IN PARTNERSHIP WITH:
CNRS

Institut national des sciences appliquées de Rennes

Université Rennes 1

Activity Report 2011

Project-Team ASAP

As Scalable As Possible: foundations of large scale dynamic distributed systems

IN COLLABORATION WITH: Institut de recherche en informatique et systèmes aléatoires (IRISA)

RESEARCH CENTER
Rennes - Bretagne-Atlantique

THEME
Distributed Systems and Services

Table of contents

Project-Team ASAP

Keywords: Peer-to-Peer, Distributed Algorithms, Epidemic Protocols, Overlay Networks, Social Networks

1. Members

Research Scientists
Anne-Marie Kermarrec [Team Leader, Research Director, HdR]
Davide Frey [Junior Researcher, INRIA]

Faculty Members
Michel Raynal [Professor (Pr), University Rennes 1, HdR]
Marin Bertier [Assistant Professor (MdC), INSA Rennes]
Achour Mostefaoui [Associate Professor (MdC), University Rennes 1 (until October 2011), HdR]
Stéphane Weiss [ATER since September 2011]

Technical Staff
Heverson Ribeiro [Ingénieur-Expert (since February 2011)]

PhD Students
Antoine Boutet [INRIA Grant]
Kévin Huguenin [MENRT Grant (until January 2011)]
Damien Imbs [MENRT Grant]
Konstantinos Kloudas [INRIA Grant]
Alexandre Van Kempen [Cifre Technicolor Grant]
Afshin Moin [INRIA Grant]
Tyler Crain [Marie-Curie European Grant]
Julien Stainer [MESR Grant]
Eleni Kanellou [Marie-Curie European Grant (since June 2011)]
Arnaud Jegou [INRIA Grant]
Mohammad Alaggan [MENRT Grant]

Post-Doctoral Fellows
Stéphane Weiss [Post-Doc from February to August 2011]
Armando Castañeda [INRIA Grant - Post-Doc]

Visiting Scientists
François Taiani [HdR]
Rida Bazzi [June 2011]
Juan Manuel Turado [From September to December 2011]

Administrative Assistant
Cécile Bouton [INRIA]

2. Overall Objectives

2.1. General objectives

The ASAP Project-Team focuses its research on a number of aspects in the design of large-scale distributed systems. Our work, ranging from theory to implementation, aims to satisfy the requirements of large-scale distributed platforms, namely scalability, personalization, and dealing with uncertainty and malicious behaviors.

2.1.1. Scalability.

The past decade has been dominated by a major shift in *scalability* requirements of distributed systems and applications mainly due to the exponential growth of network technologies (Internet, wireless technology, sensor devices, etc.). Where distributed systems used to be composed of up to a hundred of machines, they now involve thousand to millions of computing entities scattered all over the world and dealing with a huge amount of data. In addition, participating entities are highly dynamic, volatile or mobile. Conventional distributed algorithms designed in the context of local area networks do not scale to such extreme configurations. The ASAP project aims to tackle these *scalability* issues with novel distributed protocols for large-scale dynamic environments. Such protocols should be (i) fully decentralized, (ii) self organizing, and (iii) based on local system knowledge.

2.1.2. Personalization.

The need for scalability is also reflected in the huge amounts of data generated by Web 2.0 applications. Their fundamental promise, achieving *personalization*, is limited by the enormous computing capacity they require to deliver effective services like storage, search, or recommendation. Only a few companies can afford the cost of the immense cloud platforms required to process users' personal data and even they are forced to use off-line and cluster-based algorithm that operate on quasi-static data. This is not acceptable when building, for example, a large-scale news recommendation platform that must match a multitude of user interests with a continuous stream of news.

2.1.3. Uncertainty.

Effective design of distributed systems requires protocols that are able to deal with *uncertainty*. Uncertainty used to be created by the effect of asynchrony and failures in traditional distributed systems, it is now the result of many other factors. These include process mobility, low computing capacity, network dynamics, scale, and more recently the strong dependence on personalization which characterizes user-centric Web 2.0 applications. This creates new challenges such as the need to manage large quantities of personal data in a scalable manner while guaranteeing the privacy of users.

2.1.4. Malicious behaviors and privacy.

One particularly important form of uncertainty is associated with faults and *malicious* (or arbitrary) behaviors often modeled as a generic *adversary*. Protecting a distributed system partially under the control of an adversary is a multifaceted problem. On the one hand, protocols must tolerate the presence of participants that may inject spurious information, send multiple information to processes, because of a bug, an external attack, or even an unscrupulous person with administrative access (*Byzantine* behaviors). On the other hand, they must also be able to preserve *privacy* by hiding confidential data from unauthorized participants or from external observers.

2.2. Structure of the team

Our ambitious goal is to provide the algorithmic foundations of large-scale dynamic distributed systems, ranging from abstractions to real deployment. This is reflected in two major research themes: *Distributed computing models and abstractions*, and *User-centric distributed systems and applications*.

2.2.1. Models and abstractions for large-scale distributed computing

A very relevant challenge (maybe a Holy Grail) lies in the definition of a computation model appropriate to dynamic systems. This is a fundamental question. As an example there are a lot of peer-to-peer protocols but none of them is formally defined with respect to an underlying computing model. Similarly to the work of Lamport on "static" systems, a model has to be defined for dynamic systems. This theoretical research is a necessary condition if one wants to understand the behavior of these systems. As the aim of a theory is to codify knowledge in order it can be transmitted, the definition of a realistic model for dynamic systems is inescapable whatever the aim we have in mind, be it teaching, research or engineering.

2.2.1.1. Distributed computability

Among the fundamental theoretical results of distributed computing, there is a list of problems (e.g., consensus or non-blocking atomic commit) that have been proved to have no deterministic solution in asynchronous distributed computing systems prone to failures. In order such a problem to become solvable in an asynchronous distributed system, that system has to be enriched with an appropriate oracle (also called failure detector). We have been deeply involved in this research and designed optimal consensus algorithms suited to different kind of oracles. This line of research paves the way to rank the distributed computing problems according to the "power" of the additional oracle they required (think of "additional oracle" as "additional assumptions"). The ultimate goal would be the statement of a distributed computing hierarchy, according to the minimal assumptions needed to solve distributed computing problems (similarly to the Chomsky's hierarchy that ranks problems/languages according to the type of automaton they need to be solved).

2.2.1.2. Distributed computing abstractions

Major advances in sequential computing came from machine-independent data abstractions such as sets, records, etc., control abstractions such as while, if, etc., and modular constructs such as functions and procedures. Today, we can no longer envisage not to use these abstractions. In the "static" distributed computing field, some abstractions have been promoted and proved to be useful. Reliable broadcast, consensus, interactive consistency are some examples of such abstractions. These abstractions have well-defined specifications. There are both a lot of theoretical results on them (mainly decidability and lower bounds), and numerous implementations. There is no such equivalent for dynamic distributed systems.

2.2.2. User-centric fully-decentralized architectures

The Web is now centered around users. A variety of applications ranging from social networks to recommendation systems have changed the way users interact with information. Websites that used to resemble read-only data repositories have turned into fully read-write platforms in which users play a major role. Important news are continuously debated on Twitter. Facebook has become a primary means for political propaganda and protest. This is causing the amount of information available on the Web to grow by the second.

Existing paradigms for information retrieval have a hard time keeping up. Only a small portion of the web can effectively be indexed by search engines, and a lot of this information is only relevant to relatively small communities. This is particularly true when dealing with short-lived information such as news. Real-time indexing is almost impossible and even large companies resort to clustering users in an effort to provide seemingly personalized services, but without being able to harness the entire wealth of available information.

2.2.2.1. Peer-to-peer meets data mining.

To tackle the challenges posed by the Web 2.0, we are combining our expertise on large scale peer-to-peer overlays with techniques from the data-mining community. Historically, ASAP has devoted significant efforts to the design of peer-to-peer systems and overlays that directly take into account application characteristics. Within the context of Anne-Marie Kermarrec's GOSSPLE ERC grant, this task has evolved into the design of overlays and systems that gather not only network devices but also users and application objects. These become themselves peers (although obviously hosted on a physical computing entity), and their data directly influences the overlay links through similarity metrics or classification techniques.

2.2.2.2. Focus on real-world applications.

Beyond the definition of models and peer-to-peer architecture, our ambitious goal is to target real-world applications. This requires focus on technical aspects such as personalization and privacy as well as significant engineering efforts to make applications usable in a real environment. This has been reflected in several activities and of the team and has resulted, for example, in software prototypes and platforms to facilitate the design of distributed applications. The goal within this context is the involvement of users. Evaluating technologies for the social web is only possible in the presence of the Web's social components: users themselves.

2.3. Highlights

1. **A.-M. Kermarrec** received the Monpetit Award from the French Academy of Science in 2011.
2. **D. Imbs** received the best student paper award with [38], see below.

BEST PAPER AWARD :
[38] **Proc. 13th Int'l Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS'11)**. D. IMBS, M. RAYNAL.

3. Scientific Foundations

3.1. Distributed Computing

Distributed computing was born in the late seventies when people started taking into account the intrinsic characteristics of physically distributed systems. The field then emerged as a specialized research area distinct from networks, operating systems and parallelism. Its birth certificate is usually considered as the publication in 1978 of Lamport's most celebrated paper "*Time, clocks and the ordering of events in a distributed system*" [60] (that paper was awarded the Dijkstra Prize in 2000). Since then, several high-level journals and (mainly ACM and IEEE) conferences have been devoted to distributed computing. The distributed systems area has continuously been evolving, following the progresses of all the above-mentioned areas such as networks, computing architecture, operating systems.

The last decade has witnessed significant changes in the area of distributed computing. This has been acknowledged by the creation of several conferences such as NSDI and IEEE P2P. The NSDI conference is an attempt to reassemble the networking and system communities while the IEEE P2P conference was created to be a forum specialized in peer-to-peer systems. At the same time, the EuroSys conference originated as an initiative of the European Chapter of the ACM SIGOPS to gather the system community in Europe.

3.2. Theory of distributed systems

Finding models for distributed computations prone to asynchrony and failures has received a lot of attention. A lot of research in this domain focuses on what can be computed in such models, and, when a problem can be solved, what are its best solutions in terms of relevant cost criteria. An important part of that research is focused on distributed computability: what can be computed when failure detectors are combined with conditions on process input values for example. Another part is devoted to model equivalence. What can be computed with a given class of failure detectors? Which synchronization primitives is a given failure class equivalent to?). Those are among the main topics addressed in the leading distributed computing community. A second fundamental issue related to distributed models, is the definition of appropriate models suited to dynamic systems. Up to now, the researchers in that area consider that nodes can enter and leave the system, but do not provide a simple characterization, based on properties of computation instead of description of possible behaviors [61], [55], [56]. This shows that finding dynamics distributed computing models is today a "Holy Grail", whose discovery would allow a better understanding of the essential nature of dynamics systems.

3.3. Peer-to-peer overlay networks

A standard distributed system today is related to thousand or even millions of computing entities scattered all over the world and dealing with a huge amount of data. This major shift in scalability requirements has lead to the emergence of novel computing paradigms. In particular, the peer-to-peer communication paradigm imposed itself as the prevalent model to cope with the requirements of large scale distributed systems. Peer-to-peer systems rely on a symmetric communication model where peers are potentially both client and servers. They are fully decentralized, thus avoiding the bottleneck imposed by the presence of servers in traditional systems. They are highly resilient to peers arrivals and departures. Finally, individual peer behavior is based on a local knowledge of the system and yet the system converges toward global properties.

A peer-to-peer overlay network logically connect peers on top of IP. Two main classes of such overlays dominate, structured and unstructured. The differences relate to the choice of the neighbors in the overlay, and the presence of an underlying naming structure. Overlay networks represent the main approach to build large-scale distributed systems that we retained. An overlay network forms a logical structure connecting participating entities on top of the physical network, be it IP or a wireless network. Such an overlay might form a structured overlay network [62], [63], [64] following a specific topology or an unstructured network [59], [65] where participating entities are connected in a random or pseudo-random fashion. In between, lie weakly structured peer-to-peer overlays where nodes are linked depending on a proximity measure providing more flexibility than structured overlays and better performance than fully unstructured ones. Proximity-aware overlays connect participating entities so that they are connected to close neighbors according to a given proximity metric reflecting some degree of affinity (computation, interest, etc.) between peers. We extensively use this approach to provide algorithmic foundations of large-scale dynamic systems.

3.4. Epidemic protocols

Epidemic algorithms, also called gossip-based algorithms [58], [57], are consistently used in our research. In the context of distributed systems, epidemic protocols are mainly used to create overlay networks and to ensure a reliable information dissemination in a large-scale distributed system. The principle underlying the technique, in analogy with the spread of a rumor among humans via gossiping, is that participating entities continuously exchange information about the system in order to spread it gradually and reliably. Epidemic algorithms have proved efficient to build and maintain large-scale distributed systems in the context of many applications such as broadcasting [57], monitoring, resource management, search, and more generally in building unstructured peer-to-peer networks.

3.5. Malicious process behaviors

When assuming that processes fail by simply crashing, bounds on resiliency (maximum number of processes that may crash), number of exchanged messages, number of communication steps, etc. either in synchronous and augmented asynchronous systems (recall that in purely asynchronous systems some problems are impossible to solve) are known. If processes can exhibit malicious behaviors, these bounds are seldom the same. Sometimes, it is even necessary to change the specification of the problem. For example, the consensus problem for correct processes does not make sense if some processes can exhibit a Byzantine behavior and thus propose arbitrary value. In this case, the validity property of consensus, which is normally "a decided value is a proposed value", must be changed to "if all correct processes propose the same value then only this value can be decided". Moreover, the resilience bound of less than half of faulty processes is at least lowered to "less than a third of Byzantine processes". These are some of the aspects that underlie our studies in the context of the classical model of distributed systems, in peer-to-peer systems and in sensor networks.

3.6. Online Social Networks

Social Networks have rapidly become a fundamental component of today's distributed applications. Web 2.0 applications have dramatically changed the way users interact with the Internet and with each other. The number of users of websites like Flickr, Delicious, Facebook, or MySpace is constantly growing, leading to significant technical challenges. On the one hand, these websites are called to handle enormous amounts of data. On the other hand, news continue to report the emergence of privacy threats to the personal data of social-network users. Our research aims to exploit our expertise in distributed systems to lead to a new generation of scalable, privacy-preserving, social applications.

4. Application Domains

4.1. Application Domains

The results of the research targeted in ASAP span a wide range of applications. Below are a few examples.

- Personalized Web Search.
- Recommendation.
- Social Networks.
- Notification Systems.
- Distributed Storage.
- Video Streaming.

5. Software

5.1. WhatsUp: A Distributed News Recommender

Participants: Antoine Boutet, Davide Frey, Arnaud Jegou, Anne-Marie Kermarrec.

Contact:	Antoine Boutet
Licence:	Open Source
Presentation:	A Distributed News Recommender
Status:	Beta version

This work has lead to the development of WhatsUp, a distributed recommendation system aimed to distribute instant news in a large scale dynamic system. WhatsUp has two parts, an embedded application server in order to exchange with others peers in the system and a fully dynamic web interface for displaying news and collecting opinions about what the user reads. Underlying this web-based application lies Beep, a biased epidemic dissemination protocol that delivers news to interested users in a fast manner while limiting spam. Beep is parametrized on the fly to manage the orientation and the amplification of news dissemination. Every user forwards the news of interest to a randomly selected set of users with a preference towards those that have similar interests (orientation). The notion of interest does not rely on any explicit social network or subscription scheme, but rather on an implicit and dynamic overlay capturing the commonalities between users with respect to they are interested in. The size of the set of users to which a news is forwarded depends on the interest of the news (amplification). A centralized version of WhatsUp is already up and running and the decentralized one is still in beta version.

5.2. GossipLib: effective development of gossip-based applications

Participants: Davide Frey, Heverson Ribeiro, Anne-Marie Kermarrec.

Contact:	Davide Frey
Licence:	Open Source
Presentation:	Library for Gossip protocols
Status:	released version 0.7alpha

GossipLib is a library consisting of a set of JAVA classes aimed to facilitate the development of gossip-based application in a large-scale setting. It provides developers with a set of support classes that constitute a solid starting point for building any gossip-based application. GossipLib is designed to facilitate code reuse and testing of distributed application and as thus also provides the implementation of a number of standard gossip protocols that may be used out of the box or extended to build more complex protocols and applications. These include for example the peer-sampling protocols for overlay management.

GossipLib also provides facility for the configuration and deployment of applications as final-product but also as research prototype in environments like PlanetLab, clusters, network emulators, and even as event-based simulation. The code developed with GossipLib can be run both as a real application and in simulation simply by changing one line in a configuration file.

5.3. YALPS

Participants: Davide Frey, Heverson Ribeiro, Anne-Marie Kermarrec.

Contact:	Davide Frey
Licence:	Open Source
Presentation:	Library for Gossip protocols
Status:	released version 0.3alpha

YALPS is an open-source Java library designed to facilitate the development, deployment, and testing of distributed applications. Applications written using YALPS can be run both in simulation and in real-world mode without changing a line of code or even recompiling the sources. A simple change in a configuration file will load the application in the proper environment. A number of feature make YALPS useful both for the design and evaluation of research prototypes and for the development of applications to be released to the public. Specifically, YALPS makes it possible to run the same application as a simulation or in a real deployment without a single change in the code. Applications communicate by means of application-defined messages which are then routed either through UDP/TCP or through YALPS's simulation infrastructure. In both cases, YALPS's communication layer offers features for testing and evaluating distributed protocols and applications. Communication channels can be tuned to incorporate message losses or to constrain their outgoing bandwidth. Finally, YALPS includes facilities to support operation in the presence of NATs and firewalls using relaying and NAT-traversal techniques.

The work has been done in collaboration with Maxime Monod (EPFL).

5.4. HEAP: Heterogeneity-aware gossip protocol.

Participants: Davide Frey, Arnaud Jegou, Anne-Marie Kermarrec.

Contact:	Davide Frey
Licence:	Open Source
Presentation:	Java Application
Status:	release & ongoing development

This work has been done in collaboration with Vivien Quéma (CNRS Grenoble), Maxime Monod and Rachid Guerraoui (EPFL), and has lead to the development of a video streaming platform based on HEAP, *HEterogeneity-Aware gossip Protocol*. The platform is particularly suited for environment characterized by heterogeneous bandwidth capabilities such as those comprising ADSL edge nodes. HEAP is, in fact, able to dynamically leverage the most capable nodes and increase their contribution to the protocol, while decreasing by the same proportion that of less capable nodes. During the last few months, we have integrated HEAP with the ability to dynamically measure the available bandwidth of nodes, thereby making it independent of the input of the user.

6. New Results

6.1. Models and abstractions for distributed systems

This section summarizes the major results obtained by the ASAP team that relate to the foundations of distributed systems.

6.1.1. The weakest failure detector to implement a register in asynchronous systems with hybrid communication

Participants: Damien Imbs, Michel Raynal.

This work introduces an asynchronous crash-prone hybrid system model. The system is hybrid in the way the processes can communicate. On the one side, a process can send messages to any other process. On another side, the processes are partitioned into clusters and each cluster has its own read/write shared memory. In addition to the model, a main contribution of the work concerns the implementation of an atomic register in this system model. More precisely, a new failure detector (denoted $M\Sigma$) is introduced and it is shown that, when considering the information on failures needed to implement a register, this failure detector is the weakest. To that end, the work presents an $M\Sigma$-based algorithm that builds a register in the considered hybrid system model and shows that it is possible to extract $M\Sigma$ from any failure detector-based algorithm that implements a register in this model. The work also (a) shows that $M\Sigma$ is strictly weaker than Σ (which is the weakest failure detector to implement a register in a classical message-passing system) and (b) presents a necessary and sufficient condition to implement $M\Sigma$ in a hybrid communication system.

This work has been published in SSS 2011 [38].

6.1.2. The universe of symmetry breaking tasks

Participants: Damien Imbs, Michel Raynal.

Processes in a concurrent system need to coordinate using a shared memory or a message-passing subsystem in order to solve agreement tasks such as, for example, consensus or set agreement. However, coordination is often needed to "break the symmetry" of processes that are initially in the same state, for example, to get exclusive access to a shared resource, to get distinct names or to elect a leader.

This work introduces and studies the family of *generalized symmetry breaking* (GSB) tasks, that includes election, renaming and many other symmetry breaking tasks. Differently from agreement tasks, a GSB task is "inputless", in the sense that processes do not propose values; the task only specifies the symmetry breaking requirement, independently of the system's initial state (where processes differ only on their identifiers). Among various results characterizing the family of GSB tasks, it is shown that (non adaptive) perfect renaming is universal for all GSB tasks.

This work was done in collaboration with Sergio Rajsbaum from the Universidad Nacional Autonoma de Mexico and was published in SIROCCO 2011 [36].

6.1.3. Read invisibility, virtual world consistency and probabilistic permissiveness are compatible

Participants: Tyler Crain, Damien Imbs, Michel Raynal.

The aim of a Software Transactional Memory (STM) is to discharge the programmers from the management of synchronization in multiprocess programs that access concurrent objects. To that end, an STM system provides the programmer with the concept of a transaction. The job of the programmer is to design each process the application is made up of as a sequence of transactions. A transaction is a piece of code that accesses concurrent objects, but contains no explicit synchronization statement. It is the job of the underlying STM system to provide the illusion that each transaction appears as being executed atomically. Of course, for efficiency, an STM system has to allow transactions to execute concurrently. Consequently, due to the underlying STM concurrency management, a transaction commits or aborts.

This work studies the relation between two STM properties (read invisibility and permissiveness) and two consistency conditions for STM systems, namely, opacity and virtual world consistency. Both conditions ensure that any transaction (be it a committed or an aborted transaction) reads values from a consistent global state, a noteworthy property if one wants to prevent abnormal behavior from concurrent transactions that behave correctly when executed alone. A read operation issued by a transaction is invisible if it does not entail shared memory modifications. This is an important property that favors efficiency and privacy. An STM system is permissive (respectively probabilistically permissive) with respect to a consistency condition if it accepts (respectively accepts with positive probability) every history that satisfies the condition. This is a crucial property as a permissive STM system never aborts a transaction "for free". The work first shows that read invisibility, probabilistic permissiveness and opacity are incompatible, which means that there is no probabilistically permissive STM system that implements opacity while ensuring read invisibility. It then

shows that read invisibility, probabilistic permissiveness and virtual world consistency are compatible. To that end the work describes a new STM protocol called IR_VWC_P. This protocol presents additional noteworthy features: it uses only base read/write objects and locks which are used only at commit time; it satisfies the disjoint access parallelism property; and, in favorable circumstances, the cost of a read operation is $O(1)$.

This work has been published in ICA3PP 2011 [29].

6.1.4. Towards a universal construction for transaction-based multiprocess programs

Participants: Tyler Crain, Damien Imbs, Michel Raynal.

The aim of a Software Transactional Memory (STM) system is to discharge the programmer from the explicit management of synchronization issues. The programmer's job resides in the design of multiprocess programs in which processes are made up of transactions, each transaction being an atomic execution unit that accesses concurrent objects. The important point is that the programmer has to focus her/his efforts only on the parts of code which have to be atomic execution units without worrying on the way the corresponding synchronization has to be realized.

Non-trivial STM systems allow transactions to execute concurrently and rely on the notion of commit/abort of a transaction in order to solve their conflicts on the objects they access simultaneously. In some cases, the management of aborted transactions is left to the programmer. In other cases, the underlying system scheduler is appropriately modified or an underlying contention manager is used in order that each transaction be ("practically always" or with high probability) eventually committed.

This work paper proposed a deterministic STM system in which (1) every invocation of a transaction is executed exactly once and (2) the notion of commit/abort of a transaction remains unknown to the programmer. This system, which imposes restriction neither on the design of processes nor or their concurrency pattern, can be seen as a step towards the design of a deterministic universal construction to execute transaction-based multiprocess programs on top of a multiprocessor. Interestingly, the proposed construction is lock-free (in the sense that it uses no lock).

This work has been published in ICDCN 2012 [30].

6.1.5. A transaction friendly binary search tree

Participants: Tyler Crain, Michel Raynal.

Transactions, which provide optimistic synchronization by avoiding the use of blocking, greatly simplify multicore programming. In fact, the programmer has simply to encapsulate sequential operations or existing critical sections into transactions to obtain a safe concurrent program. Programmers have thus started evaluating transactional memory using data structures originally designed for pessimistic (i.e., non-optimistic) synchronization, whose prominent example is the red-black tree library developed by Oracle Labs that is part of STAMP and microbench distributions. Unfortunately, existing data structures are badly suited for optimistic synchronization as they rely on strong structural invariants, like logarithmic tree depth, to bound the step complexity of pessimistically synchronized accesses. By contrast, this complexity does not apply to optimistically synchronized accesses thus making the invariants overly conservative. More dramatically, guaranteeing such invariants tends to increase the probability of aborting and restarting the same access before it completes. We introduced a concurrent binary search tree that breaks transiently its balance structural invariants for efficiency, a property we call transaction-friendly. This new tree outperforms the existing transaction-based version of the AVL and the red-black trees. Its key novelty stems from the decoupling of update operations: they are split into one transaction that modifies the abstraction state and multiple ones that restructure its tree implementation. The resulting transaction-friendly library trades aborts for few additional access steps and, in particular, it speeds up a transaction-based travel reservation application by up to 3:5X. This work was done in collaboration with Vincent Gramoli from EPFL Lausanne, and is described in [52].

6.1.6. Relations linking failure detectors associated with k-set agreement in message-passing systems

Participants: Achour Mostefaoui, Michel Raynal, Julien Stainer.

The k-set agreement problem is a coordination problem where each process is assumed to propose a value and each process that does not crash has to decide a value such that each decided value is a proposed value and at most k different values are decided. While it can always be solved in synchronous systems, k-set agreement has no solution in asynchronous send/receive message-passing systems where up to $t \geq k$ processes may crash.

A failure detector is a distributed oracle that provides processes with additional information related to failed processes and can consequently be used to enrich the computability power of asynchronous send/receive message-passing systems. Several failure detectors have been proposed to circumvent the impossibility of k-set agreement in pure asynchronous send/receive message-passing systems. Considering three of them (namely, the generalized quorum failure detector Σ_k, the generalized loneliness failure detector $\mathcal{L}_k$ and the generalized eventual leader failure detector Ω_k) this work investigates their computability power and the relations that link them. There are three mains contributions: (a) it shows that the failure detector Ω_k and the eventual version of $\mathcal{L}_k$ have the same computational power; (b) it shows that $\mathcal{L}_k$ is realistic if and only if $k \geq n/2$; and (c) it gives an exact characterization of the difference between $\mathcal{L}_k$ (that is too strong for k-set agreement) and Σ_k (that is too weak for k-set agreement). This work was published at SSS 2011 [45].

6.1.7. The price of anonymity: optimal consensus despite asynchrony, crash and anonymity

Participant: Michel Raynal.

This work [23], done in collaboration with François Bonnet, from JAIST, Japan, addresses the consensus problem in asynchronous systems prone to process crashes, where additionally the processes are anonymous (they cannot be distinguished one from the other: they have no name and execute the same code). To circumvent the three computational adversaries (asynchrony, failures and anonymity) each process is provided with a failure detector of a class denoted ψ, that gives it an upper bound on the number of processes that are currently alive (in a non-anonymous system, the classes ψ and $\mathcal{P}$ -the class of perfect failure detectors- are equivalent).

The first part presents a simple ψ-based consensus algorithm where the processes decide in $2t+1$ asynchronous rounds (where t is an upper bound on the number of faulty processes). It then shows one of its main results, namely, $2t+1$ is a lower bound for consensus in the anonymous systems equipped with ψ. The second contribution addresses early-decision. The paper presents and proves correct an early-deciding algorithm where the processes decide in $\min(2f+2, 2t+1)$ asynchronous rounds (where f is the actual number of process failures). This leads to think that anonymity doubles the cost (wrt synchronous systems) and it is conjectured that $\min(2f+2, 2t+1)$ is the corresponding lower bound.

The work finally considers the k-set agreement problem in anonymous systems. It first shows that the previous ψ-based consensus algorithm solves the k-set agreement problem in $R_t = 2\left\lfloor \frac{t}{k} \right\rfloor + 1$ asynchronous rounds. Then, considering a family of failure detector classes $\{\psi_\ell\}_{0 \leq \ell < k}$ that generalizes the class $\psi (= \psi_0)$, the paper presents an algorithm that solves the k-set agreement in $R_{t,\ell} = 2\left\lfloor \frac{t}{k-\ell} \right\rfloor + 1$ asynchronous rounds. This last formula relates the cost ($R_{t,\ell}$), the coordination degree of the problem (k), the maximum number of failures (t) and the the strength (ℓ) of the underlying failure detector.

6.1.8. On the road to the Weakest Failure Detector for k-Set Agreement in Message-passing Systems

Participant: Michel Raynal.

In the k-set agreement problem, each process (in a set of n processes) proposes a value and has to decide a proposed value in such a way that at most k different values are decided. While this problem can easily be solved in asynchronous systems prone to t process crashes when $k > t$, it cannot be solved when $k \leq t$. Since several years, the failure detector-based approach has been investigated to circumvent this impossibility. While the weakest failure detector class to solve the k-set agreement problem in read/write shared-memory systems has recently been discovered (PODC 2009), the situation is different in message-passing systems where the weakest failure detector classes are known only for the extreme cases $k = 1$ (consensus) and $k = n - 1$ (set agreement).

This work [22], done in collaboration with François Bonnet, from JAIST, Japan, has four contributions whose aim is to help pave the way to discover the weakest failure detector class for k-set agreement in message-passing systems. These contributions are the following. (a) The first is a new failure detector class, denoted Π_k, that is such that $\Pi_1 = \Sigma \times \Omega$ (the weakest class for $k = 1$), and $\Pi_{n-1} = \mathcal{L}$ (the weakest class for $k = n - 1$). (b) The second is an investigation of the structure of Π_k that shows that Π_k is the combination of two failures detector classes Σ_k (that is new) and Ω_k (they generalize the previous "quorums" and "eventual leaders" failure detectors classes, respectively). (c) The third contribution concerns Σ_k that is shown to be necessary requirement (as far as information on failure is concerned) to solve the k-set agreement problem in message-passing systems. (d) Finally, the last contribution is a Π_{n-1}-based algorithm that solves the $(n-1)$-set agreement problem. This algorithm provides us with a new algorithmic insight on the way the $(n-1)$-set agreement problem can be solved in asynchronous message-passing systems. It is hoped that these contributions will help discover the weakest failure detector class for k-set agreement in message-passing systems.

6.1.9. A non-topological proof for the impossibility of k-set agreement.

Participant: Armando Castañeda.

This work was done in collaboration with Hagit Attiya, from Technion, Haifa, Israel. In the *k-set agreement* task each process proposes a value, and each correct process has to decide a value which was proposed, so that at most k distinct values are decided. Using topological arguments it has been proved that k-set agreement is unsolvable in the asynchronous *wait-free* read/write shared memory model, when $k < n$, the number of processes.

This work [34] focuses on a simple, non-topological impossibility proof of k-set agreement. The proof depends on two simple properties of the *immediate snapshot executions*, a subset of all possible executions, and on the well known *handshaking lemma* stating that every graph has an even number of vertices with odd degree.

The paper was presented in the 13th Int'l Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS'11) in Grenoble, France. The journal version of the paper was submitted to Theoretical Computer Science.

6.1.10. Enriching the reduction map of sub-consensus tasks

Participants: Armando Castañeda, Damien Imbs, Michel Raynal.

This work [51] was done in collaboration with Sergio Rajsbaum from the Universidad Nacional Autonoma de Mexico.

Understanding the relative computability power of tasks, in the presence of asynchrony and failures, is a central concern of distributed computing theory. In the *wait-free* case, where the system consists of n processes and any of them can fail by crashing, substantial attention has been devoted to understanding the relative power of the *subconsensus* family of tasks, which are too weak to solve consensus for two processes. The first major results showed that set agreement and renaming (except for some particular values of n) cannot be solved wait-free in read/write memory. Then it was proved that renaming is strictly weaker than set agreement (when n is odd).

This work considers a natural family of subconsensus tasks that includes set agreement, renaming and other generalized symmetry breaking (GSB) tasks. It extends previous results, and proves various new results about when there is a reduction and when not, among these tasks. Among other results, the work shows that there are incomparable subconsensus tasks.

6.1.11. Byzantine Consensus Decidability

Participants: Achour Mostefaoui, Michel Raynal.

Solving the consensus problem requires in one way or another that the underlying system satisfies synchrony assumptions. Considering a system of n processes where up to $t < n/3$ may commit Byzantine failures, we proposed in [26] a necessary and sufficient synchrony assumption to solve consensus.

Such a condition is formulated with the notions of a symmetric synchrony property and property ambiguity. A symmetric synchrony property is a set of graphs, where each graph corresponds to a set of bi-directional eventually synchronous links among correct processes. Intuitively, a property is ambiguous if it contains a graph whose connected components are such that it is impossible to distinguish a connected component that contains correct processes only from a connected component that contains faulty processes only. The paper connects then the notion of a symmetric synchrony property with the notion of eventual bi-source, and shows that the existence of a virtual $\diamond[t+1]$bi-source is a necessary and sufficient condition to solve consensus in presence of up to t Byzantine processes in systems with bi-directional links and message authentication. Finding necessary and sufficient synchrony conditions when links are timely in one direction only, or when processes cannot sign messages, still remains open (and very challenging) problems.

6.1.12. Solving k-set agreement in message-passing systems

Participants: Achour Mostefaoui, Michel Raynal, Julien Stainer.

The k-set agreement problem is a coordination problem where each process is assumed to propose a value and each process that does not crash has to decide a value such that each decided value is a proposed value and at most k different values are decided. While it can always be solved in synchronous systems, k-set agreement has no solution in asynchronous send/receive message-passing systems where up to $t \geq k$ processes may crash.

A failure detector is a distributed oracle that provides processes with additional information related to failed processes and can consequently be used to enrich the computability power of asynchronous send/receive message-passing systems. Several failure detectors have been proposed to circumvent the impossibility of k-set agreement in pure asynchronous send/receive message-passing systems. Considering three of them (namely, the generalized quorum failure detector Σ_k, the generalized loneliness failure detector $\mathcal{L}_k$ and the generalized eventual leader failure detector Ω_k), we investigated their computability power and the relations that link them in [45]. It has three mains contributions: (a) it shows that the failure detector Ω_k and the eventual version of $\mathcal{L}_k$ have the same computational power; (b) it shows that $\mathcal{L}_k$ is realistic if and only if $k \geq n/2$; and (c) it gives an exact characterization of the difference between $\mathcal{L}_k$ (that is too strong for k-set agreement) and Σ_k (that is too weak for k-set agreement).

6.1.13. Efficient Implementations of Concurrent Objects

Participants: Achour Mostefaoui, Michel Raynal.

As introduced by Taubenfeld, a contention-sensitive implementation of a concurrent object is an implementation such that the overhead introduced by locking is eliminated in the common cases, i.e., when there is no contention or when the operations accessing concurrently the object are non-interfering. In [44], we present a methodological construction of a contention-sensitive implementation of a concurrent stack. In a contention-free context a push or pop operation does not rest on a lock mechanism and needs only six accesses to the shared memory. In case of concurrency a single lock is required. Moreover, the implementation is starvation-free (any operation is eventually executed). The paper, that presents the algorithms in an incremental way, visits also a family of liveness conditions and important concurrency-related concepts such as the notion of an abortable object.

6.2. Large-scale and user-centric distributed system

This section summarizes the major results obtained by the team in 2011 in the context of large-scale distributed systems and social networks. This includes the results obtained within the GOSSPLE ERC project, which encompass two types of social networks: explicit and implicit.

Explicit networks connect users based on explicit social connections. In FACEBOOK or MYSPACE, users issue and accept friendship requests. In TWITTER, they decide that they wish to follow the tweets of specific users. In all cases, the topology of the resulting network reflects the choices of users and often consists of links that already exist between real people. Explicit networks are therefore very useful in reinforcing and exploiting existing connections but provide little support for discovering new content.

Implicit networks complement explicit ones by providing each user with a set of anonymous acquaintances that share similar interests, that visit similar websites or that have otherwise similar profiles. Different from explicit networks, implicit ones are naturally suited to support the discovery of new content. In previous work [1], we exploited this network to improve web navigation. In the following, we consider additional applications encompassing news dissemination, online transactions, and recommendation.

6.2.1. WhatsUp: P2P news recommender

Participants: Antoine Boutet, Davide Frey, Anne-Marie Kermarrec.

The main application in the context of GOSSPLE is WhatsUp, an instant news system designed for a large-scale network with no central authority. WhatsUp builds an implicit social network based on the opinions users express about the news items they receive (like-dislike). This is achieved through an obfuscation mechanism that does not require users to ever reveal their exact profiles. WhatsUp disseminates news items through a novel heterogeneous gossip protocol that biases the choice of its targets towards those with similar interests and amplifies dissemination based on the level of interest in every news item. WhatsUp outperforms various alternatives in terms of accurate and complete delivery of relevant news items while preserving the fundamental advantages of standard gossip: namely simplicity of deployment and robustness. This work has been carried out in collaboration with Rachid Guerraoui from EPFL and was demonstrated during the different local events.

6.2.2. Personalized top-k processing

Participant: Anne-Marie Kermarrec.

Another way to improve the experience of users on the web is to personalize top-k queries. In collaboration with Xiao Bai and Vincent Leroy from Yahoo! Research in Barcelona and Rachid Guerraoui from EPFL Lausanne we, therefore, introduced P4Q, a fully decentralized gossip-based protocol to personalize query processing in social tagging systems. P4Q dynamically associates each user with social acquaintances sharing similar tagging behaviors. Queries are gossiped among such acquaintances, computed on the fly in a collaborative, yet partitioned manner, and results are iteratively refined and returned to the querier. Analytical and experimental evaluations convey the scalability of P4Q for top-k query processing, as well its inherent ability to cope with users updating profiles and departing. The work appeared in the ACM transactions of database systems [12].

6.2.3. Social Market

Participants: Davide Frey, Arnaud Jegou, Anne-Marie Kermarrec.

The ability to identify people that share one's own interests is one of the most interesting promises of the Web 2.0 driving user-centric applications such as recommendation systems or collaborative marketplaces. To be truly useful, however, information about other users also needs to be associated with some notion of trust. Consider a user wishing to sell a concert ticket. Not only must she find someone who is interested in the concert, but she must also make sure she can trust this person to pay for it. Social Market (SM) solve this problem by allowing users to identify and build connections to other users that can provide interesting goods or information and that are also reachable through a trusted path on a explicit social network like Facebook. This convergence of implicit and explicit networks yields TAPS, a novel gossip protocol that can be applied in applications devoted to commercial transactions, or to add robustness to standard gossip applications like dissemination or recommendation systems.

This work has been published at SSS 2011 [33], and an extended version bringing better performances and strong privacy guaranties have recently been submitted for publication.

6.2.4. Member classification and party characteristics in Twitter

Participant: Antoine Boutet.

In modern politics, parties and individual candidates must have an online presence and usually have dedicated social media coordinators. In this context, real time member classification and party characterization, taking into account the dynamic nature of social media, are essential to highlight the main differences between parties and to monitor their activities, influences, structures, contents and mood. This work [53] was been done in collaboration with E. Yoneki from Computer Lab, Cambridge, UK.

6.2.5. Graph Drawing and Visual Recommendations

Participants: Anne-Marie Kermarrec, Afshin Moin.

An important aspect of social network is their graph structure. In a collaboration with Vincent Leroy (Yahoo! Research) and Gilles Tredan (TU Berlin) [41], we started from this structure to propose a decentralized gossip-based algorithm called SoCS (Social Coordinate Systems). SoCS achieves efficient distributed social graph embedding using a force-based graph embedding technique to extract communities from a graph. SoCS (i) scales to large dynamic graph, aggregating the computing power of individual nodes and, (ii) avoids a central entity controlling users sensitive data such as relations and preferences. We evaluated SoCS using two different force-based models and compare them in the context of a generated Kleinberg small-world topology. More specifically, we showed that the SoCS graph embedding enables to clearly distinguish between short and long-range links. We also evaluate SoCS against a real DBLP data set, showing that removed links are correctly predicted.

Graph structures are also at the basis of our work on energy/force-based models for graph visualization. We applied visualization both to social network and in the context of recommendation systems. In particular we are working on an SVD-like algorithm for drawing precise 2-dimensional visual recommendations based on Principal Component Analysis (PCA) and Curvilinear Component Analysis (CCA).

6.2.6. Private Similarity Computation in Distributed Systems: from Cryptography to Differential Privacy

Participants: Mohammad Alaggan, Anne-Marie Kermarrec.

The use of personal data in the context of social networks raises important concerns about privacy. In a collaboration [24] with Sébastien Gambs from the CIDre team, we addressed the problem of computing the similarity between two users (a key operation in an implicit social network [1]) while preserving their privacy in a fully decentralized system and for the passive adversary model. First, we introduced a two-party protocol for privately computing a threshold version of the similarity and applied it to well-known similarity measures such as the scalar product and the cosine similarity. The output of this protocol is only one bit of information telling whether or not two users are similar beyond a predetermined threshold. Afterwards, we explored the computation of the exact and threshold similarity within the context of differential privacy. Differential privacy is a recent notion developed within the field of private data analysis guaranteeing that an adversary that observes the output of the differentially private mechanism, will only gain a negligible advantage (up to a privacy parameter) from the presence (or absence) of a particular item in the profile of a user. This provides a strong privacy guarantee that holds independently of the auxiliary knowledge that the adversary might have. More specifically, we designed several differentially private variants of the exact and threshold protocols that rely on the addition of random noise tailored to the sensitivity of the considered similarity measure. We also analyzed their complexity as well as their impact on the utility of the resulting similarity measure.

6.2.7. Constellation: Programming decentralized social networks

Participants: François Taiani, Anne-Marie Kermarrec.

As they continue to grow, social and collaborative applications (e.g. twitter, Facebook, digg) are increasingly calling for disruptive distributed solutions than can cater for the millions of users these applications serve daily, in hundreds of countries, over a wide variety of devices. To address these challenges, fully decentralized versions of social and collaborative applications are progressively emerging that seek to provide naturally scalable solutions to deliver their services. Gossip protocols in particular appear as a natural solution to implement these decentralized versions, as they intrinsically tend to be highly resilient, efficient, and scalable.

Social applications based on gossip have however been limited so far to relatively homogeneous systems: They typically rely on one similarity measure to self-organize large amount of distributed users in implicit communities, and thus offer powerful means to search, mine, and serve personalized data in a distributed manner.

We posit in this work [54] that we now need to move to more complex gossip-based social applications that can cater for different types of data and similarity, organized in multiple levels of abstraction. Exploring, designing, and evaluating such novel approaches is unfortunately time-consuming and error-prone. To help in this task, we have started to design a new programming language, Constellation, that seeks to simplify the realization and experimentation with social gossip-based applications. Constellation is based on two central observations: (i) future decentralized social applications will need to handle heterogeneous forms of data and self-organization, and (ii) to offer more powerful services, these applications will need to move beyond physical nodes to encompass richer data structures organized in virtualized levels of abstractions.

6.2.8. Leveraging content interconnections for efficient data storage.

Participants: Anne-Marie Kermarrec, Konstantinos Kloudas, François Taiani.

Traffic generated by User Generated Content (UGC) sharing sites, such as YOUTUBE, accounts for a substantial fraction of today's global Internet load. This success has however brought a number of key technical challenges, crucial for system sustainability and user experience. One of them is the need to place content close to consumers, so that user perceived latency is reduced and bandwidth utilization is minimized. In a joint work with Kevin Huguenin, we try to tackle this problem by leveraging the fact that content hosted by these sites is interconnected, forming a content graph that as shown by former works, has an important impact on a file's view pattern. In our work titled *"Recommended nearby in UGC delivery networks: leveraging geographical and content locality"*, we focused on YOUTUBE and we studied how two types of locality previously analyzed in isolation in UGC systems, namely *content locality* (a.k.a graph locality, induced by the related video feature) and *geographic locality*, are in fact correlated. Leveraging the above finding, we proposed a novel algorithm for replica placement that tries to predict where *future* views for a video will come from based on the video's related videos and places its replicas accordingly. This work has been submitted for publication.

6.2.9. Transparent Componentization: High-level (Re)configurable Programming for Evolving Distributed Systems

Participants: François Taiani, Marin Bertier, Anne-Marie Kermarrec.

This work was done in collaboration Component frameworks and high-level distributed languages have been widely used to develop distributed systems, and provide complementary advantages: Whereas component frameworks foster composability, reusability, and (re)configurability; distributed languages focus on behavior, simplicity and programmability. We argue that both types of approach should be brought together to help develop complex adaptive systems, and we propose an approach to combines both technologies without compromising on any of their benefits. Our approach, termed Transparent Componentization [43], automatically maps a high-level distributed specification onto a underlying component framework. It thus allows developers to focus on the programmatic description of a distributed system's behavior, while retaining the benefits of a component architecture. As a proof of concept, we present WhispersKit, a programming environment for gossip-based distributed systems. Our evaluation shows that WhispersKit successfully retains the simplicity and understandability of high-level distributed language while providing efficient and transparent reconfigurability thanks to its component underpinnings.

6.2.10. Efficient peer-to-peer backup services through buffering at the edge

Participants: Anne-Marie Kermarrec, Alexandre Van Kempen.

The availability of end devices of peer-to-peer storage and backup systems has been shown critical for usability and for system reliability in practice. This has led to the adoption of hybrid architectures composed of both peers and servers. Such architectures mask the instability of peers thus approaching the performances of client-server systems while providing scalability at a low cost. In this work [31] - done in collaboration with Erwan Le Merrer, Serge Defrance, Nicolas Le Scouarnec and Gilles Straub from Technicolor, Rennes, France - we advocate the replacement of such servers by a cloud of residential gateways, as they are already present in users' homes, thus pushing the required stable components at the edge of the network. In our gateway-assisted system, gateways act as buffers between peers, compensating for their intrinsic instability. This enables to offload backup tasks quickly from the user's machine to the gateway, while significantly lowering the retrieval time of backed up data. We evaluate our proposal using real world traces including existing traces from Skype and Jabber as well as a trace of residential gateways for availability, and a residential broadband trace for bandwidth. Results show that the time required to backup data in the network is comparable to a server-assisted approach, while substantially improving the time to restore data, which drops from a few days to a few hours. As gateways are becoming increasingly powerful in order to enable new services, we expect such a proposal to be leveraged on a short term basis.

6.2.11. Commutative Replicated Data Type for Semantic Stores

Participant: Stéphane Weiss.

This work has been done in collaboration with Khaled Aslan (Université de Nantes - Lina), Pascal Molli (Université de Nantes - Lina) and Hala Skaf-Molli (Université de Nantes - Lina).

Web 2.0 tools are currently evolving to embrace semantic web technologies. Blogs, CMS, Wikis, social networks and real-time notifications, integrate ways to provide semantic annotations and therefore contribute to the linked data and more generally to the semantic web vision. This evolution generates a lot of semantic datasets of different qualities, different trust levels and partially replicated. This raises the issue of managing the consistency among these replicas. This issue is challenging because semantic data-spaces can be very large, they can be managed by autonomous participants and the number of replicas is unknown. A new class of algorithms called Commutative Replicated Data Type are emerging for ensuring eventual consistency of highly dynamic content on P2P networks. We define C-Set [25] a CRDT specifically designed to be integrated in Triple-stores. C-Set allows efficient P2P synchronization of an arbitrary number of autonomous semantic stores.

6.2.12. Building large scale platform for chemical program

Participants: Marin Bertier, Achour Mostefaoui.

This work [28] was done in collaboration with the Myriads project team.

Chemical programming is a promising paradigm to design autonomic systems. Within such a paradigm, computations can be seen as chemical reactions controlled by a set of chemical rules. In other words, data are molecules of a chemical solutions, reacting together to produce new data. Reactions take place in an implicitly parallel, and autonomic fashion.

Our objective was to design a distributed chemical platform bringing such concepts. This platform should be adapted to large scale distributed system to benefit at his best the inherent distribution of chemical program.

7. Contracts and Grants with Industry

7.1. Technicolor

Participants: Anne-Marie Kermarrec, Alexandre Van Kempen.

Since 2010, we have had a contract with Technicolor for collaboration on peer-assisted approaches for reliable storage. In this context, Anne-Marie Kermarrec has been the PhD adviser of Alexandre van Kempen since 2010.

8. Partnerships and Cooperations

8.1. National Initiatives

8.1.1. LABEX CominLabs

Participants: Anne-Marie Kermarrec, Davide Frey, Stéphane Weiss.

ASAP participates in the CominLabs initiative sponsored by the "Laboratoires d'Excellence" program. The initiative federates the best teams from Bretagne and Nantes regions in the broad area of telecommunications, from electronic devices to wide area distributed applications "over the top". These include, among the others, the INRIA teams: ACES, ALF, ASAP, CELTIQUE, CIDRE, DISTRIBCOM, MYRIADS, TEMICS, TEXMEX, and Visages. The scope of CominLabs covers research, education, and innovation. While being hosted by academic institutions, the CominLabs build on a strong industrial ecosystem made of large companies and competitive SMEs.

8.1.2. ANR ARPÈGE project Streams

Participants: Achour Mostefaoui, Marin Bertier, Michel Raynal, Stéphane Weiss.

The Streams project started in November 2010. Beside the ASAP group, it includes Teams from INRIA Nancy and PARIS. Its aim it to design a real-time collaborative platform based on a peer-to-peer network. For this it is necessary to design a support architecture that offers guarantees on the propagation, security and consistency of the operations and the updates proposed by the different collaborating sites.

8.1.3. ANR VERSO project Shaman

Participants: Marin Bertier, Achour Mostefaoui, Anne-Marie Kermarrec, Michel Raynal.

The Shaman project started in 2009, gathering several members of the team working on distributed systems and distributed algorithms. The aim of this project is to propose new theoretical models for distributed algorithm inspired from real platform characteristics. From these models, we elaborate new algorithms and try to evaluate their theoretical power.

8.1.4. ANR Blanc project Displexity

Participants: Achour Mostefaoui, Anne-Marie Kermarrec, Michel Raynal.

The Displexity project started in October 2011. The aim of this ANR project that also involves researchers from Paris and Bordeaux is to establish the scientific foundations for building up a consistent theory of computability and complexity for distributed computing. One difficulty to be faced by DISPLEXITY is to reconcile two non necessarily disjoint sub-communities, one focusing on the impact of temporal issues, while the other is focusing on the impact of spatial issues.

8.2. European Initiatives

8.2.1. FP7 Projects

8.2.1.1. Gossple

Participants: Mohammad Alaggan, Antoine Boutet, Davide Frey, Arnaud Jegou, Anne-Marie Kermarrec, Konstantinos Kloudas, Afshin Moin, Heverson Ribeiro, François Taiani.

Title: Gossple
Type: IDEAS
Instrument: ERC Starting Grant (Starting)
Duration: September 2008 - August 2013
Coordinator: INRIA (France)
See also: http://www.gossple.fr
Abstract: Anne-Marie Kermarrec is the principal investigator of the GOSSPLE ERC starting Grant (Sept. 2008 - Sept. 2013). GOSSPLE aims at providing a radically new approach to navigating the digital information universe. This project has been granted a 1.250.000 euros budget for 5 years.

GOSSPLE aims at radically changing the navigation on the Internet by placing users affinities and preferences at the heart of the search process. Complementing traditional search engines, GOSSPLE will turn search requests into live data to seek the information where it ultimately is: at the user. GOSSPLE precisely aims at providing a fully decentralized system, auto-organizing, able to discover, capture and leverage the affinities between users and data.

8.2.2. Collaborations in European Programs, except FP7

8.2.2.1. Transform Marie Curie Initial Training Network

Participants: Tyler Crain, Anne-Marie Kermarrec, Achour Mostefaoui, Michel Raynal.

Program: Marie Curie Initial Training Network

Project acronym: Transform

Project title: Theoretical Foundations of Transactional Memory

Duration: May 2010 - October 2013

Coordinator: Michel Raynal - Panagiota Fatourou

Other partners: Foundation for Research and Technology Hellas ICS FORTH Greece, University of Rennes 1 UR1 France, Ecole Polytechnique Federale de Lausanne EPFL Switzerland, Technische Universitaet Berlin TUB Germany, and Israel Institute of Technology Technion.

Abstract:

Transform is a Marie Curie Initial Training Networks European project devoted to the Theoretical Foundations of Transactional Memory (Grant agreement no.: 238639 Date of approval of Annex I by Commission: May 26, 2009). It involves the following universities : Foundation for Research and Technology Hellas ICS FORTH Greece, University of Rennes 1 UR1 France, Ecole Polytechnique Federale de Lausanne EPFL Switzerland, Technische Universitaet Berlin TUB Germany, and Israel Institute of Technology Technion.

Major chip manufacturers have shifted their focus from trying to speed up individual processors into putting several processors on the same chip. They are now talking about potentially doubling efficiency on a 2x core, quadrupling on a 4x core and so forth. Yet multi-core is useless without concurrent programming. The constructors are now calling for a new software revolution: the concurrency revolution. This might look at first glance surprising for concurrency is almost as old as computing and tons of concurrent programming models and languages were invented. In fact, what the revolution is about is way more than concurrency alone: it is about concurrency for the masses. The current parallel programming approach of employing locks is widely considered to be too difficult for any but a few experts. Therefore, a new paradigm of concurrent programming is needed to take advantage of the new regime of multicore computers. Transactional Memory (TM) is a new programming paradigm which is considered by most researchers as the future of parallel programming. Not surprisingly, a lot of work is being devoted to the implementation of TM systems, in hardware or solely in software. What might be surprising is the little effort devoted so far to devising a sound theoretical framework to reason about the TM abstraction. To understand properly TM systems, as well as be able to assess them and improve them, a rigorous theoretical study of the approach, its challenges and its benefits is badly needed. This is the challenging research goal undertaken by this MC-ITN. Our goal through this project is to gather leading researchers in the field of concurrent computing over Europe, and combine our efforts in order to define what might become the modern theory of concurrent computing. We aim at training a set of Early Stage Researchers (ESRs) in this direction and hope that, in turn, these ESRs will help Europe become a leader in concurrent computing. Its keywords are Transactional Memory, Parallelization Mechanisms, Parallel Programming Abstractions, Theory, Algorithms, Technological Sciences

8.2.3. Major European Organizations with which Asap has followed Collaborations

Ecole Polytechnique Federale de Lausanne EPFL Switzerland

collaboration on Gossple ERC, Transform

Foundation for Research and Technology Hellas ICS FORTH Greece

Transform

Technische Universitaet Berlin TUB Germany

Transform

Lancaster University

Gossple

8.3. International Initiatives

8.3.1. Participation In International Programs

8.3.1.1. Demdyn: INRIA/CNPq Collaboration

Participants: Achour Mostefaoui, Marin Bertier, Michel Raynal.

The aim of this project is to exploit dependable aspects of dynamic distributed systems such as VANETs, WiMax, Airborn Networks, DoD Global Information Grid, P2P, etc. Applications that run on these kind of networks have a common point: they are extremely dynamic both in terms of the nodes that take part of them and available resources at a given time. Such dynamics results in instability and uncertainty of the environment which provide great challenges for the implementation of dependable mechanisms that ensure the correct work of the system.

This requires applications to be adaptive, for instance, to less network bandwidth or degraded Quality-of-Service (QoS). Ideally, in these highly dynamic scenarios, adaptiveness characteristics of applications should be self-managing or autonomic. Therefore, being able to detect the occurrence of partitions and automatically adapting the applications for such scenarios is an important dependable requirement for such new dynamic environments.

8.3.2. Visits of International Scientists

Rachid Guerraoui, EPFL Lausanne, Switzerland, May and November 2011 (Rennes). **Darek Kovwalski**, University of Liverpool, UK, March 2011. **Florian Huc**, EPFL Lausanne, Switzerland, May 2011. **Eric Ruppert**, York University, Canada, April 2010. **George Giakkoupis**, University of Calgary, April 2011. **Rida Bazzi**, Arizona State University, June 2011. **Vincent Leroy**, Yahoo! Research, Barcelona, June 2011. **Pascal Felber**, Université de Neuchâtel, February and November 2011. **Hagit Attiya**, Technion, Haifa, Israel, February 2011 **Petr Kuznetsov**, TU Berlin, Germany, February 2011. **Srivastan Ravi**, TU Berlin, Germany, February 2011. **Panagiota Fatourou**, Foundation for Research and Technology Hellas ICS FORTH Greece, February, 2011. **Richard Schlichting**, AT&T Labs Research, November 2011. **Zhu Weiping**, Hong Kong Polytechnic University, China, November, December 2011. **Juan Manuel Turado**, University Juan Carlos 3 Madrid, Spain, September - December 2011. **François Taiani**, Lancaster University, UK, January - December, 2011.

8.3.2.1. Internship

A. Moin was an intern at ETHZ from September 2011 to November 2011.

K. Kloudas was an intern at Imperial College of London from June to September 2011.

A. Boutet was an intern at the Computer Laboratory, University of Cambridge from July 2011 to September 2011.

9. Dissemination

9.1. Animation of the scientific community

9.1.1. Awards not including best papers.

A.-M. Kermarrec received the Monpetit Award from the French Academy of Science in 2011.

9.1.2. Editorial activity, committees, event organization.

A.-M. Kermarrec was an elected member of the INRIA Evaluation Committee until September 2011. She has been a member of the "bureau du CP" since November 2009.

Anne-Marie Kermarrec has been a member of the scientific board of SPECIF (Société des Personnels Enseignants et Chercheurs en Informatique de France) since October 2011.

A.-M. Kermarrec is a nominated member of the ACM Software System Award Committee since October 2009 and the chair of the committee since October 2011.

A.-M. Kermarrec and **François Taiani** organized the second GOSSPLE workshop in December 2011.

A.-M. Kermarrec was co-chair of the program committee of the ACM/IFIP/USENIX International Middleware Conference, Lisbon, December 2001.

A.-M. Kermarrec served in the steering committee of the Eurosys Social Network Systems (SNS), and is a member of the steering committee of the Winter School Hot topics in distributed computing.

A.-M. Kermarrec is a member of the IEEE Internet Computing Editorial Board.

A.-M. Kermarrec served in the program committees for the following conferences:

DISC 2011: *International Symposium on Distributed Computing*, Rome, Italy, September 2011.

Eurosys SNS 2011: *ACM Workshop on Social Network Systems*, Salzburg, Austria, April 2011.

EuroSys Doctoral Workshop 2011: at the *European Conference on Computer Systems*, Salzburg, Austria, April 2011.

EDBT 2011 (Demo): International Conference on Extending Database Technology (Demo track), Uppsala, Sweden, March 2011.

DCOSS 2011: IEEE International Conference on Distributed Computing in Sensor Systems, Barcelona, Spain, June 2011

P2P 2011: IEEE International Conference on Peer to Peer, Kyoto, Japan, August 2011.

IPDPS 2012: *International Parallel & Distributed Processing Symposium*, Shanghai, China, May 2012.

EuroSys 2012: *European Conference on Computer Systems*, Bern, Swizerland, April 2010.

Eurosys SNS 2012: *ACM Workshop on Social Network Systems*, Bern, Swizerland, April 2012.

DEBS 2012: International Conference on Distributed Event-Based Systems, Berlin, Germany, July 2012.

ICDCS 2012: International Conference on Distributed Computing, Macau, China, June 2012.

WWW 2012 (PhD Symposium):] PhD Symposium of the International World Wide Web Conference Lyon, France, June 2012.

VLDB 2012 (Demo): International Conference on Very Large Data Bases (Demo track), Istanbul, Turkey, August 2012.

A. Mostefaoui served in the program committees for the following conferences:

EDCC'12: 9th European Dependable Computing Conference (EDCC'12). Sibiu, Romania, May, 2012.

IWSN/DCOSS 2011: Int. workshop on Interconnections of Wireless Sensor Networks, in conjunction with DCOSS'11, Barcelona, Spain, June 2011.

F. Taiani has served on the program committees of the following events.

ACM SAC 2011 - DADS 7th Dependable and Adaptive Distributed Systems (DADS) Track of the 27th ACM Symposium on Applied Computing.

ACM SAC 2011 - SCS The Self-organizing Complex Systems (SCS) Track of the 27th ACM Symposium on Applied Computing (ACM SAC).

LADC 2011 1st Workshop on Exception Handling in Contemporary Software Systems (EHCoS'11), held in conjunction with the 5th Latin-American Symposium on Dependable Computing.

SOCA 2011 2011 IEEE International Conference on Service Oriented Computing & Applications.

GCM 2011 2nd International Workshop on Green Computing Middleware (GCM'2011), held in conjunction with the ACM/IFIP/USENIX 12th International Middleware Conference.

ARM 2011 10th International Workshop on Adaptive and Reflective Middleware (ARM'11), held in conjunction with the ACM/IFIP/USENIX 12th International Middleware Conference.

MW4SOC 2011 6th Workshop on Middleware for Service Oriented Computing (MW4SOC'11), held in conjunction with the ACM/IFIP/USENIX 12th International Middleware Conference.

M-MPAC 2011 3rd International Workshop on Middleware for Pervasive Mobile and Embedded Computing (M-MPAC 2011), held in conjunction with the ACM/IFIP/USENIX 12th International Middleware Conference.

C4E 2011 Workshop on Clouds for Enterprises 2011 (C4E 2011) held in conjunction with 13th IEEE Conference on Commerce and Enterprise Computing (CEC'11)

WASA-NGI-IV 2011 4th Int. Workshop on Architectures, Services, and Applications for the Next Generation Internet (WASA-NGI-IV), held in conjunction with the IEEE Local Computer Networks (LCN) conference

Middleware 2011 12th ACM/IFIP/USENIX International Middleware Conference 2011

DSN 2011 The 41th Annual IEEE/IFIP International Conference on Dependable Systems and Networks (DSN 2011)

DAIS 2011 11th Int. IFIP Conference on Distributed Applications and Interoperable Systems (DAIS'11)

EWDC 2011 13th European Workshop on Dependable Computing Special theme: Resilience of Evolving Software Systems (EWDC)

NOTERE 2011 11éme édition de la Conférence Internationale sur les NOuvelles Technologies de la REpartition (NOTERE)

F. Taiani organized the 1st Workshop on Middleware and Architectures for Autonomic and Sustainable Computing (MAASC), held in conjunction with the 11éme édition de la Conférence Internationale sur les NOuvelles Technologies de la REpartition.

M. Bertier served in the program committees of CFSE 2011 *The 8th Conférence Française en Système d'Exploitation*, mai 2011, Saint Malo, France.

He also is a member of the organizing committee of CFSE 2011 and RENPAR 2011

D. Frey served in the program committees of the following conferences

Middleware 2011 *ACM/IFIP/USENIX 12th International Middleware Conference*, December 2011, Lisbon, Portugal.

WWW 2012 (PhD Symposium):] PhD Symposium of the International World Wide Web Conference Lyon, France, June 2012.

He also served as a referee for the following journals

- IEEE Transactions on Mobile Computing.
- IEEE Transactions on Parallel and Distributed Systems.

S. Weiss was a referee for the Journal of Parallel and Distributed Computing (JPDC).

A. Boutet has been a member of the “Commission de formation” of INRIA Rennes since 2011.

9.1.3. Invited talks and seminars.

Anne-Marie Kermarrec was invited to give a seminar at EPFL, Swizerland, June 2011.

Anne-Marie Kermarrec was invited to give a talk at the Workshop on Social network, in conjunction with DISC, Roma, September 2011.

Anne-Marie Kermarrec was invited to give a seminar at the University of Pisa, September 2011.

Anne-Marie Kermarrec was invited to give a seminar University of Santa Barbara (UCSB), Technicolor Labs (Palo Alto), Facebook (Palo Alto) and University of San Diego (UCSD), USA, August 2011.

Anne-Marie Kermarrec was invited to give a technical seminar at Google Zurich in November 201.

Marin Bertier was invited to give a talk at "Journée PucesCom", Rennes, June 2011.

François Taiani was invited to deliver a 6-hour lecture on "Cloud Computing in the biogeosciences" at the 2nd Summer School on Biogeodynamics and Earth System Sciences (BESS) organized by Istituto Veneto di Scienze, Lettere ed Arti, the University of Padua and the Natural Environment Science Research Council (NERC, UK) in Venice, Italy, in June 10-17.

Armando Castañeda was invited to give a talk at Workshop on Theoretical Computer Science, Mexico, 18/11/2011.

9.2. Teaching

Master: **Anne-Marie Kermarrec**, P2P Systems and Applications, 15h, M2, Université of Rennes 1, France.

Master: **Anne-Marie Kermarrec**, Gossip-based computing, 5 hours, M2, University of San Sebastian, Spain.

Master: **Michel Raynal**: Distributed Algorithms and Computability, 20h, Université de Rennes 1, France.

Master: **Michel Raynal**: Introduction to Distributed Algorithms 10h, Université de Rennes 1, France.

Master: **Michel Raynal**: Advanced Synchronization, 14h, Université de Rennes 1, France.

Master: **Marin Bertier**, Algorithmique distribuée, 16h, niveau M2, INSA de Rennes, France.

Master: **Marin Bertier**, Système d’exploitation, 56h, niveau M1, INSA de Rennes, France.

Master: **Marin Bertier**, Parallélisme, 20h, niveau M1, INSA de Rennes, France.

Licence: **Marin Bertier**, Programmation C, 26h, niveau L3, INSA de Rennes, France.

Licence: **Marin Bertier**, Unix, 20h, niveau L3, INSA de Rennes, France.

Licence: **Marin Bertier**, Programmation Scheme, 36h, niveau L1, INSA de Rennes, France.

Licence: **Damien Imbs**, computer literacy classes, 64h, first year, university of Rennes 1.

Licence: **Damien Imbs**, Systems and networks 1, 8h, third year, university of Rennes 1.

Master: **Damien Imbs**, Operating systems, 36h, first year, university of Rennes 1.

Licence: **Stéphane Weiss**, Bureautique, 26,67h, niveau L1, Université de Rennes 1, France.

Licence: **Stéphane Weiss**, Systèmes et Réseaux: Systèmes, 98h, niveau L3, ISTIC Université de Rennes 1, France.

Licence: **Stéphane Weiss**, Programmation Web, 18,67h, niveau L3, ISTIC Université de Rennes 1, France.

Anne-Marie Kermarrec gave a talk at the Spring School on Distribute Computing, Marrakesh, May 2011.

9.2.1. PhD & HdR

HdR : François Taiani, “Some Contributions to The Programming of Large-Scale Distributed Systems: Mechanisms, Abstractions, and Tools”, Université de Rennes 1, 17 November 2011.

10. Bibliography

Major publications by the team in recent years

[1] M. BERTIER, D. FREY, R. GUERRAOUI, A.-M. KERMARREC, V. LEROY. *The Gossple Anonymous Social Network*, in "ACM/IFIP/USENIX 11th International Middleware Conference", India Bangalore, November 2010, http://hal.inria.fr/inria-00515693/en.

[2] J. CAO, M. RAYNAL, X. YANG, W. WU. *Design and Performance Evaluation of Efficient Consensus Protocols for Mobile Ad Hoc Networks*, in "IEEE Transactions on Computers", 2007, vol. 56, n^{o} 8, p. 1055–1070.

[3] A. CARNEIRO VIANA, S. MAAG, F. ZAIDI. *One step forward: Linking Wireless Self-Organising Networks Validation Techniques with Formal Testing approaches*, in "ACM Computing Surveys", 2009, http://hal.inria.fr/inria-00429444/en/.

[4] D. FREY, R. GUERRAOUI, A.-M. KERMARREC, M. MONOD, K. BORIS, M. MARTIN, V. QUÉMA. *Heterogeneous Gossip*, in "Middleware 2009", Urbana-Champaign, IL, USA, 2009, http://hal.inria.fr/inria-00436125/en/.

[5] R. FRIEDMAN, A. MOSTEFAOUI, S. RAJSBAUM, M. RAYNAL. *Distributed agreement problems and their connection with error-correcting codes*, in "IEEE Transactions on Computers", 2007, vol. 56, n^{o} 7, p. 865–875.

[6] A. J. GANESH, A.-M. KERMARREC, E. LE MERRER, L. MASSOULIÉ. *Peer counting and sampling in overlay networks based on random walks*, in "Distributed Computing", 2007, vol. 20, n^{o} 4, p. 267-278.

[7] M. JELASITY, S. VOULGARIS, R. GUERRAOUI, A.-M. KERMARREC, M. VAN STEEN. *Gossip-Based Peer Sampling.*, in "ACM Transactions on Computer Systems", August 2007, vol. 41, n^{o} 5.

[8] B. MANIYMARAN, M. BERTIER, A.-M. KERMARREC. *Build One, Get One Free: Leveraging the Coexistence of Multiple P2P Overlay Networks.*, in "Proceedings of ICDCS 2007", Toronto, Canada, June 2007.

[9] A. MOSTEFAOUI, S. RAJSBAUM, M. RAYNAL, C. TRAVERS. *From Diamond W to Omega: a simple bounded quiescent reliable broadcast-based transformation*, in "Journal of Parallel and Distributed Computing", 2007, vol. 61, n^{o} 1, p. 125–129.

[10] J. PATEL, É. RIVIÈRE, I. GUPTA, A.-M. KERMARREC. *Rappel: Exploiting interest and network locality to improve fairness in publish-subscribe systems.*, in "Computer Networks", 2009, vol. 53, n^{o} 13, http://hal.inria.fr/inria-00436057/en/.

Publications of the year

Doctoral Dissertations and Habilitation Theses

[11] F. TAÏANI. *Programmation des grands systèmes distribués: quelques mécanismes, abstractions, et outils*, Université Rennes 1, November 2011, Habilitation à Diriger des Recherches, http://hal.inria.fr/tel-00643729/en.

Articles in International Peer-Reviewed Journal

[12] X. BAI, R. GUERRAOUI, A.-M. KERMARREC, V. LEROY. *Collaborative Personalized Top-k Processing*, in "ACM Transactions on Database Systems (TODS)", 2011, vol. 36, n° 4, http://hal.inria.fr/hal-00652036/en.

[13] Y. BUSNEL, L. QUERZONI, R. BALDONI, M. BERTIER, A.-M. KERMARREC. *Analysis of Deterministic Tracking of Multiple Objects using a Binary Sensor Network*, in "ACM Transactions on Sensor Networks", 2011, vol. 8, http://hal.inria.fr/inria-00590873/en.

[14] P. FELBER, A.-M. KERMARREC, L. LEONINI, É. RIVIÈRE, S. VOULGARIS. *PULP: an Adaptive Gossip-Based Dissemination Protocol for Multi-Source Message Streams.*, in "Peer-to-Peer Networking and Applications, Springer", 2011, http://hal.inria.fr/hal-00646616/en.

[15] R. GUERRAOUI, K. HUGUENIN, A.-M. KERMARREC, M. MONOD, Ý. VIGFÚSSON. *Decentralized Polling with Respectable Participants*, in "Journal of Parallel and Distributed Computing", January 2012, vol. 72, n° 1 [*DOI :* 10.1016/J.JPDC.2011.09.003], http://hal.inria.fr/inria-00629455/en.

[16] D. IMBS, M. RAYNAL. *A liveness condition for concurrent objects: x-wait-freedom*, in "Concurrency and Computation: Practice and Experience", 2011, vol. 23, n° 17, p. 2154-2166, http://hal.inria.fr/hal-00646908/en.

[17] D. IMBS, M. RAYNAL. *Software transactional memories: an approach for multicore programming*, in "Journal of Supercomputing", 2011, vol. 57, n° 2, p. 203-215, http://hal.inria.fr/hal-00646909/en.

[18] D. IMBS, M. RAYNAL. *Help when needed, but no more: Efficient read/write partial snapshot*, in "Journal of Parallel and Distributed Computing", 2012, vol. 72, n° 1, p. 1-12, http://hal.inria.fr/hal-00646906/en.

[19] A.-M. KERMARREC, E. LE MERRER, B. SERICOLA, G. TRÉDAN. *Second order centrality: Distributed assessment of nodes criticity in complex networks*, in "Computer Communications", 2011, vol. 34, n° 5 [*DOI :* 10.1016/J.COMCOM.2010.06.007], http://hal.inria.fr/inria-00506385/en.

[20] A.-M. KERMARREC, G. TAN. *Greedy Geographic Routing in Large-Scale Sensor Networks: A Minimum Network Decomposition Approach.*, in "IEEE/ACM Transactions on Networking", 2011, To appear, http://hal.inria.fr/inria-00619038/en.

[21] M. RAYNAL, R. BALDONI, S. BONOMI. *Implementing a Regular Register in an Eventually Synchronous Distributed System prone to Continuous Churn*, in "IEEE Transactions on Parallel and Distributed Systems", 2011, http://hal.inria.fr/hal-00649322/en.

[22] M. RAYNAL, F. BONNET. *On the road to the Weakest Failure Detector for k-Set Agreement in Message-passing Systems*, in "Theoretical Computer Science", June 2011, vol. 412, n° 33, p. 4273-4284, http://hal.inria.fr/hal-00649311/en.

[23] M. RAYNAL, F. BONNET. *The Price of Anonymity: Optimal Consensus despite Asynchrony, Crash and Anonymity*, in "ACM Transactions on Autonomous and Adaptive Systems (TAAS)", 2011, vol. 6, n^{o} 4, http://hal.inria.fr/hal-00652434/en.

International Conferences with Proceedings

[24] M. ALAGGAN, S. GAMBS, A.-M. KERMARREC. *Private Similarity Computation in Distributed Systems: from Cryptography to Differential Privacy*, in "OPODIS", Toulouse, France, LNCS, Springer-Verlag Berlin Heidelberg, December 2011, vol. 7109, p. 357 - 377, http://hal.inria.fr/hal-00646831/en.

[25] K. ASLAN, P. MOLLI, H. SKAF-MOLLI, S. WEISS. *C-Set : a Commutative Replicated Data Type for Semantic Stores*, in "RED: Fourth International Workshop on REsource Discovery", Heraklion, Greece, May 2011, http://hal.inria.fr/inria-00594590/en.

[26] O. BALDELLON, A. MOSTEFAOUI, M. RAYNAL. *A Necessary and Sufficient Synchrony Condition for Solving Byzantine Consensus in Symmetric Networks*, in "12th International Conference on Distributed Computing and Networking (ICDCN'11)", Bangalore, India, M. AGUILERA, ET AL. (editors), Lecture Notes in Computer Science, Springer Verlag, January 2011, vol. 6522, p. 215-226, http://hal.inria.fr/hal-00647764/en.

[27] O. BALDELLON, A. MOSTEFAOUI, M. RAYNAL. *A Symmetric Synchrony Condition for Solving Byzantine Consensus*, in "12th International Conference on Distributed Computing and Networking (ICDCN 2011)", Bangalore, India, M. AGUILERA, ET AL. (editors), LNCS, Springer Verlag, January 2011, vol. 6522, p. 215–226, http://hal.inria.fr/inria-00544666/en.

[28] M. BERTIER, M. OBROVAC, C. TEDESCHI. *A Protocol for the Atomic Capture of Multiple Molecules at Large Scale*, in "13th International Conference on Distributed Computing and Networking", Hong-Kong, China, January 2012, http://hal.inria.fr/hal-00644262/en.

[29] T. CRAIN, D. IMBS, M. RAYNAL. *Read Invisibility, Virtual World Consistency and Probabilistic Permissiveness are Compatible*, in "11th International Conference on Algorithms and Architectures for Parallel Processing (ICA3PP'11)", Melbourne, Australia, Springer-Verlag LNCS, 2011, p. 244-257, http://hal.inria.fr/hal-00646902/en.

[30] T. CRAIN, D. IMBS, M. RAYNAL. *Towards a universal construction for transaction-based multiprocess programs*, in "13th International Conference on Distributed Computing and Networking (ICDCN'12)", Hong Kong, Hong Kong, Springer-Verlag LNCS, 2012, http://hal.inria.fr/hal-00646911/en.

[31] S. DEFRANCE, A.-M. KERMARREC, E. LE MERRER, N. LE SCOUARNEC, G. STRAUB, A. VAN KEMPEN. *Efficient peer-to-peer backup services through buffering at the edge*, in "Peer-to-Peer Computing (P2P), 2011 IEEE International Conference on P2P systems", Kyoto, Japan, August 2011, p. 142 - 151 [*DOI :* 10.1109/P2P.2011.6038671], http://hal.inria.fr/hal-00646768/en.

[32] C. DELPORTE-GALLET, H. FAUCONNIER, R. GUERRAOUI, A.-M. KERMARREC, E. RUPPERT, T.-T. HUNG. *Byzantine Agreement with Homonyms*, in "30th Annual ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing (PODC)", San Jose, United States, 2011, http://hal.inria.fr/inria-00590866/en.

[33] D. FREY, A. JÉGOU, A.-M. KERMARREC. *Social Market: Combining Explicit and Implicit Social Networks*, in "International Symposium on Stabilization, Safety, and Security of Distributed Systems", Grenoble, France, LNCS, October 2011, http://hal.inria.fr/inria-00624129/en.

[34] A. HAGIT, A. CASTAÑEDA. *A non-topological proof for the impossibility of k-set agreement*, in "13th International Symposium on Stabilization, Safety, and Security of Distributed Systems", Grenoble, France, October 2011, http://hal.inria.fr/hal-00650623/en.

[35] K. HUGUENIN, A. YAHYAVI, B. KEMME. *Short paper: Cheat Detection and Prevention in P2P MOGs*, in "ACM/IEEE 10th International Workshop on Network and Systems Support for Games (NETGAMES)", Ottawa, Canada, October 2011, http://hal.inria.fr/inria-00616878/en.

[36] D. IMBS, S. RAJSBAUM, M. RAYNAL. *The Universe of Symmetry Breaking Tasks*, in "18th International Colloquium on Structural Information and Communication Complexity (SIROCCO'11)", Gdansk, Poland, Springer-Verlag LNCS, 2011, p. 66-77, http://hal.inria.fr/hal-00646904/en.

[37] D. IMBS, M. RAYNAL. *A Simple Snapshot Algorithm for Multicore Systems*, in "5th Latin-American Symposium on Dependable Computing (LADC'11)", São José dos Campos, Brazil, 2011, p. 17-24, http://hal.inria.fr/hal-00646903/en.

[38] *Best Paper*
D. IMBS, M. RAYNAL. *The Weakest Failure Detector to Implement a Register in Asynchronous Systems with Hybrid Communication*, in "Proc. 13th Int'l Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS'11)", Grenoble, France, Springer-Verlag LNCS, 2011, p. 268-282, http://hal.inria.fr/hal-00646899/en.

[39] A.-M. KERMARREC, N. LE SCOUARNEC, G. STRAUB. *Repairing Multiple Failures with Coordinated and Adaptive Regenerating Codes*, in "The 2011 International Symposium on Network Coding", Beijing, China, 2011, http://hal.inria.fr/inria-00590869/en.

[40] A.-M. KERMARREC, V. LEROY, C. THRAVES. *Converging Quickly to Independent Uniform Random Topologies*, in "19th Euromicro International Conference on Parallel, Distributed and Network-Based Computing (PDP), 2011.", Ayia Napa, Cyprus, February 2011, http://hal.inria.fr/inria-00543249/en.

[41] A.-M. KERMARREC, V. LEROY, G. TRÉDAN. *Distributed social graph embedding*, in "Conference on Information and Knowledge Management (CIKM)", Glasgow, United Kingdom, ACM, 2011, http://hal.inria.fr/hal-00646611/en.

[42] A.-M. KERMARREC, C. THRAVES. *Can everybody sit closer to their friends than their enemies?*, in "36th International Symposium on Mathematical Foundations of Computer Science (MFCS)", Warsow, Poland, 2011, http://hal.inria.fr/inria-00599701/en.

[43] S. LIN, F. TAÏANI, M. BERTIER, A.-M. KERMARREC. *Transparent Componentisation: High-level (Re)configurable Programming for Evolving Distributed Systems*, in "26th ACM Symposium on Applied Computing", Taichung, Taiwan, Province Of China, March 2011, http://hal.inria.fr/inria-00544510/en.

[44] A. MOSTEFAOUI, M. RAYNAL. *Looking for Efficient Implementations of Concurrent Objects*, in "11th International Conference on Parallel Computing Technologies (PaCT'11)", Kazan, Russian Federation, V. MALYSHKIN (editor), Lecture Notes in Computer Science, Springer Verlag, September 2011, vol. 6873, p. 74-87, http://hal.inria.fr/hal-00647775/en.

[45] A. MOSTEFAOUI, M. RAYNAL, J. STAINER. *Relations Linking Failure Detectors Associated with k-Set Agreement in Message-Passing Systems*, in "13th International Symposium on Stabilization, Safety, and Security of Distributed Systems (SSS 2011)", Grenoble, France, X. DÉFAGO, ET AL. (editors), Lecture Notes in Computer Science, Springer Verlag, October 2011, vol. 6976, p. 341-355, http://hal.inria.fr/hal-00648465/en.

[46] M. RAYNAL, M. GRADINARIU POTOP-BUTUCARU, S. TIXEUIL. *Distributed Computing with Mobile Robots: an Introductory Survey*, in "Proc. 14th Int'l Conference on Network-Based Information Systems (NBiS'11)", Tirana, Albania, September 2011, http://hal.inria.fr/hal-00649294/en.

[47] M. RAYNAL, S. RAJSBAUM. *A Survey on Some Recent Advances in Shared Memory Models*, in "Proc. 18th Int'l Colloquium on Structural Information and Communication Complexity (SIROCCO'11)", Gdansk, Poland, June 2011, http://hal.inria.fr/hal-00649297/en.

[48] M. RAYNAL, S. RAJSBAUM. *A Theory-Oriented Introduction to Wait-free Synchronization Based on the adaptive Renaming Problem*, in "IEEE 25th Int'l Conference on Advanced Information Networking and Applications (AINA'11)", Singapour, Singapore, March 2011, http://hal.inria.fr/hal-00649266/en.

[49] A. YAHYAVI, K. HUGUENIN, B. KEMME. *AntReckoning: A Pheromone-Based Dead Reckoning Algorithm for Games*, in "ACM/IEEE 10th International Workshop on Network and Systems Support for Games (NETGAMES)", Ottawa, Canada, October 2011, http://hal.inria.fr/inria-00616877/en.

Books or Proceedings Editing

[50] A.-M. KERMARREC, F. KON (editors). *ACM/IFIP/USENIX 12th International Middleware Conference*, Lecture Notes in Computer Science, Springer, 2011, vol. 7049, http://hal.inria.fr/hal-00646605/en.

Research Reports

[51] A. CASTAÑEDA, D. IMBS, S. RAJSBAUM, M. RAYNAL. *Enriching the reduction map of sub-consensus tasks*, INRIA, April 2011, n^{o} PI-1976, http://hal.inria.fr/inria-00591526/en.

[52] I. TRESTIAN, K. HUGUENIN, L. SU, A. KUZMANOVIC. *Understanding Human Movement Semantics: A Point of Interest Based Approach*, INRIA, August 2011, n^{o} RR-7716, http://hal.inria.fr/inria-00616876/en.

Other Publications

[53] A. BOUTET, E. YONEKI. *Member Classification and Party Characteristics in Twitter during UK Election*, December 2011, The 1st international workshop on dynamic systems (DYNAM), http://hal.inria.fr/hal-00645428/en.

[54] A.-M. KERMARREC, F. TAÏANI. *Constellation: Programming decentralised social networks*, 2011, 2 pages, to be presented at the 1st International Workshop for Languages for Distributed Algorithms (LADA-2012), Philadelphia, USA, 23-24 January 2012, co-located with POPL 2012: 39th ACM SIGPLAN-SIGACT Symp. on Principles of Programming Languages, http://hal.inria.fr/hal-00646730/en.

References in notes

[55] M. AGUILERA. *A Pleasant Stroll Through the Land of Infinitely Many Creatures.*, in "ACM SIGACT News, Distributed Computing Column", 2004, vol. 35, n^{o} 2.

[56] D. ANGLUIN. *Local and Global Properties in Networks of Processes.*, in "Proc. 12th ACM Symposium on Theory of Computing (STOC'80)", 1980.

[57] K. BIRMAN, M. HAYDEN, O. OZKASAP, Z. XIAO, M. BUDIU, Y. MINSKY. *Bimodal Multicast*, in "ACM Transactions on Computer Systems", May 1999, vol. 17, n^{o} 2, p. 41-88.

[58] A. DEMERS, D. GREENE, C. HAUSER, W. IRISH, J. LARSON, S. SHENKER, H. STURGIS, D. SWINEHART, D. TERRY. *Epidemic algorithms for replicated database maintenance*, in "Proceedings of the ACM Symposium on Principles of Distributed Computing (PODC'87)", August 1987.

[59] P. EUGSTER, S. HANDURUKANDE, R. GUERRAOUI, A.-M. KERMARREC, P. KOUZNETSOV. *Lightweight Probabilistic Broadcast*, in "ACM Transaction on Computer Systems", November 2003, vol. 21, n^{o} 4.

[60] L. LAMPORT. *Time, clocks, and the ordering of events in distributed systems*, in "Communications of the ACM", 1978, vol. 21, n^{o} 7.

[61] M. MERRITT, G. TAUBENFELD. *Computing Using Infinitely Many Processes.*, in "Proc. 14th Int'l Symposium on Distributed Computing (DISC'00)", 2000.

[62] S. RATNASAMY, P. FRANCIS, M. HANDLEY, R. KARP, S. SHENKER. *A Scalable Content-Addressable Network*, in "Conference of the Special Interest Group on Data Communication (SIGCOMM'01)", 2001.

[63] A. ROWSTRON, P. DRUSCHEL. *Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems*, in "IFIP/ACM Intl. Conf. on Distributed Systems Platforms (Middleware)", 2001.

[64] I. STOICA, R. MORRIS, D. KARGER, F. KAASHOEK, H. BALAKRISHNAN. *Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications*, in "SIGCOMM'01", 2001.

[65] S. VOULGARIS, D. GAVIDIA, M. VAN STEEN. *CYCLON: Inexpensive Membership Management for Unstructured P2P Overlays*, in "Journal of Network and Systems Management", 2005, vol. 13, n^{o} 2.