–2009)

Members: LIP6, LIRMM, Regal

Funding: Facoma project is funded by ANR SETIN

Objectives: The fault tolerance research community has developped solutions (algorithms and architectures), mostly based on the concept of replication, applied for instance to data bases. But, these techniques are almost always applied explicitely and statically. This is the responsability of the designer of the application to identify explicitly which critical servers should be made robust and also to decide which strategies (active or passive replication) and their configurations (how many replicas, their placement). Meanwhile, regarding new cooperative applications, which are very dynamic, for instance: decision support systems, distributed control, electronic commerce, crisis management systems, and intelligent sensors networks, - such applications increasingly modeled as a set of cooperative agents (multi-agent systems) -, it is very difficult, or even impossible, to identify in advance the most critical agents of the application. This is because the roles and relative importances of the agents can greatly vary during the course of computation, interaction and cooperation, the agents being able to change roles, strategies, plans, and new agents may also join or leave the application (open system). Our approach is in consequence to give the capacity to the multi-agent system itself to dynamically identify the most critical agents and to decide which abilisation strategies to apply to them.

7.1.7. PlayAll - (2007–2009)

Members: PME: Darkwoks, Atonce, Bionatics, Fandango Games, Load Inc, Kilotonn, Sixela, SpirOps, Voxler, White Birds, Wizrbox - Public: CNAM, ENST, ENJMIN, LIP6 (REgal), LIRIS

Funding: PLAYALL project is funded by Pôle de Compitivité - Cap Digital

Objectives: The goal is the build a middleware adapted to the different game platforms (Sony Play Station 3, Nitendo DS, Wii, Xbox, PC). The contribution of Regal concerns distributed algorithms taking into account QoS contraints.

7.1.8. Fracas - (2007–2009)

Members: ARES (Rhones-Alpes), DIONYSOS (IRISA), Grand-Large (Futurs), Regal(Paris-Rocquencourt)

Funding: Fracas is funded by INRIA (Action de Recherche Coopérative)

Objectives: We propose to define a new middleware dedicated for sensor networks. This middleware must tolerate failures and specific attacks these networks are subject.

7.1.9. PACTOL - (2009–2011)

Members: LIP6 (NPA, Regal), CNAM

Funding: Digiteo

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

7.1.10. MOTAR2 - 2009

Members: NPA, Regal (LIP6)

Funding: LIP6

Objectives: The study of fault tolerance in robot networks.

7.2. European initiatives

7.2.1. Google Research Award "CRDTs: Consistency without concurrency control"

A CRDT is a data type whose operations commute when they are concurrent. Replicas of a CRDT eventually converge without any complex concurrency control or the need for any centralised component. This makes CRDTs very appealing for managing data in large-scale environments, such as cloud computing or web-based environments. We have previously developed two non-trivial CRDTs: a shared edit buffer, Treedoc, and a graph structure, the multilog. This work allowed us to identify some general properties for the design of CRDTs [46], [44]. The goal of this work is to generalise this approach to manage data in large-scale environments, by designing CRDTs for specific problems (e.g. replicated key-value store as used in Amazon Dynamo).

The principal investigators of this award are Marc Shapiro and Nuno Preguiça of UNL. This award includes a grant of $80 000 over one year.

7.2.2. Grant from Microsoft Research Cambridge

Data replication enables cooperative work, improves access latency to data shared through the network, and improves availability in the presence of failures. This grant supports a doctoral student for studying consistency between replicas of mutable, semantically-rich data in a peer-to-peer fashion. This study should enable to engineer distributed systems and applications based on them, supporting cooperative applications in large-scale collaboration networks. It includes a systematic exploration of the solution space, in order to expose the cost vs. performance vs. availability vs. quality trade-offs, and understanding fault tolerance and recovery aspects. This work combines formal approaches, simulation, implementation, and measurement.

7.2.3. Grid4All - (2006-2009)

Members: France Télécom Recherche et Développement, INRIA (Regal, Atlas and Grand-Large), SICS, KTH, ICCS, UPRC, UPC, Redidia.

Funding: European Commission, 6th Framework Programme, STREP (Specific Targeted Research Project)

Objectives: Grid4All embraces the vision of a "democratic" Grid as a ubiquitous utility whereby domestic users, small organizations and enterprises may draw on resources on the Internet without having to individually invest and manage computing and IT resources. This project is funded by the 6th Framework Programme of the European Commission. It involves institutional and industrial partners. Its budget is slightly over 4.8 million euros.

Grid4All has the following objectives:

- To alleviate administration and management of large scale distributed IT infrastructure, by pioneering the application of component based management architectures to self-organizing peer-to-peer overlay services.

- To provide self-management capabilities, to improve scalability, resilience to failures and volatility thus paving the way to mature solutions enabling deployment of Grids on the wide Internet.
- To widen the scope of Grid technologies by enabling on-demand creation and maintenance of dynamically evolving scalable virtual organisations even short lived.
- To apply advanced application frameworks for collaborative data sharing applications executing in dynamic environments.
- To capitalize on Grids as revenue generating sources to implement utility models of computing but using resources on the Internet.

Grid4All will help to bring global computing to the broader society beyond that of academia and large enterprises by providing an opportunity to small organisations and individuals to reap the cost benefit of resource sharing without however the burdens of management, security, and administration.

The consortium will demonstrate this by applying Grid4All in two different application domains: collaborative tools for e-learning targeting schools and digital content processing applications targeting residential users.

7.2.4. FTH-GRID - (2009–2010)

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

Funding: Egyde

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.

7.3. International initiatives

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 (SUA) 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 Fundação para a Ciência e a Tecnologia (FCT, Portuguese equivalent of ANR). Finally, Marc Shapiro is a Member of the CITI Advisory Board.

8. Dissemination

8.1. Program committees and responsibilities

Luciana Arantes is:

- Member of the program committee of the 6ème Conférence française sur les systèmes d'exploitation, CFSE-6, Friburg, Switzerland, february 2008.
- Member of PC of Workshop de Sistemas Operacionais, SBC, Brésil, 2009.
- Member of PC of WTF 2009 - Workshop of Fault Tolerance, Brésil, 2009
- Member of PC of International Conference on Grid and Pervasive Computing (2009-2010).
- Reviewer for JPDC and TPDS journals.

Bertil Folliot is:

- Head of the Network and Distributed Systems department at LIP6.
- Elected member of the “Commission de spécialistes” of the Paris 6 University.
- Head of selection committee for recruiting an associate professor.
- Member of the scientific committee of LIP6.
- Co-chair of the middleware group of GdR ASR (Hardware, System and Network), until, until september 2009.
- 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 2009 High Performance Computing & Simulation Conference (HPCS'09), Leipzig, Germany, june 2009.
- Member of the program committee of the 7th International Conference on the Principles and Practice of Programming in Java, Calgary, Alberta, Canada, august 2009.
- Member of the program committee of the 2010 International Conference on High Performance Computing & Simulation (HPCS 2010), Caen, France, july 2010.
- Member of the program committee of the 8th International Conference on the Principles and Practice of Programming in Java, Vienna, Austria, september 2010.

Maria Gradinariu Potop-Butucaru is:

- Member of the program commitee of Algosensor 2009 (Fifth International Workshop on Algorithmic Aspects of Wireless Sensor Networks)
- Member of the program commitee SSS 2009 (Internation Symposium on Self-stabilizing and Secure Systems)
- Member of the program committee ISORC 2009 (International Symposium Object/component/service oriented real-time distributed computing)
- Member of the program commitee Algotel 2009 (11eme rencontres francophones sur les aspects algorithmiques de telecommunications)
- Co-chair of Algotel 2010 (12eme rencontres francophones sur les aspects algorithmiques de telecommunications)
- Member of the program commitee of ICDCN 2010 (International Conference on distributed computing and networking)
- Member of the University Pierre et Marie Curie UFR Computer Science Council
- Reviewer for Distributed Computing, SIAM Journal of Computing, IEEE TPDS, IEEE Transactions on Computers, Theoretical Computer Science Journal ...

Olivier Marin is:

- Member of the board of Distributed Systems Online (DSO)
- Community editor for DSO Distributed Agents

Sébastien Monnet is:

- Member of the PC of the first 1st International Workshop on Fault-Tolerance for HPC at Extreme Scale (held with DSN 2010, Chicago, Illinois, USA, June 2010).
- Member of the program commitee of the storage track for the 5th IEEE International Conference on Networking, Architecture, and Storage (NAS 2010) July 2010, Macau SAR, China.
- GDR ASR mailing lists moderator

Gilles Muller is:

- Member of PC of the DSN 2010 conference, June 2010 http://www.dsn.org/.
- Chair of PC of the EuroSys 2010 conference, April 2010 http://eurosys2010.sigops-france.fr/.
- Chair of PC of the PLOS'09 workshop, October 2009 http://www.plosworkshop.org/2009/.
- Member of PC of the SSS'09 conference, track on multicore computing, November 2009 http://graal.ens-lyon.fr/SSS09/.
- Member of PC of the 7ème Conférence française sur les systèmes d'exploitation, CFSE-7, October 2009 http://www.irit.fr/Toulouse2009/.
- Member of PC of the ICWS'09 conference, July 2009 http://conferences.computer.org/icws/2009.
- Member of PC of the ACP4IS 2009 workshop, March 2009 http://www.aosd.net/workshops/acp4is/2009/.
- Member of the jury of the best European thesis on systems (EuroSys) 2009.
- Member of the evaluation committee of the ANR ARPEGE program.
- Organizer with Julia Lawall of the 1st Coccinelle Workshop, Lip6, November 2009.

Franck Petit is:

- Co-program chair, chair of stabilization track, and co-organizing chair of SSS 2009, International Symposium on Stabilization, Safety, and Security of Distributed Systems,Ed. LNCS, Nov. 2009, Lyon, France.
- Member of PC of WRAS 2009, 2^{nd} International Workshop on Reliability, Availability, and Security, Ed. IEEE, Dec. 2009, Hiroshima, Japan.
- Member of PC of PODC 2010, Twenty-Ninth Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC 2010), Ed. ACM, July 2010, Zürich, Switzerland.
- Invited Talk in *PDCAT'09, International Conference on Parallel and Distributed Computing, Applications, and Technologies*, Dec. 2009, Hiroshima, Japan. Title: Pattern Formation and Leader Election in Swarms of Robots.
- Invited Talk in Workshop APRETAF: Algorithmes Parallèles, Répartis Et Tolérance Aux Fautes, LIG/VERIMAG, Janv. 2009, Grenoble, France. Title: Stabilisation et synchronisation d'horloges logiques.

Pierre Sens is:

- Global chair of Topic 8 "Distributed systems and algorithms" of EuroPar 2010
- co-Chair of DAMAP 2009: Data Management in Peer-to-peer system, Workshop in conjunction EDBT, St. Petersburg, March, 2009
- Invited keynote speaker of 14th International Conference on Reliable Software Technologies, Ada-Europe 2009. Title: Fault-tolerance issues in large scale systems.
- Member of PC of OPODIS 2009 (International Conference On Principles Of Distributed Systems).
- Member of PC of SSS 2009 (International Symposium on Stabilization, Safety, and Security of Distributed Systems).
- Member of PC of the 7ème Conférence française sur les systèmes d'exploitation, CFSE-7, October, 2009.
- Member of PC of COLIBRI, COLloque d'Informatique: Brésil / INRIA, Coopérations, Avancées et Défis, 2009.
- 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.
- Elected member of the "Institut de Formation Doctorale" of Paris 6 University.
- Reviewer for JPDC and TPDS journals.
- Reviewer of ANR project Telecom, Blanc, Jeunes Chercheurs

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).
- PC co-chair at LADIS 2009 (Large-Scale Distributed Systems and Middleware).
- Member of PC of EuroSys 2010.
- Member of PC of CFSE 2009 (Conférence française 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 Swedish Research Council (Vetenskapsrådet).
- Reviewer for Springer Distributed Computing.
- Reviewer for IEEE Transactions on Parallel and Distributed Systems (TPDS).
- Chair, ACM Distinguished Service Award Commitee 2009 http://awards.acm.org/distinguished%5Fservice/
- Member, ACM Europe Council http://europe.acm.org/.
- Member, ACM Taskforce on Chapters.
- Co-chair, ACM Europe Subcommitee on Members and Awards.
- Member, committee on “ICT scientific societies at the dawn of the 21st century” advising the European Commission Directorate General Information Society and Media.

8.2. PhD reviews

Bertil Folliot was the reviewer of:

- Fabien Hermenier. PhD Université de Nantes. Gestion dynamique des tâches dans les grappes, une approche à base de machines virtuelles. (Advisors: Gilles Muller, Jean-Marc Menaud), Nantes, November 2009.
- Benoit Claudel, PhD INPG Grenoble, Mécanismes logiciels de protection mémoire (Advisor: Olivier Gruber, LIG), December 2009.

Olivier Marin was part of the PhD committee (examinateur) of:

- Khaled Barbaria, "Architectures intergicielles pour la tolÃ©rance aux fautes et le consensus", 19 septembre 2008, (Advisor: Laurent Pautet, Telecom ParisTech)

Gilles Muller was the reviewer of

- Fabienne Boyer, HDR Université de Grenoble, Gestion de l’adaptabilité dans les applications réparties, December 2009.
- Benoit Claudel, PhD INPG Grenoble, Mécanismes logiciels de protection mémoire (Advisor: Olivier Gruber, LIG), December 2009.
- Youssef Laarouchi, PhD INSA Toulouse, Développement d’architectures de sécurités pour les applications à criticités multiples en avionique (Advisor: J. Arlat, LAAS), November 2009.

Pierre Sens was the reviewer of:

- G. Antoniu. HDR ENS Cachan, March 2009
- E. Almeidia, PhD Univ. Nantes, Test et validation des systèmes pair-à-pair (Advisor: P. Valduriez)
- G. Philippe, PhD INPG, Auto-adaptation du Niveau Service dans les Systèmes Distribués, February 2009 (Advisor: J. Mossière).
- A. Ortiz, PhD IRIT, Contrôle de la concurrence dans les grilles informatiques. Application au projet ViSaGe, December 2009 (Advisor: A. Zhougi).
- N. Ouahla, PhD EURECOM, Sécurité et coopération pour le stockage de données pair-à-pair, June 2009.
- A. Rezmerita, PhD Univ. Paris 11, Contribution aux intergiciels et protocoles pour les grappes virtuelles, September 2009 (Advisor: J. Beauquier).
- G. Secret, PhD Univ. Picardie, La maintenance des données dans les systèmes de stockage pair-à-pair, December 2009.
- S. Sicard, PhD INPG, Vers de l’auto-adaptation dans les Systèmes Répartis, March 2009 (Advisor: J-B. Stefani).

Marc Shapiro was member of thesis commitee

- Mateo Varvello, Télécom ParisTech and Eurécom, Sophia-Antipolis (Advisors: Christophe Diot and Ernst Biersack).

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
  - Principles of operating systems in Licence d'Informatique, 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 Gradinariu Potop-Butucaru
  - Principles of Operating Systems, Licence d'Informatique, Université Paris 6
  - Distributed algorithms, Master d'Informatique, Université Paris 6
- Mesaac Makpangou
  - Client/server architecture, Licence professionelle d'Informatique, Université Paris 6: TD, 16 heures
- Oliver Marin
  - Operating system programming, Master d'Informatique, Université Paris 6
  - Operating system Principles, Licence d'Informatique, Université Paris 6
  - Parallel and distributed systems, Master d'Informatique, Université Paris 6
  - Client/server arcitecture, Licence professionelle d'Informatique, Université Paris 6
- Sébastien Monnet
  - Responsible of "Middleware for advanced computing systems in Master 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
  - System and Internet programmation in Licence d'Informatique (2), Université 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.", vol. 101, n^{o} 5, 2007, 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", vol. 11, n^{o} 1, 2001, 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", vol. 20, n^{o} 1, 2007, 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", vol. 66, 2006, p. 128–144.

[5] M. BERTIER, O. MARIN, P. SENS. *Implementation and performance of an adaptable failure detecto r*, 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] J.-M. BUSCA, F. PICCONI, P. SENS. *Pastis: a Highly-Scalabel Multi-User Peer-to-Peer File Systems*, in "Euro-Par’05 - Parallel Processing, Lisboa, Portugal", Lecture Notes in Computer Science, Springer-Verlag, August 2005.

[8] A.-M. KERMARREC, A. ROWSTRON, M. SHAPIRO, P. DRUSCHEL. *The IceCube approach to the reconciliation of divergent replicas*, in "20th Symp. on Principles of Dist. Comp. (PODC), Newport RI (USA)", ACM SIGACT-SIGOPS, August 2001.

[9] 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.

[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] F. Ogel, G. Thomas, A. Galland, B. Folliot. *MVV : une Plate-forme à Composants Dynamiquement Reconfigurables — La Machine Virtuelle Virtuelle*, in "Numéro Spécial Technique et Science Informatiques (TSI)", 2004.

[12] Y. Padioleau, J. Lawall, R. R. Hansen, G. Muller. *Documenting and Automating Collateral Evolutions in Linux Device Drivers*, in "EuroSys 2008, Glasgow, Scotland", March 2008, p. 247–260.

[13] Y. Padioleau, J. Lawall, G. Muller. *Understanding Collateral Evolution in Linux Device Drivers*, in "The first ACM SIGOPS EuroSys conference (EuroSys 2006), Leuven, Belgium", April 2006, p. 59-71, http://hal.inria.fr/inria-00070251/en/, Also available as INRIA Research Report RR-5769.

Year Publications

Doctoral Dissertations and Habilitation Theses

[14] C. Clément. *Isolation des extensions de systèmes d'exploitation dans une machine virtuelle*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, september 2009, Ph. D. Thesis.

[15] N. Geoffray. *Fostering Systems Research with Managed Runtimes*, Université Pierre et Marie Curie (Paris 6), 4, place Jussieu, Paris, september 2009, Ph. D. Thesis.

Articles in International Peer-Reviewed Journal

[16] E. Caron, F. Desprez, F. Petit, C. Tedeschi. *Snap-Stabilizing Prefix Tree for Peer-to-Peer Systems*, in "Parallel Processing Letters", 2009, To appear.

[17] Y. Dieudonné, F. Petit. *Scatter of Weak Robots*, in "Parallel Processing Letters", vol. 19, n^{o} 1, 2009, p. 175-184.

[18] J. Sopena, L. Arantes, F. Legond, P. Sens. *Building Effective Mutual Exclusion Services for Grids*, in "Journal of Supercomputing, Special Issue on "Secure, Manageable and Controllable Grid Services"", vol. 49, n^{o} 1, July 2009, p. 84-107.

International Peer-Reviewed Conference/Proceedings

[19] L. Arantes, A. Goldman, M. V. dos Santos. *Using Evolving Graphs to evaluate DTN routing protocols.*, in "ExtremeCom 2009", 2009 BR .

[20] L. Arantes, A. Goldman, P. Sens. *Towards a distributed computing model that characterizes dynamics of mobile networks*, in "Colibri: Colloque d'Informatique Brésil / INRIA", 2009, p. 151-155 BR .

[21] L. BENMOUFFOK, J.-M. BUSCA, J. M. MARQUÈS, M. SHAPIRO, P. SUTRA, G. TSOUKALAS. *Telex: A Semantic Platform for Cooperative Application Development*, in "Conf. Française sur les Systèmes d'Exploitation (CFSE), Toulouse, France", September 2009 GR ES .

[22] S. BERNARD, S. DEVISMES, M. G. POTOP-BUTUCARU, K. PARROUX, S. TIXEUIL. *Optimal probabilistic self-stabilizing vertex coloring in unidirectional anonymous networks*, in "ICDCN", 2010, to appear.

[23] S. BERNARD, S. DEVISMES, M. G. POTOP-BUTUCARU, S. TIXEUIL. *Optimal deterministic self-stabilizing vertex coloring in unidirectional anonymous networks*, in "IPDPS", 2009, p. 1-8.

[24] L. BLIN, M. G. POTOP-BUTUCARU, S. ROVEDAKIS. *A Superstabilizing log()-Approximation Algorithm for Dynamic Steiner Trees*, in "SSS", 2009, p. 133-148.

[25] L. BLIN, M. G. POTOP-BUTUCARU, S. ROVEDAKIS. *Self-stabilizing minimum-degree spanning tree within one from the optimal degree*, in "IPDPS", 2009, p. 1-11.

[26] L. BLIN, M. G. POTOP-BUTUCARU, S. ROVEDAKIS, S. TIXEUIL. *A New Self-stabilizing Minimum Spanning Tree Construction with Loop-Free Property*, in "DISC", 2009, p. 407-422.

[27] M. BOUILLAGUET, L. ARANTES, P. SENS. *A Timer-Free Fault Tolerant K-Mutual Exclusion Algorithm*, in "Dependable Computing, Latin-American Symposium on", vol. 0, 2009, p. 41-48.

[28] Z. BOUZID, M. G. POTOP-BUTUCARU, S. TIXEUIL. *Byzantine Convergence in Robot Networks: The Price of Asynchrony*, in "OPODIS", 2009, p. 54-70.

[29] Z. BOUZID, M. G. POTOP-BUTUCARU, S. TIXEUIL. *Byzantine-Resilient Convergence in Oblivious Robot Networks*, in "ICDCN", 2009, p. 275-280.

[30] Z. BOUZID, M. G. POTOP-BUTUCARU, S. TIXEUIL. *Optimal Byzantine Resilient Convergence in Asynchronous Robots Networks*, in "SSS, Lyon, France", F. PETIT, R. GUERRAOUI (editors), Lecture Notes in Computer Science, vol. 5873, Springer, 2009, p. 165-179 CH .

[31] F. CARRIER, S. DEVISMES, F. PETIT, Y. RIVIERRE. *Space-Optimal Deterministic Rendezvous*, in "Second International Workshop on Reliability, Availability, and Security (WRAS 2009), Hiroshima, Japan", 2009, to appear.

[32] S. DEVISMES, C. DELPORTE-GALLET, H. FAUCONNIER, F. PETIT, S. TOUEG. *Quand le consensus est plus simple que la diffusion fiable*, in "11 ièmes rencontres francophones sur les aspects algorithmiques des télécommunications (Algotel 2009), Carry-Le-Rouet, France", 2009, p. 101-104 US .

[33] S. DEVISMES, F. PETIT, S. TIXEUIL. *Exploration Optimale Probabiliste d'un Anneau par des Robots Asynchrones et Amnésiques*, in "11 ièmes rencontres francophones sur les aspects algorithmiques des télécommunications (Algotel 2009), Carry-Le-Rouet, France", 2009, p. 109-112.

[34] S. DEVISMES, F. PETIT, S. TIXEUIL. *Optimal Probabilistic Ring Exploration by Semi-Synchronous Oblivious Robots*, in "16th International Colloquium on Structural Information and Communication Complexity (SIROCCO 2009), Piran, Slovenia", Lecture Notes in Computer Science, Springer, 2009, p. 203-217.

[35] Y. DIEUDONNÉ, S. DOLEV, F. PETIT, M. SEGAL. *Brief Announcement: Deaf, Dumb, and Chatting Robots*, in "28th Annual ACM Symposium on Principles of Distributed Computing (PODC 2009), Calgary, Canada", 2009, p. 308-309 IL .

[36] Y. DIEUDONNÉ, S. DOLEV, F. PETIT, M. SEGAL. *Deaf, Dumb, and Chatting Robots: Enabling Distributed Computation and Fault-Tolerance Among Stigmergic Robots*, in "Thirteenth International Conference On Principle Of DIstributed Systems (OPODIS 2009), Nîmes, France", Lecture Notes in Computer Science, Springer, 2009, to appear IL .

[37] Y. DIEUDONNÉ, F. PETIT. *Self-stabilizing Deterministic Gathering*, in "5th International Workshop on Algorithmic Aspects of Wireless Sensor Networks (Algosensors 2009), Rhodes, Greece", Lecture Notes in Computer Science, Springer, vol. 5804, 2009, p. 230-241.

[38] Y. DIEUDONNÉ, F. PETIT. *Squaring the Circle with Weak Mobile Robots*, in "11 ièmes rencontres francophones sur les aspects algorithmiques des télécommunications (Algotel 2009), Carry-Le-Rouet, France", 2009, p. 105-108.

[39] S. DUBOIS, M. G. POTOP-BUTUCARU, S. TIXEUIL. *Brief Announcement: Dynamic FTSS in Asynchronous Systems: The Case of Unison*, in "DISC", 2009, p. 291-293.

[40] N. GEOFFRAY, G. THOMAS, G. MULLER, P. PARREND, S. FRÉNOT, B. FOLLIOT. *I-JVM: a Java Virtual Machine for Component Isolation in OSGi*, in "Proceedings of the 39th International Conference on Dependable Systems and Networks (DSN 2009), Estoril, Portugal", IEEE Computer Society, June 2009, p. 544-553.

[41] N. GEOFFRAY, G. THOMAS, G. MULLER, P. PARREND, S. FRÉNOT, B. FOLLIOT. *I-JVM: une machine virtuelle Java pour l'isolation de composants dans OSGi*, in "Actes de la 7éme Conférence Française sur les Systèmes d'Exploitation (CFSE'07), Chapitre français de l'ACM-SIGOPS, GDR ARP, Toulouse, France", Sept. 2009, p. 1-12.

[42] F. HERMENIER, X. LORCA, J.-M. MENAUD, G. MULLER, J. LAWALL. *Entropy: a Consolidation Manager for Clusters*, in "the 2009 International Conference on Virtual Execution Environments (VEE'09)", March 2009, To Appear DK .

[43] S. LEGTCHENKO, S. MONNET, P. SENS, G. MULLER. *Churn-Resilient Replication Strategy for Peer-to-Peer Distributed Hash-Tables*, in "The 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2009), Lyon, Fr", Lecture Notes in Computer Science, vol. 5873, Springer Verlag, November 2009, p. 485–499.

[44] M. LETIA, N. PREGUIÇA, M. SHAPIRO. *CRDTs: Consistency without concurrency control*, in "SOSP W. on Large Scale Distributed Systems and Middleware (LADIS), Big Sky, MT, USA", ACM SIG on Operating Systems (SIGOPS), October 2009 PT .

[45] M. MAKPANGOU. *P2P based Hosting System for Scalable Replicated Databases*, in "EDBT09 International Workshop on Data Management in Peer-to-peer Systems (DAMAP), Saint Petersburg, Russia", ACM, 2009.

[46] N. PREGUIÇA, J. M. MARQUÈS, M. SHAPIRO, M. LETIA. *A commutative replicated data type for cooperative editing*, in "Int. Conf. on Distributed Comp. Sys. (ICDCS), Montréal, Canada", June 2009, p. 395–403 PT ES .

[47] T. PREUD'HOMME, G. THOMAS, B. FOLLIOT. *GCKernel : Composition of garbage collectors*, in "The EuroSys 2009 Doctoral Workshop, Nuremberg, Germany", March. 2009, p. 1 - 2.

[48] N. SCHIPER, P. SUTRA, F. PEDONE. *Genuine versus Non-Genuine Atomic Multicast Protocols for Wide Area Networks : an Empirical Study*, in "The 28th IEEE Symposium on Reliable Distributed Systems (SRDS 2009)", 2009 CH .

National Peer-Reviewed Conference/Proceedings

[49] S. LEGTCHENKO. *Churn-Resilient Replication Strategy for Peer-to-Peer Distributed Hash-Tables*, in "The 7th Conférence Française en Systèmes d'Exploitation CFSE (CFSE 2009), Toulouse, Fr", September 2009.

[50] C. MÉHAT, O. MARIN, F. PESCHANSKI. *Intégration des fautes dans un modèle de programmation pour réseaux mobiles*, in "MajecSTIC'09", 22 November 2009, http://majecstic2009.univ-avignon.fr/Actes_MajecSTIC_RJCP/MajecSTIC/articles/1282.pdf.

Scientific Books (or Scientific Book chapters)

[51] E. CARON, F. DESPREZ, F. PETIT, C. TEDESCHI. *Peer-to-Peer Service Discovery for Grid Computing*, in "Handbook of Research on P2P and Grid Systems for Service-Oriented Computing: Models, Methodologies and Applications", N. ANTONOPOULOS, G. EXARCHAKOS, M. LI, A. LIOTTA (editors), IGI Global, Information Science Publishing, 2009, Released: December 2009. ISBN-13: 978-1615206865..

[52] B. FOLLIOT, G. THOMAS. *Virtualisation logicielle : de la machine réelle à la machine virtuelle abstraite*, in "Techniques de l'Ingénieur", Hermès, 2009, p. 1-15.

[53] M. SHAPIRO, B. KEMME. *Eventual Consistency*, in "Encyclopedia of Database Systems", M. T. ÖZSU, L. LIU (editors), Springer-Verlag GmbH, October 2009 CA .

[54] M. SHAPIRO. *Optimistic Replication and Resolution*, in "Encyclopedia of Database Systems", M. T. ÖZSU, L. LIU (editors), Springer-Verlag GmbH, October 2009.

[55] J. SOPENA, L. ARANTES, F. LEGOND, P. SENS. *Synchronization protocols for sharing resources in grid environments*, in "Fundamental of Grid computing", Chapman and Hall, 2009.

Books or Proceedings Editing

[56] F. PETIT, R. GUERRAOUI (editors). *Proceedings of the 11th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2009)*, Lecture Notes in Computer Science, vol. 5873, Springer, Lyon, France, 2009 CH .

Research Reports

[57] L. ARANTES, M. G. POTOP-BUTUCARU, P. SENS, M. VALERO. *Efficient filtering for massively distributed video games*, n^{o} RR-7008, INRIA, 2009, http://hal.inria.fr/inria-00408209/en/, Research Report.

[58] L. ARANTES, P. SENS, G. THOMAS, D. CONAN, L. LIM. *Partition Participant Detector with Dynamic Paths in MANETs*, n^{o} RR-7002, INRIA, 2009, http://hal.inria.fr/inria-00407685/en/, Research Report.

[59] M. LETIA, N. PREGUIÇA, M. SHAPIRO. *CRDTs: Consistency without concurrency control*, n^{o} RR-6956, Institut Nat. de la Recherche en Informatique et Automatique (INRIA), Rocquencourt, France, June 2009, Rapport de recherche PT ES .