Activity Report 2015

Team CTRL-A

Control techniques for Autonomic, adaptive and Reconfigurable Computing systems

Inria teams are typically groups of researchers working on the definition of a common project, and objectives, with the goal to arrive at the creation of a project-team. Such project-teams may include other partners (universities or research institutions).

RESEARCH CENTER
Grenoble - Rhône-Alpes

THEME
Distributed Systems and middleware

Table of contents

Team CTRL-A

Creation of the Team: 2014 January 01

Keywords:

Computer Science and Digital Science:

1.1.10. - Reconfigurable architectures
1.1.4. - High performance computing
1.1.6. - Cloud
1.3. - Distributed Systems
1.6. - Green Computing
2.1.10. - Domain-specific languages
2.1.8. - Synchronous languages
2.2. - Compilation
2.3.1. - Embedded systems
2.3.2. - Cyber-physical systems
2.4. - Reliability, certification
2.5. - Software engineering
2.6.2. - Middleware
4.9. - Security supervision
6.4. - Automatic control

Other Research Topics and Application Domains:

4.4.1. - Green computing
4.4.2. - Embedded sensors consumption
5. - Industry of the future
6.1.1. - Software engineering
6.4. - Internet of things
6.5. - Information systems
6.6. - Embedded systems
8.1. - Smart building/home

1. Members

Research Scientist

Eric Rutten [Team leader, Inria, Researcher, HdR]

Post-Doctoral Fellow

Frederico Alvares de Oliveira Junior [Inria, until Feb 2015]

Administrative Assistants

Carol Loquet [Inria, until Jun 2015]
Marion Ponsot [Inria, since July 2015]

Others

Gwenaël Delaval [Univ. Grenoble I, Associate Professor, LIG]
Stéphane Mocanu [INP Grenoble, Associate Professor, Gipsa-lab]
Bogdan Robu [Univ. Grenoble I, Associate Professor, Gipsa-lab]

Hassane Alla [Univ. Grenoble I, Professor, Gipsa-lab, until may 2015]
Nicolas Marchand [CNRS, Senior Researcher, Gipsa-lab]
Adja Sylla [CEA, PhD, from Feb 2015]
Mengxuan Zhao [Cifre Orange labs, PhD, until may 2015]
Naweiluo Zhou [Univ. Grenoble I, PhD]
Mihaly Berekmeri [Univ. Grenoble I, PhD,Gipsa-lab]
Sophie Cerf [Univ. Grenoble I, Masters2 intern, Aug 2015]
Alia Hajjar [Inria, Masters2 intern, from Feb until June 2015]
Abdoul-Razak Hassimi Harouna [Univ. Grenoble I, Masters1 intern, from Feb 2015 until June 2015]

2. Overall Objectives

2.1. Objective: control support for autonomic computing

Computing systems are more and more ubiquitous, at scales from tiny embedded systems to large-scale cloud infrastructures. They are more and more adaptive and reconfigurable, for resource management, energy efficiency, or by functionality. Furthermore, these systems are increasingly complex and autonomous: their administration cannot any longer rely on a strong interaction with a human administrator. The correct design and implementation of automated control of the reconfigurations and/or the tuning is recognized as a key issue for the effectiveness of these adaptive systems.

In the last dozen years, the notion of Autonomic Computing has been proposed and supported by industrials like IBM, as a framework for the design of self-adaptive systems. It addresses objectives of self-configuration, w.r.t. deployment issues, self-optimization, w.r.t; resources management, self-healing, w.r.t. robustness and fault-tolerance, and self-protection, w.r.t. security aspects. It relies on a feedback control loop architecture, with: monitors and reconfiguration actions, connected to the API infrastructure of the system under control ; an autonomic management component , transforming flows of monitoring information into adaptation actions, which can be addressed naturally by reactive programming ; a decision mechanism inside the latter manager, which can rely on behavioral models of the managed system.

Our objective is to build methods and tools for the design of safe controllers for autonomic, adaptive, reconfigurable computing systems. To attain this goal, we propose to combine Computer Science and Control Theory, at the levels of systems infrastructures, programming support, and modeling and control techniques. We explore this topic along three main thematic axes: **modeling and control theory** with discrete time, continuous time and hybrid models (distributed control, event-based control, discrete event systems, supervisory control) ; **programming support** with reactive (synchronous languages, controller synthesis, higher-order) and component-based approaches (Fractal framework, language-level support of reconfiguration, Event-Condition-Action (ECA) rules) ; **infrastructure-level support** (operating system, middleware) for monitors/sensors, administration actions/actuators, architectures for controllability, software and hardware reconfiguration mechanisms.

We aim to address applications with reconfiguration control problems from the small scale (facing variability for system on chips (SoCs), reconfigurable architectures, networks on chips (NoCs), etc.) up to the extra large scale (administration, coordination, optimization of data centers and cloud computing, green computing, smart grids, etc.)

We propose to form a team grouping the most active community in France on Control of Computing, with members until now separated by laboratories structure, in order to contribute more efficiently in the local context to the high potential impact on micro- and nanotechnologies in Grenoble, and more widely nationally and internationally in the emerging community on Feedback Computing.

2.2. Motivation: safe and optimal autonomic management

2.2.1. The problem of automating computing systems administration

It lies in the difficulty of manual management in a safe or optimal way, of computing systems which become more and more complex and flexible. There is a deep need for the automation of their management, handled in a closed-loop: the system is monitored by sensors, which enable updating a self-representation of the system, upon which reconfiguration actions are decided, and in turn they are executed, with an effect on the system, that will be measured by sensors. Such dynamically reconfigurable systems, also called adaptive or autonomic computing systems, are characterized by the ability to modify, on-line, their computing structure, in reaction to conditions in their execution environment or platform.

Motivations for dynamic adaptivity are in important questions like : resource management e.g., energy, computation, memory, bandwidth, circuit area, time ; quality of service e.g., levels of precision in computing, of urgence of treatment, graceful degradation ; dependability and fault tolerance, e.g., controlling migrations in response to loss of a processor. Adaptivity concerns systems ranging from hardware to operating systems to services and applications, and in size from tiny embedded systems to large-scale data-centers, from multi-core processors to the Cloud. Their complexity is growing, in scale (software of hardware), but also in interactions between different aspects of reconfiguration.

The design of the adaptation controllers is largely done in an *ad-hoc* fashion, involving lots of different approaches, intuitions, and heuristics. There is an important need for well-founded models and techniques for the safe design of these control loops, which can provide designers with a support to master the complexity of designs, and with guarantees w.r.t. the correctness of the designed controllers.

2.2.2. Our model-based control approach

We aim to build general methodologies and tools for the model-based control of reconfigurable computing system, validated in a representative range of reconfigurable systems.

The classical approach in computer science consists of: first programming, and then verifying. We want to explore, in contrast to this, an alternative approach, more effective (easier for the designer) and safer (better guarantees), inspired by control techniques: first model behaviors of the (uncontrolled) system, and its control interfaces, at each component's level ; then specify the adaptation strategy or policy, i.e., control objectives, and possibly check controllability ; finally, derive the controller solution: automatically synthesize the controller (for classes of problems where it is possible).

Our general topic is considering computing systems as object of automatic control, which is a newly emerging scientific theme, often considering continuous models. We will be using our complementary backgrounds in reactive systems and synchronous languages, in Control Theory and in experiences in applying various control techniques to computing systems, as well as a general orientation to apply formal methods to real-world systems. We are reversing the classical view of computer science for control systems, and consider, more originally, *control techniques for computing systems*.

2.2.3. Opportunities between computer science and control theory

This new and emerging combination of computing and control can bring novel contributions both ways : adaptive computing can benefit from control techniques, which provide designers with a broad range of results, and begin to be equipped with efficient tools e.g., connected to the synchronous technology, which is essential for concrete impact on real-world systems. Research in Control Theory can benefit from computing systems, embedded or large scale, as new application domain, where theoretical results can be evaluated and transferred, and from where new interesting and relevant problems can come up.

Risks could be in e.g., the fact that such a new and mixed topic of systems and control techniques does not yet correspond to an identified scientific community, making it difficult to find people involving themselves rather than staying in their respective community of origin. But this is in direct relationship to the originality of the subject, and is compensated by the identification of concrete potentials in our ongoing work. This multi-disciplinary work takes much more exploratory time and cooperative discussion than more classical research programs, but it brings all the more original results.

Control for Computing is in its founding phase: the disadvantage is there are no comfortable inherited results and community, the advantages are in the novelty and relevance of founding a new direction.

3. Research Program

3.1. Modeling and control techniques for autonomic computing

3.1.1. Continuous control

Continuous control was used to control computer systems only very recently and in few occasions, despite the promising results that were obtained. This is probably due to many reasons, but the most important seems to be the difficulty by both communities to transform a computer system problem into an automatic control problem. The aim of the team is to explore how to formalize typical autonomic commuting cases into typical control problems. Many new methodological tools will probably be useful for that, e.g., we can cite the hybrid system approach, predictive control or event-based control approach. Computer systems are not usual for the control system community and they often present non-conventional control aspects like saturation control. New methodological tools are required for an efficient use of continuous-time control in computer science.

3.1.2. Discrete control

Discrete control techniques are explored at long-term, to integrate more control in the BZR language, and adress more general control issues, wider than BZR's limitations. Directions are : expressiveness (taking into account in the LTS models value domains of the variables in the program) ; adaptive control (where the controller itself can dynamically switch between differents modes) ; distributed control (for classes of problems where communicating controllers can be designed) ; optimal control (w.r.t. weight functions, on states, transitions, and paths, with multicriteria techniques) ; timed and hybrid control bringing a new dimension for modeling and control, giving solutions where discrete models fail.

3.2. Design and programming for autonomic computing

3.2.1. Reactive programming

Autonomic systems are intrinsically reconfigurable. To describe, specify or design these systems, there is a need to take into account this reconfigurability, within the programming languages used. We propose to consider the reconfigurability of systems from the angle of two properties: the notion of time, as we want to describe the state and behavior of the system before, and after its reconfiguration; the notion of dynamicity of the system, i.e., considering that the system's possible behaviors throughout execution are not completely known, neither at design-time nor at initial execution state. To describe and design such reactive systems, we propose to use the synchronous paradigm. It has been successfully used, in industry, for the design of embedded systems. It allows the description of behaviors based on a specific model of time (discrete time scale, synchronous parallel composition), providing properties which are important w.r.t. the safety of the described system: reactivity, determinism, preservation of safety properties by parallel composition (with other parts of the system or with its environment). Models and languages for control, proposed in this framework, provide designers, experts of the application domain, with a user-friendly access to highly technical formal methods of DCS, by encapsulating them in the compilation of concrete programming languages, generating concrete executable code. They are based on discrete models, but also support programming of sampled continuous controllers.

3.2.2. Component-based approach and domain-specific languages

For integration of the previous control kernels into wider frameworks of reconfigurable systems, they have to be integrated in a design flow, and connected on the one side with higher-level specification languages (with help of DSLs), and on the other side with the generated code level target execution machines. This calls for the adoption of a component-based approach with necessary features, available typically in Fractal, for explicitly identifying the control interfaces and mechanisms.

Structuring and instrumentation for controllability will involve encapsulation of computations into components, specification of their local control (activation, reconfiguration, suspension, termination), and exporting appropriate interfaces (including behavior abstraction). Modeling the configurations space requires determining the controlled aspects (e.g., heterogenous CPUs loads, fault-tolerance and variability, memory, energy/power consumption, communication/bandwidth, QoS level) and their control points, as well as APIs for monitors and actions. Compilation and execution will integrate this in a complete design flow involving : extraction of a reactive model from components; instrumentation of execution platforms to be controllable; combination with other controllers; general "glue" and wrapper code.

Integration of reactive languages and control techniques in component-based systems brings interesting questions of co-existence w.r.t. other approaches like Event-Condition-Action (ECA) rules, or Complex Event Processing (CPE).

3.3. Infrastructure-level support for autonomic computing

The above general kernel of model-based control techniques can be used in a range of different computing infrastructures, representing complementary targets and abstraction levels, exploring the two axes :

- from hardware, to operating system/virtual machine, to middleware, to applications/service level;
- across different criteria for adaptation: resources and energy, quality of service, dependability.

3.3.1. Software and adaptive systems

Autonomic administration loops at operating systems or middleware level are already very widespread. An open problem remains in design techniques for controllers with predictability and safety, e.g. w.r.t. the reachable states. We want to contribute to the topic of discrete control techniques for these systems, and tackle e.g. problems of coordination of multiple autonomic loops in data-centers, as in the ANR project CtrlGreen. Another target application is the control of clusters in map-reduce applications. The objective is to use continuous time control in order to tune finely the number of required clusters for an application running on a map-reduce server. This will use results of the ANR project MyCloud that enables to simulate clients on a real map-reduce server. On a longer term, we are interested in control problems in administration loops of event-based virtual machines, or in the deployment of massively parallel computation of the Cloud.

3.3.2. Hardware and reconfigurable architectures

Reconfigurable architectures based on Field Programmable Gate Arrays (FPGA) are an active research area, where infrastructures are more and more supportive of reconfiguration, but its correct control remains an important issue. Work has begun in the ANR Famous project on identifying domain-specific control criteria and objectives, monitors and management APIs, and on integrating control techniques in the high-level RecoMARTE environment. On a longer term, we want to work on methods and tools for the programming of **multicore architectures**, exploiting the reconfigurability potentials and issues (because of variability, loss of cores), e.g. in our cooperation with ST Microelectronics, using a Fractal-based programming framework in the P2012 project, and in cooperation with Inria Lille (Adam), or with the CEA and TIMA on integrating control loops in the architecture for a fine control of the energy and of the required nodes for running a given application task.

3.3.3. Applications and autonomic systems

In autonomic systems, control systems remain a lively source of inspiration, partly because the notion of control loop implementation is known and practiced naturally. On a wider scale, we started a cooperation with Orange Labs on "intelligent" building automation and control for the Smart Grid, through modeling and control of appliances w.r.t. their power consumption modes, at home, building, and city levels. Other partners on these topics are CEA LETI/DACLE and Schneider Electric.

We could explore more systems and applications e.g., Human-Machine Interfaces, or the orchestration of services. They can help design more general solutions, and result in a more complete methodology.

4. Application Domains

4.1. Distributed systems and High-Performance Computing

Distributed systems have grown to levels of scale and complexity where it is difficult to master their administration and resources management, in dynamic ans open environments. One of the growing concerns is that the energy consumption has reached levels where it can not be considered negligible anymore, ecologically or economically. Data centers or high performance computing grids need to be controlled in order to combine minimized power needs with sustained performance and quality of service. As mentioned above, this motivates the automation of their management, and is the major topic of, amongst others, our ANR project Ctrl-Green.

Another challenge in distributed systems is in the fast growing amounts of data to process and store. Currently one of the most common ways of dealing with these challenges is the parallel programming paradigm MapReduce which is slowly becoming the de facto tool for Big Data analytics. While its use is already widespread in the industry, ensuring performance constraints while also minimizing costs provides considerable challenges. Current approaches to ensure performance in cloud systems can be separated into three categories: static, reactive, predictive and hybrid approaches. In the industry, static deployments are the standard and usually tuned based on the application peak demand and are generally over-provisioned. Reactive approaches are usually based on reacting to an input metric such as the current CPU utilisation, request rate, response time by adding and removing servers as necessary. Some public cloud providers offer reactive techniques such as the Amazon Auto Scaler. They provide the basic mechanisms for reactive controllers, but it is up to the user to define the static scaling thresholds which is difficult and not optimal. To deal with this issue, we propose a control theoretical approach, based on techniques that have already proved their usefulness for the control community.

In the domain of parallel systems and High Performance Computing, systems are traditionally less open and more controlled by administrators, but this trend is changing, as they are facing the same challenges in energy consumption, needs for adaptivity in reaction to changing workloads, and security issues in computation outsourcing. Topics of interest for us in this domain concern problem in dynamical management of memory and communications features, which we are exploring in the HPES project of the Labex Persybal-lab (see 9.1).

4.2. Reconfigurable architectures in embedded systems

Dynamically reconfigurable hardware has been identified as a promising solution for the design of energy efficient embedded systems. A common argument in favor of this kind of architecture is the specialization of processing elements, that can be adapted to application functions in order to minimize the delay, the control cost and to improve data locality. Another key benefit is the hardware reuse to minimise the area, and therefore the static power and cost. Further advantages such as hardware updates in long-life products and self-healing capabilities are also often mentioned. In presence of context changes (e.g. environment or application functionality), self-adaptive technique can be applied as a solution to fully benefit from the runtime reconfigurability of a system.

Dynamic Partial Reconfiguration (DPR) of FPGA is another accessible solution to implement and experiment reconfigurable hardware. It has been widely explored and detailed in literature. However, it appears that such solutions are not extensively exploited in practice for two main reasons: i) the design effort is extremely high and strongly depends on the available chip and tool versions; and ii) the simulation process, which is already complex for non-reconfigurable systems, is prohibitively large for reconfigurable architectures. As a result, new adequate methods are required to fully exploit the potential of dynamically reconfigurable and self-adaptive architectures. We are working in this topic, especially on the reconfiguration control aspect, in cooperation with teams specialized in reconfigurable architectures such as the former DaRT team at Inria Lille, and LabSticc in Lorient, as in the recently ended ANR project Famous.

A new ANR project in this application domain, starting end of 2015, is called HPeC, in cooperation with amongst others LabSticc in Lorient and Clermont-Ferrand U., will consider embedded video processing on drones (see 9.2.1).

4.3. Smart environments and Internet of Things

Another application domain for autonomic systems design and control is the Internet of Things, and especially the design of smart environments, at the level of homes, buildings, or cities. These domains are often considered at the level of sensors networks, with a strong emphasis on the acquisition of data in massive scales. The infrastructures are sometimes also equipped with actuators, with a wide range of applications, for example concerning lighting or heating, or access and security aspects. We are interested in closing the control loop in such environments, which is less often studied. In particular, rule-based languages are often used to define the automated systems, and we want to contribute to the safe design of such controllers with guarantees on their behaviors. We are working in this topic in cooperation with teams specialized in infrastructures for smart environments at CEA LETI/DACLE and Orange labs (see 8.1,8.2).

5. Highlights of the Year

5.1. Highlights of the Year

5.1.1. Community

We have been invited to participate to the organization of events, which highlight our active presence in the scientific life in the two domains which we are bridging :

- autonomic computing: Eric Rutten is PC co-chair of the 3rd IEEE International Conference on Cloud and Autonomic Computing, CAC 2015 (http://autonomic-conference.org/) [19], Part of FAS* - Foundation and Applications of Self* Computing Conferences, Collocated with: The 9th IEEE Self-Adaptive and Self-Organizing System Conference, The 15th IEEE Peer-to-Peer Computing Conference ; and PC member, as well as workshops chair, of the 12th IEEE International Conference on Autonomic Computing, ICAC 2015 (http://icac2015.imag.fr/), the two major conferences on the topic.
- control: Eric Rutten is organizer of a Special Session on Dependable Discrete control for adaptive and reconfigurable computing systems at the 5th IFAC international workshop on Dependable Control of Discrete Systems , DCDS (http://www.gdl.cinvestav.mx/dcds2015) ; he is on the IFAC Technical Committee 1.3 on Discrete Event and Hybrid Systems, (http://tc.ifac-control.org/1/3/) and on the IEEE Control Systems Society Discrete Event Systems Technical Committee (http://discrete-event-systems.ieeecss.org).

5.1.2. Invited keynote talk

Eric Rutten was invited to give a talk at the 11th International Conference on Distributed Computing and Internet Technology (ICDCIT-2015) [13] and at the seminar of the College of Information and Computer Sciences (CICS), University of Massachusetts Amherst, USA, 28 sept.2015.

6. New Software and Platforms

6.1. Heptagon/BZR

We want to produce results concretely usable by third parties, either in cooperative projects, or by free diffusion of tools. One perspective is to build tool boxes for the design of continuous control solutions for computing systems: it will be explored in the future. A readily available result concerns discrete control and programming.

FUNCTIONAL DESCRIPTION

Heptagon is an experimental language for the implementation of embedded real-time reactive systems. It is developed inside the Synchronics large-scale initiative, in collaboration with Inria Rhones-Alpes. It is essentially a subset of Lucid Synchrone, without type inference, type polymorphism and higher-order. It is thus a Lustre-like language extended with hierchical automata in a form very close to SCADE 6. The intention for making this new language and compiler is to develop new aggressive optimization techniques for sequential C code and compilation methods for generating parallel code for different platforms. This explains much of the simplifications we have made in order to ease the development of compilation techniques.

- Participants: Adrien Guatto, Marc Pouzet, Cédric Pasteur, Léonard Gérard, Brice Gelineau, Gwenaël Delaval and Eric Rutten
- Contact: Gwenaël Delaval
- URL: http://bzr.inria.fr

HEPTAGON has been used to build BZR [1], which is an extension of the former with contracts constructs. These contracts allow to express dynamic temporal properties on the inputs and outputs of HEPTAGON node. These properties are then enforced, within the compilation of a BZR program, by discrete controller synthesis, using the SIGALI tool [2]. The synthesized controller is itself generated in HEPTAGON, allowing its analysis and compilation towards different target languages (C, Java, VHDL).

Heptagon/BZR has been recently integrated with the ReaX verification and controller synthesis tool. The ReaX tool allows the handling of numerico-boolean programs by using abstract interpretation for controller synthesis.

Prospects about Heptagon/BZR lie in developping methodological and programming tools for : precise diagnosis in case of controller synthesis failure ; identification of relevant domain of abstractions, in relation with the use of the ReaX tool ; integration in various execution platforms (Fractal, reconfigurable FPGA, etc.)

7. New Results

7.1. Discrete control and reactive language support

Participants: Gwenaël Delaval, Eric Rutten, Stéphane Mocanu, Alia Hajjar, Abdoul-Razak Hassimi Harouna.

Concerning language support, we have designed and implemented BZR, a mixed imperative/declarative programming language: declarative contracts are enforced upon imperatively described behaviors (see 6.1). The semantics of the language uses the notion of Discrete Controller Synthesis (DCS) [5]. This work is done in close cooperation with the Inria team Sumo at Inria Rennes (H. Marchand).

[1] http://bzr.inria.fr
[2] http://www.irisa.fr/vertecs/Logiciels/sigali.html

New results concern the master internship of Alia Hajjar, co-directed by Gwenaël Delaval an Stéphane Mocanu, on the subject of Application of control of reactive environments and probabilistic models on Transactional Memory. Multiprocessor environments which use concurrent programs and data structures showed the need of techniques to organize the usage of the shared structures, to reduce the unpredicted delay and reduce the contention between concurrent processors. Transactional Memory (TM) is a programming model that eases development of concurrent applications. Concurrent programming causes conflicts and TM is a way to resolve these conflicts with the transaction paradigm. To control conflict, techniques are provided to optimize (identify the best) degree of parallelism. In this framework, the aim is to control the TM system by adapting the degree of parallelism in order to maximize the throughput, i.e., number of committed transactions per time. The main objective is to minimize the execution time of a parallel application, thus maximize the throughput. During this master's thesis, the behavior of a multithreaded TM environment has been modeled as a stochastic discrete event system. The Heptagon/BZR language has then been used to implement this model for simulation, and evaluation of control strategies.

Ongoing work concerns aspects of compilation and debugging and exploring the notion of adaptive discrete control, which is yet an open question in discrete control in contrast to the well-known adaptive continuous control.

Another activity related to discrete control is or work with Leiden University and CWI (N. Khakpour, now at Linnaeus U., and F. Arbab) on enforcing correctness of the behavior of an adaptive software system during dynamic adaptation is an important challenge along the way to realize correct adaptive systems [11].

7.2. Design and programming

7.2.1. Component-based approaches

Participants: Frederico Alvares de Oliveira Junior, Eric Rutten.

Architecting in the context of variability has become a real need in today's software development. Modern software systems and their architecture must adapt dynamically to events coming from the environment (e.g., workload requested by users, changes in functionality) and the execution platform (e.g., resource availability). Component-based architectures have shown to be very suited for self-adaptation especially with their dynamical reconfiguration capabilities. However, existing solutions for reconfiguration often rely on low level, imperative, and non formal languages. We have defined Ctrl-F, a domain-specific language whose objective is to provide high-level support for describing adaptation behaviors and policies in component-based architectures. It relies on reactive programming for formal verification and control of reconfigurations. We integrate Ctrl-F with the FraSCAti Service Component Architecture middleware platform, and apply it to the Znn.com self-adaptive case study [20], [15], [14], [18].

We work on the topic in cooperation with the Spirals Inria team at Inria Lille (L. Seinturier). It constitutes a follow-up on previous work in the ANR Minalogic project MIND, industrializing the Fractal component-based framework, with a continuation of contacts with ST Microelectronics (V. Bertin). Our integration of BZR and Fractal [4], [2] is at the basis of our current work.

7.2.2. Rule-based systems

Participants: Adja Sylla, Eric Rutten.

We are starting a cooperation with CEA LETI/DACLE on the topic of a high-level language for safe rule-based programming in the LINC platform. The general context os that of the runtime redeployment of distributed applications, for example managing smart buildings. Motivations for redeployment can be diverse: load balancing, energy saving, upgrading, or fault tolerance. Redeployment involves changing the set of components in presence, or migrating them. The basic functionalities enabling to start, stop, migrate, or clone components, and the control managing their safe coordination, will have to be designed in the LINC middleware developed at CEA.

The transactional nature of the LINC platform insures the correct execution of each of the rules constituting the program, but there still is a need to insure the safety of their coordination, and of the behavior resulting from their sequential execution. For example, in the smart environments application domain, we must insure safety of control decisions, so that all the configurations that can be reached are safe, as well as the sequences of actions in switching between them. For this we will rely on automata-based models and control, using the BZR language, and integrating it in a domains specific language. Our work builds upon preliminary results involving colored Petri nets models [17].

The PhD of Adja Sylla at CEA on this topic is co-advised with F. Pacull and M. Louvel.

7.3. Infrastructure-level support

We apply the results of the previous axes of the team's activity to a range of infrastructures of different natures, but sharing a transversal problem of reconfiguration control design. From this very diversity of validations and experiences, we draw a synthesis of the whole approach [13], towards a general view of Feedback Control as MAPE-K loop in Autonomic Computing [21].

7.3.1. Autonomic Cloud and Big-Data systems

7.3.1.1. Coordination in multiple-loop autonomic Cloud systems

Participants: Soguy Gueye, Gwenaël Delaval, Eric Rutten.

Complex computing systems are increasingly self-adaptive, with an autonomic computing approach for their administration. Real systems require the co-existence of multiple autonomic management loops, each complex to design. However their uncoordinated co-existence leads to performance degradation and possibly to inconsistency. There is a need for methodological supports facilitating the coordination of multiple autonomic managers. To tackle this problem, we take a global view and underscore that Autonomic Management Systems (AMS) are intrinsically reactive, as they react to flows of monitoring data by emitting flows of reconfiguration actions. Therefore we propose a new approach for the design of AMSs, based on synchronous programming and discrete controller synthesis techniques. They provide us with high-level languages for modeling the system to manage, as well as means for statically guaranteeing the absence of logical coordination problems. Hence, they suit our main contribution, which is to obtain guarantees at design time about the absence of logical inconsistencies in the taken decisions. We detail our approach, illustrate it by designing an AMS for a realistic multi-tier application, and evaluate its practicality with an implementation [10].

In order to coordinate managers without breaking their natural modularity. we address the problem with a method stressing modularity, and focusing on the discrete control of the interactions of managers. We make proposals for the distributed execution of modular controllers, first in synchronized way, and then relaxing this synchronization. We apply and validate our method on a multi-loop multi-tier system in a data-center [16].

We addressed these problems in the context of the ANR project Ctrl-Green, in cooperation with LIG (N. de Palma) in the framework of the PhD of S. Gueye and the post-doc of N. Berthier.

7.3.1.2. Control for Big data

Participants: Bogdan Robu [Gipsa-lab], Mihaly Berekmeri [Gipsa-lab], Nicolas Marchand [Gipsa-lab].

Companies have a fast growing amounts of data to process and store, a data explosion is happening next to us. Currently one of the most common approaches to treat these vast data quantities is the MapReduce parallel programming paradigm. While it?s use is widespread in the industry, ensuring performance constraints, while also minimizing costs, provides considerable challenges. To deal with these issues we propose a control theoretical approach, based on techniques that have already proved their usefulness in the control community. We developed an algorithm to create the first linear dynamic model for a Big Data MapReduce Cloud system, running a concurrent workload. Furthermore we identify two important control use cases: relaxed performance - minimal resource and strict performance. We developed the first feedback control mechanism for such systems. Then to minimize the number of control actuations, an event-based feedback controller was also introduced. Furthermore to address the strict performance challenges a feedforward controller that efficiently suppresses the effects of large workload size variations is developed. On top of this issues an optimal predictive control which deals with concurrent objectives (dependability and performance) is implemented. The approach is validated online in a benchmark running in a real 60 node MapReduce cluster, using a data intensive Business Intelligence [22], [23].

This work is performed in cooperation with LIG (S. Bouchenak) in the framework of the PhD of M. Berekmeri.

7.3.2. Reconfiguration control in DPR FPGA

Participant: Eric Rutten.

Dynamically reconfigurable hardware has been identified as a promising solution for the design of energy efficient embedded systems. However, its adoption is limited by the costly design effort including verification and validation, which is even more complex than for non dynamically reconfigurable systems. We worked on this topic in the context of a ensign environment, developed in the framework of the ANR project Famous, in cooperation with LabSticc in Lorient and Inria Lille (DaRT team) [12]. We proposed a tool-supported formal method to automatically design a correct-by-construction control of the reconfiguration. By representing system behaviors with automata, we exploit automated algorithms to synthesize controllers that safely enforce reconfiguration strategies formulated as properties to be satisfied by control. We design generic modeling patterns for a class of reconfigurable architectures, taking into account both hardware architecture and applications, as well as relevant control objectives. We validate our approach on two case studies implemented on FPGAs [1].

We are currently valorizing results in more publications [12], [9], and extending the use of control techniques by evaluating the new tool ReaX developed at Inria Rennes (Sumo).

We are starting a new ANR project called HPeC, within which some of these topics will be extended, especially regarding hierarchical and modular control, and logico-numeric aspects.

7.3.3. Autonomic memory management in HPC

Participants: Naweiluo Zhou, Gwenaël Delaval, Bogdan Robu, Eric Rutten.

Parallel programs need to manage the time trade-off between synchronization and computation. A high parallelism may decrease computing time but meanwhile increase synchronization cost among threads. Software Transactional Memory (STM) has emerged as a promising technique, which bypasses locks, to address synchronization issues through transactions. A way to reduce conflicts is by adjusting the parallelism, as a suitable parallelism can maximize program performance. However, there is no universal rule to decide the best parallelism for a program from an offline view. Furthermore, an offline tuning is costly and error-prone. Hence, it becomes necessary to adopt a dynamical tuning-configuration strategy to better manage a STM system. Autonomic control techniques begin to receive attention in computing systems recently. Control technologies offer designers a framework of methods and techniques to build autonomic systems with well-mastered behaviors. The key idea of autonomic control is to implement feedback control loops to design safe, efficient and predictable controllers, which enable monitoring and adjusting controlled systems dynamically while keeping overhead low. We propose to design feedback control loops to automate the choice of parallelism at runtime and diminish program execution time.

In the context of the action-team HPES of the Labex Persyval-lab[3] (see 9.1), this work is performed in cooperation with LIG (J.F. Méhaut) in the framework of the PhD of N. Zhou.

7.3.4. Control of smart environments

Participants: Adja Sylla, Mengxuan Zhao, Eric Rutten, Hassane Alla [Gipsa-lab].

[3] https://persyval-lab.org/en/sites/hpes

7.3.4.1. Generic supervision architecture

New application domains of control, such as in the Internet of Things (IoT) and Smart Environments, require generic control rules enabling the systematization and the automation of the controller synthesis. We worked on an approach for the generation of Discrete Supervisory Controllers for these applications. A general modeling framework is proposed for the application domain of smart home. We formalize the design of the environment manager as a Discrete Controller Synthesis (DCS) problem, w.r.t. multiple constraints and objectives, for example logical issues of mutual exclusion, bounding of power peaks. We validate our models and manager computations with the BZR language and an experimental simulator This work was performed in cooperation with Orange labs (G. Privat) in the framework of the Cifre PhD of M. Zhao [8].

7.3.4.2. Rule-based specification

In the context of IoT applications like mart home environments, the rules for programming in the LINC framework are used as a flexible tool to govern the relations between sensors and actuators. Runtime coordination and formal analysis becomes a necessity to avoid side effects mainly when applications are critical. In cooperation with CEA LETI/DACLE, we are working on a case study for safe applications development in IoT and smart home environments [17].

8. Bilateral Contracts and Grants with Industry

8.1. CIFRE PhD grant Orange

This Cifre PhD started in the beginning of 2012, and was defended in may 2015, on the topic of "Discrete Control in the Internet of things and Smart Environments through a Shared Infrastructure" [8]. Hassane Alla and Eric Rutten advised the PhD student for 10%.

One result of this cooperation is that a patent deposited at the INPI on "Configuration automatique du controle discret d'entites physiques dans un systeme de supervision et de controle", by Gilles Privat et Mengxuan Zhao (Orange labs), Hassane Alla (Gipsa-lab), Eric Rutten (Inria).

8.2. Bilateral Grants with Industry

Our cooperation with CEA LETI/LIST DACLE at Grenoble Minatec is bilateral, involving the CEA PhD grant of Adja Sylla, to work with F. Pacull and M. Louvel on high-level programming on top of a rule-based middleware.

9. Partnerships and Cooperations

9.1. Regional Initiatives

The Labex Persyval-lab is a large regional initiative, supported by ANR, where we are contributing through two projects:

9.1.1. Equipe-action HPES

This project (2013-17) groups members from Inria, LIG, Gipsa-lab, TIMA and Gipsa-lab, around the topic of High-Performance Computing benefitting from technologies originally developed for Embedded Systems. Ctrl-A is directly involved in the co-advising of the PhD of Naweiluo Zhou, with J.F. Méhaut (LIG), on the topic of autonomic management of software transactional memory mechanisms: https://persyval-lab.org/en/sites/hpes

9.1.2. Projet Exploratoire CASE

This project (2015-16) grouped members from Inria, LIG, Gipsa-lab and CEA LETI/DACLE and concerned the general topic of Control techniques for Autonomic Smart Environments, with a special emphasis on relating discrete and stochastic control models with middleware platforms applied to smart environments. It enables us to hire two Masters students for 2016.

9.2. National Initiatives

9.2.1. ANR

HPeC is an ANR project on Self-Adaptive, Energy Efficient High Performance Embedded Computing, with a UAV case study. The Coordinator is Lab-STICC / MOCS (Lorient / Brest), and the duration: 42 month from october 2015. Others Partners are: Inria Rennes, IRIT, Eolas.

In Ctrl-A, it is funding a PhD thesis or a post-doc position, to be hired in Grenoble and co-adivsed with Lorient. Another PhD based in Brest is co-advised by Stéphane Mocanu.

9.2.2. Informal National Partners

We have contacts with colleagues in France, in addition to the cooperation mentioned before, and with whom we are submitting collaboration projects, co-organizing events and workshops, etc. They feature : Avalon Inria team in Lyon (F. Desprez), LIP6 (J. Malenfant), Scales Inria team in Sophia-Antipolis (L. Henrio), LIRRM in Montpellier (A. Gamatié, K. Godary, D. Simon), IRISA/Inria Rennes (J. Buisson, J.L. Pazat, ...), Telecom Paris-Tech (A. Diaconescu, E. Najm), LAAS (Thierry Monteil), LURPA ENS Cachan (J.M. Faure, J.J. Lesage), . . .

9.2.3. Informal National Industrial Partners

We have ongoing discussions with several industrial actors in our application domains, some of them in the framework of cooperation contracts, other more informal: Eolas/Business decision (G. Dulac), ST Microelectronics (V. Bertin), Schneider Electric (C. El-Kaed, P. Nappey, M. Pitel), Orange labs (J. Pulou, G. Privat).

9.3. International Initiatives

9.3.1. Inria International Partners

9.3.1.1. Informal International Partners

We have ongoing relations with international colleagues in the emerging community on our topic of control for computing e.g., in Sweden at Lund (K.E. Arzen, M. Maggio) and Linnaeus Universities (D. Weyns, N. Khakpour), in the Netherlands at CWI/leiden University (F. Arbab), in China at Heifei University (Xin An), in Italy at University Milano (C. Ghezzi, A. Leva), in the USA at Ann Arbor University (S. Lafortune) and UMass (P. Shenoy, E. Cecchet).

9.3.2. Participation In other International Programs

Eric Rutten is a member of the IFAC Technical Committee 1.3 on Discrete Event and Hybrid Systems, for the 2011-2014 triennium, and for the 2014-2017 triennum http://tc.ifac-control.org/1/3 ; and of the IEEE Control Systems Society Discrete Event Systems Technical Committee http://discrete-event-systems.ieeecss.org.

10. Dissemination

10.1. Promoting Scientific Activities

10.1.1. Scientific events organisation

10.1.1.1. General chair, scientific chair

Bogdan Robu, Stéphane Mocanu and Eric Rutten organized the 3rd Grenoble Workshop on Autonomic Computing and Control Wednesday October 7th, 2015, GIPSA-lab, Seminary room (2nd floor B208 - Grenoble campus) https://team.inria.fr/ctrl-a/members/eric-rutten/autoctrl/

Eric Rutten organized the DCDS 2015 Special Session on Dependable Discrete control for adaptive and reconfigurable computing systems http://www.gdl.cinvestav.mx/dcds2015/ may 2015
proposed the WODES 2016 Special Session on Discrete Control for Adaptive and Reconfigurable Computing Systems, with H. Marchand http://wodes2016.diee.unica.it/home1-6-7.html
organized the Seminar (Grenoble and Rhône-Alpes region, Lyon, Valence, Annecy) on Supervisory control Techniques for Autonomic, Adaptive and Reconfigurable computing Systems https://team.inria.fr/ctrl-a/members/eric-rutten/labex/staars-labex.html

10.1.1.2. Member of the organizing committees

Eric Rutten was workshops co-chair for the 12th IEEE International Conference on Autonomic Computing. ICAC'2015, July 7-10, 2015 in Grenoble, France. http://icac2015.imag.fr

10.1.2. Scientific events selection

10.1.2.1. Chair of conference program committees

Eric Rutten was program co-chair of the IEEE International Conference on Cloud and Autonomic Computing, CAC 2015, MIT, USA, September 2015 [19] (http://autonomic-conference.org), Part of FAS* - Foundation and Applications of Self* Computing Conferences, Collocated with: the IEEE Self-Adaptive and Self-Organizing System Conference, the IEEE Peer-to-Peer Computing Conference.

10.1.2.2. Member of the conference program committees

Eric Rutten is PC member for:

- 13th IEEE International Conference on Autonomic Computing (ICAC 2016) Wuerzburg, Germany, July 19-22, 2016 (http://icac2016.uni-wuerzburg.de/) ;
- 13th International Workshop on Discrete Event Systems (WODES 2016), Xi'an, China on May 30 - June 1, 2016 (http://wodes2016.diee.unica.it) ;
- SEfSAS Book 3 (Software Engineering for Self-Adaptive Systems: Assurances) Volume 3 to be published by Springer ;
- CTSE2015 (First International Workshop on Control Theory for Software Engineering), in conjunction with 10th Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT Symposium on the Foundations of Software Engineering BERGAMO, ITALY, August 31 – September 4 (http://www.softwarecontrol.org/) ;
- MSR 2015, 10ème Colloque Francophone sur la Modélisation des Systèmes Réactifs, Nancy, du 18 au 20 Novembre 2015 (http://msr2015.loria.fr/) ;
- 2nd International Workshop on Adaptive Discrete Event Control Systems (ADECS 2015) that will be held June 22, 2014 at Brussels (http://adecs2015.cnam.fr/) A satellite event of PETRI NETS 2015 and ACSD 2015,Brussels, Belgium, Monday June 22, 2015 ;
- 3rd INTERNATIONAL WORKSHOP ON SELF-AWARE INTERNET OF THINGS 2015, in conjunction with ICAC 2015, The 12th IEEE International Conference on Autonomic Computing, July 7-10 2015, Grenoble, France (http://clout-project.eu/international-workshop-on-self-aware-internet-of-things-2015/) ;
- 5th IFAC Conference on Analysis and Design of Hybrid Systems (ADHS) October 14-16, 2015, Georgia Tech, Atlanta, Georgia, USA (http://adhs15.gatech.edu) ;
- 12th IEEE International Conference on Autonomic Computing. ICAC 2015, July 7-10, 2015 in Grenoble, France (http://icac2015.imag.fr) ;
- 5th IFAC international workshop on Dependable Control of Discrete Systems , DCDS, Cancun, Mexico, may 27-29, 2015 (http://www.gdl.cinvestav.mx/dcds2015) ;
- The 2015 Electronic System Level Synthesis Conference, ESLsyn 2015, co-located with the Design Automation Conference (DAC), San Francisco (CA), USA, June 10-11th, 2015. (http://www.ecsi.org/eslsyn)

10.1.3. Journal

10.1.3.1. Reviewer - Reviewing activities

Eric Rutten is reviewer for Journal of Systems and Software ; j. Microprocessors and Microsystems ; and guest co-editor with P. Shenoy for a ICCAC 2015 Best Papers special issue of the Cluster Computing journal (The Journal of Networks, Software Tools and Applications, Springer http://link.springer.com/journal/10586)

10.1.4. Invited talks

Eric Rutten was invited to give talks at : 11th International Conference on Distributed Computing and Internet Technology (ICDCIT-2015) http://www.icdcit.ac.in/, Bhubaneswar, India, February 5–8, 2015 ; College of Information and Computer Sciences (CICS), University of Massachusetts Amherst, USA, 28 sept.2015 https://www.cics.umass.edu/event/discrete-control-based-design-adaptive-and-autonomic-computing-systems

10.1.5. Scientific expertise

Eric Rutten was in the hiring commission for Inria CR2 research positions at Inria Grenoble-Rhône-Alpes and evaluated pre-proposals of projects submitted to the ANR generic call 2016. ; the CRI PILSI board of directors http://cri-grenoble.com and in charge of the development of scientific relationships between CEA and Inria in Grenoble on the topics of software and computer science ; PCS (Pervasive Computing Systems) committee of the Labex Persyval-lab.

10.2. Teaching - Supervision - Juries

10.2.1. Supervision

- PhD in progress : Adja Ndeye SYLLA ; Generation of coordination rules from an automaton, in the context of the redeployment of distributed contra; applications ; feb. 2015 ; E. Rutten, F. Pacull and M. Louvel (CEA)
- PhD in progress : Naweiluo Zhou ; Application-aware policies and control for transactional memory systems ; oct. 2013 ; E. Rutten, G. Delaval, J.F. Mehaut (LIG)

10.2.2. Juries

Eric Rutten was on the PhD defense committees for : Cédric Eichler, LAAS Toulouse, 9 june 2015 ; Mathilde Machin, LAAS Toulouse, 12 nov 2015.

11. Bibliography

Major publications by the team in recent years

[1] X. An, E. Rutten, J.-P. Diguet, N. Le Griguer, A. Gamatie. *Discrete Control for the Autonomic Management of Dynamically Partially Reconfigurable FPGA Architectures*, in "Proceedings of the 13th International Conference on Autonomic Computing (ICAC'13), june 26-28, San Jose, California", 2013

[2] T. Bouhadiba, Q. Sabah, G. Delaval, E. Rutten. *Synchronous Control of Reconfiguration in Fractal Component-based Systems – a Case Study*, in "Proceedings of the International Conference on Embedded Software. EMSOFT, Taipei, Taiwan. October 9-14", 2011, pp. 309–318, http://dx.doi.org/10.1145/2038642.2038690

[3] G. Delaval, H. Marchand, E. Rutten. *Contracts for Modular Discrete Controller Synthesis*, in "Proceedings of the ACM SIGPLAN/SIGBED Conference on Languages, Compilers and Tools for Embedded Systems, LCTES, in conjunction with CPSWeek, Stockholm, Sweden, April 12-16", 2010, pp. 57–66, http://doi.acm.org/10.1145/1755951.1755898

[4] G. DELAVAL, E. RUTTEN. *Reactive model-based control of reconfiguration in the Fractal component-based model*, in "Proceedings of the 13th International Symposium on Component Based Software Engineering (CBSE), Prague, Czech Republic, 23-25 June", 2010, pp. 93–112, best paper award, http://dx.doi.org/10.1007/978-3-642-13238-4_6

[5] G. DELAVAL, E. RUTTEN, H. MARCHAND. *Integrating Discrete Controller Synthesis in a Reactive Programming Language Compiler*, in "journal of Discrete Event Dynamic System, jDEDS, special issue on Modeling of Reactive Systems", 2013, vol. 23, n^o 4, pp. 385-418, http://dx.doi.org/10.1007/s10626-013-0163-5

[6] E. DUMITRESCU, A. GIRAULT, H. MARCHAND, E. RUTTEN. *Multicriteria optimal discrete controller synthesis for fault-tolerant tasks*, in "Proceedings of the 10th International Workshop on Discrete Event Systems (WODES), Berlin, Germany, August 30 – September 1", 2010, pp. 366–373 [*DOI :* 10.3182/20100830-3-DE-4013.00059], http://hal.inria.fr/inria-00510019

[7] S. M.-K. GUEYE, N. DE PALMA, E. RUTTEN. *Coordination Control of Component-based Autonomic Administration Loops*, in "Proceedings of the 15th International Conference on Coordination Models and Languages, COORDINATION, 3-6 June, Florence, Italy", 2013, http://dx.doi.org/10.3182/20110828-6-IT-1002.01548

Publications of the year

Doctoral Dissertations and Habilitation Theses

[8] M. ZHAO. *Discrete Control in the Internet of things and Smart Environments through a Shared Infrastructure*, Université Grenoble Alpes, May 2015, https://tel.archives-ouvertes.fr/tel-01163806

Articles in International Peer-Reviewed Journals

[9] X. AN, A. GAMATIE, E. RUTTEN. *High-level design space exploration for adaptive applications on multiprocessor systems-on-chip*, in "Journal of Systems Architecture", March 2015, vol. 61, n^o 3-4, 13 p. [*DOI :* 10.1016/J.SYSARC.2015.02.002], https://hal.inria.fr/hal-01162488

[10] N. BERTHIER, É. RUTTEN, N. DE PALMA, S. M.-K. GUEYE. *Designing Autonomic Management Systems by using Reactive Control Techniques*, in "IEEE Transactions on Software Engineering", December 2015, 18 p. , https://hal.inria.fr/hal-01242853

[11] N. KHAKPOUR, F. ARBAB, E. RUTTEN. *Synthesizing structural and behavioral control for reconfigurations in component-based systems*, in "Formal Aspects of Computing", December 2015 [*DOI :* 10.1007/s00165-015-0346-Y], https://hal.inria.fr/hal-01247524

[12] G. OCHOA RUIZ, G. SÉBASTIEN, F. DE LAMOTTE, É. RUTTEN, E.-B. BOURENNANE, J.-P. DIGUET, G. GOGNIAT. *An MDE Approach for Rapid Prototyping and Implementation of Dynamic Reconfigurable Systems*, in "ACM Transactions on Design Automation of Electronic Systems (TODAES)", June 2015, https://hal.archives-ouvertes.fr/hal-01172123

Invited Conferences

[13] X. AN, G. DELAVAL, J.-P. DIGUET, A. GAMATIE, S. M.-K. GUEYE, H. MARCHAND, N. DE PALMA, E. RUTTEN. *Discrete Control-Based Design of Adaptive and Autonomic Computing Systems*, in "ICDCIT: International Conference on Distributed Computing and Internet Technology", Bhubaneswar, India, 11th Interna-

tional Conference on Distributed Computing and Internet Technology, ICDCIT 2015, Springer, February 2015, vol. LNCS, n° 8956 [*DOI :* 10.1007/978-3-319-14977-6_6], https://hal.archives-ouvertes.fr/hal-01116015

International Conferences with Proceedings

[14] F. ALVARES DE OLIVEIRA JR., E. RUTTEN, L. SEINTURIER. *High-level Language Support for the Control of Reconfigurations in Component-based Architectures*, in "9th European Conference on Software Architecture (ECSA)", Dubrovnick, Croatia, LNCS, Springer, September 2015, vol. 9278, pp. 285-293, https://hal.inria.fr/hal-01160612

[15] F. ALVARES DE OLIVEIRA JR., É. RUTTEN, L. SEINTURIER. *Behavioural Model-based Control for Autonomic Software Components*, in "12th IEEE International Conference on Autonomic Computing (ICAC)", Grenoble, France, IEEE, July 2015, https://hal.inria.fr/hal-01143196

[16] G. DELAVAL, S. M.-K. GUEYE, E. RUTTEN. *Distributed Execution of Modular Discrete Controllers for Data Center Management*, in "Proc. of the 5th IFAC international workshop on Dependable Control of Discrete Systems, DCDS'15", Cancun, France, May 2015, https://hal.inria.fr/hal-01161509

National Conferences with Proceedings

[17] A. N. SYLLA, M. LOUVEL, F. PACULL, E. RUTTEN. *Génération de règles de coordination à partir de réseaux de Pétri colorés*, in "10è Colloque Francophone sur la Modélisation des Systèmes Réactifs, MSR'15", Nancy, France, November 2015, https://hal.inria.fr/hal-01218242

Conferences without Proceedings

[18] F. ALVARES DE OLIVEIRA JR., E. RUTTEN, L. SEINTURIER. *Modèles comportementaux pour le contrôle de composants logiciels autonomes*, in "MSR 2015 - Modélisation des Systèmes Réactifs", Nancy, France, S. MERZ, J.-F. PÉTIN (editors), November 2015, https://hal.inria.fr/hal-01224291

Scientific Books (or Scientific Book chapters)

[19] *Proceedings of the IEEE International Conference on Cloud and Autonomic Computing (ICCAC), 2015*, September 2015, https://hal.inria.fr/hal-01218964

Research Reports

[20] 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 ; Inria, January 2015, n° RR-8669, https://hal.inria.fr/hal-01103548

[21] E. RUTTEN, N. MARCHAND, D. SIMON. *Feedback Control as MAPE-K loop in Autonomic Computing*, Inria Sophia Antipolis - Méditerranée ; Inria Grenoble - Rhône-Alpes, December 2015, n° RR-8827, draft soumis à LNCS, http://hal-lirmm.ccsd.cnrs.fr/lirmm-01241594

References in notes

[22] M. BEREKMERI, D. SERRANO, S. BOUCHENAK, N. MARCHAND, B. ROBU. *A Control Approach for Performance of Big Data Systems*, in "IFAC World Congress", 2014, hal-00940282, http://hal.archives-ouvertes.fr/hal-00940282

[23] M. BEREKMERI, D. SERRANO, S. BOUCHENAK, N. MARCHAND, B. ROBU. *Application du contrôle pour garantir la performance des systèmes Big Data*, in "Conférence francophone d'informatique en parallélisme, architecture et systeme (ComPAS)", 2014, hal-00982404, http://hal.archives-ouvertes.fr/hal-00982404