

RESEARCH CENTER

FIELD Networks, Systems and Services, Distributed Computing

Activity Report 2017

Section Software

Edition: 2018-02-19

DISTRIBUTED SYSTEMS AND MIDDLEWARE
1. ASAP Project-Team
2. COAST Project-Team
3. CTRL-A Project-Team
4. MIMOVE Team
5. MYRIADS Project-Team
6. REGAL Project-Team
7. SPIRALS Project-Team
8. WHISPER Project-Team
DISTRIBUTED AND HIGH PERFORMANCE COMPUTING
9. ALPINES Project-Team
10. AVALON Project-Team 22
11. DATAMOVE Project-Team
12. HIEPACS Project-Team
13. KERDATA Project-Team 33
14. POLARIS Team
15. ROMA Project-Team
16. STORM Project-Team 40
17. TADaaM Project-Team
DISTRIBUTED PROGRAMMING AND SOFTWARE ENGINEERING
18. ASCOLA Project-Team 49
19. DIVERSE Project-Team 51
20. FOCUS Project-Team
21. INDES Project-Team
22. PHOENIX Project-Team
23. RMOD Project-Team
24. TACOMA Team
NETWORKS AND TELECOMMUNICATIONS
25. AGORA Team
26. COATI Project-Team
27. DANTE Project-Team
28. DIANA Project-Team
29. DIONYSOS Project-Team
30. DYOGENE Project-Team
31. EVA Project-Team
32. FUN Project-Team
33. GANG Project-Team
34. INFINE Project-Team
35. MADYNES Team
36. NEO Project-Team
37. RAP2 Team (section vide)

4 Distributed Systems and middleware - Software and Platforms - Project-Team ASAP

38. SOCRATE Project-Team		94
--------------------------	--	----

ASAP Project-Team

5. New Software and Platforms

5.1. WebGC

Web-based Gossip Communication

KEYWORDS: WebRTC - Recommendation systems - Decentralized architectures - Personalized systems - Web - Peer-to-peer - Gossip protocols - Epidemic protocols - Decentralized web

SCIENTIFIC DESCRIPTION: The library currently includes the implementation of two peer sampling protocols, Cyclon and the generic peer-sampling protocol from, as well as a clustering protocol. All protocols implement a common GossipProtocol "interface"

FUNCTIONAL DESCRIPTION: WebGC consists of a WebRTC-based library that supports gossip-based communication between web browsers and enables them to operate with Node-JS applications. WebGC comprises the implementation of standard gossip protocols such as Peer Sampling or Clustering, and simplifies the development of new protocols. It comprises a decentralized signaling service that makes it easier to build completely decentralized browser-based applications.

- Participants: Anne-Marie Kermarrec, Davide Frey, Matthieu Simonin and Raziel Carvajal Gomez
- Contact: Davide Frey

5.2. Asapknn (MediEgo)

KEYWORDS: Widget web - Social network - Recommendation

FUNCTIONAL DESCRIPTION: Asapknn (MediEgo) is a solution for content recommendation based on the users navigation history. The solution 1) collects the usages of the Web users and store them in a profile, 2) uses this profile to associate to each user her most similar users, 3) leverages this implicit network of close users in order to infer their preferences and recommend advertisements and recommendations. MediEgo achieves scalability using a sampling method, which provides very good results at a drastically reduced cost.

- Participants: Anne Marie Kermarrec, Antoine Boutet, Arnaud Jegou, Davide Frey, Jacques Falcou, Jean-Francois Verdonck, Rachid Guerraoui and Sébastien Campion
- Partner: EPFL Ecole Polytechnique Fédérale de Lausanne
- Contact: Sébastien Campion

5.3. YALPS

KEYWORDS: Simulator - Peer-to-peer - Experimentation - Nat traversal - Traffic-shaping - Deployment FUNCTIONAL DESCRIPTION: YALPS is an open-source Java library designed to facilitate the development, deployment, and testing of distributed applications. Applications written using YALPS can be run both in simulation and in real-world mode without changing a line of code or even recompiling the sources. A simple change in a configuration file will load the application in the proper environment. A number of features make YALPS useful both for the design and evaluation of research prototypes and for the development of applications to be released to the public. Specifically, YALPS makes it possible to run the same application as a simulation or in a real deployment. Applications communicate by means of application-defined messages which are then routed either through UDP/TCP or through YALPS's simulation infrastructure. In both cases, YALPS's communication layer offers features for testing and evaluating distributed protocols and applications. Communication channels can be tuned to incorporate message losses or to constrain their outgoing bandwidth. Finally, YALPS includes facilities to support operation in the presence of NATs and firewalls using relaying and NAT-traversal techniques. The implementation of YALPS includes approximately 16K lines of code, and is used in several projects by ASAP, including HEAP, AllYours-P2P, and Behave.

- Participants: Anne Marie Kermarrec, Arnaud Jegou, Davide Frey, Heverson Borba Ribeiro and Maxime Monod
- Contact: Davide Frey
- URL: http://yalps.gforge.inria.fr/

5.4. GossipLib

KEYWORDS: Nat traversal - Epidemic protocols - Gossip protocols - Overlay maintenance - Peer-to-peer - Dissemination

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

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

RELEASE FUNCTIONAL DESCRIPTION: Library for gossip-based applications and experiments

- Participants: Anne Marie Kermarrec, Davide Frey, Ilham Ikbal, Imane Al Ifdal and Ribeiro Heverson
- Contact: Davide Frey
- URL: http://gossiplib.gforge.inria.fr/

COAST Project-Team

4. New Software and Platforms

4.1. BeGoood

FUNCTIONAL DESCRIPTION: BeGoood is a generic system for managing non-regression tests on knowledge bases. BeGoood allows to define test plans in order to monitor the evolution of knowledge-bases. Any system answering queries by providing results in the form of set of strings can be tested with BeGoood. BeGoood has been developed following a REST architecture and is independent of any application domain. BeGoood is a part of the Kolflow infrastructure.

- Participant: Gérôme Canals
- Contact: Gérôme Canals
- URL: https://github.com/kolflow/begoood

4.2. MUTE

Multi-User Text Editor

FUNCTIONAL DESCRIPTION: MUTE (Multi-User Text Editor) is a web-based text editing tool that allows to edit documents collaboratively in real-time. It implements our recent work on collaborative editing algorithms and more specifically the LogootSplit+ approach. Compared to existing web-based collaborative text editing tool this editor does not require a powerful central server since the server is not performing any computation and acts as a simple broadcast server. Our editor offers support for working offline while still being able to reconnect at a later time.

- Participants: Claudia-Lavinia Ignat, François Charoy, Gérald Oster and Luc André
- Contact: Gérald Oster
- URL: https://github.com/coast-team/mute-demo/

4.3. Replication Benchmarker

FUNCTIONAL DESCRIPTION: The Replication Benchmarker is a performance evaluation framework for optimistic replication mechanisms used in collaborative applications. It contains a library of implementation of several CRDT (Commutative Replicated Data Type) and OT (Operational Transformation) algorithms for different data types: text, set, trees. The framework is able to evaluate the performance of comparable algorithms on different corpus of events traces. These events traces can be produced randomly according to different parameters, can be extracted from real real-time editing session that have been recorded, or can be automatically extracted from distributed version control repositories such as the one produced with Git. Performances of the algorithms are measured in term of execution time, memory footprint and merge result quality (compared to manual merge history stored in git repositories).

- Participants: Gérald Oster, Mehdi Ahmed-Nacer and Pascal Urso
- Contact: Pascal Urso
- URL: https://github.com/score-team/replication-benchmarker/

4.4. Rivage

Real-tIme Vector grAphic Group Editor

FUNCTIONAL DESCRIPTION: Rivage is a real-time collaborative graphical editor. Several users can edit at the same time and in real-time a graphical document, user changes being immediately seen by the other users. The editor relies on a peer-to-peer architecture where users can join and leave the group at any time. Each user has a copy of the shared document and user changes on the document copies are merged in real-time by using a CRDT (Commutative Replicated Data Type) algorithm.

- Participant: Claudia-Lavinia Ignat
- Contact: Claudia-Lavinia Ignat
- URL: https://github.com/stephanemartin/rivage/

CTRL-A Project-Team

5. New Software and Platforms

5.1. Heptagon

KEYWORDS: Compilers - Synchronous Language - Controller synthesis

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.

The current version of the compiler includes the following features: - Inclusion of discrete controller synthesis within the compilation: the language is equipped with a behavioral contract mechanisms, where assumptions can be described, as well as an "enforce" property part. The semantics of this latter is that the property should be enforced by controlling the behaviour of the node equipped with the contract. This property will be enforced by an automatically built controller, which will act on free controllable variables given by the programmer. This extension has been named BZR in previous works. - Expression and compilation of array values with modular memory optimization. The language allows the expression and operations on arrays (access, modification, iterators). With the use of location annotations, the programmer can avoid unnecessary array copies.

- Participants: Adrien Guatto, Brice Gelineau, Cédric Pasteur, Eric Rutten, Gwenaël Delaval, Léonard Gérard and Marc Pouzet
- Partners: UGA ENS Paris Inria LIG
- Contact: Gwenaël Delaval
- URL: http://heptagon.gforge.inria.fr

MIMOVE Team

6. New Software and Platforms

6.1. SoundCity - Ambiciti

KEYWORDS: Crowd-sensing - Mobile application

FUNCTIONAL DESCRIPTION: Is your exposure to noise too high on certain days? How is air pollution in your street? Will air quality improve in the next hours? Do you want to measure the noise pollution on the way between your home and your office? What pollution levels are considered harmful for your health? Ambiciti (previously SoundCity) provides answers to these questions and many others.

Noise pollution. Ambiciti (previously called SoundCity) measures the actual noise levels to which you are exposed. Ambiciti can monitor noise levels throughout the day and inform you about your instantaneous, hourly and daily exposures. If you want to contribute to the improvement of the noise map in your city, you can anonymously send your measurements.

Air pollution. Ambiciti computes the air quality index in your region or at the exact location where you stand. You can also access to forecasts and find information about the main pollutants. Depending on your location, you may have access to hourly air quality maps, at street resolution, in real time and for the next two days. Currently, only Paris (France) enjoys such fine maps, but other cities are on the way to be included.

Since 2017, the software is exclusively licensed to the Ambiciti start-up company.

- Authors: Fadwa Rebhi, Pierre-Guillaume Raverdy, Cong Kinh Nguyen, Rajiv Bhatia, Valérie Issarny and Vivien Mallet
- Partners: Ambientic The Civic Engine
- Contact: Valérie Issarny

6.2. SocialBus

Universal Social Network Bus

KEYWORDS: Middleware - Interoperability - Social networks - Software Oriented Service (SOA)

FUNCTIONAL DESCRIPTION: Online social network services (OSNSs) have become an integral part of our daily lives. At the same time, the aggressive market competition has led to the emergence of multiple competing siloed OSNSs that cannot interoperate. As a consequence, people face the burden of creating and managing multiple OSNS accounts and learning how to use them, to stay connected. The goal of the Universal Social Network Bus (USNB) is to relieve users from such a burden, letting them use their favorite applications to communicate.

Social Entities. Social entities can be humans or systems. They can create a profile in the USNB and link it with their OSNSs identities. Social entities can also choose the OSNS identity they want to use when contacted through the USNB or specify one or more OSNS identities for message reception concerning specific events or senders.

Personae. Personae are USNB entities interacting with users within concrete OSNSs or systems, achieving interoperability between heterogeneous OSNSs. New personae can be developed, registered in the USNB, discovered and used to include additional OSNSs.

Privacy & Security. The USNB is designed to be as less intrusive as possible. It does not ask users their credentials nor any kind of authorization concerning their OSNS accounts.

- Authors: Rafael Angarita Arocha, Nikolaos Georgantas and Valérie Issarny
- Contact: Valérie Issarny
- URL: https://gitlab.inria.fr/usnb/universal-social-network-bus

6.3. WeBrowse

KEYWORDS: Web Usage Mining - Content analysis - Recommendation systems

FUNCTIONAL DESCRIPTION: The amount of information available on the web today, and the fast rate with which new information appears, overwhelm most users. The goal of our research is to assist Web users in discovering content. One of the most powerful means today to help people discover new web content is sharing between members of online communities. In the case of communities of a place (e.g., people who live, study, or work together) people share common interests, but often fail to actively share content. To address this problem, we have developed WeBrowse, a passive crowdsourced content discovery system for communities of a place.

WeBrowse leverages the passive observation of web-clicks (i.e., the URLs users intentionally visit) as an indication of users' interest in a piece of content. Intuitively, the more users click on a URL, the higher the interest in the content on the corresponding page. Our approach is then to leverage the collective clicks in a community to automatically discover relevant content to promote to users of the community.

To implement passive crowdsourcing, one must be in a position to observe the aggregated web-clicks of the community. Luckily, in many communities of a place, users will connect to the Internet from the same network, such as, e.g., the campus/enterprise network or the network of a residential Internet Service Provider (ISP) in a neighborhood. WeBrowse (i) observes web packets flowing through a network link, (ii) passively extracts HTTP logs (i.e., streams recording the headers of HTTP requests), and (iii) detects and decides on-the-fly the set of URLs to show to users.

- Contact: Renata Cruz Teixeira
- URL: https://team.inria.fr/muse/webrowse-info-page/

6.4. TA

TA - Traffic Analysis

KEYWORDS: Quality of Experience - Network monitoring - Video analysis

FUNCTIONAL DESCRIPTION: System running at the home getaway that analyzes traffic generated by DASH on-demand and live video streams. The system tracks traffic patterns to infer key video QoE metrics such as average bitrate and re-buffering events. Moreover, the system exploits novel algorithms that use probing techniques, i.e. lightweight pings and traceroutes, to detect possible congestion location.

- Participants: Francesco Bronzino and Renata Cruz Teixeira
- Contact: Francesco Bronzino

6.5. HostView

KEYWORDS: Quality of Experience - Network monitoring

FUNCTIONAL DESCRIPTION: End-host performance monitoring and user feedback reporting.

- Participants: Anna-Kaisa Pietilainen, Francesco Bronzino, George Rosca and Renata Cruz Teixeira
- Contact: Renata Cruz Teixeira
- URL: https://github.com/inria-muse/hostview-win

6.6. VSB

eVolution Service Bus

KEYWORDS: Service and Thing choreographies - Middleware protocol interoperability - Enterprise service bus

FUNCTIONAL DESCRIPTION: VSB is a development and runtime environment dedicated to complex distributed applications of the Future Internet. Such applications are open, dynamic choreographies of extremely heterogeneous services and Things, including lightweight embedded systems (e.g., sensors, actuators and networks of them), mobile systems (e.g., smartphone applications), and resource-rich IT systems (e.g., systems hosted on enterprise servers and Cloud infrastructures). VSB's objective is to seamlessly interconnect, inside choreographies, services and Things that employ heterogeneous interaction protocols at the middleware level, e.g., SOAP Web services, REST Web services, Things using CoAP. This is based on runtime conversions between such protocols, with respect to their primitives and data type systems, while properly mapping between their semantics. This also includes mapping between the public interfaces of services/Things, regarding their operations and data, from the viewpoint of the middleware: the latter means that operations and data are converted based on their middleware-level semantics, while their business semantics remains transparent to the conversion. VSB follows the well-known Enterprise Service Bus (ESB) paradigm. We propose a generic interface description, which we call GIDL, for application components that employ VSB. Based on GIDL, we enable automated synthesis of binding components for connecting heterogeneous services and Things onto VSB.

- Participants: Georgios Bouloukakis, Nikolaos Georgantas and Patient Ntumba
- Contact: Nikolaos Georgantas
- URL: https://github.com/sidutta/vsb

MYRIADS Project-Team

6. New Software and Platforms

6.1. ConPaaS

KEYWORDS: Cloud computing - PaaS SCIENTIFIC DESCRIPTION: Contact:

Guillaume Pierre, Guillaume.Pierre@irisa.fr URL:

http://www.conpaas.eu/ Status:

Version 1.4.2 License:

BSD Presentation:

ConPaaS [60] is a runtime environment for hosting applications in the cloud. It aims at offering the full power of the cloud to application developers while shielding them from the associated complexity of the cloud. ConPaaS is designed to host both high-performance scientific applications and online Web applications. It automates the entire life-cycle of an application, including collaborative development, deployment, performance monitoring, and automatic scaling. This allows developers to focus their attention on application-specific concerns rather than on cloud-specific details. Active contributors (from the Myriads team):

Eliya Buyukkaya, Ancuta Iordache, Morteza Neishaboori, Guillaume Pierre, Dzenan Softic, Genc Tato, Teodor Crivat. Impact:

ConPaaS is recognized as one of the major open-source PaaS environments. It is being developed by teams in Rennes, Amsterdan, Berlin and Ljubljana. Technology transfer of ConPaaS technology is ongoing in the context of the MC-DATA EIT ICT Labs project.

FUNCTIONAL DESCRIPTION: ConPaaS is a runtime environment for hosting applications in the cloud. It aims at offering the full power of the cloud to application developers while shielding them from the associated complexity of the cloud. ConPaaS is designed to host both high-performance scientific applications and online Web applications. It automates the entire life-cycle of an application, including collaborative development, deployment, performance monitoring, and automatic scaling. This allows developers to focus their attention on application-specific concerns rather than on cloud-specific details.

- Participants: Ancuta Iordache, Dzenan Softic, Eliya Buyukkaya, Genc Tato, Guillaume Pierre, Morteza Neishaboori and Teodor Crivat
- Contact: Guillaume Pierre
- URL: http://www.conpaas.eu/

6.2. GinFlow

KEYWORDS: Dynamic adaptation - Distributed Applications - Distributed - Distributed computing - Workflow - Framework

FUNCTIONAL DESCRIPTION: GinFlow decentralizes the coordination of the execution of workflow-based applications. GinFlow relyies on an architecture where multiple service agents (SA) coordinate each others through a shared space containing the workflow description and current status. GinFlow allows the user to define several variants of a workflow and to switch from one to the other during run time.

- Participants: Cédric Tedeschi, Hector Fernandez, Javier Rojas Balderrama, Matthieu Simonin and Thierry Priol
- Partner: Université de Rennes 1
- Contact: Cédric Tedeschi
- URL: http://ginflow.inria.fr

6.3. Merkat

FUNCTIONAL DESCRIPTION: Merkat is a platform that allows users of an organization to automatically manage and scale their applications while maximizing the infrastructure's utilization. Merkat is generic and extensible, allowing users to automate the application deployment and management process. Users have the flexibility to control how many resources are allocated to their applications and to define their own resource demand adaptation policies. Merkat applies an unique approach to multiplex the infrastructure capacity between the applications, by implementing a proportional-share market and allowing applications to adapt autonomously to resource price and their given performance objectives. The price of the acquired resources acts as a control mechanism to ensure that resources are distributed to applications according to the user's value for them. Merkat was evaluated on Grid'5000 with several scientific applications.

- Participants: Christine Morin, Nikolaos Parlavantzas and Stefania Costache
- Contact: Nikolaos Parlavantzas
- URL: http://www.irisa.fr/myriads/software/Merkat/

6.4. PaaSage Adapter

KEYWORDS: Cloud computing - Dynamic adaptation - Cloud applications management

FUNCTIONAL DESCRIPTION: The purpose of the Adapter is to transform the current configuration of a cloud application into a target configuration in an efficient and safe way. The Adapter is part of PaaSage, an open-source platform for modeling, deploying and executing applications on different clouds in an optimal manner. The Adapter has the following responsibilities: (1) validating reconfiguration plans, (2) applying the plans to the running system, and (3) maintaining an up-to-date representation of the current system state.

- Contact: Nikolaos Parlavantzas
- URL: https://team.inria.fr/myriads/software-and-platforms/paasage-adapter/

6.5. SAIDS

self-adaptable intrusion detection system

KEYWORDS: Cloud - Security

FUNCTIONAL DESCRIPTION: SAIDS is a self-adaptable intrusion detection system for IaaS clouds. To maintain an effective level of intrusion detection, SAIDS monitors changes in the virtual infrastructure of a Cloud environment and reconfigures its components (security probes) accordingly. SAIDS can also reconfigure probes in the case of a change in the list of running services.

- Authors: Anna Giannakou and Jean-Léon Cusinato
- Contact: Christine Morin

6.6. SimGrid

KEYWORDS: Large-scale Emulators - Grid Computing - Distributed Applications

SCIENTIFIC DESCRIPTION: SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. The simulation engine uses algorithmic and implementation techniques toward the fast simulation of large systems on a single machine. The models are theoretically grounded and experimentally validated. The results are reproducible, enabling better scientific practices.

Its models of networks, cpus and disks are adapted to (Data)Grids, P2P, Clouds, Clusters and HPC, allowing multi-domain studies. It can be used either to simulate algorithms and prototypes of applications, or to emulate real MPI applications through the virtualization of their communication, or to formally assess algorithms and applications that can run in the framework.

The formal verification module explores all possible message interleavings in the application, searching for states violating the provided properties. We recently added the ability to assess liveness properties over arbitrary and legacy codes, thanks to a system-level introspection tool that provides a finely detailed view of the running application to the model checker. This can for example be leveraged to verify both safety or liveness properties, on arbitrary MPI code written in C/C++/Fortran.

RELEASE FUNCTIONAL DESCRIPTION:

- Four releases in 2017. Major changes:
 - S4U: many progress, toward SimGrid v4.0. About 80% of the features offered by SimDag and MSG are now integrated, along with examples. Users can now write plugins to extend SimGrid.
 - SMPI: Support MPI 2.2, RMA support, Convert internals to C++.
 - Java: Massive memleaks and performance issues fixed.
 - New models: Multi-core VMs, Energy consumption due to the network
 - All internals are now converted to C++, and most of our internally developped data containers were replaced with std::* constructs.
 - (+ bug fixes, cleanups and documentation improvements)
- Participants: Adrien Lèbre, Arnaud Legrand, Augustin Degomme, Florence Perronnin, Frédéric Suter, Jean-Marc Vincent, Jonathan Pastor, Jonathan Rouzaud-Cornabas, Luka Stanisic, Mario Südholt and Martin Quinson
- Partners: CNRS ENS Rennes
- Contact: Martin Quinson
- URL: http://simgrid.gforge.inria.fr/

6.7. DiFFuSE

Distributed framework for cloud-based epidemic simulations

KEYWORDS: Simulation - Cloud

FUNCTIONAL DESCRIPTION: The DiFFuSE framework enables simulations of epidemics to take full advantage of cloud environments. The framework provides design support, reusable code, and tools for building and executing epidemic simulations. Notably, the framework automatically handles failures and supports elastic allocation of resources from multiple clouds.

- Contact: Nikolaos Parlavantzas
- URL: https://team.inria.fr/myriads/software-and-platforms/diffuse/

REGAL Project-Team

4. New Software and Platforms

4.1. Antidote

KEYWORDS: Distributed computing - Distributed Data Management - Cloud storage - Large scale FUNCTIONAL DESCRIPTION: Antidote is the flexible cloud database platform currently under development in the SyncFree and LightKone European projects. Antidote aims to be both a research platform for studying replication and consistency at the large scale, and an instrument for exploiting research results. The platform supports replication of CRDTs, in and between sharded (partitioned) data centres (DCs). The current stable version supports strong transactional consistency inside a DC, and causal transactional consistency between DCs. Ongoing research includes support for explicit consistency , for elastic version management, for adaptive replication, for partial replication, and for reconfigurable sharding.

- Participants: Marc Shapiro, Paolo Viotti, Alejandro Tomsic, Ilyas Toumlilt and Dimitrios Vasilas
- Partners: Université Catholique de Louvain (UCL), Louvain-la-Neuve,Belgium Universidade NOVA de Lisboa, Portugal Technische Universität Kaiserslautern (UniKL), Allemagne
- Contact: Marc Shapiro
- Publications: Bringing the cloud closer to users Write Fast, Read in the Past: Causal Consistency
 for Client-side Applications Extending Eventually Consistent Cloud Databases for Enforcing Numeric Invariants Designing a causally consistent protocol for geo-distributed partial replication

 Towards Fast Invariant Preservation in Geo-replicated Systems Putting Consistency back into
 Eventual Consistency The Case for Fast and Invariant-Preserving Geo-Replication Improving the
 scalability of geo-replication with reservations Conflict-free Replicated Data Types An encounter
 with Marc Shapiro and his SyncFree European project PhysiCS-NMSI: efficient consistent snap shots for scalable snapshot isolation Geo-Replication: Fast If Possible, Consistent If Necessary Cure: Strong semantics meets high availability and low latency Cure: Strong semantics meets high
 availability and low latency

4.2. CISE Tool

KEYWORDS: Distributed Applications - Program verification

FUNCTIONAL DESCRIPTION: Static analysis of the model of a distributed application, to prove (under the assumption of causal consistency) whether the invariants of the application are always satisfied, and to provide a counterexample if not.

- Participants: Sreeja Nair and Marc Shapiro
- Contact: Marc Shapiro
- Publications: Evaluation of the CEC (Correct Eventual Consistency) Tool The CISE Tool: Proving Weakly-Consistent Applications Correct - The CISE Tool: Proving Weakly-Consistent Applications Correct - CISE Safety Tool - 'Cause I'm Strong Enough: Reasoning about Consistency Choices in Distributed Systems - Putting Consistency back into Eventual Consistency

4.3. PUMA

Puma: pooling unused memory in virtual machines KEYWORDS: Virtualization - Operating system - Distributed systems - Linux kernel FUNCTIONAL DESCRIPTION: PUMA is a system that is based on a kernel-level remote caching mechanism that provides the ability to pool VMs memory at the scale of a data center. An important property while lending memory to another VM, is the ability to quickly retrieve memory in case of need. Our approach aims at lending memory only for clean cache pages: in case of need, the VM which lent the memory can retrieve it easily. We use the system page cache to store remote pages such that: (i) if local processes allocate memory the borrowed memory can be retrieved immediately, and (ii) if they need cache the remote pages have a lower priority than the local ones.

- Participants: Maxime Lorrillere, Julien Sopena and Pierre Sens
- Partner: LIP6
- Contact: Julien Sopena
- Publications: Conception et évaluation d'un système de cache réparti adapté aux environnements virtualisés Puma: pooling unused memory in virtual machines for I/O intensive applications
- URL: https://github.com/mlorrillere/puma

SPIRALS Project-Team

6. New Software and Platforms

6.1. APISENSE

KEYWORDS: Mobile sensing - Crowd-sensing - Mobile application - Crowd-sourcing - Android FUNCTIONAL DESCRIPTION: APISENSE platform is a software solution to collect various contextual information from Android devices (client application) and automatically upload collected data to a server (deployed as a SaaS). APISENSE is based on a Cloud computing infrastructure to facilitate datasets collection from significant populations of mobile users for research purposes.

- Participants: Antoine Veuiller, Christophe Ribeiro, Julien Duribreux, Nicolas Haderer and Romain Rouvoy
- Partner: Université Lille 1
- Contact: Romain Rouvoy
- URL: http://apisense.io

6.2. Nopol

KEYWORD: Automatic software repair

FUNCTIONAL DESCRIPTION: Nopol is an automatic software repair tool for buggy conditional statements (i.e., if-then-else statements) in Java programs. Nopol takes a buggy program as well as a test suite as input and generates a patch with a conditional expression as output. The test suite is required to contain passing test cases to model the expected behavior of the program and at least one failing test case that reveals the bug to be repaired. The process of Nopol consists of three major phases. First, Nopol employs angelic fix localization to identify expected values of a condition during the test execution. Second, runtime trace collection is used to collect variables and their actual values, including primitive data types and objected-oriented features (e.g., nullness checks), to serve as building blocks for patch generation. Third, Nopol encodes these collected data into an instance of a Satisfiability Modulo Theory (SMT) problem, then a feasible solution to the SMT instance is translated back into a code patch.

- Contact: Martin Monperrus
- URL: https://github.com/SpoonLabs/nopol/

6.3. PowerAPI

KEYWORDS: Energy efficiency - Energy management

FUNCTIONAL DESCRIPTION: PowerAPI is a library for monitoring the energy consumption of software systems.

PowerAPI differs from existing energy process-level monitoring tool in its software orientation, with a fully customizable and modular solution that let the user to precisely define what he/she wants to monitor. PowerAPI is based on a modular and asynchronous event-driven architecture using the Akka library. PowerAPI offers an API which can be used to define requests about energy spent by a process, following its hardware resource utilization (in term of CPU, memory, disk, network, etc.).

- Participants: Adel Noureddine, Loïc Huertas, Maxime Colmant and Romain Rouvoy
- Contact: Romain Rouvoy
- URL: http://powerapi.org

6.4. Saloon

KEYWORDS: Feature Model - Software Product Line - Cloud computing - Model-driven engineering - Ontologies

FUNCTIONAL DESCRIPTION: 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.

- Participants: Clément Quinton, Daniel Romero Acero, Laurence Duchien, Lionel Seinturier and Romain Rouvoy
- Partner: Université Lille 1
- Contact: Lionel Seinturier
- URL: https://gitlab.irisa.fr/drome00A/saloon

6.5. SPOON

KEYWORDS: Java - Code analysis

FUNCTIONAL DESCRIPTION: Spoon is an open-source library that enables you to transform (see below) and analyze Java source code (see example). Spoon provides a complete and fine-grained Java metamodel where any program element (classes, methods, fields, statements, expressions...) can be accessed both for reading and modification. Spoon takes as input source code and produces transformed source code ready to be compiled.

- Participants: Gérard Paligot, Lionel Seinturier, Martin Monperrus and Nicolas Petitprez
- Contact: Martin Monperrus
- URL: http://spoon.gforge.inria.fr

WHISPER Project-Team

6. New Software and Platforms

6.1. Coccinelle

KEYWORDS: Code quality - Evolution - Infrastructure software

FUNCTIONAL DESCRIPTION: Coccinelle is a tool for code search and transformation for C programs. It has been extensively used for bug finding and evolutions in Linux kernel code.

- Participants: Gilles Muller, Julia Lawall, Nicolas Palix, Rene Rydhof Hansen and Thierry Martinez
- Partners: LIP6 IRILL
- Contact: Julia Lawall
- URL: http://coccinelle.lip6.fr

6.2. Prequel

KEYWORDS: Code search - Git

SCIENTIFIC DESCRIPTION: The commit history of a code base such as the Linux kernel is a gold mine of information on how evolutions should be made, how bugs should be fixed, etc. Nevertheless, the high volume of commits available and the rudimentary filtering tools provided mean that it is often necessary to wade through a lot of irrelevant information before finding example commits that can help with a specific software development problem. To address this issue, we propose Prequel (Patch Query Language), which brings the descriptive power of code matching to the problem of querying a commit history.

FUNCTIONAL DESCRIPTION: Prequel is a tool for searching for complex patterns in the commits of software managed using git.

- Participants: Gilles Muller and Julia Lawall
- Partners: LIP6 IRILL
- Contact: Julia Lawall
- URL: http://prequel-pql.gforge.inria.fr/

ALPINES Project-Team

6. New Software and Platforms

6.1. FreeFem++

SCIENTIFIC DESCRIPTION: FreeFem++ is a partial differential equation solver. It has its own language. freefem scripts can solve multiphysics non linear systems in 2D and 3D.

Problems involving PDE (2d, 3d) from several branches of physics such as fluid-structure interactions require interpolations of data on several meshes and their manipulation within one program. FreeFem++ includes a fast 2d^-tree-based interpolation algorithm and a language for the manipulation of data on multiple meshes (as a follow up of bamg (now a part of FreeFem++).

FreeFem++ is written in C++ and the FreeFem++ language is a C++ idiom. It runs on Macs, Windows, Unix machines. FreeFem++ replaces the older freefem and freefem+.

FUNCTIONAL DESCRIPTION: FreeFem++ is a PDE (partial differential equation) solver based on a flexible language that allows a large number of problems to be expressed (elasticity, fluids, etc) with different finite element approximations on different meshes.

- Partner: UPMC
- Contact: Frederic Hecht
- URL: http://www.freefem.org/ff++/

6.2. HPDDM

SCIENTIFIC DESCRIPTION: HPDDM is an efficient implementation of various domain decomposition methods (DDM) such as one- and two-level Restricted Additive Schwarz methods, the Finite Element Tearing and Interconnecting (FETI) method, and the Balancing Domain Decomposition (BDD) method. This code has been proven to be efficient for solving various elliptic problems such as scalar diffusion equations, the system of linear elasticity, but also frequency domain problems like the Helmholtz equation. A comparison with modern multigrid methods can be found in the thesis of Pierre Jolivet.

FUNCTIONAL DESCRIPTION: HPDDM is an efficient implementation of various domain decomposition methods (DDM) such as one- and two-level Restricted Additive Schwarz methods, the Finite Element Tearing and Interconnecting (FETI) method, and the Balancing Domain Decomposition (BDD) method.

- Participants: Frédéric Nataf and Pierre Jolivet
- Contact: Pierre Jolivet
- URL: https://github.com/hpddm

6.3. LORASC

LORASC preconditioner KEYWORD: Preconditioner

- Participants: Laura Grigori and Rémi Lacroix
- Contact: Laura Grigori

6.4. Platforms

6.4.1. HTOOL

KEYWORD: Hierarchical Matrices

FUNCTIONAL DESCRIPTION: HTOOL is a C++ header-only library implementing compression techniques (e.g. Adaptive Cross Approximation) using hierarchical matrices. The library uses MPI and OpenMP for parallelism, and is interfaced with HPDDM for the solution of linear systems.

- Partners: CNRS UPMC ANR NonlocalDD
- Contact: Pierre Marchand
- URL: https://github.com/PierreMarchand20/htool

6.4.2. BemTool

KEYWORD: Boundary Element Method

FUNCTIONAL DESCRIPTION: BemTool is a C++ header-only library implementing the boundary element method for the discretisation of the Laplace, Helmholtz and Maxwell equations, in 2D and 3D. Its main purpose is the assembly of classic boundary element matrices, which can be compressed and inverted through its interface with HTOOL.

- Partners: UPMC ANR NonlocalDD
- Contact: Xavier Claeys
- URL: https://github.com/xclaeys/BemTool

AVALON Project-Team

5. New Software and Platforms

5.1. Kwapi

FUNCTIONAL DESCRIPTION: Kwapi is a software framework dealing with energy monitoring of large scale infrastructures through heterogeneous energy sensors. Kwapi has been designed inside the FSN XLCloud project for Openstack infrastructures. Through the support of Hemera Inria project, kwapi has been extended and deployed in production mode to support easy and large scale energy profiling of the Grid5000 resources.

- Participants: François Rossigneux, Jean-Patrick Gelas, Laurent Lefèvre and Laurent Pouilloux
- Contact: Laurent Lefèvre
- URL: https://launchpad.net/kwapi

5.2. DIET

Distributed Interactive Engineering Toolbox

KEYWORDS: Scheduling - Clusters - Grid - Cloud - HPC - Middleware - Data management. FUNCTIONAL DESCRIPTION: Middleware for grids and clouds. Toolbox for the use and porting of intensive computing applications on heterogeneous architectures.

RELEASE FUNCTIONAL DESCRIPTION: - Upgrade to support Cmake 3.3 and later - Update workflow unit tests to take the results of the execution into account - DIET workflow engine was improved NEWS OF THE YEAR: New release (DIET 2.10) DIET at SC'17 Rutgers University Collaboration

- Participants: Joel Faubert, Hadrien Croubois, Abdelkader Amar, Arnaud Lefray, Aurélien Bouteiller, Benjamin Isnard, Daniel Balouek, Eddy Caron, Eric Bois, Frédéric Desprez, Frédéric Lombart, Gaël Le Mahec, Guillaume Verger, Huaxi Zhang, Jean-Marc Nicod, Jonathan Rouzaud-Cornabas, Lamiel Toch, Maurice Faye, Peter Frauenkron, Philippe Combes, Philippe Laurent, Raphaël Bolze and Yves Caniou
- Partners: CNRS ENS Lyon UCBL Lyon 1 Sysfera
- Contact: Eddy Caron
- URL: http://graal.ens-lyon.fr/diet/

5.3. Sam4C

Security-Aware Models for Clouds

SCIENTIFIC DESCRIPTION: This editor is generated in Java from an EMF -Eclipse Modeling Frameworkmetamodel to simplify any modifications or extensions. The application model and the associated security policy are compiled in a single XML file which serves as input for an external Cloud security-aware scheduler. Alongside with this editor, Cloud architecture models and provisioning algorithms are provided for simulation (in the current version) or real deployments (in future versions).

FUNCTIONAL DESCRIPTION: Sam4C (https://gforge.inria.fr/projects/sam4c/) -Security-Aware Models for Clouds- is a graphical and textual editor to model Cloud applications (as virtual machines, processes, files and communications) and describe its security policy. Sam4C is suitable to represent any static application without deadline or execution time such as n-tiers or parallel applications.

- Participants: Arnaud Lefray, Eddy Caron and Jonathan Rouzaud-Cornabas
- Contact: Eddy Caron
- URL: https://gforge.inria.fr/projects/sam4c/

22

5.4. L2C

Low Level Components

KEYWORDS: Software Components - HPC

FUNCTIONAL DESCRIPTION: L2C (http://hlcm.gforge.inria.fr) is a Low Level Component model implementation targeting at use-cases where overhead matters such as High-Performance Computing. L2C does not offer network transparency neither language transparency. Instead, L2C lets the user choose between various kinds of interactions between components, some with ultra low overhead and others that support network transport. L2C is extensible as additional interaction kinds can be added quite easily. L2C currently supports C++, FORTRAN 2013, MPI and CORBA interactions.

- Participants: Christian Pérez, Hélène Coullon, Jérôme Richard and Vincent Lanore
- Partner: Maison de la simulation
- Contact: Christian Pérez
- URL: http://hlcm.gforge.inria.fr/l2c:start

5.5. Halley

KEYWORDS: Software Components - HPC

SCIENTIFIC DESCRIPTION: Halley is an implementation of the COMET component model that enable to efficiently compose independent parallel code using task graph for multi-core shared-memory machines. NEWS OF THE YEAR: First operational version.

- Participants: Jérôme Richard and Christian Pérez
- Contact: Christian Pérez
- Publications: Conception of a software component model with task scheduling for many-core based parallel architecture, application to the Gysela5D code Combining Both a Component Model and a Task-based Model for HPC Applications: a Feasibility Study on GYSELA COMET: A High-Performance Model for Fine-Grain Composition

5.6. SimGrid

KEYWORDS: Large-scale Emulators - Grid Computing - Distributed Applications

SCIENTIFIC DESCRIPTION: SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. The simulation engine uses algorithmic and implementation techniques toward the fast simulation of large systems on a single machine. The models are theoretically grounded and experimentally validated. The results are reproducible, enabling better scientific practices.

Its models of networks, cpus and disks are adapted to (Data)Grids, P2P, Clouds, Clusters and HPC, allowing multi-domain studies. It can be used either to simulate algorithms and prototypes of applications, or to emulate real MPI applications through the virtualization of their communication, or to formally assess algorithms and applications that can run in the framework.

The formal verification module explores all possible message interleavings in the application, searching for states violating the provided properties. We recently added the ability to assess liveness properties over arbitrary and legacy codes, thanks to a system-level introspection tool that provides a finely detailed view of the running application to the model checker. This can for example be leveraged to verify both safety or liveness properties, on arbitrary MPI code written in C/C++/Fortran.

RELEASE FUNCTIONAL DESCRIPTION:

- Four releases in 2017. Major changes:
 - S4U: many progress, toward SimGrid v4.0. About 80% of the features offered by SimDag and MSG are now integrated, along with examples. Users can now write plugins to extend SimGrid.
 - SMPI: Support MPI 2.2, RMA support, Convert internals to C++.
 - Java: Massive memleaks and performance issues fixed.
 - New models: Multi-core VMs, Energy consumption due to the network
 - All internals are now converted to C++, and most of our internally developped data containers were replaced with std::* constructs.
 - (+ bug fixes, cleanups and documentation improvements)
- Participants: Adrien Lèbre, Arnaud Legrand, Augustin Degomme, Florence Perronnin, Frédéric Suter, Jean-Marc Vincent, Jonathan Pastor, Jonathan Rouzaud-Cornabas, Luka Stanisic, Mario Südholt and Martin Quinson
- Partners: CNRS ENS Rennes
- Contact: Martin Quinson
- URL: http://simgrid.gforge.inria.fr/

5.7. execo

KEYWORDS: Toolbox - Deployment - Orchestration - Python

FUNCTIONAL DESCRIPTION: Execo offers a Python API for asynchronous control of local or remote, standalone or parallel, unix processes. It is especially well suited for quickly and easily scripting workflows of parallel/distributed operations on local or remote hosts: automate a scientific workflow, conduct computer science experiments, perform automated tests, etc. The core python package is execo. The execo_g5k package provides a set of tools and extensions for the Grid5000 testbed. The execo_engine package provides tools to ease the development of computer sciences experiments.

- Participants: Florent Chuffart, Laurent Pouilloux and Matthieu Imbert
- Contact: Matthieu Imbert
 URL: http://execo.gforge.inria.fr
- 5.8. Grid'5000

Participants: Laurent Lefèvre, Simon Delamare, David Loup, Christian Perez.

FUNCTIONAL DESCRIPTION

The Grid'5000 experimental platform is a scientific instrument to support computer science research related to distributed systems, including parallel processing, high performance computing, cloud computing, operating systems, peer-to-peer systems and networks. It is distributed on 10 sites in France and Luxembourg, including Lyon. Grid'5000 is a unique platform as it offers to researchers many and varied hardware resources and a complete software stack to conduct complex experiments, ensure reproducibility and ease understanding of results. In 2016, a new cluster financially supported by Inria has been deployed on the Grid'5000 Lyon site.

- Contact: Laurent Lefèvre
- URL: https://www.grid5000.fr/

5.9. Leco

Participants: Thierry Gautier, Laurent Lefèvre, Christian Perez.

FUNCTIONAL DESCRIPTION

The LECO experimental platform is a new medium size scientific instrument funded by DRRT to investigate research related to BigData and HPC. It is located in Grenoble as part of the the HPCDA computer managed by UMS GRICAD. The installation starts in December 2017.

• Contact: Thierry Gautier

DATAMOVE Project-Team

6. New Software and Platforms

6.1. FlowVR

SCIENTIFIC DESCRIPTION: FlowVR adopts the "data-flow" paradigm, where your application is divided as a set of components exchanging messages (think of it as a directed graph). FlowVR enables to encapsulate existing codes in components, interconnect them through data channels, and deploy them on distributed computing resources. FlowVR takes care of all the heavy lifting such as application deployment and message exchange.

The base entity, called a module or component, is an autonomous process, potentially multi-threaded with tools like OpenMP, TBB, or deferring computations to a GPU or Xeon Phi. This module processes data coming from input ports and write data on output ports. A module has no global insight on where the data comes from or goes to. The programming interface is designed to limit code refactoring, easing turning an existing code into a FlowVR component. The three main functions are:

wait(): Blocking function call that waits for the availability of new messages on input ports. get(): Retrieve a handle to access the message received at the previous wait() call on a given input port. put(): Notify FlowVR that a new message on a given output port is ready for dispatch. FlowVR manages data transfers. Intra-node communications between two components take place through a shared memory segment, avoiding copies. Once the sender has prepared the data in a shared memory segment, it simply handles a pointer to the destination that can directly access them. Inter-node communications extend this mechanism, FlowVR taking care of packing and transferring the data from the source shared memory segment to the destination shared memory segment.

Assembling components to build an application consists in writing a Python script, instanciate it according to the target machine. FlowVR will process it and prepare everything so that in one command line you can deploy and start your application.

FUNCTIONAL DESCRIPTION: FlowVR adopts the "data-flow" paradigm, where your application is divided as a set of components exchanging messages (think of it as a directed graph). FlowVR enables to encapsulate existing codes in components, interconnect them through data channels, and deploy them on distributed computing resources. FlowVR takes care of all the heavy lifting such as application deployment and message exchange.

- Participants: Bruno Raffin, Clément Ménier, Emmanuel Melin, Jean Denis Lesage, Jérémie Allard, Jérémy Jaussaud, Matthieu Dreher, Sébastien Limet, Sophie Robert and Valérie Gourantou
- Contact: Bruno Raffin
- URL: http://flowvr.sf.net

6.2. OAR

KEYWORDS: Resource manager - Clusters - Cloud - HPC - Light grid

SCIENTIFIC DESCRIPTION: This batch system is based on a database (PostgreSQL (preferred) or MySQL), a script language (Perl) and an optional scalable administrative tool (e.g. Taktuk). It is composed of modules which interact mainly via the database and are executed as independent programs. Therefore, formally, there is no API, the system interaction is completely defined by the database schema. This approach eases the development of specific modules. Indeed, each module (such as schedulers) may be developed in any language having a database access library.

FUNCTIONAL DESCRIPTION: OAR is a versatile resource and task manager (also called a batch scheduler) for HPC clusters, and other computing infrastructures (like distributed computing experimental testbeds where versatility is a key).

- Participants: Bruno Bzeznik, Olivier Richard and Pierre Neyron
- Partners: LIG CNRS Grid'5000 CIMENT
- Contact: Olivier Richard
- URL: http://oar.imag.fr

6.3. MELISSA

Modular External Library for In Situ Statistical Analysis

KEYWORD: Sensitivity Analysis

FUNCTIONAL DESCRIPTION: Melissa is an in situ solution for sensitivity analysis. It implements iterative algorithms to compute spatio-temporal statistic fields over results of large scale sensitivity studies. Melissa relies on a client/server architecture, composed of three main modules:

Melissa Server: an independent parallel executable. It receives data from the simulations, updates iterative statistics as soon as possible, then trow data away. Melissa API: a shared library to be linked within the simulation code. It mainly transmit simulation data to Melissa Server at each timestep. The simulations of the sensitivity analysis become the clients of Melissa Server. Melissa Launcher: A Python script in charge of generating and managing the whole global sensitivity analysis.

- Authors: Théophile Terraz, Bruno Raffin, Alejandro Ribes and Bertrand Iooss
- Partner: Edf
- Contact: Bruno Raffin
- Publications: In Situ Statistical Analysis for Parametric Studies Melissa: Large Scale In Transit Sensitivity Analysis Avoiding Intermediate Files
- URL: https://melissa-sa.github.io

6.4. Platforms

6.4.1. Grid'5000 (https://www.grid5000.fr/) and Meso Center Ciment (https://ciment.ujf-grenoble.fr)

We have been very active in promoting the factorization of compute resources at a regional and national level. We have a three level implication, locally to maintain a pool of very flexible experimental machines (hundreds of cores), regionally through the CIMENT meso center (Equipex Grant), and nationally by contributing to the Grid'5000 platform, our local resources being included in this platform. Olivier Richard is member of Grid'5000 scientific committee and Pierre Neyron is member of the technical committee. The OAR scheduler in particular is deployed on both infrastructures. We are currently preparing proposals for the next generation machines within the context of the new university association (Univ. Grenoble-Alpes).

HIEPACS Project-Team

6. New Software and Platforms

6.1. Chameleon

KEYWORDS: Runtime system - Task-based algorithm - Dense linear algebra - HPC - Task scheduling SCIENTIFIC DESCRIPTION: Chameleon is part of the MORSE (Matrices Over Runtime Systems @ Exascale) project. The overall objective is to develop robust linear algebra libraries relying on innovative runtime systems that can fully benefit from the potential of those future large-scale complex machines.

We expect advances in three directions based first on strong and closed interactions between the runtime and numerical linear algebra communities. This initial activity will then naturally expand to more focused but still joint research in both fields.

1. Fine interaction between linear algebra and runtime systems. On parallel machines, HPC applications need to take care of data movement and consistency, which can be either explicitly managed at the level of the application itself or delegated to a runtime system. We adopt the latter approach in order to better keep up with hardware trends whose complexity is growing exponentially. One major task in this project is to define a proper interface between HPC applications and runtime systems in order to maximize productivity and expressivity. As mentioned in the next section, a widely used approach consists in abstracting the application as a DAG that the runtime system is in charge of scheduling. Scheduling such a DAG over a set of heterogeneous processing units introduces a lot of new challenges, such as predicting accurately the execution time of each type of task over each kind of unit, minimizing data transfers between MPI will be designed to allow applications to provide scheduling hints to the runtime system and to get real-time feedback about the consequences of scheduling decisions.

2. Runtime systems. A runtime environment is an intermediate layer between the system and the application. It provides low-level functionality not provided by the system (such as scheduling or management of the heterogeneity) and high-level features (such as performance portability). In the framework of this proposal, we will work on the scalability of runtime environment. To achieve scalability it is required to avoid all centralization. Here, the main problem is the scheduling of the tasks. In many task-based runtime environments the scheduler is centralized and becomes a bottleneck as soon as too many cores are involved. It is therefore required to distribute the scheduling decision or to compute a data distribution that impose the mapping of task using, for instance the so-called "owner-compute" rule. Expected advances: We will design runtime systems that enable an efficient and scalable use of thousands of distributed multicore nodes enhanced with accelerators.

3. Linear algebra. Because of its central position in HPC and of the well understood structure of its algorithms, dense linear algebra has often pioneered new challenges that HPC had to face. Again, dense linear algebra has been in the vanguard of the new era of petascale computing with the design of new algorithms that can efficiently run on a multicore node with GPU accelerators. These algorithms are called "communication-avoiding" since they have been redesigned to limit the amount of communication between processing units (and between the different levels of memory hierarchy). They are expressed through Direct Acyclic Graphs (DAG) of fine-grained tasks that are dynamically scheduled. Expected advances: First, we plan to investigate the impact of these principles in the case of sparse applications (whose algorithms are slightly more complicated but often rely on dense kernels). Furthermore, both in the dense and sparse cases, the scalability on thousands of nodes is still limited, new numerical approaches need to be found. We will specifically design sparse hybrid direct/iterative methods that represent a promising approach.

Overall end point. The overall goal of the MORSE associate team is to enable advanced numerical algorithms to be executed on a scalable unified runtime system for exploiting the full potential of future exascale machines.

27

FUNCTIONAL DESCRIPTION: Chameleon is a dense linear algebra software relying on sequential task-based algorithms where sub-tasks of the overall algorithms are submitted to a Runtime system. A Runtime system such as StarPU is able to manage automatically data transfers between not shared memory area (CPUs-GPUs, distributed nodes). This kind of implementation paradigm allows to design high performing linear algebra algorithms on very different type of architecture: laptop, many-core nodes, CPUs-GPUs, multiple nodes. For example, Chameleon is able to perform a Cholesky factorization (double-precision) at 80 TFlop/s on a dense matrix of order 400 000 (e.i. 4 min).

RELEASE FUNCTIONAL DESCRIPTION: Chameleon includes the following features:

- BLAS 3, LAPACK one-sided and LAPACK norms tile algorithms - Support QUARK and StarPU runtime systems - Exploitation of homogeneous and heterogeneous platforms through the use of BLAS/LAPACK CPU kernels and cuBLAS/MAGMA CUDA kernels - Exploitation of clusters of interconnected nodes with distributed memory (using OpenMPI)

- Participants: Cédric Castagnede, Samuel Thibault, Emmanuel Agullo, Florent Pruvost and Mathieu Faverge
- Partners: Innovative Computing Laboratory (ICL) King Abdullha University of Science and Technology University of Colorado Denver
- Contact: Emmanuel Agullo
- URL: https://project.inria.fr/chameleon/

6.2. Fabulous

Fast Accurate Block Linear krylOv Solver

KEYWORDS: Numerical algorithm - Block Krylov solver

SCIENTIFIC DESCRIPTION: Versatile and flexible numerical library that implements Block Krylov iterative schemes for the solution of linear systems of equations with multiple right-hand sides

FUNCTIONAL DESCRIPTION: Versatile and flexible numerical library that implements Block Krylov iterative schemes for the solution of linear systems of equations with multiple right-hand sides. The library implements block variants of minimal norm residual variants with partial convergence management and spectral information recycling. The package already implements regular block-GMRES (BGMRES), Inexact Breakdown BGMRES (IB-BMGRES), Inexact Breakdown BGMRES with Deflated Restarting (IB-BGMRES-DR), Block Generalized Conjugate Residual with partial convergence management. The C++ library relies on callback mechanisms to implement the calculations (matrix-vector, dot-product, ...) that depend on the parallel data distribution selected by the user.

- Participants: Emmanuel Agullo, Luc Giraud and Cyrille Piacibello
- Contact: Luc Giraud
- Publication: Block GMRES method with inexact breakdowns and deflated restarting
- URL: https://gitlab.inria.fr/solverstack/fabulous/

6.3. HIPS

Hierarchical Iterative Parallel Solver KEYWORDS: Simulation - HPC - Parallel calculation - Hybrid direct iterative method SCIENTIFIC DESCRIPTION: The key point of the methods implemented in HIPS is to define an ordering and a partition of the unknowns that relies on a form of nested dissection ordering in which cross points in the separators play a special role (Hierarchical Interface Decomposition ordering). The subgraphs obtained by nested dissection correspond to the unknowns that are eliminated using a direct method and the Schur complement system on the remaining of the unknowns (that correspond to the interface between the subgraphs viewed as sub-domains) is solved using an iterative method (GMRES or Conjugate Gradient at the time being). This special ordering and partitioning allows for the use of dense block algorithms both in the direct and iterative part of the solver and provides a high degree of parallelism to these algorithms. The code provides a hybrid method which blends direct and iterative solvers. HIPS exploits the partitioning and multistage ILUU techniques to enable a highly parallel scheme where several subdomains can be assigned to the same process. It also provides a scalar preconditioner based on the multistage ILUT factorization.

HIPS can be used as a standalone program that reads a sparse linear system from a file, it also provides an interface to be called from any C, C++ or Fortran code. It handles symmetric, unsymmetric, real or complex matrices. Thus, HIPS is a software library that provides several methods to build an efficient preconditioner in almost all situations.

FUNCTIONAL DESCRIPTION: HIPS (Hierarchical Iterative Parallel Solver) is a scientific library that provides an efficient parallel iterative solver for very large sparse linear systems.

- Participants: Jérémie Gaidamour, Pascal Hénon and Yousef Saad
- Contact: Pierre Ramet
- URL: http://hips.gforge.inria.fr/

6.4. MAPHYS

Massively Parallel Hybrid Solver

KEYWORD: Parallel hybrid direct/iterative solution of large linear systems

FUNCTIONAL DESCRIPTION: MaPHyS is a software package that implements a parallel linear solver coupling direct and iterative approaches. The underlying idea is to apply to general unstructured linear systems domain decomposition ideas developed for the solution of linear systems arising from PDEs. The interface problem, associated with the so called Schur complement system, is solved using a block preconditioner with overlap between the blocks that is referred to as Algebraic Additive Schwarz. A fully algebraic coarse space is available for symmetric positive definite problems, that insures the numerical scalability of the preconditioner.

The parallel implementation is based on MPI+thread. Maphys relies on state-of-the art sparse and dense direct solvers.

MaPHyS is essentially a preconditioner that can be used to speed-up the convergence of any Krylov subspace method and is coupled with the ones implemented in the Fabulous package.

- Participants: Emmanuel Agullo, Luc Giraud, Matthieu Kuhn, Gilles Marait and Louis Poirel
- Contact: Emmanuel Agullo
- Publications: Hierarchical hybrid sparse linear solver for multicore platforms Robust coarse spaces for Abstract Schwarz preconditioners via generalized eigenproblems
- URL: https://gitlab.inria.fr/solverstack/maphys

6.5. MetaPart

KEYWORDS: High performance computing - HPC - Parallel computing - Graph algorithmics - Graph - Hypergraph

FUNCTIONAL DESCRIPTION: MetaPart is a framework for graph or hypergraph manipulation that addresses different problems, like partitioning, repartitioning, or co-partitioning, ... MetaPart is made up of several projects, such as StarPart, LibGraph or CoPart. StarPart is the core of the MetaPart framework. It offers a wide variety of graph partitioning methods (Metis, Scotch, Zoltan, Patoh, ParMetis, Kahip, ...), which makes it easy to compare these different methods and to better adjust the parameters of these methods. It is built upon the LibGraph library, that provides basic graph & hypergraph routines. The Copart project is a library used on top of StarPart, that provides co-partitioning algorithms for the load-blancing of parallel coupled simulations.

- Participant: Aurélien Esnard
- Contact: Aurélien Esnard
- URL: https://gitlab.inria.fr/metapart

6.6. MPICPL

MPI CouPLing

KEYWORDS: MPI - Coupling software

FUNCTIONAL DESCRIPTION: MPICPL is a software library dedicated to the coupling of parallel legacy codes, that are based on the well-known MPI standard. It proposes a lightweight and comprehensive programing interface that simplifies the coupling of several MPI codes (2, 3 or more). MPICPL facilitates the deployment of these codes thanks to the mpicplrun tool and it interconnects them automatically through standard MPI intercommunicators. Moreover, it generates the universe communicator, that merges the world communicators of all coupled-codes. The coupling infrastructure is described by a simple XML file, that is just loaded by the mpicplrun tool.

- Participant: Aurélien Esnard
- Contact: Aurélien Esnard
- URL: https://gitlab.inria.fr/esnard/mpicpl

6.7. OptiDis

KEYWORDS: Dislocation dynamics simulation - Fast multipole method - Large scale - Collision

FUNCTIONAL DESCRIPTION: OptiDis is a new code for large scale dislocation dynamics simulations. Its purpose is to simulate real life dislocation densities (up to 5.1022 dislocations/m-2) in order to understand plastic deformation and study strain hardening. The main application is to observe and understand plastic deformation of irradiated zirconium. Zirconium alloys are the first containment barrier against the dissemination of radioactive elements. More precisely, with neutron irradiated zirconium alloys we are talking about channeling mechanism, which means to stick with the reality, more than tens of thousands of induced loops, i. e. 100 million degrees of freedom in the simulation. The code is based on Numodis code developed at CEA Saclay and the ScalFMM library developed in H/14iePACS project. The code is written in C++ language and using the last features of C++11. One of the main aspects is the hybrid parallelism MPI/OpenMP that gives the software the ability to scale on large cluster while the computation load rises. In order to achieve that, we use a thinner level for threads, dividing the domain by an Octree representation. All theses parts are controlled by the ScalFMM library. On the last level, our data are stored in an adaptive structure that absorbs the dynamics of this type of simulation and manages the parallelism of tasks..

- Participant: Olivier Coulaud
- Contact: Olivier Coulaud
- URL: http://optidis.gforge.inria.fr/

6.8. PaStiX

Parallel Sparse matriX package

KEYWORDS: Sparse Matrices - Factorisation - High-performance calculation - Linear algebra - Linear Systems Solver

SCIENTIFIC DESCRIPTION: PaStiX is based on an efficient static scheduling and memory manager, in order to solve 3D problems with more than 50 million of unknowns. The mapping and scheduling algorithm handle a combination of 1D and 2D block distributions. A dynamic scheduling can also be applied to take care of NUMA architectures while taking into account very precisely the computational costs of the BLAS 3 primitives, the communication costs and the cost of local aggregations.

FUNCTIONAL DESCRIPTION: PaStiX is a scientific library that provides a high performance parallel solver for very large sparse linear systems based on block direct and block ILU(k) methods. It can handle lowrank compression techniques to reduce the computation and the memory complexity. Numerical algorithms are implemented in single or double precision (real or complex) for LLt, LDLt and LU factorization with static pivoting (for non symmetric matrices having a symmetric pattern). The PaStiX library uses the graph partitioning and sparse matrix block ordering packages Scotch or Metis.

The PaStiX solver is suitable for any heterogeneous parallel/distributed architecture when its performance is predictable, such as clusters of multicore nodes with GPU accelerators or KNL processors. In particular, we provide a high-performance version with a low memory overhead for multicore node architectures, which fully exploits the advantage of shared memory by using an hybrid MPI-thread implementation.

- Participants: Grégoire Pichon, Mathieu Faverge and Pierre Ramet
- Partner: Université Bordeaux 1
- Contact: Pierre Ramet
- URL: http://pastix.gforge.inria.fr/

6.9. ScalFMM

Scalable Fast Multipole Method

KEYWORDS: N-body - Fast multipole method - Parallelism - MPI - OpenMP

SCIENTIFIC DESCRIPTION: ScalFMM is a software library to simulate N-body interactions using the Fast Multipole Method. The library offers two methods to compute interactions between bodies when the potential decays like 1/r. The first method is the classical FMM based on spherical harmonic expansions and the second is the Black-Box method which is an independent kernel formulation (introduced by E. Darve @ Stanford). With this method, we can now easily add new non oscillatory kernels in our library. For the classical method, two approaches are used to decrease the complexity of the operators. We consider either matrix formulation that allows us to use BLAS routines or rotation matrix to speed up the M2L operator.

ScalFMM intends to offer all the functionalities needed to perform large parallel simulations while enabling an easy customization of the simulation components: kernels, particles and cells. It works in parallel in a shared/distributed memory model using OpenMP and MPI. The software architecture has been designed with two major objectives: being easy to maintain and easy to understand. There is two main parts:

the management of the octree and the parallelization of the method the kernels. This new architecture allow us to easily add new FMM algorithm or kernels and new paradigm of parallelization.

FUNCTIONAL DESCRIPTION: Compute N-body interactions using the Fast Multipole Method for large number of objects

- Participants: Bramas Bérenger and Olivier Coulaud
- Contact: Olivier Coulaud
- URL: https://gitlab.inria.fr/solverstack/ScalFMM

6.10. VITE

Visual Trace Explorer KEYWORDS: Visualization - Execution trace FUNCTIONAL DESCRIPTION: VITE is a trace explorer. It is a tool made to visualize execution traces of large parallel programs. It supports Pajé, a trace format created by Inria Grenoble, and OTF and OTF2 formats, developed by the University of Dresden and allows the programmer a simpler way to analyse, debug and/or profile large parallel applications.

- Participant: Mathieu Faverge
- Contact: Mathieu Faverge
- URL: http://vite.gforge.inria.fr/

6.11. PlaFRIM

Plateforme Fédérative pour la Recherche en Informatique et Mathématiques

KEYWORDS: High-Performance Computing - Hardware platform

FUNCTIONAL DESCRIPTION: PlaFRIM is an experimental platform for research in modeling, simulations and high performance computing. This platform has been set up from 2009 under the leadership of Inria Bordeaux Sud-Ouest in collaboration with computer science and mathematics laboratories, respectively Labri and IMB with a strong support in the region Aquitaine.

It aggregates different kinds of computational resources for research and development purposes. The latest technologies in terms of processors, memories and architecture are added when they are available on the market. It is now more than 1,000 cores (excluding GPU and Xeon Phi) that are available for all research teams of Inria Bordeaux, Labri and IMB. This computer is in particular used by all the engineers who work in HiePACS and are advised by F. Rue from the SED.

- Contact: Olivier Coulaud
- URL: https://www.plafrim.fr/en/home/

KERDATA Project-Team

5. New Software and Platforms

5.1. BlobSeer

BlobSeer : A Storage System For The Exascale Era

KEYWORDS: Versioning - HPC - Cloud storage - Distributed metadata - MapReduce

SCIENTIFIC DESCRIPTION: BlobSeer is a large-scale distributed storage service that addresses advanced data management requirements resulting from ever-increasing data sizes. It is centered around the idea of leveraging versioning for concurrent manipulation of binary large objects in order to efficiently exploit data-level parallelism and sustain a high throughput despite massively parallel data access.

FUNCTIONAL DESCRIPTION: BlobSeer is a large-scale distributed storage service for advanced management of massive data. Validated on Nimbus, OpenNebula and Microsoft Azure cloud platforms.

- Participants: Bogdan Nicolae, Gabriel Antoniu and Luc Bougé
- Partners: Université de Rennes 1 ENS Cachan
- Contact: Gabriel Antoniu
- URL: http://blobseer.gforge.inria.fr/

5.2. Damaris

KEYWORDS: Big data - Visualization - I/O - HPC - Exascale

SCIENTIFIC DESCRIPTION: Damaris is a middleware for multicore SMP nodes enabling them to efficiently handle data transfers for storage and visualization. The key idea is to dedicate one or a few cores of each SMP node to the application I/O. It is developed within the framework of a collaboration between KerData and the Joint Laboratory for Petascale Computing (JLPC). The current version enables efficient asynchronous I/O, hiding all I/O related overheads such as data compression and post-processing, as well as direct (in situ) interactive visualization of the generated data.

Damaris has been preliminarily evaluated at NCSA (Urbana-Champaign) with the CM1 tornado simulation code. CM1 is one of the target applications of the Blue Waters supercomputer in production at NCSA/UIUC (USA), in the framework of the Inria-UIUC-ANL Joint Lab (JLPC). Damaris now has external users, including (to our knowledge) visualization specialists from NCSA and researchers from the France/Brazil Associated research team on Parallel Computing (joint team between Inria/LIG Grenoble and the UFRGS in Brazil). Damaris has been successfully integrated into three large-scale simulations (CM1, OLAM, Nek5000). Works are in progress to evaluate it in the context of several other simulations including HACC (cosmology code) and GTC (fusion).

FUNCTIONAL DESCRIPTION: Damaris is a middleware for data management targeting large-scale HPC simulations: • «In-situ» data analysis by some dedicated cores of the simulation platform • Asynchronous and fast data transfer from HPC simulations to Damaris • Semantic-aware dataset processing through Damaris plug-ins

- Participants: Gabriel Antoniu, Lokman Rahmani, Luc Bougé, Matthieu Dorier and Orçun Yildiz
- Partner: ENS Rennes
- Contact: Matthieu Dorier
- URL: https://project.inria.fr/damaris/

5.3. iHadoop

FUNCTIONAL DESCRIPTION: iHadoop is a Hadoop simulator developed in Java on top of SimGrid to simulate the behavior of Hadoop and therefore accurately predict the performance of Hadoop in normal scenarios and under failures.

iHadoop is an internal software prototype, which was initially developed to validate our idea for exploring the behavior of Hadoop under failures. iHadoop has preliminarily evaluated within our group and it has shown very high accuracy when predicating the execution time of a Map-Reduce application. We intend to integrate iHadoop within the SimGrid distribution and make it available to the SimGrid community.

- Participants: Shadi Ibrahim and Tien Dat Phan
- Contact: Shadi Ibrahim

5.4. JetStream

FUNCTIONAL DESCRIPTION: JetStream is a middleware solution for batch-based, high-performance streaming across cloud data centers. JetStream implements a set of context-aware strategies for optimizing batchbased streaming, being able to self-adapt to changing conditions. Additionally, the system provides multiroute streaming across cloud data centers for aggregating bandwidth by leveraging the network parallelism. It enables easy deployment across .Net frameworks and seamless binding with event processing engines such as StreamInsight.

JetStream is currently used at Microsoft Research ATLE Munich for the management of the Azure cloud infrastructure.

- Participants: Alexandru Costan, Gabriel Antoniu and Radu Marius Tudoran
- Contact: Alexandru Costan

5.5. OverFlow

FUNCTIONAL DESCRIPTION: OverFlow is a uniform data management system for scientific workflows running across geographically distributed sites, aiming to reap economic benefits from this geo-diversity. The software is environment-aware, as it monitors and models the global cloud infrastructure, offering high and predictable data handling performance for transfer cost and time, within and across sites. OverFlow proposes a set of pluggable services, grouped in a data-scientist cloud kit. They provide the applications with the possibility to monitor the underlying infrastructure, to exploit smart data compression, deduplication and georeplication, to evaluate data management costs, to set a tradeoff between money and time, and optimize the transfer strategy accordingly.

Currently, OverFlow is used for data transfers by the Microsoft Research ATLE Munich team as well as for synthetic benchmarks at the Politehnica University of Bucharest.

- Participants: Alexandru Costan, Gabriel Antoniu and Radu Marius Tudoran
- Contact: Alexandru Costan

POLARIS Team

6. New Software and Platforms

6.1. Framesoc

FUNCTIONAL DESCRIPTION: Framesoc is the core software infrastructure of the SoC-Trace project. It provides a graphical user environment for execution-trace analysis, featuring interactive analysis views as Gantt charts or statistics views. It provides also a software library to store generic trace data, play with them, and build other analysis tools (e.g., Ocelotl).

- Participants: Arnaud Legrand and Jean-Marc Vincent
- Contact: Guillaume Huard
- URL: http://soctrace-inria.github.io/framesoc/

6.2. GameSeer

FUNCTIONAL DESCRIPTION: GameSeer is a tool for students and researchers in game theory that uses Mathematica to generate phase portraits for normal form games under a variety of (user-customizable) evolutionary dynamics. The whole point behind GameSeer is to provide a dynamic graphical interface that allows the user to employ Mathematica's vast numerical capabilities from a simple and intuitive front-end. So, even if you've never used Mathematica before, you should be able to generate fully editable and customizable portraits quickly and painlessly.

- Contact: Panayotis Mertikopoulos
- URL: http://mescal.imag.fr/membres/panayotis.mertikopoulos/publications.html

6.3. marmoteCore

Markov Modeling Tools and Environments - the Core

KEYWORDS: Modeling - Stochastic models - Markov model

FUNCTIONAL DESCRIPTION: marmoteCore is a C++ environment for modeling with Markov chains. It consists in a reduced set of high-level abstractions for constructing state spaces, transition structures and Markov chains (discrete-time and continuous-time). It provides the ability of constructing hierarchies of Markov models, from the most general to the particular, and equip each level with specifically optimized solution methods.

This software is developed within the ANR MARMOTE project: ANR-12-MONU-00019.

- Participants: Alain Jean-Marie, Hlib Mykhailenko, Benjamin Briot, Franck Quessette, Issam Rabhi, Jean-Marc Vincent and Jean-Michel Fourneau
- Partner: UVSQ
- Contact: Alain Jean-Marie
- Publications: marmoteCore: a Markov Modeling Platform marmoteCore: a software platform for Markov modeling
- URL: http://marmotecore.gforge.inria.fr/

6.4. Moca

Memory Organisation Cartography and Analysis

KEYWORDS: High-Performance Computing - Performance analysis

- Contact: David Beniamine
- URL: https://github.com/dbeniamine/MOCA

6.5. Ocelotl

Multidimensional Overviews for Huge Trace Analysis

FUNCTIONAL DESCRIPTION: Ocelotl is an innovative visualization tool, which provides overviews for execution trace analysis by using a data aggregation technique. This technique enables to find anomalies in huge traces containing up to several billions of events, while keeping a fast computation time and providing a simple representation that does not overload the user.

- Participants: Arnaud Legrand and Jean-Marc Vincent
- Contact: Jean-Marc Vincent
- URL: http://soctrace-inria.github.io/ocelotl/

6.6. PSI

Perfect Simulator

FUNCTIONAL DESCRIPTION: Perfect simulator is a simulation software of markovian models. It is able to simulate discrete and continuous time models to provide a perfect sampling of the stationary distribution or directly a sampling of functional of this distribution by using coupling from the past. The simulation kernel is based on the CFTP algorithm, and the internal simulation of transitions on the Aliasing method.

- Contact: Jean-Marc Vincent
- URL: http://psi.gforge.inria.fr/

6.7. SimGrid

KEYWORDS: Large-scale Emulators - Grid Computing - Distributed Applications

SCIENTIFIC DESCRIPTION: SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. The simulation engine uses algorithmic and implementation techniques toward the fast simulation of large systems on a single machine. The models are theoretically grounded and experimentally validated. The results are reproducible, enabling better scientific practices.

Its models of networks, cpus and disks are adapted to (Data)Grids, P2P, Clouds, Clusters and HPC, allowing multi-domain studies. It can be used either to simulate algorithms and prototypes of applications, or to emulate real MPI applications through the virtualization of their communication, or to formally assess algorithms and applications that can run in the framework.

The formal verification module explores all possible message interleavings in the application, searching for states violating the provided properties. We recently added the ability to assess liveness properties over arbitrary and legacy codes, thanks to a system-level introspection tool that provides a finely detailed view of the running application to the model checker. This can for example be leveraged to verify both safety or liveness properties, on arbitrary MPI code written in C/C++/Fortran.
RELEASE FUNCTIONAL DESCRIPTION:

- Four releases in 2017. Major changes:
 - S4U: many progress, toward SimGrid v4.0. About 80% of the features offered by SimDag and MSG are now integrated, along with examples. Users can now write plugins to extend SimGrid.
 - SMPI: Support MPI 2.2, RMA support, Convert internals to C++.
 - Java: Massive memleaks and performance issues fixed.
 - New models: Multi-core VMs, Energy consumption due to the network
 - All internals are now converted to C++, and most of our internally developped data containers were replaced with std::* constructs.
 - (+ bug fixes, cleanups and documentation improvements)
- Participants: Adrien Lèbre, Arnaud Legrand, Augustin Degomme, Florence Perronnin, Frédéric Suter, Jean-Marc Vincent, Jonathan Pastor, Jonathan Rouzaud-Cornabas, Luka Stanisic, Mario Südholt and Martin Quinson
- Partners: CNRS ENS Rennes
- Contact: Martin Quinson
- URL: http://simgrid.gforge.inria.fr/

6.8. Tabarnac

Tool for Analyzing the Behavior of Applications Running on NUMA ArChitecture KEYWORDS: High-Performance Computing - Performance analysis - NUMA

- Contact: David Beniamine
- URL: https://dbeniamine.github.io/Tabarnac/

ROMA Project-Team

6. New Software and Platforms

6.1. DCC

DPN C Compiler

KEYWORDS: Polyhedral compilation - Automatic parallelization - High-level synthesis

FUNCTIONAL DESCRIPTION: Dcc (Data-aware process network C compiler) analyzes a sequential regular program written in C and generates an equivalent architecture of parallel computer as a communicating process network (Data-aware Process Network, DPN). Internal communications (channels) and external communications (external memory) are automatically handled while fitting optimally the characteristics of the global memory (latency and throughput). The parallelism can be tuned. Dcc has been registered at the APP ("Agence de protection des programmes") and transferred to the XtremLogic start-up under an Inria license.

- Participants: Alexandru Plesco and Christophe Alias
- Contact: Christophe Alias

6.2. MUMPS

A MUltifrontal Massively Parallel Solver

KEYWORDS: High-Performance Computing - Direct solvers - Finite element modelling

FUNCTIONAL DESCRIPTION: MUMPS is a software library to solve large sparse linear systems (AX=B) on sequential and parallel distributed memory computers. It implements a sparse direct method called the multifrontal method. It is used worldwide in academic and industrial codes, in the context numerical modeling of physical phenomena with finite elements. Its main characteristics are its numerical stability, its large number of features, its high performance and its constant evolution through research and feedback from its community of users. Examples of application fields include structural mechanics, electromagnetism, geophysics, acoustics, computational fluid dynamics. MUMPS is developed by INPT(ENSEEIHT)-IRIT, Inria, CERFACS, University of Bordeaux, CNRS and ENS Lyon. In 2014, a consortium of industrial users has been created (http://mumps-consortium.org).

RELEASE FUNCTIONAL DESCRIPTION: MUMPS versions 5.1.0, 5.1.1 and 5.1.2, all released in 2017 include many new features and improvements. The two main new features are Block Low-Rank compression, decreasing the complexity of sparse direct solvers for various types of applications, and selective 64-bit integers, allowing to process matrices with more than 2 billion entries.

- Participants: Gilles Moreau, Abdou Guermouche, Alfredo Buttari, Aurélia Fevre, Bora Uçar, Chiara Puglisi, Clément Weisbecker, Emmanuel Agullo, François-Henry Rouet, Guillaume Joslin, Jacko Koster, Jean-Yves L'excellent, Marie Durand, Maurice Bremond, Mohamed Sid-Lakhdar, Patrick Amestoy, Philippe Combes, Stéphane Pralet, Theo Mary and Tzvetomila Slavova
- Partners: Université de Bordeaux CNRS CERFACS ENS Lyon INPT IRIT Université de Lyon Université de Toulouse LIP
- Contact: Jean-Yves L'excellent
- URL: http://mumps-solver.org/

6.3. PoCo

Polyhedral Compilation Library KEYWORDS: Polyhedral compilation - Automatic parallelization FUNCTIONAL DESCRIPTION: PoCo (Polyhedral Compilation Library) is a compilation framework allowing to develop parallelizing compilers for regular programs. PoCo features many state-of-the-art polyhedral program analysis and a symbolic calculator on execution traces (represented as convex polyhedra). PoCo has been registered at the APP ("agence de protection des programmes") and transferred to the XtremLogic start-up under an Inria licence.

- Participant: Christophe Alias
- Contact: Christophe Alias

STORM Project-Team

5. New Software and Platforms

5.1. Chameleon

KEYWORDS: Runtime system - Task-based algorithm - Dense linear algebra - HPC - Task scheduling SCIENTIFIC DESCRIPTION: Chameleon is part of the MORSE (Matrices Over Runtime Systems @ Exascale) project. The overall objective is to develop robust linear algebra libraries relying on innovative runtime systems that can fully benefit from the potential of those future large-scale complex machines.

We expect advances in three directions based first on strong and closed interactions between the runtime and numerical linear algebra communities. This initial activity will then naturally expand to more focused but still joint research in both fields.

1. Fine interaction between linear algebra and runtime systems. On parallel machines, HPC applications need to take care of data movement and consistency, which can be either explicitly managed at the level of the application itself or delegated to a runtime system. We adopt the latter approach in order to better keep up with hardware trends whose complexity is growing exponentially. One major task in this project is to define a proper interface between HPC applications and runtime systems in order to maximize productivity and expressivity. As mentioned in the next section, a widely used approach consists in abstracting the application as a DAG that the runtime system is in charge of scheduling. Scheduling such a DAG over a set of heterogeneous processing units introduces a lot of new challenges, such as predicting accurately the execution time of each type of task over each kind of unit, minimizing data transfers between MPI will be designed to allow applications to provide scheduling hints to the runtime system and to get real-time feedback about the consequences of scheduling decisions.

2. Runtime systems. A runtime environment is an intermediate layer between the system and the application. It provides low-level functionality not provided by the system (such as scheduling or management of the heterogeneity) and high-level features (such as performance portability). In the framework of this proposal, we will work on the scalability of runtime environment. To achieve scalability it is required to avoid all centralization. Here, the main problem is the scheduling of the tasks. In many task-based runtime environments the scheduler is centralized and becomes a bottleneck as soon as too many cores are involved. It is therefore required to distribute the scheduling decision or to compute a data distribution that impose the mapping of task using, for instance the so-called "owner-compute" rule. Expected advances: We will design runtime systems that enable an efficient and scalable use of thousands of distributed multicore nodes enhanced with accelerators.

3. Linear algebra. Because of its central position in HPC and of the well understood structure of its algorithms, dense linear algebra has often pioneered new challenges that HPC had to face. Again, dense linear algebra has been in the vanguard of the new era of petascale computing with the design of new algorithms that can efficiently run on a multicore node with GPU accelerators. These algorithms are called "communication-avoiding" since they have been redesigned to limit the amount of communication between processing units (and between the different levels of memory hierarchy). They are expressed through Direct Acyclic Graphs (DAG) of fine-grained tasks that are dynamically scheduled. Expected advances: First, we plan to investigate the impact of these principles in the case of sparse applications (whose algorithms are slightly more complicated but often rely on dense kernels). Furthermore, both in the dense and sparse cases, the scalability on thousands of nodes is still limited, new numerical approaches need to be found. We will specifically design sparse hybrid direct/iterative methods that represent a promising approach.

Overall end point. The overall goal of the MORSE associate team is to enable advanced numerical algorithms to be executed on a scalable unified runtime system for exploiting the full potential of future exascale machines.

FUNCTIONAL DESCRIPTION: Chameleon is a dense linear algebra software relying on sequential task-based algorithms where sub-tasks of the overall algorithms are submitted to a Runtime system. A Runtime system such as StarPU is able to manage automatically data transfers between not shared memory area (CPUs-GPUs, distributed nodes). This kind of implementation paradigm allows to design high performing linear algebra algorithms on very different type of architecture: laptop, many-core nodes, CPUs-GPUs, multiple nodes. For example, Chameleon is able to perform a Cholesky factorization (double-precision) at 80 TFlop/s on a dense matrix of order 400 000 (e.i. 4 min).

RELEASE FUNCTIONAL DESCRIPTION: Chameleon includes the following features:

- BLAS 3, LAPACK one-sided and LAPACK norms tile algorithms - Support QUARK and StarPU runtime systems - Exploitation of homogeneous and heterogeneous platforms through the use of BLAS/LAPACK CPU kernels and cuBLAS/MAGMA CUDA kernels - Exploitation of clusters of interconnected nodes with distributed memory (using OpenMPI)

- Participants: Cédric Castagnede, Samuel Thibault, Emmanuel Agullo, Florent Pruvost and Mathieu Faverge
- Partners: Innovative Computing Laboratory (ICL) King Abdullha University of Science and Technology University of Colorado Denver
- Contact: Emmanuel Agullo
- URL: https://project.inria.fr/chameleon/

5.2. hwloc

Hardware Locality

KEYWORDS: NUMA - Multicore - GPU - Affinities - Open MPI - Topology - HPC - Locality

FUNCTIONAL DESCRIPTION: Hardware Locality (hwloc) is a library and set of tools aiming at discovering and exposing the topology of machines, including processors, cores, threads, shared caches, NUMA memory nodes and I/O devices. It builds a widely-portable abstraction of these resources and exposes it to applications so as to help them adapt their behavior to the hardware characteristics. They may consult the hierarchy of resources, their attributes, and bind task or memory on them.

hwloc targets many types of high-performance computing applications, from thread scheduling to placement of MPI processes. Most existing MPI implementations, several resource managers and task schedulers, and multiple other parallel libraries already use hwloc.

- Participants: Brice Goglin and Samuel Thibault
- Partners: Open MPI consortium Intel AMD
- Contact: Brice Goglin
- Publications: hwloc: a Generic Framework for Managing Hardware Affinities in HPC Applications - Managing the Topology of Heterogeneous Cluster Nodes with Hardware Locality (hwloc) - A Topology-Aware Performance Monitoring Tool for Shared Resource Management in Multicore Systems - Exposing the Locality of Heterogeneous Memory Architectures to HPC Applications -Towards the Structural Modeling of the Topology of next-generation heterogeneous cluster Nodes with hwloc - On the Overhead of Topology Discovery for Locality-aware Scheduling in HPC
- URL: http://www.open-mpi.org/projects/hwloc/

5.3. KaStORS

The KaStORS OpenMP Benchmark Suite

KEYWORDS: OpenMP - Task scheduling - Task-based algorithm - HPC - Benchmarking - Data parallelism

FUNCTIONAL DESCRIPTION: The KaStORS benchmarks suite has been designed to evaluate implementations of the OpenMP dependent task paradigm, introduced as part of the OpenMP 4.0 specification.

- Participants: François Broquedis, Nathalie Furmento, Olivier Aumage, Philippe Virouleau, Pierrick Brunet, Samuel Thibault and Thierry Gautier
- Contact: Thierry Gautier
- URL: http://kastors.gforge.inria.fr/#!index.md

5.4. KStar

The KStar OpenMP Compiler

KEYWORDS: Source-to-source compiler - OpenMP - Task scheduling - Compilers - Data parallelism FUNCTIONAL DESCRIPTION: The KStar software is a source-to-source OpenMP compiler for languages C and C++. The KStar compiler translates OpenMP directives and constructs into API calls from the StarPU runtime system or the XKaapi runtime system. The KStar compiler is virtually fully compliant with OpenMP 3.0 constructs. The KStar compiler supports OpenMP 4.0 dependent tasks and accelerated targets.

- Participants: Nathalie Furmento, Olivier Aumage, Samuel Pitoiset and Samuel Thibault
- Contact: Olivier Aumage
- URL: http://kstar.gforge.inria.fr/#!index.md

5.5. MAQAO

SCIENTIFIC DESCRIPTION: MAQAO relies on binary codes for Intel x86 and ARM architectures. For x86 architecture, it can insert probes for instrumention directly inside the binary. There is no need to recompile. The static/dynamic approach of MAQAO analysis is the main originality of the tool, combining performance model with values collected through instrumentation.

MAQAO has a static performance model for x86 and ARM architectures. This model analyzes performance of the codes on the architectures and provides some feed-back hints on how to improve these codes, in particular for vector instructions.

The dynamic collection of data in MAQAO enables the analysis of thread interactions, such as false sharing, amount of data reuse, runtime scheduling policy, ...

FUNCTIONAL DESCRIPTION: MAQAO is a performance tuning tool for OpenMP parallel applications. It relies on the static analysis of binary codes and the collection of dynamic information (such as memory traces). It provides hints to the user about performance bottlenecks and possible workarounds.

- Participants: Christopher Haine, Denis Barthou, James Tombi A Mba and Olivier Aumage
- Contact: Denis Barthou

5.6. StarPU

The StarPU Runtime System KEYWORDS: Multicore - GPU - Scheduling - HPC - Performance SCIENTIFIC DESCRIPTION: Traditional processors have reached architectural limits which heterogeneous multicore designs and hardware specialization (eg. coprocessors, accelerators, ...) intend to address. However, exploiting such machines introduces numerous challenging issues at all levels, ranging from programming models and compilers to the design of scalable hardware solutions. The design of efficient runtime systems for these architectures is a critical issue. StarPU typically makes it much easier for high performance libraries or compiler environments to exploit heterogeneous multicore machines possibly equipped with GPGPUs or Cell processors: rather than handling low-level issues, programmers may concentrate on algorithmic concerns.Portability is obtained by the means of a unified abstraction of the machine. StarPU offers a unified offloadable task abstraction named "codelet". Rather than rewriting the entire code, programmers can encapsulate existing functions within codelets. In case a codelet may run on heterogeneous architectures, it is possible to specify one function for each architectures (eg. one function for CUDA and one function for CPUs). StarPU takes care to schedule and execute those codelets as efficiently as possible over the entire machine. In order to relieve programmers from the burden of explicit data transfers, a high-level data management library enforces memory coherency over the machine: before a codelet starts (eg. on an accelerator), all its data are transparently made available on the compute resource. Given its expressive interface and portable scheduling policies, StarPU obtains portable performances by efficiently (and easily) using all computing resources at the same time. StarPU also takes advantage of the heterogeneous nature of a machine, for instance by using scheduling strategies based on auto-tuned performance models.

StarPU is a task programming library for hybrid architectures

The application provides algorithms and constraints: - CPU/GPU implementations of tasks - A graph of tasks, using either the StarPU's high level GCC plugin pragmas or StarPU's rich C API

StarPU handles run-time concerns - Task dependencies - Optimized heterogeneous scheduling - Optimized data transfers and replication between main memory and discrete memories - Optimized cluster communications

Rather than handling low-level scheduling and optimizing issues, programmers can concentrate on algorithmic concerns!

FUNCTIONAL DESCRIPTION: StarPU is a runtime system that offers support for heterogeneous multicore machines. While many efforts are devoted to design efficient computation kernels for those architectures (e.g. to implement BLAS kernels on GPUs), StarPU not only takes care of offloading such kernels (and implementing data coherency across the machine), but it also makes sure the kernels are executed as efficiently as possible.

- Participants: Corentin Salingue, Andra Hugo, Benoît Lize, Cédric Augonnet, Cyril Roelandt, François Tessier, Jérôme Clet-Ortega, Ludovic Courtes, Ludovic Stordeur, Marc Sergent, Mehdi Juhoor, Nathalie Furmento, Nicolas Collin, Olivier Aumage, Pierre-André Wacrenier, Raymond Namyst, Samuel Thibault, Simon Archipoff and Xavier Lacoste
- Contact: Olivier Aumage
- URL: http://starpu.gforge.inria.fr/

5.7. PARCOACH

PARallel Control flow Anomaly CHecker

KEYWORDS: High-Performance Computing - Program verification - Debug - MPI - OpenMP - Compilation SCIENTIFIC DESCRIPTION: PARCOACH verifies programs in two steps. First, it statically verifies applications with a data- and control-flow analysis and outlines execution paths leading to potential deadlocks. The code is then instrumented, displaying an error and synchronously interrupting all processes if the actual scheduling leads to a deadlock situation.

FUNCTIONAL DESCRIPTION: Supercomputing plays an important role in several innovative fields, speeding up prototyping or validating scientific theories. However, supercomputers are evolving rapidly with now millions of processing units, posing the questions of their programmability. Despite the emergence of more widespread and functional parallel programming models, developing correct and effective parallel applications still remains a complex task. As current scientific applications mainly rely on the Message Passing Interface (MPI) parallel programming model, new hardwares designed for Exascale with higher node-level parallelism clearly advocate for an MPI+X solutions with X a thread-based model such as OpenMP. But integrating two different programming models inside the same application can be error-prone leading to complex bugs - mostly detected unfortunately at runtime. PARallel COntrol flow Anomaly CHecker aims at helping developers in their debugging phase.

- Participants: Emmanuelle Saillard, Denis Barthou and Pierre Huchant
- Partner: CEA
- Contact: Emmanuelle Saillard
- URL: https://esaillar.github.io/PARCOACH/

5.8. AFF3CT

A Fast Forward Error Correction Tool

KEYWORDS: High-Performance Computing - Signal processing - Error Correction Code

FUNCTIONAL DESCRIPTION: AFF3CT proposes high performance Error Correction algorithms for Polar, Turbo, LDPC, RSC (Recursive Systematic Convolutional), Repetition and RA (Repeat and Accumulate) codes. These signal processing codes can be parameterized in order to optimize some given metrics, such as Bit Error Rate, Bandwidth, Latency, ...using simulation. For the designers of such signal processing chain, AFF3CT proposes also high performance building blocks so to develop new algorithms. AFF3CT compiles with many compilers and runs on Windows, Mac OS X, Linux environments and has been optimized for x86 (SSE, AVX instruction sets) and ARM architectures (NEON instruction set).

- Authors: Adrien Cassagne, Bertrand Le Gal, Camille Leroux, Denis Barthou and Olivier Aumage
- Partner: IMS
- Contact: Adrien Cassagne
- URL: https://aff3ct.github.io/

5.9. MORSE

KEYWORDS: High performance computing - Matrix calculation - Fast multipole method - Runtime system FUNCTIONAL DESCRIPTION: MORSE (Matrices Over Runtime Systems @ Exascale) is a scientific project, its objectives are to solve matrix problems on complex architectures, using runtime systems. More specifically, the goal is to write codes that reach a high level of performance for all architectures. The algorithms are written independently of the architecture, and the runtime system dispatches the different computational parts to the different computing units. This methodology has been validated on three classes of problems: dense linear algebra, sparse and dense, and fast multipole methods. The corresponding codes have been incorporated into several softwares, MAGMA, Pastix and ScalFMM.

- Contact: Emmanuel Agullo
- URL: http://icl.cs.utk.edu/morse/

TADaaM Project-Team

6. New Software and Platforms

6.1. Hsplit

Hierarchical communicators split

KEYWORDS: MPI communication - Topology - Hardware platform

SCIENTIFIC DESCRIPTION: Hsplit is a library that implements an abstraction allowing the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.

FUNCTIONAL DESCRIPTION: Hsplit implements an abstraction that allows the programmer using MPI in their parallel applications to access the underlying hardware structure through a hierarchy of communicators. Hsplit is based on the MPI_Comm_split_type routine and provides a new value for the split_type argument that specifically creates a hierarchy a subcommunicators where each new subcommunicator corresponds to a meaningful hardware level. The important point is that only the structure o the hardware is exploited and the number of levels or the levels names are not fixed so as to propose a solution independent from future hardware evolutions (such as new levels for instance). Another flavor of this MPI_Comm_split_type function is provided that creates a roots communicators at the same time a subcommunicator is produced, in order to ease the collective communication and/or synchronization among subcommunicators.

NEWS OF THE YEAR: A new working group in the MPI Forum to champion the integration of this proposal in the MPI standard has been created. This working group includes Inria, CEA, Atos/Bull, Paratools, the University of Tennessee - Knoxville and many other institutions/companies are interested to join in.

- Participants: Guillaume Mercier, Brice Goglin, Emmanuel Jeannot and Farouk Mansouri
- Contact: Guillaume Mercier
- Publications: A hierarchical model to manage hardware topology in MPI applications A Hierarchical Model to Manage Hardware Topology in MPI Applications
- URL: http://mpi-topology.gforge.inria.fr/

resources, their attributes, and bind task or memory on them.

6.2. hwloc

Hardware Locality

KEYWORDS: NUMA - Multicore - GPU - Affinities - Open MPI - Topology - HPC - Locality FUNCTIONAL DESCRIPTION: Hardware Locality (hwloc) is a library and set of tools aiming at discovering and exposing the topology of machines, including processors, cores, threads, shared caches, NUMA memory nodes and I/O devices. It builds a widely-portable abstraction of these resources and exposes it to applications so as to help them adapt their behavior to the hardware characteristics. They may consult the hierarchy of

45

hwloc targets many types of high-performance computing applications, from thread scheduling to placement of MPI processes. Most existing MPI implementations, several resource managers and task schedulers, and multiple other parallel libraries already use hwloc.

- Participants: Brice Goglin and Samuel Thibault
- Partners: Open MPI consortium Intel AMD
- Contact: Brice Goglin
- Publications: hwloc: a Generic Framework for Managing Hardware Affinities in HPC Applications - Managing the Topology of Heterogeneous Cluster Nodes with Hardware Locality (hwloc) - A Topology-Aware Performance Monitoring Tool for Shared Resource Management in Multicore Systems - Exposing the Locality of Heterogeneous Memory Architectures to HPC Applications -Towards the Structural Modeling of the Topology of next-generation heterogeneous cluster Nodes with hwloc - On the Overhead of Topology Discovery for Locality-aware Scheduling in HPC
- URL: http://www.open-mpi.org/projects/hwloc/

6.3. NetLoc

Network Locality

KEYWORDS: Topology - Locality - Distributed networks - HPC - Parallel computing - MPI communication FUNCTIONAL DESCRIPTION: netloc (Network Locality) is a library that extends hwloc to network topology information by assembling hwloc knowledge of server internals within graphs of inter-node fabrics such as Infiniband, Intel OmniPath or Cray networks.

Netloc builds a software representation of the entire cluster so as to help applications properly place their tasks on the nodes. It may also help communication libraries optimize their strategies according to the wires and switches.

Netloc targets the same challenges as hwloc but focuses on a wider spectrum by enabling cluster-wide solutions such as process placement. It interoperates with the Scotch graph partitioner to do so.

Netloc is distributed within hwloc releases starting with hwloc 2.0.

- Participants: Brice Goglin, Clement Foyer and Cyril Bordage
- Contact: Brice Goglin
- Publications: netloc: Towards a Comprehensive View of the HPC System Topology Netloc: a Tool for Topology-Aware Process Mapping
- URL: http://www.open-mpi.org/projects/netloc/

6.4. NewMadeleine

KEYWORDS: High-performance calculation - MPI communication

FUNCTIONAL DESCRIPTION: NewMadeleine is the fourth incarnation of the Madeleine communication library. The new architecture aims at enabling the use of a much wider range of communication flow optimization techniques. Its design is entirely modular: drivers and optimization strategies are dynamically loadable software components, allowing experimentations with multiple approaches or on multiple issues with regard to processing communication flows.

The optimizing scheduler SchedOpt targets applications with irregular, multi-flow communication schemes such as found in the increasingly common application conglomerates made of multiple programming environments and coupled pieces of code, for instance. SchedOpt itself is easily extensible through the concepts of optimization strategies (what to optimize for, what the optimization goal is) expressed in terms of tactics (how to optimize to reach the optimization goal). Tactics themselves are made of basic communication flows operations such as packet merging or reordering.

The communication library is fully multi-threaded through its close integration with PIOMan. It manages concurrent communication operations from multiple libraries and from multiple threads. Its MPI implementation Mad-MPI fully supports the MPI_THREAD_MULTIPLE multi-threading level.

- Participants: Alexandre Denis, Clement Foyer, Nathalie Furmento and Raymond Namyst
- Contact: Alexandre Denis
- URL: http://pm2.gforge.inria.fr/newmadeleine/

6.5. PaMPA

Parallel Mesh Partitioning and Adaptation

KEYWORDS: Dynamic load balancing - Unstructured heterogeneous meshes - Parallel remeshing - Subdomain decomposition - Parallel numerical solvers

SCIENTIFIC DESCRIPTION: PaMPA is a parallel library for handling, redistributing and remeshing unstructured meshes on distributed-memory architectures. PaMPA dramatically eases and speeds-up the development of parallel numerical solvers for compact schemes. It provides solver writers with a distributed mesh abstraction and an API to: - describe unstructured and possibly heterogeneous meshes, on the form of a graph of interconnected entities of different kinds (e.g. elements, faces, edges, nodes), - attach values to the mesh entities, - distribute such meshes across processing elements, with an overlap of variable width, - perform synchronous or asynchronous data exchanges of values across processing elements, - describe numerical schemes by means of iterators over mesh entities and their connected neighbors of a given kind, - redistribute meshes so as to balance computational load, - perform parallel dynamic remeshing, by applying adequately a user-provided sequential remesher to relevant areas of the distributed mesh.

PaMPA runs concurrently multiple sequential remeshing tasks to perform dynamic parallel remeshing and redistribution of very large unstructured meshes. E.g., it can remesh a tetrahedral mesh from 43Melements to more than 1Belements on 280 Broadwell processors in 20 minutes.

FUNCTIONAL DESCRIPTION: Parallel library for handling, redistributing and remeshing unstructured, heterogeneous meshes on distributed-memory architectures. PaMPA dramatically eases and speeds-up the development of parallel numerical solvers for compact schemes.

NEWS OF THE YEAR: PaMPA has been used to remesh an industrial mesh of a helicopter turbine combustion chamber, up to more than 1 billion elements.

- Participants: Cécile Dobrzynski, Cedric Lachat and François Pellegrini
- Partners: Université de Bordeaux CNRS IPB
- Contact: Cedric Lachat
- URL: http://project.inria.fr/pampa/

6.6. TreeMatch

KEYWORDS: Intensive parallel computing - High-Performance Computing - Hierarchical architecture - Placement

SCIENTIFIC DESCRIPTION: TreeMatch embeds a set of algorithms to map processors/cores in order to minimize the communication cost of the application.

Important features are : the number of processors can be greater than the number of applications processes, it assumes that the topology is a tree and does not require valuation of the topology (e.g. communication speeds), it implements different placement algorithms that are switched according to the input size.

Some core algorithms are parallel to speed-up the execution. Optionally embeds scotch for fix-vertex mapping. enable exhaustive search if required. Several metric mapping are computed. Allow for oversubscribing of ressources. multithreaded.

TreeMatch is integrated into various software such as the Charm++ programming environment as well as in both major open-source MPI implementations: Open MPI and MPICH2.

FUNCTIONAL DESCRIPTION: TreeMatch is a library for performing process placement based on the topology of the machine and the communication pattern of the application.

- Participants: Adele Villiermet, Emmanuel Jeannot, François Tessier, Guillaume Mercier and Pierre Celor
- Partners: Université de Bordeaux CNRS IPB
- Contact: Emmanuel Jeannot
- URL: http://treematch.gforge.inria.fr/

6.7. SCOTCH

KEYWORDS: Mesh partitioning - Domain decomposition - Graph algorithmics - High-performance calculation - Sparse matrix ordering

FUNCTIONAL DESCRIPTION: Scotch is a graph partitioner. It helps optimise the division of a problem, by means of a graph, into a set of independent sub-problems of equivalent sizes. These sub-problems can also be solved at the same time.

RELEASE FUNCTIONAL DESCRIPTION: Version 6.0 offers many new features:

sequential graph repartitioning

sequential graph partitioning with fixed vertices

sequential graph repartitioning with fixed vertices

new, fast, direct k-way partitioning and mapping algorithms

multi-threaded, shared memory algorithms in the (formerly) sequential part of the library

exposure in the API of many distributed graph handling routines

embedded pseudo-random generator for improved reproducibility

and even more

NEWS OF THE YEAR: In the context of the PhD of Rémi Barat, the sequential version of Scotch has been extended so as to manage graphs with multiple vertex weights, and multi-constraint graph partitioning algorithms have been implemented as prototypes.

- Participants: Sébastien Fourestier, François Pellegrini and Cédric Chevalier
- Partners: CNRS IPB Region Aquitaine
- Contact: François Pellegrini
- Publications: Process Mapping onto Complex Architectures and Partitions Thereof Contributions au partitionnement de graphes parallèle multi-niveaux Adaptation au repartitionnement de graphes d'une méthode d'optimisation globale par diffusion A parallelisable multi-level banded diffusion scheme for computing balanced partitions with smooth boundaries PT-Scotch: A tool for efficient parallel graph ordering Design and implementation of efficient tools for parallel partitioning and distribution of very large numerical problems PT-Scotch : Un outil pour la renumérotation parallèle efficace de grands graphes dans un contexte multi-niveaux PT-Scotch: A tool for efficient parallel graph ordering Improvement of the Efficiency of Genetic Algorithms for Scalable Parallel Graph Partitioning in a Multi-Level Framework
- URL: http://www.labri.fr/~pelegrin/scotch/

ASCOLA Project-Team

6. New Software and Platforms

6.1. btrCloud

KEYWORDS: Cloud computing - Data center - Cluster - Placement - Autonomic system - Orchestration - Energy - Grid - Virtualization - Scheduler

FUNCTIONAL DESCRIPTION: Orchestration, virtualization, energy, autonomic system, placement, cloud computing, cluster, data center, scheduler, grid

btrCloud is a virtual machine manager for clusters and provides a complete solution for the management and optimization of virtualized data centers. btrCloud (acronym of better cloud) is composed of three parts.

The analysis function enables operatives and people in charge to monitor and analyze how a data-center works - be it on a daily basis, on the long run, or in order to predict future trends. This feature includes boards for performance evaluation and analysis as well as trends estimation.

btrCloud, by the integration of btrScript, provides (semi-)automated VM lifecycle management, including provisioning, resource pool management, VM tracking, cost accounting, and scheduled deprovisioning. Key features include a thin client interface, template-based provisioning, approval workflows, and policy-based VM placement.

Finally, several kinds of optimizations are currently available, such as energy and load balancing. The former can help save up to around 20

- Participants: Frédéric Dumont, Guillaume Le Louët and Jean-Marc Menaud
- Contact: Guillaume Le Louët
- URL: http://www.btrcloud.org/btrCloud/index_EN.html

6.2. SimGrid

KEYWORDS: Large-scale Emulators - Grid Computing - Distributed Applications

SCIENTIFIC DESCRIPTION: SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. The simulation engine uses algorithmic and implementation techniques toward the fast simulation of large systems on a single machine. The models are theoretically grounded and experimentally validated. The results are reproducible, enabling better scientific practices.

Its models of networks, cpus and disks are adapted to (Data)Grids, P2P, Clouds, Clusters and HPC, allowing multi-domain studies. It can be used either to simulate algorithms and prototypes of applications, or to emulate real MPI applications through the virtualization of their communication, or to formally assess algorithms and applications that can run in the framework.

The formal verification module explores all possible message interleavings in the application, searching for states violating the provided properties. We recently added the ability to assess liveness properties over arbitrary and legacy codes, thanks to a system-level introspection tool that provides a finely detailed view of the running application to the model checker. This can for example be leveraged to verify both safety or liveness properties, on arbitrary MPI code written in C/C++/Fortran.

RELEASE FUNCTIONAL DESCRIPTION:

- Four releases in 2017. Major changes:
 - S4U: many progress, toward SimGrid v4.0. About 80% of the features offered by SimDag and MSG are now integrated, along with examples. Users can now write plugins to extend SimGrid.
 - SMPI: Support MPI 2.2, RMA support, Convert internals to C++.
 - Java: Massive memleaks and performance issues fixed.
 - New models: Multi-core VMs, Energy consumption due to the network
 - All internals are now converted to C++, and most of our internally developped data containers were replaced with std::* constructs.
 - (+ bug fixes, cleanups and documentation improvements)
- Participants: Adrien Lèbre, Arnaud Legrand, Augustin Degomme, Florence Perronnin, Frédéric Suter, Jean-Marc Vincent, Jonathan Pastor, Jonathan Rouzaud-Cornabas, Luka Stanisic, Mario Südholt and Martin Quinson
- Partners: CNRS ENS Rennes
- Contact: Martin Quinson
- URL: http://simgrid.gforge.inria.fr/

6.3. VMPlaces

FUNCTIONAL DESCRIPTION: VMPlaces is a dedicated framework to evaluate and compare VM placement algorithms. This framework is composed of two major components: the injector and the VM placement algorithm. The injector is the generic part of the framework (i.e. the one you can directly use) while the VM placement algorithm is the part you want to study (or compare with available algorithms). Currently, the VMPlaceS is released with three algorithms:

Entropy, a centralized approach using a constraint programming approach to solve the placement/reconfiguration VM problem

Snooze, a hierarchical approach where each manager of a group invokes Entropy to solve the placement/reconfiguration VM problem. Note that in the original implementation of Snooze, it is using a specific heuristic to solve the placement/reconfiguration VM problem. As the sake of simplicity, we have simply reused the entropy scheduling code.

DVMS, a distributed approach that dynamically partitions the system and invokes Entropy on each partition.

- Participants: Adrien Lèbre, Flavien Quesnel, Jonathan Pastor, Mario Südholt and Takahiro Hirofuchi
- Contact: Adrien Lèbre
- URL: http://beyondtheclouds.github.io/VMPlaceS/

6.4. ENOS

Experimental eNvironment for OpenStack KEYWORDS: OpenStack - Experimentation - Reproducibility FUNCTIONAL DESCRIPTION: Enos workflow :

A typical experiment using Enos is the sequence of several phases:

- enos up : Enos will read the configuration file, get machines from the resource provider and will prepare the next phase - enos os : Enos will deploy OpenStack on the machines. This phase rely highly on Kolla deployment. - enos init-os : Enos will bootstrap the OpenStack installation (default quotas, security rules, ...)
- enos bench : Enos will run a list of benchmarks. Enos support Rally and Shaker benchmarks. - enos backup : Enos will backup metrics gathered, logs and configuration files from the experiment.

- Partner: Orange Labs
- Contact: Adrien Lèbre

DIVERSE Project-Team

6. New Software and Platforms

6.1. amiunique

KEYWORDS: Privacy - Browser fingerprinting

SCIENTIFIC DESCRIPTION: The amiunique web site has been deployed in the context of the DiverSE's research activities on browser fingerprinting and how software diversity can be leveraged in order to mitigate the impact of fingerprinting on the privacy of users. The construction of a dataset of genuine fingerprints is essential to understand in details how browser fingerprints can serve as unique identifiers and hence what should be modified in order to mitigate its impact privacy. This dataset also supports the large-scale investigation of the impact of web technology advances on fingerprinting. For example, we can analyze in details the impact of the HTML5 canvas element or the behavior of fingerprinting on mobile devices.

The whole source code of amiunique is open source and is distributed under the terms of the MIT license.

Similar sites: Panopticlick https://panopticlick.eff.org/ BrowserSpy http://browserspy.dk/ http://noc.to/ Main innovative features: canvas fingerprinting WebGL fingerprinting advanced JS features (platform, DNT, etc.)

Impact: The website has been showcased in several professional forums in 2014 and 2015 (Open World Forum 2014, FOSSA'14, FIC'15, ICT'15) and it has been visited by more than 100000 unique visitors in one year. FUNCTIONAL DESCRIPTION: This web site aims at informing visitors about browser fingerprinting and possible tools to mitigate its effect, as well as at collecting data about the fingerprints that can be found on the web. It collects browser fingerprints with the explicit agreement of the users (they have to click on a button on the home page). Fingerprints are composed of 17 attributes, which include regular HTTP headers as well as the most recent state of the art techniques (canvas fingerprinting, WebGL information).

- Participants: Benoit Baudry and Pierre Laperdrix
- Partner: INSA Rennes
- Contact: Benoit Baudry
- URL: https://amiunique.org/

6.2. FAMILIAR

KEYWORDS: Software line product - Configators - Customisation

SCIENTIFIC DESCRIPTION: FAMILIAR (for FeAture Model scrIpt Language for manIpulation and Automatic Reasoning) is a language for importing, exporting, composing, decomposing, editing, configuring, computing "diffs", refactoring, reverse engineering, testing, and reasoning about (multiple) feature models. All these operations can be combined to realize complex variability management tasks. A comprehensive environment is proposed as well as integration facilities with the Java ecosystem.

FUNCTIONAL DESCRIPTION: Familiar is an environment for large-scale product customisation. From a model of product features (options, parameters, etc.), Familiar can automatically generate several million variants. These variants can take many forms: software, a graphical interface, a video sequence or even a manufactured product (3D printing). Familiar is particularly well suited for developing web configurators (for ordering customised products online), for providing online comparison tools and also for engineering any family of embedded or software-based products.

- Participants: Aymeric Hervieu, Benoit Baudry, Didier Vojtisek, Edward Mauricio Alferez Salinas, Guillaume Bécan, Joao Bosco Ferreira-Filho, Julien Richard-Foy, Mathieu Acher, Olivier Barais and Sana Ben Nasr
- Contact: Mathieu Acher
- URL: http://familiar-project.github.com

52 Distributed programming and Software engineering - Software and Platforms - Project-Team DIVERSE

6.3. GEMOC Studio

KEYWORDS: DSL - Language workbench - Model debugging

SCIENTIFIC DESCRIPTION: The language workbench put together the following tools seamlessly integrated to the Eclipse Modeling Framework (EMF):

- Melange, a tool-supported meta-language to modularly define executable modeling languages with execution functions and data, and to extend (EMF-based) existing modeling languages. - MoCCML, a tool-supported meta-language dedicated to the specification of a Model of Concurrency and Communication (MoCC) and its mapping to a specific abstract syntax and associated execution functions of a modeling language. - GEL, a tool-supported meta-language dedicated to the specification of the protocol between the execution functions and the MoCC to support the feedback of the data as well as the callback of other expected execution functions. - BCOoL, a tool-supported meta-language dedicated to the specification of language coordination patterns to automatically coordinates the execution of, possibly heterogeneous, models. - Sirius Animator, an extension to the model editor designer Sirius to create graphical animators for executable modeling languages.

FUNCTIONAL DESCRIPTION: The GEMOC Studio is an eclipse package that contains components supporting the GEMOC methodology for building and composing executable Domain-Specific Modeling Languages (DSMLs). It includes the two workbenches: The GEMOC Language Workbench: intended to be used by language designers (aka domain experts), it allows to build and compose new executable DSMLs. The GEMOC Modeling Workbench: intended to be used by domain designersto create, execute and coordinate models conforming to executable DSMLs. The different concerns of a DSML, as defined with the tools of the language workbench, are automatically deployed into the modeling workbench. They parametrize a generic execution framework that provide various generic services such as graphical animation, debugging tools, trace and event managers, timeline, etc.

- Participants: Didier Vojtisek, Dorian Leroy, Erwan Bousse, Fabien Coulon and Julien Deantoni
- Partners: IRIT ENSTA I3S OBEO Thales TRT
- Contact: Benoît Combemale
- URL: http://gemoc.org/studio.html

6.4. Kevoree

Kevoree Core

KEYWORDS: M2M - Dynamic components - Iot - Heterogeneity - Smart home - Cloud - Software architecture - Dynamic deployment

SCIENTIFIC DESCRIPTION: Kevoree is an open-source models@runtime platform (http://www.kevoree.org) to properly support the dynamic adaptation of distributed systems. Models@runtime basically pushes the idea of reflection [132] one step further by considering the reflection layer as a real model that can be uncoupled from the running architecture (e.g. for reasoning, validation, and simulation purposes) and later automatically resynchronized with its running instance.

Kevoree has been influenced by previous work that we carried out in the DiVA project [132] and the Entimid project [135]. With Kevoree we push our vision of models@runtime [131] farther. In particular, Kevoree provides a proper support for distributed models@runtime. To this aim we introduced the Node concept to model the infrastructure topology and the Group concept to model semantics of inter node communication during synchronization of the reflection model among nodes. Kevoree includes a Channel concept to allow for multiple communication semantics between remoteComponents deployed on heterogeneous nodes. All Kevoree concepts (Component, Channel, Node, Group) obey the object type design pattern to separate deployment artifacts from running artifacts. Kevoree supports multiple kinds of very different execution node technology (e.g. Java, Android, MiniCloud, FreeBSD, Arduino, ...).

Kevoree is distributed under the terms of the LGPL open source license.

Main competitors:

- the Fractal/Frascati eco-system (http://frascati.ow2.org).
- SpringSource Dynamic Module (http://spring.io/)

GCM-Proactive (http://proactive.inria.fr/)

OSGi (http://www.osgi.org)

Chef

Vagran (http://vagrantup.com/)

Main innovative features:

distributed models@runtime platform (with a distributed reflection model and an extensible models@runtime dissemination set of strategies).

Support for heterogeneous node type (from Cyber Physical System with few resources until cloud computing infrastructure).

Fully automated provisioning model to correctly deploy software modules and their dependencies.

Communication and concurrency access between software modules expressed at the model level (not in the module implementation).

Impact:

Several tutorials and courses have been performed this year at EJCP for French PhD student, at ECNU summer school for 82 chineese PhD students. See also the web page http://www.kevoree.org .

In 2015, we mainly created a new implementation in C# and we created an implementation for system containers for driving resources using Kevoree. We also use Kevoree in the context of Mohammed's PhD to create testing infrastructure on-demand.

FUNCTIONAL DESCRIPTION: Kevoree is an open-source models@runtime platform to properly support the dynamic adaptation of distributed systems. Models@runtime basically pushes the idea of reflection one step further by considering the reflection layer as a real model that can be uncoupled from the running architecture (e.g. for reasoning, validation, and simulation purposes) and later automatically resynchronized with its running instance.

- Participants: Aymeric Hervieu, Benoit Baudry, Francisco-Javier Acosta Padilla, Inti Gonzalez Herrera, Ivan Paez Anaya, Jacky Bourgeois, Jean Emile Dartois, Johann Bourcier, Manuel Leduc, Maxime Tricoire, Mohamed Boussaa, Noël Plouzeau and Olivier Barais
- Contact: Olivier Barais
- URL: http://kevoree.org/

6.5. Melange

KEYWORDS: Modeling language - Meta-modelisation - Language workbench - Dedicated langage - Modeldriven software engineering - DSL - MDE - Meta model - Model-driven engineering - Meta-modeling SCIENTIFIC DESCRIPTION: Melange is a follow-up of the executable metamodeling language Kermeta, which provides a tool-supported dedicated meta-language to safely assemble language modules, customize them and produce new DSMLs. Melange provides specific constructs to assemble together various abstract syntax and operational semantics artifacts into a DSML. DSMLs can then be used as first class entities to be reused, extended, restricted or adapted into other DSMLs. Melange relies on a particular model-oriented type system that provides model polymorphism and language substitutability, i.e. the possibility to manipulate a model through different interfaces and to define generic transformations that can be invoked on models written using different DSLs. Newly produced DSMLs are correct by construction, ready for production (i.e., the result can be deployed and used as-is), and reusable in a new assembly. Melange is tightly integrated with the Eclipse Modeling Framework ecosystem and relies on the meta-language Ecore for the definition of the abstract syntax of DSLs. Executable meta-modeling is supported by weaving operational semantics defined with Xtend. Designers can thus easily design an interpreter for their DSL in a non-intrusive way. Melange is bundled as a set of Eclipse plug-ins.

FUNCTIONAL DESCRIPTION: Melange is a language workbench which helps language engineers to mashup their various language concerns as language design choices, to manage their variability, and support their reuse. It provides a modular and reusable approach for customizing, assembling and integrating DSMLs specifications and implementations.

- Participants: Arnaud Blouin, Benoît Combemale, David Mendez Acuna, Didier Vojtisek, Dorian Leroy, Erwan Bousse, Fabien Coulon, Jean-Marc Jézéquel, Olivier Barais and Thomas Degueule
- Contact: Benoît Combemale
- URL: http://melange-lang.org

6.6. Opencompare

KEYWORD: Software Product Line

FUNCTIONAL DESCRIPTION: Product comparison matrices (PCMs) are tabular data: supported and unsupported features are documented for both describing the product itself and for discriminating one product compared to another. PCMs abound – we are all using PCMs – and constitute a rich source of knowledge for easily comparing and choosing product. Yet the current practice is suboptimal both for humans and computers, mainly due to unclear semantics, heterogeneous forms of data, and lack of dedicated support.

OpenCompare.org is an ambitious project for the collaborative edition, the sharing, the standardisation, and the open exploitation of PCMs. The goal of OpenCompare.org is to provide an integrated set of tools (e.g., APIs, visualizations, configurators, editors) for democratizing their creation, import, maintenance, and exploitation.

- Participants: Guillaume Bécan, Mathieu Acher and Sana Ben Nasr
- Contact: Mathieu Acher
- URL: http://opencompare.org

FOCUS Project-Team

6. New Software and Platforms

6.1. HoCA

Higher-Order Complexity Analysis

KEYWORDS: Ocaml - Verification - Runtime Complexity Analysis

SCIENTIFIC DESCRIPTION: Over the last decade, various tools for the static analysis of resource properties of programs have emerged. In particular, the rewriting community has recently developed several tools for the time complexity analysis of term rewrite systems. These tools have matured and are nowadays able to treat non-trivial programs, in a fully automatic setting. However, none of these automatic complexity analysers can deal with higher-order functions, a pervasive feature of functional programs. HoCA (Higher-Order Complexity Analyser) overcomes this limitation by translating higher-order programs – in the form of side-effect free OCaml programs - into equivalent first-order rewrite systems. At the heart of our tool lies Reynold's defunctionalization technique. Defunctionalization however is not enough. Resulting programs have a recursive structure too complicated to be analysed automatically in all but trivial cases. To overcome this issue, HoCA integrates a handful of well established program transformation techniques, noteworthy dead-code elimination, inlining, instantiation and uncurrying. A complexity bound on the resulting first-order program can be relayed back reliably to the higher-order program of interest. A detailed description of HoCA is available on http://arxiv.org/abs/1506.05043.

FUNCTIONAL DESCRIPTION: HoCA is an abbreviation for Higher-Order Complexity Analysis, and is meant as a laboratory for the automated complexity analysis of higher-order functional programs. Currently, HoCA consists of one executable pcf2trs which translates a pure subset of OCaml to term rewrite systems, in a complexity reflecting manner. As a first step, HoCA desugars the given program to a variation of Plotkin's PCF with data-constructors. Via Reynold's defunctionalization, the PCF program is turned into an applicative term rewrite system (ATRS for short), call-by-value reductions of the PCF program are simulated by the ATRS step-by-step, on the ATRS, and various complexity reflecting transformations are performed: inlining, deadcode-elminiation, instantiation of higher-order variables through a call-flow-analysis and finally uncurrying. This results finally in a first-order rewrite system, whose runtime-complexity reflects the complexity of the initial program, asymptotically.

- Participants: Martin Avanzini and Ugo Dal Lago
- Contact: Ugo Dal Lago
- URL: http://cbr.uibk.ac.at/tools/hoca/

6.2. JOLIE

Java Orchestration Language Interpreter Engine

KEYWORD: Microservices

SCIENTIFIC DESCRIPTION: Jolie is a service-oriented programming language. Jolie can be used to program services that interact over the Internet using different communication protocols.

Differently from other Web Services programming languages such as WS-BPEL, Jolie is based on a userfriendly C/Java-like syntax (more readable than the verbose XML syntax of WS-BPEL) and, moreover, the language is equipped with a formal operational semantics. This language is used for the *proof of concepts* developed around Focus activities. For instance, contract theories can be exploited for checking the conformance of a Jolie program with respect to a given contract.

FUNCTIONAL DESCRIPTION: Developments in 2017: 2017 has seen many efforts around the language to increase its usage in industry. These include:

- Organisation of two events. One in Italy, called Meeting on Microservices, organised by italianaSoftware and Monrif SpA in December 2016. The second one in Denmark, organised by Southern Denmark University and Università di Bologna in October 2017. Common aim of both events was presenting the language from a practical, industrial point of view, to illustrate with real-world cases how its abstractions can increase productivity of companies. Both venues contributed in growing the community of companies that have adopted the language or plan to adopt it in the near future.

- Revision of the language documentation, migrating it to GitBook. In this way, Jolie users can access its documentation as HTML pages, as a PDF, and as an eBook. The choice of GitBook has been guided by the need to give a proper tool to users to collaborate, discuss, and request fixes and extensions on the documentation.

- Development of several tools, frameworks, and libraries to ease the management of architectures of microservices. The main ones are:

- the publication of libraries to interact with and orchestrate the Docker containerisation technology. This work, called Jocker, has been the fulcrum of other projects that streamline the creation and management of container-based microservice architectures,

- the publication of a fundamental companion for any industrial-grade language: a packing system. The project, called jpm, automatises the process of publishing, installing, upgrading, configuring, and removing libraries in Jolie software projects,

- the inclusion in the language interpreter of hooks for modular, distributed tracing, a renowned problem of microservices and distributed systems. Developed to output program traces in JSON, this work maintained an open perspective on both output formats and logging deployment, which can be extended in a modular way. The project also includes a visualiser of several distributed traces for debugging purposes,

- the publication of a unit testing framework for microservices, a fundamental building block for continuous integration processes. This framework includes also functionalities to automatically test microservices within a distributed, sandboxed environment, thanks to its integration with Jocker. The framework is also the first step towards a more comprehensive suite to test complete microservice architectures,

- the creation of a deployment framework that automatises the deployment of microservice architectures. This is an important issue in microservice and distributed system deployment, where correctly installing programs on execution nodes and making sure they are properly linked to each other is a daunting and time-consuming task. The framework, given a deployment schema, i) automatises the creation of containers where one or more microservices coexist, ii) deploys the containers into assigned machines, and iii) binds the deployed containers so that microservices within different containers can communicate,

- the creation of the Jiot project, aimed at integrating IoT-related technologies into the Jolie language. The final goal is to provide easy-to-use and flexible communication abstractions to interconnect and make interact disparate IoT islands. Work in 2017 comprised the inclusion of the CoAP/UDP and MQTT/TCP protocols among the communication technologies supported by the language.

Jolie also transitioned from version 1.6 to 1.6.2, which are minor releases, however they contain many performance optimisations and bug fixes.

RELEASE FUNCTIONAL DESCRIPTION: There are many fixes to the HTTP extension, improvements to the embedding engine for Javascript programs, and improvements to the support tools jolie2java and wsdl2jolie.

- Participants: Claudio Guidi, Fabrizio Montesi, Maurizio Gabbrielli and Saverio Giallorenzo
- Contact: Fabrizio Montesi
- URL: http://www.jolie-lang.org/

6.3. NightSplitter

KEYWORD: Constraint-based programming

FUNCTIONAL DESCRIPTION: Nightsplitter deals with the group preference optimization problem. We propose to split users into subgroups trying to optimize members' satisfaction as much as possible. In a large city with a huge volume of activity information, designing subgroup activities and avoiding time conflict is a challenging task. Currently, the Demo is available only for restaurant and movie activities in the city of Paris.

- Contact: Tong Liu
- URL: http://cs.unibo.it/t.liu/nightsplitter/

6.4. AIOCJ

Adaptive Interaction-Oriented Choreographies in Jolie

SCIENTIFIC DESCRIPTION: AIOCJ is a framework for programming adaptive distributed systems based on message passing. AIOCJ comes as a plugin for Eclipse, AIOCJ-ecl, allowing to edit descriptions of distributed systems as adaptive interaction-oriented choreographies (AIOC). From interaction-oriented choreographies the description of single participants can be automatically derived. Adaptation is specified by rules allowing to replace predetermined parts of the AIOC with a new behaviour. A suitable protocol ensures that all the participants are updated in a coordinated way. As a result, the distributed system follows the specification given by the AIOC under all changing sets of adaptation rules and environment conditions. In particular, the system is always deadlock-free. AIOCJ can interact with external services, seen as functions, by specifying their URL and the protocol they support (HTTP, SOAP, ...). Deadlock-freedom guarantees of the application are preserved provided that those services do not block.

FUNCTIONAL DESCRIPTION: AIOCJ is an open-source choreography programming language for developing adaptive systems.

- Participants: Ivan Lanese, Jacopo Mauro, Maurizio Gabbrielli, Mila Dalla Preda and Saverio Giallorenzo
- Contact: Saverio Giallorenzo
- URL: http://www.cs.unibo.it/projects/jolie/aiocj.html

6.5. CauDEr

Causal-consistent Debugger for Erlang

KEYWORDS: Debug - Reversible computing

SCIENTIFIC DESCRIPTION: The reversible debugger is based on the theory of causal-consistent reversibility, which states that any action can be undone provided that its consequences, if any, are undone beforehand. This theory relies on a causal semantic for the target language, and can be used even if different processes have different notions of time

FUNCTIONAL DESCRIPTION: CauDEr is a debugger allowing one to explore the execution of concurrent Erlang programs both forward and backward. Notably, when going backward, any action can be undone provided that its consequences, if any, are undone beforehand. This enables one to find a bug by following the causality links from the visible misbehaviour to the bug. The debugger takes an Erlang program but debugging is done on its translation into Core Erlang.

- Partner: Universitat Politècnica de València
- Contact: Ivan Lanese
- URL: https://github.com/mistupv/cauder

6.6. SUNNY-AS

SUNNY FOR ALGORITHM SELECTION

KEYWORDS: Optimisation - Machine learning

FUNCTIONAL DESCRIPTION: SUNNY-AS is a portfolio solver derived from SUNNY-CP for Algorithm Selection Problems (ASLIB). The goal of SUNNY-AS is to provide a flexible, configurable, and usable portfolio solver that can be set up and executed just like a regular individual solver.

- Contact: Tong Liu
- URL: https://github.com/lteu/oasc

INDES Project-Team

4. New Software and Platforms

4.1. Bigloo

KEYWORD: Compilers

FUNCTIONAL DESCRIPTION: Bigloo is a Scheme implementation devoted to one goal: enabling Scheme based programming style where C(++) is usually required. Bigloo attempts to make Scheme practical by offering features usually presented by traditional programming languages but not offered by Scheme and functional programming. Bigloo compiles Scheme modules. It delivers small and fast stand alone binary executables. Bigloo enables full connections between Scheme and C programs, between Scheme and Java programs.

RELEASE FUNCTIONAL DESCRIPTION: modification of the object system (language design and implementation), new APIs (alsa, flac, mpg123, avahi, csv parsing), new library functions (UDP support), new regular expressions support, new garbage collector (Boehm's collection 7.3alpha1).

- Participant: Manuel Serrano
- Contact: Manuel Serrano
- URL: http://www-sop.inria.fr/teams/indes/fp/Bigloo/

4.2. Camloo

KEYWORD: Compilers

FUNCTIONAL DESCRIPTION: Camloo is a caml-light to bigloo compiler, which was developed a few years ago to target bigloo 1.6c. New major releases 0.4.x of camloo have been done to support bigloo 3.4 and bigloo 3.5. Camloo makes it possible for the user to develop seamlessly a multi-language project, where some files are written in caml-light, in C, and in bigloo. Unlike the previous versions of camloo, 0.4.x versions do not need a modified bigloo compiler to obtain good performance. Currently, the only supported backend for camloo is bigloo/C. We are currently rewriting the runtime of camloo in bigloo to get more portability and to be able to use HOP and camloo together.

• Contact: Manuel Serrano

4.3. Hop

KEYWORDS: Programming language - Multimedia - Iot - Web 2.0 - Functional programming

SCIENTIFIC DESCRIPTION: The Hop programming environment consists in a web broker that intuitively combines in a single architecture a web server and a web proxy. The broker embeds a Hop interpreter for executing server-side code and a Hop client-side compiler for generating the code that will get executed by the client.

An important effort is devoted to providing Hop with a realistic and efficient implementation. The Hop implementation is validated against web applications that are used on a daily-basis. In particular, we have developed Hop applications for authoring and projecting slides, editing calendars, reading RSS streams, or managing blogs.

FUNCTIONAL DESCRIPTION: Multitier web programming language and runtime environment.

- Participant: Manuel Serrano
- Contact: Manuel Serrano
- URL: http://hop.inria.fr

4.4. IFJS

Infomation Flow monitor inlining for JavaScript

FUNCTIONAL DESCRIPTION: The IFJS compiler is applied to JavaScript code. The compiler generates JavaScript code instrumented with checks to secure code. The compiler takes into account special features of JavaScript such as implicit type coercions and programs that actively try to bypass the inlined enforcement mechanisms. The compiler guarantees that third-party programs cannot (1) access the compiler internal state by randomizing the names of the resources through which it is accessed and (2) change the behaviour of native functions that are used by the enforcement mechanisms inlined in the compiled code.

- Contact: Manuel Serrano
- URL: http://www-sop.inria.fr/indes/ifJS/

4.5. iflowsigs.js

KEYWORDS: Compilers - Monitoring

FUNCTIONAL DESCRIPTION: iflowsigs.js is a JavaScript library designed to inline an information flow monitor into JavaScript code. iflowsigs.js support is able to track information flow even in programs that interact with arbitrary Web APIs.

- Participants: José Fragoso Santos and Tamara Rezk
- Contact: Tamara Rezk
- URL: http://j3fsantos.github.io/PersonalPage/IFMonitor/

4.6. iflowTYPES.js

FUNCTIONAL DESCRIPTION: iflowtypes.js is a JavaScript library designed to type secure information flow in JavaScript. iflowtypes.js has two main modes of operation: fully static and hybrid. In the hybrid mode, the program to be typed is instrumented with runtime assertions that are verified at runtime. By deferring rejection to runtime, the hybrid type system is able to type more programs than fully static mechanisms.

- Contact: Tamara Rezk
- URL: http://j3fsantos.github.io/PersonalPage/TypeSystem/

4.7. Mashic

FUNCTIONAL DESCRIPTION: The Mashic compiler is applied to mashups with untrusted scripts. The compiler generates mashups with sandboxed scripts, secured by the same origin policy of the browsers. The compiler is written in Bigloo.

- Contact: Manuel Serrano
- URL: http://web.ist.utl.pt/~ana.matos/Mashic/mashic.html

4.8. scheme2JS

Scheme to JavaScript

KEYWORD: Compilers

FUNCTIONAL DESCRIPTION: Scm2JS is a Scheme to JavaScript compiler distributed under the GPL license. Even though much effort has been spent on being as close as possible to R5rs, we concentrated mainly on efficiency and interoperability. Usually Scm2JS produces JavaScript code that is comparable (in speed) to hand-written code. In order to achieve this performance, Scm2JS is not completely R5rs compliant. In particular it lacks exact numbers.

Interoperability with existing JavaScript code is ensured by a JavaScript-like dot-notation to access JavaScript objects and by a flexible symbol-resolution implementation.

Scm2JS is used on a daily basis within Hop, where it generates the code which is sent to the clients (webbrowsers).

- Contact: Manuel Serrano
- URL: http://www-sop.inria.fr/indes/scheme2js/

4.9. Hiphop.js

KEYWORDS: Web 2.0 - Synchronous Language

FUNCTIONAL DESCRIPTION: Hiphop.js is a DSL which extends JavaScript with temporal constructions. It makes easier the orchestration of asynchronous Web applications.

- Contact: Colin Vidal
- URL: http://www-sop.inria.fr/members/Colin.Vidal/hiphop/

4.10. Server-Side Protection against Third Party Web Tracking

KEYWORDS: Privacy - Web Application - Web - Architecture - Security by design - Program rewriting techniques

FUNCTIONAL DESCRIPTION: We present a new web application architecture that allows web developers to gain control over certain types of third party content. In the traditional web application architecture, a web application developer has no control over third party content. This allows the exchange of tracking information between the browser and the third party content provider.

To prevent this, our solution is based on the automatic rewriting of the web application in such a way that the third party requests are redirected to a trusted third party server, called the Middle Party Server. It may be either controlled by a trusted party, or by a main site owner and automatically eliminates third-party tracking cookies and other technologies that may be exchanged by the browser and third party server

- Contact: Doliere Some
- URL: http://www-sop.inria.fr/members/Doliere.Some/essos/

4.11. BELL

Browser fingerprinting via Extensions and Login-Leaks

KEYWORDS: Browser Extensions - Security and Privacy in Web Services - Social Networks Security and Privacy

FUNCTIONAL DESCRIPTION: Recent studies show that users can be tracked based on their web browser properties. This software is designed to conduct an experiment on such kinds of user tracking. In this experiment, we demonstrate that a Web user can also be tracked by

- her browser extensions (such as AdBlock, Pinterest, or Ghostery), and

- the websites she has logged in (such as Facebook, Gmail, or Twitter).

In the experiment, we collect user's browser fingerprint, together with the browser extensions installed and a list of websites she has logged in. We only collect anonymous data during the experiment (more details in our Privacy Policy⁰), we will securely store the data on an Inria server, use it only for research purposes and not share it with anyone outside of Inria.

- Contact: Gabor Gulyas
- URL: https://extensions.inrialpes.fr/

4.12. webstats

Webstats

⁰https://extensions.inrialpes.fr/privacy.php

KEYWORDS: Web Usage Mining - Statistic analysis - Security

FUNCTIONAL DESCRIPTION: The goal of this tool is to perform a large-scale monthly crawl of the top Alexa sites, collecting both inline scripts (written by web developers) and remote scripts, and establishing the popularity of remote scripts (such as Google Analytics and jQuery). With this data, we establish whether the collected scripts are actually written in a subset of JavaScript by analyzing the different constructs used in those scripts. Finally, we collect and analyze the HTTP headers of the different sites visited, and provide statistics about the usage of HTTPOnly and Secure cookies, and the Content Security Policy in top sites.

- Contact: Doliere Some
- URL: https://webstats.inria.fr

PHOENIX Project-Team

6. New Software and Platforms

6.1. College

KEYWORDS: Neurosciences - Health - Autism - Mobile application

FUNCTIONAL DESCRIPTION: College+ is an iPad app gathering an assistance module and a training module for school inclusion in of children with autism spectrum disorders and children with intellectual disabilities in mainstream classrooms. The assistance module, used in mainstream classroom, comprises 3 functionalities: - emotion regulation - classroom routines - verbal communication. The training module, used on a daily basis at home or in special education classroom, comprises two functionalities, presented as serious games: - attention training - emotion identification training. All contents of College+ app can be modified, to fit the unique needs of each student.

- Participants: Alexandre Spriet, Charles Consel, Charles Fage, Damien Martin Guillerez and Hélène Sauzéon
- Partners: Université de Bordeaux CNRS IPB
- Contact: Charles Consel
- URL: http://phoenix.inria.fr/research-projects/school

6.2. College +

KEYWORDS: Neurosciences - Health - Autism - Mobile application

FUNCTIONAL DESCRIPTION: College+ is an iPad app gathering an assistance module and a training module for school inclusion in of children with autism spectrum disorders and children with intellectual disabilities in mainstream classrooms. The assistance module, used in mainstream classroom, comprises 3 functionalities: - emotion regulation - classroom routines - verbal communication. The training module, used on a daily basis at home or in special education classroom, comprises two functionalities, presented as serious games: - attention training - emotion identification training. All contents of College+ app can be modified, to fit the unique needs of each student.

- Participants: Alexandre Spriet, Charles Fage, Damien Martin Guillerez and Hélène Sauzéon
- Contact: Charles Consel
- URL: http://phoenix.inria.fr/research-projects/school

6.3. DiaSuite

KEYWORDS: Pervasive computing - Code generation - Specification language

SCIENTIFIC DESCRIPTION: DiaSuite is a suite of tools covering the development life-cycle of a pervasive computing application:

Defining an application area. First, an expert defines a catalog of entities, whether hardware or software, that are specific to a target area. These entities serve as building blocks to develop applications in this area. They are gathered in a taxonomy definition, written in the taxonomy layer of the DiaSpec language.

Designing an application. Given a taxonomy, the architect can design and structure applications. To do so, the DiaSpec language provides an application design layer. This layer is dedicated to an architectural pattern commonly used in the pervasive computing domain. Describing the architecture application allows to further model a pervasive computing system, making explicit its functional decomposition.

Implementing an application. We leverage the taxonomy definition and the architecture description to provide dedicated support to both the entity and the application developers. This support takes the form of a Java programming framework, generated by the DiaGen compiler. The generated programming framework precisely guides the developer with respect to the taxonomy definition and the architecture description. It consists of high-level operations to discover entities and interact with both entities and application components. In doing so, it abstracts away from the underlying distributed technologies, providing further separation of concerns.

Testing an application.DiaGen generates a simulation support to test pervasive computing applications before their actual deployment. An application is simulated in the DiaSim tool, without requiring any code modification. DiaSim provides an editor to define simulation scenarios and a 2D-renderer to monitor the simulated application. Furthermore, simulated and actual entities can be mixed. This hybrid simulation enables an application to migrate incrementally to an actual environment.

Deploying a system. Finally, the system administrator deploys the pervasive computing system. To this end, a distributed systems technology is selected. We have developed a back-end that currently targets the following technologies: Web Services, RMI, SIP and OSGI. This targeting is transparent for the application code. The variety of these target technologies demonstrates that our development approach separates concerns into well-defined layers.

FUNCTIONAL DESCRIPTION: DiaSuite is developed as a research project by the Inria/LaBRI Phoenix research group. The DiaSuite approach covers the development life-cycle of a pervasive computing application. It takes the form of a methodology, supported by (1) a high-level design language and (2) a suite of tools covering the development life-cycle of a pervasive computing application. Specifically, we have developed a design language dedicated to describing pervasive computing systems and a suite of tools providing customized support for each development stage of a pervasive computing system, namely, implementation (e.g., programming support), testing (e.g., unit test, 2D simulator), and deployment (e.g., distribution platforms like SIP and Web Services).

- Participants: Adrien Carteron, Alexandre Spriet, Charles Consel, Milan Kabac, Paul Van Der Walt and Quentin Barlas
- Contact: Charles Consel
- URL: http://phoenix.inria.fr/software/diasuite

6.4. DiaSuiteBOX

KEYWORDS: Dedicated langage - Iot - Orchestration - Toolbox - Development tool suite FUNCTIONAL DESCRIPTION: DiaSuiteBOX proposes an application store that gathers the devices deployed at home. This store is open and available online such as an application store for Smartphone.

- Participants: Adrien Carteron, Amélie Marzin, Bertran Benjamin, Bruneau Julien, Consel Charles, Damien Cassou, Damien Martin Guillerez, Emilie Balland, Eugène Volanschi, Hélène Sauzéon, Joan Rieu, Julien Durand, Ludovic Fornasari, Milan Kabac, Quentin Barlas and Quentin Enard
- Contact: Charles Consel
- URL: https://domassist.bordeaux.inria.fr/

6.5. DomAssist

KEYWORDS: Health - Mobile application - Persons attendant - Home care

FUNCTIONAL DESCRIPTION: 3 mobile applications for assistive living : - DiAndroid : Interface for the main tablet with the DiaSuiteBox applications including those for the daily activities, the meetings scheduling, etc. and for home and personal safety - Accueil : home screen restraining the use of a secondary tablet and offering communications and social activities applications with simplified communication means (ie. eMail), collaborative games, etc. - eMail : mail client made for older people

Other Vera gateway tools : - Controlling connected objects to the Vera home automation gateway from Android - Vera push plugin to domassist cloud: vera plugin to communicate sensor information associated with the gateway directly to DomAssist servers - Vera HeartBeat Plugin: Regular sending of a frame to know that the gateway is still online

- Participants: Alexandre Spriet, Quentin Barlas, Charles Consel, Hélène Sauzéon and Julien Durand
- Partners: Université de Bordeaux CNRS IPB
- Contact: Charles Consel
- URL: http://phoenix.inria.fr/research-projects/homeassist

RMOD Project-Team

6. New Software and Platforms

6.1. Moose

Moose: Software and Data Analysis Platform

KEYWORDS: Software engineering - Meta model - Software visualisation

FUNCTIONAL DESCRIPTION: Moose is an extensive platform for software and data analysis. It offers multiple services ranging from importing and parsing data, to modeling, to measuring, querying, mining, and to building interactive and visual analysis tools. The development of Moose has been evaluated to 200 man/year.

Mots-cles : MetaModeling, Program Visualization, Software metrics, Code Duplication, Software analyses, Parsers

- Participants: Anne Etien, Nicolas Anquetil, Olivier Auverlot and Stéphane Ducasse
- Partners: Université de Berne Sensus Synectique Pleiad USI Vrije Universiteit Brussel
- Contact: Stéphane Ducasse
- URL: http://www.moosetechnology.org

6.2. Pharo

KEYWORDS: Live programmation objet - Reflective system - Web Application

FUNCTIONAL DESCRIPTION: Pharo is a pure object reflective and dynamic language inspired by Smalltalk. In addition, Pharo comes with a full advanced programming environment developed under the MIT License. It provides a platform for innovative development both in industry and research. By providing a stable and small core system, excellent developer tools, and maintained releases, Pharo's goal is to be a platform to build and deploy mission critical applications, while at the same time continue to evolve. Pharo 60 got 100 contributors world-wide. It is used by around 30 universities, 15 research groups and around 40 companies.

RELEASE FUNCTIONAL DESCRIPTION: Inspector/Playground/Spotter are new moldable development tools for inspecting, coding and searching objects. Slots model instance variables as first class entities and enable meta-programming on this level. ShoreLine reporter introduces a way to report system errors and collect statistics, that we will use for future improvements Dark theme.

- Participants: Christophe Demarey, Clement Bera, Damien Pollet, Esteban Lorenzano, Marcus Denker and Stéphane Ducasse
- Partners: Université de Berne Cadence Inceptive Netstyle Feenk ObjectProfile GemstoneSystems - Greyc Université de Caen - Basse-Normandie - BetaNine - Yesplan - RMod - Pleiad - Synectique - Sensus - Université de Bretagne Occidentale - École des Mines de Douai - Reveal
- Contact: Marcus Denker
- URL: http://www.pharo.org

6.3. Pillar

KEYWORDS: HTML - LaTeX - HTML5

FUNCTIONAL DESCRIPTION: Pillar is a markup syntax and associated tools to write and generate documentation and books. Pillar is currently used to write several books and other documentation. Two platforms have already been created on top of Pillar: PillarHub and Marina.

- Contact: Stéphane Ducasse
- URL: https://github.com/Pillar-markup/pillar

TACOMA Team

5. New Software and Platforms

5.1. THEGAME

SCIENTIFIC DESCRIPTION: Context-aware applications have to sense the environment in order to adapt themselves and provide with contextual services. This is the case of Smart Homes equipped with sensors and augmented appliances. However, sensors can be numerous, heterogeneous and unreliable. Thus the data fusion is complex and requires a solid theory to handle those problems. The aim of the data fusion, in our case, is to compute small pieces of context we call context attributes. Those context attributes are diverse and could be for example the presence in a room, the number of people in a room or even that someone may be sleeping in a room. For this purpose, we developed an implementation of the belief functions theory (BFT). THE GAME (THeory of Evidence in a lanGuage Adapted for Many Embedded systems) is made of a set of C-Libraries. It provides the basics of belief functions theory, computations are optimized for an embedded environment (binary representation of sets, conditional compilation and diverse algorithmic optimizations).

THE GAME is published under apache licence (https://github.com/bpietropaoli/THEGAME/). It is maintained and experimented by Aurélien Richez within a sensor network platform developed by TACOMA since June 2013.

FUNCTIONAL DESCRIPTION: THEGAME is a set of software services for detecting different types of situation in a building (presence in a room, activity level, etc.) based on a set of raw data sourced from all sorts of sensors. Written in C or Java, it can be integrated in an embedded computer: tablet, smartphone, box, etc., and can be connected to different sensor networks. It can be used to implement context-aware services: for example, to alert the user if s/he forgets to close a window when leaving the building, or to turn off the heating in an empty room, etc.

- Participants: Aurélien Richez and Bastien Pietropaoli
- Contact: Frédéric Weis
- URL: https://github.com/bpietropaoli/THEGAME/

5.2. Platform Pervasive_RFID

KEYWORDS: Composite objects - RFID

- Participants: Paul Couderc and Anthony Blair (Univ. Rennes 1)
- Partner: Univ. Rennes 1 (IETR)
- Contact: Paul Couderc

SCIENTIFIC DESCRIPTION

The RFID experiment testbed has been designed and deployed in collaboration with IETR (see Figure 2). This system allows both interactive testing as well as long running experiments of RFID reading protocols. It comprises a software platform allowing fine control over all dynamic aspects influencing RFID readings: movements for target and antenna, RFID reader configuration, and smart antenna configuration (diversity and power control).

5.3. Metamorphic Housing platform and Software - On-demand room

KEYWORDS: Smart Home - Metamorphic House - Sharing

- Participants: Ghislain Nouvel (Univ. Rennes 1), Guillermo Andrade Barroso and Michele Dominici
- Partner: Univ. Rennes 1
- Partner: Delta Dore Néotoa⁰
- Contact: Michele Dominici and Frédéric Weis

⁰http://www.neotoa.fr/



Figure 2. RFID testbed

SCIENTIFIC DESCRIPTION- SOFTWARE

As part of the experimentation of the On-demand room, we have developed a software system that will be used to manage the room and provide functionalities to end users and building managers (access control, electrical and time consumption monitoring and report, room state display...). The software has been deployed in the building that hosts the experimentation. This software is co-developed by Michele Dominici (Univ. Rennes 1), Guillermo Andrade (SED Inria) and Ghislain Nouvel (MobBI platform ⁰). Contributions are provided by members of the Diverse project-team. Intellectual protection is expected to be applied on such software.

SCIENTIFIC DESCRIPTION- PLATFORM

We realized a prototype of the on-demand room as an immersive interactive virtual-reality application, leveraging the Immersia platform (see https://raweb.inria.fr/rapportsactivite/RA2015/tacoma/uid29.html), with real domestic appliances connected to Immersa. In 2016-2017, the experimentation of the On-demand room is organized in the following steps: modification of the original building to create a common, On-demand room between two apartments; deployment of the computer and hardware and software that we are developing; rental of the apartments to two households, for an estimated duration of one year. The building that hosted the experimentation is showed in Figure 3 . During the rental of the apartments, data has been collected and stored about the use of the room by households. Data included time of occupation, mode (private or shared), consumptions, errors etc. The On-demand room thus constitutes an experimentation platform, where real people live and produce data that can be analyzed for statistical purposes. Produced data could also be used in combination with interviews of the occupants to improve the functionalities of the On-demand room, evaluate acceptance and appropriation.

5.4. ISO/IEC 15118-2 Open source Implementation

KEYWORDS: Smart Grid - Intelligent Transport System

- Partner: IMT Atlantique
- Participants: Guillaume Le Gall
- Contact: Jean-Marie Bonnin

SCIENTIFIC DESCRIPTION

⁰https://mobbi.univ-rennes1.fr/



Figure 3. On-demand room real experimentation

The ISO/IEC 15118 standard, named "Road vehicles – Vehicle-to-Grid Communication Interface", defines how an electric vehicle and a charging station should communicate. It enables the Smart Charging of electric vehicles by allowing them to plan their charging sessions. As we want to be able to manage the charge of electric vehicles in our micro Smart Grid systems, we decided to implement the protocol defined by this standard. The goal is also to participate actively in the design of the new version of this protocol. During a charging session the charging station provides the vehicle with the status of the electric power grid. The vehicle is then able to plan its sharing session accordingly. It sends back its charge plan to the charging station, so that the Smart Grid is aware of it. The protocol also provides security and authentication features.

This software platform was implemented onto small PCs, and was used to control the charge in a small and portable demonstration platform, to demonstrate how it is possible to interconnect this high level decision and communication software with low level components, such as a Battery Management System (BMS), and a battery charger. In 2016, in the context of the Greenfeed project our software has been demonstrated to control the charge of the electric vehicle during the final demonstration of the project. The integration work has been done in collaboration with VeDeCom⁰.

⁰http://www.vedecom.fr/

AGORA Team

6. New Software and Platforms

6.1. TAPASCologne

Travel and Activity PAtterns Simulation Cologne

KEYWORDS: Mobility - Traces

FUNCTIONAL DESCRIPTION: TAPASCologne is an initiative by the Institute of Transportation Systems at the German Aerospace Center (ITS-DLR), aimed at reproducing, with the highest level of realism possible, car traffic in the greater urban area of the city of Cologne, in Germany.

To that end, different state-of-art data sources and simulation tools are brought together, so to cover all of the specific aspects required for a proper characterization of vehicular traffic:

The street layout of the Cologne urban area is obtained from the OpenStreetMap (OSM) database, The microscopic mobility of vehicles is simulated with the Simulation of Urban Mobility (SUMO) software, The traffic demand information on the macroscopic traffic flows across the Cologne urban area (i.e., the O/D matrix) is derived through the Travel and Activity PAtterns Simulation (TAPAS) methodology, The traffic assignment of the vehicular flows described by the TAPASCologne O/D matrix over the road topology is performed by means of Gawron's dynamic user assignment algorithm.

- Participants: Marco Fiore and Razvan Stanica
- Contact: Marco Fiore
- URL: http://kolntrace.project.citi-lab.fr/#download

6.2. Sense in the City

KEYWORDS: Sensors - Sensors network - Wireless Sensor Networks

FUNCTIONAL DESCRIPTION: Sense in the city is a lightweight experimentation platform for wireless sensor networks in development. The main objective of this platform is to be easily transferable and deployable on the field. It allows a simplified deployment of the code running on the sensors and the collection of logs generated by the instrumentation of the code on a centralized database. In the early stage of the platform, the sensors are powered by small PCs, e.g. Raspberry Pis, but we are investigating the integration of energy harvesting capabilities such as solar panels.

- Participants: Hervé Rivano and Khaled Boussetta
- Contact: Khaled Boussetta

6.3. PrivaMovApp

KEYWORD: Crowd-sensing

FUNCTIONAL DESCRIPTION: Agora is leading the development of an Android application for user data collection purposes. The application is based on the Funf framework, and is currently available on Google Play.

- Participants: Stéphane D'alu, Hervé Rivano and Razvan Stanica
- Contact: Razvan Stanica

6.4. WSNet

KEYWORD: Network simulator

FUNCTIONAL DESCRIPTION: WSNet is a modular event-driven simulator targeted to Wireless Sensor Networks. Its main goals are to offer scalabiliy, extensibility and modularity for the integration of new protocols/hardware models and a precise radio medium simulation. We still hope to find the proper resource to make WSNet evolve into a wireless capillary network simulator suitable for conducting simulations at the urban scale.

- Participants: Rodrigue Domga Komguem and Fabrice Valois
- Partner: CEA-LETI
- Contact: Guillaume Chelius
- URL: https://gforge.inria.fr/projects/wsnet-3/

COATI Project-Team

6. New Software and Platforms

6.1. BigGraphs

KEYWORDS: Graph algorithmics - Distributed computing - Java - Graph processing

FUNCTIONAL DESCRIPTION: The objective of BigGraphs is to provide a distributed platform for very large graphs processing. A typical data set for testing purpose is a sample of the Twitter graph : 240GB on disk, 398M vertices, 23G edges, average degree of 58 and max degree of 24635412.

We started the project in 2014 with the evaluation of existing middlewares (GraphX / Spark and Giraph / Hadoop). After having tested some useful algorithms (written according to the BSP model) we decided to develop our own platform.

This platform is based on the existing BIGGRPH library and we are now in the phasis where we focus on the quality and the improvement of the code. In particular we have designed strong test suites and some non trivial bugs have been fixed. We also have solved problems of scalability, in particular concerning the communication layer with billions of messages exchanged between BSP steps. We also have implemented specific data structures for BSP and support for distributed debugging. This comes along with the implementation of algorithms such as BFS or strongly connected components that are run on the NEF cluster.

In 2017 we have developed a multi-threaded shared-memory parallel version of the Bulk Synchronous Parallel framework. This new version uses advanced synchronization mechanisms and strategies to minimize the congestion of multiple threads working on the same graph. Using the NEF cluster (Inria Sophia Antipolis), this parallel version exhibits speed-ups up to 6.5 using 8 nodes (16 cores each) when computing a BFS on the 23 G edges Twitter graph sample.

- Participants: Luc Hogie, Michel Syska and Nicolas Chleq
- Partner: CNRS
- Contact: Luc Hogie

6.2. GRPH

The high performance graph library for Java

KEYWORDS: Graph - Graph algorithmics - Java

FUNCTIONAL DESCRIPTION: Grph is an open-source Java library for the manipulation of graphs. Its design objectives are to make it portable, simple to use/extend, computationally/memory efficient, and, according to its initial motivation: useful in the context of graph experimentation and network simulation. Grph also has the particularity to come with tools like an evolutionary computation engine, a bridge to linear programming solvers, a framework for distributed computing, etc.

Grph offers a very general model of graphs. Unlike other graph libraries which impose the user to first decide if he wants to deal with directed, undirected, hyper (or not) graphs, the model offered by Grph is unified in a general class that supports mixed graphs made of undirected and directed simple and hyper edges. Grph achieves great efficiency through the use of multiple code optimization techniques such as multi-core parallelism, caching, adequate data structures, use of primitive objects, exploitation of low-level processor caches, on-the-fly compilation of specific C/C++ code, etc. Grph attempts to access the Internet in order to check if a new version is available and to report who is using it (login name and hostname). This has no impact whatsoever on performance and security.

- Participants: Aurélien Lancin, David Coudert, Issam Tahiri, Luc Hogie and Nathann Cohen
- Contact: Luc Hogie
- URL: http://www.i3s.unice.fr/~hogie/grph/

6.3. Sage

SageMath

SCIENTIFIC DESCRIPTION: SageMath is a free open-source mathematics software system. It builds on top of many existing open-source packages: NumPy, SciPy, matplotlib, Sympy, Maxima, GAP, FLINT, R and many more. Access their combined power through a common, Python-based language or directly via interfaces or wrappers.

FUNCTIONAL DESCRIPTION: SageMath is an open-source mathematics software initially created by William Stein (Professor of mathematics at Washington University). We contribute the addition of new graph algorithms along with their documentations and the improvement of underlying data structures.

- Contact: David Coudert
- URL: http://www.sagemath.org/
DANTE Project-Team

6. New Software and Platforms

6.1. GraSP

Graph Signal Processing KEYWORDS: Signal processing - Graph visualization - Graph - LaTeX - Matlab - GNU Octave FUNCTIONAL DESCRIPTION: Matlab / GNU Octave toolbox to manipulate and visualize signals on graphs. LaTeX package to draw signals.

• Contact: Benjamin Girault

6.2. IoT-LAB aggregation-tools

KEYWORD: Internet of things

FUNCTIONAL DESCRIPTION: IoT-LAB aggregation-tools allow aggregating data results from many nodes at a time. It connects to several tcp connections and handle the received data.

- Participant: Gaetan Harter
- Contact: Eric Fleury
- URL: https://github.com/iot-lab/aggregation-tools

6.3. IoT-LAB cli-tools

KEYWORD: Internet of things

FUNCTIONAL DESCRIPTION: IoT-LAB cli-tools provide a basic set of operations for managing IoT-LAB experiments from the command-line.

- Participants: Frédéric Saint-Marcel and Gaetan Harter
- Contact: Eric Fleury
- URL: https://github.com/iot-lab/cli-tools

6.4. IoT-LAB gateway

KEYWORD: Internet of things

FUNCTIONAL DESCRIPTION: IoT-LAB software embedded on a IoT-LAB gateway node new generation provides the local management of the experiment on that node. It is a software bridge between the IoT-LAB server, the user open node and the control node.

- Contact: Frédéric Saint-Marcel
- URL: https://github.com/iot-lab/iot-lab-gateway

6.5. IoT-LAB robots

KEYWORDS: Internet of things - Robotics

FUNCTIONAL DESCRIPTION: IoT-LAB robots is an embedded robot controler on a Turtlebot2 providing the IoT-LAB node mobility functionnality

- Partner: Université de Strasbourg
- Contact: Julien Vandaële
- URL: https://github.com/iot-lab/

6.6. Queueing Systems

FUNCTIONAL DESCRIPTION: This tool aims at providing a simple web interface to promote the use of our proposed solutions to numerically solve classical queueing systems.

- Participants: Alexandre Brandwajn and Thomas Begin
- Contact: Thomas Begin
- URL: http://queueing-systems.ens-lyon.fr/

6.7. WSNet

KEYWORD: Network simulator

FUNCTIONAL DESCRIPTION: WSNet is a modular event-driven simulator targeted to Wireless Sensor Networks. Its main goals are to offer scalabiliy, extensibility and modularity for the integration of new protocols/hardware models and a precise radio medium simulation. We still hope to find the proper resource to make WSNet evolve into a wireless capillary network simulator suitable for conducting simulations at the urban scale.

- Participants: Rodrigue Domga Komguem and Fabrice Valois
- Partner: CEA-LETI
- Contact: Guillaume Chelius
- URL: https://gforge.inria.fr/projects/wsnet-3/

6.8. Platforms

6.8.1. FIT IoT-LAB

FUNCTIONAL DESCRIPTION

IoT-LAB provides full control of network IoT nodes and direct access to the gateways to which nodes are connected, allowing researchers to monitor nodes energy consumption and network-related metrics, e.g. end-to-end delay, throughput or overhead. The facility offers quick experiments deployment, along with easy evaluation, results collection and analysis. Defining complementary testbeds with different node types, topologies and environments allows for coverage of a wide range of real-life use-cases.

- Partner: FIT is one of 52 winning projects from the first wave of the French Ministry of Higher Education and Research (Équipement d'Excellence (Equipex) research grant programme. The FIT consortium is composed of: Université Pierre et Marie Curie (UPMC), Inria, Université de Strasbourg, Institut Mines Télécom and CNRS
- Contact: Éric Fleury
- URL: https://www.iot-lab.info/

74

DIANA Project-Team

5. New Software and Platforms

5.1. ACQUAmobile

KEYWORDS: Android - Internet access - Performance measure - Quality of Experience

FUNCTIONAL DESCRIPTION: ACQUA is an Application for predicting QUality of Experience (QoE) at Internet Access. It is developed by the Diana team at Inria Sophia Antipolis – Méditerranée and is supported by Inria under the ADT ACQUA grant. The scientific project around ACQUA is supported by Inria Project Lab BetterNet and the French National Project ANR BottleNet. ACQUA presents a new way for the evaluation of the performance of Internet access. Starting from network-level measurements as the ones we often do today (bandwidth, delay, loss rates, jitter, etc), ACQUA targets the estimated Quality of Experience (QoE) related to the different applications of interest to the user without the need to run them (e.g., estimated Skype quality, estimated video streaming quality).

An application in ACQUA is a function, or a model, that links the network-level and device-level measurements to the expected Quality of Experience. Supervised machine learning techniques are used to establish such link between measurements both at the network level and the device level, and estimations of the Quality of Experience for different Internet applications. The required data for such learning can be obtained either by controlled experiments as we did in [19] on YouTube Quality of Experience, or by soliciting the crowd (i.e. crowdsourcing) for combinations (i.e. tuples) of measurements and corresponding application-level Quality of Experience. Our current work is concentrating on using the ACQUA principle in the estimation and prediction of the Quality of Experience for main user's applications. We refer to the web site of the project for further details.

The ACQUA Android application is supposed to be on one hand the reference application for QoE forecasting and troubleshooting for end users at their Internet access, and on the other hand, the feedback channel that allows end users to report to us (if they are willing) on their experience together with the corresponding network measurements so as to help us calibrating better and more realistic models. For this calibration, we are currently performing extensive, efficient and automatic measurements in the laboratory, we will count on end users to help us completing this dataset with further applications and more realistic network and user conditions.

ACQUA is mainly meant for end users, but it is also of interest to (mobile) network operators and to content providers to estimate the QoE of their customers and their networks without each time having to run expensive application-level traffic and to involve real users.

- Authors: Thierry Spetebroot and Chadi Barakat
- Contact: Chadi Barakat
- URL: http://project.inria.fr/acqua/

5.2. ElectroSmart

KEYWORDS: Crowd-sourcing - UMTS - GSM - Bluetooth - Wi-Fi - 4G - 3G - 2G - Electromagnetic waves - Android - LTE

FUNCTIONAL DESCRIPTION: The Internet and new devices such as smartphones have fundamentally changed the way people communicate, but this technological revolution comes at the price of a higher exposition of the general population to microwave electromagnetic fields (EMF). This exposition is a concern for health agencies and epidemiologists who want to understand the impact of such an exposition on health, for the general public who wants a higher transparency on its exposition and the health hazard it might represent, but also for cellular operators and regulation authorities who want to improve the cellular coverage while limiting the exposition, and for computer scientists who want to better understand the network connectivity in order to optimize communication protocols. Despite the fundamental importance to understand the exposition of the general public to EMF, it is poorly understood because of the formidable difficulty to measure, model, and analyze this exposition.

The goal of the ElectroSmart project is to develop the instrument, methods, and models to compute the exposition of the general public to microwave electromagnetic fields used by wireless protocols and infrastructures such as Wi-Fi, Bluetooth, or cellular. Using a pluri-disciplinary approach combining crowd-based measurements, in-lab experiments, and modeling using sparse and noisy data, we address challenges such as designing and implementing a measuring instrument leveraging on crowd-based measurements from mobile devices such as smartphones, modeling the exposition of the general public to EMF to compute the most accurate estimation of the exposition, and analyzing the evolution of the exposition to EMF with time. This technological breakthrough will have scientific, technical, and societal applications, notably on public health politics, by providing the scientific community and potential users with a unique measuring instrument, methods, and models to exploit the invaluable data gathered by the instrument.

This project is supported by the UCN@Sophia Labex in 2016/2017/2018 (funding the engineer Mondi Ravi), by an Inria ADT (funding the engineer Abdelhakim Akodadi) 2017/2018, by and Inria ATT (funding the business developer David Migliacci) in 2017/2018, and by the academy 1 of UCAJedi (funding a Ph.D. student Yanis Boussad) 2017/2020.

In August 2016, we released the first stable public release of ElectroSmart. On the 22th December 2017 we have 35 836 downloads in Google Play, an average score of 4,6/5, 10 538 active users, 350 millions measured signals.

We are in a process of creating a startup to commercialize the exposition maps we can build with the data we are collecting.

- Participants: Arnaud Legout, Hackob Melconian, Inderjeet Singh and Mondi Ravi
- Contact: Arnaud Legout
- URL: https://es.inria.fr/home/index?path_prefix=en

5.3. OpenLISP

KEYWORDS: LISP - Routing - Control-plane

FUNCTIONAL DESCRIPTION: Among many options tackling the scalability issues of the current Internet routing architecture, the Locator/Identifier Separation Protocol (LISP) appears as a viable solution. LISP improves a network's scalability, flexibility, and traffic engineering, enabling mobility with limited overhead. As for any new technology, implementation and deployment are essential to gather and master the real benefits that it provides. We propose a complete open source implementation of the LISP control plane. Our implementation is deployed in the worldwide LISP Beta Network and the French LISP-Lab testbed, and includes the key standardized control plane features. Our control plane software is the companion of the existing OpenLISP dataplane implementation, allowing the deployment of a fully functional open source LISP network compatible with any implementation respecting the standards.

- Contact: Damien Saucez
- URL: http://www.openlisp.org/downloads

5.4. nepi-ng

KEYWORDS: Wireless network - Experimentation

FUNCTIONAL DESCRIPTION: In the specific context of R2lab, we have created a tool suite for orchestrating network experiments, that for historical reasons we refer to collectively as nepi-ng, for NEPI new generation. An umbrella website is available at https://nepi-ng.inria.fr/.

At this point, nepi-ng has a much smaller scope than its NEPI ancestor used to have, in that it only supports remote control of network experiments over ssh. As a matter of fact, in practice, this is the only access mechanism that we need to have for running experiments on both R2lab, and PlanetLab Europe.

The design of nepi-ng of course is modular, so that it will be perfectly possible to add other control mechanisms to this core if and when it becomes necessary.

nepi-ng is currently made of 2 separate python libraries :

- asynciojobs:
 - URL: http://asynciojobs.readthedocs.io/en/latest/
 - Version: asynciojobs v0.5.4
 - Keywords: networking experimentation, orchestration
 - License: CC BY-SA 4.0
 - Type of human computer interaction: python library
 - OS/Middleware: Linux
 - Required library or software: python-3.5 / asyncio
 - Programming language: python3
- apssh:
 - URL: http://apssh.readthedocs.io/en/latest/
 - Version: apssh v0.7.1
 - Keywords: networking experimentation, orchestration
 - License: CC BY-SA 4.0
 - Type of human computer interaction: python library
 - OS/Middleware: Linux
 - Required library or software: python-3.5 / asyncio
 - Programming language: python3
- Contact: Thierry Parmentelat
- URL: http://nepi-ng.inria.fr

5.5. Platforms

5.5.1. Reproducible research laboratory (R²lab)

Scientific evaluation of network protocols requires for experiments to be reproducible before they can be deemed valid. This is particularly difficult to obtain in the wireless networking area, where characteristics of wireless channels are known to be variable, unpredictable and hardly controllable.

The R^2 lab wireless testbed is built around an isolated and anechoic chamber, featuring RF absorbers preventing radio waves reflections and a Faraday cage blocking external interferences. This lab, named R^2 lab, represents an ideal environment for experiments reproducibility.

It represents a perfect facility for making wireless experiments reproducible. It has been operated for 3 years now, in the context of the FIT Equipment of Excellence project, and as such, it is now federated with the other testbeds that are part of the FIT initiative. This testbed is for the long-haul, and is scheduled to remain operational until at least 2020.

During 2017, our focus regarding R^2 lab has been set on deploying more, and more diverse USRPs (Universal Software Radio Peripherals). The chamber now offers more Ettus-based devices, as well as a couple of limesdr devices, and a couple of E3372 LTE dongles. An apple iphone will soon be available, in addition to the first nexus phone. All these additions aim at widening even further the spectrum of experiments that the testbed can support.

For more details see http://r2lab.inria.fr.

DIONYSOS Project-Team

6. New Software and Platforms

6.1. IPv6 Test Toolkit

FUNCTIONAL DESCRIPTION: These test suites are developed using the TTCN-3 environment.

The packages contains the full Abstract Test Suites written in TTCN-3 and the source files for building the codecs and adapters with the help of T3DevKit.

- Participants: Annie Floch, Anthony Baire, Ariel Sabiguero, Bruno Deniaud, César Viho and Frédéric Roudaut
- Contact: César Viho

6.2. Passive Test Tool

- Participants: Anthony Baire and César Viho
- Contact: Anthony Baire

6.3. T3DevKit

KEYWORDS: IPv6 - Conformance testing - TTCN-3

SCIENTIFIC DESCRIPTION: We have built a toolkit for easing executing tests written in the standardized TTCN-3 test specification language. This toolkit is made of a C++ library together with a highly customizable CoDec generator that allows fast development of external components (that are required to execute a test suite) such as CoDec (for message Coding/Decoding), System and Platform Adapters. It also provides a framework for representing and manipulating TTCN-3 events so as to ease the production of test reports. The toolkit addresses issues that are not yet covered by ETSI standards while being fully compatible with the existing standard interfaces: TRI (Test Runtime Interfaces) and TCI (Test Control Interfaces), it has been tested with four TTCN-3 environments (IBM, Elvior, Danet and Go4IT) and on three different platforms (Linux, Windows and Cygwin).

FUNCTIONAL DESCRIPTION: T3DevKit is a free open source toolkit to ease the development of test suites in the TTCN-3 environment. It provides:

a CoDec generator (t3cdgen) that automates the development process of the CoDec needed for coding TTCN-3 values into physically transmittable messages and decoding incoming messages a library (t3devlib) that provides an object oriented framework to manipulate TTCN-3 entities (values, ports, timers, external functions...) an implementation of the TRI and TCI standard interfaces default implementations for the system adapter (SA), platform adapter (PA), test management (TM), test logging (TL) and component handling (CH) modules default codecs build scripts for the generation of executable test suites, these are tool-independent and facilitate the distribution of test suite sources

- Participants: Annie Floch, Anthony Baire, Ariel Sabiguero, César Viho and Frédéric Roudaut
- Contact: Federico Sismondi

6.4. ttproto

Testing Tool Prototype

KEYWORDS: Interoperability - Conformance testing - TTCN-3

FUNCTIONAL DESCRIPTION: ttproto is an experimental tool for implementing testing tools, for conformance and interoperability testing.

It was first implemented to explore new features and concepts for the TTCN-3 standard, but we also used it to implement a passive interoperability test suite we provided for the CoAP interoperability event held in Paris in March 2012.

This tool is implemented in python3 and its design was influenced mainly by TTCN-3 (abstract model, templates, snapshots, behaviour trees, communication ports, logging) and by Scapy (syntax, flexibility, customisability)

Its purpose is to facilitate rapid prototyping rather than experimentations (rather than production use). We choosed to maximise its modularity and readability rather than performances and real-time considerations.

Now you should have a look at the Features page: https://www.irisa.fr/tipi/wiki/doku.php/ testing_tool_prototype:features

- Contact: Federico Sismondi
- URL: https://www.irisa.fr/tipi/wiki/doku.php/testing_tool_prototype

6.5. CoAP Testing Tool

KEYWORDS: Test - Interoperability - Conformance testing - Plugtests

FUNCTIONAL DESCRIPTION: The software helps developers of the CoAP protocol assessing if their implementations (either CoAP clients or CoAP servers) are conformant to protocol specifications, and interoperable with other implementations. It encompasses:

- Coordination of CoAP interoperability tests
- Analysis of CoAP traces & issuing verdicts
- Automation of open source CoAP implementations for based reference interop testing
- Authors: Federico Sismondi and César Viho
- Contact: Federico Sismondi

6.6. ioppytest

Interoperability testing

KEYWORDS: Interoperability - Conformance testing - CoAP - 6LoWPAN - OneM2M

FUNCTIONAL DESCRIPTION: The software is a framework for developing interoperability tests. The interoperability tests help developers of network protocol assessing if their implementations are conformant to protocol specifications, and interoperable with other implementations.

The software already integrates interoperability tests for CoAP, OneM2M and 6LoWPAN The framework provides the following features to the users:

- Coordination of the interoperability tests (enabling remote testing)
- VPN-like connectivity between users' implementations (enabling remote testing)
- Analysis of exchanged network traces & issuing verdicts
- Automation of open source implementations for based reference interop testing

This framework is the evolution of the CoAP Testing Tool (https://bil.inria.fr/fr/software/view/2937/tab)

- Contact: Federico Sismondi
- URL: https://gitlab.f-interop.eu/f-interop-contributors/ioppytest

6.7. AdaComp

Participants: Corentin Hardy, Bruno Sericola

Our recent works, in collaboration with Technicolor, on deep learning and distributed learning led us to study a kind of data parallelism called the Parameter Server model. This model consists in sharing the learning of a deep neural network between many devices (called the workers) via a centralized Parameter Server (PS). We deployed a platform which allow us to experiment different state-of-the-art algorithms based on the PS model. The platform is composed of a unique powerful machine where many Linux containers (LXC) are running. Each LXC executes a Tensorflow session and can be a worker or a PS. The first experimentations were used to validate the correct functioning of the platform, to better understand its limitations and to determine what can be measured in an unbiased way. Others experimentations helped us to understand the role of different parameters of the overall model, mainly those related to the distribution on user-devices, and their impact on the learning (accuracy of the model, number of iterations to learn the model). During these experimentations, we noted that the main bottleneck is the ingress traffic of PS during the learning phase. To reduce this ingress traffic, we chose to compress the messages sent by the workers to the PS. We proposed in [43] a method to reduce up to 2 orders of magnitude this ingress traffic, keeping a good accuracy on the learned model. This new method, called AdaComp, is available in github (https://github.com/Hardy-c/AdaComp).

6.8. DNN-withRNL

Participants: Corentin Hardy, Gerardo Rubino, Bruno Sericola

The extension of the AdaComp method, presented in 6.7, to Random Neural Networks started with the introduction of Random Neural Layers, see [65]. Concerning the associated software, see https://github.com/Hardy-c/DNN-with-RNL.

DYOGENE Project-Team

6. New Software and Platforms

6.1. CloNES

CLOsed queueing Networks Exact Sampling FUNCTIONAL DESCRIPTION: Clones is a Matlab toolbox for exact sampling of closed queueing networks.

- Participant: Christelle Rovetta
- Contact: Christelle Rovetta
- URL: http://www.di.ens.fr/~rovetta/Clones/index.html

6.2. Platforms

6.2.1. CapRadio

Cellular network dimensioning toolbox *CapRadio* is being developed by Orange in a long-term collaboration between TREC/DYOGENE represented by B. Blaszczyszyn, and Orange Labs, represented by M. K. Karray. This year it has been enriched by the results of the contract titled "Scheduling effect on the distribution of QoS over cells in 4G wireless cellular networks"; cf 8.1.1.

EVA Project-Team

6. New Software and Platforms

6.1. OpenWSN

KEYWORDS: Internet of things - 6TiSCH - 6LoWPAN - CoAP

FUNCTIONAL DESCRIPTION: OpenWSN is an open-source implementation of a fully standards-based protocol stack for the Internet of Things. It has become the de-facto implementation of the IEEE802.15.4e TSCH standard, has a vibrant community of academic and industrial users, and is the reference implementation of the work we do in the IETF 6TiSCH standardization working group.

- Partner: University of California Berkeley
- Contact: Thomas Watteyne
- URL: http://www.openwsn.org/

6.2. 6TiSCH Simulator

High-level simulator of a 6TiSCH network

KEYWORDS: Network simulator - 6TiSCH

FUNCTIONAL DESCRIPTION: The simulator is written in Python. While it doesn't provide a cycle-accurate emulation, it does implement the functional behavior of a node running the full 6TiSCH protocol stack. This includes RPL, 6LoWPAN, CoAP and 6P. The implementation work tracks the progress of the standardization process at the IETF.

• Contact: Malisa Vucinic

6.3. Argus

KEYWORDS: Cloud - Low-Power WIreless - Sniffer FUNCTIONAL DESCRIPTION: There are three piece to the Argus:

The Argus Probe is the program which attaches to your low-power wireless sniffer and forwards its traffic to the Argus Broker.

The Argus Broker sits somewhere in the cloud. Based on MQTT, it connect Argus Probes with Argus Clients based on a pub-sub architecture.

Several Argus Clients can the started at the same time. It is a program which subscribes to the Argus Broker and displays the frames in Wireshark.

• Contact: Rémy Leone

6.4. SolSystem

Sensor Object Library System

KEYWORDS: Low-Power WIreless - Back-End System - SmartMesh IP

FUNCTIONAL DESCRIPTION: The source code is composed of the definition of the SOL structure (https://github.com/realms-team/sol), the code that runs on the manager (https://github.com/realms-team/solmanager, written in Python) and the code that runs on the server receiving the data (https://github.com/realms-team/solserver, written in Python)

- Contact: Keoma Brun-Laguna
- URL: http://www.solsystem.io/

6.5. 6TiSCH Wireshark Dissector

KEYWORDS: 6TiSCH - Wireshark

FUNCTIONAL DESCRIPTION: Implementation on the dissectors is done through an open-source repository, stable code is regularly contributed back to the main Wireshark code base.

• Contact: Jonathan Munoz

6.6. F-Interop

Remote Conformance and Interoperability Tests for the Internet of Thing KEYWORDS: Interoperability - Iot - Conformance testing - Standardization

- Partners: UPMC IMEC ETSI EANTC Mandat International Digital Catapult University of Luxembourg Device Gateway
- Contact: Rémy Leone

6.7. Mercator

KEYWORDS: Deployment - Low-Power WIreless - Testbeds - Connectivity FUNCTIONAL DESCRIPTION: The firmware is written as part of the OpenWSN project. Scripts and analysis tools are written in Python.

• Contact: Keoma Brun-Laguna

6.8. Platforms

6.8.1. SolSystem

In collaboration with University College London and IBM, we have designed a cloud-based low-power network management solution called SolSystem. It serves as a "control tower" for the networks we deploy, allowing us to manage both the network and data produced by those networks. It is architected following the micro-service principle, and we are in the process of switching all of our deployments to that interface. Fig. 1 gives an example of the visualization the SolSystem web interface gives us.



Figure 1. Topological view of the canopy network deployed across the Robert's building at University College London from February to April 2017, using SolSystem.

6.8.2. OpenMote B

In collaboration with OpenMote (http://www.openmote.com/), we have designed the OpenMote B platform. This board contains both a CC2538 IEEE802.15.4 radio, and an AT86RF215 IEEE802.15.4g radio, offering communication on both 2.4 GHz and sub-GHz frequency bands, 4 modulations schemes, and data rates from 50 kbps to 800 kbps. The first prototypes (shown in Fig. 2) started being tested in December 2017.



Figure 2. The OpenMote B. sub-GHz radio (and antenna connector) on top, 2.4 GHz radio on the bottom.

FUN Project-Team

6. New Software and Platforms

6.1. AspireRFID ALE

FUNCTIONAL DESCRIPTION: AspireRFID middleware is a modular OW2 open source RFID middleware. It is compliant with EPC Global standards. This new module integrates the modifications of the new standard release, including new RP and LLRP definitions and fixing bugs. This module has been implemented in the framework of the MIAOU project.

- Participants: Ibrahim Amadou, Julien Vandaële, Nathalie Mitton and Rim Driss
- Contact: Nathalie Mitton

6.2. ETINODE-CONTIKI-PORT

FUNCTIONAL DESCRIPTION: Contiki is an open source embedded OS for Internet of Things (IoT). It is light and portable to different hardware architectures. It embeds communication stacks for IoT II embarque aussi des piles de communication pour l'internet des objets. This driver allows the running of Contiki OS over Etinode-MSP430. The code dalso allows the use of radio chip and embedded sensors. This module has been implemented in the framework of the ETIPOPS project.

- Participants: Nathalie Mitton, Roudy Dagher and Salvatore Guzzo Bonifacio
- Contact: Salvatore Guzzo Bonifacio

6.3. ETINODE-DRIVERS

FUNCTIONAL DESCRIPTION: These drivers for Etinode-MSP430 control the different embedded sensors and hardware components available on an Etinode-MSP430 node such as gyroscope, accelerometer and barometric sensor. This module has been implemented in the framework of the ETIPOPS project.

- Participants: Nathalie Mitton, Roudy Dagher and Salvatore Guzzo Bonifacio
- Contact: Salvatore Guzzo Bonifacio

6.4. EVe-TCF

Embedded Verifier for Transitive Control Flow

KEYWORDS: Security - Embedded - Embedded systems - JavaCard - Control Flow - Code analysis FUNCTIONAL DESCRIPTION: Verification of transitive control flow policies on JavaCard 2.x bytecode. Control flow policies expressed using a DSL language are embedded in JavaCard packages (CAP files) using EVe-TCF convert tool. Control flow policies are then statically verified on-device at loading-time thanks to an embedded verifier (designed for smart cards in EVe-TCF). EVe-TCF (Embedded Verifier for Transitive Control Flow) also contains an off-device (i.e. PC tool) to simulate on-device loading process of JavaCard 2.x platforms with GlobalPlatform 2.x installed.

- Participants: Arnaud Fontaine and Isablle Simplot Ryl
- Contact: Nathalie Mitton

6.5. GOLIATH

Generic Optimized LIghtweight communication stack for Ambient TecHnologies KEYWORDS: WSN - WSN430 FUNCTIONAL DESCRIPTION: GOLIATH (Generic Optimized LIghtweight communication stack for Ambient TecHnologies) is a full protocol stack for wireless sensor networks. This module has been implemented in the framework of the ETIPOPS project.

- Participants: David Simplot Ryl, Fadila Khadar, Nathalie Mitton and Salvatore Guzzo Bonifacio
- Contact: Nathalie Mitton
- URL: https://gforge.inria.fr/projects/goliath/

6.6. IoT-LAB robots

KEYWORDS: Internet of things - Robotics

FUNCTIONAL DESCRIPTION: IoT-LAB robots is an embedded robot controler on a Turtlebot2 providing the IoT-LAB node mobility functionnality

- Partner: Université de Strasbourg
- Contact: Julien Vandaële
- URL: https://github.com/iot-lab/

6.7. T-SCAN

KEYWORDS: Rfid - RFID Middleware

FUNCTIONAL DESCRIPTION: T-Scan is an interface ensuring the translation from a SGTIN tag format to an ONS hostname format according to the EPCGlobal standards. It allows the sending of a DNS request to look up the EPC-IS aides to which the product belongs in order to access the data relative to that product. This module has been implemented in the framework of the TRACAVERRE project.

- Participants: Gabriele Sabatino and Nathalie Mitton
- Contact: Gabriele Sabatino

6.8. FIT IoT-Lab

Participants: Nathalie Mitton [correspondant], Julien Vandaele, Matthieu Berthome.

FIT IoT-LAB is a very large scale open testbed that features over 2700 wireless sensor nodes and more than 200 robots spread across six different sites in France. Nodes are either fixed or mobile and can be allocated in various topologies throughout all sites. A variety of wireless sensors are available, with different processor architectures (MSP430, STM32 and Cortex-A8) and different wireless chips (802.15.4 PHY at 800 MHz or 2.4 GHz). In addition, "open nodes" can receive custom wireless sensors for inclusion in IoT-LAB testbed. This platform is completely open and can be used by any one wishing to run experiment on wireless sensors and robots.

The Lille site displays 3 subsets of the platforms:

- Euratechnologies: this site features 256 WSN430 sensor nodes operating in the 2.4GHz band. 64 nodes are mobile, embedded on mobile trains.
- Haute Borne: this site features 256 M3 sensor nodes operating in the 2.4GHz band and 64 mobile robots (32 turtlebots and 32 wifibots) completely remotely programmable.
- Opennodes: this site will feature (opening beginning 2015) 64 hardware open slots to allow any one to plug his own hardware and benefits from the platform debugging and monitoring tools.

GANG Project-Team

5. New Software and Platforms

5.1. big-graph-tools

FUNCTIONAL DESCRIPTION: Gang is developping a software for big graph manipulation. A preliminary library offering diameter and skeleton computation is available at https://who.rocq.inria.fr/Laurent.Viennot/dev/big-graph-tools/. This library was used to compute the diameters of the worldwide road network (200M edges) and the largest strongly connected component of the Twitter follower-followee graph (23G edges).

- Contact: Laurent Viennot
- URL: https://who.rocq.inria.fr/Laurent.Viennot/dev/big-graph-tools/

5.2. GRPH

The high performance graph library for Java

KEYWORDS: Graph - Graph algorithmics - Java

FUNCTIONAL DESCRIPTION: Grph is an open-source Java library for the manipulation of graphs. Its design objectives are to make it portable, simple to use/extend, computationally/memory efficient, and, according to its initial motivation: useful in the context of graph experimentation and network simulation. Grph also has the particularity to come with tools like an evolutionary computation engine, a bridge to linear programming solvers, a framework for distributed computing, etc.

Grph offers a very general model of graphs. Unlike other graph libraries which impose the user to first decide if he wants to deal with directed, undirected, hyper (or not) graphs, the model offered by Grph is unified in a general class that supports mixed graphs made of undirected and directed simple and hyper edges. Grph achieves great efficiency through the use of multiple code optimization techniques such as multi-core parallelism, caching, adequate data structures, use of primitive objects, exploitation of low-level processor caches, on-the-fly compilation of specific C/C++ code, etc. Grph attempts to access the Internet in order to check if a new version is available and to report who is using it (login name and hostname). This has no impact whatsoever on performance and security.

- Participants: Aurélien Lancin, David Coudert, Issam Tahiri, Luc Hogie and Nathann Cohen
- Contact: Luc Hogie
- URL: http://www.i3s.unice.fr/~hogie/grph/

INFINE Project-Team

5. New Software and Platforms

5.1. Gardinet

KEYWORD: Distributed networks

FUNCTIONAL DESCRIPTION: Gardinet (previously DragonNet) is a generic framework for network coding in wireless networks. It is a initially result of the GETRF project of the Hipercom2 team.

It is based on intra-flow coding where the source divides the flow in a sequence of payloads of equal size (padding may be used). The design keys of DragonNet are simplicity and universality, DragonNet does not use explicit or implicit knowledge about the topology (such as the direction or distance to the source, the loss rate of the links, ...). Hence, it is perfectly suited to the most dynamic wireless networks. The protocol is distributed and requires minimal coordination. DragonNet architecture is modular, it is based on 5 building blocks (LIB, SIG, Protocol, SEW and DRAGON). Each block is almost independent. This makes DragonNet generic and hence adaptable to many application scenarios. DragonNet derives from a prior protocol called DRAGONCAST. Indeed, DragonNet shares the same principles and theoretical overview of DRAGONCAST. It enriches DRAGONCAST by the information base and signaling required to perform broadcast in wireless networks and in wireless sensor networks in particular.

- Participants: Antonia Masucci, Cédric Adjih, Hana Baccouch and Ichrak Amdouni
- Contact: Cédric Adjih
- URL: http://gitlab.inria.fr/gardinet

5.2. MACACO

Mobile context-Adaptive CAching for COntent-centric networking

FUNCTIONAL DESCRIPTION: MACACOapp is developped in the context of the EU CHIST-ERA MACACO project. It consists in a mobile phone application that periodically samples phone's information on the mobility (through, e.g., GPS sensor, accelerometer and WiFi/Bluetooth/Cellular environment, connectivity type) and on the data traffic it generates (through, e.g., Internet browser history and applications data consumption). The information collected will be time-stamped and will be periodically sent to the central servers for analysis and visualization. We expect that (1) the collected information will allow us studying the correlation between mobility and content demand patterns and that (2) the results of this analysis will allow us inferring the best times and places to transfer content from/to users' phones location and/or from/to the wireless infrastructure closest to the users' phones location. Users will be also invited to fill a non-mandatory questionnaire relevant to this study. Our questionnaire collects information about the personality traits and application preferences of people. We expect that the information collected from questionnaire will allow us to analyse the correlation between users personality traits and their application preferences and interests. User's application preferences and interests will be inferred from the Internet browsing history and running app information obtained from the MACACO App.

- Participants: Aline Carneiro Viana, Katia Jaffres and Marco Fiore
- Contact: Aline Carneiro Viana
- URL: https://macaco.inria.fr/macacoapp/

5.3. RIOT

KEYWORDS: Internet of things - Wireless Sensor Networks - Iot - Sensors - Operating system - Internet protocols

SCIENTIFIC DESCRIPTION: While requiring as low as 1,5kB of RAM and 5kB or ROM, RIOT offers real time and energy efficiency capabilities, as well as a single API (partially POSIX compliant) across heterogeneous 8-bit, 16-bit and 32-bit low-hardware. This API is developer-friendly in that it enables multi-threading, standard C and C++ application programming and the use of standard debugging tools (which was not possible so far for embedded programming). On top of this, RIOT includes several network stacks, such as a standard IPv6/6LoWPAN stack and a information-centric network stack (based on CCN).

FUNCTIONAL DESCRIPTION: RIOT is an Open Source operating system that provides standard protocols for embedded systems. RIOT allows, for example, the development of applications that collect sensor data and transmit it to a central node (e.g. a server). This data can then be used for smart energy management for instance.

RIOT is specially designed for embedded systems, which are strongly constrained in memory and energy. Further, RIOT can easily be ported to different hardware devices and follows the latest evolution of IP standards.

RIOT applications can readily be tested in the FIT IoT-Lab, which provides a large-scale infrastructure facility with 3000 nodes for testing remotely small wireless devices.

- Participants: Emmanuel Baccelli and Oliver Hahm
- Partner: Freie Universität Berlin
- Contact: Emmanuel Baccelli
- URL: http://www.riot-os.org

MADYNES Team

6. New Software and Platforms

6.1. Distem

KEYWORDS: Large scale - Experimentation - Virtualization - Emulation

FUNCTIONAL DESCRIPTION: Distem is a distributed systems emulator. When doing research on Cloud, P2P, High Performance Computing or Grid systems, it can be used to transform an homogeneous cluster (composed of identical nodes) into an experimental platform where nodes have different performance, and are linked together through a complex network topology, making it the ideal tool to benchmark applications targetting such environments, or aiming at tolerating performance degradations or variations which are frequent in the Cloud or in other applications distributed at large scale (P2P for example).

- Participants: Luc Sarzyniec, Lucas Nussbaum and Tomasz Buchert
- Partners: CNRS Université de Lorraine Loria Grid'5000 Inria
- Contact: Lucas Nussbaum
- URL: http://distem.gforge.inria.fr

6.2. Grid'5000 testbed

FUNCTIONAL DESCRIPTION: Grid'5000 is a scientific instrument designed to support experiment-driven research in all areas of computer science related to parallel, large-scale or distributed computing and networking. It gathers 10 sites, 25 clusters, 1200 nodes, for a total of 8000 cores. It provides its users with a fully reconfigurable environment (bare metal OS deployment with Kadeploy, network isolation with KaVLAN) and a strong focus on enabling high-quality, reproducible experiments.

- Participants: Arthur Garnier, Clement Parisot, Émile Morel, Emmanuel Jeanvoine, Jérémie Gaidamour, Luc Sarzyniec and Lucas Nussbaum
- Contact: Lucas Nussbaum
- URL: https://www.grid5000.fr/

6.3. Kadeploy

KEYWORD: Operating system provisioning

FUNCTIONAL DESCRIPTION: Kadeploy is a scalable, efficient and reliable deployment (provisioning) system for clusters and grids. It provides a set of tools for cloning, configuring (post installation) and managing cluster nodes. It can deploy a 300-nodes cluster in a few minutes, without intervention from the system administrator. It plays a key role on the Grid'5000 testbed, where it allows users to reconfigure the software environment on the nodes, and is also used on a dozen of production clusters both inside and outside Inria.

- Participants: Emmanuel Jeanvoine, Luc Sarzyniec and Lucas Nussbaum
- Partners: CNRS Université de Lorraine Loria Grid'5000 Inria
- Contact: Lucas Nussbaum
- URL: http://kadeploy3.gforge.inria.fr

6.4. MECSYCO-RE-C++

en Multi-agent Environment for Complex SYstems COsimulation. Coeur C++ KEYWORDS: Agent - Multi-agent - Multi-model - Simulator - Simulation - Modeling - Artefact FUNCTIONAL DESCRIPTION: MECSYCO is a project aiming at the modeling and simulation of complex systems. It provides concepts and tools to describe and then simulate a system as a set of heterogeneous models (namely a multi-model). MECSYCO-RE-C++ is the C++ implementation of the central part (core) of MECSYCO. It can be complimente by mecsyco-com (a communication package for distributed exécution) and mecsyco-visu (a set of tools for vizualizaing simulations).

- Participants: Benjamin Camus, Benjamin Segault, Julien Vaubourg, Laurent Ciarletta, Nicolas Kirchner, Victorien Elvinger, Vincent Chevrier and Yannick Presse
- Partners: Université de Lorraine Inria
- Contact: Vincent Chevrier

6.5. MECSYCO-RE-java

Multi-agent Environment for Complex SYstems COsimulation. Coeur java

KEYWORDS: Agent - Multi-agent - Co-simulation - Multi-model - Simulator - Simulation - Modeling - Artefact

FUNCTIONAL DESCRIPTION: MECSYCO is a project aiming at the modeling and simulation of complex systems. It provides concepts and tools to describe and then simulate a system as a set of heterogeneous models (namely a multi-model). MECSYCO-RE-java is the Java implementation of the central part (core) of MECSYCO. It can be complemented by mecsyco-com (a communication package for distributed exécution) and mecsyco-visu (a set of tools for vizualizaing simulations).

- Participants: Benjamin Camus, Christine Bourjot, Julien Siebert, Julien Vaubourg, Laurent Ciarletta, Victorien Elvinger, Vincent Chevrier and Yannick Presse
- Partners: Université de Lorraine Inria
- Contact: Vincent Chevrier
- URL: http://www.mecsyco.com

6.6. Ruby-cute

KEYWORDS: Experimentation - HPC - Cloud

FUNCTIONAL DESCRIPTION: Ruby-Cute is a set of Commonly Used Tools for Experiments, or Critically Useful Tools for Experiments, depending on who you ask. It is a library aggregating various Ruby snippets useful in the context of (but not limited to) development of experiment software on distributed systems testbeds such as Grid'5000.

- Contact: Lucas Nussbaum
- URL: http://ruby-cute.github.io/

6.7. Platforms

6.7.1. CPS Security Assessment Platform

This year, we have extended our Cyber-Physical systems security assessment platform with new hardware components including multiple types of Programmable Logic Controllers (PLS) and a small scale distribution and sorting testbed. The physical platform is also extended with several IoT devices dedicated to residential networks (heating control, lightning system, home gateways, etc). The platform will be mainly used for building security assessment and evaluation experimentation on the available devices to identify and validate their associated attack patterns and discover new vulnerabilities.

NEO Project-Team

6. New Software and Platforms

6.1. marmoteCore

Markov Modeling Tools and Environments - the Core

KEYWORDS: Modeling - Stochastic models - Markov model

FUNCTIONAL DESCRIPTION: marmoteCore is a C++ environment for modeling with Markov chains. It consists in a reduced set of high-level abstractions for constructing state spaces, transition structures and Markov chains (discrete-time and continuous-time). It provides the ability of constructing hierarchies of Markov models, from the most general to the particular, and equip each level with specifically optimized solution methods.

This software is developed within the ANR MARMOTE project: ANR-12-MONU-00019.

- Participants: Alain Jean-Marie, Hlib Mykhailenko, Benjamin Briot, Franck Quessette, Issam Rabhi, Jean-Marc Vincent and Jean-Michel Fourneau
- Partner: UVSQ
- Contact: Alain Jean-Marie
- Publications: marmoteCore: a Markov Modeling Platform marmoteCore: a software platform for Markov modeling
- URL: http://marmotecore.gforge.inria.fr/

RAP2 Team (section vide)

SOCRATE Project-Team

5. New Software and Platforms

5.1. fftweb

KEYWORDS: Experimentation - Data visualization - SDR (Software Defined Radio)

FUNCTIONAL DESCRIPTION: fftweb is a real-time spectral (FFT) visualization of one or several signal, embedded in a web page. The FFT is computed in a GNURadio block, then sent to a gateway server, which serves the web page, associated javascripts, and signal websockets. The end user only has to use the GNURadio block and and the web page, and doesn't need to bother about the internal details of the system. fftweb has been developped specially for the CorteXlab testbed but with minor adaptations, it can be used in other contexts, and also can be used to draw more generic real-time graphs, not only FFTs. Technologies: GNURadio, python, python-gevent, Javascript, D3JS

• Contact: Matthieu Imbert

5.2. FloPoCo

Floating-Point Cores, but not only

KEYWORD: Synthesizable VHDL generator

FUNCTIONAL DESCRIPTION: The purpose of the open-source FloPoCo project is to explore the many ways in which the flexibility of the FPGA target can be exploited in the arithmetic realm.

- Participants: Antoine Martinet, Florent Dupont De Dinechin, Matei Istoan and Nicolas Brunie
- Partners: CNRS ENS Lyon UCBL Lyon 1 UPVD
- Contact: Florent Dupont De Dinechin
- URL: http://flopoco.gforge.inria.fr/

5.3. minus

KEYWORDS: Experimentation - SDR (Software Defined Radio)

FUNCTIONAL DESCRIPTION: Minus is an experiment control system able to control, the whole lifecycle of a radio experiment in CorteXlab or any other testbed inspired by it. Minus controls and automates the whole experiment process starting from node power cycling, experiment deployment, experiment start and stop, and results collection and transfer. Minus is also capable of managing multiple queues of experiments which are executed simultaneously in the testbed.

• Contact: Matthieu Imbert

5.4. WiPlan

FUNCTIONAL DESCRIPTION: Wiplan is a software including an Indoor propagation engine and a wireless LAN optimization suite, which has been registered by INSA-Lyon. The heart of this software is the propagation simulation core relying on an original method, MR-FDPF (multi-resolution frequency domain ParFlow), proposed by JM Gorce in 2001 and further extended. The discrete ParFlow equations are translated in the Fourier domain providing a large linear system, solved in two steps taking advantage of a multi- resolution approach. The first step computes a cell-based tree structure referred to as the pyramid. In the second phase, a radiating source is simulated, taking advantage of the pre-processed pyramidal structure. Using of a full-space discrete simulator instead of classical ray-tracing techniques is a challenge due to the inherent high computational load to be restricted to a pre-processing phase. Extensive works have been done to make predictions more realistic.

• Contact: Tanguy Risset

5.5. Sytare

KEYWORDS: Embedded systems - Operating system - Non volatile memory

FUNCTIONAL DESCRIPTION: Sytare is an embedded operating system targeting tiny platforms with intermittent power. In order to make power failures transparent for the application, the system detects imminent failures and saves a checkpoint of program state to non-volatile memory. Hardware peripherals are also made persistent without requiring developer attention.

- Authors: Gautier Berthou, Tristan Delizy, Kevin Marquet and Guillaume Salagnac
- Contact: Guillaume Salagnac
- Publication: Peripheral State Persistence For Transiently Powered Systems
- URL: https://hal.inria.fr/hal-01460699

5.6. Platform - FIT/CortexLab

FIT (Future Internet of Things) is a french Equipex (Équipement d'excellence) which aims to develop an experimental facility, a federated and competitive infrastructure with international visibility and a broad panel of customers. FIT is be composed of four main parts: a Network Operations Center (FIT NOC), a set of IoT test-beds (FIT IoT-Lab), a set of wireless test-beds (FIT-Wireless) which includes the FIT/CorteXlab platform deployed by the Socrate team in the Citi lab, and finaly a set of Cloud test-beds (FIT-Cloud). In 2014 the construction of the room was finished see Figure 5 . SDR nodes have installed in the room, 42 industrial PCs (Aplus Nuvo-3000E/P), 22 NI radio boards (USRP) and 18 Nutaq boards (PicoSDR, 2x2 and 4X4) can be programmed from internet now.

A very successfully inauguration took place int 2014⁰, with the noticable venue of Vincent Poor, Dean of School of Engineering and Applied Science of Princeton University. Since that date, the platform is open to public experiments. CorteXlab has been used for innovative experiments such as Interference Alignment in Cellular Networks for Energy Efficiency Improvement demonstrated in the GreenTouch Consortium.



Figure 5. Photo of the FIT/CortexLab experimentation room installed and a snaptshot of the inauguration meeting

⁰http://www.inria.fr/centre/grenoble/actualites/inauguration-reussie-de-la-plateforme-cortexlab-equipex-fit