



RESEARCH CENTER

FIELD

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

Activity Report 2015

Section Software

Edition: 2016-03-21

DISTRIBUTED SYSTEMS AND MIDDLEWARE

1. ASAP Project-Team	5
2. ATLANMODELS Team	8
3. CIDRE Project-Team	14
4. COAST Project-Team	17
5. CTRL-A Team	18
6. MIMOVE Team	19
7. MYRIADS Project-Team	22
8. REGAL Project-Team	26
9. SCALE Team	28
10. SPIRALS Project-Team	30
11. WHISPER Project-Team	32

DISTRIBUTED AND HIGH PERFORMANCE COMPUTING

12. ALPINES Project-Team	35
13. AVALON Project-Team	37
14. HIEPACS Project-Team	40
15. KERDATA Project-Team	45
16. MESCAL Project-Team	48
17. MOAIS Project-Team	53
18. ROMA Project-Team	56
19. STORM Team	59
20. TADAAM Team	63

DISTRIBUTED PROGRAMMING AND SOFTWARE ENGINEERING

21. ASCOLA Project-Team	66
22. DIVERSE Project-Team	68
23. FOCUS Project-Team	72
24. INDES Project-Team	75
25. PHOENIX Project-Team	76
26. RMOD Project-Team	83
27. TACOMA Team	84

NETWORKS AND TELECOMMUNICATIONS

28. COATI Project-Team	88
29. DANTE Project-Team	90
30. DIANA Project-Team	92
31. DIONYSOS Project-Team	97
32. DYOGENE Project-Team	99
33. EVA Team	100
34. FUN Project-Team	103
35. GANG Project-Team	104
36. INFINE Team	105
37. MADYNES Project-Team	107

38. MAESTRO Project-Team	110
39. MUSE Team	111
40. RAP Project-Team (section vide)	113
41. SOCRATE Project-Team	114
42. URBANET Team	117

ASAP Project-Team

5. New Software and Platforms

5.1. 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: Davide Frey

5.2. 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 AllYours-P2P.

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

5.3. 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/>

5.4. MediEgo

KEYWORDS: Widget web - Social network - Recommendation

FUNCTIONAL DESCRIPTION

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, Sébastien Campion, Rachid Guerraoui, Davide Frey and Arnaud Jegou
- Partner: EPFL - Ecole Polytechnique Fédérale de Lausanne
- Contact: Sébastien Campion
- URL: <http://www.mediego.com>

5.5. WebGC

Web-based Gossip Communication

SCIENTIFIC DESCRIPTION

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

FUNCTIONAL DESCRIPTION

WebGC 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

5.6. 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

5.7. 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, AllYours-P2P, and Behave.

This year's new work consisted in adding support for Non-Blocking I/O. It was carried out during the internship of Nominoe Kervadec.

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

5.8. 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 P2PAllYours, - 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

ATLANMODELS Team

6. New Software and Platforms

6.1. AM3

AtlanMod MegaModel Management

KEYWORDS: Modeling artifact - MDA - MDE - Megamodeling

FUNCTIONAL DESCRIPTION

AM3 (AtlanMod MegaModel Management) is a generic and extensible tool/framework dedicated to global model management, i.e., the management of different modeling artifacts as well as their interrelationships.

- Participant: Hugo Brunelière
- Partner: Ecole des Mines de Nantes
- Contact: Hugo Brunelière
- URL: <http://wiki.eclipse.org/AM3>

6.2. AMW

Atlas Model Weaver

KEYWORDS: MDA - MDE - Weaving - Link - Model element

FUNCTIONAL DESCRIPTION

AMW is a generic and extensible tool dedicated to the creation and handling of weavings between models, these weavings representing sets of links existing between elements potentially coming from different models.
Language

- Participant: Frédéric Jouault
- Contact: Hugo Brunelière
- URL: <http://wiki.eclipse.org/AMW>

6.3. ATL

Atlanmod Transformation Language

KEYWORDS: MDA - MDE - ATL - QVT - Model - Transformation

FUNCTIONAL DESCRIPTION

ATL is a tool dedicated to model transformation, complete, integrated into Eclipse and including its own development and execution environment, concrete examples as well as the corresponding associated documentation.

- Participants: Jean Bézivin, Frédéric Jouault and Patrick Valduriez
- Partner: Ecole des Mines de Nantes
- Contact: Hugo Brunelière
- URL: <http://www.eclipse.org/m2m/atl/>

6.4. ATL-MR

ATL-MapReduce

KEYWORDS: Transformation - Distributed

SCIENTIFIC DESCRIPTION

ATL-MapReduce

ATL-MapReduce (ATL-MR) is a prototype tool for running complex ATL transformation in the cloud using Hadoop MapReduce. ATL-MapReduce is implemented on top of an extended ATL VM that can be found on (<https://github.com/atlanmod/org.eclipse.atl.atlMR/tree/master>). Coupling ATL-MR with the [the extended VM](<https://github.com/atlanmod/org.eclipse.atl.atlMR/tree/master>) has proved a good performance, especially in terms of execution time. [In our experiments](<http://www.emn.fr/z-info/atlanmod/index.php/Image:Atlmr-experiments-raw-data.zip>), ATL-MR runs up to 6x faster compared to the regular VM while distributing it over 8 machines.

- Participants: Amine Benelallam, Abel Gomez Llana and Massimo Tisi
- Contact: Amine Benelallam
- URL: https://github.com/atlanmod/ATL_MR

6.5. ATLAS model weaver

ATLAS Model Weaver

KEYWORDS: Model - Metamodel - Weaving model

SCIENTIFIC DESCRIPTION

AMW is a component-based platform for model weaving that can be used to establish and manage abstract correspondences between models. The platform is generic and based on the Eclipse contribution mechanism: components are defined in separate plugins. The plugins are further interconnected to create the model weaver workbench. Components for user interface, matching algorithms and serialization of models may be plugged as necessary. We extended the Eclipse EMF architecture for model manipulation to coordinate the weaving actions. We use the EMF reflective API to obtain a standard weaving editor which adapts its interface according to metamodels modifications. The ATL transformation engine is plugged as the standard transformation platform.

FUNCTIONAL DESCRIPTION

The AMW is a tool for establishing relationships (i.e., links) between models. The links are stored in a model, called weaving model. It is created conforming to a weaving metamodel.

- Participants: Jean Bézivin, Erwan Breton, Marcos Didonet Del Fabro, Guillaume Gueltas, Frédéric Jouault and Patrick Valduriez
- Contact: Frédéric Jouault
- URL: <http://www.eclipse.org/gmt/amw/>

6.6. AmmA

Atlas Model Management Architecture

FUNCTIONAL DESCRIPTION

The AMMA platform is a model management platform. It is composed of several elements. Three of them: AM3, AMW, and MoDisco are available as GMT components. ATL is a component of M2M project.

- Participants: Mikaël Barbero and Frédéric Jouault
- Contact: Frédéric Jouault
- URL: <https://wiki.eclipse.org/AMMA>

6.7. Collaboro

SCIENTIFIC DESCRIPTION

Collaboro is an approach to make language development processes more participative, meaning that both developers and users of the language can collaborate together to create and evolve it. Collaboro supports both the collaborative definition of the abstract (i.e., metamodel) and concrete (i.e., notation) syntaxes for your DSL by providing a collaborative environment enabling the discussion.

Anyone has the chance to request changes, propose solutions and give an opinion (and vote) about those from others. This discussion enrichs the language definition significantly and ensures that the end result satisfies as much as possible the expectations of the end-users.

- Participants: Jordi Cabot, Robin Boncorps and Javier Canovas Izquierdo
- Contact: Hugo Brunelière
- URL: <http://atlanmod.github.io/collaboro/#/>

6.8. EMF Facet

KEYWORDS: MDE - Model - Metamodel - Extension - Dynamicity - Querying

FUNCTIONAL DESCRIPTION

EMF Facet is a tool/framework dedicated to the non-intrusive and dynamic extension of metamodels (adding of new types, attributes, references, etc), based on a model query generic mechanism

- Partners: Ecole des Mines de Nantes - Mia-Software
- Contact: Hugo Brunelière
- URL: <http://www.eclipse.org/modeling/emft/facet/>

6.9. EMF Views

Eclipse Modeling Framework Views

FUNCTIONAL DESCRIPTION

Some users only need to see some parts of a model, others have to get the full model extended with data from another model, and others simply access to a combination of information coming from different models. Based on the unquestionable success/usefulness of database views to solve similar problems in databases, EMF Views aims to bring the same concept to the modeling world.

- Contact: Hugo Brunelière
- URL: <https://github.com/atlanmod/emfviews>

6.10. EMF-REST

Eclipse Modeling Framework Rest

FUNCTIONAL DESCRIPTION

EMF is the modeling framework of the Eclipse community. While EMF is able to automatically generate Java APIs from Ecore models, it is still missing support to deal with Web APIs such as RESTful ones that could boost the use of modeling techniques in the Web. However, the creation of RESTful APIs requires from developers not only an investment in implementation but also a good understanding of the REST Principles to apply them correctly. We therefore created EMF-REST, a tool that empowers EMF to get Truly RESTful APIs from Ecore models, thus allowing web developers to generate JSON-based Web APIs for their applications. It generates both a JavaScript API to work with models as Javascript Objects in the client-side (without any EMF dependency) and REST services in the server-side based on the Java JAX-RS specification.

- Contact: Gerson Sunyé
- URL: <http://emf-rest.com/>

6.11. EMFtoCSP

SCIENTIFIC DESCRIPTION

Essentially, the EMFtoCSP is a sophisticated bounded model finder that yields instances of the model that conform not only to the structural definition of the model (e.g. the multiplicity constraints), but also to the OCL constraints. Based on this core, several correctness properties can be verified:

Satisfiability – is the model able to express our domain? For this check, the minimal number of instances and links can be specified to ensure non-trivial instances.

Unsatisfiability – is the model unable to express undesirable states? To verify this, we add further constraints to the model that state undesired conditions. Then we can check if it is impossible to instantiate the amended model.

Constraint subsumption – is one constraint already implied by others (and could therefore be removed)?

Constraint redundancy – do different constraints express the same fact (and could therefore be removed)?

FUNCTIONAL DESCRIPTION

EMFtoCSP is a tool for the verification of precisely defined conceptual models and metamodels. For these models, the definition of the general model structure (using UML or EMF) is supplemented by OCL constraints. The Eclipse Modeling Development Tools (MDT) provides mature tool support for such OCL-annotated models with respect to model definition, transformation, and validation.

- Contact: Hugo Brunelière
- URL: <https://github.com/SOM-Research/EMFtoCSP>

6.12. GiLA

GitHub Label Analyzer

FUNCTIONAL DESCRIPTION

Reporting bugs, asking for new features and in general giving any kind of feedback is the easiest way to contribute to an Open-Source Software (OSS) project. In GitHub, the largest code hosting service for OSS, this feedback is typically expressed as new issues for the project managed by an issue-tracking system available in each new project repository. Among other features, the issue tracker allows creating and assigning labels to issues with the goal of helping the project community to better classify and manage those issues (e.g., facilitating the identification of issues for top priority components or candidate developers that could solve them). Nevertheless, as the project grows a manual browsing of the project issues is no longer feasible.

- Contact: Gerson Sunyé

6.13. JSON Discoverer

FUNCTIONAL DESCRIPTION

The JSON discoverer allows you to discover the implicit schema of your JSON documents. Any JSON document includes both metadata (i.e., the schema) and data (i.e., the objects/values conforming to the schema). Given a (set of) JSON documents our discoverer analyzes the JSON definitions and generates for you a class diagram showing graphically the implicit JSON schema of your documents plus an object diagram representing their data.

- Participants: Jordi Cabot and Javier Canovas Izquierdo
- Contact: Hugo Brunelière
- URL: <http://atlanmod.github.io/json-discoverer/#/>

6.14. MODISCO

analyser, refactorer et comprendre le legacy

KEYWORDS: MDA - MDE - Model - Modernisation - Discoverer

SCIENTIFIC DESCRIPTION

MoDisco is an open source Eclipse project that provides a generic and extensible framework dedicated to the elaboration of Model Driven Reverse Engineering (MDRE) solutions. Gathering contributions from both academics and industrials, the goal of the project is to federate common efforts in the model-based transformation of legacy software systems implemented using different technologies (e.g. Java, COBOL, C). The first principle is to discover models out of legacy artifacts, representing appropriately all the relevant information, to be then used as part of reverse engineering processes for software understanding, evolution or modernization. Targeted scenarios include software (technical or architectural) migration of large legacy systems, but also retro-documentation, refactoring, quality assurance, etc. Within this context, MoDisco has collaborations with the OMG Architecture Driven Modernization (ADM) Task Force, for which the project provides several reference implementations of its standards: Knowledge Discovery Metamodel (KDM), Software Measurement Metamodel (SMM) and Abstract Syntax Tree Metamodel (ASTM).

The MoDisco framework is composed of a set of Eclipse plugins, and relies on the de-facto standard Eclipse Modeling Framework (EMF) for model handling. Thanks to its modular architecture, it allows completely covering the three steps of a standard MDRE approach: 1) Discovery (i.e. extracting a complete model of the source code), 2) Understanding (i.e. browsing and providing views on this model for a given purpose) and 3) Transformation (evolving the model towards a new technology, architecture, etc). More specifically, as part of its Infrastructure layer, MoDisco offers the set of generic (i.e., legacy technology-independent) reusable components really useful to build the core of MDRE solutions: Discovery Manager and Workflow for MDRE task orchestration, Model Browser for advanced navigation in complex models, model extension and customization capabilities for understanding (e.g. views definition), etc. As part of its Technologies layer, it provides an advanced support for the Java, JEE and XML technologies, including complete metamodels, corresponding model discoverers, transformations, code generators, customizations, query libraries, etc.

MoDisco (or some of its components) is being used by different partners including other academics, industrials (e.g. Sodifrance on several of their real modernization projects for their customers) or Eclipse projects (e.g. Eclipse-MDT Papyrus as developed by CEA). Moreover, the Eclipse-EMFT EMF Facet project has been initiated as a MoDisco spin-off, in order to externalize some features which are not actually specific to reverse engineering problems and thus may be reused in many different contexts (cf. corresponding EMF Facet section).

The initiative continues to be developed within the context of the European FP7-ICT project named ARTIST (<http://www.artist-project.eu/>), and also to a lower extent within the context of the French FUI 13 project named TEAP.

FUNCTIONAL DESCRIPTION

MoDisco is an Eclipse generic and extensible tool/framework providing to developers a set of reusable base components and dedicated to the elaboration of model driven reverse engineering (MDRE) solutions.

- Participant: Hugo Brunelière
- Partners: Ecole des Mines de Nantes - Mia-Software
- Contact: Hugo Brunelière
- URL: <http://www.eclipse.org/MoDisco/>

6.15. NeoEMF

FUNCTIONAL DESCRIPTION

NeoEMF is an open source software distributed under the terms of the Eclipse Public License that provides a backend-agnostic persistence solution for big, complex and highly interconnected EMF models. NeoEMF is a model repository and persistence framework allowing on-demand loading, storage, and unloading of large-scale EMF models.

- Partner: Mia-Software
- Contact: Gerson Sunyé
- URL: <http://www.neoemf.com/>

6.16. Reactive-ATL

KEYWORDS: Transformation - Reactive

SCIENTIFIC DESCRIPTION

Reactive engine for the ATL transformation language. In a model-driven application environment it works by activating only the strictly needed computation in response to updates or requests of model elements. Computation is updated when necessary, in an autonomous and optimized way by using incrementality and lazy evaluation.

- Participants: Salvador Martinez Perez and Massimo Tisi
- Contact: Salvador Martinez Perez
- URL: <http://atlanmod.github.io/org.eclipse.atl.reactive>

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 at the OS level for a single node and a set of nodes.

FUNCTIONAL DESCRIPTION

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

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

6.2. ELVIS

Extensible Log VISualization

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

SCIENTIFIC DESCRIPTION

The studies that were performed last year 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 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. In the PANOPESEC project, we will also perform some research on visualization for security monitoring in the context of SCADA systems.

FUNCTIONAL DESCRIPTION

ELVIS is a log visualization tool that allows analyst-friendly log explorations through automated selection of adequate representations. Many log formats can be used and it is quite simple to add new ones. ELVIs has been presented in VizSec 2013 (part of Vis 2013) in October in Atlanta.

- Participant: Nicolas Prigent
- Partner: CentraleSupélec
- Contact: Nicolas Prigent

6.3. GEPETO

GEoPrivacy-Enhancing TOolkit

KEYWORDS: 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 Moïse) 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: Université de Rennes 1 - CNRS
- 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: CentraleSupélec
- Contact: Eric Totel
- URL: <http://www.rennes.supelec.fr/ren/perso/etotel/GNG/index.html>

6.5. JBlare

FUNCTIONAL DESCRIPTION

JBlare is a Java Virtual Machine (JVM) hypervisor, able to track information flows inside Java programs. Being a modified JVM, it runs vanilla java applications. A cooperation mode with KBlare affords both IDS more precision. JBlare can use hybrid analysis combining dynamic analysis with static analysis using Soot.

- Contact: Guillaume Hiet
- URL: <https://www.blare-ids.org/flavors/jblare/>

6.6. 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/>

6.7. GroddDroid

Automatic Triggering of Android Malware

KEYWORDS: Malware analysis

SCIENTIFIC DESCRIPTION GroddDroid is a tool dedicated to the automatic triggering of suspicious code in Android applications. GroddDroid copes with a classical problem in dynamic analysis which is the triggering of malicious actions. To avoid dynamic analysis, malware authors develop some protections that delay the malicious executions. GroddDroid overrides these protections by modifying the bytecode of the infected applications and reconstructing the application. The modified application can thus be executed and monitored.

- Partner: CentraleSupélec
- Contact: Valérie Viet Triem Tong
- URL: <http://kharon.gforge.inria.fr/grodddroid.html>

COAST Project-Team

4. New Software and Platforms

4.1. Replication Benchmark

Participants: Pascal Urso [contact], Mehdi Ahmed Nacer, Gérald Oster.

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 events traces. These events traces can be produced randomly according to different parameters, can be extracted from real real-time editing session that have been recorded, or can be automatically extracted from distributed version control repositories such as the one produced with Git. Performances of the algorithms are measured in term of execution time, memory footprint and merge result quality (compared to manual merge history stored in git repositories). The source code of this evaluation framework is available at <https://github.com/coast-team/replication-benchmark/>.

4.2. MUTE

Participants: Claudia-Lavinia Ignat, Luc André, François Charoy, Gérald Oster [contact].

MUTE (Multi-User Text Editor) is a web-based text editing tool that allows to edit documents collaboratively in real-time. It implements our recent work on collaborative editing algorithms and more specifically the LOGOOT-SPLIT+ approach [17]. 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. This prototype is distributed under the term of GNU GPLv3 licence and is freely available at <https://github.com/coast-team/mute-demo/>. A demo server is hosted at <http://www.coedit.re/>.

4.3. OpenPaaS POC

Participants: Olivier Perrin [contact], Ahmed Bouchami.

The OpenPaaS Proof of Concept was presented during the final review of the FSN OpenPaaS project. Our contribution was dedicated to the authentication mechanism, and to the authorization framework. These two functionalities was delivered thanks to two RESTful services. The authentication service wrapped the LemonLDAP::NG product within a fully RESTful service, while the authorization service uses the reputation of people within the Enterprise Social Network, and computes the decision/rejection of access based on the trust level of the subject requesting the access. The source code can be obtained via request addressed to Olivier Perrin.

CTRL-A Team

6. New Software and Platforms

6.1. Heptagon/BZR

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

FUNCTIONAL DESCRIPTION

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

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

HEPTAGON has been used to build BZR⁰, which is an extension of the former with contracts constructs. These contracts allow to express dynamic temporal properties on the inputs and outputs of HEPTAGON node. These properties are then enforced, within the compilation of a BZR program, by discrete controller synthesis, using the SIGALI tool⁰. The synthesized controller is itself generated in HEPTAGON, allowing its analysis and compilation towards different target languages (C, Java, VHDL).

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

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

⁰<http://bzs.inria.fr>

⁰<http://www.irisa.fr/vertecs/Logiciels/sigali.html>

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], Maël Besson.

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, and Things using CoAP (<https://tools.ietf.org/html/rfc7252>). 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 *GM-IDL*, 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 GM-IDL, 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;
- Besides the client-server, publish/subscribe and tuple space paradigms, VSB also provides support for the data streaming paradigm;

- 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.

VSB is being developed within the H2020 CHOReVOLUTION project (see § 8.2.1.1) with additional support from the iCONNECT Inria ADT (see § 8.1.1.1). It is also based on previous development carried out in the FP7 CHOReOS project (<http://www.choreos.eu>). VSB is available for download under open source license.

6.3. SoundCity: Urban-scale Noise Monitoring using the Urban Civics Middleware

Participants: Valérie Issarny [contact], Fadwa Rebhi, Animesh Pathak, Sara Hachem.

URL: http://urbancivics.com/soundcity_app.html

The interest in noise pollution and its effects, especially in terms of its influence on citizens' health, has been increasing throughout the years, as illustrated by the plethora of available research and health studies. Traditionally, cities perform periodic noise monitoring to assess the noisiness of neighborhoods and update their city planning policies accordingly. However, in its current state, noise monitoring for city planning takes place once every several years and leverages only a few static sensors (e.g., BruitParif activity, <http://www.bruitparif.fr>). Additionally, the outcome is not representative of citizens' personal exposure to noise as it is coarse grained and requires preplanning to determine where and when the sensing should take place. It is worth mentioning that, even if we assume that the city administration is able to use applications running on the smart phones of the city employees, gathering data by following a pre-determined spatio-temporal schedule – although an excellent strategy in terms of the quality of data gathered – would be expensive and cumbersome to organize.

Noise sensing is also complemented by data modeling approaches to generate maps that allow citizens and governments to visualize the distribution of noise over a region of interest. Those techniques range from basic data plotting to data assimilation techniques that leverage large scale data analytics further enabling the simulation and estimation of missing noise values. However, assimilation techniques are usually restricted to static sensing with predetermined locations and sensing times. This approach limits the capacity of assimilation models to provide estimates with high accuracy.

A solution to the above issues is through mobile participatory sensing, where noise monitoring is no longer restricted to a few expensive sensors. Participatory sensed data can be then leveraged, along with data provided from static sensors, by the assimilation techniques to further reduce the errors in the simulated maps. Toward that goal, we have been building the Urban Civics middleware, which we have further customized for noise monitoring [20]. The refined Urban Civics comprises middleware solutions for noise sensing, crowd-sourcing and data assimilation with the data assimilation component being more specifically developed by the Inria CLIME team. The noise-sensing itself is performed through existing dedicated applications, integrated with our middleware. Such approach enables us to alleviate the burden of domain-specific development and exploit the knowledge of domain experts.

SoundCity and supporting Urban Civics middleware are developed in collaboration with the Inria CLIME team, Ambientic (FR) and the Civic Engine at Berkeley (USA) in the context of CityLab@Inria and Inria@SiliconValley.

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

Participants: Valérie Issarny [contact], Cristhian Parra Trepowski, Animesh Pathak.

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.

Although a large array of ICT tools exist to support citizens' engagement, their use in PB is still limited and scattered. Mostly, ICT have been leveraged for communication for promotion purposes (through multiple channels such as TV, radio and social media) and for facilitating voting for citizens (usually, with custom-made web sites or SMS).

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.

MYRIADS Project-Team

6. New Software and Platforms

6.1. ConPaaS

Hosting complex applications in the cloud

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

URL: <http://www.conpaas.eu/>

Status: Version 1.5.1

License: BSD

Presentation: ConPaaS [55] 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.

New features developed in 2015 include: a “Generic” service which allows the easy deployment and execution of arbitrary applications in ConPaaS; the “ConPaaS Nutshell edition” which allows users to deploy a fully-functional ConPaaS installation within a single VirtualBox VM; and the “ConPaaS Raspberry Pi edition” which allows users to deploy ConPaaS on a set of Raspberry Pi devices, paving the way toward the development of a fully-featured mobile edge cloud.

Active contributors (from the Myriads team): Ancuta Iordache, Genc Tato, Teodor Crivat, Guillaume Pierre.

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

6.2. GinFlow

Contact: Cédric Tedeschi, Cedric.Tedeschi@irisa.fr

Status: Version 1.0 released in open source, and registered at APP.

License: LGPL-3

Presentation: GinFlow is a decentralized workflow engine. It relies on a set of processes deployed over a cluster. Every task of the workflow is basically encapsulated into a GinFlow worker, workers being able to coordinate with others through read and write operations of a shared space containing the status of the workflow. GinFlow provides the ability to change the workflow logic on-the-fly upon the detection of a non-desired behavior within the execution of some of the tasks of the workflow.

Users typically submit a workflow through its JSON representation (both the by-default workflow and its alternate version if adaptation is needed). It can also use the API to describe its workflow in a more *programmatic* fashion.

Active contributors (from Myriads project-team): Matthieu Simonin, Cédric Tedeschi, Javier Rojas Balderrama.

Impact: GinFlow is a tool meant to provide support for workflow-based applications needing adaptation at run time. It is also targeted at offering a platform for future development and researches around the decentralized execution of workflows. It has been used in the framework of the DALHIS⁰ associate team, as a workflow template executor, integrated with the TIGRES workflow manager developed at the Lawrence Berkeley National Lab. It has been supported by the GinFlow ADT funded by Inria since 2014 (see Section 9.2.1).

⁰<http://project.inria.fr/dalhis>

6.3. Merkat

Contact: Nikolaos Parlavantzas, Nikolaos.Parlavantzas@irisa.fr

URL: <http://www.irisa.fr/myriads/software/Merkat/>

Status: Version 1.0

Presentation: Merkat is a market-based private PaaS (Platform-as-a-Service) system, supporting dynamic, fine-grained resource allocation and automatic application management [47], [46] [3]. Merkat implements a proportional-share auction that ensures maximum resource utilization while providing incentives to applications to regulate their resource usage. Merkat includes generic mechanisms for application deployment and automatic scaling. These mechanisms can be adapted to support diverse performance goals and application types, such as master-worker, MPI, or MapReduce applications. Merkat is implemented in Python and uses OpenNebula for virtual machine management. Experimental results on the Grid'5000 testbed show that using Merkat increases resource utilization and improves application performance. The development was initiated in the framework of Stefania Costache PhD's thesis.

Active contributors (from the Myriads team): Christine Morin, Nikolaos Parlavantzas.

Other contributors: Stefania Costache.

Impact: Merkat has been integrated in EDF R&D portal providing access to internal computing resources and is currently used on a testbed at EDF R&D.

6.4. Meryn

Contact: Nikolaos Parlavantzas, Nikolaos.Parlavantzas@irisa.fr

URL: <http://www.irisa.fr/myriads/software/Meryn/>

Status: Version 1.0

Presentation: 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 [49]. The current Meryn prototype was implemented using shell scripts, builds upon the Snooze VM manager software, and supports batch and MapReduce applications using respectively the Oracle Grid Engine OGE 6.2u7 and Hadoop 0.20.2 frameworks.

Active contributors (from the Myriads team): Christine Morin, Nikolaos Parlavantzas.

Other contributors: Djawida Dib.

Impact: Meryn is not yet distributed as an open source software.

6.5. Snooze

Contact: Christine Morin, Christine.Morin@inria.fr

URL: <http://snooze.inria.fr>

Status: Version 2.1.5

License: GPLv2

Presentation: Snooze [51], [50], [52] [4] 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 [54]), 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.

Snooze is fully implemented from scratch in Java and currently comprises approximately 15.000 lines of maintainable abstractions-based code. In order to provide a uniform interface to the underlying hypervisors and support transparent VM monitoring and management, Snooze integrates the *libvirt* virtualization library. Cassandra (since 2.0.0) can be used as base backend, providing reliability and scalability to the database management system. At a higher level Snooze provides its own REST API as well as an EC2 compatible API (since 2.1.0). It can thus be controlled from the command line (using the legacy client or an EC2 compatible tool), or from different language libraries (*libcloud*, *jcloud* ...). Snooze also provides a web interface to control the system. In collaboration with the Northeastern University of Boston we built the *Checkpoint as a Service* system on top of Snooze. The service allows users to execute their computations in a cloud environment in a reliable way. Periodic checkpoints are saved making it possible to restore the computation from a previous state in the event of failures. This work is described in [16].

Active contributors (from Myriads team): Yvon Jégou, David Margery, Christine Morin, Matthieu Simonin.

Other contributors: Jiajun Cao, Gene Cooperman, Eugen Feller.

Impact: Snooze has been used by students at LIFL, IRIT in France and LBNL in the US in the framework of internships. It has also been deployed and experimented at EDF R&D. Snooze entry won the 2nd prize of the scalability challenge at CCGrid 2013. Finally, we know that it was used by external users from academia and industry as we received feed-back from them. Snooze development was supported by the Snooze ADT funded by Inria from October 2012 to September 2014.

6.6. SimGrid

Scientific Instrument for the study of Large-Scale Distributed Systems.

Contact: Martin Quinson, Martin.Quinson@ens-rennes.fr

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

Status: Version 3.12

License: LGPL 2.1

Presentation: SimGrid is a toolkit providing core features 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.

Active contributors from Myriads project-team: Martin Quinson and Anne-Cécile Orgerie.

Other active contributors: Arnaud Legrand (Mescal project-team, CNRS & Inria Rhone-Alpes), Frédéric Suter (Avalon project-team, CNRS & Inria Rhone-Alpes).

Impact: SimGrid has an active user community of more than one hundred members, mainly composed of researchers and students. In the last decade only, it grounded the experiments of 6 PhDs works, 25 journal papers, and over 100 conference papers. The community gathers every year during the SimGrid User's Days workshop.

SimGrid is also used to run the regression and performance tests and tuning of two large applications: BigDFT (a massively parallel code computing the electronic structure of chemical elements developed by the CEA) and StarPU (a Unified Runtime System for Heterogeneous Multicore Architectures developed by Inria Bordeaux). Both of these programs enjoy large user communities themselves.

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 [37], [50], for elastic version management, for adaptive replication, for partial replication, and for reconfigurable sharding.

- Participants: Tyler Crain, Marc Shapiro, Serdar Tasiran 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, Alejandro Tomsic
- 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: Marc Shapiro
- URL: <https://scm.gforge.inria.fr/anonscm/git/transgc/>.

5.4. SwiftCloud

FUNCTIONAL DESCRIPTION

Client-side (e.g., mobile or in-browser) apps need local access to shared cloud data, but current technologies either do not provide fault-tolerant consistency guarantees, or do not scale to high numbers of unreliable and resource-poor clients, or both. Addressing this issue, the SwiftCloud distributed object database supports high numbers of client-side partial replicas. SwiftCloud offers fast reads and writes from a causally-consistent client-side cache. It is scalable, thanks to small and bounded metadata, and available, tolerating faults and intermittent connectivity by switching between data centres. The price to pay is a modest amount of staleness. A recent Inria Research Report (submitted for publication) presents the SwiftCloud algorithms, design, and experimental evaluation, which shows that client-side apps enjoy the same guarantees as a cloud data store, at a small cost.

- Participants: Marc Shapiro, Serdar Tasiran, Marek Zawirski and Mahsa Najafzadeh
- Contact: Marc Shapiro
- URL: <git+ssh://scm.gforge.inria.fr//gitroot/swiftcloud>

5.5. PUMA

FUNCTIONAL DESCRIPTION

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

- Participants: Maxime Lorrillere, Sébastien Monnet, Pierre Sens, Julien Sopena
- Contact: Maxime Lorrillere
- URL: <https://github.com/mlorrillere/puma>

SCALE Team

6. New Software and Platforms

6.1. BigGrph

- Participants: Eric Madelaine, Fabrice Huet
- Contact: Luc Hogie
- The objective of “biggrph” 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 24,635,412.

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

The development of the “biggrph” platform is now at the stage 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 have also solved problems of scalability, in particular concerning the communication layer with billions of messages exchanged between BSP steps. Moreover, we 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 (a facility maintained at Inria Sophia Antipolis).

- This project is a joint work of the three EPs Coati, Diana and Scale and is supported by an ADT grant.
- URL : <http://www.i3s.unice.fr/~hogie/software/?name=biggrph>

6.2. BtrPlace

FUNCTIONAL DESCRIPTION

BtrPlace dynamically adapts the VM placement depending on pluggable expectations using a network and memory-aware migration scheduler. It currently addresses affinity constraints, resource booking, node state manipulation and hosting restrictions. BtrPlace is a complete rewrite of the reconfiguration algorithm that was inside OW2 project Entropy with a clear focus on extensibility. It embeds the constraint programming library Choco to compute solutions.

BtrPlace has been released 5 times this year. The current code amounts for 44000 lines of Java (production and test code, comments) and the online documentation amounts for around 1500 lines. BtrPlace is available from the Maven central repository. In 2015, it has been downloaded around 480 times from unique IPs and 2400 unique visitors accessed the Website.

- Contact: Fabien Hermenier
- Participants: Fabien Hermenier, Vincent Kherbache
- URL: <http://www.btrplace.org/>
- APP registration code: IDDN.FR.001.330025.000.S.C.2014.000.31235

6.3. EventCloud

SCIENTIFIC DESCRIPTION

The EventCloud architecture is based on a structured P2P overlay network targeting high-performance elastic data processing. Consequently it aims to be deployed on infrastructures like grids, clouds, i.e. whose nodes acquisition and relinquishment can be dynamic and subject to a pay-per-use mode. Each node participating in the overlay networks constituting EventCloud instances, is responsible for managing the storage of subsets of the events, and helps in matching potential looked up events and disseminating them in a collaborative manner. As such, each node is also potentially an event broker responsible for managing subscriptions and routing notifications. The EventCloud provides a high level publish-subscribe API where users can register their interests using SPARQL. When matching RDF data are added, subscribers are automatically notified. Recent work around the EventCloud has focused on efficient algorithms for managing subscription and notification.

FUNCTIONAL DESCRIPTION

The EventCloud is an open source middleware that aims to act as a distributed datastore for data fulfilling the W3C RDF specification (<http://www.w3.org/RDF/>). It allows to store and retrieve quadruples (RDF triples with context) through SPARQL but also to manage events represented as quadruples.

- Participants: Laurent Pellegrino, Fabrice Huet, Françoise Baude, Maeva Antoine and Iyad Alshabani
- Partner: CNRS
- Contact: Françoise Baude

6.4. OSA

Open Simulation Architecture

FUNCTIONAL DESCRIPTION

OSA on of new or existing contributions at every level of its architecture. The platform core supports discrete-event simulation engine(s) built on top of the ObjectWeb Consortium's Fractal component model. In OSA, the systems to be simulated are modeled and instrumented using Fractal components. In OSA, the event handling is mostly hidden in the controller part of the components, which alleviates noticeably the modeling process, but also eases the replacement of any part of the simulation engine. Apart the simulation engine, OSA aims at integrating useful tools for modeling, developing, experimenting, and analysing simulations. OSA is also a platform for experimenting new techniques and approaches in simulation, such as aspect oriented programming, separation of concerns, innovative component architectures, and so on.

- Participant: Olivier Dalle
- Contact: Olivier Dalle
- URL: <http://osa.inria.fr/>

6.5. Vercors Component Editor (VCE)

VERification of models for distributed communicating COmponents, with safety and Security

FUNCTIONAL DESCRIPTION

The Vercors tools include front-ends for specifying the architecture and behaviour of components in the form of UML diagrams. We translate these high-level specifications, into behavioural models in various formats, and we also transform these models using abstractions. In a final step, abstract models are translated into the input format for various verification toolsets. Currently we mainly use the various analysis modules of the CADP toolset.

- Participants: Eric Madelaine, Antonio Cansado, Ludovic Henrio, Marcela Rivera, Oleksandra Kulkankhina, Bartlomiej Szejna, Nassim Jibai and Siqu Li
- Contact: Eric Madelaine
- URL: <http://team.inria.fr/scale/software/vercors/>

SPIRALS Project-Team

6. New Software and Platforms

6.1. APISENSE®

Participants: Clive Ferret-Canape, Julien Duribreux, Maria Gomez Lacruz, Christophe Ribeiro, Romain Rouvoy, Antoine Veuille.

- Contact: Romain Rouvoy
- URL: <https://bil.inria.fr/fr/software/view/614/tab>

In 2015, APISENSE® has been extended to include the support for iOS smartphones and tablets. The infrastructure, hosted by the LHS (*Laboratoire Haute Sécurité*), is now fully secured to protect the privacy of contributors.

APISENSE® is a distributed platform dedicated to crowd-sensing activities. Crowd-sensing intends to leverage mobile devices to seamlessly collect valuable dataset for different categories of stakeholders. APISENSE® intends to be used in a wide variety of scientific and industrial domains, including network quality monitoring, social behavior analysis, epidemy predictions, emergency crisis support, open maps initiatives, debugging of applications in the wide. APISENSE® is composed of HIVE delivered as a *Platform-as-a-Service* (PaaS) to the stakeholders who can pilot and customize their own crowd-sensing environment [79], and *Bee* supporting participants with a mobile application to control the sensors to be shared with the rest of the world [68], [69]. The platform is used by the *MetroScope* consortium, an Internet scientific observatory initiative supported by Inria. APISENSE® originates as the output of Nicolas Haderer PhD thesis [67].

APISENSE® is at the core of the Inria ADT Focus CrowdLab project (see Section 9.2) and of an industrial transfer action that aims at creating the Crowdify spin-off company.

Registered with the APP (*Agence pour la Protection des Programmes*) under reference FR.001.080006.000.S.P.2013.000.10000. License: Proprietary.

6.2. PowerAPI

Participants: Maxime Colmant, Loïc Huertas, Romain Rouvoy.

- Contact: Romain Rouvoy
- URL: <https://bil.inria.fr/fr/software/view/1642/tab>

In 2015, POWERAPI has been further extended to support the monitoring of *Software-as-a-Service* (SaaS) [23] by including a support for hypervision technologies like KVM (*Kernel-based Virtual Machine*), and Docker.

POWERAPI is a Scala-based library for monitoring energy in software systems. It is based on a modular and asynchronous event-driven architecture using the Akka library. POWERAPI differs from existing energy process-level monitoring tool in its pure software, fully customizable and modular aspect which let users precisely define what they want to monitor, without plugging any external device. POWERAPI offers an API which can be used to express requests about energy spent by a process, following its hardware resource utilization (in terms of CPU, memory, disk, network, etc.). Its applications cover energy-driven benchmarking [75], [62], [60], [61], energy hotspots and bugs detection [76], [77], and real-time distributed system monitoring. POWERAPI originates as the output of Adel Nouredine PhD thesis [78].

POWERAPI is at the core of the Inria ADT eSurgeon project (see Section 9.2).

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

6.3. Saloon

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

- Contact: Lionel Seinturier
- URL: <https://bil.inria.fr/fr/software/view/1492/tab>

In 2015, SALOON was extended to fit with the requirements of the FP7 PaaSage project (see Section 9.3). New mechanisms were added to deal with ontologies for describing cloud computing commercial offers.

SALOON is a framework for the selection and configuration of Cloud providers according to application requirements. The framework enables the specification of such requirements by defining ontologies. Each ontology provides a unified vision of provider offers in terms of frameworks, databases, languages, application servers and computational resources (i.e., memory, storage and CPU frequency). Furthermore, each provider is related to a Feature Model (FM) with attributes and cardinalities, which captures its capabilities. By combining the ontology and FMs, the framework is able to match application requirements with provider capabilities and select a suitable one. Specific scripts to the selected provider are generated in order to enable its configuration. SALOON originates as the output of Clément Quinton PhD thesis [82].

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

6.4. Spoon

Participants: Thomas Durieux, Matias Martinez, Martin Monperrus, Gérard Paligot, Nicolas Petitprez.

- Contact: Martin Monperrus
- URL: <https://bil.inria.fr/fr/software/view/251/tab>

SPOON is a library for analyzing and transforming Java source code [18] [80]. In 2015, there was one engineer (Géradar Paligot) working full-time on Spoon. This has incredibly boosted the project. In particular, the new features of SPOON are: a) the complete support for Java 8 (incl. lambda expressions) b) the support for Maven and Gradle build systems. Also, a major testing effort has been done and consequently, the code quality now meets the highest standards, using a comprehensive test suite as well three continuous integration servers (incl. the Inria one). Spoon has an international community of users and contributors, which is very active on the Github project.

SPOON is at the core of the Inria ADT Spoon3R project (see Section 9.1) SPOON is the foundation behind an industrial transfer action that aims at creating the Spoonware spin-off company on automatic hot patching.

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

WHISPER Project-Team

6. New Software and Platforms

6.1. Platforms

6.1.1. Coccinelle

Our recent research is in the area of code manipulation tools for C code, particularly targeting Linux kernel code. This work has led to the Coccinelle tool that we are continuing to develop. Coccinelle serves both as a basis for our future research and the foundation of our interaction with the Linux developer community.

The need to find patterns of code, and potentially to transform them, is pervasive in software development. Examples abound. When a bug is found, it is often fruitful to see whether the same pattern occurs elsewhere in the code. For example, the recent Heartbleed bug in OpenSSL partly involves the same fragment of code in two separate files.⁰ Likewise, when the interface of an API function changes, all of the users of that function have to be updated to reflect the new usage requirements. This generalizes to the case of code modernization, in which a code base needs to be adapted to a new compiler, new libraries, or a new coding standards. Finding patterns of code is also useful in code understanding, *e.g.*, to find out whether a particular function is ever called with a particular lock held, and in software engineering research, *e.g.*, to understand the prevalence of various kinds of code structures, which may then be correlated with other properties of the software. For all of these tasks, there is a need for an easy to use tool that will allow developers to express patterns and transformations that are relevant to their source code, and to apply these patterns and transformations to the code efficiently and without disrupting the overall structure of the code base.

The Coccinelle program matching and transformation tool for C code addresses these needs. Coccinelle has been under development for over 10 years, and is mature software, available in a number of Linux distributions (Ubuntu, Debian, Fedora, etc.). It allows matching and transformation rules to be expressed in terms of fragments of C code, more precisely in the form of a *patch*, in which code to add and remove is highlighted by using + and -, respectively, in the leftmost column, and other, unannotated, code fragments may be provided to describe properties of the context. The C language is extended with a few operators, such as metavariables, for abstracting over subterms, and a notion of positions, which are useful for reporting bugs. The pattern matching rules can interspersed with rules written in Python or OCaml, for further expressiveness. The process of matching patterns against the source code furthermore takes into account some semantic information, such as the types of expressions and reachability in terms of a function's (intraprocedural) control-flow graph, and thus we refer to Coccinelle matching and transformation specifications as *semantic patches*.

Coccinelle was originally motivated by the goal of modernizing Linux 2.4 drivers for use with Linux 2.6, and was originally validated on a collection of 60 transformations that had been used in modernizing Linux 2.4 drivers [8]. Subsequent research involving Coccinelle included a formalization of the logic underlying its implementation [1] and a novel mechanism for identifying API usage protocols [45]. More recently, Coccinelle has served as a practical and flexible tool in a number of research projects that somehow involve code understanding or transformation. These include identifying misuses of named constants in Linux code [47], extracting critical sections into procedures to allow the implementation of a centralized locking service [53], generating a debugging interface for Linux driver developers [23], detecting resource release omission faults in Linux and other infrastructure software [64], and understanding the structure of device driver code in our current DrGene project [66].

⁰<http://git.openssl.org/gitweb/?p=openssl.git;a=commitdiff;h=96db902>

Throughout the development of Coccinelle, we have also emphasized contact with the developer community, particularly the developers of the Linux kernel. We submitted the first patches to the Linux kernel based on Coccinelle in 2007. Since then, over 4500 patches have been accepted into the Linux kernel based on the use of Coccinelle, including around 3000 by over 500 developers from outside our research group. Over 50 semantic patches are available in the Linux kernel source code itself, with appropriate infrastructure for developers to apply these semantic patches to their code within the normal make process. Many of these semantic are also included in a 0-day build-testing system for Linux patches maintained by Intel.⁰ Julia Lawall was invited to the Linux Kernel Summit as a core attendee (invitation only) in 2010 and 2014, and has been invited to the internal 2014 SUSE Labs Conference. She has also presented Coccinelle at developer events such as LinuxCon Europe, Kernel Recipes (Paris), FOSDEM (Brussels), and RTWLS, and has supervised three interns using Coccinelle financed by the Linux Foundation, as part of the Outreachy internship program.

Finally, we are aware of several companies that use Coccinelle for modernizing code bases. These include Metaware in Paris, with whom we had a 5-month contract in 2013-2014 for the customization and maintenance of Coccinelle. We hope to be able to organize other such contracts in the future.

6.1.2. Better Linux

Over the past few years, Julia Lawall and Gilles Muller have designed and developed a number of tools such as Coccinelle, *Diagnosys* [23] [22] and *Hector* [64], to improve the process of developing and maintaining systems code. The *BtrLinux* action aims to increase the visibility of these tools, and to highlight Inria's potential contributions to the open source community. We are developing a web site <https://btrlinux.inria.fr/>, to centralize the dissemination of the tools, collect documentation, and collect results. This action is supported by Inria by the means of a young engineer (ADT), Quentin Lambert. In the case of Coccinelle, we will focus on enhancing its visibility and its dissemination, by using it to find and fix faults in Linux kernel code, and by submitting the resulting patches to the Linux maintainers. Our work on *Diagnosys* and *Hector* is described below.

Diagnosys is a hybrid static and dynamic analysis tool that first collects information about Linux kernel APIs that may be misused, and then uses this information to generate wrapper functions that systematically log at runtime any API invocations or return values that may reflect such misuse. A developer can then use a specific make-like command to build an executable driver that transparently uses these wrapper functions. At runtime, the wrappers write log messages into a crash resilient region of memory that the developer can inspect after any crash. *Diagnosys* is complementary to Coccinelle in the kind of information that it provides to developers. While Coccinelle directly returns a report for every rule match across the code base, often including false positives that have to be manually isolated by the developer, *Diagnosys* only reports on conditions that occur in the actual execution of the code. *Diagnosys* thus produces less information, but the information produced is more relevant to the particular problem currently confronting the developer. As such, it is well suited to the case of initial code development, where the code is changing frequently, and the developer wants to debug a specific problem, rather than ensuring that the complete code base is fault free. *Diagnosys* is a complete functioning system, but it needs to be kept up to date with changes in the kernel API functions. As part of the *BtrLinux* action, we will regularly run the scripts that collect information about how to create the wrappers, and then validate and make public the results.

Hector addresses the problem of leaking resources in error-handling code. Releasing resources when they are no longer needed is critical, so that adequate resources remain available over the long execution periods characteristic of systems software. Indeed, when resource leaks accumulate, they can cause unexpected resource unavailability, and even single leaks can put the system into an inconsistent state that can cause crashes and open the door to possible attacks. Nevertheless, developers often forget to release resources, because doing so often does not make any direct contribution to a program's functionality. A major challenge in detecting resource-release omission faults is to know when resource release is required. Indeed, the C language does not provide any built-in support for resource management, and thus resource acquisition and release are typically implemented using ad hoc operations that are, at best, only known to core developers.

⁰E.g., <http://comments.gmane.org/gmane.linux.kernel.kbuild/269>

Previous work has focused on mining sequences of such functions that are used frequently across a code base, [35], [51] but these approaches have very high rates of false negatives and false positives [48]. We have proposed Hector, a static analysis tool that finds resource-release omission faults based on inconsistencies in the operations performed within a single function, rather than on usage frequency. This strategy allows Hector to have a low false positive rate, of 23% in our experiments, while still being able to find hundreds of faults in Linux and other systems.

Hector was developed as part of the PhD thesis of Suman Saha and was presented at DSN 2013, where it received the William C. Carter award for the best student paper. Hector is complementary to Coccinelle, in that it has a more restricted scope, focusing on only one type of fault, but it uses a more precise static analysis, tailored for this type of fault, to ensure a low false positive rate. Hector, like Coccinelle, is also complementary to Diagnosys, in that it exhaustively reports on faults in a code base, rather than only those relevant to a particular execution, and is thus better suited for use by experienced developers of relatively stable software. Over 70 patches have been accepted into Linux based on the results of Hector. The current implementation, however, is somewhat in a state of disarray. As part of the BtrLinux action, we are currently working on returning the code to working condition and then will actively use it to find faults in Linux. Based on these results, we will either submit appropriate patches to the Linux developers or notify the relevant developer when the corresponding fix is not clear.

6.2. New Software

6.2.1. *coq-bitset library*

As part of Arthur Blot's internship, we have developed the `coq-bitset` library, a certified library implementing bitsets in the Coq proof assistant [17]. It enables abstract and formal reasoning about efficient low-level code within a proof assistant, thus paving the way for further certified results in the field of low-level system code (such as device drivers).

As part of this effort, we have also extended a pre-existing formalization of bit vectors in Coq [41] with a trustworthy extraction mechanism. This enables manipulating and reasoning about native integers in the Coq proof assistant, while supporting an efficient execution in OCaml.

Both libraries have been made available on Github as well as on the Coq-opam repository.

ALPINES Project-Team

6. New Software and Platforms

6.1. 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 2d-tree-based interpolation algorithm and a language for the manipulation of data on multiple meshes (as a follow up of bamg (now a part of FreeFem++)).

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

FUNCTIONAL DESCRIPTION

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

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

6.2. HPDDM

SCIENTIFIC DESCRIPTION

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

FUNCTIONAL DESCRIPTION

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

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

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/>

Submodules:

- Sparse Toolbox
KEYWORDS: Preconditioner - Interactive method - Linear system
 - Participants: Laura Grigori and Rémi Lacroix
 - Contact: Laura Grigori
 - not yet publicly available

- 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, Rémi Lacroix and Frédéric Nataf
 - Partners: CNRS - UPMC
 - Contact: Laura Grigori
 - not yet publicly available
- LORASC preconditioner
KEYWORD: Preconditioner
 - Participants: Laura Grigori and Rémi Lacroix
 - Contact: Laura Grigori
 - URL: not yet publicly available
 - NFF Nested Filtering Factorization
KEYWORDS: Preconditioner - Interactive method - Linear system
 - Participants: Laura Grigori, Frédéric Nataf and Long Qu
 - Partners: UPMC - Université Paris-Sud
 - Contact: Laura Grigori
 - not yet publicly available

AVALON Project-Team

6. New Software and Platforms

6.1. Active Data

Participants: Gilles Fedak [correspondant], Anthony Simonet.

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>

6.2. BitDew

Participants: Gilles Fedak [correspondant], Anthony Simonet.

FUNCTIONAL DESCRIPTION

The BitDew framework is a programmable environment for management and distribution of data for Grid, Desktop Grid and Cloud Systems. BitDew offers programmers a simple API for creating, accessing, storing and moving data with ease, even on highly dynamic and volatile environments. The BitDew programming model relies on 5 abstractions to manage the data : i) replication indicates how many occurrences of a data should be available at the same time on the network, ii) fault-tolerance controls the policy in presence of machine crash, iii) lifetime is an attribute absolute or relative to the existence of other data, which decides the life cycle of a data in the system, iv) affinity drives movement of data according to dependency rules, v) protocol gives the runtime environment hints about the protocol to distribute the data (http, ftp or bittorrent). Programmers define for every data these simple criteria, and let the BitDew runtime environment manage operations of data creation, deletion, movement, replication, and fault-tolerance operation.

- URL: <http://www.bitdew.net>

6.3. DIET

Participants: Daniel Balouek Thomert, Eddy Caron [correspondant], Maurice Faye, Arnaud Lefray.

FUNCTIONAL DESCRIPTION

Middleware for grids and clouds. Toolbox for the use and porting of intensive computing applications on heterogeneous architectures.

In 2015 we have published a new release of DIET. A short list of the major improvements over the version 2.9 of DIET:

Security: SSL communications are now available.

Cloud: some improvements

More information are provided to the user

OAR binding improvement

QuickStart documentation is provided

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

6.4. Kwapi

Participants: Laurent Lefèvre [correspondant], Jean-Patrick Gelas, Laurent Pouilloux.

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.

- URL: <https://launchpad.net/kwapi>

6.5. DirectL2C

Participants: Vincent Lanore, Christian Perez [correspondant].

KEYWORDS: HPC, Software Components, Reconfiguration

FUNCTIONAL DESCRIPTION

L2C (<http://hlcm.gforge.inria.fr/l2c:start>) 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.

In 2015, we have prototyped DirectL2C on top of L2C. DirectL2C enables efficient and consistent reconfiguration of large scale L2C based assemblies. It provides an assembly model enhanced with domains, transformations, and transformation adapters.

- URL: <http://hlcm.gforge.inria.fr/l2c:start>

6.6. Sam4C

Participants: Eddy Caron [correspondant], Arnaud Lefray.

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 (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.

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

6.7. SimGrid

Participant: Frédéric Suter [correspondant].

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

FUNCTIONAL DESCRIPTION

Scientific Instrument for the study of Large-Scale Distributed Systems. SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments. In 2015, we published a new release, SimGrid 3.12.

- Partners: CNRS - Ecole Normale Supérieure de Rennes - University of Hawaii - Université de Reims Champagne-Ardenne - Femto-st
- URL: <http://simgrid.gforge.inria.fr/>

6.8. execo

Participants: Matthieu Imbert [correspondant], Laurent Pouilloux.

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.

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

6.9. Grid'5000 Experimental Platform

Participants: Laurent Lefèvre [correspondant], Simon Delamare, David Loup, Christian Perez, Marc Pinhède, Laurent Pouilloux.

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.

- URL: <https://www.grid5000.fr/mediawiki/index.php/Grid5000:Home>

HIEPACS Project-Team

6. New Software and Platforms

6.1. Introduction

We describe in this section the software that we are developing. The first list will be the main milestones of our project. The other software developments will be conducted in collaboration with academic partners or in collaboration with some industrial partners in the context of their private R&D or production activities. For all these software developments, we will use first the various (very) large parallel platforms available through GENCI in France (CCRT, CINES and IDRIS Computational Centers), and next the high-end parallel platforms that will be available via European and US initiatives or projects such that PRACE.

6.2. Chameleon

Participant: Emmanuel Agullo [corresponding member].

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.

6.3. MaPHyS

Participant: Emmanuel Agullo [corresponding member].

MaPHyS (Massively Parallel Hybrid Solver) is a software package that implements a parallel linear solver coupling direct and iterative approaches. The underlying idea is to apply to general unstructured linear systems domain decomposition ideas developed for the solution of linear systems arising from PDEs. The interface problem, associated with the so called Schur complement system, is solved using a block preconditioner with overlap between the blocks that is referred to as Algebraic Additive Schwarz.

The **MaPHyS** package is very much a first outcome of the research activity described in Section 3.3 . Finally, **MaPHyS** is a preconditioner that can be used to speed-up the convergence of any Krylov subspace method. We foresee to either embed in **MaPHyS** some Krylov solvers or to release them as standalone packages, in particular for the block variants that will be some outcome of the studies discussed in Section 3.3 .

MaPHyS can be found at <http://maphys.gforge.inria.fr>.

6.4. PaStiX

Participant: Pierre Ramet [corresponding member].

Complete and incomplete supernodal sparse parallel factorizations.

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** library uses the graph partitioning and sparse matrix block ordering package **Scotch**. **PaStiX** is based on an efficient static scheduling and memory manager, in order to solve 3D problems with more than 50 million of unknowns. The mapping and scheduling algorithm handles a combination of 1D and 2D block distributions. This algorithm computes an efficient static scheduling of the block computations for our supernodal parallel solver which uses a local aggregation of contribution blocks. This can be done by taking into account very precisely the computational costs of the BLAS 3 primitives, the communication costs and the cost of local aggregations. We also improved this static computation and communication scheduling algorithm to anticipate the sending of partially aggregated blocks, in order to free memory dynamically. By doing this, we are able to reduce the aggregated memory overhead, while keeping good performance.

Another important point is that our study 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.

Direct methods are numerically robust methods, but the very large three dimensional problems may lead to systems that would require a huge amount of memory despite any memory optimization. A studied approach consists in defining an adaptive blockwise incomplete factorization that is much more accurate (and numerically more robust) than the scalar incomplete factorizations commonly used to precondition iterative solvers. Such incomplete factorization can take advantage of the latest breakthroughs in sparse direct methods and particularly should be very competitive in CPU time (effective power used from processors and good scalability) while avoiding the memory limitation encountered by direct methods.

PaStiX is publicly available at <http://pastix.gforge.inria.fr> under the Inria CeCILL licence.

6.5. HIPS

Participant: Pierre Ramet [corresponding member].

Multilevel method, domain decomposition, Schur complement, parallel iterative solver.

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). This special ordering and partitioning allows for the use of dense block algorithms both in the direct and iterative part of the solver and provides a high degree of parallelism to these algorithms. The code provides a hybrid method which blends direct and iterative solvers. **HIPS** exploits the partitioning and multistage ILU techniques to enable a highly parallel scheme where several subdomains can be assigned to the same process. It also provides a scalar preconditioner based on the multistage ILUT factorization.

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

HIPS is publicly available at <http://hips.gforge.inria.fr> under the Inria CeCILL licence.

6.6. MetaPart

Participant: Aurélien Esnard [corresponding member].

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 software packages such as **Scotch**, PaToH, METIS, Zoltan, Mondriaan, etc. 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).

The framework is publicly available at Inria Gforge: <http://metapart.gforge.inria.fr>.

6.7. MPICPL

Participant: Aurélien Esnard [corresponding member].

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

MPICPL was developed by HIEPACS for the purpose of the ANR NOSSI. It uses advanced features of MPI2 standard. The framework is publicly available at Inria Gforge: <http://mpicpl.gforge.inria.fr>.

6.8. ScalFMM

Participant: Olivier Coulaud [corresponding member].

ScalFMM (Parallel Fast Multipole Library for Large Scale Simulations) is a software library to simulate N-body interactions using the Fast Multipole Method.

ScalFMM intends to offer all the functionalities needed to perform large parallel simulations while enabling an easy customization of the simulation components: kernels, particles and cells. It works in parallel in a shared/distributed memory model using OpenMP and MPI. The software architecture has been designed with two major objectives: being easy to maintain and easy to understand. There are two main parts: 1) the management of the octree and the parallelization of the method ; 2) the kernels. This new architecture allows us to easily add new FMM algorithm or kernels and new paradigm of parallelization. The code is extremely documented and the naming convention fully respected. Driven by its user-oriented philosophy, **ScalFMM** is using CMAKE as a compiler/installer tool. Even if **ScalFMM** is written in C++ it will support a C and fortran API soon.

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

The **ScalFMM** package is available at <http://scalform.gforge.inria.fr>

6.9. ViTE

Participant: Mathieu Faverge [corresponding member].

Visualization, Execution trace

ViTE is a trace explorer. It is a tool made to visualize execution traces of large parallel programs. It supports Pajé, a trace format created by Inria Grenoble, and OTF and OTF2 formats, developed by the University of Dresden and allows the programmer a simpler way to analyse, debug and/or profile large parallel applications. It is an open source software licenced under CeCILL-A.

The **ViTE** software is available at <http://vite.gforge.inria.fr> and has been developed in collaboration with the Inria Bordeaux - Sud-Ouest SED team, Telecom SudParis and Inria Grenoble.

In the same context we also contribute to the EZtrace and GTG libraries in collaboration with F. Trahay from Telecom SudParis. EZTrace (<http://eztrace.gforge.inria.fr>) is a tool that aims at generating automatically execution trace from HPC programs. It generates execution trace files thanks to the GTG library (<http://gtg.gforge.inria.fr>) that can be later interpreted by visualization tools such as **ViTE**.

6.10. Other software

For the materials physics applications, a lot of development will be done in the context of older ANR projects (**NOSSI** and **OPTIDIS**) in collaboration with LaBRI, CPMOH, IPREM, EPFL and with CEA Saclay and Bruyère-le-Châtel.

- **OptiDis**

Participant: Olivier Coulaud [corresponding member].

OptiDis is a new code for large scale dislocation dynamics simulations. Its aim is to simulate real life dislocation densities (up until $5 \cdot 10^{22}$ dislocations/ m^{-2}) in order to understand plastic deformation and study strain hardening. The main application is to observe and understand plastic deformation on irradiated zirconium. Zirconium alloys is the first containment barrier against the dissemination of radioactive elements. More precisely, with neutron irradiated zirconium alloys we are talking of channeling mechanism, which means to stick with the reality, more than tens of thousands of induced loops so 10^8 degrees of freedom in the simulation.

The code is based on Numodis code developed at CEA Saclay and the **ScalFMM** library developed in our Inria project. The code is written in C++ language and using the last features of C++11. One of the main aspects is the hybrid parallelism MPI/OpenMP that gives the software the ability to scale on large cluster while the computation load rises. In order to achieve that, we use different levels of parallelism. First of all, the simulation box is spread over MPI processes, we then use a thinner level for threads, dividing the domain using an Octree representation. All these parts are driven by the **ScalFMM** library. On the last level our data are stored in an adaptive structure absorbing dynamic of this kind of simulation and handling well task parallelism.

The two following packages are mainly designed and developed in the context of a US initiative led by ICL and to which we closely collaborate through the associate team **MORSE**.

- **PLASMA**

Participant: Mathieu Faverge [corresponding member].

The **PLASMA** (Parallel Linear Algebra for Scalable Multi-core Architectures) project aims at addressing the critical and highly disruptive situation that is facing the Linear Algebra and High Performance Computing community due to the introduction of multi-core architectures.

The **PLASMA** ultimate goal is to create software frameworks that enable programmers to simplify the process of developing applications that can achieve both high performance and portability across a range of new architectures.

The development of programming models that enforce asynchronous, out of order scheduling of operations is the concept used as the basis for the definition of a scalable yet highly efficient software framework for Computational Linear Algebra applications.

The **PLASMA** library is available at <http://icl.cs.utk.edu/plasma>.

- **PaRSEC/DPLASMA**

Participant: Mathieu Faverge [corresponding member].

PaRSEC Parallel Runtime Scheduling and Execution Controller, is a generic framework for architecture aware scheduling and management of micro-tasks on distributed many-core heterogeneous architectures. Applications we consider can be expressed as a Direct Acyclic Graph of tasks with labeled edges designating data dependencies. DAGs are represented in a compact problem-size independent format that can be queried on-demand to discover data dependencies in a totally distributed fashion. **PaRSEC** assigns computation threads to the cores, overlaps communications and computations and uses a dynamic, fully-distributed scheduler based on architectural features such as NUMA nodes and algorithmic features such as data reuse.

The framework includes libraries, a runtime system, and development tools to help application developers tackle the difficult task of porting their applications to highly heterogeneous and diverse environments.

DPLASMA (Distributed Parallel Linear Algebra Software for Multicore Architectures) is the leading implementation of a dense linear algebra package for distributed heterogeneous systems. It is designed to deliver sustained performance for distributed systems where each node featuring multiple sockets of multicore processors, and if available, accelerators like GPUs or Intel Xeon Phi. **DPLASMA** achieves this objective through the state of the art **PaRSEC** runtime, porting the **PLASMA** algorithms to the distributed memory realm.

The **PaRSEC** runtime and the **DPLASMA** library are available at <http://icl.cs.utk.edu/parsec>.

6.11. Platforms

6.11.1. PlaFRIM: an experimental parallel computing platform

PLAFRIM is an open experimental platform for research and development 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. Since mid-2015, this platform is now open to laboratories involved in the CPU cluster and SMIs located 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 2,500 cores (excluding GPU and Xeon Phi) that are available for all research teams of Inria Bordeaux, Labri and IMB. Those computers are in particular used by all the engineers who work in HiePACS and are advised by F. Rue from the SED.

The PlaFRIM platform initiative is coordinated by O. Coulaud.

KERDATA Project-Team

6. New Software and Platforms

6.1. Major Software

6.1.1. BlobSeer

Participants: Alexandru Costan, Gabriel Antoniu, Luc Bougé, Loïc Cloatre.

Contact: Gabriel Antoniu.

Presentation: BlobSeer is the core software platform for many current cloud-oriented projects of the KerData team. It is a data storage service specifically designed to deal with the requirements of large-scale, data-intensive distributed applications that abstract data as huge sequences of bytes, called BLOBs (Binary Large Objects). It provides a versatile versioning interface for manipulating BLOBs that enables reading, writing and appending to them.

BlobSeer offers both scalability and performance with respect to a series of issues typically associated with the data-intensive context: *scalable aggregation of storage space* from the participating nodes with minimal overhead, ability to store *huge data objects*, *efficient fine-grain access* to data subsets, *high throughput in spite of heavy access concurrency*, as well as *fault-tolerance*. This year we have mainly focused on the deployment in production of the BlobSeer software on IBM's cluster at Montpellier, in the context of the ANR MapReduce project. To this end, several bugs were solved, and several optimizations were brought to the communication layer of BlobSeer. To showcase the benefits of BlobSeer on this platform we focused on the Terasort benchmark. Currently, preliminary tests on Grid5000 with this benchmark show that BlobSeer performs better than HDFS for block sizes lower than 2 MB. We have also improved the continuous integration process of BlobSeer by deploying daily builds and automatic tests on Grid5000.

Users: Work is currently in progress in several formalized projects (see previous section) to integrate and leverage BlobSeer as a data storage back-end in the reference cloud environments: a) Microsoft Azure; b) the Nimbus cloud toolkit developed at Argonne National Lab (USA); and c) the Open-Nebula IaaS cloud toolkit developed at UCM (Madrid).

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

License: GNU Lesser General Public License (LGPL) version 3.

Status: This software is available on Inria's forge. Version 1.0 (released late 2010) registered with APP: IDDN.FR.001.310009.000.S.P.000.10700.

A *Technology Research Action* (ADT, *Action de recherche technologique*) was active for two years until January 2015, aiming to robustify the BlobSeer software and to make it a safely distributable product. This project is funded by Inria *Technological Development Office* (D2T, *Direction du Développement Technologique*).

6.1.2. Damaris

Participants: Matthieu Dorier, Gabriel Antoniu, Orçun Yildiz, Lokman Rahmani, Shadi Ibrahim.

Contact: Gabriel Antoniu.

Presentation: Damaris is a middleware for multicore SMP nodes enabling them to handle data transfers for storage and visualization efficiently. The key idea is to dedicate one or a few cores of each SMP node to the application I/O. It is developed within the framework of a collaboration between KerData and the *Joint Laboratory for Petascale Computing* (JLPC). Damaris 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. Version 1.0 was released in November 2014 and enables other approaches such as the use of dedicated nodes instead of dedicated cores.

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

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

License: GNU Lesser General Public License (LGPL) version 3.

Status: This software is available on Inria's forge and registered with APP. Registration of the latest version with APP is in progress.

6.2. Other Software

6.2.1. *Omnisc'IO*

Participants: Matthieu Dorier, Shadi Ibrahim, Gabriel Antoniu.

Contact: Matthieu Dorier

Presentation: Omnisc'IO is a middleware integrated in the POSIX and MPI-I/O stacks to observe, model and predict the I/O behavior of any HPC application transparently. It is based on formal grammars, implementing a modified version of the Sequitur algorithm. Omnisc'IO has been used on Grid'5000 with the CM1 atmospheric simulation, the LAMMPS molecular dynamics simulation, the GTC fusion simulation and the Nek5000 CFD simulation. Omnisc'IO was subject to a publication at SC14.

Users: Omnisc'IO is currently used only within the KerData team and at Argonne National Lab.

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

License: GNU Lesser General Public License (LGPL) version 3.

Status: Currently unavailable for distribution (subject to major changes). Version 1.0 (released in November 2015) registered with APP: IDDN.FR.001.540003.000.S.P.2015.000.10000.

6.2.2. *JetStream*

Participants: Radu Tudoran, Alexandru Costan, Gabriel Antoniu.

Contact: Alexandru Costan

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

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

License: Microsoft Public License.

Status: Prototype and demo available.

6.2.3. *OverFlow*

Participants: Radu Tudoran, Alexandru Costan, Gabriel Antoniu.

Contact: Alexandru Costan.

Presentation: *OverFlow* is a uniform data management system for scientific workflows running across geographically distributed sites, aiming to reap economic benefits from this geo-diversity. The software is environment-aware, as it monitors and models the global cloud infrastructure, offering high and predictable data handling performance for transfer cost and time, within and across sites. *OverFlow* proposes a set of pluggable services, grouped in a data-scientist cloud kit. They provide the applications with the possibility to monitor the underlying infrastructure, to exploit smart data compression, deduplication and geo-replication, to evaluate data management costs, to set a tradeoff between money and time, and optimize the transfer strategy accordingly. In 2015, *OverFlow* was extended with support for efficient metadata operations: the newly implemented strategies leverage workflow semantics in a 2-level metadata partitioning hierarchy that combines distribution and replication.

Users: 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.

License: GNU Lesser General Public License (LGPL) version 3.

Status: Registration of the latest version with APP is in progress

6.2.4. *iHadoop*

Participants: Tien Dat Phan, Shadi Ibrahim.

Contact: Shadi Ibrahim

Presentation: *iHadoop* is a Hadoop simulator developed in Java on top of SimGrid to simulate the behavior of Hadoop and therefore accurately predict the performance of Hadoop in normal scenarios and under failures. In 2015, *iHadoop* was extended to simulate the execution and predict the performance of multiple Map-Reduce applications, sharing the same Hadoop cluster. Two schedulers (Fifo, Fair) are now available in *iHadoop*.

Users: *iHadoop* is an internal software prototype, which was initially developed to validate our idea for exploring the behavior of Hadoop under failures. *iHadoop* has preliminarily evaluated within our group and it has shown very high accuracy when predicating the execution time of a Map-Reduce application. *iHadoop* was discussed with the SimGrid community during the SimGrid user days in Lyon (June 2015). We intend to add *iHadoop* to the contributions site of the SimGrid project and make it available to the SimGrid community.

License: GNU Lesser General Public License (LGPL) version 3.

Status: Available on Inria's forge. Registration of the latest version with APP is in progress.

MESCAL Project-Team

5. New Software and Platforms

5.1. CiGri

FUNCTIONAL DESCRIPTION

CiGri is a middleware which gathers the unused computing resource from intranet infrastructure and makes it available for the processing of large set of tasks. It manages the execution of large sets of parametric tasks on lightweight grid by submitting individual jobs to each batch scheduler. It is associated to the OAR resource management system (batch scheduler). Users can easily monitor and control their set of jobs through a web portal. CiGri provides mechanisms to identify job error causes, to isolate faulty components and to resubmit jobs in a safer context.

- Contact: Olivier Richard
- URL: <https://www.projet-plume.org/fiche/cigri>

5.2. ComputeMode

ComputeMode: On-demand HPC cluster manager

KEYWORDS: HPC - Clusters - Operating system provisioning

FUNCTIONAL DESCRIPTION

ComputeMode is a on-demand HPC cluster manager, it allows deploying lightweight clustering framework on intranets.

ComputeMode is a software infrastructure that allows to extend or create a Grid through the aggregation of unused computing resources. For instance, a virtual cluster can be built using anyone's PC while not in use. Indeed, most PCs in large companies or university campus are idle at night, on weekends, and during vacations, training periods or business trips.

The main benefits of ComputeMode are the following

Easy deployment: the integration into an existing infrastructure is very easy: no modification is required on your PCs. ComputeMode comes as a software-only solution. The integration with major batch manager systems such as Sun Grid Engine, Platform LSF and Portable Batch System (PBS) can also be achieved. Seamless integration for the scientist: he/she submits unmodified computational jobs through his/her usual interface (batch submission engine), just like with any Beowulf type cluster. Seamless integration for the PC owner/user: ComputeMode runs when his/her PC is idle (night, weekends, ...) so annoyance is minimal if existant

Using ComputeMode, the life cycle of the PCs is basically split between 2 modes of operation a user mode, where the company's installation of Microsoft Windows or GNU/Linux remains a computation mode (hence the product name): uses a diskless boot of a GNU/Linux system and offers the PC's CPU power, RAM and connectivity to the Grid.

- Participants: Pierre Neyron, Olivier Richard and Bruno Bzeznik
- Partners: LIG - ANDRA
- Contact: Olivier Richard
- URL: <http://computemode.imag.fr>

5.3. 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: Jean-Marc Vincent
- URL: <http://soctrace-inria.github.io/framesoc/>

5.4. GameSeer

FUNCTIONAL DESCRIPTION

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

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

5.5. KA-Tools

FUNCTIONAL DESCRIPTION

The KA-Tools is a software suite developed by MESCAL for exploitation of clusters and grids. It uses a parallelization technique based on spanning trees with a recursive starting of programs on nodes. Industrial collaborations were carried out with Mandrake, BULL, HP and Microsoft.

- Contact: Olivier Richard
- URL: <http://ka-tools.imag.fr/>

5.6. Kadeploy

KEYWORD: Operating system provisioning

FUNCTIONAL DESCRIPTION

Kadeploy is a scalable, efficient and reliable deployment system (cluster provisioning solution) for cluster and grid computing. 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.

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

5.7. Kameleon

FUNCTIONAL DESCRIPTION

Kameleon is a simple but powerful tool to generate customized appliances. With Kameleon, you make your recipe that describes how to create step by step your own distribution. At start Kameleon is used to create custom kvm, docker, VirtualBox, ..., but as it is designed to be very generic you can probably do a lot more than that.

- Participant: Olivier Richard
- Partner: Grid'5000
- Contact: Olivier Richard
- URL: <http://kameleon.imag.fr/>

5.8. 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).

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

5.9. 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/>

5.10. PEPS

FUNCTIONAL DESCRIPTION

The main objective of PEPS is to facilitate the solution of large discrete event systems, in situations where classical methods fail. PEPS may be applied to the modelling of computer systems, telecommunication systems, road traffic, or manufacturing systems.

- Participants: Luka Stanasic, Arnaud Legrand, Augustin Degomme, Jean-Marc Vincent and Florence Perronnin
- Contact: Arnaud Legrand
- URL: <http://www-id.imag.fr/Logiciels/peps/>

5.11. 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: Arnaud Legrand
- URL: <https://gforge.inria.fr/projects/psi/>

5.12. Pajé

FUNCTIONAL DESCRIPTION

The Pajé generic tool provides interactive and scalable behavioral visualizations of parallel and distributed applications, helping to capture the dynamics of their executions, because of its genericity, it can be used unchanged in a large variety of contexts.

- Participants: Arnaud Legrand and Jean-Marc Vincent
- Contact: Jean-Marc Vincent
- URL: <http://paje.sourceforge.net/>

5.13. PajéNG

Pajé Next Generation

FUNCTIONAL DESCRIPTION

Pajé Next Generation is a re-implementation (in C++) and direct heir of the well-known Paje visualization tool for the analysis of execution traces (in the Paje File Format) through trace visualization (space/time view). The tool is released under the GNU General Public License 3. PajeNG comprises the libpaje library, the space-time visualization tool in pajeng and a set of auxiliary tools to manage Paje trace files (such as pj_dump and pj_validate).

- Participants: Jean-Marc Vincent and Arnaud Legrand
- Contact: Jean-Marc Vincent
- URL: <https://github.com/schnorr/pajeng>

5.14. SimGrid

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

FUNCTIONAL DESCRIPTION

Scientific Instrument for the study of Large-Scale Distributed Systems. SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments.

- Participants: Jonathan Rouzaud-Cornabas, Frédéric Suter, Martin Quinson, Arnaud Legrand, Takahiro Hirofuchi, Adrien Lèbre, Jonathan Pastor, Mario Südholt, Flavien Quesnel, Luka Stanisic, Augustin Degomme, Jean-Marc Vincent and Florence Perronnin
- Partners: CNRS - Université de Nancy - University of Hawaii - Université de Reims Champagne-Ardenne - Femto-st
- Contact: Arnaud Legrand
- URL: <http://simgrid.gforge.inria.fr/>

5.15. Viva

FUNCTIONAL DESCRIPTION

Viva is an open-source tool used to analyze traces (in the Paje File Format) registered during the execution of parallel or distributed applications. The tool also serves as a sandbox to the development of new visualization techniques. Current features include: Temporal integration using dynamic time-intervals Spatial aggregation through hierarchical traces Interactive Graph Visualization with a force-directed algorithm, with viva Squarified Treemap to compare processes behavior on scale, with vv_treemap.

- Contact: Arnaud Legrand
- URL: <https://github.com/schnorr/viva>

5.16. Platforms

5.16.1. Grid'5000

The MESCAL project-team is involved in development and management of Grid'5000 platform. The Digitalis and IDPot clusters are integrated in Grid'5000 as well as of CIMENT.

5.16.2. Local cluster computing platforms: ICluster-2, IDPot, Digitalis

The MESCAL project-team manages a cluster computing center on the Grenoble campus. The center manages different architectures: a 48 bi-processors PC (ID-POT), and the center is involved with a cluster based on 110 bi-processors Itanium2 (ICluster-2) and another based on 34 bi-processor quad-core XEON (Digitalis) located at Inria. The three of them are integrated in the Grid'5000 grid platform.

More than 60 research projects in France have used the architectures, especially the 204 processors Icluster-2. Half of them have run typical numerical applications on this machine, the remainder has worked on middleware and new technology for cluster and grid computing. The Digitalis cluster is also meant to replace the Grimage platform in which the MOAIS project-team is very involved.

5.16.3. The Bull Machine

In the context of our collaboration with Bull the MESCAL project-team exploits a Novascale NUMA machine. The configuration is based on 8 Itanium II processors at 1.5 Ghz and 16 GB of RAM. This platform is mainly used by the Bull PhD students. This machine is also connected to the CIMENT Grid.

MOAIS Project-Team

4. New Software and Platforms

4.1. FlowVR

FUNCTIONAL DESCRIPTION

The goal of the FlowVR library is to provide users with the necessary tools to develop and run high performance interactive applications on PC clusters and Grids. The main target applications include virtual reality, scientific visualization and Web3D. FlowVR enforces a modular programming that leverages software engineering issues while enabling high performance executions on distributed and parallel architectures.

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

4.2. K'Star

K'Star Action

KEYWORDS: Parallel computing - Task-based algorithm - Runtime system - Task scheduling - OpenMP - Source-to-source compiler - Data parallelism

FUNCTIONAL DESCRIPTION

The K'Star action supports the development of Klang-Omp, a source-to-source compiler that turns C and C++ programs with OpenMP pragmas to C programs with calls to either the StarPU or the XKaapi runtime system, as well as the development of the KaStORS benchmarks suite for experimenting with OpenMP dependent tasks.

- Participants: Philippe Virouleau, Pierrick Brunet, Thierry Gautier, Olivier Aumage, Samuel Thibault, Nathalie Furmento, Samuel Pitoiset and François Broquedis
- Contact: Thierry Gautier
- URL: <http://kstar.gforge.inria.fr/>

4.3. KAAPI

KAAPI – *Kernel for Adaptive, Asynchronous Parallel and Interactive programming* – is a C library that allows to execute fine/medium grain multithreaded computation with dynamic data flow synchronizations.

FUNCTIONAL DESCRIPTION

KAAPI means Kernel for Adaptive, Asynchronous Parallel and Interactive programming. It is a C library that allows to execute multithreaded computation with data flow synchronization between threads. The library is able to schedule fine/medium size grain programs on a multicore machine with several GPUs. The data flow graph is unfold at runtime.

- Contact: Thierry Gautier
- URL: <http://kaapi.gforge.inria.fr>

4.4. 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/>

4.5. LinBox

FUNCTIONAL DESCRIPTION

LinBox is an open-source C++ template library for exact, high-performance linear algebra computations. It is considered as the reference library for numerous computations (such as linear system solving, rank, characteristic polynomial, Smith normal forms,...) over finite fields and integers with dense, sparse, and structured matrices.

- Participants: Clément Pernet and Thierry Gautier
- Contact: Clément Pernet
- URL: <http://linalg.org/>

4.6. 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).

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

4.7. QuickCSG

KEYWORDS: 3D modeling - CAD - 3D reconstruction - Geometric algorithms

FUNCTIONAL DESCRIPTION

QuickCSG is a library and command-line application that computes boolean operations between polyhedra.

- Participants: Matthys Douze, Jean-Sébastien Franco and Bruno Raffin
- Partner: Grenoble-INP
- Contact: Matthys Douze
- URL: <http://pascal.inrialpes.fr/data2/douze/QuickCSG>

4.8. Triva

FUNCTIONAL DESCRIPTION

Triva is an open-source tool used to analyze traces (in the pajé format) registered during the execution of parallel applications. The tool serves also as a sandbox to the development of new visualization techniques.

- Participant: Guillaume Huard
- Contact: Guillaume Huard
- URL: <http://triva.gforge.inria.fr/>

4.9. XKaapi

FUNCTIONAL DESCRIPTION

XKaapi is a library for high performance applications running on multi-cores/multi-processors with support for multi-GPUs. XKaapi provides ABI compliant implementations of libGOMP (GCC runtime for OpenMP) and was one of the target runtime of the K*Star compiler,

- Contact: Thierry Gautier
- URL: <http://kaapi.gforge.inria.fr>

4.10. TakTuk

TakTuk: Adaptive large scale remote executions deployment

KEYWORD: Deployment

FUNCTIONAL DESCRIPTION

TakTuk is a tool for deploying parallel remote executions of commands to a potentially large set of remote nodes. It spreads itself using an adaptive algorithm and sets up an interconnection network to transport commands and perform I/Os multiplexing/demultiplexing. The TakTuk mechanics dynamically adapt to environment (machine performance and current load, network contention) by using a reactive work-stealing algorithm that mixes local parallelization and work distribution.

- Participants: Guillaume Huard, Pierre Neyron, Benoît Claudel, Johann Bourcier and Olivier Richard
- Partner: LIG
- Contact: Guillaume Huard
- URL: <http://taktuk.gforge.inria.fr/>

4.11. Platforms

4.11.1. Multi-camera Platforms Grimage and Kinovis

MOAIS has managed with the LJK-Inria Morpheo team the Grimage platform (<http://grimage.inrialpes.fr>) dedicated to off-line and on-line 3D modeling from multiple cameras and telepresence. In 2012, we received an Equipex funding, Kinovis (<http://kinovis.inrialpes.fr>), to renew this platform with an installation which consists of a significantly larger acquisition space with 68 cameras connected to a new computation cluster. Moais participated to the installation and setup of the new platform (in charge of the network, compute and storage infrastructures) which is operational since summer 2015. FlowVR is the software backbone of both platforms for live processing. MOAIS is participating to the FP7 infrastructure project Visionair to enable European research teams to experiment on both platforms.

4.11.2. HPC experimentation platforms: Digitalis, Grid'5000 and Ciment

MOAIS (with MESCAL) develops and operates the Digitalis experimental platform (<http://digitalis.inria.fr>) which hosts local experimental HPC machines as well as a site of the Grid'5000 national testbed and nodes of the CIMENT regional HPC center (mesocentre of the University of Grenoble). MOAIS with the Mescal and LIG-Erods team obtained in 2014 a grant (FAIRE from Grenoble-INP and LIG) to acquire new experimental machines. 2 ARM64 development boards and one last generation 4 CPU machine which can hosts up to 4 accelerators are now part of the platform.

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 has been developed by INPT(ENSEEIH)-IRIT, Inria, CERFACS, University of Bordeaux, CNRS and ENS Lyon.

- 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 and François-Henry Rouet
- Partners: Université de Bordeaux - CNRS - CERFACS - ENS Lyon - INPT - IRIT - Université de Lyon - Université de Toulouse - LIP
- Contact: Jean-Yves L'Excellent
- Public releases in 2015: MUMPS 5.0.0 (February 2015), including major improvements in terms of performance and robustness, and MUMPS 5.0.1 (July 2015)
- URL: <http://mumps-solver.org/>

Following the creation in 2014 of a consortium for industrial users of MUMPS (<http://mumps-consortium.org>), some collaborations with industry (scientific exchanges, support, releases in advance) are mentioned in Section 8.1 . We pursued our work on block low-rank solvers [2] (Section 7.13), which was extended and applied to 3D frequency domain seismic modeling [19], [18] (Section 7.15) in the context of an on-going collaboration with the Seiscope consortium (<https://seiscope2.obs.ujf-grenoble.fr/?lang=en?>). We also worked on the parallel computation of selected entries of the inverse of a sparse matrix [3] (Section 7.14).

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
- Contact: Christophe Alias
- Software transferred by Inria under an exclusive license, no web page.

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 (dependences, affine scheduling, code generation) 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
- Software transferred by Inria under an exclusive license, no web page.

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 derive 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 our new algorithm “Counter-example based generation of ranking functions” (see Section 7.29). 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 our new algorithm “An encoding of array verification problems into array-free Horn clauses” (see Section 7.30). 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
- Software not yet published, under consolidation.

STORM Team

6. New Software and Platforms

6.1. 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.

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

FUNCTIONAL DESCRIPTION

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

- Participants: Marc Sergent, Suraj Kumar, Samuel Thibault, Andra Hugo, Terry Cojean, Nathalie Furmento
- Partners: Innovative Computing Laboratory (ICL) - King Abdullha University of Science and Technology - University of Colorado Denver
- Contact: Emmanuel Agullo
- URL: <https://project.inria.fr/chameleon/>

6.2. KLANG-OMP

The Klang-Omp OpenMP Compiler

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

FUNCTIONAL DESCRIPTION

The Klang-Omp software is a source-to-source OpenMP compiler for languages C and C++. The Klang-Omp compiler translates OpenMP directives and constructs into API calls from the StarPU runtime system or the XKaapi runtime system. The Klang-Omp compiler is virtually fully compliant with OpenMP 3.0 constructs. The Klang-Omp 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>

6.3. 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>

6.4. MAQAO

SCIENTIFIC DESCRIPTION

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

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

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

FUNCTIONAL DESCRIPTION

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

- Participants: Denis Barthou, Olivier Aumage, Christopher Haine and James Tombi A Mba
- Contact: Denis Barthou
- URL: <https://gforge.inria.fr/projects/maqao>

6.5. P-EDGE

An Efficient and Portable Library for Error Correction Code Design and Optimization

KEYWORDS: Code generation - Error Correction Code

FUNCTIONAL DESCRIPTION

The P-Edge 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.

- Authors: Adrien Cassagne, Olivier Aumage, Bertrand Le Gal, Camille Leroux and Denis Barthou
- Partner: IMS
- Contact: Adrien Cassagne

6.6. 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 (e.g. 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 (e.g. 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 (e.g. 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: Samuel Thibault, Nathalie Furmento, Jérôme Clet-Ortega, Pierre-André Wacrenier, Terry Cojean, Andra Hugo, Raymond Namyst, Olivier Aumage, Marc Sergent and Samuel Pitoiset.
- Contact: Raymond Namyst
- URL: <http://starpu.gforge.inria.fr/>

6.7. hwloc

Hardware Locality

KEYWORDS: HPC - Topology - Open MPI - Affinities - GPU

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 the application so as to help them adapt their behavior to the hardware characteristics.

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
- Contact: Brice Goglin
- URL: <http://www.open-mpi.org/projects/hwloc/>

TADAAM Team

5. New Software and Platforms

5.1. Hardware Locality

KEYWORDS: Topology - 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 the application so as to help them adapt their behavior to the hardware characteristics. HWLOC also offers monitoring abilities to identify application bottlenecks [12]. Moreover it focuses on modeling the network topology by embedding the NETLOC subproject in its future releases.

HWLOC targets many types of high-performance computing applications [1], [2], from thread scheduling to placement of MPI processes. Most existing MPI implementations, many resource managers, task schedulers and parallel libraries already use HWLOC.

HWLOC is developed in collaboration within the OPEN MPI consortium. The core development is carried out by Brice GOGLIN and other members of TADAAM team-project, with external contribution from many academic and industrial partners.

HWLOC is composed of 100,000 lines of C.

- Participants: Brice Goglin, Nicolas Denoyelle, Cyril Bordage
- Contact: Brice Goglin
- URL: <http://www.open-mpi.org/projects/hwloc/>

5.2. NewMadeleine

KEYWORDS: High-performance computing - 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
- Contact: Alexandre Denis
- URL: <http://pm2.gforge.inria.fr/newmadeleine/>

5.3. PaMPA

KEYWORDS: parallel numerical solvers, unstructured heterogeneous meshes, dynamic load balancing, subdomain decomposition, parallel remeshing

FUNCTIONAL DESCRIPTION PAMPA (“Parallel Mesh Partitioning and Adaptation”) is a library dedicated to the management of distributed meshes. Its purpose is to relieve solver writers from the tedious and error-prone task of writing again and again service routines for mesh handling, data communication and exchange, remeshing, and data redistribution. It is based on a distributed data structure that represents meshes as a set of entities (elements, faces, edges, nodes, etc.), linked by relations (that is, computation dependencies).

PAMPA interfaces with SCOTCH for mesh redistribution, and with MMG3D for parallel remeshing of tetrahedral elements. Other sequential remeshers can be plugged-in, in order to handle other types of elements.

Version 1.0 of PAMPA allows users to declare distributed meshes, to declare values attached to the entities of the meshes (e.g. temperature attached to elements, pressures to the faces, etc.), to exchange values between overlapping entities located at the boundaries of subdomains assigned to different processors, to iterate over the relations of entities (e.g. iterate over the faces of elements), to remesh in parallel the areas of a mesh that need to be remeshed, and to redistribute evenly the remeshed mesh across the processors of the parallel architecture. PAMPA is already used as the data structure manager for the AeroSol solver developed by teams CARDAMOM and CAGIRE.

Version 2.0 of PAMPA features parallel remeshing using any sequential remesher. At the time being, PAMPA is interfaced with the MMG3D tetrahedral remesher designed within team CARDAMOM. Its coupling with Tetgen is in progress. Meshes above one billion elements are generated with a quality similar to that offered by the sequential remesher alone. More than 1 billion of elements are obtained on a cluster with 600 processors in less than 20 minutes. Tests show a quite good weak scalability up to 600 processors, limited by cluster size. Extensive scalability testing will be performed during year 2016. An collaboration with Dassault Aviation demonstrated the use of PAMPA with their meshes during solver computations.

- Participants: Cédric Lachat, François Pellegrini, Cécile Dobrzynski.
- Contact: Cédric Lachat, François Pellegrini
- URL: project.inria.fr/pampa

5.4. Scotch

KEYWORDS: parallel graph partitioning, parallel static mapping, parallel sparse matrix block ordering, graph repartitioning, fixed vertices, mesh partitioning.

FUNCTIONAL DESCRIPTION

SCOTCH is a software package for parallel and sequential sparse matrix ordering, parallel and sequential graph partitioning, as well as sequential static mapping and remapping, without and with fixed vertices, and mesh and hypergraph partitioning.

The initial purpose of SCOTCH was to compute high-quality static mappings of valued graphs representing parallel computations, onto target architectures of arbitrary topologies. Taking into account the topology and heterogeneity of the target architecture, in terms of processor speed and link bandwidth, allows SCOTCH to provide partitions that maximize communication locality.

This feature, which was meant for the NUMA machines of the 1980’s, has not been widely used in the past because high performance computers in the 1990’s were UMA architectures, thanks to hardware advances. As today’s high-end architectures are NUMA again, the mapping feature is regaining popularity.

The SCOTCH package consists of two libraries: the sequential SCOTCH library, and the parallel PT-SCOTCH library (for “Parallel Threaded SCOTCH”) that operates according to the distributed memory paradigm, using MPI. SCOTCH was the first full 64-bit implementation of a general purpose graph partitioner.

Version 6.0, released on December 2012, corresponding to the 20th anniversary of SCOTCH, offers many new features: static mapping with fixed vertices, static remapping, and static remapping with fixed vertices. Several critical algorithms of the formerly strictly sequential SCOTCH library can now run in a multi-threaded way. All of these features, which exist only in the sequential version, will be ported to the parallel PT-SCOTCH library in the upcoming 6.1 major release.

A recent work on branch 6.0 enables SCOTCH to map onto (possibly disconnected) parts of a regular architecture, thanks to the new sub target architecture. This feature aims at allowing programs to distribute their data so as to maximize locality, according to the assignment of nodes performed by the batch scheduler, which do not always represent a regular, nor a connected, part of a parallel machine.

SCOTCH has been integrated into numerous third-party software, which indirectly contribute to its diffusion.

- Participants: François Pellegrini, Cédric Lachat, Astrid Casadei.
- Contact: François Pellegrini
- URL: <https://gforge.inria.fr/projects/scotch/>

5.5. TreeMatch

KEYWORDS: High-performance computing. Process Placement

FUNCTIONAL DESCRIPTION

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

- Participants: Emmanuel Jeannot, Guillaume Mercier, François Tessier.
- Contact: Emmanuel Jeannot
- URL: <http://treematch.gforge.inria.fr>

5.6. Platforms

5.6.1. Platform: The Daltons

The Daltons are a set of machines available for the team members and the STORM team members enabling fast prototyping and benchmarking of our solutions, algorithms and software. It features up-to-date hardware (e.g. latest infiniband or GPU card) with the most recent processors.

ASCOLA Project-Team

5. New Software and Platforms

5.1. 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 Md Sabbir Hasan
- Contact: Thomas Ledoux
- URL: <http://www.emn.fr/z-info/csla/>

5.2. JEScala

FUNCTIONAL DESCRIPTION

JEScala is a Scala library which implements a seamless programming model combining object-oriented, aspect-oriented, event-based and concurrent programming. Events are object members. They can be explicitly triggered as in standard event-based programming but also implicitly triggered, as join points in aspect-oriented programming. Event expressions make it possible to compose events, filter them, and alter their content. Event handlers can be registered, and unregistered, dynamically.

Concurrency can be handled without any explicit thread manipulation. By default, primitive events are synchronous but they can also be declared as asynchronous. This creates concurrency between the source of the event and its handlers. This concurrency can then be handled by composing events with the join operator and creating disjunctions of the resulting events.

- Participants: Jurgen Van Ham, Guido Salvaneschi, Mira Mezini and Jacques Noyé
- Partners: *Technische Universität Darmstadt*
- Contact: Jacques Noyé
- URL: http://www.stg.tu-darmstadt.de/research/jescale_menu/index.en.jsp

5.3. SimGrid

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

FUNCTIONAL DESCRIPTION

Scientific Instrument for the study of Large-Scale Distributed Systems. SimGrid is a toolkit that provides core functionalities for the simulation of distributed applications in heterogeneous distributed environments.

The contributions by the ASCOLA team are related to the virtualization abstractions in particular the different models to simulate VM manipulations (e.g., the live migration model).

- Participants: Jonathan Rouzaud-Cornabas, Frédéric Suter, Martin Quinson, Arnaud Legrand, Adrien Lebre, Jonathan Pastor, Mario Südholt, Luka Stanisic, Augustin Degomme, Jean-Marc Vincent and Florence Perronnin
- Partners: CNRS - Université de Nancy - University of Hawaii - Université de Reims Champagne-Ardenne - Femto-st
- Contact: Arnaud Legrand
- URL: <http://simgrid.gforge.inria.fr/>

5.4. VMPlaces

FUNCTIONAL DESCRIPTION

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

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

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

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

- Participants: Adrien Lebre, Jonathan Pastor, and Mario Südholt
- Contact: Adrien Lebre
- URL: <http://beyondtheclouds.github.io/VMPlaceS/>

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

DIVERSE Project-Team

6. New Software and Platforms

6.1. FAMILIAR

FeAture Model scrIpt Language for manIpulation and Automatic Reasoning

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.

- Participants: Mathieu Acher, Guillaume Bécan, Sana Ben Nasr, Jose Galindo, Olivier Barais
- Contact: Mathieu Acher
- URL: <http://familiar-project.github.com>

6.2. OpenCompare

OpenCompare.org

KEYWORDS: product comparison matrix - tabular data - comparison

SCIENTIFIC 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 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 a 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.

MatrixMiner is also part of opencompare. It is a tool for automatically synthesizing PCMs from a set of product descriptions written in natural language. MatrixMiner is capable of identifying and organizing features and values in a PCM despite the informality and absence of structure in the textual descriptions of products. More information here: <https://matrix-miner.variability.io/>

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

6.3. Kermeta

KEYWORDS: Eclipse - Model-driven engineering

SCIENTIFIC DESCRIPTION

Kermeta is used in several cases:

to give a precise semantic of the behavior of a metamodel, which then can be simulated,

to act as a model transformation language,

to act as a constraint language.

FUNCTIONAL DESCRIPTION

The Kermeta workbench is a powerful meta-programming environment based on an object-oriented DSL (Domain Specific Language) optimized for metamodel engineering. Kermeta is now integrated into Melange (see next tool), and will continue to be supported and developed under this new project with a broader scope.

- Participants: Zoé Drey, Cyril Faucher, Franck Fleurey, Jean Marc Jezequel, Pierre Alain Muller, Jim Steel, François Tanguy, Didier Vojtisek, Benoît Combemale, Olivier Barais, Arnaud Blouin, Benoit Baudry, Thomas Degueule, David Mendez Acuna, Erwan Bousse and Fabien Coulon
- Partners: Université de Rennes 1 - UHA
- Contact: Benoît Combemale
- URL: <http://www.kermeta.org>

6.4. Melange

KEYWORDS: Language workbench, Domain-Specific (Modeling) Language, Model-Driven Engineering, Model execution and debugging, Execution trace management

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 type system that statically ensures the structural correctness of the produced DSMLs, and specific subtyping relationships between DSMLs to reason about their substitutability. 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 a language workbench that support a modular and reusable approach for domain-specific language design and implementation.

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. The language workbench embeds a 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 DSMLs. Melange also provides a dedicated meta-language where models are first-class citizens and languages are used to instantiate and manipulate them. By analogy with the class-based, object-oriented paradigm, Melange can be classified as a language-based, model-oriented programming language. 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 DSMLs. Executable meta-modeling is supported by weaving operational semantics defined with Kermeta (defined on top of Xtend). Melange is bundled as a set of Eclipse plug-ins.

- Participants: Thomas Degueule, Erwan Bousse, Fabien Coulon, Dorian Leroy, Didier Vojtisek, Olivier Barais, Arnaud Blouin, Benoit Combemale, Jean-Marc Jézéquel
- Partners: Université de Rennes 1
- Contact: Benoît Combemale
- URL: <http://melange-lang.org>

6.5. Kevoree

Kevoree Core C++

KEYWORDS: Embedded - Software Components - Software component - Dynamic adaptation

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.

Version: 5.3.1

Programming language: Java, Scala, Kermeta, Kotlin, Javascript, c#

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: Olivier Barais, Johann Bourcier, Noël Plouzeau, Benoit Baudry, Maxime Tricoire, Jacky Bourgeois, Inti Gonzalez Herrera, Ivan Paez Anaya, Manuel Leduc, Francisco-Javier Acosta Padilla and Mohamed Boussaa
- Partner: Université de Rennes 1
- Contact: Olivier Barais
- URL: <http://kevoree.org/>

6.6. amiunique

KEYWORDS: Privacy - Browser fingerprinting

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).

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 2015, FOSSA'14, FIC'15, ICT'15) and it has been visited by more than 100000 unique visitors in one year.

Programming language: Java, JavaScript, Scala

- Participants: Pierre Laperdrix, Benoit Baudry
- Partner: INSA Rennes
- Contact: Benoit Baudry
- URL: <https://amiunique.org/>
- URL source code: <https://github.com/DIVERSIFY-project/amiunique>

FOCUS Project-Team

6. New Software and Platforms

6.1. AIOCJ

SCIENTIFIC DESCRIPTION

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

- 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 currently extended with primitives for cloud-computing in the EU project ENVISAGE. ABS is very similar to ASP, developed by the 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 analysed by a back-end that uses a fix-point technique to derive in a deterministic way the deadlock information.

- Contact: Elena Giachino
- 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.

Our tool HoCA (Higher-Order Complexity Analyser) overcomes this limitation by translating higher-order programs – in the form of side-effect free OCaml programs – into equivalent first-order rewrite systems. At the heart of our tool lies Reynold’s defunctionalization technique. Defunctionalization however is not enough. Resulting programs have a recursive structure too complicated to be analysed automatically in all but trivial cases. To overcome this issue, HoCA integrates a handful of well established program transformation techniques, noteworthy dead-code elimination, inlining, instantiation and uncurrying. All these techniques have been specifically suited to the methods integrated in modern first-order complexity analysers. Of course, the complete transformation pipeline underlying our tool is not only proven semantically correct, but also to reflect the runtime behavior. This way, 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>

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

6.4. JOLIE

Java Orchestration Language Interpreter Engine

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 2015: Jolie has transitioned from version 1.1 to version 1.4.1. The releases are the result of more than 400 commits with hundreds of bug fixes and enhancements. Highlights include: a new web site and documentation, a new pre-compiled installer, 2 new IDEs as plugins for the editors Sublime Text and Atom, a transition from SourceForge to GitHub, introduction of new behavioural and architectural constructs, structured support for the development of REST applications, introduction of the construct of internal services: embedded Jolie services defined directly within the embedder program (internal services offer a convenient way of reusing code as in procedural programming, without breaking the principle that such code should be easily exported to an external microservice), increased compliance with protocol standards (foremost HTTP, SSL), enhanced stability and performances. Moreover, 2015 has seen the development of Jolie Redeployment Optimiser (JRO), a tool for the automatic and optimised deployment of microservices written in Jolie. JRO uses Zephyrus, a state-of-the-art tool that automatically generates configurations starting from partial and abstract descriptions of the target application. Given the output configuration from Zephyrus, JRO interacts with Jolie Enterprise, an administrative tool for the deployment of Jolie services on remote nodes, to deploy the wanted architecture.

- Contact: Fabrizio Montesi, Saverio Giallorenzo
- 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: Elena Giachino
- 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 parallel portfolio solver capable of solving Constraint (Satisfaction/Optimization) Problems 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

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: Roberto Amadini
- URL: <https://github.com/CP-Unibo/sunny-cp>

6.7. Blender

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

5. New Software and Platforms

5.1. Web programming

Participants: Yoann Couillec, Colin Vidal, Vincent Prunet, Manuel Serrano [correspondant].

5.1.1. The HOP web programming environment

HOP is a higher-order language designed for programming interactive web applications such as web agendas, web galleries, music players, etc. It exposes a programming model based on two computation levels. The first one is in charge of executing the logic of an application while the second one is in charge of executing the graphical user interface. HOP separates the logic and the graphical user interface but it packages them together and it supports strong collaboration between the two engines. The two execution flows communicate through function calls and event loops. Both ends can initiate communications.

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

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

HOP has won the software *open source contest* organized by the ACM Multimedia Conference 2007. It is released under the GPL license. It is available at <http://hop.inria.fr>.

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

5.1.2. The Bigloo compiler

The programming environment for the Bigloo compiler [7] is available on the Inria Web site at the following URL: <http://www-sop.inria.fr/teams/indes/fp/Bigloo>. The distribution contains an optimizing compiler that delivers native code, JVM bytecode, and .NET CLR bytecode. It contains a debugger, a profiler, and various Bigloo development tools. The distribution also contains several user libraries that enable the implementation of realistic applications.

BIGLOO was initially designed for implementing compact stand-alone applications under Unix. Nowadays, it runs harmoniously under Linux and MacOSX. The effort initiated in 2002 for porting it to Microsoft Windows is pursued by external contributors. In addition to the native back-ends, the BIGLOO JVM back-end has enabled a new set of applications: Web services, Web browser plug-ins, cross platform development, etc. The new BIGLOO .NET CLR back-end that is fully operational since release 2.6e enables a smooth integration of Bigloo programs under the Microsoft .NET environment.

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

PHOENIX Project-Team

6. New Software and Platforms

6.1. DiaSuite: a Development Environment for Sense/Compute/Control

Applications

Participants: Charles Consel [correspondent], Julien Durand, Milan Kabac, Paul Van Der Walt, Adrien Carteron.

Despite much progress, developing a pervasive computing application remains a challenge because of a lack of conceptual frameworks and supporting tools. This challenge involves coping with heterogeneous devices, overcoming the intricacies of distributed systems technologies, working out an architecture for the application, encoding it in a program, writing specific code to test the application, and finally deploying it.

DIASUITE is a suite of tools covering the development life-cycle of a pervasive computing application.

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

Designing an application. Given a taxonomy, the architect can design and structure applications. To do so, the DIASPEC language provides an application design layer [39]. This layer is dedicated to an architectural pattern commonly used in the pervasive computing domain [35]. Describing the architecture application allows to further model a pervasive computing system, making explicit its functional decomposition.

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

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

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

This development cycle is summarized in the Figure 2 .

See also the web page <http://diasuite.inria.fr>.

6.1.1. DiaSpec: a Domain-Specific Language for Networked Entities

The core of the DIASUITE development environment is the domain specific language called DIASPEC and its compiler DIAGEN.

DIASPEC It is composed of two layers.

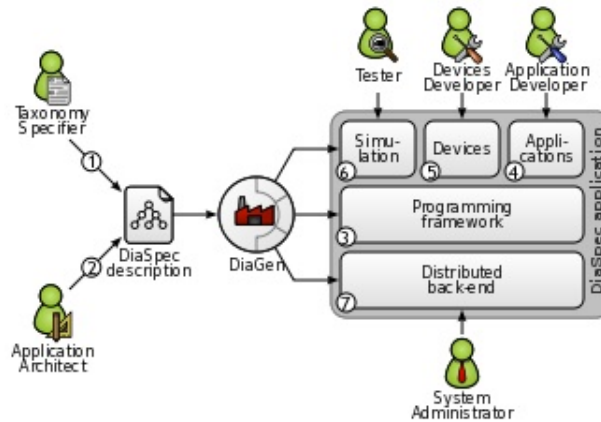


Figure 2. DIASUITE Development Cycle

The Taxonomy Layer. It allows the declaration of entities that are relevant to the target application area. An entity consists of sensing capabilities, producing data, and actuating capabilities, providing actions. Accordingly, an entity description declares a data source for each one of its sensing capabilities. As well, an actuating capability corresponds to a set of method declarations. An entity declaration also includes attributes, characterizing properties of entity instances. Entity declarations are organized hierarchically allowing entity classes to inherit attributes, sources and actions. A taxonomy allows separation of concerns in that the expert can focus on the concerns of cataloging area-specific entities. The entity developer is concerned about mapping a taxonomical description into an actual entity, and the application developer concentrates on the application logic.

The Architecture Layer. It is based on an architectural pattern commonly used in the pervasive computing domain [35]. It consists of context components fueled by sensing entities. These components process gathered data to make them amenable to the application needs. Context data are then passed to controller components that trigger actions on entities. Using an architecture description enables the key components of an application to be identified, allowing their implementation to evolve with the requirements (*e.g.*, varying light management implementations in a controller component to optimize energy consumption).

DIAGEN. It is the DIASPEC compiler that performs both static and runtime verifications over DIASPEC declarations and produces a dedicated programming framework that guides and eases the implementation of components. The generated framework is independent of the underlying distributed technology. As of today, DIAGEN supports multiple targets: Local, RMI, SIP, Web Services and OSGI.

6.2. DiaSuiteBox: an Open Orchestration Platform

Participants: Charles Consel, Julien Durand, Adrien Carteron, Milan Kabac.

The DiaSuiteBox platform runs an open-ended set of applications leveraging a range of appliances and web services. Our solution consists of a dedicated development environment, a certifying application store, and a lightweight runtime platform. This solution is based on the DIASUITE project.

The DiaSuiteBox platform can be embedded in a small plug-computer or deployed in the cloud. Thanks to the application store and the developer community, the platform is fed by a full offer of new innovative applications. During the submission process, an application is automatically analyzed and checked in order to be certified. The user is ensured the behavior of its applications are innocuous and correct with respect to the provided information. Finally, DiaSuiteBox provides an extensible software architecture. This allows the easily

connect new device technologies to the platform. For example, the support for new wireless communication technologies such as Zigbee, Z-Wave or Sigfox can be easily added to the DiaSuiteBox platform.

More details can be found on the web page <http://diasuitebox.inria.fr>.

The iQSpot startup uses DiaSuiteBox as a software platform to ease the management of Smart Buildings. In this project, the DiaSuiteBox platform is first used to host building management functionalities such as lighting management, heating/ventilating/air conditioning management, energy efficiency monitoring. It is also used to host software drivers that allow the building management functionalities to interact with the connected devices deployed in buildings. These devices can use wired communication technologies such as LonWorks, BACNet or KNX, as well as wireless communication technologies such as Z-Wave or Zigbee.

6.3. DiaSwarm: Orchestrating masses of objects

Participants: Charles Consel [correspondent], Milan Kabac, Eugène Volanschi.

DiaSwarm provides a design language dedicated to the domain of orchestrating masses of objects. The language provides high-level, declarative constructs that allow a developer to deal with masses of objects at design time, prior to programming the application. The DiaSwarm compiler generates programming frameworks, which provide high-level support to the developer, while ensuring that programming is driven by design.

DiaSwarm consists of two main components.

- DiaSwarm Studio. It is available as an Eclipse plugin. The tool provides of a domain-specific language and a compiler dedicated to the domain of orchestrating masses of sensors.
- Runtime environment. This is a library allowing you to execute DiaSwarm applications locally. The runtime environment comprises the Apache Hadoop framework v. 1.2.1 allowing you to run MapReduce jobs locally without the need to install the framework separately.

More details can be found on a dedicated web page <http://phoenix.inria.fr/software/diaswarm>. In particular, an example application is provided: it is dedicated to the management of parking lots in a city. It uses presence sensors to monitor the availability of parking spaces through magnetic field variations. The application defines a few context components, which transform sensed data to determine the availability of parking lots and average occupancy of parking lots in 24 hr. Furthermore, the application suggests parking lots to drivers entering the city. Please note that the availability of parking places is computed every 30 seconds. The average occupancy of parking lots is computed every 2 minutes.

6.4. School+ Apps: Assistive tablet applications for school Inclusion

Participants: Charles Consel [correspondent], H el ene Sauz eon, Charles Fage.

School+ 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.

6.4.1. Assistive applications

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.

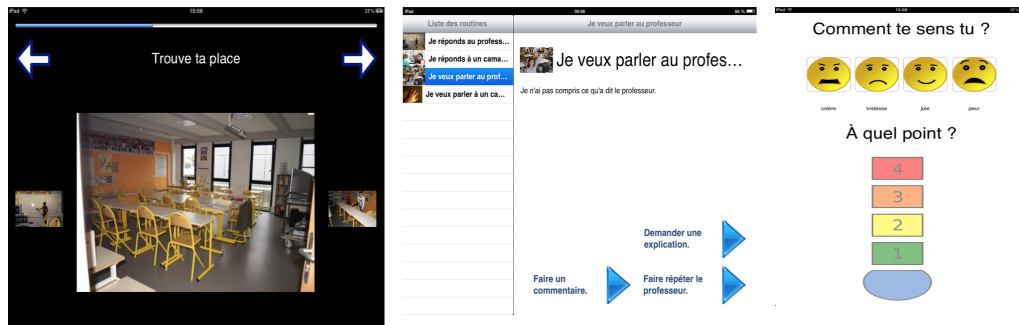


Figure 3. Assistive applications

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.

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.

6.4.2. Training applications

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

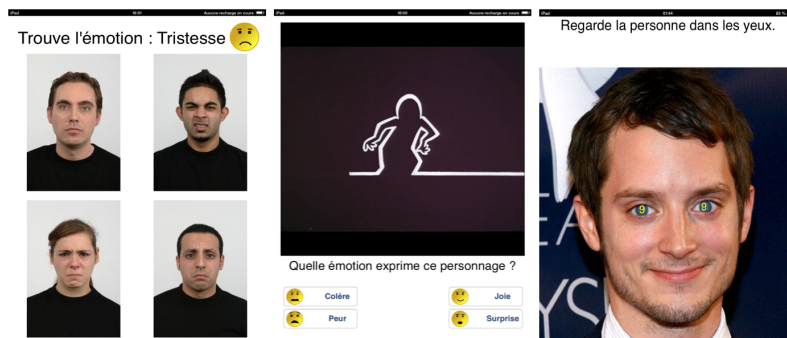


Figure 4. Training applications

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.

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.

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.

6.5. HomeAssist: A Platform for Assistive Living

Participants: Charles Consel [correspondent], Adrien Carteron, Julien Durand, Lucile Dupuy, H  l  ne Sauz  on.

The HomeAssist platform 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.

6.5.1. 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.*

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 num  rique    la personne”, 2014.

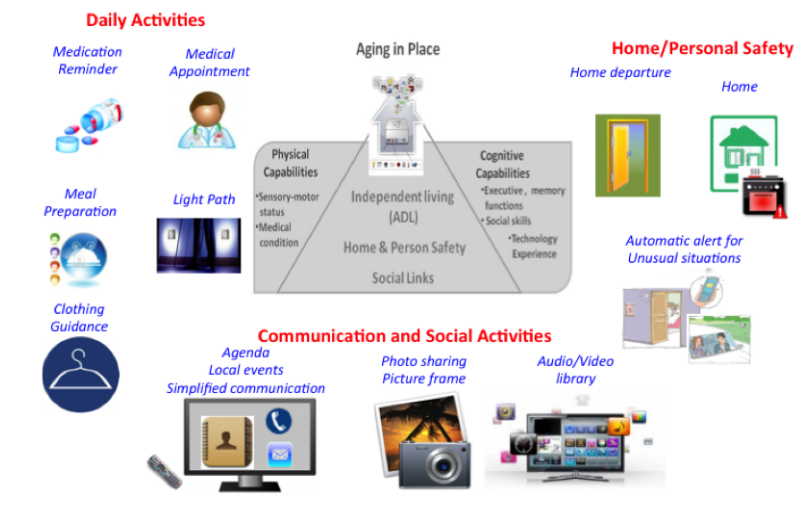


Figure 5. The HomeAssist platform and applications

6.5.2. 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

6.5.3. Experimental validation

A field study is currently being conducted with elderly people. The major purpose of this study is to identify the benefits of using HomeAssist for this population in an ecological framework. We selected 24 elderly people with different levels of autonomy (GIR scores). The HomeAssist technology has been installed in their house during 9 months. Twenty-four non-equipped older adults were also selected to participate to the study, as control participants.

The expected impact of HomeAssist reflects the trans-disciplinary nature of the project. We aim to deliver results in the domain of (1) elderly care, (2) ergonomics and human factors, and (3) pervasive computing.

The major expected results are that HomeAssist (1) prolongs ageing in place, improves well-being of the users, and improves the efficiency of the caregiving environment; (2) is a cognitively low-cost assistive technology, and is well accepted and perceived as useful and usable by the users; (3) is technologically robust, and is a validated assistive platform.

The preliminary results of this field study show that the platform is well adopted (highly accessible and usable) by the older users and their families or caregivers.

Additionally, these results support our claim that DiaSuiteBox is effective in terms of:

1. autonomy and well-being of older adults, comparing pre- vs. post-deployment, as well as not-equipped, control participants;
2. gains pre- vs. post deployment in management and load of caregiving tasks, and
3. longitudinal efficacy (gains unchanged at 6 vs. 9 months of use).

Most notably, this pilot project also uses a systemic approach to assistive living as illustrated by (1) its funding sources involving key stakeholders of the caregiving domain, including a territorial agency of Gironde dedicated to ageing, as well as the regional branch of the national public retirement organization (*i.e.*, CARSAT of Aquitaine), (2) their partners operating caregiving services (UDCCAS Gironde), and (3) The regional Chamber of Trades and Crafts to assist the older adults in installing and positioning the devices.

Final results are expected in March 2016.

6.5.4. External Partners

The HomeAssist platform is being developed with support from the following partners:

- Équipe “Handicap et Système Nerveux” (EA 4136), Bordeaux University
- Chaire TSA, Université du Québec Trois-Rivières
- CRIUGM, Université de Montréal
- UDCCAS Gironde
- CARSAT Aquitaine
- Conseil Général 33
- Conseil Régional d’Aquitaine

RMOD Project-Team

6. New Software and Platforms

6.1. 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: Stéphane Ducasse, Muhammad Bhatti, Andre Cavalcante Hora, Nicolas Anquetil, Anne Etien, Guillaume Larcheveque and Alexandre Bergel
- Partners: Université de Berne - Sensus - Synectique - Pleiad - USI - Vrije Universiteit Brussel
- Contact: Stéphane Ducasse
- URL: <http://www.moosetechnology.org>

6.2. Pharo

KEYWORDS: Live programmation objet - Reflective system

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.

- Participants: Marcus Denker, Damien Cassou, Christophe Demarey, Stéphane Ducasse, Esteban Lorenzano, Damien Pollet, Camille Teruel and Clément Béra
- Partners: Université de Berne - École des Mines de Douai - Uqbar foundation Argentina - Sensus - Synectique - Pleiad - Debris publishing - Yesplan - HR Works - MAD - BetaNine - Vmware
- Contact: Marcus Denker
- URL: <http://www.pharo.org>

6.3. Pillar

KEYWORDS: HTML - LaTeX - HTML5

FUNCTIONAL DESCRIPTION

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

- Contact: 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: Aurélien Richez
- Partner: Université de Rennes 1
- Contact: Frédéric Weis
- URL: <https://github.com/bpietropaoli/THEGAME/>

DESCRIPTION

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

THE GAME is published under apache licence. It is maintained and experimented within a sensor network platform developed by TACOMA since June 2013.

5.2. Platforms

5.2.1. Platform Pervasive_RFID

KEYWORDS: Composite objects - RFID

- Participants: Paul Couderc and Nebil Ben Mabrouck
- Partner: Université de Rennes 1 (IETR)
- Contact: Paul Couderc

SCIENTIFIC DESCRIPTION

In 2015 we completed the RFID experiment testbed realized in 2014 in collaboration with IETR (see Figure 1).

This system allows both interactive testing as well as long running experiments of RFID reading protocols. It comprises a software platform (see Figure 2) 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). This testbed supports the reproduction of most situations found in real applications. We achieved the following improvements on the testbed in 2015:

- The implementation of a software controllable smart antenna, with dynamic radiating pattern, was completed along with its the hardware interface. This is an important feature when experimenting pervasive application, in particular to determine the radio performance improvements expectable from antenna diversity.

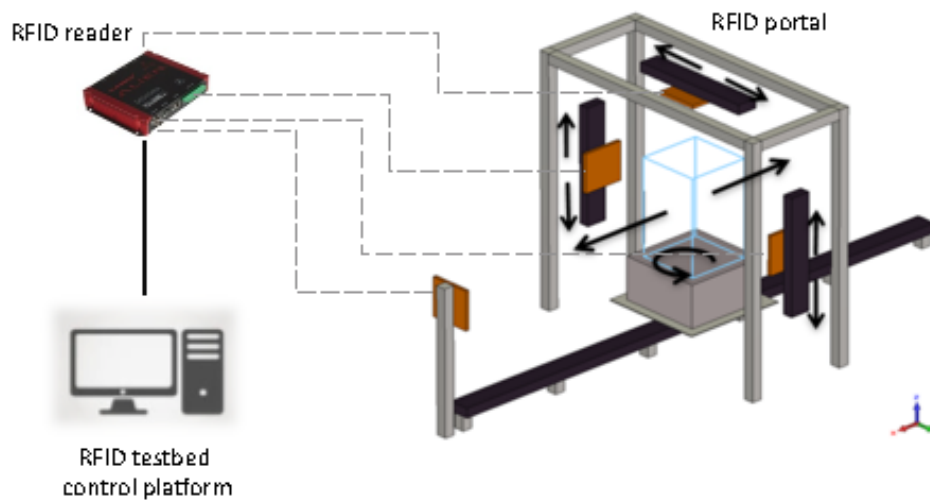


Figure 1. RFID testbed

- A particular effort was put on improving the operational performance and robustness of the system: the initial implementation was completed in 2014 in a context tightly coupled with our on-going RFID research, and with an important dependency on the technical expertise of short term positions staff-member. In order to widen the testbed's perspectives as a research tool, two aspects were improved:
 1. a high-level scripting interface was added to offer easy automatized experimentation campaign. Iterative RFID experiments with fine-tuning parameters can be specified using a variety of script languages, and further executed in a remote fashion via the added interface.
 2. A virtualization of RFID readers and motion drives was also developed to simulate the behavior of critical experiments or new software updates before executing them on the real testbed.
 3. Functional tests were developed to ascertain that crucial functions of the testbed would work correctly after future software updates, thus ensuring the maintainability and sustainability of the system.

5.2.2. On-demand room

KEYWORDS: Smart Home - Metamorphic House

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

DESCRIPTION

As part of the demonstration activities, we realized a prototype of the on-demand room as an immersive interactive virtual-reality application, leveraging the Immersia platform. Two iterations were achieved so far.

For the first iteration of the demonstrator, we realized a three-dimensional model of the on-demand room and two adjacent apartments, using the software SketchUp. This model was then imported in Unity3D and MiddleVR, which allow to display and navigate inside the model using the virtual reality platform Immersia⁰.

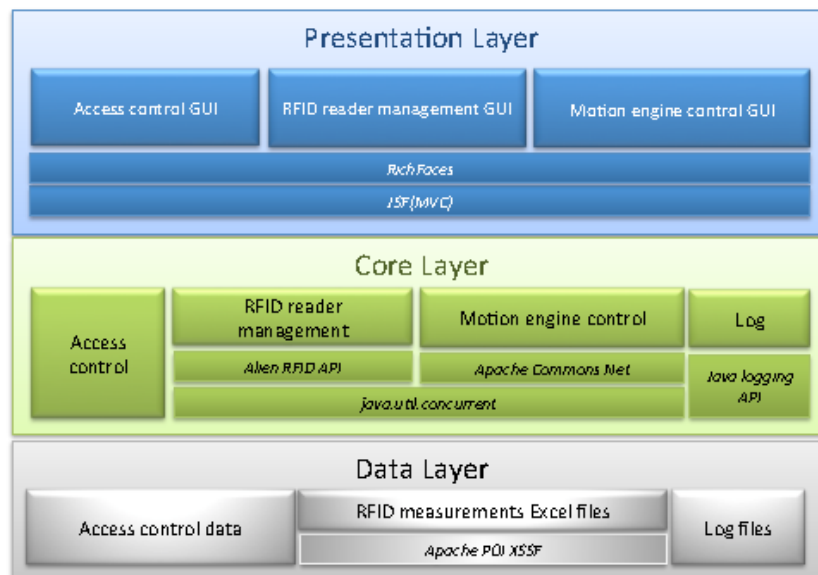


Figure 2. Software architecture of the RFID testbed

We then implemented the application logic of the on-demand room using Unity scripting facilities. By wearing 3D glasses and a marker on their hand, users can literally walk inside the apartments, open doors and observe how the configuration of the room changes to become a part of one dwelling or another, as shown in Figure 3.



Figure 3. 3D model in the Immersia platform

⁰<http://www.irisa.fr/immersia/>

The second (and current) iteration of the demonstrator introduced a major feature: the real/virtual integration. Actual domestic appliances can now be connected to Immersia and participate to the on-demand room demonstration. In this phase we showed that a real light switch, located in the virtual on-demand room, can change its behavior and alternatively control one of the two real lamps, each located in one of the apartments, as shown in Figure 4 .

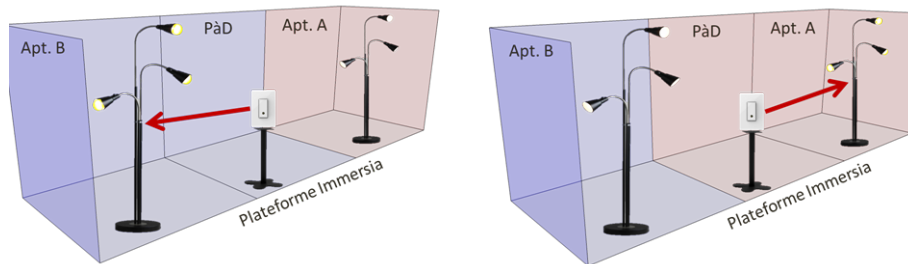


Figure 4. Integration of real devices

To develop the real/virtual integration feature, we used home automation controllers and devices implementing the KNX standard. After wiring and programming the domotic network, we developed an application that leverages the Falcon library, provided by the KNX association, to dynamically change the behavior of the real light switch.

COATI Project-Team

6. New Software and Platforms

6.1. Grph

Participants: Luc Hogue [Contact], Nathann Cohen, David Coudert.

FUNCTIONAL DESCRIPTION

GRPH is an open-source Java library for the manipulation of graphs. In 2015, the library has been maintained and augmented for users needs, especially with a new algorithm for iterating over the cycles of a given graph. This was requested by the EPI AOSTE for the TimeSquare tool.

URL: <http://www.i3s.unice.fr/~hogie/grph/>

6.2. JourneyPlanner

Participant: Marco Biazzi [Contact].

FUNCTIONAL DESCRIPTION

JourneyPlanner is a Java implementation of a recursive algorithm to solve a TSP problem on small dense graphs, where non-trivial constraints must be satisfied, that make commonly used paradigms (as dynamic programming) unfit to the task.

This work is done in collaboration with the R&D service of the "Train Transportation" division of Amadeus.

6.3. Sagemath

Participants: David Coudert [Contact], Nathann Cohen.

SCIENTIFIC DESCRIPTION

Sagemath is a free open-source mathematics software system initially created by William Stein (Professor of mathematics at Washington University). 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.

OUR CONTRIBUTION

We contribute the addition of new graph algorithms to **Sagemath**, along with their documentation and the improvement of underlying data structures.

URL : <http://www.sagemath.org/>

6.4. TripPlanner

Participants: David Coudert [Contact], Stéphane Pérennes.

FUNCTIONAL DESCRIPTION

TripPlanner is a tool for computing a minimum cost trip across multiple cities when neither the order in which to visit the cities nor the sojourn duration in these cities are fully specified. The cost of a trip includes both the price of all airplane tickets necessarily for the trip plus the price of the hotels (both costs depend on the exact travel date) at which the user will sojourn. The trip planner is also able to compute the k cheapest trips.

TripPlanner is written in Python and uses the linear programming interface of **Sagemath**.

This work is done in collaboration with the R&D service of the "Train Transportation" division of Amadeus.

BNF Antepedia Deposit 2015-09-23-16-11-18

6.5. Platforms

6.5.1. BigGraphs

Participants: Luc Hogie [Contact], Nicolas Chleq [SED-SOP], Michel Syska [Coordinator], David Coudert, Paul Bertot, Flavian Jacquot, Arnaud Legout [DIANA], Fabrice Huet [SCALE], Éric Madelaine [SCALE].

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 with 3 millions of nodes and 200 millions of edges. Last year we started the project with the evaluation of existing middlewares (GraphX/Spark and Giraph/Hadoop). After having tested some useful algorithms (written in the BSP model) we decided to develop our own platform.

This platform is based on the existing BIGGRPH library and this year we have focused 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 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.

This project is a joint work of the three EPI COATI, DIANA and SCALE and is supported by an ADT grant.

URL: <http://www.i3s.unice.fr/~hogie/software/index.php?name=grph>

The following software are useful tools that bring basic services to the platform (they are not dedicated to BIGGRPH).

JAC-A-BOO: is a framework aiming at facilitating the deployment of distributed Java scientific applications over clusters and is used to start BIGGRPH. computers.

LDJO: (Live Distributed Java Objects) is a framework for the development and the deployment of Java distributed data structures

OCTOJUS: provides an object-oriented RPC (Remote Procedure Call) implementation in Java

Participants : Luc Hogie [Contact], Nicolas Chleq

URL : <http://www.i3s.unice.fr/~hogie/> {jacaboo,ldjo,octojus}

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: <http://perso.ens-lyon.fr/benjamin.girault/>

6.2. IoT-LAB aggregation-tools

KEYWORD: Internet of things

FUNCTIONAL DESCRIPTION

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

- Participant: Gaetan Harter
- Contact: Éric 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: Éric Fleury
- URL: <https://github.com/iot-lab/cli-tools>

6.4. IoT-LAB gateway

KEYWORD: Internet of things

FUNCTIONAL DESCRIPTION

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

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

6.5. IoT-LAB robots

KEYWORDS: Internet of things - Robotics

FUNCTIONAL DESCRIPTION

IoT-LAB robots is an embedded robot controller on a Turtlebot2 providing the IoT-LAB node mobility functionality

- Partner: Université de Strasbourg
- Contact: Frédéric Saint-Marcel

6.6. Queueing Systems

KEYWORDS: Performance Evaluation - Queueing Models

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/>

6.7. WSNet

KEYWORD: Network simulator

FUNCTIONAL DESCRIPTION

The WSNet-3.0 project objective is to develop the next evolution of the WSNet simulator. It 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.

- Participants: Rodrigue Domga Komguem, Quentin Lampin, Alexandre Mouradian and Fabrice Valois
- Partner: CEA-LETI
- Contact: Fabrice Valois
- URL: <https://gforge.inria.fr/projects/wsnet-3/>

6.8. Platforms

6.8.1. FIT IoT-LAB

FUNCTIONAL DESCRIPTION

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

- Partner: FIT is one of 52 winning projects from the first wave of the French Ministry of Higher Education and Research'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, Nicolas Aguilera Miranda, 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 (see section 6.1.1 for more details). 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], Inderjeet Singh, Maksym Gabielkov.

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 has been supported by the Inria ADT ElectroSmart in 2014 and 2015 and will be supported by a Labex funding in the next two years.

In 2015, we released the first alpha version of the application for tests with real users (10 volunteers) and we published an associated Web site <http://es.inria.fr>.

- URL: <http://es.inria.fr>
- Version: 1.0alpha
- 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

Participants: Thierry Turetletti [correspondant], Alina Ludmila Quereilhac, Thierry Parmentelat, Mario Antonio Zancanaro.

NEPI, the Network Experimentation Programming Interface, is a framework to describe and orchestrate network experiments on a variety of network experimentation platforms, including simulators, emulators, live testbeds, and testbed federations. NEPI is capable of supporting arbitrary platforms through the use of a generic network experiment description model, based on abstracting network experiments as a collection of arbitrary resource objects, and through the generalization of the experiment life cycle for all resources. The common resource life cycle consists of the sequence of operations deploy, start, stop, and release. Different resource objects can implement specific versions of those operations to adapt to any platform. NEPI resolves experiment orchestration as an online scheduling problem.

In the context of Alina Quereilhac PhD, we generalized in 2015 the network experiments automation framework for arbitrary evaluation platforms, and for scenarios targeting any networking research domain. The proposed approach is based on abstracting the experiment life cycle for different platforms into generic steps that are valid for simulators, emulators, and testbeds. Based on these steps, a generic experimentation architecture was proposed and implemented, composed of an experiment model, an experimentation interface, and an orchestration algorithm. Three main aspects of the framework were evaluated: its extensibility to support heterogeneous platforms, its efficiency to orchestrate experiments, and its flexibility to support diverse use cases for different networking research domains, including education, platform management, and experimentation with testbed federations, and cross-platform and multi-platform scenarios. The results show that the proposed approach can be used to efficiently automate experimentation on heterogeneous evaluation platforms, for a wide range of scenarios.

On a much more practical level, NEPI is now available in a version numbered 6, that can run within both python2 and python3 environments.

- URL: <http://nepi.inria.fr>
- Version: 6.0
- ACM: C.2.2, C.2.4
- Keywords: networking experimentation, simulation, emulation
- License: GPL (3)
- Type of human computer interaction: python library
- OS/Middleware: Linux
- Required library or software: matplotlib - graphviz (both optional)
- Programming language: python2 or python3

5.4. ns-3

Participants: Walid Dabbous [contact], Thierry Turletti.

ns-3 is a discrete-event network simulator for Internet systems, targeted primarily for research and educational use. ns-3 includes a solid event-driven simulation core as well as an object framework focused on simulation configuration and event tracing, a set of realistic 802.11 MAC and PHY models, an IPv4, UDP, and TCP stack and support for nsc (integration of Linux and BSD TCP/IP network stacks). ns-3 is free software, licensed under the GNU GPLv2 license, and it is publicly available for research, development, and use. Our team has been involved in ns-3 project since 2006 and we are founding member of the ns-3 consortium including Washington University, Georgia Tech, CTTC, INESC PORTO as executive members. In 2015, using the NEPI framework, we worked on the automation of ns-3 experiments in multi-host scenarios with three example cases: a) running parallel simulations on a cluster of hosts, b) running distributed simulations spanning multiple hosts, and c) integrating live and simulated networks.

- URL: <http://www.nsnam.org>
- Version: ns-3.21
- Keywords: networking event-driven simulation
- License: GPL (GPLv2)
- Type of human computer interaction: programmation C++/python, No GUI
- OS/Middleware: Linux, cygwin, osX
- Required library or software: standard C++ library: GPLv2
- Programming language: C++, python
- Documentation: doxygen

5.5. DCE

Participants: Thierry Turletti [contact], Walid Dabbous.

DCE enables developers and researchers to develop their protocols and applications in a fully controllable and deterministic environment, where tests can be repeated with reproducible results. It allows unmodified protocol implementations and application code to be tested over large and possibly complex network topologies through the ns-3 discrete-event network simulator. The single-process model used in the DCE virtualization core brings key features, such as the possibility to easily debug a distributed system over multiple simulated nodes without the need of a distributed and complex debugger. Examples of tested applications over DCE include Quagga, iperf, torrent, thttpd, CCNx and various Linux kernel versions (from 2.6.36 to 3.12 versions). DCE was initially developed by Mathieu Lacage during his PhD thesis and is maintained by engineers in the team in collaboration with Hajime Tazaki from University of Tokyo. Our effort on DCE was reduced in 2015 due to lack of resources, but DCE/ns-3 represents an important component of R2lab in particular for performance comparison and hybrid (real/simulation/emulation) experiments. DCE is free software, licensed under the GNU GPLv2 license, and is publicly available for research, development, and use.

- URL: <https://www.nsnam.org/overview/projects/direct-code-execution/>
- Version: DCE-1.7
- Keywords: emulation, virtualization, networking event-driven simulation
- License: GPL (GPLv2)
- Type of human computer interaction: programming C/C++, No GUI
- OS/Middleware: Linux
- Required library or software: standard C++ library: GPLv2
- Programming language: C++, python
- Documentation: doxygen

5.6. 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 2015, OpenLISP is still used to provide connectivity between satellite sites of the LISP-Lab project and has been used as LISP implementation for PACAO, an overlay aiming at optimising Cloud access in distributed data-centers.

- <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: ddt-root.org

5.7. Platforms

5.7.1. *Reproducible research laboratory* (R²LAB)

Scientific evaluation of network protocols requires that experiment results must 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 2014, an anechoic chamber, with RF absorbers preventing radio waves reflections and with Faraday cage blocking external interferences. This lab, named R²lab, represents an ideal environment for experiments reproducibility. R²lab has been announced for usage by the general public at the end of 2015. It was developed, and is now operated, 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. Future work, in addition to regular operations, includes adding new hardware capabilities to the wireless nodes, such as USRP for running Software Defined Radio, as well as possibly OpenAirInterface for supporting 5G-like experiments.

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

DIONYSOS Project-Team

5. New Software and Platforms

5.1. IPv6 Test Toolkit

FUNCTIONAL DESCRIPTION

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

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

- Participants: Cesar Pascal Viho
- Contact: Cesar Pascal Viho
- URL: <http://www.irisa.fr/tipi>

5.2. Passive Test Tool

- Participants: Cesar Pascal Viho
- Contact: Cesar Pascal Viho
- URL: <http://www.irisa.fr/tipi>

5.3. T3DevKit

SCIENTIFIC DESCRIPTION

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

FUNCTIONAL DESCRIPTION

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

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

5.4. Interoperability Assessment

Participant: César Viho.

Our experience in interoperability assessment (since 1996) and in using the TTCN-3 standard allowed us to develop a tool (called `ttproto`) that helps in: (i) experimenting new concepts for long term evolution of the TTCN-3 standard and (ii) facilitating new approaches and methods for interoperability assessment. For instance, new passive approaches that we developed have been implemented and validated using `ttproto`. This tool `ttproto` has been used to develop test suites for 6LoWPAN-ND (IPv6 for Low Power Networks) and CoAP (Constrained Application Protocol). The CoAP test suites have been successfully used for two Plugtest interoperability events organized by ETSI, IPSO Alliance and the FP7 PROBE-IT project. The tool `ttproto` and the test suites indicated above are freely available at <http://www.irisa.fr/tipi>.

5.5. 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 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 Team

6. New Software and Platforms

6.1. OpenWSN (Software)

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

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 2015:

- Development:
 - Moving the project from UC Berkeley to Inria (**Thomas Watteyne**)
 - Implementation of a layer-2 security based on AES-128 and CCM* (Malisa Vucinic)
 - Implementation of draft-ietf-6tisch-minimal (Tengfei Chang)
 - Implementation of draft-dujovne-6tisch-6top-sf0 (Tengfei Chang)
 - Implementation of draft-wang-6tisch-6top-sublayer (Tengfei Chang)
 - Creation of “Golden Image” used as a reference during interoperability testing (Tengfei Chang)
- Recognition:
 - OpenWSN was selected by ETSI as the 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**
OpenWSN & OpenMote: Hands-on Tutorial on Open Source Industrial IoT. Thomas Watteyne, Xavier Vilajosana, Pere Tuset. IEEE Global Telecommunications Conference (GLOBECOM), San Diego, CA, USA, 6-10 December 2015.
 - **Tutorial**
OpenWSN Tutorial [presented by Xavi Vilsajosana] Workshop Internet Of Things / Equipex FIT IoT-LAB, Lille, France, 15 October 2015.
 - **Hackathon**
OpenWSN/6TiSCH Hackathon, Czech Republic, 19 July 2015.
 - **Interop event**
First ETSI 6TiSCH plugtest (interop event) in Prague, Czech Republic, 17-18 July 2015.

6.2. OPERA and OCARI (Software)

Participants: Erwan Livolant, Pascale Minet.

The OPERA software was developed by the Hipercom2 team in the OCARI project (see <https://ocari.org/>). 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 2015, Erwan Livolant maintained the code and corrected some bugs.

6.3. CONNEXION (Software)

Participants: Ines Khoufi, Pascale Minet, Erwan Livolant.

These developments are part of the CONNEXION project. In 2015, Ines khoufi developed two softwares:

- a distributed algorithm, called OA-DVFA, to deploy autonomous and mobile sensor nodes to ensure full coverage of a 2-D area with unknown obstacles. It is based on virtual forces and virtual grid.
- an algorithm, called MRDS, to compute the tours of mobile robots in charge of placing static sensor nodes at the positions given (e.g. points of interest). This is a multi-objective optimization problem: to minimize the deployment duration, to balance the durations of robots tours and to minimize the number of robots used. A genetic heuristic is used to solve this problem.

With regard to the wireless sensor network OCARI, in 2015 we designed and developed the mobility support for OCARI. The solution proposed to support mobility in the OCARI network is simple and limits the overhead induced by mobile nodes. This mobility support is designed to be efficient in its use of resources (e.g. bandwidth, energy, memory). The properties of energy efficiency, determinism, latency and robustness provided by OCARI to static wireless sensor nodes are ensured. In the absence of mobile nodes, the OCARI network behaves exactly as without mobility support and exhibits exactly the same performances. Similarly, the overhead induced by mobile sinks is paid only if mobile sinks are present. Data gathering by the static sink, being the most important objective of the OCARI network from the application point of view, its performances are not altered by mobility support. Data gathering by a mobile sink is a new functionality provided by mobility support. This mobility support has been demonstrated with a mobile robot embedding a sensor node and transferring its data to the static sink via router nodes that depend on the position of the robot.

With Telecom ParisTech, the integration of OCARI in a Service-Oriented Architecture using the OPC-UA/ROSA middleware went on. More precisely, we developed the creation of services corresponding to newly available physical sensor measurements and the suppression of services that are no longer available.

Erwan Livolant developed an OCARI frame dissector plugin for Wireshark (<https://www.wireshark.org>) available from the Git repository at OCARI website (<https://www.ocari.org/gitlab/tools/wireshark.git>). 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.

6.4. SAHARA (Software)

Participants: Erwan Livolant, Pascale Minet.

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.

6.5. FIT IoT-LAB (Platform)

Participant: Thomas Watteyne.

Note well: IoT-lab is NOT strictly speaking 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 highlight Inria-EVA activity and contribution to the IoT-lab testbed in 2015.

- The Paris-Rocquencourt deployment has been stable throughout 2015.
- Thomas Watteyne and the OpenWSN community have been using the platform (all sites, not just Rocquencourt) extensively throughout 2015. Highlights include:
 - Nicola Accettura (then postdoc at UC Berkeley) created scripts to automate running OpenWSN on the IoT-lab, under the co-supervision of Thomas Watteyne and Prof. Kris Pister. Source code is available at <https://github.com/openwsn-berkeley/openwsn-on-iotlab>.
 - This work was presented during the OpenWSN hackathon held in conjunction with the IETF93 standardization meeting in Prague in July 2015.
 - Prof. Diego Dujovne from Universidad Diego Portales (Chile) visited Thomas Watteyne in July 2015 to work on the Mercator project (<https://github.com/openwsn-berkeley/mercator>) to collect Dense Wireless Connectivity Datasets for the IoT on the IoT-lab.
- The Inria-EVA team supported the IoT-lab admin team to remove malfunctioning batteries from the Inria-Rocquencourt deployment in December 2015.
- Thomas Watteyne integrated the IoT-lab admin team in December 2015. Together, they are working on a smaller test deployment with the Inria-EVA premises at Inria-Paris, on which development will be done to:
 - Allow commercial hardware to be plugged into the IoT-lab gateways.
 - Allow multiple motes to be plugged into the same IoT-lab gateway.
 - Use the IoT-lab for deploying and verifying the correct functioning of the OpenWSN implementation on all supported hardware board.
 - Use the IoT-lab for deploying and verifying the correct functioning of the OpenWSN implementation at small/medium/large scale.
- The activities above are lead by Tengfei Chang from the Inria-EVA team, under the supervision of Thomas Watteyne, and in close collaboration with the IoT-lab core team.

FUN Project-Team

6. New Software and Platforms

6.1. IoT-LAB robots

KEYWORDS: Internet of things - Robotics

FUNCTIONAL DESCRIPTION

IoT-LAB robots is an embedded robot controller on a Turtlebot2 providing the IoT-LAB node mobility functionality.

- Partner: Université de Strasbourg

6.2. FIT IoT-Lab

Participants: Raymond Borenstein, 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

6. New Software and Platforms

6.1. Svvpamp: Simulator of Various Voting Algorithms in Manipulating Populations

Svvpamp is a Python package dedicated to the study of voting systems with an emphasis on manipulation analysis. Svvpamp can generate datasets based on a large library of artificial models, or use any kind of real dataset as input. It currently implements more than 20 voting systems. Using state of the art algorithms, it can analyze multiple variants of tactical voting (e.g. absence of weak/strong Nash equilibrium). Svvpamp is free software, under the GNU General Public License version 3. Its documentation includes installation procedure, tutorials, reference guide and instructions for new contributors.

Svvpamp represents about ten thousands lines of code, and according to the Python Software Foundation, 2568 downloads have been reported in the last month (as of December, 10th, 2015). It is available at <https://svvpamp.readthedocs.org>.

Svvpamp [39] will be demonstrated in The Thirtieth Conference on Artificial Intelligence.

6.1.1. Svvpamp self-assessment

A3: ambitious software, usable by people inside and outside the team but without a clear and strong dissemination and support action plan. So3up4: original software reusing known ideas and introducing a few new ideas / original software implementing a fair number of original ideas.

SM3: well-developed software, fairly extensive documentation, reasonable software engineering and testing, attention to usability, dissemination, bug fixes, and user feedback;

EM2: basic maintenance to keep the software alive;

SDL4: public source or binary distribution on the web, organized by the development team;

François Durand is the main contributor (4) in: a) design and architecture (DA) b) coding and debugging (CD) c) maintenance and support (MS) d) team/project management (TPM)

6.2. Big Graph Tools

The team is starting a software development activity around big graph manipulation. A preliminary library offering diameter and skeleton computation is available at <https://who.rocq.inria.fr/Laurent.Viennot/dev/big-graph-tools/>. This library was used to compute the diameters of the worldwide road network (200M edges) and the Twitter follower-followee graph (23G edges).

INFINE Team

5. New Software and Platforms

5.1. RIOT

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

SCIENTIFIC DESCRIPTION

While requiring as low as 1,5kB of RAM and 5kB of ROM, the RIOT operating system offers both advanced capabilities such as real time and energy efficiency capabilities, as well as standard capabilities, such as a single API (partially POSIX compliant) across heterogeneous IoT hardware (8-bit, 16-bit and 32-bit low-power hardware). This API is developer-friendly in that it enables Linux-like developer experience which was not possible so far for embedded programming: multi-threading, standard C and C++ application programming and the use of standard debugging tools (gdb, valgrind etc.). On top of this, as expected from a modern OS, RIOT also provides standard connectivity capabilities, by means of several networks stacks that are readily available, 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 an API and SDK for energy-, memory- and CPU-constrained IoT devices and similar communicating embedded systems. This API builds upon standard systems concepts (e.g. POSIX) and standard communication protocols (e.g. IPv6). RIOT thus allows the development of applications that collect sensor data and transmit it to the cloud over the Internet, using end-to-end communication means e.g. IPv6 communication from sensor to cloud servers, that can be protected by transport layer security (e.g. DTLS). This data can then be used for smart energy management for example.

An important part of the design of RIOT is that it can easily be ported to different hardware devices (tens of types of heterogeneous IoT devices are already supported), and it can easily be extended to support the latest evolution of communication standards (a wide range of protocols are already supported).

For example, RIOT already enables tests and experiments of arbitrary IoT applications and protocols on FIT IoT-Lab, which provides a large-scale infrastructure facility with 2700 nodes for testing remotely small wireless devices. Once validated, the code for such applications and protocols can then run on any other IoT device hardware supported by RIOT.

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

5.2. GardiNet (previously known as DragonNet)

FUNCTIONAL DESCRIPTION

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

5.3. 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. Users' 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/>

MADYNES Project-Team

6. New Software and Platforms

6.1. Distem

KEYWORDS: Large scale - Experimentation - Virtualization - Emulation

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 targeting such environments.

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

6.2. Escape

KEYWORDS: Security - Web - Privacy - TLS (Transport Layer Security)

Escape is a Firefox web browser add-on that offers the ability to bypass HTTPS firewalls that filter websites based on the SNI value of the TLS connection. In addition, it can be used to bypass legacy filtering of DNS requests. The extension is implemented in JavaScript and is based on another security add-on named Convergence.

- Participants: Shbair Wazen, Thibault Cholez, Antoine Goichot and Isabelle Chrisment
- Contact: Thibault Cholez
- URL: <http://madynes.loria.fr/Research/Software#toc1>

6.3. Flowoid

KEYWORDS: Android - NetFlow - Monitoring

Flowoid is a flow based monitoring probe dedicated to Android environments. It uses the NetFlow protocol to send to a collector, information related to the network activities of running Android applications. The information about each connection are grouped into records containing traditional properties including source IP address, destination IP address, bytes, packets, etc. In addition, Flowoid associates and sends for each NetFlow record a set of information related to the geographic location of the device, the name of the application that established the connection, the state of the device screen (ON, OFF, locked, unlocked) and the type of the traffic (foreground, background).

- Participants: Abdelkader Lahmadi, Frédéric Beck, Julien Vaubourg and Olivier Festor
- Contact: Abdelkader Lahmadi

6.4. Grid'5000 testbed

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 Émile Morel
- Contact: Lucas Nussbaum
- URL: <https://www.grid5000.fr/>

6.5. Kadeploy

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 (see below), 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 - Université de Lorraine - Loria - Grid'5000 - Inria
- Contact: Emmanuel Jeanvoine
- URL: <http://kadeploy3.gforge.inria.fr>

6.6. MECSYCO suite

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

The MECSYCO suite (formerly AA4MM) is 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). It features MECSYCO-RE-java the Java implementation of the central part (core) and VisuAA4MM a visualization tool.

- Participants: Vincent Chevrier (Former LORIA Maia team), Laurent Ciarletta, Julien Siebert, Yannick Presse, Benjamin Segault, Benjamin Camus, Victorien Elvinger, Julien Vaubourg, Christine Bourjot, Benjamin Vouillaume and David Michel
- Partners: Université de Lorraine - Inria
- Contact: Vincent Chevrier

6.7. MPIGate

KEYWORDS: Internet of things - Home Automation - Health - Smart home - Home care - Connected object
MPIGate stands for Multi Protocol Interface GATEway for Tele-care, Environment Monitoring and Control. It is a set of softwares aiming at facilitating the development of both home automation and ambient assisted living applications thanks to the abstraction of heterogeneous sensor data and the facility of access to read and write functions over the devices plugged to the networks. This year, its evolution has mainly been carried out within SATELOR project and LAR project. Bluetooth Low Emission (BLE) has been integrated this year. It can be used by people working on home automation and ambient assisted living applications.

- Participants: Mandar Harshe and Ye-Qiong Song
- Contact: Yeqiong Song
- URL: <http://mpigate.loria.fr/>

6.8. Ruby-cute

KEYWORDS: Experimentation - HPC - Cloud

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

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

6.9. WISCAN

KEYWORDS: Network monitoring - Scanning

This tool allows to scan the entire IPv4 space in an efficient way. It relies on the Zmap (smap.io) while optimizing the randomness of the scanned IP addresses to avoid overloading destination networks and reduce the probability to scan two successive addresses of the same IP address block. Our approach to generate random permutation of IP addresses outperforms the approach used by Zmap in terms of the distribution of distance between successive IP addresses. Besides, our scan methodology can be distributed from multiple sources with few efforts.

- Contact: Jérôme François
- URL: <http://gforge.inria.fr/projects/wiscan>

6.10. XPFlow

XPFlow is an implementation of a new, workflow-inspired approach to control experiments involving large-scale computer installations. Such systems pose many difficult problems to researchers due to their complexity, their numerous constituents and scalability problems. The main idea of the approach consists in describing the experiment as a workflow and execute it using achievements of Business Process Management (BPM), workflow management techniques and scientific workflows.

- Participants: Tomasz Buchert and Lucas Nussbaum
- Contact: Lucas Nussbaum
- URL: <http://xpflow.gforge.inria.fr/>

6.11. Platforms

6.11.1. SCADA and IoT security assessment platform

Participants: Abdelkader Lahmadi [contact], Jérôme François, Olivier Festor.

SCADA *Supervisory Control and Data Acquisitions* refers to a centralized control and monitoring system for a variety of machinery and equipment involved with many industrial activities including: power generation and distribution, transportation, nuclear plants, manufacturing processes, etc. The most threaten accidents in SCADA networks are caused by targeted attacks, where adversaries exploit vulnerabilities available in software or network protocols components to disturb and make damage to the physical process. Therefore, it is important to provide new methods and tools for protecting SCADA networks from malicious cyber attacks targeting physical processes and infrastructures.

We are developing and maintaining a platform to assess and analyse the security of SCADA systems. The current version of the testbed combines real hardware Programmable Logic Controllers (PLCs) and simulation tools of physical processes. It also provides a set of tools that we have developed to capture and analyse control messages exchanged between a PLC and the physical processes. During the year 2015, we have received a regional funding to extend our platform with more devices and off-the-shelf solutions for home automation.

We have also extended the platform with IoT devices dedicated to home automation solutions (smart plugs, home boxes, lighting systems , door locks and detectors, etc). Our main goal is to rely on Software Defined Radio solution to evaluate the security of these devices and finding their communication protocol vulnerabilities.

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
- 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. Last, we have developed objects that allow to generate easily simulation scenarios.

- Participants: Sara Alouf, Abdulhalim Dandoush, Giovanni Neglia and Alina Tuholukova

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 Stéphane 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://muse.inria.fr/hostview/>

5.3. Online HoA

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

FUNCTIONAL DESCRIPTION

Implementation of HoA as collectd plugin for OpenWRT.

- 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: <http://muse.inria.fr/tagit>

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://muse.inria.fr/ucn>

5.6. WeBrowse

FUNCTIONAL DESCRIPTION

WeBrowse is the first passive crowdsourced 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 Dinechin, Nicolas Brunie, Matei Istioan and Antoine Martinet
- Partners: CNRS - ENS Lyon - UCBL Lyon 1 - UPVD
- Contact: Florent de Dinechin
- URL: <http://flopoco.gforge.inria.fr/>

5.2. 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>

5.3. 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>

5.4. WSNet

KEYWORD: Network simulator

FUNCTIONAL DESCRIPTION

The WSNNet-3.0 project objective is to develop the next evolution of the WSNNet simulator. It 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.

- Participants: Rodrigue Domga Komguem, Quentin Lampin, Alexandre Mouradian and Fabrice Valois
- Partner: CEA-LETI
- Contact: Fabrice Valois
- URL: <https://gforge.inria.fr/projects/wsnet-3/>

5.5. WiPlan

FUNCTIONAL DESCRIPTION

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

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 composed of four main parts: a Network Operations Center (NOC), a set of Embedded Communicating Object (ECO) test-beds, 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 on the 28th October 2014⁰, with the noticable venue of Vincent Poor, Dean of School of Engineering and Applied Science of Princeton University. Since that date, the platform is open to public experiments.

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: Patrice Raveneau, Hervé Rivano, Razvan Stanica.
- 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, Diala Naboulsi and Razvan Stanica.
- Contact: Marco Fiore
- URL: <http://kolntrace.project.citi-lab.fr/#download>

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