IN PARTNERSHIP WITH:
Université des sciences et technologies de Lille (Lille 1)

Activity Report 2014

Team SPIRALS

Self-adaptation for distributed services and large software systems

IN COLLABORATION WITH: Laboratoire d'informatique fondamentale de Lille (LIFL)

RESEARCH CENTER
Lille - Nord Europe

THEME
Distributed Systems and middleware

Table of contents

Team SPIRALS

Keywords: Self-adaptive Systems, Distributed Systems, Software Engineering, Autonomic Computing, Cloud Computing, Program Transformation

Creation of the Team: 2014 January 01*, updated into Project-Team:* 2015 January 01.

1. Members

Research Scientist

Philippe Merle [Inria, Researcher]

Faculty Members

Lionel Seinturier [Team leader, Univ. Lille I, Professor, HdR]
Laurence Duchien [Univ. Lille I, Professor, HdR]
Matias Martinez [Univ. Lille I, ATER]
Martin Monperrus [Univ. Lille I, Associate Professor]
Romain Rouvoy [Univ. Lille I, Associate Professor, Délégation Inria 50% from Sep 2014, HdR]
Walter Rudametkin Ivey [Univ. Lille I, Associate Professor, from Sep 2014]
Alexandre Feugas [Univ. Lille I, ATER, until Aug 2014]
Nicolas Haderer [Univ. Lille I, ATER, until Aug 2014]
Adel Noureddine [Univ. Lille I, ATER, until Jul 2014]

Engineers

Clive Ferret-Canape [Inria, from Jun 2014]
Julien Duribreux [Inria, from Oct 2014]
Loïc Huertas [Inria]
Gérard Paligot [Inria, from Oct 2014]
Nicolas Petitprez [Inria, from Sep 2014]
Christophe Ribeiro [Inria]
Antoine Veuiller [Inria, from Oct 2014]

PhD Students

Maxime Colmant [ADEME/Univ. Lille I]
Benoit Cornu [Univ. Lille I]
María Gómez Lacruz [Inria]
Geoffrey Hecht [Univ. Lille I, co-supervision with UQAM]
Vincenzo Musco [Univ. Lille III/Univ. Lille I, co-supervision with P. Preux from the SequeL project-team]
Fawaz Paraiso [Inria, until Sep 2014]
Clément Quinton [Inria, until Oct 2014]
Marc Sango [IFSTTAR]
Gustavo Sousa [Erasmus Mundus, from Sep 2014]
Amal Tahri [Orange]
Bo Zhang [Univ. Lille I]

Post-Doctoral Fellows

Daniel Romero Acero [Inria]
Filip Krikava [Univ. Lille I]
Jifeng Xuan [Inria]

Visiting Scientists

Earl Barr [Inria, Jun 2014]
Mohamed Berkane [University Constantine 2, until Sep 2014]
Wendpanga Ouedraogo [INSA Lyon, Jan 2014]

Francis Palma [UQAM, from Oct 2014 until Nov 2014]

Administrative Assistants

Karine Lewandowski [Inria, Jan 2014]
Aurore Hermant [Inria, from Feb 2014]

Others

Maxime Clement [Univ. Lille I, Bachelor Intern, from Mar 2014 until Jun 2014]
Julien Duribreux [Inria, Master Intern, from Mar 2014 until Aug 2014]
Mickaël Duruisseau [Inria, Master Intern, from Mar 2014 until Aug 2014]
Sebastian Lamelas Marcote [Inria, Master Intern, from May 2014 until Oct 2014]
Pierre Monnet [Univ. Lille I, Bachelor Intern, from Apr 2014 until Jun 2014]
Carolina Valdez Gandara [Inria, Master Intern, from May 2014 until Nov 2014]
Christophe Gransart [IFSTTAR, Researcher, External Collaborator]
Daniel Le Berre [Univ. Artois, Professor, External Collaborator, until Aug 2014, HdR]

2. Overall Objectives

2.1. Introduction

Our research is based on two complementary fields: distributed systems and software engineering. We aim at introducing more automation in the adaptation processes of software systems, that is, transitioning from the study of adaptive systems to self-adaptive systems. In particular, we work towards two directions: self-healing software systems with data mining solutions, and self-optimizing software systems with context monitoring. These two objectives are declined for two target environments, mobile computing and cloud computing.

2.2. Scientific Foundations

Distributed software services and systems are central to many human activities, such as communication, commerce, education, defense, etc. Distributed software services consist of an ever growing number of devices, often highly heterogeneous, from cloud platforms, sensor networks, to application servers, desktop machines, and mobile devices such as smartphones. The future of this huge number of interconnected software services has been called the Internet of Services, a vision *"where everything that is needed to use software applications is available as a service on the Internet, such as the software itself, the tools to develop the software, the platform servers, storage and communication to run the software."* [1] This pervasiveness continuously leads to new usages that in turn foster the emergence of novel requirements and concepts for new software services. Hence, it is necessary to establish new paradigms to design and execute software programs in these highly interconnected and heterogeneous environments, and it is necessary to ensure not only that these software systems can be adapted to new usages, new infrastructures, and new execution environments in the long term, but also that after the adaptation process the services still perform as expected.

This research project focuses on defining *self-adaptive* software services and middleware. From the perspective of the Internet of Services, this project fits in the vision sketched by *e.g.*, the FP8 Expert Group Services in the Future Internet [114], the NESSI Research Priorities for the next Framework Programme for Research and Technological Development FP8 [120], the Roadmap for Advanced Cloud Technologies under H2020 [115], and research roadmaps such as [126], [113], [100].

3. Research Program

3.1. Introduction

Our research program on self-adaptive software targets two key properties that are detailed in the remainder of this section: self-healing and self-optimization.

[1] http://cordis.europa.eu/fp7/ict/ssai

3.2. Objective #1: Self-healing - Mining software artifacts to automatically evolve systems

Software systems are under the pressure of changes all along their lifecycle. Agile development blurs the frontier between design and execution and requires constant adaptation. The size of systems (millions of lines of code) multiplies the number of bugs by the same order of magnitude. More and more systems, such as sensor network devices, live in "surviving" mode, in the sense that they are neither rebootable nor upgradable.

Software bugs are hidden in source code and show up at development-time, testing-time or worse, once deployed in production. Except for very specific application domains where formal proofs are achievable, bugs can not be eradicated. As an order of magnitude, on 16 Dec 2011, the Eclipse bug repository contains 366,922 bug reports. Software engineers and developers work on bug fixing on a daily basis. Not all developers spend the same time on bug fixing. In large companies, this is sometimes a full-time role to manage bugs, often referred to as Quality Assurance (QA) software engineers. Also, not all bugs are equal, some bugs are analyzed and fixed within minutes, others may take months to be solved [123].

In terms of research, this means that: (i) one needs means to automatically adapt the design of the software system through automated refactoring and API extraction, (ii) one needs approaches to automate the process of adapting source code in order to fix certain bugs, (iii) one needs to revisit the notion of error-handling so that instead of crashing in presence of errors, software adapts itself to continue with its execution, e.g., in degraded mode.

There is no one-size-fits-all solution for each of these points. However, we think that novel solutions can be found by using **data mining and machine learning techniques tailored for software engineering** [124]. This body of research consists of mining some knowledge about a software system by analyzing the source code, the version control systems, the execution traces, documentation and all kinds of software development and execution artifacts in general. This knowledge is then used within recommendation systems for software development, auditing tools, runtime monitors, frameworks for resilient computing, etc.

The novelty of our approach consists of using and tailoring data mining techniques for analyzing software artifacts (source code, execution traces) in order to achieve the **next level of automated adaptation** (*e.g.*, automated bug fixing). Technically, we plan to mix unsupervised statistical learning techniques (*e.g.* frequent item set mining) and supervised ones (*e.g.* training classifiers such as decision trees). This research is currently not being performed by data mining research teams since it requires a high level of domain expertise in software engineering, while software engineering researchers can use off-the-shelf data mining libraries, such as Weka [98].

We now detail the two directions that we propose to follow to achieve this objective.

3.2.1. Learning from software history how to design software and fix bugs

The first direction is about mining techniques in software repositories (*e.g.*, CVS, SVN, Git). Best practices can be extracted by data mining source code and the version control history of existing software systems. The design and code of expert developers significantly vary from the artifacts of novice developers. We will learn to differentiate those design characteristics by comparing different code bases, and by observing the semantic refactoring actions from version control history. Those design rules can then feed the test-develop-refactor constant adaptation cycle of agile development.

Fault localization of bugs reported in bug repositories. We will build a solid foundation on empirical knowledge about bugs reported in bug repository. We will perform an empirical study on a set of representative bug repositories to identify classes of bugs and patterns of bug data. For this, we will build a tool to browse and annotate bug reports. Browsing will be helped with two kinds of indexing: first, the tool will index all textual artifacts for each bug report; second it will index the semantic information that is not present by default in bug management software (*i.e.*, “contains a stacktrace”). Both indexes will be used to find particular subsets of bug reports, for instance “all bugs mentioning invariants and containing a stacktrace”. Note that queries with this kind of complexity and higher are mostly not possible with the state-of-the-art of bug management software. Then, analysts will use annotation features to annotate bug reports. The main outcome of the empirical study will be the identification of classes of bugs that are appropriate for automated localization. Then, we will run machine learning algorithms to identify the latent links between the bug report content and source code features. Those algorithms would use as training data the existing traceability links between bug reports and source code modifications from version control systems. We will start by using decision trees since they produce a model that is explicit and understandable by expert developers. Depending on the results, other machine learning algorithms will be used. The resulting system will be able to locate elements in source code related to a certain bug report with a certain confidence.

Automated bug fix generation with search-based techniques. Once a location in code is identified as being the cause of the bug, we can try to automatically find a potential fix. We envision different techniques: (1) infer fixes from existing contracts and specifications that are violated; (2) infer fixes from the software behavior specified as a test suite; (3) try different fix types one-by-one from a list of identified bug fix patterns; (4) search fixes in a fix space that consists of combinations of atomic bug fixes. Techniques 1 and 2 are explored in [91] and [122]. We will focus on the latter techniques. To identify bug fix patterns and atomic bug fixes, we will perform a large-scale empirical study on software changes (also known as changesets when referring to changes across multiple files). We will develop tools to navigate, query and annotate changesets in a version control system. Then, a grounded theory will be built to master the nature of fixes. Eventually, we will decompose change sets in atomic actions using clustering on changeset actions. We will then use this body of empirical knowledge to feed search-based algorithms (*e.g.* genetic algorithms) that will look for meaningful fixes in a large fix space. To sum up, our research on automated bug fixing will try not only to point to source code locations responsible of a bug, but to search for code patterns and snippets that may constitute the skeleton of a valid patch. Ultimately, a blend of expert heuristics and learned rules will be able to produce valid source code that can be validated by developers and committed to the code base.

3.2.2. Run-time self-healing

The second proposed research direction is about inventing a self-healing capability at run-time. This is complementary to the previous objective that mainly deals with development time issues. We will achieve this in two steps. First, we want to define frameworks for resilient software systems. Those frameworks will help to maintain the execution even in the presence of bugs, *i.e.* to let the system survive. As exposed below, this may mean for example to switch to some degraded modes. Next, we want to go a step further and to define solutions for automated runtime repair, that is, not simply compensating the erroneous behavior, but also determining the correct repair actions and applying them at run-time.

Mining best effort values. A well-known principle of software engineering is the “fail-fast” principle. In a nutshell, it states that as soon as something goes wrong, software should stop the execution before entering incorrect states. This is fine when a human user is in the loop, capable of understanding the error or at least rebooting the system. However, the notion of “failure-oblivious computing” [112] shows that in certain domains, software should run in a resilient mode (*i.e.* capable of recovering from errors) and/or best-effort mode (*i.e.* a slightly imprecise computation is better than stopping). Hence, we plan to investigate data mining techniques in order to learn best-effort values from past executions (*i.e.* somehow learning what is a correct state, or the opposite what is not a completely incorrect state). This knowledge will then be used to adapt the software state and flow in order to mitigate the error consequences, the exact opposite of fail-fast for systems with long-running cycles.

Embedding search based algorithms at runtime. Harman recently described the field of search-based software engineering [99]. We think that certain search based approaches can be embedded at runtime with the goal of automatically finding solutions that avoid crashing. We will create software infrastructures that allow automatically detecting and repairing faults at run-time. The methodology for achieving this task is based on three points: (1) empirical study of runtime faults; (2) learning approaches to characterize runtime faults; (3) learning algorithms to produce valid changes to the software runtime state. An empirical study will be performed to analyze those bug reports that are associated with runtime information (*e.g.* core dumps or stacktraces). After this empirical study, we will create a system that learns on previous repairs how to produce small changes that solve standard runtime bugs (*e.g.* adding an array bound check to throw a handled domain exception rather than a spurious language exception). To achieve this task, component models will be used to (1) encapsulate the monitoring and reparation meta-programs in appropriate components and (2) support runtime code modification using scripting, reflective or bytecode generation techniques.

3.3. Objective #2: Self-optimization - Sharing runtime behaviors to continuously adapt software

Complex distributed systems have to seamlessly adapt to a wide variety of deployment targets. This is due to the fact that developers cannot anticipate all the runtime conditions under which these systems are immersed. A major challenge for these software systems is to develop their capability to continuously reason about themselves and to take appropriate decisions and actions on the optimizations they can apply to improve themselves. This challenge encompasses research contributions in different areas, from environmental monitoring to realtime symptoms diagnosis, to automated decision making. The variety of distributed systems, the number of optimization parameters, and the complexity of decisions often resign the practitioners to design monolithic and static middleware solutions. However, it is now globally acknowledged that the development of dedicated building blocks does not contribute to the adoption of sustainable solutions. This is confirmed by the scale of actual distributed systems, which can—for example—connect several thousands of devices to a set of services hosted in the Cloud. In such a context, the lack of support for smart behaviours at different levels of the systems can inevitably lead to its instability or its unavailability. In June 2012, an outage of Amazon's Elastic Compute Cloud in North Virginia has taken down Netflix, Pinterest, and Instagram services. During hours, all these services failed to satisfy their millions of customers due to the lack of integration of a self-optimization mechanism going beyond the boundaries of Amazon.

The research contributions we envision within this area will therefore be organized as a reference model for engineering **self-optimized distributed systems** autonomously driven by *adaptive feedback control loops*, which will automatically enlarge their scope to cope with the complexity of the decisions to be taken. This solution introduces a multi-scale approach, which first privileges local and fast decisions to ensure the homeostasis [2] property of a single node, and then progressively propagates symptoms in the network in order to reason on a longer term and a larger number of nodes. Ultimately, domain experts and software developers can be automatically involved in the decision process if the system fails to find a satisfying solution. The research program for this objective will therefore focus on the study of mechanisms for **monitoring, taking decisions, and automatically reconfiguring software at runtime and at various scales**. As stated in the self-healing objective, we believe that there is no one-size-fits-all mechanism that can span all the scales of the system. We will therefore study and identify an optimal composition of various adaptation mechanisms in order to produce long-living software systems.

The novelty of this objective is to exploit the wisdom of crowds to define new middleware solutions that are able to continuously adapt software deployed in the wild. We intend to demonstrate the applicability of this approach to distributed systems that are deployed from mobile phones to cloud infrastructures. The key scientific challenges to address can be summarized as follows: *How does software behave once deployed in the wild? Is it possible to automatically infer the quality of experience, as it is perceived by users? Can the runtime optimizations be shared across a wide variety of software? How optimizations can be safely operated on large populations of software instances?*

The remainder of this section further elaborates on the opportunities that can be considered within the frame of this objective.

3.3.1. Monitoring software in the wild

Once deployed, developers are generally no longer aware of how their software behave. Even if they heavily use testbeds and benchmarks during the development phase, they mostly rely on the bugs explicitly reported by users to monitor the efficiency of their applications. However, it has been shown that contextual artifacts collected at runtime can help to understand performance leaks and optimize the resilience of software

[2] Homeostasis is the property of a system that regulates its internal environment and tends to maintain a stable, relatively constant condition of properties [Wikipedia].

systems [125]. Monitoring and understanding the context of software at runtime therefore represent the first building block of this research challenge. Practically, we intend to investigate crowdsensing approaches, to smartly collect and process runtime metrics (*e.g.*, request throughput, energy consumption, user context). Crowdsensing can be seen as a specific kind of crowdsourcing activity, which refers to the capability of lifting a (large) diffuse group of participants to delegate the task of retrieving trustable data from the field. In particular, crowdsensing covers not only *participatory sensing* to involve the user in the sensing task (*e.g.*, surveys), but also *opportunistic sensing* to exploit mobile sensors carried by the user (*e.g.*, smartphones).

While reported metrics generally enclose raw data, the monitoring layer intends to produce meaningful indicators like the *Quality of Experience* (QoE) perceived by users. This QoE reflects representative symptoms of software requiring to trigger appropriate decisions in order to improve its efficiency. To diagnose these symptoms, the system has to process a huge variety of data including runtime metrics, but also history of logs to explore the sources of the reported problems and identify opportunities for optimizations. The technics we envision at this level encompass machine learning, principal component analysis, and fuzzy logic [111] to provide enriched information to the decision level.

3.3.2. Collaborative decision-making approaches

Beyond the symptoms analysis, decisions should be taken in order to improve the *Quality of Service* (QoS). In our opinion, collaborative approaches represent a promising solution to effectively converge towards the most appropriate optimization to apply for a given symptom. In particular, we believe that exploiting the wisdom of the crowd can help the software to optimize itself by sharing its experience with other software instances exhibiting similar symptoms. The intuition here is that the body of knowledge that supports the optimization process cannot be specific to a single software instance as this would restrain the opportunities for improving the quality and the performance of applications. Rather, we think that any software instance can learn from the experience of others.

With regard to the state-of-the-art, we believe that a multi-levels decision infrastructure, inspired from distributed systems like Spotify [95], can be used to build a decentralized decision-making algorithm involving the surrounding peers before requesting a decision to be taken by more central control entity. In the context of collaborative decision-making, peer-based approaches therefore consist in quickly reaching a consensus on the decision to be adopted by a majority of software instances. Software instances can share their knowledge through a micro-economic model [89], that would weight the recommandations of experienced instances, assuming their age reflects an optimal configuration.

Beyond the peer level, the adoption of algorithms inspired from evolutionary computations, such as genetic programming, at an upper level of decision can offer an opportunity to test and compare several alternative decisions for a given symptom and to observe how does the crowd of applications evolves. By introducing some diversity within this population of applications, some instances will not only provide a satisfying QoS, but will also become naturally resilient to unforeseen situations.

3.3.3. Smart reconfigurations in the large

Any decision taken by the crowd requires to propagate back to and then operated by the software instances. While simplest decisions tend to impact software instances located on a single host (*e.g.*, laptop, smartphone), this process can also exhibit more complex reconfiguration scenarios that require the orchestration of various actions that have to be safely coordinated across a large number of hosts. While it is generally acknowledged that centralized approaches raise scalability issues, we think that self-optimization should investigate different reconfiguration strategies to propagate and apply the appropriate actions. The investigation of such strategies can be addressed in two steps: the consideration of *scalable data propagation protocols* and the identification of *smart reconfiguration mechanisms*.

With regard to the challenge of scalable data propagation protocols, we think that research opportunities encompass not only the exploitation of gossip-based protocols [94], but also the adoption of publish/subscribe abstractions [101] in order to decouple the decision process from the reconfiguration. The fundamental issue here is the definition of a communication substrate that can accomodate the propagation of decisions with relaxed properties, inspired by *Delay Tolerant Networks* (DTN), in order to reach weakly connected software instances. We believe that the adoption of asynchronous communication protocols can provide the sustainable foundations for addressing various execution environments including harsh environments, such as developing countries, which suffer from a partial connectivity to the network. Additionally, we are interested in developing the principle of *social networks of applications* in order to seamlessly group and organize software instances according to their similarities and acquaintances. The underlying idea is that grouping application instances can contribute to the identification of optimization profiles not only contributing to the monitoring layer, but also interested in similar reconfigurations. Social networks of applications can contribute to the anticipation of reconfigurations by exploiting the symptoms of similar applications to improve the performance of others before that problems actually happen.

With regard to the challenge of smart reconfiguration mechanisms, we are interested in building on our established experience of adaptive middleware [8] in order to investigate novel approaches to efficient application reconfigurations. In particular, we are interested in adopting seamless micro-updates and micro-reboot technics to provide in-situ reconfiguration of pieces of software. Additionally, the provision of safe and secured reconfiguration mechanisms is clearly a key issue that requires to be carefully addressed in order to avoid malicious exploitation of dynamic reconfiguration mechanisms against the software itself. In this area, although some reconfiguration mechanisms integrate transaction models [102], most of them are restricted to local reconfigurations, without providing any support for executing distributed reconfiguration transactions. Additionally, none of the approached published in the literature include security mechanisms to preserve from unauthorized or malicious reconfigurations.

4. Application Domains

4.1. Introduction

Although our research is general enough to be applied to many application domains, we currently focus on applications and distributed services for the retail industry and for the digital home. These two application domains are supported by a strong expertise in mobile computing and in cloud computing that are the two main target environments on which our research prototypes are build, for which we are recognized, and for which we have already established strong collaborations with the industrial ecosystem.

4.2. Distributed software services for the retail industry

This application domain is developed in relation with the PICOM (*Pôle de compétitivité Industries du Commerce*) cluster. We have established strong collaborations with local companies in the context of former funded projects, such as Macchiato, which focused on the development of a new generation of mobile computing platforms for e-commerce. We are also involved in the Datalyse and OCCIware funded projects that define cloud computing environments with applications for the retail industry. Finally, our activities in terms of crowdsensing and data gathering on mobile devices with the APISENSE® platform share also applications for the retail industry.

4.3. Distributed software services for the digital home

We are developing new middleware solutions for the digital home, in particular through our long standing collaboration with Orange Labs. We are especially interested in developing energy management and saving solutions with the POWERAPI software library for distributed environments such the ones that equip digital homes. We are also working to bridge the gap between distributed services hosted on home gateways and distributed services hosted on the cloud to be able to smoothly transition between both environments. This work is especially conducted with the SALOON platform.

5. New Software and Platforms

5.1. APISENSE®

Participants: Clive Ferret-Canape, Julien Duribreux, María Gómez Lacruz, Nicolas Haderer, Christophe Ribeiro, Romain Rouvoy [correspondant], Antoine Veuiller.

In 2014, new developments have been made on our APISENSE® distributed crowdsensing platform. APISENSE® now builds on a distributed infrastructure hosted in the Cloud that can better cope with scalability issues in the number of experiments, users, and volume of data to be collected in the wild. Data collected by participants can be exposed to applications and stakeholders via an Open Data API, which provides the ability to build realtime web applications from crowdsourced datasets. The APISENSE® mobile app, named BEE, can be freely downloaded from the Google Play Store. APISENSE® is part of the results of the PhD thesis of Nicolas Haderer [12] that was defended in November 2014. In 2014, APISENSE® has also been at the core of an industrial transfer action that aims at creating a spin-off company. The project is managed by Christophe Ribeiro and Romain Rouvoy. The project has been accepted (so-called qualification) in 2014 by the Inria investment fund IT-Translation. The project in supported by Direction Transfert & Innovation which will fund in 2015 the 1-year engineer contract of Christophe Ribeiro for maturing the project.

APISENSE® is a distributed platform dedicated to crowdsensing activities. Crowdsensing intends to leverage mobile devices to seamlessly collect valuable dataset for different categories of stakeholders. APISENSE® intends to be used in a wide variety of scientific and industrial domains, including network quality monitoring, social behavior analysis, epidemy predictions, emergency crisis support, open maps initiatives, debugging of applications in the wide. APISENSE® is composed of BEE.HIVE delivered as a *Platform-as-a-Service* (PaaS) to the stakeholders who can pilot and customize their own crowdsensing environment [108], and *Bee.mob* supporting participants with a mobile application to control the sensors to be shared with the rest of the world [96], [97]. The platform is used by the Metroscope consortium, an Internet scientific observatory initiative supported by Inria.

APISENSE® is at the core of the Inria ADT Focus CrowdLab project (see Section 8.2).

Web site: http://www.apisense.fr. Registered with the APP (*Agence pour la Protection des Programmes*) under reference IDDN.FR.001.080006.000.S.P.2013.000.10000 is pending. License: Proprietary.

5.2. FraSCAti

Participants: Philippe Merle [correspondant], Fawaz Paraiso, Romain Rouvoy, Lionel Seinturier.

The novelty of 2014 consists in the development of the SOCLOUD platform for distributed multi-cloud systems. This platform has been defined in the context of the PhD thesis of Fawaz Paraiso [15] that was defended in June 2014. SOCLOUD is built on top of our existing FRASCATI platform. SOCLOUD enables to deploy, execute and manage an application that spans on several different cloud systems.

FRASCATI is a service-oriented component-based middleware platform implementing OASIS *Service Component Architecture* (SCA) specifications. The main originality of FRASCATI is to bring FRACTAL-based reflectivity to SCA, *i.e.*, any FRASCATI software component is equipped with both the SOA capabilities brought by SCA and the reflective capabilities (*i.e.*, introspection and reconfiguration) brought by FRACTAL. Various micro-benchmarks have shown that FRASCATI reflectivity is achieved without hindering its performance relative to the de facto reference SCA implementation, *i.e.*, Apache Tuscany. Non-functional concerns (logging, transaction, security, etc.), so-called intents in SCA terms, are also programmed as FRASCATI components and are (un)woven on business components dynamically at runtime, this is based on aspect-oriented concepts defined in FAC [110]. FRASCATI supports various implementation technologies (SCA Composite, Java, WS-BPEL, Spring Framework, OSGi, Fractal ADL, native C library, Apache Velocity templates, and seven scripting languages as BeanShell, FScript, Groovy, JavaScript, JRuby, Jython, XQuery) for programming services or integrating legacy code, various binding protocols (SOAP, REST, JSON-RPC, UPnP, HTTP servlets, Java RMI, JMS, JGroups) and interface definition languages (WSDL, Java, WADL) for interoperating with existing services. FRASCATI provides management tools like standalone, Web-based, and JMX-based graphical consoles and a dedicated scripting language for reconfiguring SCA applications. The whole FRASCATI platform is itself built as a set of reflective SCA components.

Inria Evaluation Committee Criteria for Software Self-Assessment: A-4-up, SO-4, SM-4-up, EM-3-up, SDL-4-up, DA-4, CD-4, MS-4, TPM-4. FRASCATI is a project of the OW2 consortium for open-source middleware. Web site: http://frascati.ow2.org. 292 Kloc (mainly Java). Registered with the APP (Agence pour la Protection des Programmes) under reference FR.001.050017.000.S.P.2010.000.10000. License: LGPL. Embedded into several industrial software systems: EasySOA, Petals Link EasyViper, EasyBPEL, EasyESB, OW2 PEtALS, OW2 Scarbo. Various demonstrators built during funded projects: ANR SCOrWare, FP7 SOA4All, ANR ITEmIS, ANR SALTY, ANR SocEDA, FUI Macchiato, FUI EasySOA, ADT Galaxy and ADT Adapt. Main publications: [117], [116], [103], [104], [93], [92].

5.3. PowerAPI

Participants: Maxime Colmant, Loïc Huertas, Adel Noureddine, Romain Rouvoy [correspondant].

In 2014, new developments have been made on our POWERAPI library for monitoring energy in software systems. POWERAPI now includes an accurate power model, which supports both DFVS, hyper threads and turbo boost features of modern processors. This model has been assessed on acknowledged benchmarks (PARSEC, SPEC CPU, SPECjbb) and is used as a basis to estimate the power consumption of applications running in virtualised environments. Finally, POWERAPI has evolved towards a modular toolkit that can be used to build software-defined power meters supporting a wide range of input sources (*e.g.*, hardware performance counters, RAPL, PowerSpy). POWERAPI is part of the results of the PhD thesis of Adel Noureddine [14] that was defended in March 2014.

POWERAPI is a Scala-based library for monitoring energy in software systems. It is based on a modular and asynchronous event-driven architecture using the Akka library. POWERAPI differs from existing energy process-level monitoring tool in its pure software, fully customizable and modular aspect which let users precisely define what they want to monitor, without plugging any external device. POWERAPI offers an API which can be used to express requests about energy spent by a process, following its hardware resource utilization (in terms of CPU, memory, disk, network, etc.). Its applications cover energy-driven benchmarking [105], [88], [86], [87], energy hotspots and bugs detection [106], [107], and real-time distributed system monitoring.

POWERAPI is at the core of the Inria ADT eSurgeon project (see Section 8.2).

Web site: http://www.powerapi.org. Registered with the APP (Agence pour la Protection des Programmes) under reference IDDN.FR.001.400015.000.S.P.2012.000.10000. License: AGPL.

5.4. Saloon

Participants: Laurence Duchien, Clément Quinton, Daniel Romero Acero, Lionel Seinturier [correspondant].

SALOON is a framework for the selection and configuration of Cloud providers according to application requirements. The framework enables the specification of such requirements by defining ontologies. Each ontology provides a unified vision of provider offers in terms of frameworks, databases, languages, application servers and computational resources (i.e., memory, storage and CPU frequency). Furthermore, each provider is related to a Feature Model (FM) with attributes and cardinalities, which captures its capabilities. By combining the ontology and FMs, the framework is able to match application requirements with provider capabilities and select a suitable one. Specific scripts to the selected provider are generated in order to enable its configuration.

SALOON is the result of the PhD thesis of Clément Quinton [16] that was defended in October 2014. SALOON is partially developed in the context of the FP7 PaaSage project (see Section 8.3).

Registered with the APP (Agence pour la Protection des Programmes) under reference IDDN.FR.001.300002.000.S.P.2014.000.10800.

5.5. Spoon

Participants: Martin Monperrus [correspondant], Gérard Paligot, Nicolas Petitprez.

In 2014, SPOON has been at the core of an industrial transfer action that aims at creating a spin-off company. The project is managed by Nicolas Petitprez and Martin Monperrus. The project has been accepted (so-called qualification) in 2014 by the Inria investment fund IT-Translation. The project in supported by Direction Transfert & Innovation which will fund in 2015 the 1-year engineer contract of Nicolas Petitprez for maturing the project. As an open source project Spoon has attracted new contributors in 2014. The Spoon development team is now composed of 8 active members, including 4 that are not at all related to Inria. Second, Spoon now supports analyzing and transforming Java 7 code, which is the now the dominant version of Java. Third, Spoon is the technical foundation of five important papers published in 2014. To sum up, year 2014 was a major year for warming up the Spoon project. Thanks to the support of Inria through the ADT, year 2015 is expected to be as vibrant and rich.

SPOON is a library for analyzing and transforming Java source code [76] [109]. SPOON provides a core API and associated tools for static analysis and generative programming within the Java 5+ environment. SPOON must be seen as a basis to ensure Software Quality through code validation and generation. It can be used in the software development process during the validation phase, as well as for engineering and re-engineering software. The first key point of SPOON is to provide a well-typed and comprehensive AST API which is designed to facilitate analysis and transformation work for programmers. Scanners and processors allow the programmer to implement various program traversal strategies on the Java program. Also, the program representation is built with a well-known and well-tested open source Java compiler: the Eclipse JDT compiler, which ensures the support of the latest Java features. The second key point of SPOON is to provide a pure Java API to specify program transformations using a well-typed generative programming technique (called Spoon Templates). By using well-typed templates, SPOON makes programming of transformations easier and safer for the end-user programmers.

SPOON is at the core of the Inria ADT Spoon3R project (see Section 8.1).

Web site: http://spoon.gforge.inria.fr. Registered with the APP (Agence pour la Protection des Programmes) under reference IDDN.FR.001.070037.000.S.P.2007.000.10600. License: CeCILL-C.

6. New Results

6.1. Highlights of the Year

In 2014, we are proud to have organized the 17th ACM SIGSOFT International Conference on Component-Based Software Engineering and Software Architecture (CompArch) that has been held in Lille from 30 June to 3 July 2014.

CompArch is the main conference of the ACM SIGSOFT group on software architectures and software components. The conference is held alternatively in North America and in Europe. The 17th edition has been held this year in France for the first time. The conference brings together about 100 researchers from the academia and the industry.

6.2. Distributed Context Monitoring

In 2014, we obtained some new results in the area of distributed context monitoring solutions to support the development of self-optimising software systems. Context monitoring has emerged as a key capability in various domains to connect software systems to the underlying hardware platform or to the physical world (in the case of ubiquitous systems). In particular, we have investigated to the capability of inferring high-level contextual situations from a large volume of raw data collected from a single device or in the wild. Both hardware (*e.g.*, accelerometer) or software (*e.g.*, performance counters) sensors tend to continuously produce raw data that a context monitoring solution has to quickly filter, process, and convert it into information that can be used by an application or understood by a user.

As a result of the PhD thesis of Adel Noureddine [14], defended in March 2014, we have developed a middleware toolkit to support *in-depth context monitoring* in the domain of green computing. In particular, we introduce a software library, named POWERAPI, that can estimate the power consumption in real-time at various granularities of software: from system processes to code methods (see Section 5.3). This non-invasive solution provides accurate insights on energy hotspots of software and can be used to derive the energy profile of any software library, thus guiding the developers in optimising the energy consumption of their developments.

As a result of the PhD thesis of Nicolas Haderer [12], defended in November 2014, we have contributed to the development of a middleware platform to support *in-breadth context monitoring* in the area of mobile computing. In particular, we promote the distributed middleware solution APISENSE® as an efficient approach to deploy mobile crowd-sensing tasks across a large population of volunteer participants (see Section 5.1). In particular, APISENSE® includes a task orchestration algorithm that preserves the privacy and the battery of sensing devices, while maintaining specific sensing coverage objectives (including time and space dimensions). The server-side infrastructure of APISENSE® is generated from a dedicated software product line, while the implementation is based on the FRASCATI platform (see Section 5.2).

6.3. Design and Runtime Support for Cloud Computing

In 2014, we obtained some new results in the domain both of the design and the runtime support of distributed applications for multi-cloud systems. The purpose is to deal with applications that span across several different cloud systems. Several reasons justify such a goal. For example, in order to avoid the so-called vendor lock-in syndrome, cloud application stakeholders need to be able to migrate as easily as possible their assets from one cloud system to another one. Other examples include the possibility of introducing diversify and fault-tolerance by deploying applications on different cloud systems, or hot migrating applications where computing resources are less expensive.

For the design of multi-cloud systems, we proposed a solution based on software product lines (SPL) [90] and ontologies. In order to specify the variability of such environments, we extended SPL with attributes, cardinalities, and constraints. In order to enable the evolution of these environments, we provided an automated support for maintaining the consistency based on constraint programming. Finally, we proposed an ontology based approach to bridge the gap between the concepts and artefacts defined by different cloud systems. This global solution is the result of the PhD thesis of Clément Quinton [16] that was defended in October 2014, and has been partially supported by the FP7 PaaSage project (see Section 8.3).

For the runtime support of multi cloud systems, we proposed the SOCLOUD platform. This solution enables to deploy, execute and manage an application that spans on several different cloud systems. SOCLOUD tackles the challenges of portability, provisioning, elasticity, and high availability. SOCLOUD defines a component-based and service-oriented architecture that provides an unified view of a set of cloud systems. SOCLOUD is the result of the PhD thesis of Fawaz Paraiso [15] that was defended in June 2014. SOCLOUD is implemented on top of the FRASCATI platform (see Section 5.2).

6.4. Extraction and Analysis of Knowledge for Automatic Software Repair

Automated software repair aims at assisting developers in order to improve the quality of software systems, for example by recommending some repair actions to fix bugs. Matias Martinez has presented in his PhD thesis [13] that was defended in June 2014, new results in this domain. These results aim at reducing the search space when repairing a software system. The solution relies on two techniques. The first one consists in building change models learnt from repairs performed by other developers. These repairs are mined from existing software repositories of open source projects, and analysed based on their types and frequencies. The second proposed technique is based on the inherent redundancy of code patterns. The assumption is that the probability that the repair code for a particular kind of defect is already present in the software system under study is high. We then take advantage of this inherent redundancy to reduce the search space when looking for repair actions.

7. Bilateral Contracts and Grants with Industry

7.1. ip-label

Participants: Nicolas Haderer, Christophe Ribeiro, Romain Rouvoy [correspondant].

A software exploitation licence of the APISENSE® crowdsensing platform has been sold to the ip-label company. They use this platform as a solution to monitor the quality of the GSM signal in the wild. The objective is to provide developers and stakeholders with a feedback on the quality of experience of GSM connection depending on their location.

7.2. Microsoft Windows Azure for Research Grant

Participants: Nicolas Haderer, Christophe Ribeiro, Romain Rouvoy [correspondant].

The research program associated with this grant consists in leveraging the APISENSE® crowdsensing platform to support the real-time processing of "big" datasets collected in the physical world by a "large" crowd of smartphones. Examples of case studies covered in this area include the automatic inference of roadmaps, the continuous cartography of network coverage quality, or even the detection and the dynamic analysis of earthquakes. However, the unpredictable volume of data to be collected in the wild requires the adoption of elastic computation models and infrastructures to continuously provision the processing capabilities to fit uploads of information reports.

The grant takes the form of virtual credits for accessing the Microsoft Azure cloud computing platform.

7.3. Orange Labs

Participants: Laurence Duchien [correspondant], Amal Tahri.

This collaboration aims at bridging the gap between home networks and cloud environments for the design, the provisionning and the administration of distributed services. The purpose is to define solutions, essentially software design tools and runtime infrastructures, for the seamless migration of distributed applications and services between home networks and cloud environments. The envisioned approach is based on the research activities that we are conducting in the domain of software product lines.

This collaboration is conducted in the context of the ongoing PhD thesis of Amal Tahri.

8. Partnerships and Cooperations

8.1. Regional Initiatives

8.1.1. ADT eSurgeon

Participants: Maxime Colmant, Loïc Huertas, Romain Rouvoy [correspondant].

ADT eSurgeon (2013–15) is a technology development initiative supported by the Inria Lille - Nord Europe Center that aims at supporting the development of the POWERAPI software library (see Section 5.3) for measuring and monitoring the energy consumption of middleware and software systems.

8.1.2. ADT Spoon3R

Participants: Gérard Paligot, Martin Monperrus [correspondant].

ADT Spoon3R (2014–16) is a technology development initiative supported by the Inria Lille - Nord Europe Center that aims at supporting the development of the SPOON software library. (see Section 5.5) Spoon3R aims at extending SPOON with the features defined in the context of our research activities on automated software repair.

8.1.3. North European Lab SOCS

Participants: María Gómez Lacruz, Nicolas Haderer, Christophe Ribeiro, Romain Rouvoy [correspondant], Lionel Seinturier.

North European Lab SOCS (2013–15) is an international initiative supported by the Inria Lille - Nord Europe Center that takes place in the context of a well-established collaboration between Inria and *Universitetet i Oslo* (UiO) initiated in 2008. SOCS (Self-Optimization of Cyber-physical Systems) focuses on the self-optimization issues in cyber-physical systems. Cyber-Physical Systems (CPS) are complex systems-of-systems that blend hardware and software to fulfill specific missions. However, traditional CPS are statically configured to achieve predefined goals, which not only limit their sharing and their reuse, but also hinder their sustainability. We believe that this waste of resources stems from the lack of agility of CPS to adapt to change in their environment or objectives. The SOCS Inria Lab takes advantage of the technologies developed as part of the APISENSE® crowdsensing platform (see Section 5.1) to leverage the development of agile CPS.

8.1.4. LEDA

Participant: Philippe Merle [correspondant].

LEDA (2013–16) Laboratoire d'Expérimentation et de Démonstrations Ambiantes is a demonstration space allocated by the Inria Lille - Nord Europe Center whose goal is to show the scientific results of the Spirals team in the domains of distributed systems, adaptable middleware, software product lines, green computing, and ambiant computing. These results are illustrated around the scenario of a mock digital home.

8.2. National Initiatives

8.2.1. ANR

8.2.1.1. ANR MOANO

Participant: Laurence Duchien [correspondant].

MOANO (Models & Tools for Pervasive Applications focusing on Territory Discovery) is a 46-month project of the ANR CONTINT program which started in December 2010. The partners are LIUPPA/University of Pau and Pays de L'Adour, University of Toulouse/IRIT, University of Grenoble/LIG, University Lille 1/LIFL. While going through a territory, mobile users often encounter problems with their handheld computers/mobiles. Some locally stored data become useless or unnecessary whereas other data is not included in the handheld computer. Some software components, part of the whole applications can become unnecessary to process some information or documents that the user did no plan to manage during his mission. In order to answer such difficulties, our project has three operational studies which are i) to enlarge the communication scale, ii) to provide people without computer-science skills with a toolset that will enable them to produce/configure mapping applications to be hosted on their mobile phone and iii) to process all the documents of interest in order to make their spatial and thematic semantics available to mobile users.

8.2.1.2. ANR YourCast

Participants: Laurence Duchien [correspondant], Clément Quinton, Daniel Romero Acero.

YourCast (Software Product Lines for Broadcasting Systems) is a 36-month ANR Emergence project that started in January 2012 and that involves University of Nice Sophia Antipolis, Valorpaca and University Lille 1. The project aims at defining an information broadcasting system by a dedicated software product line which will be used in schools or events, such as gatherings of scouts.

8.2.2. Competitivity Clusters

8.2.2.1. FUI Hermes

Participants: Laurence Duchien, Romain Rouvoy, Lionel Seinturier [correspondant].

Hermes is a 41-month project funded by FUI and labelized by the PICOM (Pôle des Industries du COMmerce) competitivity cluster which has started in August 2012. The goal of the project is to define a modular and context-aware marketing platform for the retail industry. The focus is put on the interactions with customers in order to extract and mine relevant informations related to shopping habits, and on a multi-device, cross-canal, approach to better match customer usages.

8.2.3. Programme Investissement d'Avenir (PIA)

8.2.3.1. PIA Datalyse

Participants: Filip Krikava, Romain Rouvoy, Lionel Seinturier [correspondant], Bo Zhang.

Datalyse is a 36-month project of the Programme Investissement d'Avenir Cloud Computing 3rd call for projects. The project started in May 2013. The partners are Business & Decision Eolas, Groupement des Mousquetaires, Université Grenoble 1, Université Lille 1, Inria, Université Montpellier 2. The project aims at defining an elastic cloud computing infrastructure for processing big volumes of data. The originality of the project is to consider jointly data generated by users and by the infrastructure, and to correlate data at these two levels.

8.2.3.2. PIA OCCIware

Participants: Romain Rouvoy, Philippe Merle [correspondant], Lionel Seinturier.

OCCIware is a 36-month project of the Programme Investissement d'Avenir Cloud Computing and Big Data 4th call for projects. The project started in Decembre 2014. The partners are Open Wide (leader), ActiveEon SA, CSRT, Institut Mines-Télécom/Télécom SudParis, Inria, Linagora GSO, Obeo, OW2 Consortium, Pôle Numérique, and Université Joseph Fourier - Grenoble. The project aims at defining a formal framework for managing every digital resources in the clouds, based on Open Cloud Computing Interface (OCCI) recommendations from Open Grid Forum (OGF).

8.2.4. Inria National Initiatives

8.2.4.1. Inria ADT AntDroid

Participants: María Gómez Lacruz, Nicolas Haderer, Christophe Ribeiro, Romain Rouvoy [correspondant].

ADT AntDroid (2012–14) is a technology development initiative supported by Inria that aims at pushing the results of Nicolas Haderer PhD thesis [12] into production. AntDroid therefore focuses on deploying and disseminating the APISENSE® crowdsensing platform (see Section 5.1) to the public and to support the users of the platform.

8.2.4.2. Inria ADT Focus CrowdLab

Participants: Clive Ferret-Canape, Julien Duribreux, María Gómez Lacruz, Christophe Ribeiro, Romain Rouvoy [correspondant], Antoine Veuiller.

The purpose of the ADT Focus CrowdLab (2014–2016) is to strengthen the technological part of the Metroscope consortium and to promote the APISENSE® crowdsensing platform (see Section 5.1) as a reference platform fo gathering mobile data within the scientific community. The CrowdLab project focuses on three stringent goals: (1) consolidating the current technological solutions, (2) technical and logistical support of the research activities initiated in different scientific domains, and (3) the improvement of security and anonymity of collected data. In addition to the Metroscope consortium, the Inria research teams participating of the ADT Focus CrowdLab project are: Spirals (coordinator), Madynes, Diana, Muse.

8.3. European Initiatives

8.3.1. FP7 & H2020 Projects

Program: FP7 ICT.

Project acronym: PaaSage.

Project title: Model Based Cloud Platform Upperware.

Duration: October 2012–September 2016.

Coordinator: ERCIM.

Other partners: ERCIM (Fr), SINTEF (No), STFC (UK), U. of Stuttgart (De), Inria (Fr), CETIC (Be), FORTH (El), Be.Wan (Be), EVRY Solutions (No), SysFera (Fr), Flexiant (UK), Lufthansa Systems AG (De), Gesellschaft fur wissenschaftliche Datenverarbeitung mbh Gottingen (De), Automotive Simulation Center Stuttgart (De).

Abstract: Cloud computing is a popular and over-hyped concept in ICT. The concept of infinitely scalable elastic resources changing without complex systems administration and paying only for resources used is attractive. These benefits are not immediately realizable. Within organisation benefits are realizable at considerable cost. IaaS (*Infrastructure-as-a-Service*) public Clouds have different interfaces and conditions of use thus for an organisation to "scale out" requires considerable investment using skilled technical staff. The business need is to allow organisations to "scale out" from their private Cloud to public Clouds without a technical chasm between. This cannot easily be achieved. Aligned with the EU strategic direction of an open market for services, SOA (*Service-Oriented architecture*) offers a way to virtualize across heterogeneous public Clouds and organizational private Clouds. It opens a market for European SMEs to provide services to be utilized (and paid for) by business applications and for all organisations to benefit from a catalogue of services that can be used across the environment. PaaSage will deliver an open and integrated platform, to support both deployment and design of Cloud applications, together with an accompanying methodology that allows model-based development, configuration, optimisation, and deployment of existing and new applications independently of the existing underlying Cloud infrastructures. Specifically it will deliver an IDE (*Integrated Development Environment*) incorporating modules for design time and execution time optimisation of applications specified in the Cloud Modeling Language (Cloud ML), execution-level mappers and interfaces and a metadata database.

Participants: Laurence Duchien, Clément Quinton, Daniel Romero Acero [correspondant], Romain Rouvoy, Lionel Seinturier.

Program: FP7 FET.

Project acronym: DIVERSIFY.

Project title: More software diversity. More adaptivity in CAS.

Duration: 36 months (2013-16).

Coordinator: Inria.

Other partners: SINTEF (Norway), Trinity College Dublin (Ireland), University of Rennes 1 (France).

Abstract: DIVERSIFY explores diversity as the foundation for a novel software design principle and increased adaptive capacities in CASs (*Collective Adaptive Systems*). Higher levels of diversity in the system provide a pool of software solutions that can eventually be used to adapt to unforeseen situations at design time. The scientific development of DIVERSIFY is based on a strong analogy with ecological systems, biodiversity, and evolutionary ecology. DIVERSIFY brings together researchers from the domains of software-intensive distributed systems and ecology in order to translate ecological concepts and processes into software design principles.

Participants: Martin Monperrus [correspondant], Matias Martinez.

8.4. International Initiatives

8.4.1. Inria Associate Teams

Title: Service-Oriented Architecture anti-patterns in Mobile and Cloud Applications (SOMCA).

Inria principal investigator: Romain Rouvoy.

International Partner (Institution - Laboratory - Researcher):

Université du Québec à Montréal (Canada) - LATECE Laboratory

Duration: 2014–2016.

See also: http://seas.ifi.uio.no.

The long-term goal of this research program is to propose a novel and innovative methodology embodied in a software platform, to support the runtime detection and correction of anti-patterns in large-scale service-oriented distributed systems in order to continuously optimize their quality of service. One originality of this program lies in the dynamic nature of service-oriented environments, the application on emerging frameworks for embedded and distributed systems (e.g., Android/iOS for mobile devices, PaaS/SaaS for Cloud environments), and in particular mobile systems interacting with remote services hosted on the Cloud. To achieve this goal, we propose to follow a three-step methodology targeting three objectives: (1) Identify and specify service-oriented anti-patterns, (2) Develop an approach to detect automatically, at runtime, service-oriented anti-patterns, (3) Develop an approach to suggest refactorings and automatically, at runtime, correct service-oriented anti-patterns. The ongoing PhD thesis of Geoffrey Hecht, in co-supervision between Montréal and Lille, is part of this associated team.

Participants: Laurence Duchien, María Gómez Lacruz, Geoffrey Hecht, Philippe Merle, Romain Rouvoy [correspondant], Lionel Seinturier.

8.4.2. Inria International Partners

8.4.2.1. Declared Inria International Partners

8.4.2.1.1. University of Los Andes, Bogota, Colombia

We have a long term collaboration since 2005 with this university. Over the years, four PhD thesis (Carlos Noguera, Carlos Parra, Daniel Romero Acero, Gabriel Tamura) have been defended in our team with students who obtained their MSc in this university. The first three were full French PhD, whereas the last one was a co-tutelle with this university. Professor Rubby Casallas from University of Los Andes is frequently visiting our team. The most recently defended PhD thesis, that of Gabriel Tamura, deals with QoS (quality-of-service) contract preservation in distributed service-oriented architectures. A formal theory to perform, in a safe way, the process of self-adaptation in response to quality-of-service (QoS) contracts violation has been proposed. The results have been published in [121], [119] and in the PhD thesis document itself [118].
Participants: Laurence Duchien [correspondant], Clément Quinton, Daniel Romero Acero, Romain Rouvoy, Lionel Seinturier.

8.4.2.1.2. University of Oslo, Norway

The scientific collaboration with this international partner deals with complex distributed systems that have to seamlessly adapt to a wide variety of deployment targets. This is due to the fact that developers cannot anticipate all the runtime conditions under which these systems are immersed. A major challenge for these software systems is to develop their capability to continuously reason about themselves and to take appropriate decisions and actions on the optimizations they can apply to improve themselves. This challenge encompasses research contributions in different areas, from environmental monitoring to real-time symptoms diagnosis, to automated decision making. The collaboration has been supported by the SEAS Inria associated team (2012-14).
Participants: María Gómez Lacruz, Nicolas Haderer, Daniel Romero Acero, Romain Rouvoy [correspondant], Lionel Seinturier.

8.4.3. Participation In Other International Programs

8.4.3.1. OW2

Participants: Philippe Merle [correspondant], Fawaz Paraiso, Romain Rouvoy, Lionel Seinturier.

OW2, previously ObjectWeb, is an international consortium to promote high quality open source middleware. The vision of OW2 is that of a set of components which can be assembled to offer high-quality middleware systems. We are members of this consortium since 2002. Philippe Merle is the leader of both FRACTAL and FRASCATI projects, which are hosted by this consortium. Philippe Merle and Lionel Seinturier are members of the Technology Council of OW2.

8.4.3.2. ERCIM Working Group on Software Evolution

Participant: Laurence Duchien [correspondant].

The Working Group (WG) on Software Evolution is one of the working groups supported by ERCIM. The main goal of the WG is to identify a set of formally-founded techniques and associated tools to support software developers with the common problems they encounter when evolving large and complex software systems. With this initiative, the WG plans to become a Virtual European Research and Training Centre on Software Evolution.

8.5. International Research Visitors

8.5.1. Visits of International Scientists

Participant: Earl Barr.

Subject: Anti-fragility of Software Systems

Date: June 2014

Institution: University College London (UK)

8.5.1.1. Internships

Sebastian Lamelas Marcote

Subject: Automatic Software Repair

Date: from May 2014 until Oct 2014

Institution: University of Buenos Aires (Argentina)

Carolina Valdez Gandara

Subject: SmartGate: An Android-based Gateway for the Internet of Things

Date: from May 2014 until Nov 2014

Institution: University of Central Buenos Aires (Argentina)

Mohamed Lamine Berkane

Subject: Advanced Modularity Concepts in Distributed Applications

Date: from Jan 2014 until Sep 2014

Institution: University Constantine 2 (Algeria)

9. Dissemination

9.1. Promoting Scientific Activities

9.1.1. Scientific events organisation

9.1.1.1. General chair, Scientific chair

Martin Monperrus has been General Chair of the 1st European Open Symposium on Empirical Software Engineering (EOSESE'2014), organized in Lille on June 30 2014.

Lionel Seinturier has been General Chair of the 17th ACM SIGSOFT International Conference on Component-Based Software Engineering and Software Architecture (CompArch) that has been held in Lille from 30 June to 3 July 2014.

9.1.1.2. Member of the organizing committee

Laurence Duchien, **Martin Monperrus** and **Romain Rouvoy** have been members of the organisation committee of the ACM SIGSOFT 17th International Conference on Component-Based Software Engineering and Software Architecture (CompArch) that has been held in Lille from 30 June to 3 July 2014.

9.1.2. Scientific events selection

9.1.2.1. Responsable of the conference program committee

Romain Rouvoy has been Poster Chair of the 15th ACM/IFIP/USENIX International Middleware Conference (Middleware).

Romain Rouvoy is member of the Steering Committee of the IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS).

Lionel Seinturier has been Program Committee co-chair of the 3rd International Workshop on Advanced Information Systems for Enterprises (IWAISE).

9.1.2.2. Member of the conference program committee

Laurence Duchien

International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)

Track on Software Engineering In Practice (SEIP) at International Conference on Software Engineering (ICSE)

International Workshop on Software Engineering Research and Industrial Practices (SER&IPs)

International Track on Adaptive and Reconfigurable Service-oriented and component-based Applications and Architectures (AROSA) at the IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)

International Conference on Software Maintenance and Evolution

European Conference on Software Architecture (ECSA)

International Software Product Line Conference (SPLC)

Philippe Merle

Middleware Workshop on Cloud Brokers

International Workshop on Cloud Security and Cryptography (CloudCrypto)

IEEE INFOCOM Workshop on Cross-Cloud Systems (CrossCloud)

International Workshop on Adaptive and Reflective Middleware (ARM)

Conférence en IngénieriE du Logiciel (CIEL)

Filip Krikava

Transformation Tool Contest (TTC) of the Software Technologies: Applications and Foundations (STAF)

Martin Monperrus

Track on New Ideas and Emerging Results Track (NIER) at the International Conference on Software Engineering (ICSE)

International Symposium on Search Based Software Engineering (SSBSE)

Asia-Pacific Software Engineering Conference (APSEC)

Romain Rouvoy

ACM/IFIP/USENIX Middleware Conference (Middleware)

International Conference on Ambient Systems, Networks and Technologies (ANT)

Track on Dependable and Adaptive Distributed Systems (DADS) of the ACM Symposium on Applied Computing (SAC)

Track on Software Engineering Aspects of Green Computing (SEGC) of the ACM Symposium on Applied Computing (SAC)

International Workshop on Middleware for CyberPhysical Systems (MW4CPS)

International Workshop on Middleware for Next Generation Internet Computing Workshop (MW4NG)

International Workshop on Adaptive and Reflective Middleware (ARM)

International Workshop on Adaptive Services for the Future Internet (WAS4FI)

International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC)

International Workshop on Big Data Management for the Internet-of-Things (BIOT) at COMPSAC

International Workshop on Green in Software Engineering, Green by Software Engineering (GIBSE) at the International Conference on Modularity (Modularity)

French Conference on Systems, Architecture and Parallelism (ComPAS)

Lionel Seinturier

Working IEEE/IFIP Conference on Software Architecture (WICSA)

Track on Model-based development, Components and Services (MOCS) at the Euromicro Conference series on Software Engineering and Advanced Applications (SEAA)

IEEE International Symposium on Software Crowdsourcing (ISCC)

Track on Software Engineering Aspects of Green Computing (SEGC) of the ACM Symposium on Applied Computing (SAC)

International Workshop on Middleware for CyberPhysical Systems (MW4CPS)

IEEE International Conference on Service Oriented Computing & Applications (SOCA)

Workshop on Distributed Architecture modeling for Novel component based Embedded systems (DANCE) at the ACM/IFIP/USENIX Middleware Conference (Middleware)

International Workshop on Model-Driven Engineering for Component-Based Software System (ModComp)

IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS)

Australasian Software Engineering Conference (ASWEC)

International Conference on Ambient Systems, Networks and Technologies (ANT)

International Workshop on Green in Software Engineering, Green by Software Engineering (GIBSE) at the International Conference on Modularity (Modularity)

Track on Formal Verification of Service Based Systems (FVSBS) at the IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises (WETICE)

International Symposium on Software Engineering for Adaptive and Self-Managing Systems (SEAMS)

9.1.2.3. Reviewer

Philippe Merle

European Conference on Software Architecture (ECSA)

Track on Software Engineering In Practice (SEIP) at the International Conference on Software Engineering (ICSE)

Jifeng Xuan

IEEE International Conference on Software Engineering (ICSE)
Asia-Pacific Software Engineering Conference (APSEC)
International Conference on Quality Software (QSIC)
IEEE International Conference on Software Maintenance and Evolution (ICSME)

9.1.3. Journal

9.1.3.1. Member of the editorial board

Laurence Duchien is member of the editorial board of Lavoisier Technique et Science Informatiques (TSI).

Lionel Seinturier is editor for software engineering of the ISTE-Wiley Computer Science and Information Technology book collection. He has been co-editor of the special issue on "Greening Distributed Systems" of the Springer Journal of Internet Services and Applications (JISA).

9.1.3.2. Reviewer

Laurence Duchien: Elsevier Future Generation Computer Systems, ACM Transactions on Autonomous and Adaptive Systems, IEEE Software.

Philippe Merle: Elsevier Journal of Systems and Software (JSS), Springer Journal of Internet Services and Applications (JISA), The Scientific World Journal, Special issue "Ingénierie du logiciel" of Lavoisier Technique et Science Informatiques (TSI).

Martin Monperrus: Elsevier Journal of Systems and Software (JSS), IEEE Software, IEEE Transactions on Software Engineering (TSE), Springer Empirical Software Engineering, Wiley Journal of Software: Evolution and Process.

Romain Rouvoy: IEEE Transactions on Network and Service Management (TSNM), IEEE Software, IEEE Internet of Things Journal, IEEE Communications Magazine, Lavoisier Technique et Science Informatiques (TSI), Journal of Universal Computer Science (JUCS), Journal of Network and Computer Applications (JNCA), Elsevier Computer Communications (COMCOM).

Lionel Seinturier: IEEE Software, Elsevier Science and Computation, The Computer Journal, MDPI Sensors, Springer Software Quality Journal, Elsevier Sustainable Computing, Annals of Telecom, Wiley Software Practice and Experience, Lavoisier TSI.

Jifeng Xuan: Elsevier Science of Computer Programming (SCP).

9.2. Teaching - Supervision - Juries

9.2.1. Teaching

Laurence Duchien teaches at the University of Lille 1, IEEA faculty. She heads the research program in Master of Computer Science at University Lille 1. She heads the Carrières et Emplois service and is referent for the professional insertion in the PhD program in Computer Science at PRES University Lille Nord de France. She is also Director of Doctoral Studies for Computer Science in Doctoral School Engineering Science (SPI) - COMUE Lille Nord de France.

Software Project Management, 50h, Level M2, Master MIAGE,
Design of distributed applications, 42h, Level M1, Master of Computer Science,
Software Product Lines, 8h, Level M2, Master of Computer Science,
Research and Innovation Initiation, 22h, Level M2 IAGL, Master of Computer Science,
Tutoring Internship, 16h, Level M2, Master of Computer Science.

Martin Monperrus teaches at the University of Lille 1, IEEA faculty.

Introduction to programming, 48h, Level L1, Licence of Computer Science,
Object-oriented design, 39h, Level L3, Licence of Computer Science,
Automated software engineering, 40h, Level M2 IAGL, Master of Computer Science.

Romain Rouvoy teaches at the University of Lille 1, IEEA faculty. He heads the IAGL specialty of the Master of Computer Science at the University Lille 1. He supervises the Agil-IT Junior Enterprise.

Conception d'Applications Réparties, 42h, Level M1, Master of Computer Science,

Infrastructures et Frameworks Internet, 32h, Level M2 IAGL, Master of Computer Science,

Innovation & Initiation à la Recherche, 14h, Level M2 IAGL, Master of Computer Science specialty,

Outils pour la Programmation des Logiciels, 12h, Level M2 IAGL, Master of Computer Science,

Suivi de projets, 20h, Level M2, Master of Computer Science,

Suivi d'alternants, 20h, Level M2, Master of Computer Science.

Walter Rudametkin teaches at the University of Lille 1, Polytech engineering school.

GIS4 Programmation par Objets

GIS4 Architectures Logicielles

GIS2A3 (apprentissage) Projet programmation par Objet

IMA2A4 (apprentissage) Conception Modélisation Objet

GBIAAL4 Bases de données

Lionel Seinturier teaches at the University of Lille 1, IEEA faculty.

Conception d'Applications Réparties, 18h, Level M1, Master of Computer Science,

Infrastructures et Frameworks Internet, 6h, Level M2 E-Services and TIIR, Master of Computer Science.

E-learning

: Lionel Seinturier, Infrastructure et frameworks Internet, 10 semaines, plate-forme Corolia formation, Telecom Lille & Université Lille 1, niveau Bac+5, formation continue, 12 inscrits par session

9.2.2. Supervision

Defended HdR: Romain Rouvoy, Contributions to the Autonomy of Ubiquitous Software Systems, University Lille 1, 5 December 2014

Defended PhD: Alexandre Feugas, Une approche agile, fiable et minimale pour le maintien de la qualité de service lors de l'évolution d'applications à base de processus métiers, 8 October 2014, Laurence Duchien & Sébastien Mosser (University Nice Sophia-Antipolis).

Defended PhD: Nicolas Haderer, APISENSE®: a distributed platform for deploying, executing and managing data collection campaigns using smart devices, 5 November 2014, Lionel Seinturier & Romain Rouvoy.

Defended PhD: Matias Martinez, Extraction and Analysis of Knowledge for Automatic Software Repair, 10 October 2014, Laurence Duchien & Martin Monperrus.

Defended PhD: Adel Noureddine, Towards a Better Understanding of the Energy Consumption of Software Systems, 19 March 2014, Lionel Seinturier & Romain Rouvoy.

Defended PhD: Fawaz Paraiso, soCloud : une plateforme multi-nuages distribuée pour la conception, le déploiement et l'exécution d'applications distribuées à large échelle, 18 June 2014, Lionel Seinturier & Philippe Merle.

Defended PhD: Clément Quinton, Cloud Environment Selection and Configuration: A Software Product Lines-Based Approach, 22 October 2014, Laurence Duchien.

In progress PhD: Maxime Colmant, Amélioration de l'efficience énergétique des logiciels dans les systèmes multi-coeurs, October 2013, Lionel Seinturier & Romain Rouvoy.

In progress PhD: Benoit Cornu, Automated Runtime Software Repair, October 2012, Lionel Seinturier & Martin Monperrus.

In progress PhD: Maria Gomez Lacruz, Self-Optimization of Software Systems Driven by Wisdom of the Crowds, October 2013, Lionel Seinturier & Romain Rouvoy.

In progress PhD: Geoffrey Hecht, Auto-optimisation des architectures orientées services : Application aux applications mobiles et Cloud, October 2013, Laurence Duchien & Romain Rouvoy.

In progress PhD: Vincenzo Musco, Etude de la topologie et de l'évolution des graphes logiciels, October 2013, Philippe Preux (Inria SequeL) & Martin Monperrus.

In progress PhD: Marc Sango, Composants logiciels, boucle de contrôle et adaptation pour applications safety critical dans le domaine ferroviaire, October 2012, Laurence Duchien & Christophe Gransart.

In progress PhD: Gustavo Sousa, Towards dynamic software product lines to optimize management and reconfiguration of cloud applications, October 2012, Laurence Duchien & Walter Rudametkin.

In progress PhD: Amal Tahri, Evolution logicielle multi-vues : des réseaux domestiques au Cloud, March 2013, Laurence Duchien.

In progress PhD: Bo Zhang, Elasticité spontanée des services et infrastructures dans le Cloud, October 2013, Lionel Seinturier & Romain Rouvoy.

9.2.3. Juries

Laurence Duchien

Boris Baldassari (University Lille 3), chair

Calvante Hora André (University Lille 1), chair

Koutheir Attouchi (University Paris 6), reviewer

Cédric Teyton (University of Bordeaux), reviewer

Jean-Christophe Bach (University of Lorraine), reviewer

Jonathan Michaux (Telecom ParisTech), reviewer

Hamza Samih (University Rennes 1), reviewer

Alexandre Feugas (University Lille 1), director

Matias Martinez (University Lille 1), director

Clément Quinton (University Lille 1), director

Philippe Merle

Fawaz Paraiso (University Lille 1), co-supervisor

Martin Monperrus

Boris Baldassari (University Lille 3), examiner

Alexandre Bartel (University of Luxembourg), co-supervisor

Matias Martinez (University Lille 1), co-supervisor

Romain Rouvoy

Keling Da (University of Pau et des Pays de l'Ardour), examiner

Adel Noureddine (University Lille 1), co-supervisor

Nicolas Haderer (University Lille 1), co-supervisor

Yunbo Li (University of Nantes), member of the comité de suivi de thèse

Lionel Seinturier

HDR Xavier Bonnaire (University Paris 6), chair

HDR Yvan Peter (University Lille 1), chair

HDR Walid Gaaloul (University Paris 6), reviewer

HDR Nicolas Anquetil (University Lille 1), examiner

HDR Romain Rouvoy (University Lille 1), examiner

Nadia Elouali (University Lille 1), chair

Mohamed Amroune (University of Toulouse), reviewer

Etienne Gandrille (University of Grenoble), reviewer

Soguy Gueye (University of Grenoble), reviewer

Loïc Letondeur (University of Grenoble), reviewer

Julien Richard-Foy (University Rennes 1), reviewer

Nicolas Haderer (University Lille 1), director

Adel Noureddine (University Lille 1), director

Fawaz Paraiso (University Lille 1), director

Nour Assy (Telecom SudParis), member of the comité de suivi de thèse

Anca Iordache (University Rennes 1), member of the comité de suivi de thèse

9.3. Popularization

Lionel Seinturier has participated in October to the event Chercheur itinérant organized by the Inria Lille Nord Europe research center in the context of La Fête de la science. Two classes (3ème and 1ère) have been visited. The theme of the visit was crowdsensing and data gathering from mobile devices.

Walter Rudametkin has given some demonstrations at two major open source events: the Open World Forum in October in Paris, and Fossa in November in Rennes.

Loïc Huertas and **Maxime Colmant** have presented in November the PowerAPI software library (see Section 5.3) at the GreenCodeLab that is a challenge for the design of sustainable software

Loïc Huertas has given a demonstration of the PowerAPI software library (see Section 5.3) at the RIC day in September. The event targets students from computer science master programs and engineering schools.

The team has participated to three Inria organized events in relation with the industry: in January in Rennes for the Imatch event with the software industry (Clément Quinton, Loïc Huertas, Philippe Merle, Romain Rouvoy, Christophe Ribeiro, Daniel Romero Acero, Lionel Seinturier), in November in Rocquencourt for the RII Telecom du futur (Christophe Ribeiro), in November in Lille for the RII Web technologies (Philippe Merle, Christophe Ribeiro, Romain Rouvoy).

10. Bibliography

Major publications by the team in recent years

[1] S. HENSS, M. MONPERRUS, M. MEZINI. *Semi-Automatically Extracting FAQs to Improve Accessibility of Software Development Knowledge*, in "ICSE - 34th International Conference on Software Engineering", Zurich, Suisse, March 2012, pp. 793-803 [*DOI :* 10.1109/ICSE.2012.6227139], http://hal.inria.fr/hal-00681906

[2] N. MOHA, Y.-G. GUÉHÉNEUC, L. DUCHIEN, A.-F. LE MEUR. *DECOR: A Method for the Specification and Detection of Code and Design Smells*, in "IEEE Transactions on Software Engineering", January 2010, vol. 36, n^o 1, pp. 20-36, Rank (CORE) : A+

[3] M. MONPERRUS, M. EICHBERG, E. TEKES, M. MEZINI. *What Should Developers Be Aware Of? An Empirical Study on the Directives of API Documentation*, in "Empirical Software Engineering", December 2011, vol. 17, n° 6, pp. 703-737 [*DOI :* 10.1007/S10664-011-9186-4], http://hal.inria.fr/hal-00702183

[4] M. MONPERRUS, M. MEZINI. *Detecting Missing Method Calls as Violations of the Majority Rule*, in "ACM Transactions on Software Engineering and Methodology", 2013, vol. 22, n° 1, pp. 1-25 [*DOI :* 10.1145/2430536.2430541], http://hal.archives-ouvertes.fr/hal-00702196

[5] A. NOUREDDINE, A. BOURDON, R. ROUVOY, L. SEINTURIER. *Runtime Monitoring of Software Energy Hotspots*, in "ASE - The 27th IEEE/ACM International Conference on Automated Software Engineering - 2012", Essen, Germany, September 2012, pp. 160-169 [*DOI :* 10.1145/2351676.2351699], http://hal.inria.fr/hal-00715331

[6] F. PARAISO, P. MERLE, L. SEINTURIER. *soCloud: A service-oriented component-based PaaS for managing portability, provisioning, elasticity, and high availability across multiple clouds*, in "Springer Computing", August 2014 [*DOI :* 10.1007/S00607-014-0421-X], https://hal.inria.fr/hal-01019385

[7] C. PARRA, X. BLANC, A. CLEVE, L. DUCHIEN. *Unifying design and runtime software adaptation using aspect models*, in "Science of Computer Programming", January 2011, vol. 76, n° 12, pp. 1247-1260 [*DOI :* 10.1016/J.PHYSLETB.2003.10.071], http://hal.inria.fr/inria-00564592

[8] L. SEINTURIER, P. MERLE, R. ROUVOY, D. ROMERO, V. SCHIAVONI, J.-B. STEFANI. *A Component-Based Middleware Platform for Reconfigurable Service-Oriented Architectures*, in "Software: Practice and Experience", May 2012, vol. 42, n° 5, pp. 559-583 [*DOI :* 10.1002/SPE.1077], http://hal.inria.fr/inria-00567442

[9] A. TAHERKORDI, F. LOIRET, R. ROUVOY, F. ELIASSEN. *Optimizing Sensor Network Reprogramming via In-situ Reconfigurable Components*, in "ACM Transactions on Sensor Networks", May 2013, vol. 9, n° 2, pp. 1-37, http://hal.inria.fr/hal-00658748

[10] J. XUAN, M. MONPERRUS. *Test Case Purification for Improving Fault Localization*, in "FSE - 22nd ACM SIGSOFT International Symposium on the Foundations of Software Engineering", Hong Kong, China, November 2014 [*DOI :* 10.1145/2635868.2635906], https://hal.inria.fr/hal-01061911

Publications of the year

Doctoral Dissertations and Habilitation Theses

[11] A. FEUGAS. *An Agile, Reliable and Minimalist Approach to Preserve the QoS of Business-Processes Based Applications during their Evolutions*, Université des Sciences et Technologie de Lille - Lille I, October 2014, https://tel.archives-ouvertes.fr/tel-01073193

[12] N. HADERER. *APISENSE®: a distributed platform for deploying, executing and managing data collection campaigns using smart devices*, Université des Sciences et Technologie de Lille - Lille I, November 2014, https://hal.inria.fr/tel-01087240

[13] M. MARTINEZ. *Extraction and Analysis of Knowledge for Automatic Software Repair*, Université Lille 1, October 2014, https://hal.archives-ouvertes.fr/tel-01078911

[14] A. NOUREDDINE. *Towards a Better Understanding of the Energy Consumption of Software Systems*, Université des Sciences et Technologie de Lille - Lille I, March 2014, https://tel.archives-ouvertes.fr/tel-00961346

[15] F. PARAISO. *socloud: distributed Multi-Cloud Platform for deploying, executing and managing distributed applications*, Université des Sciences et Technologie de Lille - Lille I, June 2014, https://tel.archives-ouvertes.fr/tel-01009918

[16] C. QUINTON. *Cloud Environment Selection and Configuration: A Software Product Lines-Based Approach*, Université Lille 1, October 2014, https://tel.archives-ouvertes.fr/tel-01079956

[17] R. ROUVOY. *Contributions to the Autonomy of Ubiquitous Software Systems*, Université de Lille 1, Sciences et Technologies, December 2014, Habilitation à diriger des recherches, https://tel.archives-ouvertes.fr/tel-01091798

Articles in International Peer-Reviewed Journals

[18] S. ALLIER, O. BARAIS, B. BAUDRY, J. BOURCIER, E. DAUBERT, F. FLEUREY, M. MONPERRUS, H. SONG, M. TRICOIRE. *Multi-tier diversification in Web-based software applications*, in "IEEE Software", 2014, 7 p. , https://hal.archives-ouvertes.fr/hal-01089268

[19] A. BARTEL, J. KLEIN, M. MONPERRUS, Y. LE TRAON. *Static Analysis for Extracting Permission Checks of a Large Scale Framework: The Challenges And Solutions for Analyzing Android*, in "IEEE Transactions on Software Engineering", June 2014, https://hal.inria.fr/hal-01055656

[20] B. CORNU, L. SEINTURIER, M. MONPERRUS. *Exception Handling Analysis and Transformation Using Fault Injection: Study of Resilience Against Unanticipated Exceptions*, in "Information and Software Technology", January 2015, vol. 57, pp. 66-76, https://hal.inria.fr/hal-01062969

[21] M. MONPERRUS, A. MAIA, R. ROUVOY, L. SEINTURIER. *Debugging with the Crowd: A Debug Recommendation System Based on StackOverflow*, in "ERCIM News", 2014, pp. 26-27, https://hal.archives-ouvertes.fr/hal-01064058

[22] A. NOUREDDINE, R. ROUVOY, L. SEINTURIER. *Monitoring Energy Hotspots in Software*, in "Journal of Automated Software Engineering", 2015, pp. 1-42, https://hal.inria.fr/hal-01069142

[23] F. PARAISO, P. MERLE, L. SEINTURIER. *soCloud: A service-oriented component-based PaaS for managing portability, provisioning, elasticity, and high availability across multiple clouds*, in "Springer Computing", August 2014 [*DOI :* 10.1007/S00607-014-0421-X], https://hal.inria.fr/hal-01019385

[24] C. QUINTON, D. ROMERO, L. DUCHIEN. *SALOON: a platform for selecting and configuring cloud environments*, in "Software Practice and Experience", January 2015, forthcoming [*DOI :* 10.1002/SPE.2311], https://hal.inria.fr/hal-01103560

[25] M. SANGO, O. HOINARU, C. GRANSART, L. DUCHIEN. *A Temporal QoS Ontology for ERTMS/ETCS*, in "International Journal of Computer, Information, Systems and Control Engineering", January 2015, vol. 9, n^{o} 1, 7 p. , https://hal.inria.fr/hal-01110240

[26] G. TAMURA, R. CASALLAS, A. CLEVE, L. DUCHIEN. *QoS Contract Preservation through Dynamic Reconfiguration: A Formal Semantics Approach*, in "Science of Computer Programming", 2014, vol. 94, n^{o} 3, 25 p. [*DOI :* 10.1016/J.SCICO.2013.12.003], https://hal.inria.fr/hal-00911844

[27] J. XUAN, H. JIANG, Y. HU, Z. REN, W. ZOU, Z. LUO, X. WU. *Towards Effective Bug Triage with Software Data Reduction Techniques*, in "IEEE Transactions on Knowledge and Data Engineering", May 2014 [*DOI :* 10.1109/TKDE.2014.2324590], https://hal.inria.fr/hal-01018934

International Conferences with Proceedings

[28] E. ABGRALL, S. GOMBAULT, Y. LE TRAON, M. MONPERRUS. *An Empirical Investigation of the Web Browser Attack Surface under Cross-Site Scripting: an Urgent Need for Systematic Security Regression Testing*, in "SECTEST'2014", Cleveland, United States, 2014, https://hal.archives-ouvertes.fr/hal-00979586

[29] B. BAUDRY, S. ALLIER, M. MONPERRUS. *Tailored Source Code Transformations to Synthesize Computationally Diverse Program Variants*, in "ISSTA", United States, 2014, pp. 149-159 [*DOI :* 10.1145/2610384.2610415], https://hal.archives-ouvertes.fr/hal-00938855

[30] B. BAUDRY, M. MONPERRUS, C. MONY, F. CHAUVEL, F. FLEUREY, S. CLARKE. *DIVERSIFY - Ecology-inspired software evolution for diversity emergence*, in "CSMR", Antwerp, Belgium, IEEE, February 2014, pp. 395-398, https://hal.inria.fr/hal-00916281

[31] M. BIAZZINI, M. MONPERRUS, B. BAUDRY. *On Analyzing the Topology of Commit Histories in Decentralized Version Control Systems*, in "ICSME 2014", Victoria, Canada, 2014, pp. 261-270, pages: 10, https://hal.archives-ouvertes.fr/hal-01063789

[32] M. COLMANT, R. ROUVOY, L. SEINTURIER. *Improving the Energy Efficiency of Software Systems for Multi-Core Architectures*, in "Middleware 2014 Doctoral Symposium", Bordeaux, France, December 2014 [*DOI :* 10.1145/2684080.2684081], https://hal.inria.fr/hal-01078822

[33] F. DEMARCO, J. XUAN, D. LE BERRE, M. MONPERRUS. *Automatic Repair of Buggy If Conditions and Missing Preconditions with SMT*, in "CSTVA'2014", Hyderabad, India, 2014, https://hal.archives-ouvertes.fr/hal-00977798

[34] J.-R. FALLERI, F. MORANDAT, X. BLANC, M. MARTINEZ, M. MONPERRUS. *Fine-grained and Accurate Source Code Differencing*, in "ASE 2014", France, 2014, 11 p. , https://hal.archives-ouvertes.fr/hal-01054552

[35] M. GOMEZ, M. MARTINEZ, M. MONPERRUS, R. ROUVOY. *When App Stores Listen to the Crowd to Fight Bugs in the Wild*, in "37th International Conference on Software Engineering (ICSE), track on New Ideas and Emerging Results (NIER)", Firenze, Italy, T. MENZIES, A. ROYCHOUDHURY (editors), IEEE, May 2015, 4 p. , https://hal.inria.fr/hal-01105173

[36] G. HECHT, B. JOSE-SCHEIDT, C. DE FIGUEIREDO, N. MOHA, F. KHOMH. *An Empirical Study of the Impact of Cloud Patterns on Quality of Service (QoS)*, in "IEEE CloudCom 2014", Singapour, Singapore, December 2014, https://hal.inria.fr/hal-01088298

[37] H. HU, H. ZHANG, J. XUAN, W. SUN. *Effective Bug Triage based on Historical Bug-Fix Information*, in "ISSRE - The 25th IEEE International Symposium on Software Reliability Engineering", Naples, Italy, IEEE , November 2014, https://hal.inria.fr/hal-01087444

[38] M. A. JIMÉNEZ, A. VILLOTTA GOMEZ, N. M. VILLEGAS, G. TAMURA, L. DUCHIEN. *A Framework for Automated and Composable Testing of Component-based Services*, in "Maintenance and Evolution of Service-Oriented Systems and Cloud-Based Environments", Victoria BC, Canada, IEEE Computer Society, September 2014, https://hal.inria.fr/hal-01055906

[39] F. KRIKAVA, P. COLLET, R. FRANCE. *ACTRESS: Domain-Specific Modeling of Self-Adaptive Software Architectures*, in "Symposium On Applied Computing", Gyeongju, South Korea, March 2014, https://hal.inria.fr/hal-00951798

[40] F. KRIKAVA, P. COLLET, R. FRANCE. *Manipulating Models Using Internal Domain-Specific Languages*, in "Symposium On Applied Computing", Gyeongju, South Korea, March 2014, https://hal.inria.fr/hal-00951803

[41] F. KRIKAVA, P. COLLET, R. FRANCE. *SIGMA: Scala Internal Domain-Specific Languages for Model Manipulations*, in "MODELS - 17th International Conference on Model Driven Engineering Languages and Systems", Valencia, Spain, September 2014, https://hal.inria.fr/hal-01010339

[42] F. KRIKAVA, P. COLLET. *Solving the TTC'14 FIXML Case Study with SIGMA*, in "Proceedings of the 7th Transformation Tool Contest part of the Software Technologies: Applications and Foundations (STAF 2014) federation of conferences", York, United Kingdom, L. M. ROSE, C. KRAUSE, T. HORN (editors), University of York, United Kingdom, July 2014, https://hal.inria.fr/hal-01015149

[43] F. KRIKAVA, P. COLLET, R. ROUVOY. *Integrating Adaptation Mechanisms Using Control Theory Centric Architecture Models: A Case Study*, in "ICAC - 11th International Conference on Autonomic Computing", Philadelphia, United States, USENIX, June 2014, https://hal.inria.fr/hal-00991114

[44] M. MARTINEZ, W. WEIMER, M. MONPERRUS. *Do the Fix Ingredients Already Exist? An Empirical Inquiry into the Redundancy Assumptions of Program Repair Approaches*, in "ICSE - 36th IEEE International Conference on Software Engineering", Hyderabad, India, June 2014, https://hal.inria.fr/hal-00965410

[45] M. MONPERRUS. *A Critical Review of "Automatic Patch Generation Learned from Human-Written Patches": Essay on the Problem Statement and the Evaluation of Automatic Software Repair*, in "ICSE 2014", India, 2014, pp. 234-242 [*DOI :* 10.1145/2568225.2568324], https://hal.archives-ouvertes.fr/hal-01054549

[46] A. NOUREDDINE, R. ROUVOY, L. SEINTURIER. *Unit Testing of Energy Consumption of Software Libraries*, in "Symposium On Applied Computing", Gyeongju, South Korea, March 2014, pp. 1200-1205 [*DOI :* 10.1145/2554850.2554932], https://hal.inria.fr/hal-00912613

[47] W. F. OUEDRAOGO, F. BIENNIER, P. MERLE. *Contextualised security operation deployment through MDS@run.time architecture*, in "ISC 2014 - Intelligent Service Clouds Workshop at the 12th International Conference on Services Oriented Computing 2014", Paris, France, November 2014, https://hal.archives-ouvertes.fr/hal-01088034

[48] C. QUINTON, A. PLEUSS, D. LE BERRE, L. DUCHIEN, G. BOTTERWECK. *Consistency Checking for the Evolution of Cardinality-based Feature Models*, in "SPLC - 18th International Software Product Line Conference", Florence, Italy, September 2014, https://hal.inria.fr/hal-01054604

[49] C. QUINTON, D. ROMERO, L. DUCHIEN. *Automated Selection and Configuration of Cloud Environments Using Software Product Lines Principles*, in "IEEE CLOUD 2014", Anchorage, United States, June 2014, https://hal.inria.fr/hal-00965836

[50] Z. REN, H. JIANG, J. XUAN, S. ZHANG, Z. LUO. *Learning from evolved next release problem instances*, in "GECCO - Genetic and Evolutionary Computation Conference", Vancouver, BC, Canada, ACM SIGEVO, July 2014, pp. 189 - 190 [*DOI :* 10.1145/2598394.2598427], https://hal.inria.fr/hal-01087436

[51] M. SANGO, L. DUCHIEN, C. GRANSART. *Component-Based Modeling and Observer-Based Verification for Railway Safety-Critical Applications*, in "The 11th International Symposium on Formal Aspects of Component Software", Bertinoro, Italy, September 2014, https://hal.inria.fr/hal-01051754

[52] M. SANGO, C. GRANSART, L. DUCHIEN. *Safety component-based approach and its application to ERTMS/ETCS on-board train control system*, in "TRA2014 Transport Research Arena 2014", Paris, France, April 2014, https://hal.inria.fr/hal-00918907

[53] F. WENDPANGA, F. BIENNIER, P. MERLE. *Optimizing Service Protection with Model Driven Security@run.time*, in "9th International IEEE Symposium on Service-Oriented System Engineering - IEEE SOSE 2015", Redwood City, United States, March 2015, https://hal.archives-ouvertes.fr/hal-01109967

[54] J. XUAN, M. MONPERRUS. *Learning to Combine Multiple Ranking Metrics for Fault Localization*, in "ICSME - 30th International Conference on Software Maintenance and Evolution", Victoria, Canada, September 2014, https://hal.inria.fr/hal-01018935

[55] J. XUAN, M. MONPERRUS. *Test Case Purification for Improving Fault Localization*, in "FSE - 22nd ACM SIGSOFT International Symposium on the Foundations of Software Engineering", Hong Kong, China, November 2014 [*DOI :* 10.1145/2635868.2635906], https://hal.inria.fr/hal-01061911

National Conferences with Proceedings

[56] W. F. OUEDRAOGO, F. BIENNIER, P. MERLE. *Gestion contextualisée de la sécurité : implémentation MDS@Runtime avec FraSCAti*, in "SAR-SSI 2014 - 9ème Conférence sur la Sécurité des Architectures Réseaux et des Systèmes d'Information", Saint-Germain-Au-Mont-d'Or (Lyon), France, May 2014, https://hal.archives-ouvertes.fr/hal-01088013

[57] G. SOUSA, W. RUDAMETKIN, L. DUCHIEN. *Challenges for Automatic Multi-Cloud Configuration*, in "JLDP - Journée Lignes de Produits", Luxembourg, December 2014, https://hal.inria.fr/hal-01093220

Scientific Books (or Scientific Book chapters)

[58] N. HADERER, F. PARAISO, C. RIBEIRO, P. MERLE, R. ROUVOY, L. SEINTURIER. *A Cloud-based Infrastructure for Crowdsourcing Data from Mobile Devices*, in "Cloud-based Software Crowdsourcing", W. WU (editor), Springer, 2015, pp. 255-278, https://hal.inria.fr/hal-01022816

[59] G. WAIGNIER, A.-F. LE MEUR, L. DUCHIEN. *Contrôle de la fiabilité des logiciels à évoluer : un canevas pour le développement incrémental et itératif de logiciels à composants et orientés service*, in "Les systèmes logiciels : évolution, maintenance et rénovation (traité IC2, série informatique et si)", A.-D. SERIAI (editor), Hermes Science Publications, January 2014, https://hal.inria.fr/hal-00911845

Books or Proceedings Editing

[60] C. DUBOIS, L. DUCHIEN, N. LEVY (editors). *Actes des Sixièmes journées nationales du Groupement De Recherche CNRS du Génie de la Programmation et du Logiciel*, Conservatoire National des Arts et MétiersFrance, June 2014, 239 p. , https://hal.inria.fr/hal-01055907

[61] L. DUCHIEN, J.-L. GIAVITTO (editors). *Editorial*, 3, Technique et Science Informatiques, May 2014, vol. 33, 4 p. , https://hal.inria.fr/hal-01055909

Research Reports

[62] F. ALVARES DE OLIVEIRA JR., E. RUTTEN, L. SEINTURIER. *High-level Language Support for the Control of Reconfiguration in Component-based Architectures*, Inria Grenoble - Rhône-Alpes ; Inria Lille - Nord Europe ; Laboratoire d'Informatique Fondamentale de Lille, January 2015, n° RR-8669, https://hal.inria.fr/hal-01103548

[63] B. BAUDRY, M. MONPERRUS. *The Multiple Facets of Software Diversity: Recent Developments in Year 2000 and Beyond*, 2014, https://hal.archives-ouvertes.fr/hal-01067782

[64] A. BSILA, N. FERRY, G. HORN, T. KIRKHAM, M. MALAWSKI, N. PARLAVANTZAS, C. PÉREZ, J. ROUZAUD-CORNABAS, D. ROMERO, A. ROSSINI, A. SOLBERG, H. SONG. *D3.1.1 - Upperware Prototype*, March 2014, https://hal.inria.fr/hal-00991340

[65] L. DUCHIEN, C. QUINTON, L. SEINTURIER. *Recommandations d'architecture pour une optimisation de la configuration d'applications entre le Digital Home et le Cloud*, Orange - France Télécom, July 2014, n° Livrable L4, https://hal.inria.fr/hal-01087316

[66] J. DURIBREUX, R. ROUVOY, M. MONPERRUS. *An Energy-efficient Location Provider for Daily Trips*, August 2014, n° RR-8586, 18 p. , https://hal.inria.fr/hal-01058830

[67] M. DURUISSEAU, R. ROUVOY. *Automatic Inference of Roadmaps from Raw Mobility Traces*, August 2014, n° RR-8585, 19 p. , https://hal.inria.fr/hal-01058824

[68] M. GOMEZ, R. ROUVOY, M. MONPERRUS, L. SEINTURIER. *A Recommender System of Buggy App Checkers for App Store Moderators*, Inria Lille, October 2014, n° RR-8626, https://hal.inria.fr/hal-01079681

[69] M. MARTINEZ, L. DUCHIEN, M. MONPERRUS. *Accurate Extraction of Bug Fix Pattern Occurrences using Abstract Syntax Tree Analysis*, Inria, 2014, https://hal.archives-ouvertes.fr/hal-01075938

[70] M. MARTINEZ, M. MONPERRUS. *ASTOR: Evolutionary Automatic Software Repair for Java*, Inria, 2014, https://hal.archives-ouvertes.fr/hal-01075976

[71] D. MENDEZ, B. BAUDRY, M. MONPERRUS. *Analysis and Exploitation of Natural Software Diversity: The Case of API Usages*, Inria, 2014, https://hal.archives-ouvertes.fr/hal-01095501

[72] M. MONPERRUS, M. GERMAIN DE MONTAUZAN, B. CORNU, R. MARVIE, R. ROUVOY. *Challenging Analytical Knowledge On Exception-Handling: An Empirical Study of 32 Java Software Packages*, Laboratoire d'Informatique Fondamentale de Lille, 2014, https://hal.inria.fr/hal-01093908

[73] M. MONPERRUS, A. MAIA. *Debugging with the Crowd: a Debug Recommendation System based on Stackoverflow*, May 2014, https://hal.archives-ouvertes.fr/hal-00987395

[74] M. MONPERRUS. *Principles of Antifragile Software*, April 2014, https://hal.archives-ouvertes.fr/hal-00978310

[75] V. MUSCO, M. MONPERRUS, P. PREUX. *A Generative Model of Software Dependency Graphs to Better Understand Software Evolution*, Inria, 2014, https://hal.archives-ouvertes.fr/hal-01078716

[76] R. PAWLAK, M. MONPERRUS, N. PETITPREZ, C. NOGUERA, L. SEINTURIER. *Spoon v2: Large Scale Source Code Analysis and Transformation for Java*, Inria, 2014, n^{o} hal-01078532, https://hal.inria.fr/hal-01078532

[77] C. QUINTON, D. ROMERO, L. DUCHIEN. *Handling Constraints in Cardinality-Based Feature Models: The Cloud Environment Case Study*, February 2014, n^{o} RR-8478, 25 p. , https://hal.inria.fr/hal-00946922

[78] A. ROSSINI, N. NIKOLOV, D. ROMERO, J. DOMASCHKA, K. KRITIKOS, T. KIRKHAM, A. SOLBERG. *D2.1.2 - CloudML Implementation Documentation - First version*, April 2014, https://hal.inria.fr/hal-00991133

Other Publications

[79] M. COLMANT, M. KURPICZ, P. FELBER, L. HUERTAS, R. ROUVOY, A. SOBE. *BitWatts: A Process-level Power Monitoring Middleware*, December 2014, Middleware 2014 [*DOI :* 10.1145/2678508.2678529], https://hal.inria.fr/hal-01078825

[80] N. HADERER, C. RIBEIRO, R. ROUVOY, L. SEINTURIER, V. RIVRON. *Offloading Cyber-Physical Tasks using Mobile Crowdsourcing*, September 2014, Crowdsourcing and human computation multidisciplinary workshop, https://hal.inria.fr/hal-01087396

[81] G. HECHT, L. DUCHIEN, N. MOHA, R. ROUVOY. *Detection of Anti-patterns in Mobile Applications*, June 2014, COMPARCH 2014, https://hal.inria.fr/hal-01029994

[82] M. KURPICZ, M. COLMANT, L. HUERTAS, A. SOBE, P. FELBER, R. ROUVOY. *How energy-efficient is your cloud app?*, April 2014, Conférence d'informatique en Parallélisme, Architecture et Système (ComPAS), https://hal.inria.fr/hal-00974400

[83] C. QUINTON, D. ROMERO, L. DUCHIEN. *SALOON, a Platform for Selecting and Configuring Cloud Environments*, June 2014, https://hal.inria.fr/hal-01011039

[84] M. SANGO, L. DUCHIEN, C. GRANSART. *SARA component approach for the development of railway safety-critical applications*, June 2014, ACM SIGSOFT CompArch 2014, https://hal.inria.fr/hal-01110253

[85] A. TAHRI, L. DUCHIEN, J. PULOU. *Software Evolution Multi-View : From the Smart Home to the Cloud*, June 2014, 6es Journées Nationales du GDR GPL, https://hal.inria.fr/hal-01012015

References in notes

[86] A. BOURDON, F. BORDAGE. *PowerAPI : mesurer la consommation électrique des logiciels sans wattmètre*, in "GreenIT.fr", January 2013, https://hal.inria.fr/hal-00783426

[87] A. BOURDON, A. NOUREDDINE, R. ROUVOY, L. SEINTURIER. *PowerAPI: A Software Library to Monitor the Energy Consumed at the Process-Level*, in "ERCIM News", January 2013, vol. 92, pp. 43-44, https://hal.inria.fr/hal-00772454

[88] A. BOURDON, R. ROUVOY, L. SEINTURIER. *Mesurer la consommation en énergie des logiciels avec précision*, in "01 Business & Technologies", January 2013, https://hal.inria.fr/hal-00772461

[89] L. CAPRA, W. EMMERICH, C. MASCOLO. *A micro-economic approach to conflict resolution in mobile computing*, in "SIGSOFT FSE", 2002, pp. 31-40

[90] P. CLEMENTS, L. NORTHROP. *Software Product Lines Practices and Patterns*, Addison-Wesley, 2002

[91] V. DALLMEIER, A. ZELLER, B. MEYER. *Generating Fixes from Object Behavior Anomalies*, in "Proceedings of the 2009 IEEE/ACM International Conference on Automated Software Engineering", Washington, DC, USA, ASE '09, IEEE Computer Society, 2009, pp. 550–554, http://dx.doi.org/10.1109/ASE.2009.15

[92] C. DEMAREY, D. FOURNIER. *FraSCAti, prenez le contrôle sur vos applications*, in "Programmez", December 2010, http://hal.inria.fr/inria-00531342

[93] C. DEMAREY, D. FOURNIER. *SOA facile avec SCA*, in "Programmez", November 2010, n^o 135, http://hal.inria.fr/inria-00531338

[94] F. FOUQUET, E. DAUBERT, N. PLOUZEAU, O. BARAIS, J. BOURCIER, J.-M. JÉZÉQUEL. *Dissemination of Reconfiguration Policies on Mesh Networks*, in "Proceedings of 12th IFIP International Conference on Distributed Applications and Interoperable Systems (DAIS'12)", Stockholm, Sweden, K. M. GÖSCHKA, S. HARIDI (editors), Lecture Notes in Computer Science, Springer, June 2012, vol. 7272, pp. 16-30

[95] M. GOLDMANN, G. KREITZ. *Measurements on the spotify peer-assisted music-on-demand streaming system*, in "Peer-to-Peer Computing", T. ASAMI, T. HIGASHINO (editors), IEEE, 2011, pp. 206-211

[96] N. HADERER, R. ROUVOY, L. SEINTURIER. *A preliminary investigation of user incentives to leverage crowdsensing activities*, in "2nd International IEEE PerCom Workshop on Hot Topics in Pervasive Computing (PerHot)", San Diego, États-Unis, IEEE Computer Society, March 2013, http://hal.inria.fr/hal-00783873

[97] N. HADERER, R. ROUVOY, L. SEINTURIER. *Dynamic Deployment of Sensing Experiments in the Wild Using Smartphones*, in "13th International IFIP Conference on Distributed Applications and Interoperable Systems (DAIS)", 6 2013, pp. 43-56, http://hal.inria.fr/hal-00804114

[98] M. HALL, E. FRANK, G. HOLMES, B. PFAHRINGER, P. REUTEMANN, I. H. WITTEN. *The WEKA data mining software: an update*, in "SIGKDD Explorer Newsletter", 2009, vol. 11, n^o 1, pp. 10–18

[99] M. HARMAN. *Automated patching techniques: the fix is in*, in "Communications of the ACM", May 2010, vol. 53, n^o 5, pp. 108–108, http://doi.acm.org/10.1145/1735223.1735248

[100] V. ISSARNY, N. GEORGANTAS, S. HACHEM, A. ZARRAS, P. VASSILIADIS, M. AUTILI, M. A. GEROSA, A. BEN HAMIDA. *Service-Oriented Middleware for the Future Internet: State of the Art and Research Directions*, in "Journal of Internet Services and Applications", May 2011, vol. 2, n^{o} 1, pp. 23-45, http://dx.doi.org/10.1007/s13174-011-0021-3

[101] W. LI, S. HU, J. LI, H.-A. JACOBSEN. *Community Clustering for Distributed Publish/Subscribe Systems*, in "CLUSTER", IEEE, 2012, pp. 81-89

[102] M. LÉGER, T. LEDOUX, T. COUPAYE. *Reliable Dynamic Reconfigurations in a Reflective Component Model*, in "CBSE", L. GRUNSKE, R. REUSSNER, F. PLASIL (editors), Lecture Notes in Computer Science, Springer, 2010, vol. 6092, pp. 74-92

[103] P. MERLE, R. ROUVOY, L. SEINTURIER. *A Reflective Platform for Highly Adaptive Multi-Cloud Systems*, in "10th International Workshop on Adaptive and Reflective Middleware (ARM'2011) at the 12th ACM/IFIP/USENIX International Middleware Conference", Lisbonne, Portugal, December 2011, pp. 1-7, http://hal.inria.fr/inria-00628643

[104] P. MERLE, R. ROUVOY, L. SEINTURIER. *FraSCAti: Adaptive and Reflective Middleware of Middleware*, in "12th ACM/IFIP/USENIX International Middleware Conference", Lisbonne, Portugal, December 2011, http://hal.inria.fr/hal-00650608

[105] A. NOUREDDINE, A. BOURDON, R. ROUVOY, L. SEINTURIER. *A Preliminary Study of the Impact of Software Engineering on GreenIT*, in "First International Workshop on Green and Sustainable Software", Zurich, Suisse, June 2012, pp. 21-27, http://hal.inria.fr/hal-00681560

[106] A. NOUREDDINE, A. BOURDON, R. ROUVOY, L. SEINTURIER. *Runtime Monitoring of Software Energy Hotspots*, in "ASE - The 27th IEEE/ACM International Conference on Automated Software Engineering - 2012", Essen, Allemagne, September 2012, pp. 160-169 [*DOI :* 10.1145/2351676.2351699], http://hal.inria.fr/hal-00715331

[107] A. NOUREDDINE, A. BOURDON, R. ROUVOY, L. SEINTURIER. *e-Surgeon: Diagnosing Energy Leaks of Application Servers*, Inria, January 2012, n^{o} RR-7846, http://hal.inria.fr/hal-00652992

[108] F. PARAISO, N. HADERER, P. MERLE, R. ROUVOY, L. SEINTURIER. *A Federated Multi-Cloud PaaS Infrastructure*, in "5th IEEE International Conference on Cloud Computing", Hawaii, États-Unis, June 2012, pp. 392 - 399 [*DOI :* 10.1109/CLOUD.2012.79], http://hal.inria.fr/hal-00694700

[109] R. PAWLAK, C. NOGUERA, N. PETITPREZ. *Spoon: Program Analysis and Transformation in Java*, 2006, n^{o} RR-5901, https://hal.inria.fr/inria-00071366

[110] N. PESSEMIER, L. SEINTURIER, T. COUPAYE, L. DUCHIEN. *A Component-Based and Aspect-Oriented Model for Software Evolution*, in "International Journal of Computer Applications in Technology (IJCAT)", 2008, vol. 31, n^{o} 1-2, pp. 94-105, http://hal.inria.fr/inria-00269895

[111] L. PROVENSI, F. ELIASSEN, R. VITENBERG, R. ROUVOY. *Improving Context Interpretation by Using Fuzzy Policies: The Case of Adaptive Video Streaming*, in "28th ACM Symposium on Applied Computing (SAC) - 8th Track on Dependable and Adaptive Distributed Systems (DADS)", Coimbra, Portugal, K. M. GÖSCHKA, R. OLIVEIRA, P. PIETZUCH, G. RUSSELLO (editors), ACM, March 2013, vol. 1, pp. 415-422, Best paper award, http://hal.inria.fr/hal-00799136

[112] M. RINARD, C. CADAR, D. DUMITRAN, D. ROY, T. LEU, W. BEEBEE JR. *Enhancing server availability and security through failure-oblivious computing*, in "Proceedings of the 6th conference on Symposium on Operating Systems Design & Implementation (OSDI'06)", USENIX Association, 2004, pp. 21–21

[113] M. SALEHIE, L. TAHVILDARI. *Self-adaptive software: Landscape and research challenges*, in "ACM Transactions on Autonomous and Adaptive Systems", May 2009, vol. 4, n^{o} 2, pp. 14:1–14:42, http://doi.acm.org/10.1145/1516533.1516538

[114] H. SCHAFFERS, M. SHARPE. *Services in the Future Internet*, April 2011, FP8 Expert Group, EU Directorate-General Information Society & Media, http://cordis.europa.eu/fp7/ict/ssai/docs/softwareconsultationreportfeb2011.pdf

[115] L. SCHUBERT, K. JEFFERY, B. NEIDECKER-LUTZ. *A Roadmap for Advanced Cloud Technologies under H2020*, Dec 2012, http://cordis.europa.eu/fp7/ict/ssai/docs/cloud-expert-group/roadmap-dec2012-vfinal.pdf

[116] L. SEINTURIER, P. MERLE, D. FOURNIER, N. DOLET, V. SCHIAVONI, J.-B. STEFANI. *Reconfigurable SCA Applications with the FraSCAti Platform*, in "Proceedings of the 6th IEEE International Conference on Service Computing (SCC'09)", September 2009, pp. 268–275, http://hal.inria.fr/inria-00397856

[117] L. SEINTURIER, P. MERLE, R. ROUVOY, D. ROMERO, V. SCHIAVONI, J.-B. STEFANI. *A Component-Based Middleware Platform for Reconfigurable Service-Oriented Architectures*, in "Software: Practice and Experience", May 2012, vol. 42, n^{o} 5, pp. 559-583 [*DOI :* 10.1002/SPE.1077], http://hal.inria.fr/inria-00567442

[118] G. TAMURA. *QoS-CARE: Un Système Fiable pour la Préservation de Contrats de Qualité de Service à travers de la Reconfiguration Dynamique*, Université des Sciences et Technologie de Lille - Lille I and Universidad de Los Andes, May 2012, http://hal.inria.fr/tel-00713345

[119] G. TAMURA, N. M. VILLEGAS, H. MÜLLER, J. P. SOUSA, B. BECKER, M. PEZZÈ, G. KARSAI, S. MANKOVSKII, W. SCHÄFER, L. TAHVILDARI, K. WONG. *Towards Practical Runtime Verification and Validation of Self-Adaptive Software Systems*, in "Software Engineering for Self-Adaptive Systems 2", R. DE LEMOS, H. GIESE, H. MÜLLER, M. SHAW (editors), LNCS, Springer, January 2013, vol. 7475, pp. 108-132, https://hal.inria.fr/hal-00709943

[120] J. URBAN. *NESSI Research Priorities for the next Framework Programme for Research and Technological Development FP8*, May 2011, http://www.nessi-europe.eu/files/Docs/NESSI%20SRA_update_May_2011_V1-0.pdf

[121] N. M. VILLEGAS, G. TAMURA, H. MÜLLER, L. DUCHIEN, R. CASALLAS. *DYNAMICO: A Reference Model for Governing Control Objectives and Context Relevance in Self-Adaptive Software Systems*, in "Software Engineering for Self-Adaptive Systems 2", R. DE LEMOS, H. GIESE, H. MÜLLER, M. SHAW (editors), LNCS, Springer, August 2012, vol. 7475, pp. 265-293, http://hal.inria.fr/hal-00713315

[122] Y. WEI, Y. PEI, C. A. FURIA, L. S. SILVA, S. BUCHHOLZ, B. MEYER, A. ZELLER. *Automated fixing of programs with contracts*, in "Proceedings of the 19th international symposium on Software testing and analysis", New York, NY, USA, ISSTA '10, ACM, 2010, pp. 61–72, http://doi.acm.org/10.1145/1831708.1831716

[123] C. WEISS, R. PREMRAJ, T. ZIMMERMANN, A. ZELLER. *How long will it take to fix this bug?*, in "Proceedings of the Fourth International Workshop on Mining Software Repositories", IEEE Computer Society, 2007, 1 p.

[124] T. XIE, S. THUMMALAPENTA, D. LO, C. LIU. *Data Mining for Software Engineering*, in "IEEE Computer", August 2009, vol. 42, n^{o} 8, pp. 55-62, http://doi.ieeecomputersociety.org/10.1109/MC.2009.256

[125] W. XU, L. HUANG, A. FOX, D. A. PATTERSON, M. I. JORDAN. *Detecting large-scale system problems by mining console logs*, in "Proceedings of the 22nd ACM Symposium on Operating Systems Principles (SOSP'09)", Big Sky, Montana, USA, J. N. MATTHEWS, T. E. ANDERSON (editors), ACM, October 2009, pp. 117-132

[126] R. DE LEMOS, H. GIESE, H. A. MULLER, M. SHAW, J. ANDERSSON, L. BARESI, B. BECKER, N. BENCOMO, Y. BRUN, B. CUKIC, R. DESMARAIS, S. DUSTDAR, G. ENGELS, K. GEIHS, K. M. GÖSCHKA, A. GORLA, V. GRASSI, P. INVERARDI, G. KARSAI, J. KRAMER, M. LITOIU, A. LOPES, J. MAGEE, S. MALEK, S. MANKOVSKII, R. MIRANDOLA, J. MYLOPOULOS, O. NIERSTRASZ, M. PEZZÈ, C. PREHOFE, W. SCHÄFER, R. SCHLICHTING, B. SCHMERL, D. B. SMITH, J. P. SOUSA, G. TAMURA, L. TAHVILDARI, N. M. VILLEGAS, T. VOGEL, D. WEYNS, K. WONG, J. WUTTKE. *Software Engineering for Self-Adaptive Systems: A Second Research Roadmap*, in "Software Engineering for Self-Adaptive Systems", R. DE LEMOS, H. GIESE, H. MÜLLER, M. SHAW (editors), Dagstuhl Seminar Proceedings, Springer, 2013, vol. 7475, pp. 1-26, http://hal.inria.fr/inria-00638157