



RESEARCH CENTER

FIELD

**Networks, Systems and Services,
Distributed Computing**

Activity Report 2016

Section Software

Edition: 2017-08-25

DISTRIBUTED SYSTEMS AND MIDDLEWARE

1. ASAP Project-Team 5
2. CIDRE Project-Team 9
3. COAST Project-Team 12
4. CTRL-A Team 14
5. MIMOVE Team 15
6. MYRIADS Project-Team 19
7. REGAL Project-Team 24
8. SPIRALS Project-Team 25
9. WHISPER Project-Team 27

DISTRIBUTED AND HIGH PERFORMANCE COMPUTING

10. ALPINES Project-Team 29
11. AVALON Project-Team 31
12. DATAMOVE Team 34
13. HIEPACS Project-Team 36
14. KERDATA Project-Team 40
15. POLARIS Team 43
16. ROMA Project-Team 46
17. STORM Team 49
18. TADAAM Team 53

DISTRIBUTED PROGRAMMING AND SOFTWARE ENGINEERING

19. ASCOLA Project-Team 56
20. DIVERSE Project-Team 59
21. FOCUS Project-Team 63
22. INDES Project-Team 67
23. PHOENIX Project-Team 68
24. RMOD Project-Team 74
25. TACOMA Team 75

NETWORKS AND TELECOMMUNICATIONS

26. COATI Project-Team 78
27. DANTE Project-Team 80
28. DIANA Project-Team 82
29. DIONYSOS Project-Team 85
30. DYOGENE Project-Team 87
31. EVA Project-Team 88
32. FUN Project-Team 92
33. GANG Project-Team 93
34. INFINE Project-Team 94
35. MADYNES Project-Team 96
36. MAESTRO Project-Team 98
37. MUSE Team 99

38. RAP Project-Team (section vide)	101
39. SOCRATE Project-Team	102
40. URBANET Team	105

ASAP Project-Team

5. New Software and Platforms

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

5.2. Brow2Brow

Browser-to-browser serverless toolboxes

FUNCTIONAL DESCRIPTION

Brow2Brow is an “Action de Development Technologique”, i.e. a collaborative development project that aims at providing a middleware and software library for browser-to-browser applications. Brow2Brow involves the ASAP team as well as the DICE Team from Inria Grenoble (Antenne de Lyon). The project seeks to provide an alternative to the current model followed by Web2.0 applications by exploiting the recently introduced WebRTC standard. Existing Web 2.0 applications collect data on browsers and send it to servers that store and process it. The goal of Brow2Brow is to provide an alternative approach where browsers can themselves proceed to collaborative data processing. This will make it possible avoid data concentration at a single server. The project has resulted so far in the development of WebGC, a library for gossip-based applications on browsers.

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

5.3. Dashboard

MediEgo Dashboard: A personalized news dashboard

KEYWORDS: Recommender system - Personalized stream of news - Dashboard

FUNCTIONAL DESCRIPTION

This work has led to the development of MEDIEGO Dashboard, a personalized news recommendation system. In MEDIEGO Dashboard, users benefit from a personalized stream of news matching their interests. Additionally, users can use explicit subscriptions as well as post content and navigate through tags. MEDIEGO Dashboard is available through a web interface and a mobile-based Android application. To provide personalization, MEDIEGO Dashboard exploits the users’ opinions regarding their received news to identify users with similar interests. MEDIEGO Dashboard is centralized and it allows us to test and evaluate different recommendation schemes. In collaboration with EIT/ICT Lab, an experiment has been conducted with a set of users at Trento (Italie). This experiment allowed us to collect traces and to perform a user survey to assess and improve our solution. This solution will soon be interconnected to P2P-AllYours.

- Participants: Anne-Marie Kermarrec, Antoine Boutet, Yuri Barssi and Jean Francois Verdonck
- Contact: Anne-Marie Kermarrec
- URL: <http://www.mediego.com>

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.

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

5.5. HEAP

Heterogeneity-aware gossip protocol

FUNCTIONAL DESCRIPTION

A video streaming platform based on HEAP. The platform is particularly suited for environment characterized by heterogeneous bandwidth capabilities such as those comprising ADSL edge nodes. HEAP is, in fact, able to dynamically leverage the most capable nodes and increase their contribution to the protocol, while decreasing by the same proportion that of less capable nodes. During the last few months, we have integrated HEAP with the ability to dynamically measure the available bandwidth of nodes, thereby making it independent of the input of the user.

- Participants: Davide Frey, Arnaud Jegou, Anne-Marie Kermarrec, Vivien Quema, Maxime Monod and Rachid Guerraoui
- Contact: Davide Frey

5.6. HyRec

Hybrid Recommender System

FUNCTIONAL DESCRIPTION

The motivation of this work is to explore solutions that could in some sense democratize personalization by making it accessible to any content provider company without generating huge investments. HyRec implements a user-based collaborative filtering scheme and offloads CPU-intensive recommendation tasks to front-end client browsers, while retaining storage and orchestration tasks within back-end servers. HyRec seeks to provide the scalability of p2p approaches without forcing content providers to give up the control of the system.

- Participants: Antoine Boutet, Davide Frey, Anne Marie Kermarrec, Arnaud Jegou and Rachid Guerraoui
- Contact: Davide Frey

5.7. WebGC

Web-based Gossip Communication

FUNCTIONAL DESCRIPTION

WebGC is a library for gossip-based communication between web-browsers. It has been developed in collaboration with Mathieu Simonin in the context of the Brow2Brow ADT project. WebGC builds on the recent WebRTC standard as well as on PeerJS, an open-source project that provides primitives for data transfer on top of WebRTC.

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

5.8. WhatsUp

KEYWORD: Recommender system

FUNCTIONAL DESCRIPTION

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

- Participants: Davide Frey, Ribeiro Heverson, Antoine Boutet, Anne Marie Kermarrec, Arnaud Jegou, Rachid Guerraoui and Jean Francois Verdonck
- Contact: Davide Frey

5.9. YALPS

KEYWORDS: Traffic-shaping - Nat traversal - Experimentation - Peer-to-peer - Simulator - 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, P2P-AllYours, and Behave.

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

5.10. P2P-AllYours

Peer-to-Peer AllYours

FUNCTIONAL DESCRIPTION

P2P-AllYours is customization of WhatsUp developed in the context of the EIT/ICT-Labs AllYours project. In addition to WhatsUp (the distributed recommender engine), P2P-AllYours comprises the following features:

- A new web interface, which users can access through a local web-server integrated in P2P-AllYours.
- A set of automatic nodes (BOTs) that can extract news items from RSS feeds and insert them into the recommender system.
- A content-bootstrap that solves the issues related to bootstrapping the recommender system when a user connects for the first time.
- An experiment management server that allows users to register for the application in the context of the testing program.
- Participants: Davide Frey, Heverson Borba Ribeiro, Raziel Carvajal Gomez, Arnaud Jegou and Anne-Marie Kermarrec
- Contact: Davide Frey

CIDRE Project-Team

6. New Software and Platforms

6.1. Blare

To detect intrusion using information flows

KEYWORDS: Cybersecurity - Intrusion Detection Systems (IDS) - Data Leakage Protection

SCIENTIFIC DESCRIPTION

Blare implements our approach of illegal information flow detection for a single node (Android and Linux kernel, JVM) and a set of nodes (monitoring of flows between linux machines).

FUNCTIONAL DESCRIPTION

Blare IDS is a set of tools that implements our approach to illegal information flow detection for a single node and a set of nodes.

- Partner: SUPELEC
- Contact: Frédéric Tronel
- URL: <http://blare-ids.org>

6.2. ELVIS

Extensible Log VISualization

KEYWORDS: Visualization - Cybersecurity - Intrusion Detection Systems (IDS) - SIEM - Cyber attack - Forensics

SCIENTIFIC DESCRIPTION

The studies that were performed since 2012 clearly showed that there was an important need for technologies that would allow analysts to handle in a consistent way the various types of log files that they have to study in order to detect intrusion or to perform forensic analysis. Consequently, we proposed this year ELVIs, a security-oriented log visualization system that allows the analyst to import its log files and to obtain automatically a relevant representation of their content based on the type of the fields they are made of. First, a summary view is proposed. This summary displays in an adequate manner each field according to its type (i.e. categorical, ordinal, geographical, etc.). Then, the analyst can select one or more fields to obtain some details about it. A relevant representation is then automatically selected by the tool according to the types of the fields that were selected.

ELVIs [35] has been presented in VizSec 2013 (part of Vis 2013) in October 2013 in Atlanta. A working prototype is currently being tuned in order to perform field trials with our partners in DGA-MI. Next year, we are planing to perform research on how various log files can be combined in the same representation.

FUNCTIONAL DESCRIPTION

ELVIS is a visualisation tool geared to system security which enables analysts to visually explore log files using relevant representations. The tool accepts many different types of log file and can easily be extended to accept new ones opportunisticly. Thanks to its data typing mechanisms, it can automatically choose relevant representations depending on the type of data that the analyst wants to observe.

- Participant: Nicolas Prigent
- Partner: SUPELEC
- Contact: Nicolas Prigent
- URL: <https://hal.inria.fr/hal-00875668>

6.3. GEPETO

GEoPrivacy-Enhancing TOolkit

KEYWORDS: Cyber attack - Privacy - Mobility

SCIENTIFIC DESCRIPTION

(GEoPrivacy-Enhancing TOolkit) is an open source software for managing location data (currently in development in cooperation with LAAS). GEPETO can be used to visualize, sanitize, perform inference attacks and measure the utility of a particular geolocated dataset. For each of these actions, a set of different techniques and algorithms can be applied. The global objective of GEPETO is to enable a user to design, tune, experiment and evaluate various sanitization algorithms and inference attacks as well as visualizing the following results and evaluating the resulting trade-off between privacy and utility. An engineer (Izabela Moise) has contributed to the development of a distributed version of GEPETO based on the MapReduce paradigm and the Hadoop framework that is able to analyze datasets composed of millions of mobility traces in a few minutes [30].

FUNCTIONAL DESCRIPTION

GEPETO is an open source software for managing location data. GEPETO can be used to visualize, sanitize, perform inference attacks, and measures the utility of a particular geolocated dataset.

- Partners: CNRS - Université de Rennes 1
- Contact: Sébastien Gambs
- URL: <https://gforge.inria.fr/projects/gepeto/>

6.4. GNG

Security Supervision by Alert Correlation

KEYWORDS: Intrusion Detection Systems (IDS) - SIEM

SCIENTIFIC DESCRIPTION

GNG is an intrusion detection system that correlates different sources (such as different logs) in order to identify attacks against the system. The attack scenarios are defined using the Attack Description Language (ADeLe) proposed by our team, and are internally translated to attack recognition automatons. GNG intends to define time efficient algorithms based on these automatons to recognize complex attack scenarios.

- Partner: SUPELEC
- Contact: Eric Totel
- URL: <http://www.rennes.supelec.fr/ren/perso/etotel/GNG/index.html>

6.5. GroddDroid

KEYWORDS: Android - Detection - Malware

FUNCTIONAL DESCRIPTION

GroddDroid

1- locates suspicious code in Android application

2- computes execution paths towards suspicious code

3- forces executions of suspicious code

- Partners: CentraleSupélec - Insa Centre Val-de-Loire
- Contact: Valérie Viet Triem Tong
- URL: <http://kharon.gforge.inria.fr/grodddroid.html>

6.6. Kharon platform

KEYWORDS: Android - Malware - Dynamic Analysis

FUNCTIONAL DESCRIPTION

This platform executes Android applications and computes a graph representing all the information flows that occurred in the operating system due to a malicious execution. It can then classify observed behavior as benign or malicious. Access to this platform is currently in physically controlled at the high security laboratory (LHS) of Rennes.

- Partners: CentraleSupélec - Insa Centre Val-de-Loire
- Contact: Valérie Viet Triem Tong
- URL: <http://kharon.gforge.inria.fr/>

6.7. Netzob

FUNCTIONAL DESCRIPTION

Netzob is an opensource tool for reverse engineering, traffic generation and fuzzing of communication protocols. This tool allows to infer the message format (vocabulary) and the state machine (grammar) of a protocol through passive and active processes. Its objective is to bring state of art academic researches to the operational field, by leveraging bio-informatic and grammatical inferring algorithms in a semi-automatic manner.

- Participant: Georges Bossert
- Contact: Ludovic Mé
- URL: <http://www.netzob.org/>

6.8. VEGAS

Visualizing, Exploring and Grouping Alerts

KEYWORDS: Security - Visualization - Cybersecurity - Intrusion Detection Systems (IDS) - SIEM

SCIENTIFIC DESCRIPTION

VEGAS explore the hypothesis that is possible to offer to front-line security operators a visualization tool that allows the to perform a first informed triage of the alerts that were received from IDSes so as to group them and transmit them to security analysts in a relevant way.

FUNCTIONAL DESCRIPTION

VEGAS is a visualization tool that allows to easily identify, explore and group alerts generated by an IDS. This tool allows security operators to easily dispatch similar alerts to security analyst to help them study them more efficiently.

- Participants: Damien Cremilleux, Frédéric Majorczyk and Nicolas Prigent
- Partner: SUPELEC
- Contact: Damien Crémilleux

COAST Project-Team

4. New Software and Platforms

4.1. MUTE

Multi-User Text Editor

FUNCTIONAL DESCRIPTION

MUTE (Multi-User Text Editor) is a web-based text editing tool that allows users 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: Gérald Oster, François Charoy, Claudia-Lavinia Ignat, Phillippe Kalitine, Matthieu Nicolas and Victorien Elvinger
- Contact: Gérald Oster
- URL: <https://github.com/coast-team/mute/>

4.2. NetFlux

Peer-to-Peer Network Library over WebRTC

FUNCTIONAL DESCRIPTION

NetFlux is a Nodejs library that allows users to deploy a peer-to-peer network between web browsers using the WebRTC technology.

- Participants: Gérald Oster, Phillippe Kalitine, Matthieu Nicolas.
- Contact: Gérald Oster
- URL: <https://github.com/coast-team/netflux>

4.3. MUTE-structs

Peer-to-Peer Network Library over WebRTC

FUNCTIONAL DESCRIPTION

MUTE-structs is a Nodejs module that provides an implementation of the LogootSplit CRDT algorithm. It is an optimistic replication algorithm that ensures eventual consistency on replicated text sequences. It is used in the MUTE real-time collaborative text editor.

- Participants: Gérald Oster, Claudia-Lavinia Ignat, Phillippe Kalitine, Matthieu Nicolas and Victorien Elvinger
- Contact: Gérald Oster
- URL: <https://github.com/coast-team/mute-structs>

4.4. Replication Benchmark

FUNCTIONAL DESCRIPTION

The Replication Benchmark 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 event traces. These event traces can be produced randomly according to different parameters, can be extracted from actual 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 terms of execution time, memory footprint and quality of merge result (compared to manual merge history stored in git repositories).

- Participants: Pascal Urso and Gérald Oster
- Contact: Pascal Urso
- URL: <https://github.com/score-team/replication-benchmark/>

4.5. 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 Team

6. New Software and Platforms

6.1. Heptagon BZR

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.

Heptagon BZR is an extension of Heptagon, equipped with a behavioral contract mechanisms, where assumptions can be described, as well as an "enforce" property part. Its main feature is to include discrete controller synthesis within its compilation. The semantics of contracts 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.

- Participants: Adrien Guatto, Marc Pouzet, Cédric Pasteur, Léonard Gerard, Brice Gelineau, Gwenael Delaval and Eric Rutten
- Contact: Gwenaël Delaval
- <http://bzs.inria.fr>

MIMOVE Team

6. New Software and Platforms

6.1. Introduction

In order to validate our research results and, in certain cases, make them available to specific communities or to the public, our research activities encompass the development of related software as surveyed below.

6.2. VSB: eVolution Service Bus for the Future Internet

Participants: Georgios Bouloukakis, Nikolaos Georgantas [contact], Siddhartha Dutta.

URL: <https://tuleap.ow2.org/plugins/git/chorevolution/evolution-service-bus>

The *eVolution Service Bus (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 CoAP⁰. 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. In this paradigm, a common intermediate bus protocol is used to facilitate interconnection between multiple heterogeneous middleware protocols. Conversion of each protocol to the common bus protocol is done by a component associated to the service/Thing in question and its middleware, called a Binding Component (BC), as it binds the service/Thing to the service bus. We introduce a generic architecture for VSB, which relies on the notion of *Generic Middleware (GM)* connector. GM abstracts interactions among peer components that employ the same middleware protocol through generic *post* and *get* operations, in a unifying fashion for any middleware protocol. We propose an API (application programming interface) for GM and a related generic interface description, which we call *GIDL*, for application components that (abstractly) employ GM. Concrete middleware protocols and related interface descriptions of application components that employ these middleware protocols can be mapped to GM API and GIDL, respectively. Based on these abstractions, we elaborate a generic architecture for BCs, as well as a related method for BC synthesis and refinement for a concrete choreography that includes services/Things with heterogeneous middleware protocols.

The eVolution Service Bus (VSB) presents a significant rethinking of the architecture and the implementation of a service bus destined to serve dynamic choreographies of services but also Things as first-class entities. More specifically, VSB presents the following advancements:

- VSB is a unified interoperability solution for both services and Things participating in choreographies;
- VSB is flexible and lightweight: it is a completely decentralized network of BCs that are deployed as necessary; hence, no BC is needed when a service/Thing employs the same middleware protocol as the one used as common bus protocol;

⁰<https://tools.ietf.org/html/rfc7252>

- VSB provides support for the client-server, publish/subscribe, tuple space and data streaming interaction paradigms;
- Different protocols can be introduced as VSB's common bus protocol with the same easiness as for integrating support for a new middleware protocol of a service/Thing; additionally, there is no need for relying on and/or providing a full-fledged ESB platform;
- While very modular, VSB's architecture includes only few levels of indirection in the processing of primitives when converting between protocols; this makes it simple, lightweight and fast;
- In VSB, mapping between a concrete middleware protocol and the GM paradigm can be performed in different ways, thus enabling to cover all possible interaction cases; there is no unique, fixed mapping limiting the applicability of the solution;
- BC synthesis follows a systematic method allowing for its automation: we have developed related tools that support the automated generation of a BC for a service/Thing from its GIDL interface description.

VSB is being developed within the H2020 CHOReVOLUTION project (see § 8.2.1.1). It is also based on previous development carried out in the FP7 CHOReOS project⁰. VSB is available for download under open source license.

6.3. Ambiciti App & Platform: Monitoring the Exposure to Environmental Pollution

Participants: Valerie Issarny [contact], Cong Kinh Nguyen, Pierre-Guillaume Raverdy, Fadwa Rebhi.

URL: <http://ambiciti.io>

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 through dedicated Apps and Platforms that leverage Inria research results in the area of mobile distributed systems (from MiMove team) and data assimilation (from Inria CLIME team). The Ambiciti app is available for download on both the App and the Play stores. Starting December 2016, the Ambiciti software solutions are licensed to the Ambiciti start-up.

Monitoring exposure to noise pollution: Noise pollution is a major environmental health problem, with an estimated number of 10,000 premature deaths each year in Europe. The 2002 European environmental noise directive defines a common approach intended to avoid, prevent or reduce the harmful effects of noise. It requires the determination of exposure to environmental noise in major cities, through noise mapping. Until recently, this has been done solely through numerical simulation. Daytime, evening and nighttime averages are generally produced, without distinction between the different days of the year. Also, it is difficult to fill the gap between a noise map and the actual exposure of people where they live and stay. This motivates to monitor noise pollution where and when people are exposed. One promising direction is to make use of the noise sensors that people carry most of the time, i.e., the microphones embedded in their mobile phones.

Ambiciti (previously called SoundCity as it was initially focused on the monitoring of noise pollution) measures the actual noise levels to which individuals are exposed using such an approach, while taking into account the relatively low quality of the collected measurements. Ambiciti can then monitor noise levels throughout the day and inform users about their instantaneous, hourly and daily exposures.

In addition to the monitoring of the individual exposure to the noise pollution using mobile phones, the collective exposure may be derived from crowd-sensing. The adoption rate of mobile phones makes it possible to collect a huge amount of observational data about the noise pollution at the city scale. Recent studies have indeed highlighted the emergence of new environmental monitoring schemes leveraging the combination of mobile phones-embedded sensors and citizen participation. Ambiciti then leverages a mobile phone sensing middleware for collecting noise measurements at the urban scale, which are then assimilated toward the production of real-time pollution maps.

⁰<http://www.choreos.eu>

Monitoring exposure to air pollution: The Ambiciti app delivers information about the exposure to air pollution, providing hourly air quality maps, which are computed using numerical simulation. Depending on the user's location, the user may have access to hourly air quality maps, at street resolution, in real time and for the next two days. Currently, only Paris (France) and the Bay area enjoy such high resolution maps, but other cities are on the way to be included.

Mobile app features: Ambiciti is easy to use, while featuring various functions to meet different levels of user engagement (from passive monitoring to citizen scientists):

- Measuring noise level, anytime on demand or automatically during the day,
- Providing air quality indexes (according to the EU definition), in the past, present and future hours or days, together with pollution levels for nitrogen dioxide, fine particulate matter and ozone,
- Displaying statistics on exposure to pollutions, hourly, daily, during daytime and nighttime, for both noise and air.
- Displaying maps with own's noise measurements and/or hourly NO₂ levels (including street-level resolution for Paris and the San Francisco Bay area)
- Promoting citizen science where communities of users may engage into the intensive measurement of noise in order to analyze a given journey or to map neighborhoods.
- Providing pollution-aware routing.
- Calibration of the smartphone for noise monitoring, automatically with Ambiciti database or manually with a sound level meter,

Privacy: The Ambiciti app has been designed with privacy in mind, which especially holds in the case of noise pollution monitoring. It is in particular important to stress that actual sound samples are never stored on the phone or uploaded to the Ambiciti servers. Only the amplitude of the sound in dB(A) is calculated, and uploaded provided the user's permission. An anonymous identifier is further created for each device for distinguishing between the data sent by different users, while no identifying information is collected. Further detail may be found from the App information page.

The Ambiciti platform is developed in collaboration with the Inria CLIME team together with The Civic Engine and the NUMTECH SMEs in the context of CityLab@Inria and Inria@SiliconValley, and with the support of the EIT Digital Env&You activity.

6.4. AppCivist-PB: A Platform for Democratic Assembly Customized for Participatory Budgeting

Participants: Valérie Issarny [contact], Cristhian Parra Trepowski, Rafael Angarita.

Participatory budgeting processes are among the most illustrative, real-life experiences of participatory democracy. Participatory Budgeting (PB) has its beginnings in the late 1980s, when some Brazilian cities started to experiment with processes of citizen participation in decisions about how to better allocate part of the city's budget. Although PB takes different forms, they can all be considered as refining the following base process: residents of a city propose spending ideas, volunteers or delegates develop those ideas into proposals, residents then vote on the proposals, and the government finally implements the winning projects. Since the 1980s, PB processes have spread around the world as a set of administrative reforms and, more recently, as a "best practice" in mainstream international development.

With AppCivist-PB, we want to enable city governments to configure the software assemblies that best match the requirements of the kind of PB campaign they want to support, while leveraging existing software services and components. However, from the overall perspective of participatory democracy, our goal is primarily to facilitate the elaboration of proposals by citizen assemblies that form according to the citizen interests. In other words, we want to support a process that emphasizes collaborative contribution making at all stages of the elaboration of proposals by diverse citizen assemblies, which are primarily created by and for citizens. The collaborative process must in particular facilitate the assembly of groups (or sub-assemblies) on the basis of commonalities among the proposals, which is essential if one wants to sustain city-scale participation and be inclusive of citizen contributions.

AppCivist-PB helps users assemble proposal making and selection workflows, using service-oriented architecture (SOA) principles. The composition principles of SOA allow for various implementations and instances of these workflows, including the possibility of integrating and linking different workflows for the same PB campaign. For example, a city might create and manage its own workflow to receive proposals and facilitate deliberation and voting by registered residents; at the same time, citizen groups (typically activists) can create their own, independent, workflows to co-create, develop, and promote proposals for the city, following their own collaboration practices. Compared to traditional SOA, AppCivist-PB distinguishes itself by enabling the assembly of software services dedicated to the support of online-facilitated participatory democracy by and for relevant citizen assemblies.

The AppCivist-PB platform is developed in collaboration with the Social Apps Labs at CITRIS at University of California Berkeley (USA) in the context of CityLab@Inria and Inria@SiliconValley, together with the support of the EIT Digital CivicBudget activity.

MYRIADS Project-Team

6. New Software and Platforms

6.1. ConPaaS

KEYWORDS: Cloud computing - PaaS

SCIENTIFIC DESCRIPTION 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.

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

6.2. GinFlow

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

FUNCTIONAL DESCRIPTION GinFlow decentralizes the coordination of the execution of workflow-based applications. GinFlow relies 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: Matthieu Simonin, Cédric Tedeschi, Hector Fernandez, Javier Rojas Balderrama and Thierry Priol
- Partner: Université de Rennes 1
- Contact: Cédric Tedeschi
- URL: <http://ginflow.inria.fr>

6.3. Merkat

KEYWORDS: Resource management - Cloud - Elastic scaling - Market mechanisms - Service Level Objectives - HPC

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 [12]. 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 a 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: Stefania Costache, Christine Morin and Nikos Parlavantzas
- Contact: Nikos Parlavantzas
- URL: <http://www.irisa.fr/myriads/software/Merkat/>

6.4. Meryn

KEYWORDS: Resource management - PaaS - Cloud - Market mechanisms - Service Level Agreements

FUNCTIONAL DESCRIPTION

Meryn is an open, SLA-driven PaaS architecture that supports cloud bursting and allows hosting an extensible set of application types. Meryn relies on a decentralized optimization policy that aims at maximizing the overall provider profit, taking into account the penalties incurred when quality guarantees are unsatisfied. Meryn builds on the Snooze VM manager, and currently supports batch and MapReduce applications.

- Participants: Nikos Parlavantzas, Djawida Dib and Christine Morin
- Contact: Nikos Parlavantzas
- URL: <http://www.irisa.fr/myriads/software/Meryn/>

6.5. 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.

- Participants: Nikos Parlavantzas, Arnab Sinha, Manh Linh Pham, and Christine Morin
- Contact: Nikos Parlavantzas
- URL: <https://team.inria.fr/myriads/software-and-platforms/paasage-adapter/>

6.6. Resilin

KEYWORDS: Map Reduce - Parallel processing - Hadoop - Cloud - PaaS

FUNCTIONAL DESCRIPTION

Resilin is an open-source system for creating and managing MapReduce execution platforms over clouds. Resilin is compatible with the Amazon Elastic MapReduce (EMR) API, but it goes beyond Amazon's proprietary EMR solution in allowing users (e.g. companies, scientists) to leverage resources from one or more public and/or private clouds. This enables performing MapReduce computations over a large number of geographically-distributed and diverse resources. Resilin can be deployed across most of the open-source and commercial IaaS cloud management systems (e.g., OpenStack, OpenNebula, Amazon EC2). Once deployed, Resilin takes care of provisioning Hadoop clusters and submitting MapReduce jobs, allowing users to focus on writing their MapReduce applications rather than managing cloud resources. Resilin is implemented in the Python language and uses the Apache Libcloud library to interact with IaaS clouds. Resilin has been evaluated on multiple clusters of the Grid'5000 experimentation testbed. The results show that Resilin enables the use of geographically distributed resources with a limited impact on MapReduce job execution time.

- Participants: Ancuta Iodache, Christine Morin, Pierre Riteau, Nikos Parlavantzas and Matthieu Simonin
- Contact: Christine Morin
- URL: <http://resilin.inria.fr>

6.7. SAIDS

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.

- Participants: Anna Giannakou, Jean-Léon Cusinato, Christine Morin, Jean-Louis Pazat, Louis Rilling and Fergal Martin-Tricot,
- Contact: Christine Morin
- URL: <https://bil.inria.fr>

6.8. 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.

- Participants: Martin Quinson,
- Partners: Frédéric Suter, Arnaud Legrand, Adrien Lèbre, Luka Stanisic, Augustin Degomme.
- Contact: Martin Quinson
- URL: <http://simgrid.gforge.inria.fr/>

6.9. Snooze

KEYWORDS: Energy management - Fault-tolerance - Self-organization - Self-healing - Cloud computing - Consolidation - Virtualization

SCIENTIFIC DESCRIPTION

Snooze is a scalable, resilient and energy-aware virtual machine management framework for clouds. It is the result of Eugen Feller's PhD thesis which has been funded by the ANR EcoGrappe project (2008 - 2012).

The objectives of the Snooze ADT are threefold: (i) to distribute Snooze system as an open source software and to provide support to the user community (ii) to implement additional features to make it more user-friendly (iii) to integrate it with other open source software stacks and in public testbeds to favour its dissemination.

Snooze is a highly modular system for IaaS clouds. For the scientific community, it constitutes a unique framework for the experimentation of resource management policies in a real system. More generally, it allows any organization to operate a large-scale cluster as a resilient and energy-aware computing infrastructure enabling on demand provisioning of virtual clusters.

FUNCTIONAL DESCRIPTION

Snooze is a self-organizing and energy aware Cloud management framework.

Snooze is a novel Infrastructure-as-a-Service (IaaS) cloud-management system, which is designed to scale across many thousands of servers and virtual machines (VMs) while being easy to configure, highly available, and energy efficient. For scalability, Snooze performs distributed VM management based on a hierarchical architecture. To support ease of configuration and high availability Snooze implements self-configuring and self-healing features. Finally, for energy efficiency, Snooze integrates a holistic energy management approach via VM resource (i.e. CPU, memory, network) utilization monitoring, underload/overload detection and mitigation, VM consolidation (by implementing a modified version of the Sercon algorithm [59]), and power management to transition idle servers into a power saving mode. Snooze is a highly modular piece of software. It has been extensively evaluated on the Grid'5000 testbed using realistic applications.

- Participants: Eugen Feller, Christine Morin, Jiajun Cao, Gene Cooperman, Yvon Jégou, David Margery and Matthieu Simonin
- Contact: Christine Morin
- URL: <http://snooze.inria.fr/>

6.10. VEP

KEYWORDS: Cloud - Security - Computing - IaaS - Standards - OVF - CIMI - SLA

SCIENTIFIC DESCRIPTION

Virtual Execution Platform (VEP) is a Contrail service that sits just above IaaS layer at the service provider end of the Contrail cloud federation. The VEP service provides a uniform interface for managing the whole lifecycle of elastic applications on the cloud and hides the details of the IaaS layer to the user. VEP applications are described in OVF (Open Virtualization Format) standard format. Resource usage is controlled by CEE (Constrained Execution Environment) rules which can be derived from SLAs (Service Level Agreement). The VEP service integrates a monitoring system where the major events about the application, mainly resource usage, are made available to the user.

The VEP service provides a RESTful interface and can be exploited directly by users on top of the provider IaaS. OpenNebula and OpenStack IaaS frameworks were initially supported. During the VEP-S EIT ICT Labs activity in 2014, VEP was extended with a new OCCI IaaS driver which allows to control any IaaS framework providing a standard OCCI API. Support for the new OCCI SLA proposition from OGF has also been added and allows to represent the VEP CEEs in a standard format. Finally, during this activity, the Zabbix open source distributed monitoring system was integrated to VEP.

FUNCTIONAL DESCRIPTION VEP is a management tool for IaaS clouds with a REST interface and simple GUI for administrator. It is an extensible and reusable software for easy deployment of distributed applications. It provides advance reservation, pro-active fault tolerance. It is SLA aware and manages elasticity.

- Participants: Yvon Jégou, Roberto Gioacchino Cascella, Florian Dudouet, Filippo Gaudenzi, Christine Morin and Arnab Sinha
- Contact: Christine Morin
- URL: <https://project.inria.fr/vep/>

REGAL Project-Team

5. New Software and Platforms

5.1. Antidote

FUNCTIONAL DESCRIPTION

Antidote is the flexible cloud database platform currently under development in the SyncFree European project. 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 [23], for elastic version management, for adaptive replication, for partial replication, and for reconfigurable sharding.

- Participants: Tyler Crain, Marc Shapiro and Alejandro Tomsic
- Contact: Tyler Crain
- URL: <https://github.com/SyncFree/>

5.2. G-DUR

FUNCTIONAL DESCRIPTION

A large family of distributed transactional protocols have a common structure, called Deferred Update Replication (DUR). DUR provides dependability by replicating data, and performance by not re-executing transactions but only applying their updates. Protocols of the DUR family differ only in behaviors of few generic functions. Based on this insight, we offer a generic DUR middleware, called G-DUR, along with a library of finely-optimized plug-in implementations of the required behaviors.

- Participants: Marc Shapiro and Masoud Saeida Ardekani
- Contact: Marc Shapiro
- URL: <https://github.com/msaeida/jessy>

5.3. NumaGiC

FUNCTIONAL DESCRIPTION

NumaGiC is a version of the HotSpot garbage collector (GC) adapted to many-core computers with very large main memories. In order to maximise GC throughput, it manages the trade-off between memory locality (local scans) and parallelism (work stealing) in a self-balancing manner. Furthermore, the collector features several memory placement heuristics that improve locality.

- Participants: Lokesh Gidra, Marc Shapiro, Julien Sopena and Gaël Thomas
- Contact: Lokesh Gidra
- URL: <http://gforge.inria.fr/projects/transgc/>

SPIRALS Project-Team

6. New Software and Platforms

6.1. APISENSE

KEYWORDS: Mobile sensing - Crowd-sensing - 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: Nicolas Haderer, Romain Rouvoy, Christophe Ribeiro, Julien Duribreux and Antoine Veuiller
- Partner: Université Lille 1
- Contact: Romain Rouvoy
- URL: <http://www.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

KEYWORD: 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: Romain Rouvoy, Adel Nouredine, Loic Huertas and Maxime Colmant
- Contact: Romain Rouvoy
- URL: <http://www.powerapi.org>

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

6.5. 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: Clement 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>

WHISPER Project-Team

6. New Software and Platforms

6.1. Prequel

KEYWORDS: Code quality - Evolution - Infrastructure software

FUNCTIONAL DESCRIPTION

The commit history of a large, actively developed 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 imply 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 have developed Prequel (Patch Query Language) [20]. Prequel builds on the semantic patch language SmPL developed for Coccinelle, which is now well known to the Linux kernel developer community, to allow developers to scan the changes in a source code development history, taking into account not only the specific changes made, but also the context in which these changes occur. As the history of a code base under active development quickly becomes large, with the Linux kernel incorporating around 13,000 commits on each 2-3 month release cycle, a particular goal in the development of Prequel has been to provide reasonable performance. Currently, most queries in our experiments complete in under minute when running on a single core on a standard laptop. So far, we have applied Prequel to the problem of understanding how to eliminate uses of deprecated functions [20], and are investigating how it may be useful in a systematic driver porting methodology.

Prequel is publicly available under GPLv2. The development of Prequel is supported by OSADL, and Julia Lawall presented Prequel at the 2016 OSADL networking day (<https://www.osadl.org/OSADL-Networking-Day-2016.networking-day-2016.0.html>).

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

6.2. Coccinelle

KEYWORDS: Code quality - Evolution - Infrastructure software

FUNCTIONAL DESCRIPTION

Coccinelle is a tool for C code program matching and transformation that has been developed by members of the Whisper team over the last 10 years [8]. Coccinelle is widely used by the Linux kernel developer community and for other C software projects. Over the last three years, Coccinelle has benefited from the support of an engineer from the SED. Major improvements in 2016 include support for Python 3, independence from a no-longer-supported interface between Python and OCaml, better support for parallelism, and better support for integrating arbitrary predicates into the matching process. These features significantly improve performance and improve the uniformity of the rule specification language, thus providing a better experience for users. Coccinelle is at the foundation of much of our research work, including the ANR ITrans project, and these improvements will enhance and facilitate our research, accordingly.

Coccinelle is publicly available under GPLv2. In 2016, Julia Lawall presented Coccinelle in an invited keynote at the Linux Security Summit (<http://events.linuxfoundation.org/events/archive/2016/linux-security-summit>) and at a “birds of a feather” session at Linuxcon Europe (<http://events.linuxfoundation.org/events/LinuxCon-europe>).

- Participants: Julia Lawall, Gilles Muller, and Thierry Martinez
- Partners: IRILL - LIP6
- Contact: Julia Lawall
- URL: <http://coccinelle.lip6.fr>

6.3. Hector (BtrLinux)

KEYWORDS: Code quality - Evolution - Infrastructure software

FUNCTIONAL DESCRIPTION

A major source of errors in systems code is resource-release omission, which can lead to memory leaks and to crashes, if the system ends up in an inconsistent state. Currently, many tools exist that detect common patterns in software and detect faults as deviations from those patterns, but most suffer from high rates of false positives. Hector takes the novel approach of detecting inconsistencies local to a single function, and thus has been able to find over 300 faults in Linux kernel code and other C infrastructure software, with a rate of false positives of only 23%. Hector was originally the subject of the PhD thesis of Suman Saha [75]. Over the past two years, improving the robustness of the implementation of Hector has been the focus of ADT (young engineer position) BtrLinux supported by Inria, with the goal of making Hector publicly available and popularizing its use in the Linux kernel developer community. Some Linux kernel patches based on the use of Hector have been integrated into the Linux kernel, and the public release of Hector is in progress. The ADT position also involved the creation and maintenance of the website <https://btrlinux.inria.fr/> as a showcase for the work of the Whisper team around Linux kernel development tools.

Building on his experience acquired in the ADT position, Quentin Lambert has recently been offered a position as an engineer at Wolfram MathCore AB.

- Participants: Quentin Lambert, Julia Lawall, and Gilles Muller
- Partners: IRILL - LIP6
- Contact: Julia Lawall
- URL: <https://btrlinux.inria.fr/>

6.4. ssrbit

FUNCTIONAL DESCRIPTION

ssrbit is a Coq library offering an efficient formalization of bit vectors, a refinement framework for abstractly reasoning about bitsets, and a trustworthy extraction of bit vectors to OCaml integers. Initially developed by Whisper members (Pierre-Évariste Dagand, Julia Lawall), the development has attracted an external contributor (Emilio Jesús Gallego Arias, postdoctoral researcher in CRI Mines-ParisTech), which led to significant improvements. We plan to improve the overall support and documentation so as to provide a full-featured library.

- Participants: Pierre Évariste Dagand, Julia Lawall, and Emilio Jesús Gallego Arias
- Contact: Pierre Évariste Dagand
- URL: <https://github.com/ejgallego/ssrbit/>

ALPINES Project-Team

6. New Software and Platforms

6.1. BFD

Block Filtering Decomposition preconditioner

KEYWORDS: Preconditioner - Linear system

FUNCTIONAL DESCRIPTION

Iterative methods are used in many industrial and academic applications to solve large sparse linear systems of equations, and preconditioning these methods is often necessary to accelerate their convergence. Several highly used preconditioners as incomplete LU factorizations are known to have scalability problems, often due to the presence of several low frequency modes that hinder the convergence of the iterative method. To address this problem, we work on filtering preconditioners. A judicious choice of the filtering vector allows to alleviate the effect of low frequency modes, and can accelerate significantly the convergence of the iterative method.

- Participants: Laura Grigori, Remi Lacroix and Frédéric Nataf
- Partners: CNRS - UPMC
- Contact: Laura Grigori
- URL: <https://who.rocq.inria.fr/Laura.Grigori/>

6.2. CALU : communication optimal algorithms for linear algebra

KEYWORDS: Communication avoiding - Linear algebra

FUNCTIONAL DESCRIPTION

CALU solves linear systems of equations $Ax=b$ using Communication Avoiding LU.

- Contact: Laura Grigori
- URL: <https://who.rocq.inria.fr/Laura.Grigori/>

6.3. DPREPack

KEYWORD: Large scale

FUNCTIONAL DESCRIPTION

This library solves linear systems on parallel computers from PCs based on multicore processors to large scale computers. It implements recent parallel algorithms issued from domain decomposition methods and parallel approximate factorizations.

- Partners: CNRS - UPMC
- Contact: Laura Grigori
- URL: <https://team.inria.fr/alpines/>

6.4. FreeFem++

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 2^d -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: Frédéric Hecht
- URL: <http://www.freefem.org/ff++/>

6.5. 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: Pierre Jolivet and Frédéric Nataf
- Contact: Pierre Jolivet
- URL: <https://github.com/hpddm>

6.6. LORASC

LORASC preconditioner

KEYWORD: Preconditioner

- Participants: Laura Grigori and Remi Lacroix
- Contact: Laura Grigori
- URL: not available

6.7. NFF

NFF Nested Filtering Factorization

KEYWORDS: Preconditioner - Interactive method - Linear system

- Participants: Laura Grigori, Frédéric Nataf and Long Qu
- Partners: Université Paris-Sud - UPMC
- Contact: Laura Grigori
- URL: not available

6.8. SparseToolbox

KEYWORDS: Preconditioner - Interactive method - Linear system

- Participants: Laura Grigori and Remi Lacroix
- Contact: Laura Grigori
- URL: not available

AVALON Project-Team

5. New Software and Platforms

5.1. Active Data

Participants: Gilles Fedak [correspondant], Valentin Lorentz, Laurent Lefevre.

FUNCTIONAL DESCRIPTION

Active Data is a free software system that tracks the life cycle of data distributed across heterogeneous software and infrastructures.

As the volume of data grows exponentially, the management of these data becomes more complex in proportion. A key point is to handle the complexity of the Data Life Cycle, *i.e.*, the various operations performed on data: transfer, archiving, replication, deletion, etc. Indeed, data-intensive applications span over a large variety of devices and e-infrastructures which implies that many systems are involved in data management and processing. Active Data is a new approach to automate and improve the expressiveness of data management applications. Active Data consists of a formal model that captures the essential data life cycle stages and properties: creation, deletion, replication, derivation, transient unavailability, uniform naming, and many more. Active Data provides a programming model that simplify the development of data life cycle management applications. Active Data allows code execution at each stage of the data life cycle: routines provided by programmers are executed when a set of events (creation, replication, transfer, deletion) happen to any data.

- URL: <http://active-data.gforge.inria.fr>

5.2. DIET

Participants: Daniel Balouek-Thomert, Yves Caniou, Eddy Caron, Arnaud Lefray.

FUNCTIONAL DESCRIPTION

DIET (Distributed Interactive Engineering Toolbox) is a middleware designed for high-performance computing in a heterogeneous and distributed environment (workstations, clusters, grids, clouds). Three main developments are done in 2016:

Proxmox support: DIET has a new Cloud extension that can be used to interact with this PVE (Proxmox Virtual Environment) solution. Proxmox is a complete open source server virtualization management software, based on KVM virtualization and container-based virtualization. It manages KVM virtual machines, Linux containers (LXC), storage, virtualized networks, and HA clusters. This extension is a contribution provided by the NewGeneration-SR company.

DaaS support: The goal of this development was to add Data-as-a-Service (DaaS) functionality to the DIET middleware via an optional module. We have added the GDConnect plug-in to the DIET Data manager that provides the capability to deal with Google's Cloud Storage.

Energy Support: We designed, implementation and evaluation of an energy-efficient resource management system that builds upon DIET and NSDE-divisible tasks with precedence constraints.

- Partners: CNRS - ENS Lyon - UCBL Lyon 1
- Contact: Eddy Caron
- URL: <http://graal.ens-lyon.fr/diet/>

5.3. Execo

Participant: Matthieu Imbert.

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 Grid'5000 testbed. The execo_engine package provides tools to ease the development of computer sciences experiments. Execo is used directly by 15 to 30 users, in and out of the Avalon team, and is also used by a few Grid'5000 tools such as kwapi, funk, topo5k, g5k_bench_flops.

- Contact: Matthieu Imbert
- URL: <http://execo.gforge.inria.fr>

5.4. Kaapi

Participant: Thierry Gautier.

Kaapi is a library for high performance applications running on multi-cores/multi-processors with support for multi-GPUs. Kaapi provides ABI compliant implementations of libGOMP (GCC runtime for OpenMP) and libomp.so (CLANG and Intel compiler). It was one of the target runtime of the K'Star compiler (<http://kstar.gforge.inria.fr>).

Web site:

- Partners: EPI Corse (Philippe Virouleau, François Broquedis)
- Contact: Thierry Gautier
- URL: <http://kaapi.gforge.inria.fr>

5.5. Kwapi

Participants: Jean-Patrick Gelas, Laurent Lefevre.

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 Grid'5000 resources.

- Contact: Laurent Lefevre
- URL: <https://launchpad.net/kwapi>

5.6. L2C and DirectL2C

Participants: H el ene Coullon, Vincent Lanore, Christian Perez, J er ome Richard.

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.

DirectL2C is an extension to L2C that enables efficient and consistent reconfiguration of large scale L2C based assemblies. It provides an assembly model enhanced with domains, transformations, and transformation adapters.

- Partners: CEA/Maison de la Simulation (Julien Bigot)
- Contact: Christian Perez
- URL: <http://hlcm.gforge.inria.fr/l2c:start>

5.7. Sam4C

Participants: Eddy Caron, Arnaud Lefray, Marc Pinhede, Mathieu Veyrand.

SCIENTIFIC DESCRIPTION

This editor is generated in Java from an EMF -Eclipse Modeling Framework- metamodel 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.

- Contact: Eddy Caron
- URL: <https://gforge.inria.fr/projects/sam4c/>

5.8. SimGrid

Participant: Frédéric Suter.

FUNCTIONAL 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.

- Partners: CNRS - ENS Rennes
- Contact: Martin Quinson (EPC Myriads)
- URL: <http://simgrid.gforge.inria.fr/>

5.9. Grid'5000

Participants: Laurent Lefevre, 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 Grid5000 Lyon site.

- Contact: Laurent Lefevre
- URL: <https://www.grid5000.fr/>

DATAMOVE Team

5. New Software and Platforms

5.1. OAR

KEYWORDS: HPC - Cloud - Clusters - Resource manager - 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).

The OAR ecosystem also include several associated software tools that proved to be useful independently from OAR. Among theses, two softwares play a major role in the support our research studies. The first one is Kameleon (<http://kameleon.imag.fr>), a tool to help enhancing reproducibility of experiments by guarantee the ability to reproduce the complete used software stacks. The second one is Batsim (<https://gforge.inria.fr/projects/batsim>) a RJMS simulator based on SimGrid. Batsim simulates job execution taking into account the target platform hardware capabilities through SimGrid, while scheduling is performed by an actual job management system. A comprehensive API enables to easily plug into BatSim various job management systems like OAR.

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

5.2. FlowVR

KEYWORDS: HPC - In Situ Processing - Computational Steering

SCIENTIFIC DESCRIPTION FlowVR is an open source middleware to augment parallel simulations running on thousands of cores with in situ processing capabilities and live steering. FlowVR offers a very flexible environment while enabling high performance asynchronous in situ and in transit processing.

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, Matthieu Dreher, Jérémy Jaussaud
- Contact: Bruno Raffin
- URL: <http://flowvr.sf.net>

5.3. Platforms

5.3.1. *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. Spack-morse

The radical change we have adopted in terms of methodology (task-based programming strongly) changes the software design. In particular, our codes become more and more modular and the complexity of their inter-dependencies is subsequently very high.

In order to address this complexity we have chosen to rely on the **Spack** flexible package manager designed to support multiple versions, configurations, platforms, and compilers (<http://software.llnl.gov/spack> developed and maintained at LLNL). We have integrated all our libraries above this package manager in the **Spack-Morse** extension that we maintain in **HIEPACS**.

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-3 (original software reusing known ideas and introducing new ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-3 (good quality middle-term maintenance).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web). source distribution or a commercially-distributed product).
- Contact: Florent Pruvost
- URL: <http://morse.gforge.inria.fr/spack/spack.html>

6.2. Chameleon

Chameleon is a dense linear algebra software relying on the STF sequential task-based programming paradigm. It implements the tile algorithms originally designed for multicore architectures in the **PLASMA** package and extends them so that they can be processed on by a runtime system to exploit any type of hardware architecture (multicore, GPU, heterogeneous, supercomputer). This software is central for the team as it allows to investigate in a relatively simple context (regular dense linear algebra algorithms) new types of designs before implementing them for the more irregular algorithms implemented in the software packages described below.

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-3 (good quality middle-term maintenance).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web). source distribution or a commercially-distributed product).
- Contact: Emmanuel Agullo
- URL: <https://project.inria.fr/chameleon>

6.3. HIPS

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

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 sub-graphs viewed as sub-domains) is solved using an iterative method (GMRES or Conjugate Gradient at the time being).

Thus, **HIPS** is a software library that provides several methods to build an efficient preconditioner in almost all situations.

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-2 (basic maintenance to keep the software alive).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Pierre Ramet
- URL: <http://hips.gforge.inria.fr>

6.4. MaPHYS

MaPHyS (Massively Parallel Hybrid Solver) is an hybrid iterative/direct parallel (MPI-threads) sparse linear solver based on algebraic domain decomposition technique for real/complex symmetric positive definite/unsym/-metric matrices. For a given number of MPI processes/domains, **MaPHyS** solves the Schur complement computed from the adjacency graph of the sparse matrix using a preconditioned Krylov subspace method (CG or GMRES). The provided preconditioners are variants of an algebraic Additive Schwarz methods. A prototype version of a two level preconditionner using an algebraic coarse space is available but not yet publicly distributed (provided upon request for beta testers).

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-4 (well-defined and implemented plans for future maintenance and evolution).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Emmanuel Agullo
- URL: <https://project.inria.fr/maphys/fr>

6.5. MetaPart

MetaPart is a library that addresses the challenge of (dynamic) load balancing for emerging complex parallel simulations, such as multi-physics or multi-scale coupling applications. First, it offers a uniform API over state-of-the-art (hyper-) graph partitioning & ordering software packages such as **Scotch**, PaToH, METIS, Zoltan, Mondriaan, etc. Based upon this API, it provides a framework that facilitates the development and the evaluation of high-level partitioning methods, such as MxN repartitioning or coupling-aware partitioning (co-partitioning).

- Audience: A-1 (internal prototype).
- Software originality: SO-3 (original software reusing known ideas and introducing new ideas).
- Software maturity: SM-2 (basic usage works, terse documentation).
- Evolution and maintenance: EM-3 (good quality middle-term maintenance).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Aurélien Esnard
- URL: <http://metapart.gforge.inria.fr>

6.6. PaStiX

PaStiX (Parallel Sparse matriX package) 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) iterative methods. Numerical algorithms are implemented in single or double precision (real or complex): LLt (Cholesky), LDLt (Crout) and LU with static pivoting (for non symmetric matrices having a symmetric pattern).

The **PaStiX** solver is suitable for any heterogeneous parallel/distributed architecture when its performance is predictable, such as clusters of multicore nodes. In particular, we now offer 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.

- Audience: A-5 (wide audience, large user's community).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-4 (major software project, strong software engineering).
- Evolution and maintenance: EM-4 (well-defined and implemented plans for future maintenance and evolution).
- Software distribution and licensing: SDL-5 (external packaging and distribution, as part of a popular open source distribution or a commercially-distributed product).
- Contact: Pierre Ramet
- URL: <http://pastix.gforge.inria.fr>

6.7. QR_Mumps

qr_mumps is a software package for the solution of sparse, linear systems on multicore computers. It implements a direct solution method based on the QR factorization of the input matrix. Therefore, it is suited to solving sparse least-squares problems and to computing the minimum-norm solution of sparse, under-determined problems. It can obviously be used for solving square problems in which case the stability provided by the use of orthogonal transformations comes at the cost of a higher operation count with respect to solvers based on, e.g., the LU factorization. **qr_mumps** supports real and complex, single or double precision arithmetic.

qr_mumps is mainly developed and maintained by the APO team of the IRIT laboratory of Toulouse. **HIEPACS** is an active contributor to this project.

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-3 (good quality middle-term maintenance).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Emmanuel Agullo
- URL: http://buttari.perso.enseeiht.fr/qr_mumps/

6.8. ScalFMM

ScalFMM is a library to compute N-body interactions using the Fast Multipole Method. This is a parallel kernel independent fast multipole method based on interpolation (Chebychev or equispaced grid points).

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 (fork-join and tasks models), MPI and runtime system (**StarPU**). 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 tree structure (hierarchical octree and Group-Tree) and the parallel algorithms ;
- the kernels (scalar, tensorial and multi-rhs). Classical kernels are available (Coulombic, Leonard-Jones, Gaussian, Stokes, ...)

This modular architecture allows us to easily add new FMM algorithms or kernels and new paradigm of parallelization. Today, we also proposed the FMM based on spherical harmonic expansion with Blas or rotation optimization for Coulombic potential) and all algorithms are designed to treat more complex kernels by adding multiple right-hand sides, tensorial structures, ...

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-4 (original software implementing a fair number of original ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-3 (good quality middle-term maintenance).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Olivier Coulaud
- URL: <http://scalfmm-public.gforge.inria.fr/doc/>

6.9. ViTE

ViTE is a trace explorer. It is a tool to visualize execution traces in Pajé or OTF format for debugging and profiling parallel or distributed applications. It is developed with C++ programming language with OpenGL and Qt technologies.

- Audience: A-4 (large audience, used by people outside the team).
- Software originality: SO-3 (original software reusing known ideas and introducing new ideas).
- Software maturity: SM-3 (well-developed software, good documentation, reasonable software engineering).
- Evolution and maintenance: EM-2 (basic maintenance to keep the software alive).
- Software distribution and licensing: SDL-4 (public source or binary distribution on the Web).
- Contact: Mathieu Faverge
- URL: <http://vite.gforge.inria.fr>

6.10. Platforms

6.10.1. PlaFRIM: Plateforme Fédérative pour la Recherche en Informatique et Mathématiques

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

6. New Software and Platforms

6.1. Týr

Title: Týr: Blob Storage Meets Built-In Transactions.

Keywords: Big Data; Transactions; Týr; BlobSeer.

Scientific Description: Týr [25] is the first blob storage system to provide built-in, multi-blob transactions, while retaining sequential consistency and high throughput under heavy access concurrency.

Functional Description: Týr offers fine-grained random write access to data and in-place atomic operations. Large-scale experiments on Microsoft Azure with a production application from CERN LHC show Týr throughput outperforms state-of-the-art solutions by more than 75%. Týr leverages the approaches developed within BlobSeer, the reference data management system for large distributed blobs, developed over the past years in KerData.

Contact data:

Participants: Pierre Matri, Alexandru Costan and Gabriel Antoniu.

Partners: INSA Rennes, Universidad Politécnica de Madrid.

Contact: Gabriel Antoniu.

URL: <http://tyr.io/>.

6.2. Damaris

Title: Damaris: I/O and data management for large-scale, MPI-based HPC simulations.

Keywords: I/O; HPC; Data management; Visualization; Big Data; 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 **JLESC**. 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 **JLESC**. Damaris now has external users, including (to our knowledge) visualization specialists from NCSA, Big Data experts from the HDF group, 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 four large-scale simulations (CM1, OLAM, Nek5000, CROCO). Works are in progress to evaluate it in the context of several other simulation codes including HACC (cosmology) and GTC (fusion).

Damaris is the object of a *Technical Development Action* (ADT) supported by Inria.

Functional Description: Damaris targets 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.

Contact data:

Participants: **Matthieu Dorier** (ANL), Lokman Rahmani, Gabriel Antoniu, Orçun Yildiz, Hadi Salimi and Luc Bougé.

Partners: ENS Rennes, Argonne National Laboratory.

Contact: Gabriel Antoniu.

URL: <http://damaris.gforge.inria.fr/>.

6.3. Other software

6.3.1. JetStream

Title: JetStream: Enabling High-Performance Event Streaming across Cloud Data-Centers.

Keywords: Big Data, streaming, data transfer, multisite cloud.

Scientific Description. JetStream is a middleware solution for batch-based, high-performance streaming across cloud data centers. JetStream implements a set of context-aware strategies to optimize batch-based streaming, being able to self-adapt to changing conditions.

Functional Description. The system provides multi-route 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: Ovidiu-Cristian Marcu, Alexandru Costan, Gabriel Antoniu.

Contact: Alexandru Costan.

6.3.2. Omnisc'IO

Title: Omnisc'IO: a Grammar-Based Approach to Spatial and Temporal I/O Patterns Prediction.

Keywords: HPC, Input-Output, Prediction, Grammar.

Scientific Description. Omnisc'IO is a library that aims to be integrated into I/O middleware.

Functional Description. It traces I/O operations, models the stream of such operations using grammar-inference techniques, and predicts when new I/O operations will be performed, as well as where and how much data will be written.

Participants: **Matthieu Dorier** (ANL), Gabriel Antoniu, Shadi Ibrahim.

Contact: Gabriel Antoniu.

6.3.3. OverFlow

Title: OverFlow: Workflow Data Management as a Service for Multi-Site Applications.

Keywords: Small data; workflow; multi-site cloud.

Scientific 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 performance for transfer cost and time, within and across sites.

Functional Description. 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 geo-replication, 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: Paul Le Noac'h, Ovidiu-Cristian Marcu, Alexandru Costan and Gabriel Antoniu.

Contact: Alexandru Costan.

6.3.4. iHadoop

Title: iHadoop: A Hadoop Simulator Developed In Java on Top of SimGrid.

Keywords: Simulation, Map-Reduce, Hadoop, SimGrid.

Scientific Description. iHadoop is a Hadoop simulator developed in Java on top of SimGrid. It simulates the behavior of Hadoop and therefore accurately predicts the performance of Hadoop in normal scenarios and under failures. iHadoop is extended to (1) simulate the execution and predict the performance of multiple Map-Reduce applications; (2) simulate the execution of Map-Reduce applications under various data distributions and data skew models.

Functional Description. iHadoop is an internal software prototype, which was initially developed to validate our idea regarding the behavior of Hadoop under failures. iHadoop has preliminarily evaluated within our group and it has shown very high accuracy to predict the execution time of a Map-Reduce applications. 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.

POLARIS Team

5. New Software and Platforms

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

5.2. 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 aim of GameSeer is a) to provide a numerical integration kernel for phase portrait and equilibrium set generation; and b) to provide a graphical user interface that allows the user to employ said capabilities from a simple and intuitive front-end.

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

5.3. Moca

Memory Organization Cartography and Analysis

MOCA is an efficient tool for the collection of complete spatiotemporal memory traces. Its objective is twofold, namely to *a*) avoid missuses of the memory hierarchy (such as false sharing of cache lines or contention); and *b*) to take advantage of the various cache levels and the memory hardware prefetcher. It is based on a Linux kernel module and provides a coarse-grained trace of a superset of all the memory accesses performed by an application over its addressing space during the time of its execution.

KEYWORDS: High-Performance Computing - Performance analysis

- Contact: Guillaume Huard
- URL: <https://github.com/dbeniamine/MOCA>

5.4. 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/>

5.5. 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/>

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.

- Participants: Frederic Suter, Martin Quinson, Arnaud Legrand, Adrien Lebre, Jonathan Pastor, Mario Sudholt, Luka Stanisic, Augustin Degomme, Jean-Marc Vincent, Florence Perronnin and Jonathan Rouzaud-Cornabas
- Partners: CNRS - ENS Rennes
- Contact: Martin Quinson
- URL: <http://simgrid.gforge.inria.fr/>

5.7. 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/>

5.8. 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, Issam Rabhi, Jean-Marc Vincent, Benjamin Briot, Jean-Michel Fourneau and Franck Quessette
- Partner: UVSQ
- Contact: Alain Jean-Marie
- URL: <http://marmotecore.gforge.inria.fr/>

ROMA Project-Team

6. New Software and Platforms

6.1. 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(ENSEEIH)-IRIT, Inria, CERFACS, University of Bordeaux, CNRS and ENS Lyon. In 2014, a consortium of industrial users has been created (<http://mumps-consortium.org>).

- Participants: Patrick Amestoy, Alfredo Buttari, Jean-Yves L'Excellent, Chiara Puglisi, Mohamed Sid-Lakhdar, Bora Uçar, Marie Durand, Abdou Guermouche, Maurice Bremond, Guillaume Joslin, Stéphane Pralet, Aurélia Fevre, Clément Weisbecker, Theo Mary, Emmanuel Agullo, Jacko Koster, Tzvetomila Slavova, François-Henry Rouet, Philippe Combes and Gilles Moreau
- Partners: CERFACS - CNRS - ENS Lyon - INPT - IRIT - LIP - Université de Bordeaux - Université de Lyon - Université de Toulouse
- Latest public release: MUMPS 5.0.2 (July 2016)
- Contact: Jean-Yves L'Excellent
- URL: <http://mumps-solver.org/>
- Next MUMPS User Days: we have started organizing the next MUMPS User days, which will be hosted by Inria on June 1 and 2, 2017 near Grenoble, France (see http://mumps.enseeiht.fr/ud_2017.php)

In the context of the MUMPS consortium (see Section 8.1 and <http://mumps-consortium.org>), we had scientific exchanges and collaborations with industrial members and released two versions in advance for the consortium (in July 2016 and November 2016), containing major improvements for large-scale problems and many other improvements. Much effort was also put on developing features and algorithms to improve the quality and performance of MUMPS, especially in the context of problems offering potential for low-rank compression. This work is done in close collaboration with the partners who co-develop MUMPS, in particular in Toulouse.

6.2. 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: Christophe Alias and Alexandru Plesco (XtremLogic SAS)
- Contact: Christophe Alias

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

6.4. Aspic

Accelerated Symbolic Polyhedral Invariant Generation

KEYWORDS: Abstract Interpretation - Invariant Generation

FUNCTIONAL DESCRIPTION

Aspic is an invariant generator for general counter automata. Used with C2fsm (a tool developed by P. Feautrier in COMPSYS), it can be used to derivate invariants for numerical C programs, and also to prove safety. It is also part of the WTC toolsuite (see <http://compsys-tools.ens-lyon.fr/wtc/index.html>), a tool chain to compute worst-case time complexity of a given sequential program.

Aspic implements the theoretical results of Laure Gonnord's PhD thesis on acceleration techniques and has been maintained since 2007.

- Participant: Laure Gonnord
- Contact: Laure Gonnord
- URL: <http://laure.gonnord.org/pro/aspic/aspic.html>

6.5. Termite

Termination of C programs

KEYWORDS: Abstract Interpretation - Termination

FUNCTIONAL DESCRIPTION

TERMITE is the implementation of the algorithm "Counter-example based generation of ranking functions". Based on LLVM and Pagai (a tool that generates invariants), the tool automatically generates a ranking function for each *head of loop*.

TERMITE represents 3000 lines of OCaml and is now available via the opam installer.

- Participants: Laure Gonnord, Gabriel Radanne (PPS, Univ Paris 7), David Monniaux (CNRS/Verimag).
- Contact: Laure Gonnord
- URL: <https://termite-analyser.github.io/>

6.6. Vaphor

Validation of C programs with arrays with Horn Clauses

KEYWORDS: Abstract Interpretation - Safety - Array Programs

FUNCTIONAL DESCRIPTION

VAPHOR (Validation of Programs with Horn Clauses) is the implementation of the algorithm “An encoding of array verification problems into array-free Horn clauses”. The tool implements a traduction from a C-like imperative language into Horn clauses in the SMT-lib Format.

VAPHOR represents 2000 lines of OCaml and its development is under consolidation.

- Participants: Laure Gonnord, David Monniaux (CNRS/Verimag).
- Contact: Laure Gonnord
- Demo page : <http://laure.gonnord.org/pro/demopage/vaphor/index.php>

STORM Team

5. New Software and Platforms

5.1. Chameleon

KEYWORDS: HPC - Dense linear algebra - Task-based algorithm - Runtime system - 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 memory banks, performing data prefetching, etc. Expected advances: In a nutshell, a new runtime system API 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.

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).

- Participants: Emmanuel Agullo, Mathieu Faverge, Cédric Castagnede and Florent Pruvost
- 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. KLANG-OMP

The KStar OpenMP Compiler

KEYWORDS: Compilers - Task scheduling - OpenMP - Source-to-source compiler - Data parallelism

FUNCTIONAL DESCRIPTION

The Klang-Omp compiler, now renamed KStar following the recommendation of the local experimentation and development service, 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: Olivier Aumage, Nathalie Furmento, Samuel Pitoiset and Samuel Thibault
- Contact: Olivier Aumage
- URL: <http://kstar.gforge.inria.fr/#!index.md>

5.3. KaStORS

The KaStORS OpenMP Benchmark Suite

KEYWORDS: Benchmarking - HPC - Task-based algorithm - Task scheduling - OpenMP - 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: Olivier Aumage, François Broquedis, Pierrick Brunet, Nathalie Furmento, Thierry Gautier, Samuel Thibault and Philippe Virouleau
- Contact: Thierry Gautier
- URL: <http://kastors.gforge.inria.fr/#!index.md>

5.4. MORSE

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

5.5. AFF3CT

A Fast Forward Error Correction Tool (previously named P-Edge).

KEYWORDS: Code generation - Error Correction Code

FUNCTIONAL DESCRIPTION

The AFF3CT library joins genericity techniques together with code generation capabilities to enable implementing efficient and portable error correction codes. The genericity offered allows to easily experiment with a large panel of algorithmic variants.

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

5.6. StarPU

The StarPU Runtime System

KEYWORDS: HPC - Scheduling - GPU - Multicore - 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: Cédric Augonnet, Samuel Thibault, Nathalie Furmento, Simon Archipoff, Jérôme Clet-Ortega, Nicolas Collin, Ludovic Courtes, Mehdi Juhour, Xavier Lacoste, Benoît Lize, Ludovic Stordeur, Cyril Roelandt, Corentin Salingue, Chiheb Sakka, Samuel Pitoiset, François Tessier, Pierre-André Wacrenier, Andra Hugo, Terry Cojean, Raymond Namyst, Olivier Aumage and Marc Sergent
- Contact: Olivier Aumage
- URL: <http://starpu.gforge.inria.fr/>

5.7. hwloc

Hardware Locality

KEYWORDS: HPC - Topology - Open MPI - Affinities - GPU - Multicore - NUMA - 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: AMD - Intel - Open MPI consortium
- Contact: Brice Goglin
- URL: <http://www.open-mpi.org/projects/hwloc/>

TADAAM Team

6. New Software and Platforms

6.1. NetLoc

Network Locality

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 application 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. NETLOC is distributed within HWLOC releases starting with HWLOC 2.0.

- Participants: Cyril Bordage and Brice Goglin
- Contact: Brice Goglin
- URL: <http://www.open-mpi.org/projects/netloc/>

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

6.3. 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 43 million elements to more than 1 billion elements 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.

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

6.4. SCOTCH

KEYWORDS: High-performance computing - Graph algorithms - Domain decomposition - Static mapping - Mesh partitioning - Sparse matrix ordering

SCIENTIFIC DESCRIPTION

SCOTCH is a software package and libraries for sequential and parallel graph partitioning, static mapping and clustering; sequential mesh and hypergraph partitioning; and sequential and parallel sparse matrix block ordering.

Its main use is to subdivide a scientific problem, expressed as a graph, into a set of subproblems as independent as possible from each other (in terms of connecting edges).

FUNCTIONAL DESCRIPTION

SCOTCH takes the form of a set of libraries, plus additional standalone programs. The sequential and parallel libraries provide a set of interfaces to describe centralized and distributed graphs to partition, the target architectures to map onto, the resulting centralized and distributed mapping and ordering structures, etc. SCOTCH takes advantage of Posix threads, and its parallel version, PT-SCOTCH, uses the MPI interface.

- Participants: François Pellegrini, Cédric Lachat, Rémi Barat and Cédric Chevalier
- Partners: CNRS - IPB - Region Aquitaine
- Contact: François Pellegrini
- URL: <http://www.labri.fr/~pelegrin/scotch/>

6.5. TreeMatch

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

SCIENTIFIC DESCRIPTION

TreeMatch provides a permutation of the processes to the 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.

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

6.6. hwloc

Hardware Locality

KEYWORDS: HPC - Topology - Open MPI - Affinities - GPU - Multicore - NUMA - 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: AMD - Intel - Open MPI consortium
- Contact: Brice Goglin
- URL: <http://www.open-mpi.org/projects/hwloc/>

ASCOLA Project-Team

6. New Software and Platforms

6.1. CSLA

Cloud Service Level Agreement language

KEYWORDS: Cloud computing - Service-level agreement - Elasticity

FUNCTIONAL DESCRIPTION

CSLA, the Cloud Service Level Agreement language, allows the definition of SLA properties for arbitrary Cloud services (XaaS). CSLA addresses QoS uncertainty in unpredictable and dynamic environment and provides a cost model of Cloud computing. Besides the standard formal definition of contracts – comprising validity, parties, services definition and guarantees/violations – CSLA is enriched with features, such as QoS degradation and an advanced penalty model, thus introducing fine-grained language support for Cloud elasticity management.

- Participants: Thomas Ledoux and Yousri Kouki
- Contact: Thomas Ledoux
- URL: <http://www.emn.fr/z-info/csla/>

6.2. CSQL

Cryptographic Composition for Query Language

SCIENTIFIC DESCRIPTION

C2QL is a compositional language of security techniques for information privacy in the cloud. A cloud service can use security techniques to ensure information privacy. These techniques protect privacy by converting the client's personal data into unintelligible text. But they also cause the loss of some functionalities of the service. As a solution, CSQL permits to compose security techniques to ensure information privacy without the loss of functionalities. But, the composition makes the writing of programs more intricate. To help the programmer, C2QL defines a query language for the definition of cloud services that enforces information privacy with the composition of security techniques. This language comes with a set of algebraic laws to, systematically, transform a local service without protection into its cloud equivalent that is protected by composition.

FUNCTIONAL DESCRIPTION

C2QL is implemented in Idris, a functional language of the Haskell family. The implementation harnesses the Idris dependant type system to ensure the correct composition of security mechanisms and provides a transformation of the implementation into a π -calculus. This transformation serves two purposes. First, it makes the distribution explicit, showing how a computation is distributed over SaaS, PaaS and client applications. Then, it helps defining an encoding into ProVerif to check that the service preserves the privacy of its clients.

- Participants: Ronan-Alexandre Cherrueau, Rémi Douence, Mario Südholt
- Contact: Ronan-Alexandre Cherrueau
- URL: <https://github.com/rcherrueau/C2QL>

6.3. EScala

SCIENTIFIC DESCRIPTION

EScala extends the idea of events as object members, as realized by C# events, with the possibility to define events declaratively by expressions over other events. The occurrences of an event can be defined by various set operations, such as union, intersection and difference, applied on the occurrences of other events. Events can be filtered by arbitrary conditions, the data attached to the events can be transformed by arbitrary functions. Event expressions make it possible to define events in terms of other events, at the lowest level relying on primitive events.

FUNCTIONAL DESCRIPTION

EScala is an extension of Scala programming language with support for events as attributes of objects. The support for events in EScala, combine the ideas of event-driven, aspect-oriented and functional-reactive programming.

- Participants: Jacques Noyé and Jurgen Van Ham
- Contact: Jurgen Van Ham
- URL: <http://www.stg.tu-darmstadt.de/research/escala/index.en.jsp>

6.4. JEScala

FUNCTIONAL DESCRIPTION

JEScala extends EScala with support for concurrent programming. Events can be declared as asynchronous so that their handling takes place concurrently. A new composition operator, the join operator, inspired by the join calculus, can also be used to synchronize the concurrent activities created by asynchronous events and communicate between them.

- Participants: Jurgen Van Ham and Jacques Noyé
- Contact: Jurgen Van Ham
- URL: http://www.stg.tu-darmstadt.de/research/jescalas_menu/index.en.jsp

6.5. SimGrid

Scientific Instrument for the study of Large-Scale Distributed Systems

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

FUNCTIONAL 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.

- Participants: Frederic Suter, Martin Quinson, Arnaud Legrand, Takahiro Hirofuchi, Adrien Lebre, Jonathan Pastor, Mario Sudholt, Luka Stanisic, Augustin Degomme, Jean Marc Vincent, Florence Perronnin and Jonathan Rouzaud-Cornabas
- Partners: CNRS - ENS Rennes - Université de Nancy
- Contact: Martin Quinson
- URL: <http://simgrid.gforge.inria.fr/>

6.6. 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: Takahiro Hirofuchi, Adrien Lebre, Jonathan Pastor, Flavien Quesnel and Mario Sudholt
- Contact: Adrien Lebre
- URL: <http://beyondtheclouds.github.io/VMPlaceS/>

6.7. btrCloud

KEYWORDS: Cloud computing - Virtualization - Grid - Energy - Orchestration - Autonomic system - Placement - Cluster - Data center - 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% of the data-center energy consumption. The latter provides optimized quality of service properties for applications that are hosted in the virtualized datacenters.

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

DIVERSE Project-Team

6. New Software and Platforms

6.1. FAMILIAR

KEYWORDS: Software line product - Configurators - 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: Mathieu Acher, Guillaume Becan, Olivier Barais
- Contact: Mathieu Acher
- URL: <http://familiar-project.github.com>

6.2. GEMOC Studio

KEYWORDS: Model-driven engineering - Meta model - MDE - DSL - Model-driven software engineering - Dedicated langage - Language workbench - Meta-modelisation - Modeling language - Meta-modeling

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 designers to 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: Benoit Combemale, Dorian Leroy, Thomas Degueule, Erwan Bousse, Fabien Coulon and Didier Vojtisek
- Contact: Benoit Combemale
- URL: <http://gemoc.org>

6.3. Kevoree

Kevoree Core

KEYWORDS: Cloud - Deployment - Embedded - Domotique - Heterogeneity - Software Components - Architecture - Software component - Dynamic adaptation - M2M - 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 chinese 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: Jean Emile Dartois, Olivier Barais, Aymeric Hervieu, Johann Bourcier, Noel Plouzeau, Benoit Baudry, Maxime Tricoire, Jacky Bourgeois, Inti Gonzalez Herrera, Ivan Paez Anaya, Francisco Javier Acosta Padilla, Mohamed Boussaa and Manuel Leduc
- Partner: Université de Rennes 1
- Contact: Olivier Barais
- URL: <http://kevoree.org/>

6.4. Melange

KEYWORDS: Model-driven engineering - Meta model - MDE - DSL - Model-driven software engineering - Dedicated langage - Language workbench - Meta-modelisation - Modeling language - 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: Thomas Degueule, Benoit Combemale, Dorian Leroy, Erwan Bousse, Didier Vojtisek, Fabien Coulon, Jean-Marc Jezequel, Arnaud Blouin, Olivier Barais and David Mendez Acuna
- Contact: Benoit Combemale
- URL: <http://melange-lang.org>

6.5. Opencompare

KEYWORD: Software Product Line, Variability, MDE, Meta model, Configuration

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: Mathieu Acher, Guillaume Becan and Sana Ben Nasr
- Contact: Mathieu Acher
- URL: <http://opencompare.org>

6.6. 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/>

FOCUS Project-Team

6. New Software and Platforms

6.1. 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 one 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: Saverio Giallorenzo, Mila Dalla Preda, Maurizio Gabbrielli, Ivan Lanese and Jacopo Mauro
- Contact: Saverio Giallorenzo
- URL: <http://www.cs.unibo.it/projects/jolie/aioj.html>

6.2. DF4ABS

Deadlock Framework for ABS

SCIENTIFIC DESCRIPTION

We have prototyped a framework for statically detecting deadlocks in a concurrent object-oriented language with asynchronous method calls and cooperative scheduling of method activations (the language is ABS, which has been developed in the EU project HATS and is currently extended with primitives for cloud-computing in the EU project ENVISAGE. ABS is very similar to ASP, developed by the former OASIS team.). Since this language features recursion and dynamic resource creation, deadlock detection is extremely complex and state-of-the-art solutions either give imprecise answers or do not scale. In order to augment precision and scalability we propose a modular framework that allows several techniques to be combined. The basic component of the framework is a front-end inference algorithm that extracts abstract behavioural descriptions of methods that retain resource dependency information. Then these behavioural descriptions are analyzed by a back-end that uses a fix-point technique to derive in a deterministic way the deadlock information.

- Contact: Cosimo Laneve
- URL: <http://df4abs.nws.cs.unibo.it/>

6.3. HoCA

Higher-Order 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 analyzed 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. All these techniques have been specifically suited to the methods integrated in modern first-order complexity analyzers. 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), and call-by-value reductions of the PCF program are simulated by the ATRS step-by-step. On the ATRS, various complexity reflecting transformations are performed: inlining, dead-code-elimination, instantiation of higher-order variables through a call-flow-analysis and finally uncurrying. This results in a first-order rewrite system, whose runtime-complexity asymptotically reflects the complexity of the initial program.

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

6.4. 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 user-friendly 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.

DEVELOPMENTS IN 2016

Jolie has transitioned from version 1.4.1 to version 1.6. The last version of Jolie that supports Java 1.6 is Jolie 1.5. Jolie 1.6 transitions from Java 1.6 to Java 1.8 and makes use of the new features and libraries found in the new version of Java. Version 1.6 of Jolie features:

- general performance improvements and bug fixes, in particular regarding concurrent data structures using Java lambdas,
- improvements of the standard library of the language,
- better error messages and improved compatibility with the main operating systems,
- support for type choices (AKA type sums),
- support of for-loop construct to iterate over arrays without explicit indexes,
- improved support for the HTTP protocol (and, by extension, web applications).
- Participants: Claudio Guidi, Fabrizio Montesi, Saverio Giallorenzo and Maurizio Gabbrielli
- Contact: Fabrizio Montesi
- URL: <http://www.jolie-lang.org/>

6.5. SRA

Static Resource Analyzer for ABS
SCIENTIFIC DESCRIPTION

We prototype a static analysis technique that computes upper bounds of virtual machine usages in a concurrent language with explicit acquire and release operations of virtual machines. In our language it is possible to delegate other (ad-hoc or third party) concurrent code to release virtual machines (by passing them as arguments of invocations, a feature that is used by Amazon Elastic Cloud Computing or by the Docker FiWare). Our technique is modular and consists of (i) a type system associating programs with behavioural descriptions that record relevant information for resource usage (creations, releases, and concurrent operations), (ii) a translation function that takes behavioural types and returns cost equations, and (iii) an automatic off-the-shelf solver for the cost equations.

- Contact: Cosimo Laneve
- URL: <http://sra.cs.unibo.it/>

6.6. SUNNY-CP

SCIENTIFIC DESCRIPTION

Within the Constraint Programming paradigm, a portfolio solver combines different constraint solvers in order to create a globally better solver. Sunny-cp is a parallel portfolio solver that allows one to solve a Constraint (Satisfaction/Optimization) Problem defined in the MiniZinc language. It essentially implements the SUNNY algorithm introduced in the team. Sunny-cp is built on top of state-of-the-art constraint solvers, including: Choco, Chuffed, CPX, G12/LazyFD, G12/FD, G12/Gurobi, G12/CBC, Gecode, HaifaCSP, iZplus, MinisatID, Opturion, OR-Tools.

FUNCTIONAL DESCRIPTION

SUNNY-CP is a portfolio solver for solving both Constraint Satisfaction Problems and Constraint Optimization Problems. The goal of SUNNY-CP is to provide a flexible, configurable, and usable CP portfolio solver that can be set up and executed just like a regular individual CP solver.

- Contact: Maurizio Gabbrielli
- URL: <https://github.com/CP-Unibo/sunny-cp>

6.7. Blender

Aeolus Blender

KEYWORDS: Automatic deployment - Cloud applications management

SCIENTIFIC DESCRIPTION

The various tools developed in the Aeolus project (Zephyrus, Metis, Armonic) have been combined in this software which represents an integrated solution for the declarative specification of cloud applications, and its subsequent automatic deployment on an OpenStack cloud system. In particular, a web-based interface is used to specify the basic software artifacts to include in the application, indicate their level of replication, and specify co-installability conflicts (i.e. when two components cannot be installed on the same virtual machines). The tool Zephyrus is then used to synthesize the final architecture of the application, the tool Metis indicates the plan of configuration actions, and the Armonic platform provides the library of components and the low-level scripts to actually install and configure the entire application.

- Partners: IRILL - Mandriva
- Contact: Gianluigi Zavattaro
- URL: <https://github.com/aeolus-project/blender>

INDES Project-Team

4. New Software and Platforms

4.1. Hop

KEYWORDS: Domotique - Web 2.0 - Iot - Functional language - Programming
SCIENTIFIC DESCRIPTION

Hop.js is a platform for web, cloud, and IoT applications. Its development environment is composed of:

- a programming language named HopScript, which is based on ECMAScript 262, //aka// JavaScript;
- an optimized web server;
- on-the-fly compilers for generating HTML, CSS, and client-side JavaScript;
- an ahead-of-time compiler for compiling JavaScript to native code;
- numerous APIs for networking, multimedia, robotics, IoT, etc.

The HopScript language extends JavaScript to consistently define the server and client part of web applications and IoT applications. HopScript supports syntactic forms that help creating HTML elements. It supports services that enable function calls over HTTP. Being at higher level than traditional Ajax programming, Hop.js services avoid the burden and pitfalls of URL management and explicit data marshalling. They combine the benefits of a high level RPC mechanism and low level HTTP compatibility.

Although Hop.js can be used to develop traditional web servers, it is particularly adapted to the development of web applications embedded into devices, where the server and client part of the application are intimately interoperating with each other. The programming model of Hop.js fosters the joint specification of server and client code, and allows the rapid development of web user interfaces, on the client, controlling the execution of the distributed application. By defining a single data model, providing functions that can run indifferently on both sides, and almost forgetting about client-server protocols, Hop.js seems well suited for agile development of web applications for this class of applications.

- Participants: Manuel Serrano and Vincent Prunet
- Contact: Manuel Serrano
- URL: <http://hop.inria.fr>

4.2. 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: Tamara Rezk
- URL: <http://web.ist.utl.pt/~ana.matos/Mashic/mashic.html>

4.3. Webstats

Webstats is a follow-up of the internship on JavaScript constructs used in top Alexa sites, started in summer 2015 by Dolière Francis Some. He analyzed the top 10,000 Alexa sites, and provided statistics about them. Among those statistics, their are:

- the most popular JavaScript libraries
- the most recurrent JavaScript constructs
- the adoption of security features such as:
 - The Content Security Policy, a policy for defending against Cross-Site-Script attacks
 - HttpOnly and Secure cookies, that prevents attacks like session hijacking.

Starting from April, 2016, this study is performed periodically, at the end of each month. The results are accessible online at <https://webstats.inria.fr>.

- Contact: Francis Some
- URL: <https://webstats.inria.fr>

PHOENIX Project-Team

6. New Software and Platforms

6.1. College +

KEYWORDS: Neurosciences - Health - Autism - Mobile application

School+ (or College+ in french) is a package of 7 applications. Three applications are assistive applications, guiding the child doing specific tasks. Three others are training applications made as serious games, addressing specific skills. The last application is a meta-application, comprising a link to the three training applications, with an access to statistics of their usage. For each application, data are separated from the design, meaning that every element of each application (pictures, texts, settings, etc.) can be changed at any time. Each application records a log file containing all the interactions performed by the child.

6.1.1. Assistive applications

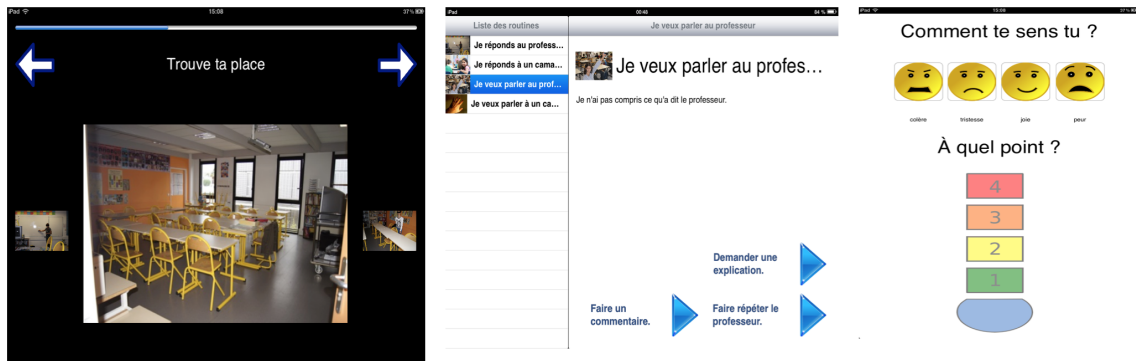


Figure 2. Assistive applications

6.1.1.1. Routines application

This application shows a list of tasks, with a short description. After clicking the starting button, a specific slideshow is shown; it decomposes a task into steps. For each step, a text and a picture can be displayed. Thumbnail of previous and next steps are also displayed. This application guides the child through classroom situations: entering classroom, taking school materials out of a backpack, writing notes, handling agenda, leaving the classroom.

6.1.1.2. Communication application

With the same design, the assistance provided by this application targets to communicating situations inside the classroom. The application covers four scenarios addressing two interaction situations (initiating and answering the interaction) and two types of interlocutors (professor and classmate). For each scenario, different slideshows guide the child, depending on the goal of the interaction.

6.1.1.3. Emotion Regulation application

This application aims to assist the child to self-regulate his/her emotions. Four simplified emoticons are proposed to the child to choose from: anger, sadness, joy and fear. Then, (s)he selects a level of intensity via a thermometer with a scale from 1 to 4. In response, the application delivers different multimedia contents according to the level selected to help the child regulate his/her emotions. Typically, a text (breathing instructions) are shown at level 1, pictures at level 2, a video at level 3 and another text at level 4.

6.1.2. Training applications

These three applications are serious games with increasing levels of difficulties, reachable after a ratio of good answers has been attained.

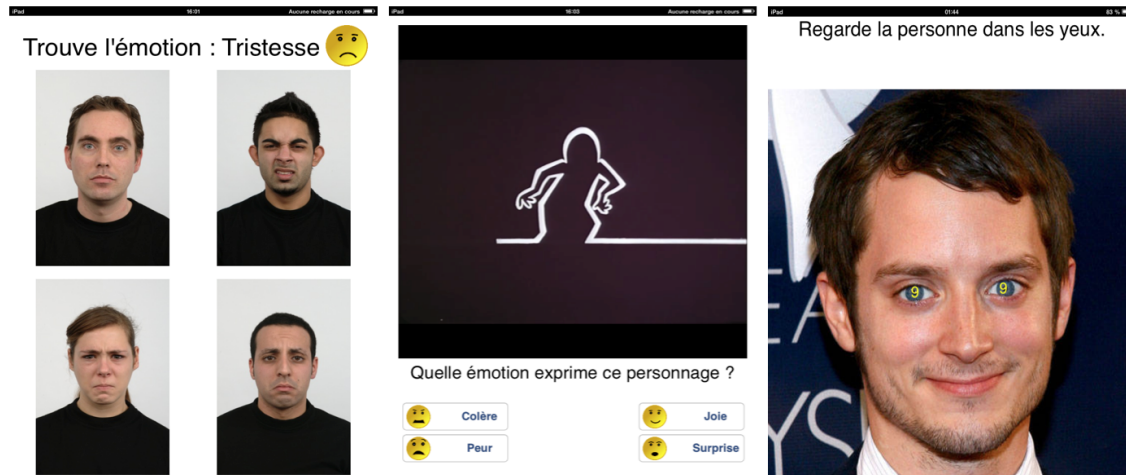


Figure 3. Training applications

6.1.2.1. Emotion Recognition application with pictures

In this application, the child is instructed to identify a specific emotion among 4 pictures showing different people exhibiting an emotion. Seven emotions are involved in this application: joy, sadness, fear, anger, surprise, disgust and neutral. The emotion to be recognized is displayed together with its simplified emoticon. The type of pictures changes with the difficulty level: level 1 contains pictures of unfamiliar people and level 2 contains pictures of friends and relatives of the child.

6.1.2.2. Emotion Recognition application with videos

In this application, the child is presented with a fragment of an animated cartoon. At some point, the video stops and the child is asked to identify the emotion of the character. Four emotions are involved in this application: joy, sadness, fear and anger. Videos are slowed down, with a speed percentage that can be changed at each level. Videos change with difficulty level: level 1 contains videos of a very basic cartoon (only one cartoon character drawn by basic form un-textured), level 2 contains a video of more sophisticated cartoons and level 3 contains movies with actors.

6.1.2.3. Attention Training

In this application, the child is presented a picture of a face and asked to make eye contact with it. Second, a symbol appears briefly in the eyes of the character. Third, the child is asked to identify the symbol shown in the previously displayed picture, to make sure he kept eye contact. The speed at which the symbol appears and disappears is changed according to the difficulty level. Types of pictures also change with the level : level 1 contains pictures of faces and level 2 contains pictures of classroom situations.

- Participants: Damien Martin Guillerez, Charles Fage, Helene Sauzeon and Alexandre Spriet
- Contact: Charles Consel

6.2. DiaSuite

SCIENTIFIC DESCRIPTION

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

6.2.1. 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.

6.2.2. 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.

6.2.3. 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.

6.2.4. 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.

6.2.5. 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. This development cycle is summarized in the Figure 2 .

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: Charles Consel, Milan Kabac, Paul Van Der Walt, Adrien Carteron and Alexandre Spriet
- Contact: Charles Consel

6.3. DiaSuiteBOX

KEYWORDS: Health - Smart home - Open application store - Development tool suite - Application certification - Home care

FUNCTIONAL DESCRIPTION

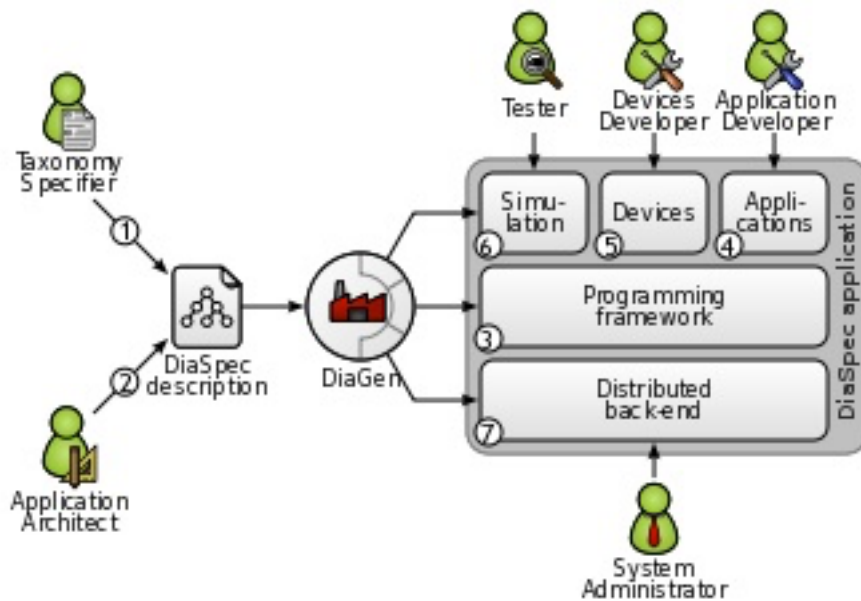


Figure 4. DiaSuite Development Cycle

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: Bertran Benjamin, Bruneau Julien, Consel Charles, Quentin Enard, Milan Kabac, Damien Martin Guillerez, Emilie Balland, Damien Cassou, Amelie Marzin, Julien Durand, Quentin Barlas, Ludovic Fornasari, Joan Rieu, Adrien Carteron, Eugene Volanschi and Helene Sauzeon
- Partners: CNRS - IPB - Université de Bordeaux
- Contact: Charles Consel
- URL: <https://diasuitebox.inria.fr/>

6.4. DomAssist

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

The HomeAssist platform (or DomAssist in french) proposes a systemic approach to introducing an assistive technological platform for older people. To do so, we formed a trans-disciplinary team that allows (1) to identify the user needs from a gerontological and psychological viewpoint; (2) to propose assistive applications designed by human factors and HCI experts, in collaboration with caregivers and users; (3) to develop and test applications by software engineers; (4) to conduct a field study for assessing the benefits of the platform and assistive applications, in collaboration with caregivers, by deploying the system at the actual home of older adults.

The HomeAssist platform is implemented on top of the DiaSuiteBox platform, using a suite of tools, namely DiaSuite, that have been designed, developed and tested by our research group at Inria. The DiaSuite tools include a dedicated integrated development environment that enables applications to be developed quickly and safely. This technology has been successfully applied to a variety of domains where environments consist of networked objects that need to be orchestrated.

6.4.1. Applications

HomeAssist offers an online catalog of applications. Using this catalog, the user and the caregiver determine what and how activities should be assisted by selecting the appropriate assistive applications and configuring them with respect to the user's requirements and preferences. The resulting set of applications forms a personalized assistive support. Additionally, to respond to evolving needs, our platform allows to stop/remove applications easily and to install new ones from the online catalog.

This platform proposes many applications in three domains of everyday life.

Daily activities: including activity monitoring, light path, and a reminder.

Home or personal safety: including entrance monitoring, stove monitoring, and warning if no movements are detected after a certain amount of time.

Communications and social activities: including collaborative games, videoconference, information about local events, TV programming, etc.

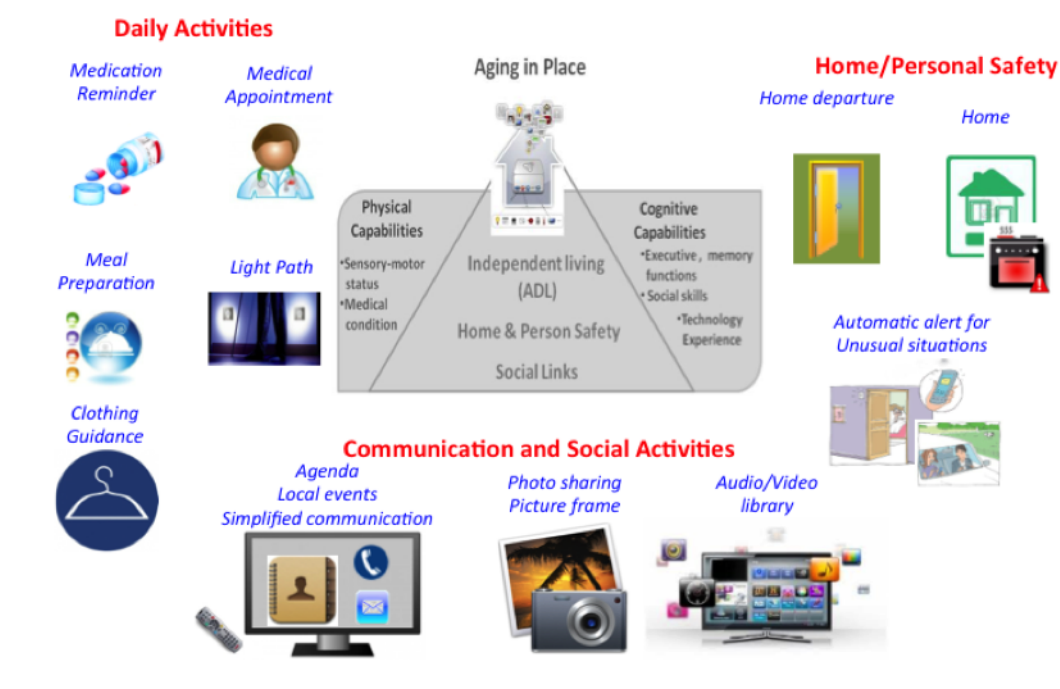


Figure 5. The HomeAssist platform and applications

For video presentations of HomeAssist, see the following:

- <http://videotheque.inria.fr/videotheque/media/23705>. Title: "DiaSuiteBox", 2013.
- <http://videotheque.inria.fr/videotheque/media/29998>. Title: "DomAssist : L'assistance numérique à la personne", 2014.

6.4.2. Devices

Several entities have been identified to deliver an assistive support. These entities include (1) technological devices: wireless sensors (motion detectors, contact sensors and smart electric switches), and two tablets, and (2) software services (agenda, address book, mail agent, and photo agent) to monitor everyday activities and

propose assistive applications. Sensors are placed in relevant rooms in the house: kitchen, bedroom, bathroom, and around the entrance.



Figure 6. HomeAssist devices

FUNCTIONAL DESCRIPTION

3 mobile applications for assistive living : (1) 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; (2) 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.; (3) eMail : mail client made for older people.

- Participants: Alexandre Spriet, Charles Consel, Helene Sauzeon and Julien Durand
- Partners: CNRS - IPB - Université de Bordeaux
- Contact: Charles Consel
- URL: <http://phoenix.inria.fr/research-projects/homeassist>

RMOD Project-Team

6. New Software and Platforms

6.1. Pharo

KEYWORDS: Live programming objet - Reflective system

FUNCTIONAL DESCRIPTION

The platform Pharo is an open-source Smalltalk-inspired language and environment. 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. In 2016, we released a new version Pharo (Pharo 5.0) completely revisited with fundamental changes in the VM (object representation, compiler, ...)

- Participants: Marcus Denker, Damien Cassou, Stephane Ducasse, Esteban Lorenzano, Damien Pollet, Igor Stasenko, Camillo Bruni, Camille Teruel and Clement Bera
- Partners: BetaNine - Debris publishing - École des Mines de Douai - HR Works - MAD - Pleiad - Sensus - Synectique - Université de Berne - Uqbar foundation Argentina - Vmware - Yesplan
- Contact: Marcus Denker
- URL: <http://www.pharo.org>

6.2. Moose

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.

- Participants: Stephane Ducasse, Muhammad Bhatti, Andre Cavalcante Hora, Nicolas Anquetil, Anne Etien, Guillaume Larcheveque and Alexandre Bergel
- Partners: Pleiad - Sensus - Synectique - Université de Berne - USI - Vrije Universiteit Brussel - Feenk
- Contact: Stephane Ducasse
- URL: <http://www.moosetechnology.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: Damien Cassou
- URL: <http://www.smalltalkhub.com/#!/~Pier/Pillar>

TACOMA Team

5. New Software and Platforms

5.1. THEGAME: data fusion for Smart Home and Smart Building

KEYWORDS: Smart home - Smart building

- Participants: Yoann Maurel and Frédéric Weis
- Partner: Université de Rennes 1
- Contact: Frédéric Weis
- URL: <https://github.com/bpietropaoli/THEGAME/> (C-version)
- URL: <https://bitbucket.org/TACOMA-irisa/java-bft> (Java-version)

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 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. It is maintained and experimented within a sensor network platform developed by TACOMA since June 2013.

5.2. Platform Pervasive_RFID

KEYWORDS: Composite objects - RFID

- Participants: Paul Couderc and Anthony Blair
- Partner: Université de Rennes 1 (IETR)
- Contact: Paul Couderc

SCIENTIFIC DESCRIPTION

In 2016 we completed the RFID experiment testbed realized in 2014-2015 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

KEYWORDS: Smart Home - Metamorphic House

- Partner: Université de Rennes 1
- Partner: Université de Rennes 1 (Fondation Rennes 1)
- Contact: Michele Dominici and Frédéric Weis

SCIENTIFIC DESCRIPTION- SOFTWARE

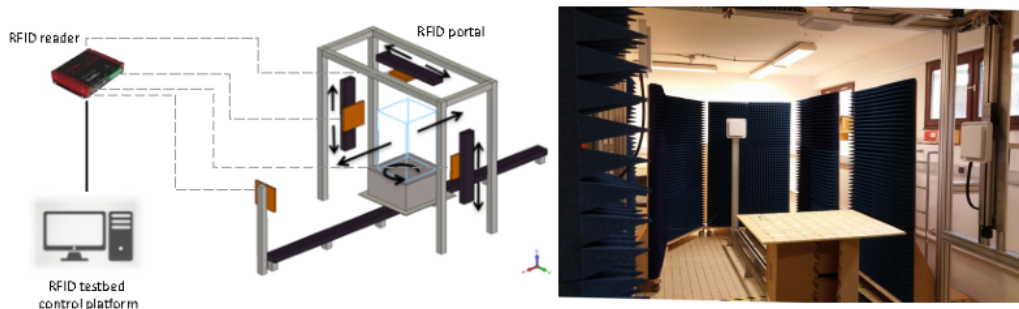


Figure 2. RFID testbed

As part of the experimentation of the On-demand room see 6.6 , 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 is expected to be deployed in the building that hosts the experimentation. This software is co-developed by Michele Dominici (University of Rennes 1), Guillermo Andrade (SED Inria) and Ghislain Nouvel (MobBI platform). Contributions might be provided by members of the Diverse project-team. Intellectual protection is expected to be applied on such software.

SCIENTIFIC DESCRIPTION- PLATFORM

In 2015, 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 Immersia. In 2016, 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 will host the experimentation is showed in Figure 3 . During the rental of the apartments, data will be collected and stored about the use of the room by households. Data will include time of occupation, mode (private or shared), consumptions, errors etc. The On-demand room will thus constitute 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: Telecom Bretagne
- Contact: Jean-Marie Bonnin

SCIENTIFIC DESCRIPTION

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



Figure 3. On-demand room real experimentation

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 VeDeCom.

COATI Project-Team

6. New Software and Platforms

6.1. BigGraphs

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.

- Participants: Luc Hogie, Nicolas Chleq, David Coudert, Michel Syska.
- Partner: This project is a joint work of the three EPI COATI, DIANA and SCALE and is supported by an ADT grant.
- Contact: Luc Hogie
- URL : <http://www.i3s.unice.fr/~hogie/biggrph/>

ADDITIONAL SOFTWARES

The following software are useful tools that bring basic services to the platform (they are not dedicated to BIGGRPH). Participants : Luc Hogie, Nicolas Chleq

- JAC-A-BOO is a framework aiming at facilitating the deployment and the bootstrapping of distributed Java applications over Share-Nothing Clusters (SNCs). The primary motivation for developing JAC-A-BOO is to have an efficient and comprehensive deployment infrastructure for the BIGGRPH distributed graph library. <http://www.i3s.unice.fr/~hogie/jacaboo>
- LDJO (Live Distributed Java Objects) is a framework for the development and the deployment of Java distributed data structures. Alongside with data aspect of distributed data structures, LDJO comes with mechanisms for processing them in a distributed/parallel way. In particular it provides implementations of Map/Reduce and Bulk Synchronous Parallel (BSP). <http://www.i3s.unice.fr/~hogie/ldjo>
- OCTOJUS provides an object-oriented RPC (Remote Procedure Call) implementation in Java. At a higher abstraction level, OCTOJUS provides a framework for the development of systolic algorithms, a batch scheduler, as well as an implementation of Map/Reduce. The latter is used in the BIGGRPH graph computing platform. <http://www.i3s.unice.fr/~hogie/octojus>

6.2. GRPH

The high performance graph library for Java

FUNCTIONAL DESCRIPTION

GRPH is an open-source Java library for the manipulation of graphs. Its main design objectives are to make it simple to use and extend, 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 solvers, a framework for distributed computing, etc.

GRPH achieves great efficiency through the use of multiple code optimization techniques such as multi-core parallelism, caching, performant data structures and use of primitive objects, interface to CPLEX linear solver, exploitation of low-level processor caches, on-the-fly compilation of specific C/C++ code, etc.

Unlike other graph libraries which impose the user to first decide if he wants to deal with directed, undirected, hyper (or not) graph, the model offered by GRPH is unified in a very general class that supports mixed graphs made of undirected and directed simple or hyper edges.

We have identified more than 600 users of GRPH since 2013. Inside Inria we collaborate with the AOSTE EPI, for example we recently added a new algorithm (proposed by N. Cohen / LRI) for iterating over the cycles of a given graph in the TimeSquare tool. We also have integrated the discrete-events simulation engine of DRMSIM and some dynamic models (evolution of the connectivity with the mobility of nodes) to GRPH. GRPH includes bridges to other graph libraries such as JUNG, JGraphT, CORESE (a software developed by the WIMMICS team Inria-I3S), LAD (C. Solnon, LIRIS), Nauty (B. D. McKay) or **Sagemath**. L. Viennot has proposed an implementation of the 4-sweep diameter algorithm designed at LIAFA .

- Participants: Luc Hogue, Nathann Cohen, David Coudert and Michel Syska.
- Contact: Luc Hogue
- URL: <http://www.i3s.unice.fr/~hogie/grph/>

6.3. Sagemath

SageMath

SageMath is a free open-source mathematics software system, initially created by William Stein (Professor of mathematics at Washington University), and now maintained by a large community of contributors. 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.

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: Matlab - LaTeX - Graph - Graph visualization - Signal processing - GNU Octave

FUNCTIONAL DESCRIPTION

Matlab / GNU Octave toolbox to manipulate and visualize signals on graphs. LaTeX package to draw signals.

- Contact: Benjamin Girault
- URL: <https://gforge.inria.fr/projects/grasp/>

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: Gaetan Harter and Frédéric Saint-Marcel
- 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. 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: Thomas Begin and Alexandre Brandwajn
- Contact: Thomas Begin
- URL: <http://queueing-systems.ens-lyon.fr/>

6.6. Data analysis tools

6.6.1. Twitter link predictions

FUNCTIONAL DESCRIPTION

Inference, study and prediction of the dynamics of the Twitter mention network

- Participants: Hadrien Hours, Eric Fleury and Márton Karsai
- Contact: Márton Karsai
- URL: <https://github.com/HadrienHours/TwitterMentionNetworkLinkPrediction>

6.7. Platforms

6.7.1. 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's Équipements d'Excellence (Equipex) research grant programme. The eFIT 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/>

DIANA Project-Team

5. New Software and Platforms

5.1. ACQUA

Participants: Chadi Barakat [contact], Thierry Spetebroot, Damien Saucez.

ACQUA is an Application for prediCting Quality of User experience at Internet Access. It was supported by the French ANR CMON project on collaborative monitoring and will be supported in 2016 by both the Inria ADT ACQUA and the ANR Project 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, etc), ACQUA targets the estimated quality of experience 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 that links the network-level measurements to the expected quality of experience. In its first version (the version available online), ACQUA was concentrating on delay measurements at the access and on the detection and estimation of the impact of delay anomalies (local problems, remote problems, etc). The current work is concentrating on using the ACQUA principle in the estimation and prediction of the quality of experience of main user's applications. An Android version is under development supported by the Inria ADT ACQUA.

- URL: <http://team.inria.fr/diana/acqua/>
- Version: 1.1
- ACM: C.2.2, C.2.3
- Keywords: Internet measurement, Internet Access, Quality of Experience
- License: GPL (3)
- Type of human computer interaction: GUI for client, Web interface for experimentation
- OS/Middleware: MS Windows
- Required library or software: visual studio <http://www.visualstudio.com/en-us/products/visual-studio-express-vs.aspx>
- Programming language: C# for client, java for server, CGI and Dummynet for experimentation

5.2. ElectroSmart

Participants: Arnaud Legout [contact], Mondri Ravi.

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 (funding the engineer Mondri Ravi) In August 2016, we released the first stable public release of ElectroSmart. On the 20th December 2016 we had 1502 downloads in Google Play, an average score of 4,66/5, 800 active users, 22 millions measured signals and 500k measured geographic zones.

- URL: <http://es.inria.fr>
- Version: 1.1
- Keywords: background electromagnetic radiations
- License: Inria proprietary licence
- Type of human computer interaction: Android application
- OS/Middleware: Android
- Required library or software: Android
- Programming language: Java
- Documentation: javadoc

5.3. nepi-ng

Participants: Thierry Parmentelat [correspondant], Thierry Turlotti, Mario Antonio Zancanaro.

During the past couple of years, we had developed NEPI, the Network Experimentation Programming Interface, as a wide spectrum tool for orchestrating network experiments on network experimentation platforms.

In the more specific context of R2lab, we have been facing more stringent requirements in terms of response time, especially when synchronizing the parallel parts of a wireless experiment. For that reason, and also because the NEPI codebase was starting to feel much too large for its actual usage, and consequently very brittle, we have decided to start and put together a new set of components, named **nepi-ng** for nepi new generation.

At this point, nepi-ng has a much smaller scope than NEPI 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.

For that reason, the actual size of the nepi-ng codebase is about 12 times smaller than the one of NEPI. However, running the same experiment on R2lab turns out to be about 10 times faster using nepi-ng rather than NEPI, that in this context is impeded by its generic model for resources, that prevents NEPI from being as reactive as what can be achieved with nepi-ng.

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 this becomes necessary.

- URL: <http://nepi-ng.inria.fr>
- Version: 0.5
- 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

5.4. OpenLISP

Participant: Damien Saucez [contact].

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. As of 2016, OpenLISP is used to provide connectivity between satellite sites of the LISP-Lab project.

- <http://www.lisp-lab.org/>
- Version: 3.2
- ACM: C.2.1, C.2.2, C.2.6
- Keywords: routing, LISP, control-plane
- License: BSD
- Type of human computer interaction: XML, CLI
- OS/Middleware: POSIX
- Required library or software: Expat 2
- Programming language: C
- Documentation: Unix man
- Deployment: <http://ddt-root.org>

5.5. Platforms

5.5.1. *Reproducible research laboratory* (R²lab)

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

We have built at Inria Sophia-Antipolis, in the last couple of years, an anechoic chamber, with RF absorbers preventing radio waves reflections and with a Faraday cage blocking external interferences. This lab, named R²lab, represents an ideal environment for experiments reproducibility.

R²lab has been operated for 2 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 2016, our focus regarding R²lab has been set on the following aspects. First, we have deployed USRPs (Universal Software Radio Peripherals) together with hardware devices for controlling these USRP extensions. This is extremely interesting, as it considerably widens the fields of application for the testbed. In particular, and that was our second angle for improvements this year, we have taken advantage of the USRP hardware to provide support for OpenAirInterface-based deployments in the chamber. That feature was demonstrated for example during the formal opening that took place on November 9 this year, where it was demonstrated how to set up a private LTE network in 3 minutes.

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

DIONYSOS Project-Team

6. New Software and Platforms

6.1. The Passive Test Tool (ttproto) and CoAP Testing Tool

ttproto is an experimental tool for implementing testing tools, for conformance and interoperability testing. It was first implemented to explore interesting features and concepts for the TTCN-3 standard. It was also used to implement a passive interoperability test suite we provided for the CoAP (Constrained Application Protocol) interoperability event held in Paris in March 2012. It is currently used for the purpose of developing testing tools (for interoperability and conformance testing) for the F-interop european project (see <http://www.f-interop.eu/>). This tool is implemented in python3 and its design was influenced mainly by TTCN-3 (abstract model, templates, snapshots, behavior trees, communication ports, logging) and by Scapy (syntax, flexibility, customizability) Its purpose is to facilitate rapid prototyping and experimentation (rather than production use). We chose to maximize its modularity and readability rather than performances and real-time considerations.

- Participants: César Viho, Federico Sismondi
- Contact: César Viho, Federico Sismondi
- <http://www.irisa.fr/tipi>

6.2. T3DevKit and IPv6 testing tools

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).

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) provides an object oriented framework to manipulate TTCN-3 entities (values, ports, timers, external functions). T3DevKit offers an implementation of the TRI and TCI standard interfaces a default implementations for the system adapter (SA), platform adapter (PA), test management (TM), test logging (TL) and component handling (CH) modules and a default codec. Built-in scripts for the generation of executable test suite, which are tool-independent facilitate the distribution of test suite sources.

IPv6 test suites have been developed using the TTCN-3 environment. The full Abstract Test Suites are written in TTCN-3 and the source files for building the codecs and adapters with the help of T3DevKit.

- Participants: César Viho, Federico Sismondi
- Contact: Federico Sismondi

6.3. Performance and dependability evaluation

Participants: Gerardo Rubino, Bruno Sericola, Bruno Tuffin.

We develop software tools for the evaluation of two classes of models: Markov models and reliability networks. The main objective is to quantify dependability aspects of the behaviors of the modeled systems, but other aspects of the systems can be handled (performance, performability, vulnerability). The tools are specialized libraries implementing numerical, Monte Carlo and Quasi-Monte Carlo algorithms.

One of these libraries has been developed for the Celar (DGA), and its goal is the evaluation of dependability and vulnerability metrics of wide area communication networks (WANs). The algorithms in this library can also evaluate the sensitivities of the implemented dependability measures with respect to the parameters characterizing the behavior of the components of the networks (nodes, lines).

We are also developing tools with the objective of building Markovian models and to compute bounds of asymptotic metrics such as the asymptotic availability of standard metrics of models in equilibrium, loss probabilities, blocking probabilities, mean backlogs, etc. A set of functions designed for dependability analysis is being built under the internal name DependLib.

We contribute to the development of SPNP (*Stochastic Petri Net Package*). SPNP is used by more than 200 companies and universities. The main designer is Duke University. Our contributions are on Monte Carlo methods. We plan to increase our participation in the development of this tool.

Pierre L'Ecuyer is also developing in Montreal a library, *Stochastic Simulation in Java* (SSJ), providing facilities for generating uniform and nonuniform random variates, computing different measures related to probability distributions, performing goodness-of-fit tests, applying quasi-Monte Carlo methods, collecting (elementary) statistics, and programming discrete-event simulations with both events and processes.

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>

EVA Project-Team

6. New Software and Platforms

6.1. OpenWSN (Software)

Participants: Tengfei Chang, Jonathan Muñoz, Malisa Vucinic, Thomas Watteyne.

OpenWSN (<http://www.openwsn.org/>) 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.

The OpenWSN ADT started in 2015, with Research Engineer Tengfei Chang who joined the EVA team.

Highlights for 2016:

- Development:
 - better (continuous) testing of the existing OpenWSN code.
 - Build a image for a RaspberryPi which contains the OpenVisualizer preinstalled.
 - Port OpenWSN to the First prototype of the “Single-Chip uMote” (SCuM), developed in Prof. Pister’s lab at UC Berkeley.
 - Create a Virtual Machine image with all OpenWSN development tools preinstalled.
 - Implementation of 6TiSCH standards as they appear, including revisions.
 - Maintenance of the “Golden Image” used as a reference during interoperability testing
- Recognition:
 - OpenWSN remains ETSI’s reference implementation for IETF 6TiSCH-related standards. It is therefore the base for the ETSI’s Golden Device for 6TiSCH standards, including IEEE802.15.4e TSCH, 6LoWPAN and RPL.
- Events:
 - **Tutorial**
Standards for the Industrial IoT: a Hands-on Tutorial with OpenWSN and OpenMote. Xavier Vilajosana, Pere Tuset-Peiro, Tengfei Chang, **Thomas Watteyne**. IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC), Valencia, Spain, 4-8 September 2016.
 - **Tutorial**
Introduction to the IETF 6TiSCH stack with OpenWSN & OpenMote. **Thomas Watteyne**, Xavier Vilajosana, Pere Tuset-Peiro, Tengfei Chang. International Conference on Telecommunications (ICT), Thessaloniki, Greece, 16-18 May 2016.
 - **Interop Event**
ETSI 6TiSCH 3 plugtests, 15-16 Juny 2016, Berlin, Germany
 - **Interop Event**
ETSI 6TiSCH 2 plugtests, 2-4 February 2016, Paris, France

This software appears in <https://bil.inria.fr/>.

6.2. OPERA and OCARI (Software)

Participants: Erwan Livolant, Pascale Minet.

The OPERA software was developed by the Hipercom2 team in the OCARI project. It includes EOLSR, an energy efficient routing protocol and OSERENA, a coloring algorithm optimized for dense wireless networks. It was registered by the APP. In 2013, OPERA has been made available for download as an open software from the InriaGForge site: https://gforge.inria.fr/scm/?group_id=4665

In 2014, OPERA has been ported on a more powerful platform based on the Atmel transceiver AT86RF233 and on a 32 bits microcontroller Cortex M3. More details and documentation about this software are available in the website made by the Eva team: <http://opera.gforge.inria.fr/index.html>

In 2016, Erwan Livolant developed extensions to allow the remote management of the OCARI network and the transmission of commands to sensors and actuators.

6.3. F-Interop (Software)

Participants: Remy Leone, Thomas Watteyne.

F-Interop is revolutionizing the way interoperability events are conducted. We are building a cloud-based system which allows implementors to meet online to test their implementations against one another, verify compliance in a automated way, and verify the performance of their implementations on large scale testbeds. This significantly cuts down time-to-market for standards-based solutions, and eventually leads to more standards-based products on the market. The F-Interop platform starts by targeting 6TiSCH, 6LoWPAN and CoAP standards, but our ambition is for F-Interop to become the standard way of doing interoperability, at least at the IETF IoT level.

This implementation is done as part of the H2020 F-Interop project. More information at <http://www.f-interop.eu/>.

This software appears in <https://bil.inria.fr/>.

6.4. 6TiSCH Simulator (Software)

Participants: Malisa Vucinic, Thomas Watteyne.

The 6TiSCH simulator allows one to conduct high-level simulator of an IETF 6TiSCH network and answer questions such as How long does it take the nodes to join the network? What is the average power consumption? What does the latency distribution look like?

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.

This implementation is done as part of the H2020 ARMOUR project and the standardization activities of the Inria-EVA team at the IETF. It is published under an open-source BSD license and maintained at <https://bitbucket.org/6tisch/simulator/>.

This software appears in <https://bil.inria.fr/>.

6.5. 6TiSCH Wireshark Dissector (Software)

Participants: Jonathan Muñoz, Thomas Watteyne.

The goal of this project is to maintain Wireshark dissectors for 6TiSCH (and 6TiSCH-related) standards up-to-date.

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

This implementation is done as part of the collaboration with Gridbee and the standardization activities of the Inria-EVA team at the IETF. It is published under an open-source BSD license and maintained at <https://github.com/openwsn-berkeley/dissectors>.

This software appears in <https://bil.inria.fr/>.

6.6. Mercator (Software)

Participants: Keoma Brun-Laguna, Thomas Watteyne.

Mercator allows one to evaluate the connectivity in a low-power wireless network. It is a collection of tools, including the firmware to load on the devices, the scripts that automate the measurements of the connectivity and the tools to structure and display results.

The firmware is written as part of the OpenWSN project. Scripts and analysis tools are written in Python.

It is published under an open-source BSD license and maintained at <https://github.com/openwsn-berkeley/mercator>.

This software appears in <https://bil.inria.fr/>.

6.7. CONNEXION (Software)

Participants: Pascale Minet, Erwan Livolant.

In the CONNEXION project, the integration of the OCARI wireless sensor network in a Service-Oriented Architecture using the OPC-UA/ROSA middleware went on with Telecom ParisTech. More precisely, we developed the remote management of the OCARI network as well as the possibility to generate commands to the sensors and actuators.

Erwan Livolant developed an OCARI frame dissector plugin for Wireshark (<https://www.wireshark.org>) available from the Git repository at OCARI website. This tool displays the contents of the packets sniffed for the MAC, the NWK and the Application layers, taking into account the specificities of OCARI.

6.8. SolSystem (Software)

Participants: Keoma Brun-Laguna, Thomas Watteyne.

SolSystem is a back-end solution for a low-power wireless mesh network based on SmartMesh IP. It defines how low-power wireless devices must format and transfer sensor data, and the tools to gather, store and display sensor data.

This system is used in several deployments, including <http://savethepeaches.com/> and <http://snowhow.io/>.

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).

It is published under an open-source BSD license. Information and overview at <http://www.solssystem.io/>.

This software appears in <https://bil.inria.fr/>.

6.9. Argus (Software)

Participants: Remy Leone, Thomas Watteyne.

Share your low-power wireless sniffer through the cloud!

Imagine you are a team of low-power wireless enthusiasts developing the next generation of products in standards. One essential tool in your toolkit is a low-power wireless sniffer, which shows you the frames which fly through the air.

Rather than requiring each person in your team to have a sniffer, Argus allows you to put in a share a sniffer through the cloud.

There are three pieces to 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 connects Argus Probes with Argus Clients based on a pub-sub architecture.
- Several Argus Clients can be started at the same time. It is a program which subscribes to the Argus Broker and displays the frames in Wireshark.

It is published under an open-source BSD license, maintained at <https://github.com/openwsn-berkeley/argus>.

This software appears in <https://bil.inria.fr/>.

6.10. SAHARA (Software)

Participants: Ines Khoufi, Erwan Livolant, Pascale Minet.

Ines Khoufi developed modules for the simulation of the Time Slotted Channel Hopping (TSCH) on the ns3 simulation tool. These modules include multi-interface management and transmission management according to a given schedule.

Erwan Livolant developed a SAHARA frame dissector plugin for Wireshark (<https://www.wireshark.org>). This tool displays the contents of the packets sniffed for the MAC and the NWK layers, taking into account the specificities of the SAHARA project.

6.11. FIT IoT-LAB (Platform)

Participants: Thomas Watteyne, Tengfei Chang.

Note well: IoT-lab is *not* a project of Inria-EVA. It is a large project which runs from 2011 to 2021 and which involves the following other partners Inria (Lille, Sophia-Antipolis, Grenoble), INSA, UPMC, Institut Télécom Paris, Institut Télécom Evry, LSIIT Strasbourg. This section highlights Inria-EVA activity and contribution to the IoT-lab testbed in 2016.

The Inria-EVA team has been using the platform extensively throughout 2016. During the process, we have been interacting closely with the IoT-lab team of engineers.

IoT-lab-related activities include:

- **Running OpenWSN networks at scale.** The IoT-lab has been an incredibly powerful tool to verify the scalability of the protocols and implementations in OpenWSN (Section 6.1). *Lead: Tengfei Chang.*
- **Assessing connectivity with Mercator.** The Mercator project (Section 6.6) is targeted at measuring the connectivity of the IoT-lab platform in time, space and frequency. *Lead: Keoma Brun-Laguna.*
- **16-channel Sniffer with Argus.** We are currently working extending the Argus project (Section 6.9) with support for IoT-lab nodes. That is, rather than using a 16-channel sniffer, turn 16 IoT-lab nodes into a distributed multi-frequency sniffer. *Lead: Remy Leone.*
- **IoT-lab for Interop Testing with F-Interop.** Through the H2020 F-Interop project, we are developing the tools (see Section 6.3) to run an F-Interop user's code on the IoT-lab to verify conformance and interoperability. *Lead: Remy Leone.*
- **Scalable Security Solution with H2020 ARMOUR.** Through the H2020 ARMOUR project, Inria-EVA is testing whether security solutions standardized at the IETF are scalable. *Lead: Malisa Vucinic.*

FUN Project-Team

6. New Software and Platforms

6.1. FIT IoT-Lab

Participants: Nathalie Mitton [correspondant], Julien Vandaele.

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 developing software for big graph manipulation. A preliminary library offering diameter and skeleton computation is available online ⁰. This library was used to compute the diameters of the worldwide road network (200M edges), skeleton subtrees of the shortest-path trees of continental-sized road networks, as well as the largest strongly connected component of the Twitter follower-follower graph (23G edges).

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

⁰<https://who.rocq.inria.fr/Laurent.Viennot/dev/big-graph-tools/>

INFINE Project-Team

5. New Software and Platforms

5.1. RIOT

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

SCIENTIFIC DESCRIPTION

While requiring as low as 1,5kB of RAM and 5kB of ROM, RIOT offers real time and energy efficiency capabilities, as well as a powerful API (partially POSIX compliant) that is consistent across heterogeneous low-end IoT hardware (8-bit, 16-bit and 32-bit architectures). This API is developer-friendly in that it enables multi-threading, standard C and C++ application programming (as well as scripting) and the use of standard debugging tools – all of which was not possible so far for such embedded programming. On top of this, RIOT supports a large number of software libraries (e.g. crypto, maths, drivers...) and aggregates in a simple manner a wide variety of third-party open source software packages. In particular, various network stacks are available with RIOT, such as a standard IPv6/6LoWPAN stack and an 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. Other use-cases include also IoT-enabled low-cost robots.

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, Oliver Hahm, Cedric Adjih, Francisco Acosta
- Partner: Freie Universität Berlin
- Contact: Emmanuel Baccelli
- URL: <https://github.com/RIOT-OS/RIOT>

5.2. MACACO

Mobile context-Adaptive CAching for COntent-centric networking

FUNCTIONAL DESCRIPTION

MACACOapp is developed 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. GardiNet (previously DragonNet)

FUNCTIONAL DESCRIPTION

GardiNet (previously known as DragonNet) is a generic framework for network coding in wireless networks. It is initially a 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, GardiNet 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: Cedric Adjih, Ichrak Amdouni, Hana Baccouch and Antonia Masucci
- Contact: Cedric Adjih
- URL: <http://getrf.gforge.inria.fr/https://github.com/GardiNet/gardinet>

MADYNES Project-Team

5. New Software and Platforms

5.1. 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 homogenous 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 - Grid'5000 - Inria - Loria - Université de Lorraine
- Contact: Lucas Nussbaum
- URL: <http://distem.gforge.inria.fr>

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

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

5.4. MECSYCO-RE-C++

Multi-agent Environment for Complex SYstems COsimulation. Cœur C++

KEYWORDS: Modeling - Simulation - Simulator - Multi-model - Multi-agent - Agent - 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 complemented by mecsyco-com (a communication package for distributed execution) and mecsyco-visu (a set of tools for visualizing simulations).

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

5.5. MECSYCO-RE-java

Multi-agent Environment for Complex SYstems COsimulation. Coeur java

KEYWORDS: Modeling - Simulation - Simulator - Multi-model - Co-simulation - Multi-agent - Agent - 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 execution) and mecsyco-visu (a set of tools for visualizing simulations).

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

5.6. NDNperf

KEYWORDS: Performance measure - Named-Data Networking

FUNCTIONAL DESCRIPTION

We designed NDNperf, an open source tool for NDN server-side performance evaluation and sizing purposes, in order to have an idea of the throughput a server can achieve when it has to generate and transmit NDN Data packets. It is very similar to iPerf and also needs a client and a server to perform the measurements while minimizing the number of instructions between Interest reception and Data emission. It has the following features: - Periodic report of performances: end-to-end throughput, latency, processing time, - Fresh NDN Data generation or NDN Data delivery from caches, - Multi-threaded (one main thread for event lookup and N threads for NDN Data generation), - Able to use all available signatures implemented in the NDN library, choose the size of the key, and the transmission size of Data packets.

- Contact: Thibault Cholez
- URL: <http://madyes.loria.fr/software/>

5.7. 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/>

MAESTRO 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, Issam Rabhi
- Partner: UVSQ (Univ. Versailles Saint-Quentin)
- Contact: Alain Jean-Marie
- URL: <http://marmotecore.gforge.inria.fr/>

6.2. ns-3

KEYWORDS: Simulation - Communication networks

FUNCTIONAL DESCRIPTION

ns-3 is a discrete-event network simulator for Internet systems, targeted primarily for research and educational use.

In the framework of the research project with ALSTOM Transport (see §8.1.3), we have extensively validated several modules of ns-3, related to the PHY and the MAC layers. We have implemented a directional antenna using 3-dimensional data for the radiation diagram. Modules related to the Automatic Train Protection function used in train systems have been implemented and validated. We have also developed a generator of video traffic and objects that allow to generate easily simulation scenarios.

We have made available the code related to the communication based train control and the one generating video traffic. Some of our contribution to the ns-3 simulator and selected results illustrating some of the issues that can be addressed using our contribution are presented and discussed in [35].

- Participants: Sara Alouf, Giovanni Neglia and Alina Tuholukova
- Contact: Alina Tuholukova
- ns-3 codereview issue of the cbtc module: <https://codereview.appspot.com/289110043>
- ns-3 codereview issue of the video generator: <https://codereview.appspot.com/286160043>

MUSE Team

5. New Software and Platforms

5.1. Fathom

Fathom - browser-based network measurement platform

KEYWORDS: Internet access - Performance measure - Network monitoring

FUNCTIONAL DESCRIPTION

Fathom is a Firefox browser extension that explores the browser as a platform for network measurement and troubleshooting. It provides a wide range of networking primitives directly to in-page JavaScript including raw TCP/UDP sockets, higher-level protocol APIs such as DNS, HTTP, and UPnP, and ready-made functionality such as pings and traceroutes.

- Participants: Anna-Kaisa Pietilainen and Stephane Archer
- Contact: Anna-Kaisa Pietilainen
- URL: <https://muse.inria.fr/fathom/>

5.2. HostView

FUNCTIONAL DESCRIPTION

End-host performance monitoring and user feedback reporting

- Participants: George Rosca, Anna-Kaisa Pietilainen and Renata Cruz Teixeira
- Contact: Renata Cruz Teixeira
- URL: <https://team.inria.fr/muse/>

5.3. Online HoA

Online implementation of home and access throughput bottleneck detection algorithm 'HoA'

FUNCTIONAL DESCRIPTION

"Home or Access" (HoA) is a system that localizes performance problems in home and access networks. Originally, we implement HoA as custom firmware that collect traces from off-the-shelf home routers. HoA uses timing and buffering information from passively monitored traffic at home routers to detect both access link and wireless network bottlenecks. HoA runs offline on a server to locate last-mile downstream throughput bottlenecks based on the analysis of packet traces collected from home routers. Our attempts to run HoA online on commodity home routers, however, revealed the challenges with performing per-packet analysis on such resource-constrained devices. The online HoA resolves this issue. We design an access bottleneck detector based on lightweight pings of the access link, and a wireless bottleneck detector based on a model of wireless capacity using metrics that are easily available in commodity home routers such as the wireless physical rate and the count of packets/bytes transmitted.

- Contact: Renata Cruz Teixeira
- URL: <https://github.com/inria-muse/browserlab>

5.4. SimilarityExplanation

Prototype implementation for explaining a set of similar and recommended movies.

FUNCTIONAL DESCRIPTION

In this web-based prototype for similar movies explanation, we propose two types of browsing for: personalized browsing and non personalized browsing. In the non personalized browsing we suppose that we don't have the user profile. Similar movie sublists are ordered only according to their similarity to the selected movie. For the personalized browsing , we select users that have different profiles from our dataset. We give these users names of actors, according to the types of movies they watch. For each user, we compute the predicted ratings using the matrix factorization model. We select pairs of genres to display to each user based on the preferred genres for the user. In our prototype we identify the preferred genres per user based on the most frequent movie genre pairs that the user has already seen. We then organize the recommended movies with a high rating prediction in sublists, according to the user most preferred genre pairs. When a user selects a movie from the sublists of recommended movies, our application suggests the similar movies presented under four sublists with the added list of words. The sublists are personalized for each user by reordering the movies according to the users predicted ratings.

- Contact: Sara El Aouad
- URL: <https://team.inria.fr/muse/>

5.5. UCN

User-Centric Networking

FUNCTIONAL DESCRIPTION

The User-Centric Networking (UCN) project is seeking to understand how people consume various kinds of content when using computer networks. Within this project we are undertaking a detailed user study across a range of environments in order to understand the practices involved in consuming media and other content according to context.

- Participants: Renata Cruz Teixeira and Anna-Kaisa Pietilainen
- Contact: Anna-Kaisa Pietilainen
- URL: <https://team.inria.fr/muse/>

5.6. WeBrowse

FUNCTIONAL DESCRIPTION

WeBrowse is the first passive crowdsource-based content curation system. Content curation is the act of assisting users to identify relevant and interesting content in the Internet. WeBrowse requires no active user engagement to promote content. Instead, it extracts the URLs users visit from traffic traversing an ISP network to identify popular content. WeBrowse contains a set of heuristics to identify the set of URLs users visit and to select the subset that are interesting to users.

- Contact: Giuseppe Scavo
- URL: <http://webbrowse.polito.it/>

RAP Project-Team (section vide)

SOCRATE Project-Team

5. New Software and Platforms

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

5.2. WSNet

KEYWORD: Network simulator

FUNCTIONAL DESCRIPTION

WSNet is a modular event-driven simulator targeted to Wireless Sensor Networks. Its main goals are to offer scalability, 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, Quentin Lampin, Alexandre Mouradian and Fabrice Valois
- Partner: CEA-LETI
- Contact: Guillaume Chelius
- URL: <https://gforge.inria.fr/projects/wsnet-3/>

5.3. 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 Jean-Marie 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 computation requests. However, we have shown that the use of a multi-resolution approach allows the main computational load to be restricted to a pre-processing phase. Extensive works have been done to make predictions more realistic.

- Contact: Jean-Marie Gorce
- URL: <https://bil.inria.fr>

5.4. FFTweb

KEYWORD: Spectrum Analyser , Data visualization , SDR

FUNCTIONAL DESCRIPTION

Visualisation tool use in CorteXlab to visualize the spectrum (or any kind vector signal) occurring in the CorteXlab room. FFTweb is a fundamental debugging and demonstration component for FIT/CorteXlab user.

- Matthieu Imbert
- Partners: Inria
- Contact: Matthieu Imbert
- URL: <http://www.cortexlab.fr>

5.5. Minus

KEYWORD: Experiment Handler , SDR

FUNCTIONAL DESCRIPTION

Handling and deployment of experiment on the Cognitive radio platform FIT/CorteXlab. On CorteXlab, the user does not have direct access to the SDR nodes, he has access to a server from which Minus deploys the programs on the different SDR nodes.

- Matthieu Imbert, Leonardo Sampaio-Cardoso, Tanguy Risset
- Partners: Inria
- Contact: Matthieu Imbert
- URL: <http://www.cortexlab.fr>

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 (NOC), a set of IoT test-beds (IoT-Lab), a set of wireless OneLab test-beds, and a cognitive radio test-bed (CorteXlab) deployed by the Socrate team in the Citi lab. 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 2014 ⁰, 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.

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



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

URBANET Team

6. New Software and Platforms

6.1. PrivaMovApp

FUNCTIONAL DESCRIPTION

UrbaNet 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: Razvan Stanica and Hervé Rivano.
- Contact: Razvan Stanica
- URL: <http://liris.cnrs.fr/privamov/project/>

6.2. TAPASCologne

Travel and Activity PAtterns Simulation Cologne

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: Razvan Stanica
- URL: <http://kolntrace.project.citi-lab.fr/>

6.3. Platforms

6.3.1. Sense in the City

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: Khaled Boussetta, Hervé Rivano.
- Contact: Khaled Boussetta

6.3.2. Extention of the FIT IoT Lab Equipex in Tech La Doua Campus

This testbed is located in an experimentation room which belongs to the CITI laboratory and the Telecommunications Department of INSA Lyon, Villeurbanne. The target usages of this room are quite diverse: practical works with students, robots/drones testing, wireless sensor networks experimentation, Wi-Fi security evaluation, services deployment, etc. During an experimentation, this room could be shared with others practical works. Basically, we claim that this room is useful to observe the behavior of nodes with this dense interactivity. 18 M3 open nodes, 11 A8 nodes and 12 mobile on robots are available for experimentation.

- Participants: Romain Pujol, Hervé Rivano, Fabrice Valois.
- Contact: Fabrice Valois
- URL: <https://www.iot-lab.info/deployment/lyon/>